, , 1–38 (1995)

Finding All Solutions of Nonlinearly Constrained Systems of Equations

COSTAS D. MARANAS AND CHRISTODOULOS A. FLOUDAS*
Department of Chemical Engineering, Princeton University, Princeton, NJ 08544–5263

Editor:

Abstract. A new approach is proposed for finding all ϵ–feasible solutions for certain classes of nonlinearly constrained systems of equations. By introducing slack variables, the initial problem is transformed into a global optimization problem **(P)** whose multiple global minimum solutions with a zero objective value (if any) correspond to all solutions of the initial constrained system of equalities. All ϵ–globally optimal points of **(P)** are then localized within a set of arbitrarily small disjoint rectangles. This is based on a branch and bound type global optimization algorithm which attains finite ϵ–convergence to each of the multiple global minima of **(P)** through the successive refinement of a convex relaxation of the feasible region and the subsequent solution of a series of nonlinear convex optimization problems. Based on the form of the participating functions, a number of techniques for constructing this convex relaxation are proposed. By taking advantage of the properties of products of univariate functions, customized convex lower bounding functions are introduced for a large number of expressions that are or can be transformed into products of univariate functions. Alternative convex relaxation procedures involve either the difference of two convex functions employed in αBB [23] or the exponential variable transformation based underestimators employed for generalized geometric programming problems [24]. The proposed approach is illustrated with several test problems. For some of these problems additional solutions are identified that existing methods failed to locate.

Keywords: Global optimization, nonlinear systems of equations, all solutions

1. Introduction

A fundamental task in applied mathematics, engineering and sciences is finding all solutions of a set of equations. This task is sometimes further complicated by requiring the simultaneous satisfaction of a number of inequality and/or variable bound constraints. Not only the problem of computing all solutions of nonlinearly constrained systems of equations is NP–hard, but it is also possible that there exists exponentially many such solutions [1]. In addition, simply checking if a solution exists is NP–hard [2]. There exists a large body of literature on methods for solving systems of equations. These methods fall within the following three broad classes: (i) Newton and quasi–Newton type methods; (ii) homotopy continuation type methods; and (iii) interval–Newton methods.

Newton and quasi–Newton type methods and their modifications achieve superlinear convergence only when they are well within the neighborhood of the solution. However, these methods are likely to fail if the initial guess is poor, or if singular points are encountered. Modifications in an attempt to avoid singularities may incorporate trust–region techniques such as Powell's "dogleg" method [31], steepest descent direction information [7], [10], [26] and alterations on the quasi–Newton Jacobian estimates [30]. This type of methods,

* Author to whom correspondence should be addressed.

although very computationally efficient, cannot provide guarantees for convergence. This is manifested in practice with their poor convergence characteristics.

One of the most widely used method for locating solutions of nonlinear systems of equations belongs to the broad class of *embedding methods*. This class of methods are also known as *continuation, homotopy continuation, or incremental loading*, and are based on the pioneering work of [19], [20], [8], [18]. The basic idea of homotopy continuation methods is to create a family of a single parameter functions so that the solution for (t=0) is known and then solve a sequence of problems with t steadily increasing from (t=0) to (t=1) using the solution of one problem as an estimate for the next. A popular variation is to use a system variable as the continuation parameter and integrate the resulting system of ordinary differential equations towards steady–state by utilizing AUTO [9]. A problem common to all homotopy variants is that variable bounds and inequality constraints cannot be handled directly. A comprehensive review of the extensive literature in this area can be found in [12]. While in practice homotopy continuation methods are frequently used in an attempt to locate all solutions of arbitrary nonlinear systems of equations, mathematical guarantees that all solutions will be found exist only in special cases (e.g. polynomial systems with no constraints). For polynomial systems of equations, however, Morgan [27] proposed a differential arclength continuation using a special homotopy that establishes a number of continuation paths guaranteed to converge to all possible real and complex roots. Two popular software packages, CONSOL [27] and POLSYS [34] have implemented this method.

Interval–Newton methods can find rectangles containing all solutions of nonlinear systems of equations within certain variable bounds with mathematical certainty. They do so by applying the classical Newton-like iterative methods on interval variables rather than variables coupled with a generalized bisection strategy [29], [13]. A version of the basic Interval–Newton method has been implemented into the public domain software program INTBIS [17] which is coupled with a portable interval standard function library INTLIB [16]. The main attractive feature of Interval–Newton methods is that they provide mathematical guarantees for convergence to all solutions of fairly arbitrary nonlinear systems of equations within certain variable bounds. However, this wide applicability to almost arbitrary nonlinear functions comes at an expense. Because no specific structure of individual expressions is analyzed the obtained interval bounds can sometimes be fairly loose.

The proposed approach is based on convex lower bounding coupled with a partitioning strategy and like Interval-Newton methods, it can provide guarantees for convergence to all ϵ–solutions. The fundamental difference, however, between our procedure and Interval–Newton methods is that while the former utilizes a single value to lower bound functions within rectangular domains, we lower bound nonconvex functions with convex functions. By exploiting the mathematical structure of the problem, this typically results in much tighter bounds. In the next section, a description of the problem is presented.

2. Problem Description

This paper addresses the problem of identifying all solutions of a nonlinear system of equations subject to inequality constraints and variable bounds and is formulated as:

$$\begin{aligned} h_j(\mathbf{x}) &= 0, \quad j \in \mathcal{N}_E \\ g_k(\mathbf{x}) &\leq 0, \quad k \in \mathcal{N}_I \\ \mathbf{x}^L &\leq \mathbf{x} \leq \mathbf{x}^U, \end{aligned} \qquad \textbf{(S)}$$

where $\mathcal{N}_E$ is the set of equalities, $\mathcal{N}_I$ the set of inequality constraints, and $\mathbf{x}$ the vector of variables. Note that in formulation **(S)** the total number of variables is allowed to be different than the total number of equalities so as neither the existence nor the uniqueness of a solution of **(S)** is postulated. Therefore, both overspecified and underspecified systems are included in the present investigations. Note that a number of important problems naturally arise as special instances of formulation **(S)**. On one hand, by omitting all inequality constraints, **(S)** corresponds to a system of nonlinear equations. On the other hand, by eliminating all equality constraints **(S)** checks the existence of feasible points for the given inequality constraint set (feasibility problem).

Formulation **(S)** can be transformed into the following min–max optimization problem [15]

$$\begin{aligned} &\min_{\mathbf{x}} \max_{j \in \mathcal{N}_E} \quad |h_j(\mathbf{x})| \\ &\text{subject to} \quad g_k(\mathbf{x}) \leq 0, \quad k \in \mathcal{N}_I \\ &\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U. \end{aligned}$$

By introducing a single slack variable s, the min-max problem can be written as the following optimization problem **(P0)**.

$$\min_{\mathbf{x}, s \geq 0} \quad s \qquad \textbf{(P0)}$$

$$\begin{aligned} \text{subject to} \quad h_j(\mathbf{x}) - s &\leq 0, \quad j \in \mathcal{N}_E \\ -h_j(\mathbf{x}) - s &\leq 0, \quad j \in \mathcal{N}_E \\ g_k(\mathbf{x}) &\leq 0, \quad k \in \mathcal{N}_I \\ \mathbf{x}^L \leq \mathbf{x} &\leq \mathbf{x}^U \end{aligned}$$

Clearly, there is a one to one correspondence between multiple global minima $(\mathbf{x}^*, s^*)$ of **(P0)** for which $s^* = 0$ and solutions of **(S)**. This means that if the global minimum of **(P0)** involves a nonzero slack variable s^* then the original problem **(S)** has no solutions. Note that, unless the functions $h_j(\mathbf{x})$ and $g_k(\mathbf{x})$ are *linear* and *convex* respectively, formulation **(P)** corresponds to a *nonconvex* optimization problem. This implies that if a local optimization approach is used to solve **(P0)**, one might miss some of the multiple global minima of **(P0)** or even erroneously deduce that there are no solutions for **(S)**. Therefore, an approach that is guaranteed to always locate *all* multiple global minima of **(P0)** appears to be necessary for solving **(S)** so that (i) the correct solution vector $(\mathbf{x}^*, s^*)$ is identified and (ii) all solutions $(\mathbf{x}^*)$ of **(S)** with $s^* = 0$ are found in all instances. In this work, a deterministic global optimization is proposed which is guaranteed to locate all ϵ–global minima of $(\mathbf{P0})$ through the successive refinement of converging lower and upper bounds on the solution based on the solution of convex optimization problems defined by a branch and bound approach. A lower bound on the solution of **(P0)** is found by first replacing each nonconvex constraint in **(P0)** with a convex underestimation of it and then finding the solution of the convex relaxation **(R)** of **(P0)** with commercially available solver such as MINOS5.4 [28] as shown in [22] and [23]. This approach naturally partitions the constraints of formulation **(P0)** into convex (for which no relaxation is required) and nonconvex constraints. This partitioning yields the following alternative formulation **(P)**:

$$\min_{\mathbf{x}, s \geq 0} \quad s \qquad \textbf{(P)}$$

$$\begin{aligned}
\text{subject to} \quad h_j^{nonc}(\mathbf{x}) - s &\leq 0, \;\; j \in \mathcal{N}_{noncE} \\
-h_j^{nonc}(\mathbf{x}) - s &\leq 0, \;\; j \in \mathcal{N}_{noncE} \\
g_k^{nonc}(\mathbf{x}) &\leq 0, \; k \in \mathcal{N}_{noncI} \\
h_j^{lin}(\mathbf{x}) &= 0, \; j \in \mathcal{N}_{linE} \\
g_k^{conv}(\mathbf{x}) &\leq 0, \; j \in \mathcal{N}_{convI} \\
\mathbf{x}^L \leq \mathbf{x} &\leq \mathbf{x}^U
\end{aligned}$$

Here $\mathcal{N}_{noncE}, \mathcal{N}_{linE}$ are the sets of nonconvex and linear equality constraints respectively, and $\mathcal{N}_{noncI}, \mathcal{N}_{convI}$ are the sets of nonconvex and convex inequality constraints,

$$\mathcal{N}_E = \mathcal{N}_{noncE} \cup \mathcal{N}_{linE}, \;\; \mathcal{N}_I = \mathcal{N}_{noncI} \cup \mathcal{N}_{convI}.$$

A convex relaxation **(R)** of **(P)** of the form,

$$\min_{\mathbf{x}, s \geq 0} \quad s \qquad \textbf{(R)}$$

$$\begin{aligned}
\text{subject to} \quad \hat{h}_{+,j}^{nonc}(\mathbf{x}) - s &\leq 0, \;\; j \in \mathcal{N}_{noncE} \\
\hat{h}_{-,j}^{nonc}(\mathbf{x}) - s &\leq 0, \;\; j \in \mathcal{N}_{noncE} \\
\hat{g}_k^{nconc}(\mathbf{x}) &\leq 0, \; k \in \mathcal{N}_{noncI} \\
h_j^{lin}(\mathbf{x}) &= 0, \; j \in \mathcal{N}_{linE} \\
g_k^{conv}(\mathbf{x}) &\leq 0, \; j \in \mathcal{N}_{convI}
\end{aligned}$$

$$\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U$$

can be obtained by replacing the original nonconvex functions $h_j^{nonc}(\mathbf{x})$,$-h_j^{nonc}(\mathbf{x})$,$g_k^{nonc}(\mathbf{x})$ with some convex tight lower bounding functions $\hat{h}_{+,j}^{nonc}(\mathbf{x})$,$\hat{h}_{-,j}^{nonc}(\mathbf{x})$, $\hat{g}_k^{nonc}(\mathbf{x})$. These lower bounding functions $\hat{h}_{+,j}^{nonc}(\mathbf{x})$,$\hat{h}_{-,j}^{nonc}(\mathbf{x})$, $\hat{g}_k^{nonc}(\mathbf{x})$ must be (i) convex in $[\mathbf{x}^L, \mathbf{x}^U]$; (ii) valid underestimators of the original functions h_j^{nonc}, $-h_j^{nonc}$, g_k^{nonc}; and (iii) for every point $\mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U]$ the maximum separation between the original functions and the convex underestimators must become arbitrarily ϵ small by appropriately reducing the size of the rectangular domain $[\mathbf{x}^l, \mathbf{x}^u]$ around the point $\mathbf{x}$ inside which the convex underestimators are defined. These requirements are expressed mathematically as follows:

Property 1: $\hat{h}_{+,j}^{nonc}(\mathbf{x})$, $\hat{h}_{-,j}^{nonc}(\mathbf{x})$, and $\hat{g}_k^{nonc}(\mathbf{x})$ be convex $\forall\ \mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U]$.

Property 2: $h_j^{nonc}(\mathbf{x}) \geq \hat{h}_{+,j}^{nonc}(\mathbf{x})$, $-h_j^{nonc}(\mathbf{x}) \geq \hat{h}_{-,j}^{nonc}(\mathbf{x})$, and $g_k^{nonc}(\mathbf{x}) \geq \hat{g}_k^{nonc}(\mathbf{x})$, $\forall\ \mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U]$.

Property 3: $\forall\ \mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U]$ and $\epsilon > 0$, $\exists\ [\mathbf{x}^l, \mathbf{x}^u] \subseteq [\mathbf{x}^L, \mathbf{x}^U]$ with $\delta(\epsilon) = ||\mathbf{x}^u - \mathbf{x}^l||_2^{1/2} > 0$ such that

$$\max_{\mathbf{x} \in [\mathbf{x}^l, \mathbf{x}^u]} \left(h_j^{nonc}(\mathbf{x}) - \hat{h}_{+,j}^{nonc}(\mathbf{x}) \right) < \epsilon,$$

$$\max_{\mathbf{x} \in [\mathbf{x}^l, \mathbf{x}^u]} \left(-h_j^{nonc}(\mathbf{x}) - \hat{h}_{-,j}^{nonc}(\mathbf{x}) \right) < \epsilon,$$

$$\max_{\mathbf{x} \in [\mathbf{x}^l, \mathbf{x}^u]} \left(g_k^{nonc}(\mathbf{x}) - \hat{g}_k^{nonc}(\mathbf{x}) \right) < \epsilon.$$

Property 3 requires that the maximum separation between the nonconvex function and its tight convex lower bounding function, defined inside some rectangular region, must go to zero

$$\lim_{\delta \rightarrow 0^+} \epsilon = 0.$$

as the size of the rectangular domain approaches zero ($\delta = 0$). This is important for proving finite ϵ–convergence. The order $\mathcal{O}(\epsilon) = \mathcal{O}(\delta^n)$ with which ϵ approaches zero as δ goes to zero is important because it determines the speed of convergence. Clearly, the largest possible value for n is desirable so as the maximum tolerance reaches an arbitrary value ϵ for a not too small variable range δ. For example, if the maximum separation ϵ goes as δ^2 then a value of just $\delta = 0.01$ suffices to meet a convergence tolerance of $\epsilon = 0.0001$.

An efficient convex lower bounding of nonconvex functionals appearing in formulation **(P)** is clearly central to the design of the proposed global optimization approach for locating all solutions. Undoubtedly, the tighter the convex lower bounding is the better the quality of the obtained lower bounds will be, and consequently the faster the algorithm will converge. The tightest possible convex lower bounding function for any arbitrary nonconvex function $f(\mathbf{x})$ inside some rectangular region P is called the *convex envelope* $\phi(\mathbf{x})$ of $f(\mathbf{x})$, and it must conform to the following properties [15]:

(i) $\phi(\mathbf{x})$ convex for all $\mathbf{x} \in P$.

(ii) $f(\mathbf{x}) \geq \phi(\mathbf{x})$ for all $\mathbf{x} \in P$.

(iii) For all functions $g(\mathbf{x})$ that satisfy (i) and (ii), $\phi(\mathbf{x}) \geq g(\mathbf{x})$ for all $\mathbf{x} \in P$.

Unfortunately, in all but the simplest cases there exists no method for deriving the convex envelope for arbitrary functions defined inside arbitrary domains. As a result, the focus in this work is to identify the maximum possible function which satisfies properties (i) and (ii). There exists a number of techniques for obtaining functions that satisfy properties (i), (ii). In the following sections, a number of convex lower bounding procedures are discussed which can be of use not only for the problem of locating all multiple solutions but also for any deterministic branch and bound global optimization algorithm based on convex lower bounding. The first convex lower bounding technique is motivated by the fact that a large number of nonconvex terms appearing in different models are or can be transformed into the product of functions of a single variable (univariate functions). By exploiting the properties of products of univariate functions, tight convex lower bounding functions are derived in the next section.

3. Products of univariate functions

A function $f : \mathcal{R} \rightarrow \mathcal{R}$ of a single variable x is called univariate function. Products of univariate functions f_i,

$$f(\mathbf{x}) \;=\; \prod_{i=1}^{N} f_i(x_i)$$

are in general nonconvex functions even if the corresponding univariate functions are convex. By utilizing appropriate linear transformations, if necessary, a large number of nonlinearities appearing in applied mathematics and engineering problems can be described as products of univariate functions.

Al–Khayyal and Falk [3] showed that the nonconvex bilinear product of xy inside the rectangular domain $\left[x^L, x^U\right] \times \left[y^L, y^U\right]$ can be tightly convex lower bounded by the following linear cut:

$$\max\left(x^L y + x y^L - x^L y^L,\; x^U y + x y^U - x^U y^U\right)$$

First, the conditions under which a similar result holds for the product of two arbitrary univariate functions $f(x)$ and $g(y)$ are investigated.

THEOREM 1 *If f, g are twice differentiable univariate functions $f(x), g(y) \in \mathcal{C}^2$ defined inside a rectangle $\left[\left(x^L, x^U\right), \left(y^L, y^U\right)\right]$ and*

$$l(x,y) = \max \left\{ \begin{array}{l} \phi\left(f^L g(y)\right) + \phi\left(g^L f(x)\right) - f^L g^L, \\ \phi\left(f^U g(y)\right) + \phi\left(g^U f(x)\right) - f^U g^U \right\} \end{array}\right.$$

$$\begin{aligned} \textit{where } f^L &= \inf_{x^L \le x \le x^U} f(x), \\ f^U &= \sup_{x^L \le x \le x^U} f(x), \\ g^L &= \inf_{x^L \le x \le x^U} g(x), \\ g^U &= \sup_{x^L \le x \le x^U} g(x) \end{aligned}$$

and $\phi\left(f^L g(y)\right), \phi\left(g^L f(x)\right), \phi\left(f^U g(y)\right),$ and $\phi\left(g^U f(x)\right)$ are the convex envelopes of the univariate functions $f^L g(y), g^L f(x), f^U g(y),$ and $g^U f(x)$ respectively then:

(i) $l(x,y)$ *is convex,* $\forall\ (x,y) \in\ \left[x^L, x^U\right] \times \left[y^L, y^U\right]$.

(ii) $f(x)g(y)\ \ge\ l(x,y),\ \forall\ (x,y) \in\ \left[x^L, x^U\right] \times \left[y^L, y^U\right]$.

Proof: Both functions $\phi\left(f^L g(y)\right) + \phi\left(g^L f(x)\right) - f^L g^L$ and $\phi\left(f^U g(y)\right) + \phi\left(g^U f(x)\right) - f^U g^U$ are convex as the sum of the convex envelopes of univariate functions. Since the maximum of two convex functions is a convex function as well, statement (i) is true and $l(x,y)$ is convex for all (x,y) in $\left[x^L, x^U\right] \times \left[y^L, y^U\right]$

Because,

$$f(x) - f^L \ge 0,\ \forall\ x\ \in\ \left[x^L, x^U\right] \quad \text{and} \quad g(y) - g^L \ge 0,\ \forall\ y\ \in\ \left[y^L, y^U\right]$$

we have

$$\left(f(x) - f^L\right)\left(g(y) - g^L\right)\ \ge\ 0,\ \forall\ (x,y)\ \in\ \left[x^L, x^U\right] \times \left[y^L, y^U\right].$$

After rearranging terms we obtain,

$$f(x)g(y)\ \ge\ f^L g(y) + f(x)g^L - f^L g^L,\ \ \forall\ (x,y) \in\ \left[x^L, x^U\right] \times \left[y^L, y^U\right].$$

By the definition of the convex envelope we know that,

$$\begin{aligned} f^L g(y)\ &\ge\ \phi\big(f^L g(y)\big),\ \forall\ y\ \in\ \left[y^L, y^U\right] \\ g^L f(x)\ &\ge\ \phi\big(g^L f(x)\big),\ \forall\ x\ \in\ \left[x^L, x^U\right] \end{aligned}$$

Therefore,

$$f(x)g(y)\ \ge\ \phi\big(f^L g(y)\big) + \phi\big(f(x)g^L\big) - f^L g^L,\ \forall\ (x,y) \in\ \left[x^L, x^U\right] \times \left[y^L, y^U\right]. \tag{1}$$

Furthermore, by following the same line of reasoning on the relation,

$$\left(f(x) - f^U\right)\left(g(y) - g^U\right) \;\geq\; 0, \; \forall\,(x,y) \in \left[x^L, x^U\right] \times \left[y^L, y^U\right],$$

we obtain:

$$f(x)g(y) \;\geq\; \phi\big(f^U g(y)\big) + \phi\big(g^U f(x)\big) - f^U g^U, \; \forall\,(x,y) \;\in\; \left[x^L, x^U\right] \times \left[y^L, y^U\right]. \tag{2}$$

Relations (1),(2) imply that statement (ii) is true. ■

Based on Theorem (1) function $l(x,y)$ can be utilized as a tight convex lower bounding function of the product of two continuous and twice differentiable functions. It can also be shown that under certain conditions $l(x,y)$ corresponds to the actual convex envelope of the product $f(x)g(y)$. These conditions are stated in the following theorem:

THEOREM 2 *If (i) the univariate functions,*

$$g^L f(x),\; g^U f(x) \textit{ and } f^L g(y),\; f^U g(y)$$

are concave in $\left[x^L, x^U\right]$ *and* $\left[y^L, y^U\right]$ *respectively and (ii) the functions* $f(x), g(y)$ *are monotonic, then* $l(x,y)$ *is the convex envelope of* $f(x)g(y)$*.*

$$\phi(f(x)g(y)) \;=\; l(x,y), \;\; \forall\,(x,y) \;\in\; \left[x^L, x^U\right] \times \left[y^L, y^U\right].$$

Proof: Theorem (1) proves that function $l(x,y)$ conforms with Properties (i) and (ii) of section 3. Therefore, it remains to show that it satisfies Property (iii) of section 3. Because the convex envelope of a univariate concave function defined in an interval is the line segment connecting the the two end points we have:

$$\begin{aligned}
\phi\big(g^L f(x)\big) &= g^L \left[\frac{f(x^U) - f(x^L)}{x^U - x^L} x + \frac{x^U f(x^L) - x^L f(x^U)}{x^U - x^L}\right] \\
\phi\big(g^U f(x)\big) &= g^U \left[\frac{f(x^U) - f(x^L)}{x^U - x^L} x + \frac{x^U f(x^L) - x^L f(x^U)}{x^U - x^L}\right] \\
\phi\big(f^L g(y)\big) &= f^L \left[\frac{g(y^U) - g(y^L)}{y^U - y^L} x + \frac{y^U g(y^L) - y^L g(y^U)}{y^U - y^L}\right] \\
\phi\big(f^U g(y)\big) &= f^U \left[\frac{g(y^U) - g(y^L)}{y^U - y^L} x + \frac{y^U g(y^L) - y^L g(y^U)}{y^U - y^L}\right]
\end{aligned}$$

After substituting these expressions into the relation for $l(x,y)$ we obtain:

$$l(x,y) \;=\; \max\left(l_1(x,y), l_2(x,y)\right)$$

$$\text{where } l_1(x,y) \;=\; \left[g^L \frac{f(x^U) - f(x^L)}{x^U - x^L}\right] x \;+\; \left[f^L \frac{g(y^U) - g(y^L)}{y^U - y^L}\right] y$$

$$+ \left[g^L \frac{x^U f(x^L) - x^L f(x^U)}{x^U - x^L} + f^L \frac{y^U g(y^L) - y^L g(y^U)}{y^U - y^L} - f^L g^L \right]$$

$$\begin{aligned} l_2(x,y) &= \left[g^U \frac{f(x^U) - f(x^L)}{x^U - x^L} \right] x + \left[f^U \frac{g(y^U) - g(y^L)}{y^U - y^L} \right] y \\ &+ \left[g^U \frac{x^U f(x^L) - x^L f(x^U)}{x^U - x^L} + f^U \frac{y^U g(y^L) - y^L g(y^U)}{y^U - y^L} - f^U g^U \right] \end{aligned}$$

Because $f(x), g(y)$ are monotonic one of the following alternatives is true:

$$\begin{array}{lllll} (a) & f(x^L) = f^L, & f(x^U) = f^U, & g(x^L) = g^L, & g(x^U) = g^U \\ (b) & f(x^L) = f^L, & f(x^U) = f^U, & g(x^L) = g^U, & g(x^U) = g^L \\ (c) & f(x^L) = f^U, & f(x^U) = f^L, & g(x^L) = g^L, & g(x^U) = g^U \\ (d) & f(x^L) = f^U, & f(x^U) = f^L, & g(x^L) = g^U, & g(x^U) = g^L \end{array}$$

Assuming that (a) is true we have:

$$\begin{array}{lcl} l_1(x^L, y^L) = f(x^L)g(y^L) & & l_2(x^U, y^U) = f(x^U)g(y^U) \\ l_1(x^L, y^U) = f(x^L)g(y^U) & \text{and} & l_2(x^L, y^U) = f(x^L)g(y^U) \\ l_1(x^U, y^L) = f(x^U)g(y^L) & & l_2(x^U, y^L) = f(x^U)g(y^L) \end{array}$$

This implies that one can partition the original rectangle

$$\mathcal{R} = \left[(x^L, y^L), (x^L, y^U), (x^U, y^L), (x^U, y^U) \right]$$

into the following two disjoint triangles,

$$\mathcal{T}_1 = \left[(x^L, y^L), (x^L, y^U), (x^U, y^L) \right], \text{ and } \mathcal{T}_2 = \left[(x^U, y^U), (x^L, y^U), (x^U, y^L) \right]$$

at whose vertices the linear functions $l_1(x,y), l_2(x,y)$ match the original product of univariate functions $f(x)g(y)$ respectively (See Figure 1). If $l(x,y)$ were not the convex envelope of $f(x)g(y)$ over the rectangular domain $\mathcal{R}$ then, there would be a third convex function $l_3(x,y)$ underestimating $f(x)g(y)$ over $\mathcal{R}$ and a point $(\bar{x}, \bar{y}) \in \mathcal{R}$ such that:

$$l(\bar{x}, \bar{y}) < l_3(\bar{x}, \bar{y})$$

Suppose that $(\bar{x}, \bar{y}) \in \mathcal{T}_1$. Then $(\bar{x}, \bar{y})$ is a unique convex combination of the three extreme points v^1, v^2, v^3 of $\mathcal{T}_1$. Hence, for the affine function l, there exists unique positive $\lambda_1, \lambda_2, \lambda_3$ satisfying $\sum_{i=1}^{3} \lambda_i = 1$ such that

$$l(\bar{x}, \bar{y}) = l\left(\sum_{i=1}^{3} \lambda_i v^i \right) = \sum_{i=1}^{3} \lambda_i l(v^i).$$

Because l_3 is the convex envelope of $f(x)g(y)$ inside $\mathcal{R}$, (i) l_3 is convex and (ii) it matches $f(x)g(y)$ at all vertex points like $l(x,y)$ does which implies:

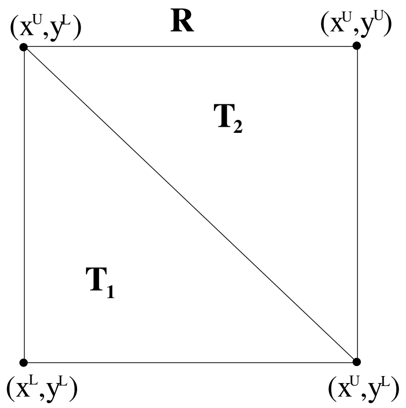

Figure 1. Decomposition of rectangle R into two triangles $\mathcal{T}_1, \mathcal{T}_2$.

$$l_3(\bar{x},\bar{y}) \;=\; l_3\left(\sum_{i=1}^{3}\lambda_i v^i\right) \;\leq\; \sum_{i=1}^{3}\lambda_i l_3(v^i) \;=\; \sum_{i=1}^{3}\lambda_i l(v^i) \;=\; l(\bar{x},\bar{y}).$$

This contradicts the initial hypothesis $l(\bar{x},\bar{y}) < l_3(\bar{x},\bar{y})$ and therefore, $l(x,y)$ is indeed the convex envelope of $f(x)g(y)$ in $\mathcal{R}$. Note that a similar argument holds if $(\bar{x},\bar{y}) \in \mathcal{T}_\in$. Moreover, depending on which monotonicity combination (a), (b) ,(c) or (d) is true it is always possible to partition $\mathcal{R}$ into two triangles $\mathcal{T}_1, \mathcal{T}_2$ by halving along one of the diagonals. Therefore, by following the same line of thought for combinations (b), (c) and (d) it is straightforward to extent this proof for all monotonicity combinations. ■

The analysis for the convex lower bounding of products of two univariate functions can be extended to accommodate the product of N univariate functions. This is accomplished by successively convex lower bounding pairs of univariate functions in a recursive manner until no pairs are left. One of the possible alternatives of combining pairs is to start with convex lower bounding the last two functions of the product and work your way to the front of the expression. Theorem (3) states that this procedure yields a convex lower bounding function for the initial product.

THEOREM 3 *If* $f_i \in \mathcal{C}^2 : \left[x_i^L, x_i^U\right] \rightarrow \mathcal{R}^+,\; i = 1,\ldots,N$ *and*

$$L(\mathbf{x}) \;=\; y_0$$

$$\begin{aligned}
\textit{where} \quad y_j &= \max\left\{\phi\left(f_{j+1}^L y_{j+1}\right) + \phi\left(y_{j+1}^L f_{j+1}(x_{j+1})\right) - y_{j+1}^L f_{j+1}^L,\right.\\
&\qquad \left.\phi\left(f_{j+1}^U y_{j+1}\right) + \phi\left(y_{j+1}^U f_{j+1}(x_{j+1})\right) - y_{j+1}^U f_{j+1}^U,\right\},\\
&\qquad\qquad j = 0, \ldots, N-3\\
y_{N-2} &= \max\left\{\phi\left(f_{N-1}^L f_N(x_N)\right) + \phi\left(f_N^L f_{N-1}(x_{N-1})\right) - f_{N-1}^L f_N^L,\right.\\
&\qquad \left.\phi\left(f_{N-1}^U f_N(x_N)\right) + \phi\left(f_N^U f_{N-1}(x_{N-1})\right) - f_{N-1}^U f_N^U\right\}
\end{aligned}$$

$$\begin{aligned}
\textit{and} \quad y_j^{L/U} &= \operatorname*{inf / sup}_{x_{j+1},\, y_{j+1}} \left\{\phi\left(f_{j+1}^L y_{j+1}\right) + \phi\left(y_{j+1}^L f_{j+1}(x_{j+1})\right) - y_{j+1}^L f_{j+1}^L,\right.\\
&\qquad \left.\phi\left(f_{j+1}^U y_{j+1}\right) + \phi\left(y_{j+1}^U f_{j+1}(x_{j+1})\right) - y_{j+1}^U f_{j+1}^U,\right\},\\
&\qquad\qquad j = 0, \ldots, N-3\\
y_{N-2}^{L/U} &= \operatorname*{inf / \mathit{sup}}_{x_N,\, x_{N-1}} \left\{\phi\left(f_{N-1}^L f_N(x_N)\right) + \phi\left(f_N^L f_{N-1}(x_{N-1})\right) - f_{N-1}^L f_N^L,\right.\\
&\qquad \left.\phi\left(f_{N-1}^U f_N(x_N)\right) + \phi\left(f_N^U f_{N-1}(x_{N-1})\right) - f_{N-1}^U f_N^U\right\}
\end{aligned}$$

then

(i) $L(\mathbf{x})$ *is convex,* $\forall\, \mathbf{x} \in \left[\mathbf{x}^L, \mathbf{x}^U\right]$.

(ii) $\prod_{i=1}^{N} f_i(x_i) \geq L(\mathbf{x}), \quad \forall\, \mathbf{x} \in \left[\mathbf{x}^L, \mathbf{x}^U\right]$.

Proof: Starting from the beginning of the recursive definition of $L(\mathbf{x})$, y_{N-2} is a convex function of (x_{N-1}, x_N) as the max of two convex functions. For the same reason y_{N-3} is a convex function of (x_{N-2}, y_{N-2}) or otherwise of (x_{N-2}, x_{N-1}, x_N). By recursively substituting y_j into the expression for y_{j-1} we deduce that for every $j = 0, \ldots, N-2$, y_j is a convex function of $(x_{j+1}, x_{j+2}, \ldots, x_N)$. Therefore, $L(\mathbf{x}) = y_0$ is a convex function of $(x_1, x_2, \ldots, x_N)$ which proves part (ii) of Theorem (3).

From Theorem (1) and the statement of Theorem (3) we have,

$$\begin{aligned}
f_{N-1}(x_{N-1}) f_N(x_N) &\geq \max\left\{\phi\left(f_{N-1}^L f_N(x_N)\right) + \phi\left(f_N^L f_{N-1}(x_{N-1})\right) - f_{N-1}^L f_N^L,\right.\\
&\qquad \left.\phi\left(f_{N-1}^U f_N(x_N)\right) + \phi\left(f_N^U f_{N-1}(x_{N-1})\right) - f_{N-1}^U f_N^U\right\}\\
&= y_{N-2}
\end{aligned}$$

$$\begin{aligned}
\text{and} \quad f_{j+1}(x_{j+1}) y_{j+1} &\geq \max\left\{\phi\left(f_{j+1}^L y_{j+1}\right) + \phi\left(y_{j+1}^L f_{j+1}(x_{j+1})\right) - y_{j+1}^L f_{j+1}^L,\right.\\
&\qquad \left.\phi\left(f_{j+1}^U y_{j+1}\right) + \phi\left(y_{j+1}^U f_{j+1}(x_{j+1})\right) - y_{j+1}^U f_{j+1}^U,\right\}\\
&= y_j,\\
&\qquad\qquad j = 0, \ldots, N-3
\end{aligned}$$

By combining these last two sets of inequalities we have,

$$\prod_{i=1}^{N} f_i(x_i) \;\geq\; y_0 \;=\; L(\mathbf{x}),\; \forall\, \mathbf{x} \;\in\; \left[\mathbf{x}^L, \mathbf{x}^U\right]$$

which proves part (ii) of Theorem (3). ■

Theorem (3) describes one possible way of recursively combining pairs of univariate functions. Theorem (4) states who many of these alternative sequences exist for convex lower bounding the product of N univariate functions.

THEOREM 4 *There are,*

$$\frac{(N!)^2}{N 2^{N-1}},$$

ways of combining pairs of univariate functions in a product of N univariate functions.

Proof: Clearly, there exists $\binom{N}{2} = \frac{N(N-1)}{2}$ ways of selecting the first pair of univariate functions to be convex lower bound. After this action, we are left with $N-1$ functions which implies that there are $\binom{N-1}{2} = \frac{(N-1)(N-2)}{2}$ alternatives for picking the next pair of functions. This recursive convex lower bounding is continued until we are left with only a pair of functions involving a single convex lower bounding alternative. Because every convex lower bounding stage is independent of the previous one, the total number of ways of combining pairs the N univariate functions in pairs of two is:

$$\prod_{i=2}^{N} \binom{i}{2} \;=\; \prod_{i=2}^{N} \frac{i\,(i-1)}{2} \;=\; \frac{N!(N-1)!}{2^{N-1}} \;=\; \frac{(N!)^2}{N 2^{N-1}}.$$

■

Examples of convex lower bounding of products of N univariate functions are given in appendix A. Furthermore, conditions for convexity/concavity are provided for generalized polynomial terms, which are a special case of products of univariate functions, in appendix B. From the analysis in the previous sections it is clear that in order to convex lower bound the product of univariate functions it is necessary to be able to obtain the convex envelope, or at least a tight convex lower bounding function, of arbitrary univariate functions. To this end, guidelines for constructing the convex envelope of arbitrary functions of a single variable inside a certain interval are presented in the following section.

3.1. Convex Envelopes of Univariate Functions

Computing the convex envelopes of arbitrary twice differentiable functions in a single variable appears frequently as a task in many complex convex lower bounding situations. In some cases, this is a straightforward task, for example if $f \in \mathcal{C}^2 : [a, b] \rightarrow \mathcal{R}$ is convex then its convex envelope coincides with the original function:

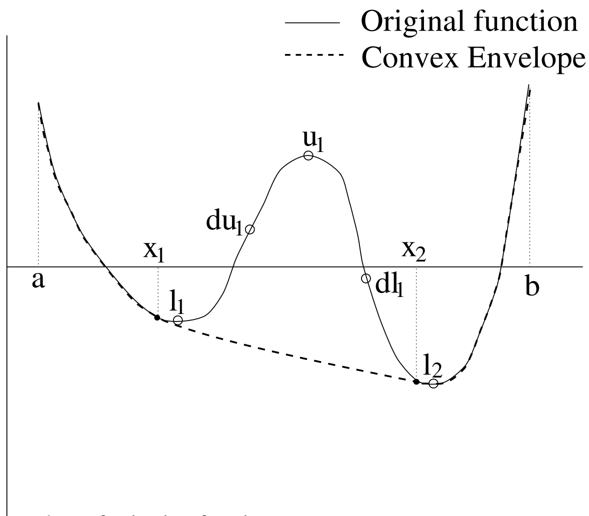

Figure 2. Convex envelope of univariate function.

$$\phi(f(x)) \;=\; f(x),\; \forall\; x \;\in\; [a,b] \;\; \text{if and only if} \;\; f(x) \text{ is convex in } \; [a,b]\,.$$

If now $f(x)$ is concave, then its convex envelope is a line segment connecting the end points of the graph of the function:

$$\phi(f(x)) \;=\; \frac{f(b)-f(a)}{b-a}x + \frac{bf(a)-af(b)}{b-a},\; \forall\; x \;\in\; [a,b]$$

Constructing the convex envelope of an arbitrary nonconvex function, however, is a much more demanding task because its graph alternates between convex and concave portions. In general, the convex envelope of nonconvex univariate functions is composed by different representations in different subintervals. More specifically, the convex envelope curve alternates between the original function (convex portions of the curve) and line segments (concave portions) (See Figure 2). The challenge here is to locate the exact points c_k^l, c_k^r $k = 1,\ldots,K$ where the convex envelope changes representation from a line segment to trace the curve of the original function and vice–versa. The number of these "switch–over" points depends on the frequency that $f''(x)$ changes sign in the interval $[a, b]$. The actual locations of these points depend not only on the shape of the function but also on the location of the end points.

Locating the exact location and number of points c_k^l, c_k^r $k = 1,\ldots,K$ requires knowledge of global information about the univariate function $f(x)$ in the interval $[a, b]$. More specifically, the location of all unconstrained local minima l_i, local maxima u_i, and inflection points dl_i, du_i is needed:

$$\begin{array}{rlcl} l_i: & f'(l_i) = 0, & & f''(l_i) \geq 0 \\ u_i: & f'(u_i) = 0, & & f''(u_i) \leq 0 \\ dl_i: & f''(dl_i) = 0, & & f'(dl_i) \leq 0 \\ du_i: & f''(du_i) = 0, & & f'(du_i) \geq 0 \end{array}$$

These points can be obtained by utilizing a robust solver guaranteed to locate all solutions of univariate functions in an interval [13]. Due to the alternating of convex and concave portions of the nonconvex function f, there is a specific order with which these points appear in the graph of the univariate function $f(x)$ which is:

$$\cdots [dl - l - du - u]_i \cdots$$

This naturally provides a partitioning of the initial interval $[a, b]$ into convex subintervals $[dl_i, du_i]$ and concave ones $[du_i, dl_i]$.

The procedure for locating the first point where the convex envelope changes representation depends on whether $f(x)$ is convex or concave at $x = a$. If f is concave at $x = a$ then the initial segment of the convex envelope is a line. The next segment of the convex envelope is the function itself starting at the point x where the slope f' of f equals the slope of the line connecting a with x.

$$\frac{f(x) - f(a)}{x - a} = f'(x)$$

Note that x belongs to one of the convex subintervals $[dl_i, du_i]$ since f must be convex at x. This implies that the task of locating x corresponds to drawing a tangent from the fixed point $(a, f(a))$ to each one of the convex function representations defined in the subintervals $[dl_i, du_i]$. Because there exists a single tangent to a convex function drawn from a point outside the a convex function [21] any standard bisection algorithm can be utilized to locate x. The correct subinterval $[dl_i, du_i]$ is then the one which provides a line that does not cut-off any portion of the curve $f(x)$.

If f is convex at $x = a$, then the initial segment of the convex envelope can be either a line or the function itself. If there exists a convex subinterval $[dl_i, du_i]\,,\ i = 2, \ldots$ where the equation $f(x) - f(a) = f'(x)\,(x - a)$ has a solution x which defines a line that does not cut-off any portion of the curve $f(x)$ then the initial segment of the convex envelope is a line connecting the points $(a, f(a))$ and $(x, f(x))$. Otherwise, the initial segment of the convex envelope is the function f itself. The last point of this segment x_1 is found by locating the end points x_1, x_2 of the next subinterval where the convex envelope becomes a line segment. This corresponds to drawing a common tangent to f inside the intervals $[dl_1, du_1]$ and $[dl_i, du_i]\,,\ i = 2, \ldots$. and is the solution of the following system of two equations:

$$\frac{f(x_2) - f(x_1)}{x_2 - x_1} = f'(x_1) = f'(x_2)$$

where $x_1 \in [dl_1, du_1]$ and $x_2 \in [dl_i, du_i]\,,\ i = 2, \ldots$. Again, the correct subinterval $[dl_i, du_i]\,,\ i = 2, \ldots$ is then the one for which the line connecting the points $(x_1, f(x_1))$ and $(x_2, f(x_2))$ does not cut-off any portion of the curve $f(x)$. The next line segment is then found by iteratively solving the system of two equations for locating the new points x_1, x_2. This time however, $x_1 \in [dl_i, du_i]$ and $x_2 \in [dl_j, du_j]\,,\; j = i+1, \ldots$. This is continued until the end point $x = b$ is met. Based on this analysis an iterative procedure is defined for constructing the convex envelope of arbitrary univariate nonconvex functions. In the next section, an alternative convex lower bounding method is discussed for problems involving only signomial terms.

4. Convex Lower Bounding of Signomial Problems

A large number of systems of nonlinear equalities subject to nonlinear inequalities have or can assume a generalized geometric problem formulation [24]:

$$\min_{\mathbf{t},\mathbf{s}} \quad s$$

$$\begin{aligned}
\text{subject to} \quad & G_j^1(\mathbf{t}) \;-\; G_j^2(\mathbf{t}) \;-\; s \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_E \\
-\; & G_j^1(\mathbf{t}) \;+\; G_j^2(\mathbf{t}) \;-\; s \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_E \\
& G_j^3(\mathbf{t}) \;-\; G_j^4(\mathbf{t}) \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_I \\
& t_i \;\geq\; 0, \quad i = 1, \ldots, N \\
\text{where} \quad & G_j^m(\mathbf{t}) \;=\; \sum_{k \in K_j^m} c_{jk} \prod_{i=1}^{N} t_i^{\alpha_{ijk}}, \; j \;\in\; \mathcal{N}_E, \; m = 1,2,3,4
\end{aligned}$$

Here $\mathbf{t} = (t_1, \ldots, t_N)$ is the positive variable vector; $G_j^m,\; m = 1,2,3,4,\; j \in \mathcal{N}_E \cup \mathcal{N}_I$ are positive posynomial functions in $\mathbf{t}$; α_{ijk} are arbitrary real constant exponents; whereas c_{jk} are given *positive* coefficients. Finally, sets $K_j^m,\; m = 1,2,3,4$ count how many positively/negatively signed monomials form the posynomials $G_j^m,\; m = 1,2,3,4$ respectively. Clearly, the above formulation corresponds to a highly nonlinear optimization problem with a nonconvex constraint set and possibly disjoint feasible region. However, after applying the transformation,

$$t_i \;=\; \exp x_i, \;\; i = 1, \ldots, N$$

to the original formulation we obtain the following optimization problem which involves constraints that are the difference of two convex functions.

$$\min_{\mathbf{x},\mathbf{s}} \quad s$$

$$\begin{aligned}
\text{subject to} \quad & G_j^1(\mathbf{x}) \;-\; G_j^2(\mathbf{x}) \;-\; s \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_E \\
-\; & G_j^1(\mathbf{x}) \;+\; G_j^2(\mathbf{x}) \;-\; s \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_E
\end{aligned}$$

$$G_j^3(\mathbf{x}) \;-\; G_j^4(\mathbf{x}) \;\leq\; 0, \;\; j \;\in\; \mathcal{N}_I$$

$$x_i^L \;\leq\; x_i \;\leq\; x_i^U, \quad i = 1, \ldots, N$$

$$\text{where} \qquad G_j^m(\mathbf{x}) \;=\; \sum_{k \in K_j^m} c_{jk} \exp\left\{ \sum_{i=1}^{N} \alpha_{ijk} x_i \right\}, \; j \;\in\; \mathcal{N}_E, \; m = 1, 2, 3, 4$$

A convex lower bounding formulation can be obtained by underestimating every separable concave function with a linear function. An analysis on the convex lower bounding procedure as well as on a number of techniques that improve the computational efficiency of the approach are described in detail in [24].

5. Convex Lower Bounding Using αBB

For arbitrary nonconvex functions $f \in \mathcal{C}^2 : \left[\mathbf{x}^L, \mathbf{x}^U\right] \rightarrow \mathcal{R}$, a convex lower bounding function $\mathcal{L}$ of f can be defined by augmenting f with the addition of a separable convex quadratic function of $\mathbf{x}$ as proposed in [23] and generalized to include equality and inequality constraints in [4].

$$\mathcal{L}(\mathbf{x}) \;=\; f(\mathbf{x}) + \alpha \left(\mathbf{x}^L - \mathbf{x}\right)^T \left(\mathbf{x}^U - \mathbf{x}\right)$$

$$\text{where} \qquad \alpha \;\geq\; \max\left\{ 0, -\frac{1}{2} \min_{\substack{k \\ \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U}} \lambda_k(\mathbf{x}) \right\}$$

Note that α is a nonnegative parameter which must be greater or equal to the negative one half of the minimum eigenvalue of f over $\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U$. The parameter α can be estimated either through the solution of an optimization problem or by using the concept of the measure of a matrix. The effect of adding the extra separable quadratic term to f is to make $\mathcal{L}$ convex by overpowering the nonconvexity characteristics of f with the addition of the term 2α to the diagonal elements of its Hessian matrix. This function $\mathcal{L}$ defined over the rectangular domain $\left[\mathbf{x}^L, \mathbf{x}^U\right]$, involves a number of properties which enable us to use as a tight convex lower bounding function of f. These properties, whose proof is given in [23], are as follows:

Property 1 $\mathcal{L}$ is a valid *underestimator* of f.

$$\forall \, \mathbf{x} \;\in\; \left[\mathbf{x}^L, \mathbf{x}^U\right], \;\; \mathcal{L}(\mathbf{x}) \leq f(\mathbf{x}).$$

Property 2 $\mathcal{L}$ *matches* f at all corner points.

Property 3 $\mathcal{L}$ is *convex* in $\left[\mathbf{x}^L, \mathbf{x}^U\right]$.

Property 4 The maximum separation between $\mathcal{L}$ and f is *bounded* and proportional to α and to the square of the diagonal of the current box constraints.

$$\max_{\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U} (f(\mathbf{x}) - \mathcal{L}(\mathbf{x})) = \frac{1}{4}\alpha||\mathbf{x}^U - \mathbf{x}^L||^2$$

Property 5 The underestimators constructed over supersets of the current set are always *less tight* than the underestimator constructed over the current box constraints for every point within the current box constraints.

Property 6 $\mathcal{L}$ corresponds to a relaxed dual bound of the original function f

This type of convex lower bounding is utilized for arbitrary nonconvex functions which lack any specific structure that might enable the construction of a more customized convex lower bounding function.

6. Procedure for Locating All Solutions

6.1. Description

A deterministic global optimization approach is proposed for locating all ϵ–solutions of nonlinear systems of equalities subject to nonlinear inequality constraints **(S)**. By introducing a slack variable, the initial problem **(S)** is transformed into a global optimization problem **(P)** whose multiple global minima (if any) correspond to the multiple solutions of **(S)**. A zero objective function value denotes the existence of a solution whereas a strictly positive objective function value implies that **(S)** has no solutions. This defines a one–to–one correspondence between solutions of the constrained system of equations **(S)** and multiple global minima with an objective value of zero for problem **(P)**. However, it has been shown [14] that no algorithm can exactly locate all multiple global minima of **(P)** with a finite number of function evaluations. A corrolary of this result [14] is that no algorithm can always localize, with a finite number of function evaluations, all globally optimal points by compact subrectangles in one–to–one correspondence with them. Therefore, a more tractable target, than finding all exact global minima of **(P)**, is to find arbitrarily small disjoint subrectangles containing all globally optimal points of **(P)**, possibly not in a one–to–one correspondence.

These multiple ϵ–global minima of **(P)**, (if any) can then be localized based on a branch and bound procedure involving the successive refinement of convex relaxations **(R)** of the initial problem **(P)**. Formulation **(R)** is obtained by replacing the nonconvex functions h_j^{nonc}, $-h_j^{nonc}$, g_k^{nonc} with *tight, convex lower bounding* functions $\hat{h}_{+,j}^{nonc}$, $\hat{h}_{-,j}^{nonc}$, $\hat{g}_k^{nonc}$, by following some of the techniques discussed in the previous section. Because **(R)** is convex, its global minimum within some box constraints can be routinely found with any commercially available local optimization algorithm (e.g. MINOS 5.4 [28]) and will always underestimate the global minimum of **(P)** within the same box constraints. Therefore, if the solution of **(R)** inside some rectangular region is strictly positive, then the solution of

(P) inside the same rectangular domain will also be strictly positive. A strictly positive solution for **(P)** implies that the slack variable s cannot be driven to zero, and thus **(S)** is guaranteed not to have any solutions inside the rectangular region at hand. This provides a mechanism for fathoming (eliminating) parts of the target region which are guaranteed not to contain any solutions. If on the other hand, the global minimum of **(R)** is negative then **(P)** may or may not involve a solution with a zero slack variable and therefore no deduction can be drawn regarding the existence or not of solutions for **(S)** inside the current rectangular domain. In this case, further partitioning of the current rectangular region is required until the global minimum of **(R)** becomes positive (fathoming) or a feasible point for **(P)** is found (convergence).

Based on Property (3) which demands that the convex lower bounding functions $\hat{h}^{nonc}_{+,j}$, $\hat{h}^{nonc}_{-,j}$, $\hat{g}^{nonc}_{k}$ must be tight, the maximum separation between the original functions and the convex underestimators can become arbitrarily ϵ–small by appropriately reducing the size of the rectangular domain. This implies that as the current box constraints $[\mathbf{x}^L, \mathbf{x}^U]$ collapse into a point the maximum difference ϵ between the original constraint set and its convex relaxation goes to zero. Therefore, any feasible point of problem **(R)** becomes at least ϵ–feasible for problem **(P)** by sufficiently tightening the bounds around this point. Tighter box constraints can be realized by partitioning the current rectangular domain into a number of smaller ones. Note that subdivision is required only for the variables which participate in nonlinear terms appearing in **(P)**.

One way of partitioning is to successively divide the current rectangle in two subrectangles by halving on the middle point of the longest side of the initial rectangle (bisection). At each iteration the lower bound of **(P)** is simply the infimum over all the minima of problem **(R)** in every subrectangle composing the initial rectangle. Therefore, a straightforward (bound improving) way of tightening the lower bound is to halve at each iteration, only the subrectangle responsible for the infimum of the minima of **(R)** over all subrectangles, according to the rules discussed earlier. Clearly, if the global minimum of **(R)** in any subrectangle is strictly greater than zero we can safely ignore this subrectangle because the global minimum of **(P)** cannot be situated inside it (fathoming step). This procedure generates a *nondecreasing* sequence for the lower bound of **(P)** yielding a set of candidate rectangles for containing a solution of **(S)**. Convergence is reached when none of the rectangles involve a negative lower bound (no solutions), or when all of the remaining rectangles with negative lower bounds are within the prespecified size tolerance ϵ_r. The basic steps of the proposed algorithm are summarized in the following subsection.

6.2. Algorithmic Steps

STEP 0 - *Initialization*

A size tolerance ϵ_r and a feasibility tolerance ϵ_f are selected and the iteration counter $Iter$ is set to one. Appropriate global bounds $\mathbf{x}^{LBD}, \mathbf{x}^{UBD}$ on $\mathbf{x}$ are chosen and local bounds $\mathbf{x}^{L,Iter}$, $\mathbf{x}^{U,Iter}$ for the first iteration are set to be equal to the global ones. Finally, select an initial point $\mathbf{x}^{c,Iter}$ that satisfies the linear equalities and convex inequalities of **(P)**.

STEP 1 - *Feasibility and Convergence Check*

If the maximum violation of all nonconvex constraints of **(P)** calculated at the current point $\mathbf{x}^{c,Iter}$ for (s=0) is less than ϵ_f,

$$\max\left\{\max_{j\in\mathcal{N}_{noncE}} \left|h_j^{nonc}(\mathbf{x}^{c,Iter})\right|,\ \max_{k\in\mathcal{N}_{noncI}} g_k^{nonc}(\mathbf{x}^{c,Iter})\right\} \leq \epsilon_f$$

then the point $\mathbf{x}^{c,Iter}$ is a ϵ_f–solution of **(S)**. Fathom current rectangle if its diagonal is less than ϵ_r,

$$||\mathbf{x}^{U,Iter} - \mathbf{x}^{L,Iter}|| \leq \epsilon_r.$$

and GO TO Step 4. Otherwise, continue with STEP 2.

STEP 2 - *Partitioning of Current Rectangle*

The current rectangle $\left[\mathbf{x}^{L,Iter}, \mathbf{x}^{U,Iter}\right]$ is partitioned into the following two rectangles $(r = 1, 2)$:

$$\begin{bmatrix} x_1^{L,Iter} & x_1^{U,Iter} \\ \vdots & \vdots \\ x_{l^{Iter}}^{L,Iter} & \frac{\left(x_{l^{Iter}}^{L,Iter} + x_{l^{Iter}}^{U,Iter}\right)}{2} \\ \vdots & \vdots \\ x_N^{L,Iter} & x_N^{U,Iter} \end{bmatrix}, \qquad \begin{bmatrix} x_1^{L,Iter} & x_1^{U,Iter} \\ \vdots & \vdots \\ \frac{\left(x_{l^{Iter}}^{L,Iter} + x_{l^{Iter}}^{U,Iter}\right)}{2} & x_{l^{Iter}}^{U,Iter} \\ \vdots & \vdots \\ x_N^{L,Iter} & x_N^{U,Iter} \end{bmatrix}$$

where l^{Iter} corresponds to the variable with the longest side in the initial rectangle,

$$l^{Iter} = \arg\max_i \left(x_i^{U,Iter} - x_i^{L,Iter}\right)$$

STEP 3 - *Solution of Convex Problems Inside Subrectangles*

Solve the following convex optimization problem **(R)** in both subrectangles $(r = 1, 2)$ by using any convex nonlinear solver (e.g. MINOS 5.4 [28]). If the solution $s_{sol}^{r,Iter}$ is negative then, it is stored along with the value of the variables $\mathbf{x}$ at the solution point $\mathbf{x}_{sol}^{r,Iter}$. If $s_{sol}^{r,Iter}$ is strictly positive then the element $(r, Iter)$ is fathomed.

STEP 4 - *Update Iteration Counter $Iter$ and Lower Bound s^{LBD}*

The iteration counter is increased by one,

$$Iter \longleftarrow Iter + 1$$

and the lower bound s^{LBD} is updated to be the minimum solution over the stored ones from previous iterations. Furthermore, the selected solution is erased from the stored set.

$$\begin{aligned} s^{LBD} &= s_{sol}^{r',Iter'} \\ \text{where} \quad s_{sol}^{r',Iter'} &= \min_{r,I} s_{sol}^{r,I}, \quad r = 1,2, \quad I = 1,\ldots,Iter-1. \end{aligned}$$

STEP 5 - *Update Current Point* $\mathbf{x}^{c,Iter}$ *and Current Bounds* $\mathbf{x}^{L,Iter}$, $\mathbf{x}^{U,Iter}$

The current point is selected to be the solution point of the previously found minimum solution in **STEP 4**,

$$\mathbf{x}^{c,Iter} = \mathbf{x}_{sol}^{r',Iter'}$$

and the current rectangle becomes the subrectangle containing the previously found solution,

$$\left[\mathbf{x}^{L,Iter}, \mathbf{x}^{U,Iter}\right] = \begin{bmatrix} x_1^{L,Iter'} & x_1^{U,Iter'} \\ \vdots & \vdots \\ x_{l^{Iter'}}^{L,Iter'} & \dfrac{\left(x_{l^{Iter'}}^{L,Iter'} + x_{l^{Iter'}}^{U,Iter'}\right)}{2} \\ \vdots & \vdots \\ x_N^{L,Iter'} & x_N^{U,Iter'} \end{bmatrix}, \quad \text{if} \quad r' = 1$$

$$\left[\mathbf{x}^{L,Iter}, \mathbf{x}^{U,Iter}\right] = \begin{bmatrix} x_1^{L,Iter'} & x_1^{U,Iter'} \\ \vdots & \vdots \\ \dfrac{\left(x_{l^{Iter'}}^{L,Iter'} + x_{l^{Iter'}}^{U,Iter'}\right)}{2} & x_{l^{Iter'}}^{U,Iter'} \\ \vdots & \vdots \\ x_N^{L,Iter'} & x_N^{U,Iter'} \end{bmatrix}, \quad \text{if} \quad r' = 2$$

STEP 7 - *Check for Convergence*

IF $s_{LBD} \leq 0$, then return to **STEP 1**

Otherwise, terminate.

Mathematical proof that the proposed procedure is guaranteed to converge to a set of disjoint rectangles containing all global minimum solutions of **(P)** is given based on the analysis of a standard deterministic global optimization algorithm presented in [15]. Because the employed branch and bound technique fathoms only rectangles *guaranteed not to contain any global minima of* **(P)** no solutions of **(P)** which are at least ϵ_r apart are missed. By following the proof in [23], a sufficient condition for the proposed branch and bound algorithm to be convergent to the global minima, requires that the bounding operation must be *consistent* and the selection operation *bound improving*.

A bounding operation is called *consistent* if (i) at every step any unfathomed partition can be further refined, and (ii) for any infinitely decreasing sequence of successively refined partition elements the gap between the lower and upper bounds goes to zero as the iterations go to infinity. Due to properties (1),(2),(3) of section 2 the gap between the lower and upper bound for any partition element goes to zero as the size of the partition element goes to zero as well. Furthermore, the employed bisection subdivision process (bisection along the longest side) is exhaustive because the size of an infinitely partitioned element goes to zero. Therefore, the bounding operation is consistent. Also, the employed selection operation is *bound improving* because the partition element where the actual lower bound is attained is selected for further partition in the immediately following iteration. Therefore according to Theorem IV.3. in [15] the employed global optimization algorithm is *convergent* to the global minima of **(P)**. In the next section the proposed global optimization algorithm is applied to a number of example problems.

7. Computational Results

In this section, a number of test problems are addressed which are aimed at determining the ability of the approach to find all solutions of constrained systems of equations with reasonable computational requirements. The proposed branch and bound convex lower bounding algorithm has been implemented in GAMS [5] and computational times are reported for all examples on a HP–730 workstation with size and feasibility tolerances of 10^{-4}.

Example 1. The first example involves the location of all the stationary points of the Himmelblau function as described in [33].

$$
\begin{aligned}
4x_1^3 + 4x_1x_2 + 2x_2^2 - 42x_1 - 14 &= 0 \\
4x_2^3 + 2x_1^2 + 4x_1x_2 - 26x_2 - 22 &= 0
\end{aligned}
$$

$$
\begin{aligned}
-5.0 \leq\ & x_1 \leq 5.0 \\
-5.0 \leq\ & x_2 \leq 5.0
\end{aligned}
$$

First, the change of variables

$$
y_1 = 1.0 + \frac{9}{10}\left(x_1 + 5.0\right), \quad y_2 = 1.0 + \frac{9}{10}\left(x_2 + 5.0\right)
$$

is performed which ensures that all variable are positive. This results in the following system of equations:

$$
\frac{2}{77}y_1^3 - \frac{30}{77}y_1^2 - \frac{20}{77}y_2 + \frac{2}{77}y_1y_2 + \frac{1}{77}y_2^2 + \frac{17}{11}y_1 - 1 = 0
$$

$$\frac{2}{121}y_2^3 - \frac{30}{121}y_2^2 + \frac{2}{121}y_1y_2 + \frac{127}{121}y_2 + \frac{1}{121}y_1^2 - \frac{20}{121}y_1 - 1 \;=\; 0$$

$$1.0 \;\leq\; y_1 \;\leq\; 10.0$$
$$1.0 \;\leq\; y_2 \;\leq\; 10.0$$

Then the exponential variable transformation, as described in section 4, is applied. The resulting problem is solved in 197 iterations and 10.89 seconds of CPU time. All nine solutions are found and shown in Table 1.

Table 1. Nine solutions of Example 1

# *Sol*	x_1^*	x_2^*
1	-0.2709	-0.9230
2	-0.1279	-1.9538
3	3.5844	-1.8481
4	3.3852	0.0739
5	3.0000	2.0000
6	0.0867	2.8843
7	-2.8051	3.1313
8	-3.0730	-0.0814
9	-3.7793	-3.2832

Example 2. This example addresses the equilibrium of the products of a hydrocarbon combustion process [25]. The problem is reformulated in the "element variables" space.

$$\begin{aligned}
y_1y_2 + y_1 - 3y_5 &= 0\\
2y_1y_2 + y_1 + 3R_{10}y_2^2 + y_2y_3^2 + R_7y_2y_3 + R_9y_2y_4 + R_8y_2 - Ry_5 &= 0\\
2y_2y_3^2 + R_7y_2y_3 + 2R_5y_3^2 + R_6y_3 - 8y_5 &= 0\\
R_9y_2y_4 + 2y_4^2 - 4Ry_5 &= 0\\
y_1y_2 + y_1 + R_{10}y_2^2 + y_2y_3^2 + R_7y_2y_3 + R_9y_2y_4 &\\
+R_8y_2 + R_5y_3^2 + R_6y_3 + y_4^2 - 1 &= 0\\
0.0001 \;\leq\; y_i \;\leq\; 100.0, \;\; i = 1, \ldots, 5 &
\end{aligned}$$

The values of the parameters $R, R_i,\ i = 5, \ldots, 10$ are shown in Table 2. Using the exponential variable transformation described in section 4, the single solution of the problem is found after 631 iterations and 31.7 seconds of CPU time (see Table 3).

Example 3. This example [6] addresses a badly scaled systems of equations:

$$
\begin{aligned}
10^4 x_1 x_2 - 1 &= 0 \\
\exp(-x_1) + \exp(-x_2) - 1.001 &= 0
\end{aligned}
$$

$$
\begin{aligned}
5.490\ 10^{-6} \leq\ & x_1 \leq 4.553 \\
2.196\ 10^{-3} \leq\ & x_2 \leq 18.210
\end{aligned}
$$

The bilinear terms $x_1 x_2, -x_1 x_2$ are underestimated based on the analysis in section 3, and the terms $\exp(-x_1), \exp(-x_2)$ are convex, however, $-\exp(-x_1), -\exp(-x_2)$ are univariate concave terms and are convex lower bounded with a line segment.

After 32 iterations and 1.5 seconds of CPU time, it is shown that

$$
(x_1^*, x_2^*) = (0.0000145067, 6.89335287)
$$

is a unique solution to the problem. Note that, the second solution $(x_1, x_2) = (0.00001098, 9.106)$ reported in [6] does not satisfy the nonlinear equations.

Example 4. This test example [11], involves a blend of trigonometric and exponential terms.

$$
\begin{aligned}
0.5 \sin(x_1 x_2) - 0.25 x_2/\pi - 0.5 x_1 &= 0 \\
(1 - 0.25/\pi)(\exp(2x_1) - e) + e x_2/\pi - 2 e x_1 &= 0
\end{aligned}
$$

$$
\begin{aligned}
0.25 \leq\ & x_1 \leq 1 \\
1.5 \leq\ & x_2 \leq 6.28
\end{aligned}
$$

The α–based underestimation, described in section 5, was chosen to address the convex lower bounding of the term $\sin(x_1 x_2)$. The eigenvalues of this term are equal to:

$$
\begin{aligned}
\lambda_{1,2} &= -\frac{1}{2}\left(x_1^2 + x_2^2\right)\sin(x_1 x_2) \\
&\pm \frac{1}{2}\sqrt{4 - 8 x_1 x_2 \sin(x_1 x_2)\cos(x_1 x_2) + \left[\left(x_1^2 + x_2^2\right)^2 - 4\right]\sin(x_1 x_2)^2 2}
\end{aligned}
$$

A lower bound on this expression is then:

$$
\lambda_{min} \geq -\max\left[(x_1^L)^2, (x_1^U)^2\right] - \max\left[(x_2^L)^2, (x_2^U)^2\right]
$$

Therefore,

$$\alpha \;=\; \frac{\max\left[(x_1^L)^2,(x_1^U)^2\right] + \max\left[(x_2^L)^2,(x_2^U)^2\right]}{2}$$

Two solutions are found for this problem in 45 iterations and 2.0 seconds of CPU time (see Table 4). Note that, both solutions were missed in [6].

Example 5. This test problem is Brown's almost linear system [17].

$$\begin{aligned}
2x_1 + x_2 + x_3 + x_4 + x_5 - 6 &= 0\\
x_1 + 2x_2 + x_3 + x_4 + x_5 - 6 &= 0\\
x_1 + x_2 + 2x_3 + x_4 + x_5 - 6 &= 0\\
x_1 + x_2 + x_3 + 2x_4 + x_5 - 6 &= 0\\
x_1x_2x_3x_4x_5 - 1 &= 0
\end{aligned}$$

$$-2 \;\le\; x_i \;\le\; 2, \quad i = 1,\ldots,5$$

This system exhibits two solutions: shown in Table 5. The α parameter was used to convex lower bound the last and only nonconvex constraint. The computational requirements for different values of α are shown in Table 6. Note that the total number of iterations remains relatively small even for very large values of α. Moreover, a value of α of as small as one appears to be sufficient.

Example 6. This example addresses a robot kinematics problem [17].

$$\begin{aligned}
4.73110^{-3}x_1x_3 - 0.3578x_2x_3 - 0.1238x_1 + x_7 &\\
-1.63710^{-3}x_2 - 0.9338x_4 - 0.3571 &= 0\\
0.2238x_1x_3 + 0.7623x_2x_3 + 0.2638x_1 - x_7 &\\
-0.07745x_2 - 0.6734x_4 - 0.6022 &= 0\\
x_6x_8 + 0.3578x_1 + 4.73110^{-3}x_2 &= 0\\
-0.7623x - 1 + 0.2238x_2 + 0.3461 &= 0\\
x_1^2 + x_2^2 - 1 &= 0\\
x_3^2 + x_4^2 - 1 &= 0\\
x_5^2 + x_6^2 - 1 &= 0\\
x_7^2 + x_8^2 - 1 &= 0
\end{aligned}$$

$$-1 \ \leq \ x_i \ \leq \ 1, \ \ i = 1, \ldots, 8$$

The only nonconvex terms in the formulation are the bilinear terms $x_1 x_3$, $x_2 x_3$, $x_6 x_8$ and are convex lower bounded based on the analysis of section 3. All distinct 16 solutions of this problem are found in 2188 iterations and 109.58 seconds of CPU time.

Example 7. This example involves the solution of a circuit design problem with extraordinary sensitivities to small perturbations [32] leading to the following set of equations.

$$\begin{aligned}
(1 - x_1 x_2)\, x_3 \left\{ \exp\left[x_5 \left(g_{1k} - g_{3k} x_7 10^{-3} - g_{5k} x_8 10^{-3} \right) \right] - 1 \right\} & \\
-g_{5k} + g_{4k} x_2 \ &= \ 0, \ \ k = 1, \ldots, 4 \\
(1 - x_1 x_2)\, x_4 \left\{ \exp\left[x_6 \left(g_{1k} - g_{2k} - g_{3k} x_7 10^{-3} + g_{4k} x_9 10^{-3} \right) \right] - 1 \right\} & \\
-g_{5k} x_1 + g_{4k} \ &= \ 0, \ \ \ k = 1, \ldots, 4 \\
x_1 x_3 - x_2 x_4 \ &= \ 0 \\
0 \ \leq \ x_i \ \leq \ 10, \ \ i &= 1, \ldots, 9
\end{aligned}$$

where

	$k=1$	$k=2$	$k=3$	$k=4$
g_{1k}	0.4850	0.7520	0.8690	0.9820
g_{2k}	0.3690	1.2540	0.7030	1.4550
g_{3k}	5.2095	10.0677	22.9274	20.2153
g_{4k}	23.3037	101.7790	111.4610	191.2670
g_{5k}	28.5132	111.8467	134.3884	211.4823

The α parameter was utilized to convex lower bound the various nonlinear terms. The single solution of the problem,

$$\begin{pmatrix}
x_1^* &= 0.899999, & x_6^* &= 7.999693 \\
x_2^* &= 0.449987, & x_7^* &= 5.000031 \\
x_3^* &= 1.000006, & x_8^* &= 0.999988 \\
x_4^* &= 2.000069, & x_9^* &= 2.000052 \\
x_5^* &= 7.999971, & &
\end{pmatrix}$$

was first reported in reference [32]. Computational requirements for various values of α are shown in Table 7. Note that these CPU requirements are only a small fraction of the ones reported in [32].

Furthermore, after relaxing the variable bounds to

$$-10 \ \leq \ x_i \ \leq \ 10, \ \ i = 1, \ldots, 9$$

a second solution was found

$$\begin{pmatrix} x_1^* &=& 0.823226, & x_6^* &=& -2.765092 \\ x_2^* &=& -0.553286, & x_7^* &=& 6.046646 \\ x_3^* &=& 0.671878, & x_8^* &=& 0.975940 \\ x_4^* &=& -0.999677, & x_9^* &=& -1.708489 \\ x_5^* &=& 8.854525. & & & \end{pmatrix}$$

which was missed in all previous attempts at solving this problem.

8. Summary and Conclusions

In this paper a deterministic branch and bound type algorithm was proposed for locating all ϵ–global solutions of certain classes of constrained systems of nonlinear equations. The approach is based on the one–to–one correspondence between the multiple solutions of the nonlinear systems and the multiple global minima with a zero objective value for the resulting nonconvex optimization problem. All multiple ϵ–global minima of the nonconvex optimization problem are localized based on a construction of upper bounds with function evaluations and lower bound on the global minimum solution through the convex relaxation of the constraint set and the solution of convex minimization problems. Based on the form of the participating functions, a number of alternative techniques for constructing this convex relaxation are proposed. In particular, by taking advantage of the properties of products of univariate functions, customized convex lower bounding functions are introduced for a large number of expressions that are or can be transformed into products of univariate functions. The utility of these convex lower bounding functions transcends the specifics of the root finding problem because they can be incorporated in any convex lower bounding algorithm. Alternative convex relaxation procedures involve either the difference of two convex functions employed in αBB [23] or the exponential variable transformation based underestimators employed for generalized geometric programming problems [24]. The proposed branch and bound approach is guaranteed to localize all ϵ–solutions of **(S)** within arbitrarily small rectangles in a finite number of iterations. A number of example problems from many areas of research have been addressed and in all cases, convergence to all multiple solutions was achieved with reasonable computational effort. Furthermore, in certain cases new solutions were identified.

Acknowledgments

Financial support from the National Science Foundation under Grants CBT–8857013, CTS–9221411, Air Force Office of Scientific Research, as well as Exxon Co., Amoco Chemicals Co., Mobil Co., and Tennessee Eastman Co. is gratefully acknowledged.

Appendix A

Convex Lower Bounding Examples of Univariate Functions

In this appendix a number of convex lower bounding situations are examined.

(1) Bilinear terms

The convex underestimation of bilinear terms xy inside the rectangular region $[x^L, x^U] \times [y^L, y^U]$ can be handled by invoking Theorem (1) and setting $f(x) = x$ and $g(y) = y$:

$$\begin{aligned} xy \ \geq \ \max \ & \{x^L y + y^L x - x^L y^L, \\ & x^U y + y^U x - x^U y^U\} \end{aligned}$$

Note that, the lower bounding procedure can be applied to a negatively–signed bilinear term $-xy$ by setting $f(x) = -x$ and $g(y) = y$:

$$\begin{aligned} -xy \ \geq \ \max \ & \{-x^U y - y^L x + x^U y^L, \\ & -x^L y - y^U x + x^L y^U\} \end{aligned}$$

Because in this case $f(x), g(y)$ are linear and therefore concave functions, we have from Theorem (2) that the obtained convex lower bounding functions are identical to the convex envelopes as were first derived by [3].

(2) Fractional terms

Convex lower bounding of the linear fractional term x/y inside the rectangular region $[x^L, x^U] \times [y^L, y^U]$ can also be accomplished based on Theorem (1) by selecting $f(x) = x$ and $g(y) = \frac{1}{y}$:

$$\begin{aligned} \frac{x}{y} \ \geq \ \max \ & \left\{\phi\left(\frac{x^L}{y}\right) + \phi\left(\frac{x}{y^U}\right) - \left(\frac{x^L}{y^U}\right),\right. \\ & \left.\phi\left(\frac{x^U}{y}\right) + \phi\left(\frac{x}{y^L}\right) - \left(\frac{x^U}{y^L}\right)\right\}. \end{aligned}$$

Note that, $\phi\left(\frac{x}{y^L}\right) = \frac{x}{y^L}$, $\phi\left(\frac{x}{y^U}\right) = \frac{x}{y^U}$ and

$$\phi\left(\frac{x^L}{y}\right) = \begin{cases} \frac{x^L}{y} & \text{if } x^L \geq 0 \\ \frac{x^L(y^L + y^U - y)}{y^L y^U} & \text{if } x^L < 0 \end{cases},$$

$$\phi\left(\frac{x^U}{y}\right) = \begin{cases} \frac{x^U}{y} & \text{if } x^U \geq 0 \\ \frac{x^U(y^L + y^U - y)}{y^L y^U} & \text{if } x^L < 0. \end{cases}$$

Therefore,

$$\frac{x}{y} \ \geq \ \max \ \left\{ \begin{bmatrix} \frac{x^L}{y} + \frac{x}{y^U} - \frac{x^L}{y^U} & \text{if } x^L \geq 0 \\ \\ \frac{x}{y^U} - \frac{x^L y}{y^L y^U} + \frac{x^L}{y^L} & \text{if } x^L < 0 \end{bmatrix} \right.$$

$$\left.\left[\begin{array}{ll} \frac{x^U}{y} + \frac{x}{y^L} - \frac{x^U}{y^L} & \text{if } x^U \geq 0 \\ \\ \frac{x}{y^L} - \frac{x^U y}{y^L y^U} + \frac{x^U}{y^U} & \text{if } x^U < 0 \end{array}\right]\right\}$$

The same approach can be used for negatively–signed linear fractional terms. In this case, however, we have $f(x) = x, g(y) = -1/y$. After following the same analysis we obtain:

$$-\frac{x}{y} \geq \max \left\{ \left[\begin{array}{ll} -\frac{x^L}{y} - \frac{x}{y^L} + \frac{x^L}{y^L} & \text{if } x^L \leq 0 \\ \\ -\frac{x}{y^L} + \frac{x^L y}{y^L y^U} - \frac{x^L}{y^U} & \text{if } x^L > 0 \end{array}\right]\right.$$

$$\left.\left[\begin{array}{ll} -\frac{x^U}{y} - \frac{x}{y^U} + \frac{x^U}{y^U} & \text{if } x^U \leq 0 \\ \\ -\frac{x}{y^U} + \frac{x^U y}{y^L y^U} - \frac{x^U}{y^L} & \text{if } x^U > 0 \end{array}\right]\right\}$$

(3) Trilinear terms

From Theorem (3) we know that a possible convex lower bounding function of xyz inside the rectangular region $[x^L, x^U] \times [y^L, y^U] \times [z^L, z^U]$ with $x^L, y^L, z^L \geq 0$ is:

$$\begin{aligned} xyz \geq s_0 &= \max \left\{ x^L s_1 + x s_1^L - x^L s_1^L,\ x^U s_1 + x s_1^U - x^U s_1^U \right\} \\ \text{where } s_1 &= \max \left\{ y^L z + y z^L - y^L z^L,\ y^U z + y z^U - y^U z^U \right\} \\ \text{and } s_1^L &= y^L z^L,\ s_1^U = y^U z^U \end{aligned}$$

However, Theorem (4) states that there exist three different convex lower bounding schemes for trilinear terms. These other two alternatives are:

$$\begin{aligned} xyz \geq s_0 &= \max \left\{ y^L s_1 + y s_1^L - y^L s_1^L,\ y^U s_1 + y s_1^U - y^U s_1^U \right\} \\ \text{where } s_1 &= \max \left\{ x^L z + x z^L - x^L z^L,\ x^U z + x z^U - x^U z^U \right\} \\ \text{and } s_1^L &= x^L z^L,\ s_1^U = x^U z^U \\ xyz \geq s_0 &= \max \left\{ z^L s_1 + z s_1^L - z^L s_1^L,\ z^U s_1 + z s_1^U - z^U s_1^U \right\} \\ \text{where } s_1 &= \max \left\{ x^L y + x y^L - x^L y^L,\ x^U y + x y^U - x^U y^U \right\} \\ \text{and } s_1^L &= x^L y^L,\ s_1^U = x^U y^U \end{aligned}$$

After eliminating s_0, s_1 and substituting for s_1^L, s_1^U we obtain for the three different convex lower bounding schemes:

$$\begin{aligned} xyz \geq \max \{ & xy^L z^L + x^L y z^L + x^L y^L z - 2x^L y^L z^L, \\ & xy^U z^U + x^U y z^L + x^U y^L z - x^U y^L z^L - x^U y^U z^U, \\ & xy^L z^L + x^L y z^U + x^L y^U z - x^L y^U z^U - x^L y^L z^L, \\ & xy^U z^U + x^U y z^U + x^U y^U z - 2x^U y^U z^U \} \end{aligned}$$

$$\begin{aligned} xyz \geq \max \; \{ & xy^Lz^L + x^Lyz^L + x^Ly^Lz - 2x^Ly^Lz^L, \\ & xy^Uz^L + x^Uyz^U + x^Ly^Uz - x^Ly^Uz^L - x^Uy^Uz^U, \\ & xy^Lz^U + x^Lyz^L + x^Uy^Lz - x^Uy^Lz^U - x^Ly^Lz^L, \\ & xy^Uz^U + x^Uyz^U + x^Uy^Uz - 2x^Uy^Uz^U \} \end{aligned}$$

$$\begin{aligned} xyz \geq \max \; \{ & xy^Lz^L + x^Lyz^L + x^Ly^Lz - 2x^Ly^Lz^L, \\ & xy^Lz^U + x^Lyz^U + x^Uy^Uz - x^Ly^Lz^U - x^Uy^Uz^U, \\ & xy^Uz^L + x^Uyz^L + x^Ly^Lz - x^Uy^Uz^U - x^Ly^Lz^L, \\ & xy^Uz^U + x^Uyz^U + x^Uy^Uz - 2x^Uy^Uz^U \} \end{aligned}$$

The combination of all three convex lower bounding alternatives yields the following eight linear functions in x, y, z whose maximum is a tight convex lower bounding function for xyz:

$$\begin{aligned} xyz \geq \max \; \{ & xy^Lz^L + x^Lyz^L + x^Ly^Lz - 2x^Ly^Lz^L, \\ & xy^Uz^U + x^Uyz^L + x^Uy^Lz - x^Uy^Lz^L - x^Uy^Uz^U, \\ & xy^Lz^L + x^Lyz^U + x^Ly^Uz - x^Ly^Uz^U - x^Ly^Lz^L, \\ & xy^Uz^L + x^Uyz^U + x^Ly^Uz - x^Ly^Uz^L - x^Uy^Uz^U, \\ & xy^Lz^U + x^Lyz^L + x^Uy^Lz - x^Uy^Lz^U - x^Ly^Lz^L, \\ & xy^Lz^U + x^Lyz^U + x^Uy^Uz - x^Ly^Lz^U - x^Uy^Uz^U, \\ & xy^Uz^L + x^Uyz^L + x^Ly^Lz - x^Uy^Uz^U - x^Ly^Lz^L, \\ & xy^Uz^U + x^Uyz^U + x^Uy^Uz - 2x^Uy^Uz^U \} \, . \end{aligned}$$

(4) Fractional trilinear terms

From Theorems (3), (4) we have that the three convex lower bounding alternatives for $\frac{xy}{z}$ inside the rectangular region $[x^L, x^U] \times [y^L, y^U] \times [z^L, z^U]$ with $x^L, y^L, z^L \geq 0$ are:

$$\frac{xy}{z} \geq s_0 = \max \; \{ x^Ls_1 + xs_1^L - x^Ls_1^L, \; x^Us_1 + xs_1^U - x^Us_1^U \}$$

$$\text{where } s_1 = \max \; \left\{ \frac{y^L}{z} + \frac{y}{z^U} - \frac{y^L}{z^U}, \; \frac{y^U}{z} + \frac{y}{z^L} - \frac{y^U}{z^L} \right\}$$

$$\text{and } s_1^L = \frac{y^L}{z^U}, \quad s_1^U = \frac{y^U}{z^L}$$

$$\frac{xy}{z} \geq s_0 = \max \; \{ y^Ls_1 + ys_1^L - y^Ls_1^L, \; y^Us_1 + ys_1^U - y^Us_1^U \}$$

$$\text{where } s_1 = \max \; \left\{ \frac{x^L}{z} + \frac{x}{z^U} - \frac{x^L}{z^U}, \; \frac{x^U}{z} + \frac{x}{z^L} - \frac{x^U}{z^L} \right\}$$

$$\text{and } s_1^L = \frac{x^L}{z^U}, \quad s_1^U = \frac{x^U}{z^L}$$

$$\frac{xy}{z} \;\geq\; s_0 \;=\; \max\;\left\{\frac{z^L}{s1}+\frac{z}{s_1^U}-\frac{z^L}{s_1^U},\;\frac{z^U}{s1}+\frac{z}{s_1^L}-\frac{z^U}{s_1^L}\right\}$$

$$\text{where}\;\; s_1 \;=\; \max\;\left\{x^Ly+xy^L-x^Ly^L,\; x^Uy+xy^U-x^Uy^U\right\}$$

$$\text{and}\;\; s_1^L \;=\; x^Ly^L,\;\; s_1^U \;=\; x^Uy^U$$

Again, after eliminating s_o, s_1 and replacing s_1^L, s_1^U we obtain:

$$\begin{aligned}\frac{xy}{z} \;\geq\; \max\;\Bigg\{&\frac{xy^L}{z^U}+\frac{x^Ly}{z^U}+\frac{x^Ly^L}{z}-2\frac{x^Ly^L}{z^U},\\ &\frac{xy^L}{z^U}+\frac{x^Ly}{z^L}+\frac{x^Ly^U}{z}-\frac{x^Ly^U}{z^L}-\frac{x^Ly^L}{z^U},\\ &\frac{xy^U}{z^L}+\frac{x^Uy}{z^U}+\frac{x^Uy^L}{z}-\frac{x^Uy^L}{z^U}-\frac{x^Uy^U}{z^L},\\ &\frac{xy^U}{z^L}+\frac{x^Uy}{z^L}+\frac{x^Uy^U}{z}-2\frac{x^Uy^U}{z^L}\Bigg\}\end{aligned}$$

$$\begin{aligned}\frac{xy}{z} \;\geq\; \max\;\Bigg\{&\frac{xy^L}{z^U}+\frac{x^Ly}{z^U}+\frac{x^Ly^L}{z}-2\frac{x^Ly^L}{z^U},\\ &\frac{xy^U}{z^U}+\frac{x^Uy}{z^L}+\frac{x^Ly^U}{z}-\frac{x^Ly^U}{z^U}-\frac{x^Uy^U}{z^L},\\ &\frac{xy^L}{z^U}+\frac{x^Ly}{z^L}+\frac{x^Uy^L}{z}-\frac{x^Uy^L}{z^L}-\frac{x^Ly^L}{z^U},\\ &\frac{xy^U}{z^L}+\frac{x^Uy}{z^L}+\frac{x^Uy^U}{z}-2\frac{x^Uy^U}{z^L}\Bigg\}\end{aligned}$$

$$\begin{aligned}\frac{xy}{z} \;\geq\; \max\;\Bigg\{&\frac{xy^L}{z^U}+\frac{x^Ly}{z^U}+\frac{x^Ly^L}{z}-2\frac{x^Ly^L}{z^U},\\ &\frac{xy^L}{z^L}+\frac{x^Ly}{z^L}+\frac{x^Uy^U}{z}-\frac{x^Ly^L}{z^L}-\frac{x^Uy^U}{z^L},\\ &\frac{xy^U}{z^U}+\frac{x^Uy}{z^U}+\frac{x^Ly^L}{z}-\frac{x^Uy^U}{z^U}-\frac{x^Ly^L}{z^U},\\ &\frac{xy^U}{z^L}+\frac{x^Uy}{z^L}+\frac{x^Uy^U}{z}-2\frac{x^Uy^U}{z^L}\Bigg\}\end{aligned}$$

After combining all three convex lower bounding alternatives, we obtain the following eight convex functions in x, y, z whose maximum is a tight convex lower bounding function for xyz:

$$
\begin{aligned}
\frac{xy}{z} \geq \quad \max \Bigg\{ & \frac{xy^L}{z^U} + \frac{x^L y}{z^U} + \frac{x^L y^L}{z} - 2\frac{x^L y^L}{z^U}, \\
& \frac{xy^L}{z^U} + \frac{x^L y}{z^L} + \frac{x^L y^U}{z} - \frac{x^L y^U}{z^L} - \frac{x^L y^L}{z^U}, \\
& \frac{xy^U}{z^L} + \frac{x^U y}{z^U} + \frac{x^U y^L}{z} - \frac{x^U y^L}{z^U} - \frac{x^U y^U}{z^L}, \\
& \frac{xy^U}{z^U} + \frac{x^U y}{z^L} + \frac{x^L y^U}{z} - \frac{x^L y^U}{z^U} - \frac{x^U y^U}{z^L}, \\
& \frac{xy^L}{z^U} + \frac{x^L y}{z^L} + \frac{x^U y^L}{z} - \frac{x^U y^L}{z^L} - \frac{x^L y^L}{z^U}, \\
& \frac{xy^U}{z^U} + \frac{x^U y}{z^L} + \frac{x^L y^U}{z} - \frac{x^L y^U}{z^U} - \frac{x^U y^U}{z^L}, \\
& \frac{xy^L}{z^U} + \frac{x^L y}{z^L} + \frac{x^U y^L}{z} - \frac{x^U y^L}{z^L} - \frac{x^L y^L}{z^U}, \\
& \frac{xy^U}{z^L} + \frac{x^U y}{z^L} + \frac{x^U y^U}{z} - 2\frac{x^U y^U}{z^L} \Bigg\}.
\end{aligned}
$$

Appendix B

Convexity/Concavity Identification of Generalized Polynomial Terms

In this appendix, necessary and sufficient conditions are provided for convexity/concavity of generalized polynomial terms of the form:

$$\left(\prod_{i=1}^{N} x_i^{d_i}\right), \quad d_i \in \Re, \; i = 1, \ldots, N$$

Generalized polynomial terms are a special case of products of univariate functions by selecting $f_i(x_i) = x_i^{d_i}$. First, the two variable case $f = x^a y^b$ is considered.

THEOREM 5 *If one of the following conditions holds,*

(1) $x, y \geq 0$.

(2) a, b *are even integers.*

(3) a, b *are odd integers, and* $xy \geq 0$.

(4) a, b *are integers,* a *is odd,* b *is even, and* $x \geq 0$.

(5) a, b *are integers,* a *is even,* b *is odd, and* $y \geq 0$

then (a) $f = x^a y^b$ *is convex in* (x, y) *if one of the following is true:*

(i) $a \leq 0,\ b \leq 0.$

(ii) $a \leq 0,\ 1 - a - b \leq 0.$

(iii) $b \leq 0,\ 1 - a - b \leq 0.$

and (b) $f = x^a y^b$ *is concave in* (x, y) *if*

(i) $a \geq 0,\ b \geq 0,\ a + b \leq 1$

Proof: The function $f = x^a y^b$ is convex in (x, y) only if all the eigenvalues of the Hessian matrix H of f are positive. The Hessian matrix H includes the second order derivatives of f with respect to x and y.

$$H = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix} \quad \text{where} \quad \begin{array}{lllll} f_{xx} & = & \frac{\partial^2 f}{\partial x^2} & & = a\,(a-1)\,x^{a-2} y^b \\ f_{xy} & = & \frac{\partial^2 f}{\partial x \partial y} & = \frac{\partial^2 f}{\partial y \partial x} & = abx^{a-1} y^{b-1} \\ f_{yy} & = & \frac{\partial^2 f}{\partial y^2} & & = b\,(b-1)\,x^a y^{b-2} \end{array}$$

The eigenvalues λ of H are the roots of the characteristic equation:

$$\lambda^2 - [f_{xx} + f_{yy}]\,\lambda + \left[f_{xx} f_{yy} - f_{xy}^2\right] = 0$$

This equation accepts only positive roots if and only if:

$$\begin{aligned} f_{xx} + f_{yy} &\geq 0 \\ f_{xx} f_{yy} - f_{xy}^2 &\geq 0 \end{aligned}$$

After substituting the expressions for f_{xx}, f_{yy}, f_{xy} we have,

$$\begin{aligned} x^{a-2} y^{b-2} \left[a\,(a-1)\,y^2 + b\,(b-1)\,x^2\right] &\geq 0 \\ y^{2b-2} x^{2a-2} \left[ab\,(1 - a - b)\right] &\geq 0. \end{aligned}$$

Note that $x^{2a-2} y^{2b-2}$ is always positive and furthermore $x^{a-2} y^{b-2}$ is also positive if one of the conditions (1)–(5) is true. In this case, the conditions for positivity of eigenvalues can be rewritten as:

$$\begin{aligned} a\,(a-1)\,y^2 + b\,(b-1)\,x^2 &\geq 0 \\ ab(1 - a - b) &\geq 0. \end{aligned}$$

These conditions are satisfied for every $x, y \in \Re$ only if all the following inequalities are satisfied:

(i) $a\,(a-1) \geq 0,$

(ii) $b\,(b-1) \geq 0,$

(iii) $ab\,(1 - a - b) \geq 0$

These inequalities decompose into the following three disjoint sufficient conditions for convexity of $f = x^a y^b$ assuming that one of the requirements (1)–(5) is true.

(i) $a \leq 0,\ b \leq 0.$

(ii) $a \leq 0,\ 1 - a - b \leq 0.$

(iii) $b \leq 0,\ 1 - a - b \leq 0.$

Note that the requirements $b \geq 1$ in (ii) and $a \geq 1$ in (iii) are implied by the other two inequalities, and therefore are not included.

The same analysis applies for checking concavity of $f = x^a y^b$. The characteristic equation accepts only negative roots if and only if

$$\begin{aligned} f_{xx} + f_{yy} &\leq 0 \\ f_{xx} f_{yy} - f_{xy}^2 &\geq 0 \end{aligned}$$

After some algebra we obtain the following set of conditions for concavity of f.

(i) $a(a-1) \leq 0,$

(ii) $b(b-1) \leq 0,$

(iii) $ab(1-a-b) \geq 0$

which alternatively can be written as

(i) $a \geq 0,\ b \geq 0,\ a + b \leq 1$

■

Note that if the term $x^{a-2}y^{b-2}$ is always negative, the conditions for convexity/concavity are reversed.

THEOREM 6 *If one of the following conditions holds,*

(1) a, b *are odd integers, and* $xy \leq 0$.

(2) a, b *are integers,* a *is odd,* b *is even, and* $x \leq 0$.

(3) a, b *are integers,* a *is even,* b *is odd, and* $y \leq 0$

then (a) $f = x^a y^b$ *is concave in* (x, y) *if one of the following is true:*

(i) $a \leq 0,\ b \leq 0.$

(ii) $a \leq 0,\ 1 - a - b \leq 0.$

(iii) $b \leq 0,\ 1 - a - b \leq 0.$

and (b) $f = x^a y^b$ *is convex in* (x, y) *if*

(i) $a \; \geq \; 0, \; b \; \geq \; 0, \; a + b \; \leq \; 1$

The proof of Theorem 6 is completely equivalent with that of Theorem 5 and therefore it is omitted.

A similar set of conditions for convexity/concavity can be obtained for the general n–dimensional case. For the sake of simplicity, we assume that $x_i \geq 0, \; i = 1, \ldots N$ which can always be achieved with simple rescaling of variables.

THEOREM 7 *The function* $f : \Re_+^N \rightarrow \Re_+, \; f(\mathbf{x}) \; = \; \prod_{i=1}^{N} x_i^{d_i}$ *is (a) is convex in* $\mathbf{x} \in \Re_+^N$ *if one of the following conditions is true:*

(i) $d_i \; \leq \; 0, \; \forall \, i = 1, \ldots, N$

(ii) $\exists \, j$ *such that* $d_j \; \geq \; 1 - \sum_{i \neq j}^{N} d_i,$ *and* $d_i \; \leq \; 0, \; \forall \, i \neq j, \; i = 1, \ldots, N$

and (b) is concave in $\mathbf{x} \in \Re_+^N$ *if*

(i) $d_i \; \geq \; 0, \; \forall \, i = 1, \ldots, N,$ *and* $\sum_{i=1}^{N} d_i \; \leq \; 1$

Proof: The second order derivatives of $f = \prod_{i=1}^{N} x_i^{d_i}$ are equal to:

$$f_{ij} \; = \; \frac{\partial^2 f}{\partial x_i \partial x_j} \; = \; \begin{cases} \frac{d_i d_j}{x_i x_j} f, & i \neq j \\ \frac{d_i (d_i - 1)}{x_i^2} f, & i = j \end{cases}$$

The expansion of the Hessian matrix of f yields the following characteristic equation:

$$\lambda^N + C_{N-1}(\mathbf{d}, \mathbf{x}) \lambda^{N-1} + \cdots + C_1(\mathbf{d}, \mathbf{x}) \lambda + C_0(\mathbf{d}, \mathbf{x}) \; = \; 0$$

where

$$\begin{aligned} C_{N-1}(\mathbf{d}, \mathbf{x}) \; &= \; \sum_{i=1}^{N} \frac{d_i \, (1 - d_i)}{x_i^2} f \\ C_{N-2}(\mathbf{d}, \mathbf{x}) \; &= \; \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{d_i d_j \, (1 - d_i - d_j)}{x_i^2 x_j^2} f^2 \\ \vdots \; &= \; \vdots \\ C_{N-k}(\mathbf{d}, \mathbf{x}) \; &= \; \sum_{\mathcal{P}_k} \frac{\prod_{j=1}^{k} d_{i_j} \left(1 - \sum_{j=1}^{k} d_{i_j} \right)}{\prod_{j=1}^{k} x_{i_j}^2} f^k \end{aligned}$$

$$\vdots = \vdots$$

$$C_0(\mathbf{d},\mathbf{x}) = \frac{\prod_{i=1}^{N} d_i \left(1 - \sum_{i=1}^{N} d_i\right)}{\prod_{i=1}^{N} x_i^2} f^N .$$

Note that the sets $\mathcal{P}_k,\ k = 1,\dots,N$ contain all possible ways that k elements of the set N can be selected.

$$\mathcal{N} = \{i \,:\, 1 \le i \le N\}$$

$$\mathcal{P}_k = \{(i_1,\dots,i_k) \,:\, i_j \in \mathcal{N},\ j = 1,\dots,k \text{ and } i_1 < i_2 < \ldots < i_k\},\ k = 1,\dots,N$$

A sufficient condition for the characteristic equation not to accept any negative roots is that all terms $C_{N-k}(\mathbf{d},\mathbf{x})\lambda^{N-k},\ k = 1,\dots,N$ maintain constant sign for every $\lambda < 0$ and for every $\mathbf{x} \in \Re_+^N$. More specifically, all terms $C_{N-k}(\mathbf{d},\mathbf{x})\lambda^{N-k},\ k = 1,\dots,N$ must be positive when N is even and negative if N is odd. This is satisfied if,

$$\forall\, k = 1\dots,N, \quad C_{N-k}(\mathbf{d},\mathbf{k}) \begin{cases} \le 0, & \text{if } k \text{ is odd} \\ \ge 0, & \text{if } k \text{ is even} \end{cases} \quad \forall\, \mathbf{x} \in \Re_+^N$$

These relations must be satisfied for every positive $\mathbf{x}$, therefore they can be written equivalently as:

$$\forall\, k = 1\dots,N, \quad \left(\prod_{j=1}^{k} d_{i_j}\right)\left(1 - \sum_{j=1}^{k} d_{i_j}\right) \begin{cases} \le 0, & \text{if } k \text{ is odd} \\ \ge 0, & \text{if } k \text{ is even} \end{cases}$$

for all $(i_1,\dots,i_k) \in \mathcal{P}_k$. Note that if $d_i \le 0,\ i = 1,\dots,N$ then

$$\forall\, k = 1\dots,N, \quad \left(1 - \sum_{j=1}^{k} d_{i_j}\right) \ge 0 \ \text{ and } \ \left(\prod_{j=1}^{k} d_{i_j}\right) \begin{cases} \le 0, & \text{if } k \text{ is odd} \\ \ge 0, & \text{if } k \text{ is even} \end{cases}$$

for all $(i_1,\dots,i_k) \in \mathcal{P}_k$, which implies that the characteristic equation accepts only positive roots for all $\mathbf{x} \in \Re_+^N$ when $d_i \le 0,\ i = 1,\dots,N$.

If we now allow only one exponent to be positive $d_{i_1} \ge 0$, and $d_{i_j} \le 0,\ \forall\, j = 2,\dots,N$ then we have

$$\forall\, k = 1\dots,N, \quad \left(\prod_{j=1}^{k} d_{i_j}\right) \begin{cases} \ge 0, & \text{if } k \text{ is odd} \\ \le 0, & \text{if } k \text{ is even} \end{cases} \quad \forall\,(i_1,\dots,i_k) \in \mathcal{P}_k$$

which means that the characteristic equation accepts only positive roots for all $\mathbf{x} \in \Re_+^N$ only if

$$\forall\, k = 1\ldots, N, \quad \left(1 - \sum_{j=1}^{k} d_{i_j}\right) \;\leq\; 0, \quad \forall\, (i_1, \ldots, i_k) \;\in\; \mathcal{P}_k$$

Or equivalently,

$$d_{i_1} \;\geq\; 1 - \sum_{j=2}^{k} d_{i_j}$$

Finally, it will be shown that these requirements cannot be satisfied if more than one $d_i,\ i = 1, \ldots, N$ is positive. Let $d_{i_1},\ d_{i_2} > 0$, then because $(i_1, i_2) \;\in\; \mathcal{P}_1$ we have:

$$d_{i_1}(1 - d_{i_1}) \;\leq\; 0 \;\text{ and }\; d_{i_2}(1 - d_{i_2}) \;\leq\; 0.$$

However, $d_{i_1},\ d_{i_2} > 0$, therefore

$$d_{i_1} \;\geq\; 1 \;\text{ and }\; d_{i_2} \;\geq\; 1.$$

Furthermore, $(i_1, i_2) \in \mathcal{P}_2$ so

$$(d_{i_1} d_{i_2})(1 - d_{i_1} - d_{i_2}) \;\geq\; 0$$

or

$$(1 - d_{i_1} - d_{i_2}) \;\geq\; 0$$

which is in contradiction with $d_{i_1}, d_{i_2} \geq 1$. Therefore, assuming conditions (i) or (ii) then f is convex in for all $\mathbf{x} \in \Re_+^N$.

By following the same line of thought, f is concave if all terms $C_{N-k}(\mathbf{d}, \mathbf{x})\lambda^{N-k},\ k = 1, \ldots, N$ maintain constant sign for every $\lambda > 0$ and for every $\mathbf{x} \in \Re_+^N$. This is true if,

$$\forall\, k = 1\ldots, N, \quad C_{N-k}(\mathbf{d}, \mathbf{k}) \;\geq\; 0, \quad \forall\, \mathbf{x} \in \Re_+^N.$$

This can be written equivalently as,

$$\forall\, k = 1\ldots, N, \quad \left(\prod_{j=1}^{k} d_{i_j}\right)\left(1 - \sum_{j=1}^{k} d_{i_j}\right) \;\geq\; 0, \quad \forall\, (i_1, \ldots, i_k) \;\in\; \mathcal{P}_k$$

For $(k = 1)$ we deduce that,

$$d_i\,(1 - d_i) \;\geq\; 0 \text{ or } 0 \;\leq\; d_i \;\leq\; 1, \;\; \forall\, i = 1, \ldots, N$$

This implies that,

$$\forall\, k = 2\ldots, N, \quad 1 - \sum_{j=1}^{k} d_{i_j} \;\geq\; 0, \quad \forall\, (i_1, \ldots, i_k) \;\in\; \mathcal{P}_k,$$

which simplifies into

$$\sum_{i=1}^{N} d_i \;\leq\; 1.$$

Therefore, if condition (iii) is true then f is concave in for all $\mathbf{x} \in \Re_{+}^{N}$. ∎

References

1. R. Horst, P.M. Pardalos and N.V. Thoai. *Introduction to Global Optimization*. Kluwer Academic Publishers, 1995.
2. R. Horst and P.M. Pardalos. *Handbook of Global Optimization*. Kluwer Academic Publishers, 1995.
3. F.A. Al-Khayyal and J.E. Falk. Jointly constrained biconvex programming. *Math. Opers. Res.*, 8:523, 1983.
4. I.P. Androulakis, C.D. Maranas, and C.A. Floudas. αBB: Global Optimization for Constrained Nonconvex Problems. *submitted to J. Global Opt.*, 1995.
5. A. Brooke, D. Kendrick, and A. Meeraus. *GAMS: A User's Guide*. Scientific Press, Palo Alto, CA., 1988.
6. L.G. Bullard and L.T. Biegler. Iterative Linear Programming Strategies for Constrained Simulation. *Computers chem. Engng.*, 15(4):239, 1991.
7. H.S. Chen and M.A. Stadtherr. On solving large sparse nonlinear equation systems. *Comp. chem. Engnr.*, 8:1, 1984.
8. D. Davidenko. *Dokl. Akad. Nauk USSR*, 88:601, 1953.
9. E. Doedel. *AUTO: Software for Continuation and Bifurcation Problems in Ordinary Differential Equations*. Applied Mathematics, Caltech, Pasadena, CA, 1986.
10. I.S. Duff, J. Nocedal, and J.K. Reid. The use of linear programming for the solution of sparse sets of nonlinear equations. *SIAM J. Sci. Stat. Comp.*, 8:99, 1987.
11. G.B. Ferraris and E. Tronconi. BUNSLI–a FORTRAN program for solution of systems of nonlinear algebraic equations. *Computers chem. Engng*, 10:12, 1986.
12. C.B. Garcia and W.I. Zangwill. *Pathways to Solutions, Fixed Points and Equilibria*. Pentice Hall, 1981.
13. E.R. Hansen. *Global Optimization Using Interval Analysis*. Marcel Dekkar, New York, NY, 1992.
14. P. Hansen, B. Jaumard, and S.H. Lu. Global optimization of univariate Lipschitz functions: I. Survey and propoerties. *Math. Prog.*, 55:251, 1992.
15. R. Horst and H. Tuy. *Global Optimization*. Springer-Verlag, 1st. edition, 1990.
16. R.B. Kearfott, M. Dawande, K. Du, and Ch. Hu. INTLIB, a Portable FORTRAN 77 Interval Standard Function Library. *in press*, 1994.
17. R.B. Kearfott and M. Novoa. INTBIS, a Portable Interval Newton/Bisection Package. *ACM Trans. Math. Soft.*, 16:152, 1990.
18. R.W. Klopfenstein. *J. Assoc. Comput. Mach.*, 8:366, 1961.
19. E. Lahaye. *C.R. Acad. Sci. Paris*, 198:1840, 1934.
20. J. Leray and J. Schauder. *Ann. Sci. Ecole Norm Sup.*, 51:45, 1934.
21. D.G. Luenberger. *Linear and Nonlinear Programming*. Addisson–Wesley, Reading, MA, 1984.
22. C.D. Maranas and C.A. Floudas. Global Optimization for Molecular Conformation Problems. *Ann. Oper. Res.*, 42:85, 1993.
23. C.D. Maranas and C.A. Floudas. Global Minimum Potential Energy Conformations of Small Molecules. *J. Glob. Opt.*, 4:135, 1994a.
24. C.D. Maranas and C.A. Floudas. Global Optimization in Generalized Geometric Porgramming. *submitted to Comp. chem. Engnr.*, 1994c.
25. K. Meintjes and A.P. Morgan. Chemical Equilibrium Systems as Numerical Test Problmes. *ACM Transactions on Mathematical Software*, 16(2):143, 1990.
26. J.J. Moré. The Levenberg–Marquardt algorithm: implementation and theory. In *Numerical Analysis–Proceedings of the Biennial Conference, Lecture Notes in Mathematics*, number 630, Berlin, Germany, 1987. Springer–Verlag.

27. A.P. Morgan. *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems*. Pentice–Hall, Inc., 1987.
28. B.A. Murtagh and M.A. Saunders. *MINOS5.0 Users Guide*. Systems Optimization Laboratory, Dept. of Operations Research, Stanford University, CA., 1983. Appendix A: MINOS5.0, Technical Report SOL 83-20.
29. A. Neumaier. *Interval Methods for Systems of Equations*. Cambridge University Press, Cambridge, UK, 1990.
30. J.R. Paloschi and J.D. Perkins. An implementation of quasi–Newton methods for solving sets of nonlinear equations. *Comp. chem. Engnr.*, 12:767, 1988.
31. M.J.D. Powell. *Numerical Methods for Nonlinear Algebraic Equations*. Gordon and Breach, London, UK, 1970.
32. H. Ratschek and J. Rokne. Experiments using interval analysis for solving a circuit design problem. *J. Glob. Opt.*, 3:501, 1993.
33. G.V. Reklaitis and K.M. Ragsdell. *Engineering Optimization*. John Wiley & Sons, New York, NY, 1983.
34. L.T. Watson, S.C. Billips, and A.P. Morgan. *ACM Trans. Math. Soft.*, 13:281, 1987.

Table 2. Parameters of Example 2

R	R_5	R_6	R_7	R_8	R_9	R_{10}
10	$1.930\,10^{-1}$	$4.106\,10^{-4}$	$5.451\,10^{-4}$	$4.497\,10^{-7}$	$3.407\,10^{-5}$	$9.615\,10^{-7}$

Table 3. Solution of Example 2

x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
0.00311410	34.59792453	0.06504178	0.85937805	0.03695186

Table 4. Solutions of Example 4

$\# \; Sol$	x_1^*	x_2^*
1	0.29944869	2.83692777
2	0.50000000	3.14159265

Table 5. Solutions of Example 5

$Sol \; \#$	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
1	1.000	1.000	1.000	1.000	1.000
2	0.916	0.916	0.916	0.916	1.418

Table 6. Computational requirements for Example 5

α	Iterations	CPU (sec)
1000	352	22.16
100	112	7.42
10	37	2.26
5	12	0.69
1	7	0.35

Table 7. Computational requirements for Example 7

α	Iterations	CPU (sec)
0.1	1645	987.91
0.01	212	143.41