J Glob Optim (2010) 47:661–685
DOI 10.1007/s10898-009-9484-1

On convex relaxations of quadrilinear terms

Sonia Cafieri · Jon Lee · Leo Liberti

Received: 28 April 2009 / Accepted: 24 October 2009 / Published online: 10 November 2009

Abstract The best known method to find exact or at least ε-approximate solutions to polynomial-programming problems is the spatial Branch-and-Bound algorithm, which rests on computing lower bounds to the value of the objective function to be minimized on each region that it explores. These lower bounds are often computed by solving convex relaxations of the original program. Although convex envelopes are explicitly known (via linear inequalities) for bilinear and trilinear terms on arbitrary boxes, such a description is unknown, in general, for multilinear terms of higher order. In this paper, we study convex relaxations of quadrilinear terms. We exploit associativity to rewrite such terms as products of bilinear and trilinear terms. Using a general technique, we formally establish the intuitive fact that any relaxation for k-linear terms that employs a successive use of relaxing bilinear terms (via the bilinear convex envelope) can be improved by employing instead a relaxation of a trilinear term (via the trilinear convex envelope). We present a computational analysis which helps establish which relaxations are strictly tighter, and we apply our findings to two well-studied applications: the Molecular Distance Geometry Problem and the Hartree–Fock Problem.

Keywords Quadrilinear · Trilinear · Bilinear · Convex relaxation · Reformulation · Global optimization · Spatial branch and bound · MINLP

S. Cafieri (✉) · L. Liberti
LIX, École Polytechnique, 91128 Palaiseau, France
e-mail: cafieri@lix.polytechnique.fr

L. Liberti
e-mail: liberti@lix.polytechnique.fr

J. Lee
Department of Mathematical Sciences, IBM T.J. Watson Research Center, PO Box 218,
Yorktown Heights, NY 10598, USA
e-mail: jonlee@us.ibm.com

1 Introduction

A polynomial programming problem is a Nonlinear Program (NLP) in the following general form:

$$\begin{array}{rl} \min & f(x) \\ & g(x) \le 0 \\ & x \in [x^L, x^U], \end{array} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$ are polynomial functions, and $x^L, x^U \in \mathbb{R}^n$. In general, the feasible region of (1) can be a nonconvex set, or the objective function may be nonconvex on the feasible region; it is such instances that are primarily of interest.

The solution method of choice is the spatial Branch-and-Bound (sBB) algorithm [2,21,34,38], which finds, in general, ε-approximate solutions for an arbitrarily small positive ε. The sBB algorithm works by recursively partitioning the search space (normally a box defined by variable ranges) along the coordinate direction that contributes most to the gap between lower and upper bounds on the optimal objective function value computed in each subproblem. For a minimization problem, the lower bound is usually computed by constructing and solving a convex relaxation, and the upper bound can simply be a local optimum found by a (local) NLP solver. When the gap for a particular subproblem is within ε, the subproblem is discarded without further partitioning, because the globally optimal objective value for that subproblem has been found. Subproblems are also discarded when the convex relaxation is infeasible or when their lower bound is greater than the current best overall solution found. The algorithm terminates when all the subproblems have been discarded.

One of the most crucial steps of sBB is the lower bound computation. For factorable problems (i.e. NLPs involving functional forms that can be written recursively, using a finite number of elementary "atomic" functions), it is possible to construct a convex relaxation automatically by means of a particular type of lifting reformulation (called MINLP standard form [24,38]) first proposed in [28] and then exploited in most existing sBB algorithms [2,5,21,34,38,41]. Because all polynomial functions are factorable (relative to multiplication), such a reformulation also applies to (1). Informally, high-order monomials are recursively rewritten as products of monomials of sufficiently low order for which a tight convex relaxation (possibly the convex envelope) is known. Each low-order monomial is replaced by an additional variable, and an equality constraint defining the additional variable in terms of the monomial it replaces is adjoined to (1). This operation is carried out recursively, until the functions f, g are linear forms. At this stage, each defining constraint is replaced by a set of constraints defining the convex relaxation of its feasible set, thus yielding a convex relaxation for the whole problem.

This iterative procedure for constructing a convex relaxation of (1) is symbolic rather than numeric, in the sense that it performs structural changes to the formulation of (1), adjoining variables and constraints, and replacing terms with variables and constraints with other constraints. The tightness of the resulting relaxation rests on the availability of a good "library" of convex relaxations/envelopes of elementary terms, such as monomials of low degree. Notice that this symbolic procedure is independent of any geometrical consideration concerning the feasible set of (1); thus, the direction of the objective function at a point is not an option for measuring convex relaxation tightness; we employ the partial order of set containment instead: convex relaxation A is tighter than convex relaxation B if $A \subsetneq B$.

Convex envelopes in explicit form are currently known for concave/convex univariate functions [1,37], bilinear terms [3,28], trilinear terms [29,30], univariate monomials of odd degree [20,23] and fractional terms [40]. General theoretical results for vertex-polyhedral convex envelopes are given in [39]. More advanced practical techniques for generating tight convex envelopes computationally are given in [11,12]. The multivariate monomial of smallest degree for which the convex envelope is not known in general is the quartic one [7]. In [35], bounding schemes for multilinear functions are compared; convex envelopes of multilinear terms are known for specific values of x^L, x^U [33]; a recent result characterizes the convex envelope of a class of functions including some multilinear functions [14]. The general quartic terms, up to symmetry on variable indices, are $x_1x_2x_3x_4$, $x_1x_2x_3^2$, $x_1x_2^3$, $x_1^2x_2^2$. Besides being fundamental building blocks for reformulating the general problem (1), these terms occur in important applications, such as the Molecular Distance Geometry Problem (MDGP) (see [16,17,26]) and the Hartree–Fock Problem (HFP) (see [18,25]).

In this paper, we focus on the quadrilinear term $x_1x_2x_3x_4$ (other types of quartic terms will be the object of future investigations). Associativity allows the iterative procedure to decomopose this term in several different ways: $((x_1x_2)x_3)x_4$ and $(x_1x_2x_3)x_4$ are two examples yielding as different outcomes the set of defining constraints $w_1 = x_1x_2$, $w_2 = w_1x_3$, $w_3 = w_2x_4$ and, respectively, $w_1 = x_1x_2x_3$, $w_2 = w_1x_4$ (where the w are added variables). It is this flexibility that we wish to investigate. We formally establish, using a general method that is not limited to monomials, that the second alternative yields a relaxation that is at least as tight as the first. Also, we establish by means of a computational assessment, that in many cases the second alternative yields strictly tighter relaxations. Finally, we study the behaviour of the proposed relaxations when used for the MDGP and HFP.

The importance of our result is emphasized by the fact that the traditional grouping used by sBB algorithms [2,5,27,36] is the slacker alternative $((x_1x_2)x_3)x_4$. Moreover, because the symbolic-relaxation procedure illustrated above also holds in the presence of integrality constraints on the variable vector x, and even when f, g include transcendental terms such as logarithm, exponentials and trigonometric functions, our results also apply to rather general Mixed-Integer Nonlinear Programming (MINLP) problems.

The rest of this paper is organized as follows. In Sect. 2 we review the explicit convex (linear) envelopes for bilinear and trilinear terms, and we use them to derive explicit convex (linear) relaxations of a quadrilinear term. In Sect. 3 we define a small formal language for expressing functions in infix form, and describe language semantics corresponding to the function itself and its reformulation/relaxation. We then exploit these formalisms to establish a formal comparison. In Sect. 4 we describe an experimental methodology to determine which type of term grouping order is best for the quadrilinear term. In Sect. 5 we introduce the MDGP and the HFP, and we report the results of computational experiments aimed at comparing different convex relaxations on application models. Section 6 concludes the paper.

2 Convex relaxations of quadrilinear terms

2.1 Existing convex envelopes

In this section, we review the known convex envelopes for bilinear and trilinear terms that we use to derive convex relaxations for the quadrilinear term. Each of these should be considered as a symbolic algorithm to be applied to the original problem in order to obtain a (convex) relaxation. Throughout, the domain of each variable x_i is the interval denoted $[x_i^L, x_i^U]$.

The bilinear term x_jx_k is replaced by a new variable x_i, and the following linear inequalities are added to the problem relaxation ("McCormick's envelope"; see [3,28]):

$$
\begin{aligned}
x_i &\geq x_j^L x_k + x_k^L x_j - x_j^L x_k^L \\
x_i &\geq x_j^U x_k + x_k^U x_j - x_j^U x_k^U \\
x_i &\leq x_j^L x_k + x_k^U x_j - x_j^L x_k^U \\
x_i &\leq x_j^U x_k + x_k^L x_j - x_j^U x_k^L.
\end{aligned}
$$

The trilinear term $x_j x_k x_h$ is replaced by a new variable x_i, and linear inequalities describing the convex envelope are added to the problem relaxation depending on the signs of the bounds on variables [29,30]. Denoting a permutation of x_j, x_k, x_h by the symbols x, y, z, in the case $x^L \geq 0$, $y^L \geq 0$, $z^L \leq 0$, $z^U \geq 0$ the following inequalities are added:

$$
\begin{aligned}
x_i &\geq y^U z^U x + x^U z^U y + x^U y^U z - 2x^U y^U z^U \\
x_i &\geq y^U z^L x + x^L z^U y + x^L y^U z - x^L y^U z^L - x^L y^U z^U \\
x_i &\geq y^U z^L x + x^L z^L y + x^L y^L z - x^L y^U z^L - x^L y^L z^L \\
x_i &\geq y^L z^U x + x^U z^L y + x^U y^L z - x^U y^L z^U - x^U y^L z^L \\
x_i &\geq y^L z^L x + x^U z^L y + x^L y^L z - x^U y^L z^L - x^L y^L z^L \\
x_i &\geq y^L z^U x + x^L z^U y + (\theta/(z^U - z^L))z \\
&\quad + (-(\theta z^L)/(z^U - z^L) - x^L y^U z^U - x^U y^L z^U + x^U y^U z^L) \\
x_i &\leq y^U z^L x + x^U z^L y + x^U y^U z - 2x^U y^U z^L \\
x_i &\leq y^L z^L x + x^U z^U y - x^U y^L z - x^U y^L z^U - x^U y^L z^L \\
x_i &\leq y^U z^U x + x^L z^U y + x^L y^L z - x^L y^U z^U - x^L y^L z^U \\
x_i &\leq y^U z^U x + x^L z^L y + x^L y^U z - x^L y^U z^U - x^L y^U z^L \\
x_i &\leq y^L z^U x + x^U z^U y + x^L y^L z - x^U y^L z^U - x^L y^L z^U \\
x_i &\leq y^L z^L x + x^L z^L y + (\bar{\theta}/(z^L - z^U))z \\
&\quad + (-(\bar{\theta} z^U)/(z^L - z^U) - x^U y^L z^L - x^L y^U z^L + x^U y^U z^U),
\end{aligned}
$$

where

$$
\begin{aligned}
\theta &:= x^L y^U z^U - x^U y^U z^L - x^L y^L z^U + x^U y^L z^U \\
\bar{\theta} &:= x^U y^L z^L - x^U y^U z^U - x^L y^L z^L + x^L y^U z^L.
\end{aligned}
$$

For details about other cases, depending on the signs of the bounds, the reader is referred to [29,30], where explicit expressions defining the facets of convex/concave envelopes are given for different combinations of variable bounds.

2.2 Obtaining convex relaxations of quadrilinear terms

Let us consider a quadrilinear term $x_1 x_2 x_3 x_4$. Exploiting associativity of the product, we rewrite it, in different ways, as products of monomials of degree two and three and use the known bilinear and trilinear convex envelopes (described in the previous subsection) to derive the corresponding convex relaxations.

We consider the four types of term groupings

$$
\begin{gathered}
((x_1 x_2) x_3) x_4 \\
(x_1 x_2)(x_3 x_4)
\end{gathered}
$$

$$(x_1x_2x_3)x_4$$
$$(x_1x_2)x_3x_4,$$

which, up to renaming the variables, exhausts the possibilities. Relative to these groupings, we consider, respectively, the sets:

$$S_1 = \{(x, w) \in \mathbb{R}^4 \times \mathbb{R}^3 \mid x_i \in [x_i^L, x_i^U],\ w_1 = x_1x_2, w_2 = w_1x_3, w_3 = w_2x_4\},$$
$$S_2 = \{(x, w) \in \mathbb{R}^4 \times \mathbb{R}^3 \mid x_i \in [x_i^L, x_i^U],\ w_1 = x_1x_2, w_2 = x_3x_4, w_3 = w_1w_2\},$$
$$S_3 = \{(x, w) \in \mathbb{R}^4 \times \mathbb{R}^2 \mid x_i \in [x_i^L, x_i^U],\ w_1 = x_1x_2x_3, w_2 = w_1x_4\},$$
$$S_4 = \{(x, w) \in \mathbb{R}^4 \times \mathbb{R}^2 \mid x_i \in [x_i^L, x_i^U],\ w_1 = x_1x_2, w_2 = w_1x_3x_4\}.$$

To derive four corresponding relaxations, we exploit a bilinear envelope thrice for the first two cases; a trilinear envelope followed by a bilinear envelope for S_3 and a bilinear envelope followed by a trilinear envelope for S_4.

3 Formal comparison

In this section, we provide a theoretical framework to investigate relaxation strength. We point out that it can be applied to any factorable mathematical program in order to compare pairs of relaxations. The general method that we propose is based on the idea of using a formal language to express the functions used in the objective and constraints of a mathematical program, and defining a semantic of strings of this language that is used to establish a formal comparison. The strings of this language are built recursively from operators and constant and variable symbols. These strings are in bijection with the "expression trees" mentioned in much of the sBB literature [5,21], but the presentation style follows the formal languages community [31].

A list of symbols used in this section, together with their meaning, is reported in Table 1.

Table 1 Table of notations

Symbol	Meaning
$\mathscr{X}$	Set of variables symbols
$\mathscr{O}$	Set of operator symbols
$\mathscr{P}$	Set of round brackets and comma symbols
$\mathscr{A}$	Alphabet
$\mathscr{L}$	Language
f, h	Strings of the language
$\beta_f(i)$	Symbol of the string f in the i-th position
$\omega(\otimes)$	Result of application of the operator $\otimes$ (to elements of the language)
$\mathscr{S}(f)$	Semantic of f
$\mathscr{R}(f)$	Relaxed semantic of f
$R(h)$	Relaxed composite semantic of h with respect to its substring

3.1 Mathematical expression language

Let $\mathscr{X} = \{x_1, \ldots, x_n\}$ be a set of variable symbols. Let $\mathscr{O}$ be a set of operator symbols. Operators are written in functional form, i.e. for an operator $\otimes \in \mathscr{O}$ with $p \geq 1$ arguments, we let $\otimes(x_1, \ldots, x_p)$ be the string describing the application of the operator $\otimes$ to the formal arguments $x_1, \ldots, x_p$ (sometimes, depending on the context, we also write the more usual infix form $x_1 \otimes \cdots \otimes x_p$). For an operator $\otimes \in \mathscr{O}$, we let the arity $\alpha(\otimes)$ be the set of numbers of arguments that the operator $\otimes$ can have. Let $\mathscr{P}$ be the set comprising the three symbols: "(", ")" and "," — that is: left round bracket, right round bracket, and comma.

Consider the alphabet $\mathscr{A} = \mathscr{X} \cup \mathbb{R} \cup \mathscr{O} \cup \mathscr{P}$, which we use to define a language whose strings (i.e. valid words) are precisely the functions used in the objective and constraints of a mathematical program.

Example 3.1 For $\mathscr{O} = \{+, -, \times, \div, \uparrow, \sqrt{\ }, \log, \exp, \sin, \cos, \tan\}$, we have the arities:

$$\begin{aligned} \alpha(+) &= \mathbb{N} \\ \alpha(-) &= \{1, 2\} \\ \alpha(\times) &= \mathbb{N} \\ \alpha(\div) = \alpha(\uparrow) &= \{2\} \\ \alpha(\sqrt{\ }) = \alpha(\log) = \alpha(\exp) = \alpha(\sin) = \alpha(\cos) = \alpha(\tan) &= \{1\}. \end{aligned}$$

The function $-\sin\left(x^2 - \frac{\sqrt{x}(y+1)(y-2)}{\log y}\right)$ is given by the following string:

$$-(\sin(-(\uparrow(x, 2)), \div(\times(\sqrt{\ }(x), +(y, 1), -(y, 2)), \log(y)))).$$

Let $\mathscr{L}$ be the language (set of strings of $\mathscr{A}$) built recursively according to the following rules: atomic expressions consisting of a single variable or real number are in the language, and for every operator and potential arity, if the arity p is compatible with the operator, then by applying the operator to p (ordered) elements of the language, we get another element of the language:

1. $\forall \ell \in \mathbb{R} \cup \mathscr{X}\ (\ell \in \mathscr{L})$
2. $\forall \otimes \in \mathscr{O}, p \in \mathbb{N}\ (p \in \alpha(\otimes) \to \forall \ell_1 \ldots, \ell_p \in \mathscr{L}\ (\otimes(\ell_1, \ldots, \ell_p) \in \mathscr{L}))$.

In the latter case, we also write $\omega(\otimes) = (\ell_1, \ldots, \ell_p)$. For $f \in \mathscr{L}$, we write $f(x_{i_1}, \ldots, x_{i_p})$ to emphasize the fact that the string for f only includes variable symbols x_{i_j} for $j \leq p$.

3.2 Relaxed semantics

We now introduce the formal definition of relaxed semantic of strings in $\mathscr{L}$.

Let $x \in \mathbb{R}^n$ be such that $x^L \leq x \leq x^U$ for $x^L, x^U \in \mathbb{R}^n$, and let $f \in \mathscr{L}$. Consider the sets:

$$\begin{aligned} \mathscr{S}(f) &= \{(w_f, x) \mid w_f = f(x),\ x^L \leq x \leq x^U\} \\ \mathscr{R}(f) &= \{(w_f, x) \mid A_f(w_f, x) \leq b_f,\ x^L \leq x \leq x^U\}, \end{aligned}$$

where $b_f \in \mathbb{R}^m$, $A_f(w_f, x) : \mathbb{R}^{(n+1)} \to \mathbb{R}^m$ is a convex function, and $\mathscr{S}(f) \subseteq \mathscr{R}(f)$. We call $\mathscr{R}(f)$ a *relaxed semantic* of f.

We also consider a relaxed semantic over substrings of f. For all $i \leq p$, let $f_i, g, h \in \mathscr{L}$ be such that $h(x) = g(f_1(x), \ldots, f_p(x))$. Let $\mathbf{w}_f = (w_{f_1}, \ldots, w_{f_p})$, $\mathbf{w} = (w_1, \ldots, w_p)$, and consider sets

$$\bar{R}(h) = \{(w_g, \mathbf{w}_f, x) \mid A_g(w_g, \mathbf{w}_f) \le b_g,\ A_{f_i}(w_{f_i}, x) \le b_{f_i}\ \forall i \le p,\ x^L \le x \le x^U\}$$
$$R(h) = \{(w, x) \mid \exists \mathbf{w}\ (w, \mathbf{w}, x) \in \bar{R}(h)\},$$

where $R(h)$ is the projection of $\bar{R}(h)$ on the subspace $\{(w, x) \in \mathbb{R}^{n+1}\}$. $R(h)$ is the *relaxed composite semantic* of h with respect to its substring $g(f_1, \dots, f_p)$.

In what follows, we always assume that $\mathscr{R}(h) \subseteq R(h)$, i.e. the relaxed semantic is at least as tight as the relaxed composite semantic.

Note that it is possible to define relaxed semantics slacker than relaxed composite semantics.

Example 3.2 Let $x = (x_1, x_2, x_3) \in \mathbb{R}^3$, $x^L = (x_1^L, x_2^L, x_3^L)$ and $x^U = (x_1^U, x_2^U, x_3^U)$ be such that $x^L \le x \le x^U$, where $x^L = -10$ and $x^U = 5$.

Consider the relaxed semantic given by the application of interval arithmetic. We recall that, for two intervals $[a, b]$ and $[c, d]$ such that $a \le 0 \le b, c \le 0 \le d$, the application of basic interval arithmetic rules gives the interval $[\min(bc, ad), \max(ac, bd)]$. Hence, applying interval arithmetic on $x_1x_2x_3$, we get the hyperrectangle R_{ia} given by $[-10, 5] \times [-10, 5] \times [-10, 5] \times [-1000, 500]$.

As relaxed composite semantic, we consider that obtained by applying the bilinear convex relaxation twice. Let $w_1 = x_1x_2$ and $w_2 = w_1x_3$. We have:

$$\begin{aligned}
\bar{R}_b = \{&(w_1, w_2, x) \in \mathbb{R}^5 \mid x^L \le x \le x^U,\\
&w_1 \ge x_1^L x_2 + x_2^L x_1 - x_1^L x_2^L,\ w_1 \ge x_1^U x_2 + x_2^U x_1 - x_1^U x_2^U,\\
&w_1 \le x_1^L x_2 + x_2^U x_1 - x_1^L x_2^U,\ w_1 \le x_1^U x_2 + x_2^L x_1 - x_1^U x_2^L,\\
&w_2 \ge w_1^L x_3 + x_3^L w_1 - w_1^L x_3^L,\ w_2 \ge w_1^U x_3 + x_3^U w_1 - w_1^U x_3^U,\\
&w_2 \le w_1^L x_3 + x_3^U w_1 - w_1^L x_3^U,\ w_2 \le w_1^U x_3 + x_3^L w_1 - w_1^U x_3^L\}.
\end{aligned}$$

The polytope obtained projecting $\bar{R}_b$ on $\mathbb{R}^4$ has extreme points $(-10, -10, -10, -1000)$, $(-10, -10, 5, 500)$, $(-10, 5, -10, 500)$, $(5, -10, -10, 500)$, $(-10, 5, 5, -250)$, $(5, -10, 5, -250)$, $(5, 5, -10, -250)$, $(5, 5, -5/2, -625)$, $(5, 5, -5/2, 500)$, $(5, 5, 5, 125)$, that are contained in R_{ia}.

3.3 Comparison of relaxed semantics

Let $F \in \mathscr{L}$. Let $\mathscr{O}' = \mathscr{O} \cup \{h\}$, where $h(x) = g(f_1(x), \dots, f_p(x))$, and let $\mathscr{A}' = \mathscr{A}$. Let F' be F rewritten using the rule $g(f_1, \dots, f_p) \to h$, i.e. using the alphabet in $\mathscr{A}'$.

Theorem 3.3 $R(F') \subseteq R(F)$.

Proof Let h be a string of the alphabet $\mathscr{A}$ such that $h(x) = g(f_1(x), \dots, f_p(x))$, and let h' be a string of the alphabet $\mathscr{A}'$ written using the operator replacing $g(f_1, \dots, f_p) \in \mathscr{L}$. The relaxed composite semantic of F and of F' are given by:

$$\begin{aligned}
\bar{R}(F) = &\{(w_g, \mathbf{w}_f, x) \mid A_g(w_g, \mathbf{w}_f) \le b_g,\ \forall i \le p\ A_{f_i}(w_{f_i}, x) \le b_{f_i},\ x^L \le x \le x^U\}\\
&\cup \bigcup_{k \le t} \{(w_{s_k}, \mathbf{w}_{r_k}, x) \mid A_{s_k}(w_{s_k}, \mathbf{w}_{r_k}) \le b_{s_k},\\
&\qquad \forall j \le p\ A_{r_{jk}}(w_{r_{jk}}, x) \le b_{r_{jk}},\ x^L \le x \le x^U\},\\
\bar{R}(F') = &\{(w_{h'}, x) \mid A'(w_{h'}, x) \le b',\ x^L \le x \le x^U\}
\end{aligned}$$

$$\cup \bigcup_{k \le t} \{(w_{s_k}, \mathbf{w}_{r_k}, x) \mid A_{s_k}(w_{s_k}, \mathbf{w}_{r_k}) \le b_{s_k},$$
$$\forall j \le p \ A_{r_{j_k}}(w_{r_{j_k}}, x) \le b_{r_{j_k}}, \ x^L \le x \le x^U\},$$

where $\mathbf{w}_f = (w_{f_1}, \ldots, w_{f_p})$, $\forall k \le t \ \mathbf{w}_{r_k} = (w_{r_1}, \ldots, w_{r_p})$, $w_{h'} = h'(x)$, $p \ge 1$ and $t \ge 0$.
Consider the relaxed composite semantic of h and of h'

$$\bar{R}(h) = \{(w_g, \mathbf{w}_f, x) \mid A_g(w_g, \mathbf{w}_f) \le b_g, \ \forall i \le p \ A_{f_i}(w_{f_i}, x) \le b_{f_i}, \ x^L \le x \le x^U\},$$
$$\bar{R}(h') = \{(w_{h'}, x) \mid A'(w_{h'}, x) \le b', \ x^L \le x \le x^U\},$$

and their projections $R(h) = \{(w, x) \mid \exists \mathbf{w} \ (w, \mathbf{w}, x) \in \bar{R}(h)\}$ and similarly for $R(h')$. Furthermore, let $\mathscr{R}(h)$ be the relaxed semantic of h:

$$\mathscr{R}(h) = \{(w_h, x) \mid A_h(w_h, x) \le b_h, \ x^L \le x \le x^U\},$$

where $w_h = h(x) = g(f_1(x), \ldots, f_p(x))$. By construction this is equal to $h'(x)$, so that $\mathscr{R}(h) \equiv R(h')$. From the hypothesis that $\mathscr{R}(h) \subseteq R(h)$ it follows that $R(h') \subseteq R(h)$. The same inclusion holds in the lifted space: $\bar{R}(h') \subseteq \bar{R}(h)$. Because all the other terms $A_{s_k}(w_{s_k}, \mathbf{w}_{r_k}) \le b_{s_k}$ $\forall k \le t$ are the same across both definitions, we have: $\bar{R}(F') \subseteq \bar{R}(F)$ and hence $R(F') \subseteq R(F)$. □

Theorem 3.3 establishes that for any relaxation of $x_1 x_2 \cdots x_k$ ($k \ge 3$) using any bilinear envelopes recursively, there is a relaxation employing also trilinear envelopes that is at least as tight. Moreover, such a relaxation can even be arrived at by replacing any single $(a \times b) \times c$ with $a \times b \times c$. In particular, if applied to the convex relaxations of $x_1 x_2 x_3 x_4$ described in Sect. 2, the theorem establishes that S_4 is at least as tight as S_1 and S_2, and that S_3 is at least as tight as S_1. Note that it does not give an indication on the relative tightness of S_3 and S_2. In the following, we show in detail how Theorem 3.3 can be applied to compare the tightness of two of the considered relaxations, namely S_1 and S_3.

Example 3.4 Let $x = (x_1, x_2, x_3, x_4) \in \mathbb{R}^4, x^L = (x_1^L, x_2^L, x_3^L, x_4^L)$ and $x^U = (x_1^U, x_2^U, x_3^U, x_4^U)$ be such that $x^L \le x \le x^U$. Let us consider a quadrilinear term $x_1 x_2 x_3 x_4$ and the two term grouping $((x_1 x_2) x_3) x_4$ and $(x_1 x_2 x_3) x_4$. We assume for this example that $x^L \ge 0, y^L \ge 0, z^L \le 0, z^U \ge 0$, where the symbols x, y, z denote a permutation of x_1, x_2, x_3.

The sets of involved variable and operator symbols are $\mathscr{X} = \{x_1, x_2, x_3, x_4\}$, $\mathscr{O} = \{\times\}$ (we also write yz meaning $y \times z$). Let $\mathscr{A}$ be the alphabet $\mathscr{A} = \mathscr{X} \cup \mathbb{R} \cup \mathscr{O} \cup \mathscr{P}$.

Consider first $((x_1 x_2) x_3) x_4$. Let $l, f, g, h \in \mathscr{L}$ be such that $h(x) = g(f(l(x)))$ and suppose that l, f, g are such that $w_l = x_1 x_2$, $w_f = w_l x_3$ and $w_g = w_f x_4$. We have:

$$
\begin{aligned}
\bar{R}(h) = \{&(w_g, w_f, w_l, x) \in \mathbb{R}^7 \mid x^L \le x \le x^U, \\
& w_l \ge x_1^L x_2 + x_2^L x_1 - x_1^L x_2^L, \ w_l \ge x_1^U x_2 + x_2^U x_1 - x_1^U x_2^U, \\
& w_l \le x_1^L x_2 + x_2^U x_1 - x_1^L x_2^U, \ w_l \le x_1^U x_2 + x_2^L x_1 - x_1^U x_2^L, \\
& w_f \ge w_l^L x_3 + x_3^L w_l - w_l^L x_3^L, \ w_f \ge w_l^U x_3 + x_3^U w_l - w_l^U x_3^U, \\
& w_f \le w_l^L x_3 + x_3^U w_l - w_l^L x_3^U, \ w_f \le w_l^U x_3 + x_3^L w_l - w_l^U x_3^L, \\
& w_g \ge w_f^L x_4 + x_4^L w_f - w_f^L x_4^L, \ w_g \ge w_f^U x_4 + x_4^U w_f - w_f^U x_4^U, \\
& w_g \le w_f^L x_4 + x_4^U w_f - w_f^L x_4^U, \ w_g \le w_f^U x_4 + x_4^L w_f - w_f^U x_4^L\}.
\end{aligned}
$$

The relaxed composite semantic $R(h)$ is given by the projection on the space of $(w, x) \in \mathbb{R}^5$.

Now consider $(x_1x_2x_3)x_4$. Let h' in $\mathscr{L}$ be written by using the operator f' replacing $f(l)$. Let

$$\theta = x^L y^U z^U - x^U y^U z^L - x^L y^L z^U + x^U y^L z^U$$
$$\bar{\theta} = x^U y^L z^L - x^U y^U z^U - x^L y^L z^L + x^L y^U z^L.$$

We have:

$$\begin{aligned}
\bar{R}(h') = \{&(w_g, w_{f'}, x) \in \mathbb{R}^6 \mid x^L \le x \le x^U,\\
&w_{f'} \ge y^U z^U x + x^U z^U y + x^U y^U z - 2x^U y^U z^U,\\
&w_{f'} \ge y^U z^L x + x^L z^U y + x^L y^U z - x^L y^U z^L - x^L y^U z^U,\\
&w_{f'} \ge y^U z^L x + x^L z^L y + x^L y^L z - x^L y^U z^L - x^L y^L z^L,\\
&w_{f'} \ge y^L z^U x + x^U z^L y + x^U y^L z - x^U y^L z^U - x^U y^L z^L,\\
&w_{f'} \ge y^L z^L x + x^U z^L y + x^L y^L z - x^U y^L z^L - x^L y^L z^L,\\
&w_{f'} \ge y^L z^U x + x^L z^U y + (\theta/(z^U - z^L))z\\
&\qquad + (-(\theta z^L)/(z^U - z^L) - x^L y^U z^U - x^U y^L z^U + x^U y^U z^L),\\
&w_{f'} \le y^U z^L x + x^U z^L y + x^U y^U z - 2x^U y^U z^L,\\
&w_{f'} \le y^L z^L x + x^U z^U y - x^U y^L z - x^U y^L z^U - x^U y^L z^L,\\
&w_{f'} \le y^U z^U x + x^L z^U y + x^L y^L z - x^L y^U z^U - x^L y^L z^U,\\
&w_{f'} \le y^U z^U x + x^L z^L y + x^L y^U z - x^L y^U z^U - x^L y^U z^L,\\
&w_{f'} \le y^L z^U x + x^U z^U y + x^L y^L z - x^U y^L z^U - x^L y^L z^U,\\
&w_{f'} \le y^L z^L x + x^L z^L y + (\bar{\theta}/(z^L - z^U))z\\
&\qquad + (-(\bar{\theta} z^U)/(z^L - z^U) - x^U y^L z^L - x^L y^U z^L + x^U y^U z^U),\\
&w_g \ge w_{f'}^L x_4 + x_4^L w_{f'} - w_{f'}^L x_4^L,\ \ w_g \ge w_{f'}^U x_4 + x_4^U w_{f'} - w_{f'}^U x_4^U,\\
&w_g \le w_{f'}^L x_4 + x_4^U w_{f'} - w_{f'}^L x_4^U,\ \ w_g \le w_{f'}^U x_4 + x_4^L w_{f'} - w_{f'}^U x_4^L\}.
\end{aligned}$$

Let $R(h) = \{(w, x) \mid \exists \mathbf{w}\ (w, \mathbf{w}, x) \in \bar{R}(h)\}$ be the projection of $\bar{R}$ and similarly let $R(h')$ be the projection of $\bar{R}(h')$. The hypotesis $\mathscr{R}(h) \subseteq R(h)$ is satisfied, hence, from Theorem 3.3, we have that $R(h') \subseteq R(h)$, i.e. S_3 is at least as tight as S_1.

4 Computational assessment for the quadrilinear term

Theorem 3.3 allows the comparison of some pairs of relaxations of quadrilinear terms, but it does not give an indication on the actual strength of the relaxations. We carried out numerical experiments to analyze the four convex relaxations in order to evaluate their relative tightness. Toward this aim, we generated a set of instances varying the signs of the bounds on the variables x_i, $i = 1, \ldots, 4$. Missing cases on sign combinations are equivalent to covered cases by simple symmetry considerations. Instances were generated having the same initial width of the bound intervals for all variables, and then progressively, for $i = 1, 2, 3$, reducing the width of the bound interval of x_i. This simulates a typical behavior of a sBB algorithm, which progressively reduces the size of the variable intervals. This reduction in the widths of the intervals is made preserving the signs of the bounds, changing a bound interval $[x_i^L, x_i^U]$ to $[x_i^L + 1/2, x_i^U - 1/2]$. Initial intervals $[x_i^L, x_i^U]$ were generated by considering the cases

$x_i^L > 0, x_i^U > 0, x_i^L < 0, x_i^U < 0$ and $x_i^L < 0, x_i^U > 0$. Specifically, we set $x_i^L = 1, x_i^U = 3$ in the first case, $x_i^L = -3, x_i^U = -1$ in the second case and $x_i^L = -1, x_i^U = 1$ in the third case.

The comparison among the considered relaxations is made in terms of the volume of the corresponding enclosing polytopes. This method of comparison, introduced in [19], is independent of any objective function. Because exploiting envelopes for bilinear and trilinear terms leads to an increased number of variables, so that the obtained polytopes belong to $\mathbb{R}^7(S_1, S_2)$ and $\mathbb{R}^6(S_3, S_4)$ respectively, we project the polytopes on $(x, f(x) := x_1x_2x_3x_4) \in \mathbb{R}^5$ in order to compare the results. The projection is computed using the software CDD [9], that calculates projections in exact rational arithmetic. Then, the volume of each of the obtained projected polytopes is computed using the LRS code [4]. Again, the results are computed in exact rational arithmetic.

In Table 2 we report the values of the volumes of the polytopes corresponding to S_1, S_2, S_3, S_4 projected onto $\mathbb{R}^5$, obtained for each problem instance. We remark that the volumes are reported in fixed-precision decimal format for easier reading, although they were computed in exact rational arithmetic. For each variable x_i, the width of the bound interval $x_i^U - x_i^L$ and the sign of x_i^L and x_i^U are also listed.

As expected, reducing the width of the bounds on variables, the polytopes have decreasing volumes, while keeping the same relative size with respect to the others. Note that for 85% of the test instances, the smallest values of the volume are obtained with the relaxation corresponding to S_4. These values in 5% of instances are the same obtained with S_3 and with S_1 and are the same obtained with S_2 again in 5% of instances (other than the previous ones). On the remaining instances, in 5% of cases we get the same value for the volume of the four enveloping polytopes, in the other cases the smallest values are obtained with S_3. These results show that S_3 and S_4 always yield the best relaxations. It is interesting to remark that in some cases the lowest value is also reached exploiting bilinear relaxations, but we never find that S_1 and S_2 provide the lowest volumes. This confirms the Theorem 3.3 (Sect. 3): the best relaxations are obtained employing convex envelopes for trilinear terms and not just bilinear ones. Although our results depend on the particular bounds that we tried, we get a significant indication of the strength of the considered relaxations and their dependence on signs and widths of bounds.

We also compute the extreme points of $w = x_1x_2x_3x_4$ by considering all the combinations of bounds on the variables and we get the convex hull for these points. This gives the tightest linear approximation of the quadrilinear term, but it is not used in sBB codes because it cannot be expressed explicitly in function of the bounds on the variables. We use it as an approximation against which to compare the others. In Table 3 we report, for each of the considered relaxations, the ratios between the volume of the enveloping polytope projected onto $\mathbb{R}^5$ and the volume of the above approximation. According to the results in Table 2, we find the lowest ratios for the relaxation corresponding to S_4. Furthermore, for the 5% of the instances all the volumes are equal; for 5% of the instances other than the previous ones, S_1, S_3 and S_4 give the same volume of that used as reference. For the remaining cases, we find some instances (10%) for which the polytope corresponding to S_4 is not only the tightest with respect to S_1, S_2, S_3, but is also equivalent to the polytope around the extreme points.

In order to easily analyze the overall pattern that is emerging in terms of variations of volumes of the enveloping polytopes, we report a graphical representation of the computational results in Fig. 1 and Fig. 2. For each of the sign combinations, we plot 4 line graphs, one for each of the linearizations. Each line has 4 points, corresponding to the instances obtained by progressively tightening the bounds for the considered sign combination. These

Table 2 Volumes of the enveloping polytopes projected onto $\mathbb{R}^5$

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*				
i_1	2	+, +	2	+, +	2	−, +	2	−, +	184.0444	**98.2667**	175.4074	**98.2667**
i_2	2	+, +	2	+, +	2	−, +	2	−, −	242.7111	245.9202	223.4074	**185.6000**
i_3	2	+, +	2	−, +	2	−, +	2	−, +	77.0370	77.0370	**69.6889**	77.0370
i_4	2	+, +	2	−, +	2	−, +	2	−, −	126.8148	203.7333	**101.6889**	126.8148
i_5	2	−, +	2	−, +	2	−, +	2	−, +	*27.7333*	*27.7333*	*27.7333*	*27.7333*
i_6	2	−, +	2	−, +	2	−, +	2	−, −	**49.0667**	77.0370	**49.0667**	**49.0667**
i_7	2	+, +	2	−, +	2	−, −	2	−, +	184.8889	203.7333	175.4074	**126.8148**
i_8	2	+, +	2	−, +	2	−, −	2	−, −	245.3333	245.9202	223.4074	**209.7778**
i_9	2	−, +	2	−, −	2	−, −	2	−, +	184.8889	203.7333	175.4074	**126.8148**
i_10	2	−, +	2	−, −	2	−, −	2	−, −	245.3333	245.9202	223.4074	**209.7778**
i_11	2	−, +	2	−, +	2	−, −	2	−, +	69.6889	77.0370	69.6889	**49.0667**
i_12	2	−, +	2	−, +	2	−, −	2	−, −	101.6889	98.2667	101.6889	**89.6000**
i_13	2	+, +	2	+, +	2	+, +	2	−, +	205.9795	245.9202	196.3487	**185.6000**
i_14	2	+, +	2	+, +	2	+, +	2	−, −	275.3128	296.2436	249.6821	**249.2667**
i_15	2	+, +	2	+, +	2	−, −	2	−, +	205.9795	245.9202	196.3487	**185.6000**
i_16	2	+, +	2	+, +	2	−, −	2	−, −	275.3128	296.2436	249.6821	**249.2667**
i_17	2	+, +	2	−, −	2	−, −	2	−, +	205.9795	245.9202	196.3487	**185.6000**
i_18	2	+, +	2	−, −	2	−, −	2	−, −	275.3128	296.2436	249.6821	**249.2667**
i_19	2	−, −	2	−, −	2	−, −	2	−, +	205.9795	245.9202	196.3487	**185.6000**
i_20	2	−, −	2	−, −	2	−, −	2	−, −	275.3128	296.2436	249.6821	**249.2667**
i_21	1	+, +	2	+, +	2	−, +	2	−, +	72.5304	**41.9111**	69.2622	**41.9111**
i_22	1	+, +	2	+, +	2	−, +	2	−, −	93.8637	97.0978	86.5956	**73.5556**
i_23	1	+, +	2	−, +	2	−, +	2	−, +	30.3147	30.3147	**28.3733**	30.3147
i_24	1	+, +	2	−, +	2	−, +	2	−, −	48.4480	77.1504	**41.7067**	48.4480
i_25	1	−, +	2	−, +	2	−, +	2	−, +	*6.9333*	*6.9333*	*6.9333*	*6.9333*
i_26	1	−, +	2	−, +	2	−, +	2	−, −	**12.2667**	19.2593	**12.2667**	**12.2667**
i_27	1	+, +	2	−, +	2	−, −	2	−, +	72.0071	77.1504	69.2622	**48.4480**
i_28	1	+, +	2	−, +	2	−, −	2	−, −	92.8071	92.6308	86.5956	**77.7387**
i_29	1	−, +	2	−, −	2	−, −	2	−, +	46.2222	50.9333	43.8519	**31.7037**
i_30	1	−, +	2	−, −	2	−, −	2	−, −	61.3333	61.4800	55.8519	**52.4444**
i_31	1	−, +	2	−, +	2	−, −	2	−, +	17.4222	19.2593	17.4222	**12.2667**
i_32	1	−, +	2	−, +	2	−, −	2	−, −	25.4222	24.5667	25.4222	**22.4000**
i_33	1	+, +	2	+, +	2	+, +	2	−, +	84.2095	97.0978	80.5206	**73.5556**
i_34	1	+, +	2	+, +	2	+, +	2	−, −	108.2095	114.5142	99.1873	**93.0000**
i_35	1	+, +	2	+, +	2	−, −	2	−, +	84.2095	97.0978	80.5206	**73.5556**
i_36	1	+, +	2	+, +	2	−, −	2	−, −	108.2095	114.5142	99.1873	**93.0000**
i_37	1	+, +	2	−, −	2	−, −	2	−, +	84.2095	97.0978	80.5206	**73.5556**
i_38	1	+, +	2	−, −	2	−, −	2	−, −	108.2095	114.5142	99.1873	**93.0000**
i_39	1	−, −	2	−, −	2	−, −	2	−, +	84.2095	97.0978	80.5206	**73.5556**
i_40	1	−, −	2	−, −	2	−, −	2	−, −	108.2095	114.5142	99.1873	**93.0000**

Table 2 continued

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*				
i_41	1	+, +	1	+, +	2	−, +	2	−, +	27.9709	**17.6104**	26.6667	**17.6104**
i_42	1	+, +	1	+, +	2	−, +	2	−, −	35.1376	36.4175	32.3333	**28.2958**
i_43	1	+, +	1	−, +	2	−, +	2	−, +	7.5787	7.5787	**7.0933**	7.5787
i_44	1	+, +	1	−, +	2	−, +	2	−, −	12.1120	19.2876	**10.4267**	12.1120
i_45	1	−, +	1	−, +	2	−, +	2	−, +	*1.7333*	*1.7333*	*1.7333*	*1.7333*
i_46	1	−, +	1	−, +	2	−, +	2	−, −	**3**.*0667*	4.8148	**3.0667**	**3.0667**
i_47	1	+, +	1	−, +	2	−, −	2	−, +	18.0018	19.2876	17.3156	**12.1120**
i_48	1	+, +	1	−, +	2	−, −	2	−, −	23.2018	23.1577	21.6489	**19.4347**
i_49	1	−, +	1	−, −	2	−, −	2	−, +	18.0018	19.2876	17.3156	**12.1120**
i_50	1	−, +	1	−, −	2	−, −	2	−, −	23.2018	23.1577	21.6489	**19.4347**
i_51	1	−, +	1	−, +	2	−, −	2	−, +	4.3556	4.8148	4.3556	**3.0667**
i_52	1	−, +	1	−, +	2	−, −	2	−, −	6.3556	6.1417	6.3556	**5.6000**
i_53	1	+, +	1	+, +	2	+, +	2	−, +	33.2785	36.4175	31.8323	**28.2958**
i_54	1	+, +	1	+, +	2	+, +	2	−, −	41.1118	42.1904	37.8323	**32.9208**
i_55	1	+, +	1	+, +	2	−, −	2	−, +	33.2785	36.4175	31.8323	**28.2958**
i_56	1	+, +	1	+, +	2	−, −	2	−, −	41.1118	42.1904	37.8323	**32.9208**
i_57	1	+, +	1	−, −	2	−, −	2	−, +	33.2785	36.4175	31.8323	**28.2958**
i_58	1	+, +	1	−, −	2	−, −	2	−, −	41.1118	42.1904	37.8323	**32.9208**
i_59	1	−, −	1	−, −	2	−, −	2	−, +	33.2785	36.4175	31.8323	**28.2958**
i_60	1	−, −	1	−, −	2	−, −	2	−, −	41.1118	42.1904	37.8323	**32.9208**
i_61	1	+, +	1	+, +	1	−, +	2	−, +	6.9927	**4.4026**	6.6667	**4.4026**
i_62	1	+, +	1	+, +	1	−, +	2	−, −	8.7844	9.1044	8.0833	**7.0740**
i_63	1	+, +	1	−, +	1	−, +	2	−, +	1.8947	1.8947	**1.7733**	1.8947
i_64	1	+, +	1	−, +	1	−, +	2	−, −	3.0280	4.8219	**2.6067**	3.0280
i_65	1	−, +	1	−, +	1	−, +	2	−, +	*0.4333*	*0.4333*	*0.4333*	*0.4333*
i_66	1	−, +	1	−, +	1	−, +	2	−, −	**0.7667**	1.2037	**0.7667**	**0.7667**
i_67	1	+, +	1	−, +	1	−, −	2	−, +	6.8539	7.2208	6.6667	**5.2947**
i_68	1	+, +	1	−, +	1	−, −	2	−, −	8.4872	8.8981	8.0833	**7.4587**
i_69	1	−, +	1	−, −	1	−, −	2	−, +	6.8539	7.2208	6.6667	**5.2947**
i_70	1	−, +	1	−, −	1	−, −	2	−, −	8.4872	8.8981	8.0833	**7.4587**
i_71	1	−, +	1	−, +	1	−, −	2	−, +	1.7733	1.8947	1.7733	**1.4333**
i_72	1	−, +	1	−, +	1	−, −	2	−, −	2.6067	2.6194	2.6067	**2.4000**
i_73	1	+, +	1	+, +	1	+, +	2	−, +	12.3542	12.9119	11.8636	**10.6573**
i_74	1	+, +	1	+, +	1	+, +	2	−, −	14.6459	15.3285	13.5303	**12.1469**
i_75	1	+, +	1	+, +	1	−, −	2	−, +	12.3542	12.9119	11.8636	**10.6573**
i_76	1	+, +	1	+, +	1	−, −	2	−, −	14.6459	15.3285	13.5303	**12.1469**
i_77	1	+, +	1	−, −	1	−, −	2	−, +	12.3542	12.9119	11.8636	**10.6573**
i_78	1	+, +	1	−, −	1	−, −	2	−, −	14.6459	15.3285	13.5303	**12.1469**

Table 2 continued

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$				
i_79	1	−, −	1	−, −	1	−, −	2	−, +	12.3542	12.9119	11.8636	**10.6573**
i_80	1	−, −	1	−, −	1	−, −	2	−, −	14.6459	15.3285	13.5303	**12.1469**

For each x_i, "$|\cdot|$" indicates the value of $x_i^U - x_i^L$, and "$sign$" indicates the pair of signs of x_i^L, x_i^U respectively. We remark that the case of all positive signs is equivalent to that of all negative signs
The best values are reported in bold face values are written in italic when they are equal for all the four considered cases

graphs suggest that the relaxation strength of quadrilinear terms follow different behaviors according to different bound widths/signs. We note that, after tightening the bounds on a variable, there are no examples where one curve goes from far below to far above another. This suggests that tightening the bounds has a comparable effect on the different relaxations.

The considered polytopes are expected to overlap. Given two polytopes, they share some common points or one is entirely contained in the other. We get a more precise information about tightness of the considered relaxations by checking relative containments of the corresponding (projected) polytopes. For each pair of polytopes P, Q, we check if P is contained in Q by checking that every extreme point of P satisfies all the inequalities defining Q. Given a pair P, Q such that the volume of P is less than or equal to that of Q, we are able to establish if Q contains P. This gives a stronger indication on the "dominance" of a relaxation with respect to another. The obtained results are reported in Table 4. As expected from results in Table 2, relaxation S_4 gives a polytope which is the most frequently contained in the others. This polytope is always contained in that corresponding to S_1 and S_2. It is sometimes equivalent to these polytopes, specifically in the 20% and in the 15% of the cases respectively. S_3 also gives a polytope always contained or equivalent to that given by S_1. This is interesting, being S_1 currently the most used relaxation in implementations. For some instances (5%, e.g. i_5), we find that all the polytopes are equivalent.

5 Application to well-known problems

We applied the obtained results to some well-known problems, namely the Molecular Distance Geometry Problem (MDGP) [17,42] and the Hartree–Fock Problem (HFP) [18,25]. Both problems, when cast in their mathematical programming formulation, are nonconvex polynomial NLPs with terms of degree up to four. Both can be solved to ε-optimality by means of the sBB algorithm.

5.1 Molecular distance geometry problem

The MDGP is the problem of finding an embedding in $\mathbb{R}^3$ of a weighted graph G such that all Euclidean distances between points in the embedding are the same as the corresponding edge weights in the graph. The main application is to find the three-dimensional structure of a molecule given a subset of the atomic distances (these are usually found using Nuclear Magnetic Resonance techniques) [6,32].

Consider an undirected graph $G = (V, E)$ with weights $d : E \to \mathbb{R}_+$, where V is the set of vertices (also called *atoms*), and E is the set of weighted edges (also called *inter-atomic*

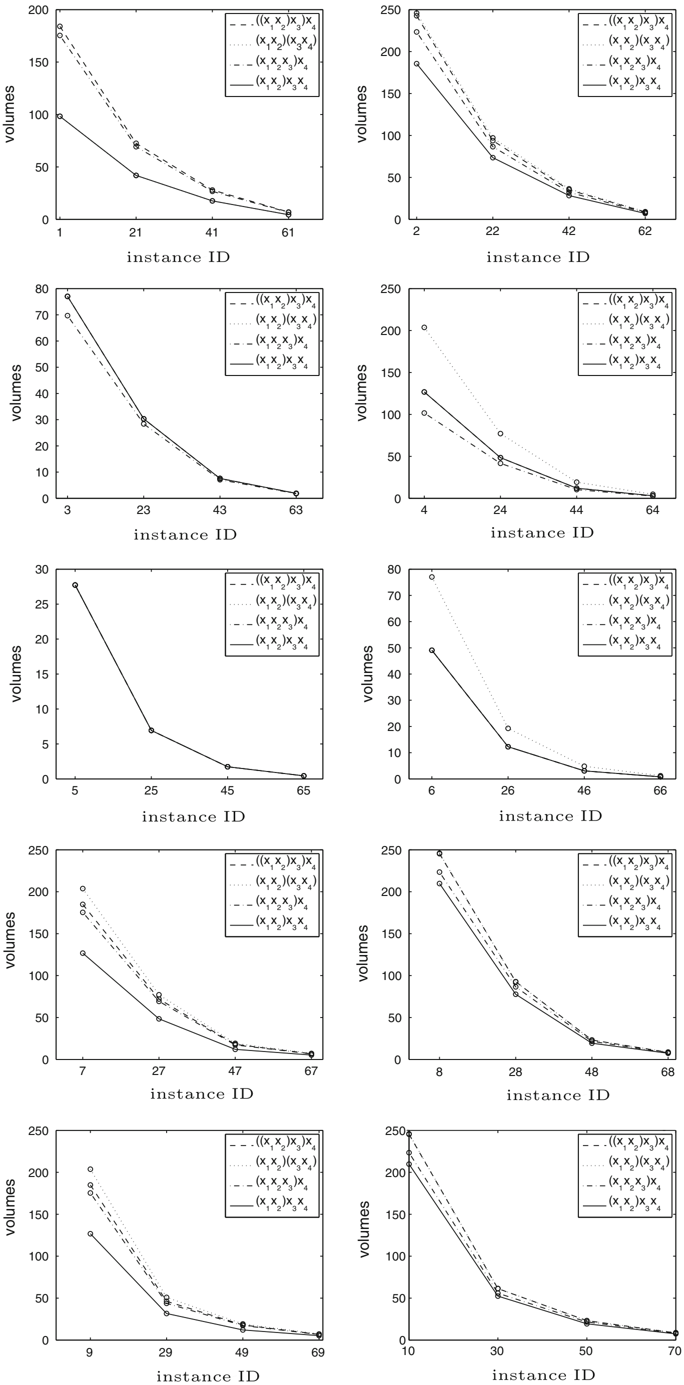

Fig. 1 Volumes of enveloping polytopes corresponding to S_1, S_2, S_3, S_4

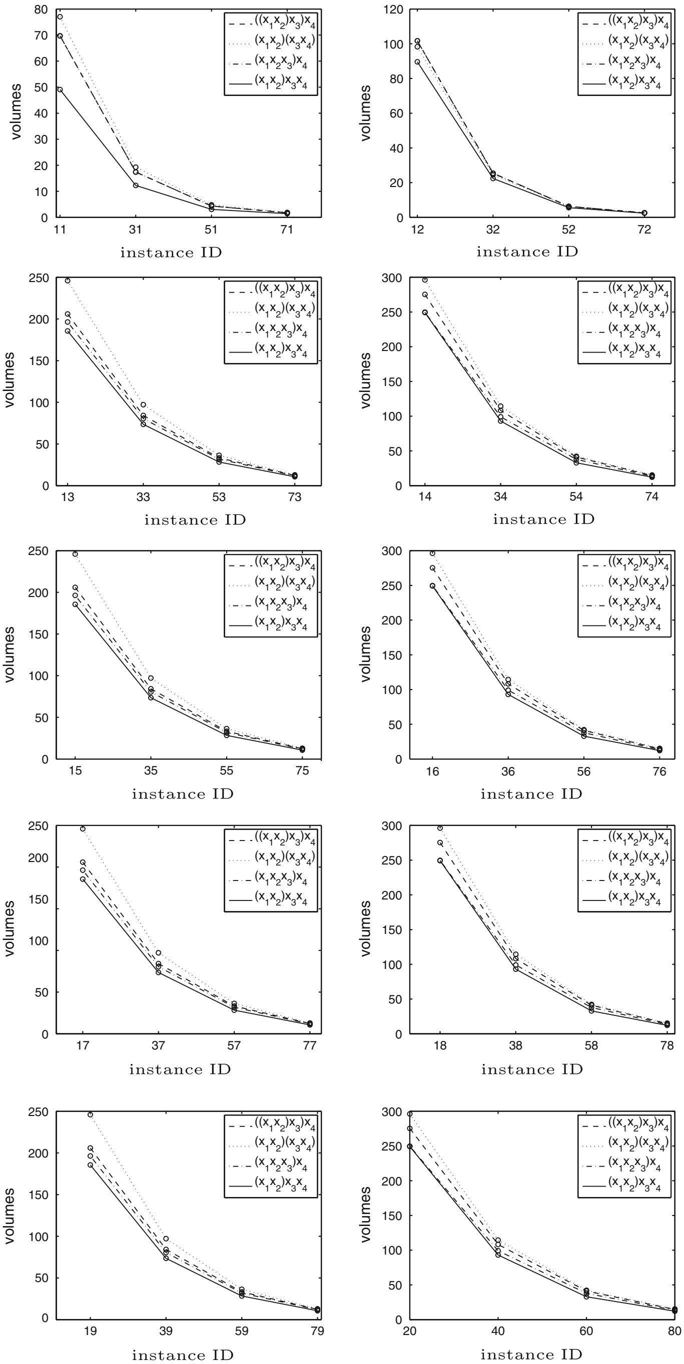

Fig. 2 Volumes of enveloping polytopes corresponding to S_1, S_2, S_3, S_4

Table 3 Ratios of the volumes of the enveloping polytopes projected onto $\mathbb{R}^5$ and the volume of the convex hull of the extreme points of $w = x_1x_2x_3x_4$

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\|\cdot\|$	*sign*	$\|\cdot\|$	*sign*	$\|\cdot\|$	*sign*	$\|\cdot\|$	*sign*				
i_1	2	+, +	2	+, +	2	−, +	2	−, +	2.0541	**1.0967**	1.9577	**1.0967**
i_2	2	+, +	2	+, +	2	−, +	2	−, −	1.5271	1.5473	1.4057	**1.1678**
i_3	2	+, +	2	−, +	2	−, +	2	−, +	1.5700	1.5700	**1.4203**	1.5700
i_4	2	+, +	2	−, +	2	−, +	2	−, −	1.4153	2.2738	**1.1349**	1.4153
i_5	2	−, +	2	−, +	2	−, +	2	−, +	*1.0000*	*1.0000*	*1.0000*	*1.0000*
i_6	2	−, +	2	−, +	2	−, +	2	−, −	**1.0000**	1.5700	**1.0000**	**1.0000**
i_7	2	+, +	2	−, +	2	−, −	2	−, +	2.0635	2.2738	1.9577	**1.4153**
i_8	2	+, +	2	−, +	2	−, −	2	−, −	1.5436	1.5473	1.4057	**1.3199**
i_9	2	−, +	2	−, −	2	−, −	2	−, +	2.0635	2.2738	1.9577	**1.4153**
i_10	2	−, +	2	−, −	2	−, −	2	−, −	1.5436	1.5473	1.4057	**1.3199**
i_11	2	−, +	2	−, +	2	−, −	2	−, +	1.4203	1.5700	1.4203	**1.0000**
i_12	2	−, +	2	−, +	2	−, −	2	−, −	1.1349	1.0967	1.1349	**1.0000**
i_13	2	+, +	2	+, +	2	+, +	2	−, +	1.2960	1.5473	1.2354	**1.1678**
i_14	2	+, +	2	+, +	2	+, +	2	−, −	1.4260	1.5344	1.2932	**1.2911**
i_15	2	+, +	2	+, +	2	−, −	2	−, +	1.2960	1.5473	1.2354	**1.1678**
i_16	2	+, +	2	+, +	2	−, −	2	−, −	1.4260	1.5344	1.2932	**1.2911**
i_17	2	+, +	2	−, −	2	−, −	2	−, +	1.2960	1.5473	1.2354	**1.1678**
i_18	2	+, +	2	−, −	2	−, −	2	−, −	1.4260	1.5344	1.2932	**1.2911**
i_19	2	−, −	2	−, −	2	−, −	2	−, +	1.2960	1.5473	1.2354	**1.1678**
i_20	2	−, −	2	−, −	2	−, −	2	−, −	1.4260	1.5344	1.2932	**1.2911**
i_21	1	+, +	2	+, +	2	−, +	2	−, +	1.8888	**1.0914**	1.8037	**1.0914**
i_22	1	+, +	2	+, +	2	−, +	2	−, −	1.4728	1.5235	1.3587	**1.1541**
i_23	1	+, +	2	−, +	2	−, +	2	−, +	1.3219	1.3219	**1.2372**	1.3219
i_24	1	+, +	2	−, +	2	−, +	2	−, −	1.2617	2.0091	**1.0861**	1.2617
i_25	1	−, +	2	−, +	2	−, +	2	−, +	*1.0000*	*1.0000*	*1.0000*	*1.0000*
i_26	1	−, +	2	−, +	2	−, +	2	−, −	**1.0000**	1.5700	**1.0000**	**1.0000**
i_27	1	+, +	2	−, +	2	−, −	2	−, +	1.8752	2.0091	1.8037	**1.2617**
i_28	1	+, +	2	−, +	2	−, −	2	−, −	1.4562	1.4534	1.3587	**1.2197**
i_29	1	−, +	2	−, −	2	−, −	2	−, +	2.0635	2.2738	1.9577	**1.4153**
i_30	1	−, +	2	−, −	2	−, −	2	−, −	1.5436	1.5473	1.4057	**1.3199**
i_31	1	−, +	2	−, +	2	−, −	2	−, +	1.4203	1.5700	1.4203	**1.0000**
i_32	1	−, +	2	−, +	2	−, −	2	−, −	1.1349	1.0967	1.1349	**1.0000**
i_33	1	+, +	2	+, +	2	+, +	2	−, +	1.3213	1.5235	1.2634	**1.1541**
i_34	1	+, +	2	+, +	2	+, +	2	−, −	1.4441	1.5282	1.3237	**1.2411**
i_35	1	+, +	2	+, +	2	−, −	2	−, +	1.3213	1.5235	1.2634	**1.1541**
i_36	1	+, +	2	+, +	2	−, −	2	−, −	1.4441	1.5282	1.3237	**1.2411**
i_37	1	+, +	2	−, −	2	−, −	2	−, +	1.3213	1.5235	1.2634	**1.1541**
i_38	1	+, +	2	−, −	2	−, −	2	−, −	1.4441	1.5282	1.3237	**1.2411**
i_39	1	−, −	2	−, −	2	−, −	2	−, +	1.3213	1.5235	1.2634	**1.1541**
i_40	1	−, −	2	−, −	2	−, −	2	−, −	1.4441	1.5282	1.3237	**1.2411**

Table 3 continued

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*				
i_41	1	+, +	1	+, +	2	−, +	2	−, +	1.7195	**1.0826**	1.6393	**1.0826**
i_42	1	+, +	1	+, +	2	−, +	2	−, −	1.4284	1.4804	1.3144	**1.1502**
i_43	1	+, +	1	−, +	2	−, +	2	−, +	1.3219	1.3219	**1.2372**	1.3219
i_44	1	+, +	1	−, +	2	−, +	2	−, −	1.2617	2.0091	**1.0861**	1.2617
i_45	1	−, +	1	−, +	2	−, +	2	−, +	*1.0000*	*1.0000*	*1.0000*	*1.0000*
i_46	1	−, +	1	−, +	2	−, +	2	−, −	**1.0000**	1.5700	**1.0000**	**1.0000**
i_47	1	+, +	1	−, +	2	−, −	2	−, +	1.8752	2.0091	1.8037	**1.2617**
i_48	1	+, +	1	−, +	2	−, −	2	−, −	1.4562	1.4534	1.3587	**1.2197**
i_49	1	−, +	1	−, −	2	−, −	2	−, +	1.8752	2.0091	1.8037	**1.2617**
i_50	1	−, +	1	−, −	2	−, −	2	−, −	1.4562	1.4534	1.3587	**1.2197**
i_51	1	−, +	1	−, +	2	−, −	2	−, +	1.4203	1.5700	1.4203	**1.0000**
i_52	1	−, +	1	−, +	2	−, −	2	−, −	1.1349	1.0967	1.1349	**1.0000**
i_53	1	+, +	1	+, +	2	+, +	2	−, +	1.3528	1.4804	1.2940	**1.1502**
i_54	1	+, +	1	+, +	2	+, +	2	−, −	1.4648	1.5032	1.3479	**1.1730**
i_55	1	+, +	1	+, +	2	−, −	2	−, +	1.3528	1.4804	1.2940	**1.1502**
i_56	1	+, +	1	+, +	2	−, −	2	−, −	1.4648	1.5032	1.3479	**1.1730**
i_57	1	+, +	1	−, −	2	−, −	2	−, +	1.3528	1.4804	1.2940	**1.1502**
i_58	1	+, +	1	−, −	2	−, −	2	−, −	1.4648	1.5032	1.3479	**1.1730**
i_59	1	−, −	1	−, −	2	−, −	2	−, +	1.3528	1.4804	1.2940	**1.1502**
i_60	1	−, −	1	−, −	2	−, −	2	−, −	1.4648	1.5032	1.3479	**1.1730**
i_61	1	+, +	1	+, +	1	−, +	2	−, +	1.7195	**1.0826**	1.6393	**1.0826**
i_62	1	+, +	1	+, +	1	−, +	2	−, −	1.4284	1.4804	1.3144	**1.1502**
i_63	1	+, +	1	−, +	1	−, +	2	−, +	1.3219	1.3219	**1.2372**	1.3219
i_64	1	+, +	1	−, +	1	−, +	2	−, −	1.2617	2.0091	**1.0861**	1.2617
i_65	1	−, +	1	−, +	1	−, +	2	−, +	*1.0000*	*1.0000*	*1.0000*	*1.0000*
i_66	1	−, +	1	−, +	1	−, +	2	−, −	**1.0000**	1.5700	**1.0000**	**1.0000**
i_67	1	+, +	1	−, +	1	−, −	2	−, +	1.6854	1.7756	1.6393	**1.3020**
i_68	1	+, +	1	−, +	1	−, −	2	−, −	1.3800	1.4469	1.3144	**1.2128**
i_69	1	−, +	1	−, −	1	−, −	2	−, +	1.6854	1.7756	1.6393	**1.3020**
i_70	1	−, +	1	−, −	1	−, −	2	−, −	1.3800	1.4469	1.3144	**1.2128**
i_71	1	−, +	1	−, +	1	−, −	2	−, +	1.2372	1.3219	1.2372	**1.0000**
i_72	1	−, +	1	−, +	1	−, −	2	−, −	1.0861	1.0914	1.0861	**1.0000**
i_73	1	+, +	1	+, +	1	+, +	2	−, +	1.3855	1.4481	1.3305	**1.1952**
i_74	1	+, +	1	+, +	1	+, +	2	−, −	1.4970	1.5668	1.3830	**1.2416**
i_75	1	+, +	1	+, +	1	−, −	2	−, +	1.3855	1.4481	1.3305	**1.1952**
i_76	1	+, +	1	+, +	1	−, −	2	−, −	1.4970	1.5668	1.3830	**1.2416**
i_77	1	+, +	1	−, −	1	−, −	2	−, +	1.3855	1.4481	1.3305	**1.1952**
i_78	1	+, +	1	−, −	1	−, −	2	−, −	1.4970	1.5668	1.3830	**1.2416**

Table 3 continued

#	x_1		x_2		x_3		x_4		S_1	S_2	S_3	S_4
	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*	$\lvert\cdot\rvert$	*sign*				
i_79	1	−, −	1	−, −	1	−, −	2	−, +	1.3855	1.4481	1.3305	**1.1952**
i_80	1	−, −	1	−, −	1	−, −	2	−, −	1.4970	1.5668	1.3830	**1.2416**

For each x_i, $|\cdot| = x_i^U - x_i^L$ and *sign* is the sign of x_i^L, x_i^U respectively. We remark that the case of all positive signs is equivalent to that of all negative signs

The best values are reported in bold face values are written in italic when they are equal for all the four considerable cases

distances). Let $N = |V|$ and $d_{ij} = d(\{i, j\})$ for $\{i, j\} \in E$. A solution of the MDGP is a set of points $x_1, \ldots, x_N \in \mathbb{R}^3$ satisfying

$$\forall \{i, j\} \in E \quad ||x_i - x_j|| = d_{ij}. \tag{2}$$

Each 3-vector x_i has components (x_{i1}, x_{i2}, x_{i3}), and we indicate the vector sequence $(x_1, \ldots, x_N)$ by x. The MDGP can be naturally cast as a continuous non-convex optimization problem $\min_x f(x)$ by minimizing the sum of the squared errors over each Eq. (2):

$$f(x) = \sum_{\{i,j\} \in E} (||x_i - x_j||^2 - d_{ij}^2)^2. \tag{3}$$

Because each Eq. (2) must be satisfied, a candidate point x is a solution of the MDGP if and only if $f(x) = 0$. Note that (3) has a large number of local minima, so this is a difficult global-optimization problem.

When expanded, a typical term $(||x_i - x_j||^2 - d_{ij}^2)^2$ of the MDGP objective function sum involves many quartic terms. We employ randomly generated MDGP instances as described in [15].

5.2 Hartree–Fock problem

The quantum behaviour of atoms and molecules, in the absence of relativistic effects and any external time-dependent perturbations, is determined by the time-independent Schrödinger equation $H\Psi_n = E\Psi_n$, where H (the Hamiltonian operator of the system) represents the total energy (kinetic + potential) of all the particles of the system. Analytical solutions for this equation are only possible for very simple systems. Hence, for the majority of problems of interest, one has to rely on some approximate model. In the Hartree–Fock (HF) model, the electrons in atoms and molecules move independently of each other, the motion of each one of the electrons being determined by the attractive electrostatic potential of the nuclei and by a repulsive average field due to all the other electrons of the system. In this model, the approximate solutions Φ_n of the Schrödinger equation are anti-symmetrized products of one-electron wave functions $\{\varphi_i\}$ (also called orbitals), which are solutions of the Hartree–Fock (HF) equations for the system under study. Since each orbital φ_i can be expanded in a complete basis set $\{\chi_s\}_{s=1}^{\infty}$, we can transform the HF equations into a less cumbersome form by only considering a finite subset $\{\chi_s \mid s \leq b\}$ of the basis, and we use it to approximate the orbitals. We define the Hartree–Fock Problem (HFP) as the problem of finding a set of coefficients c_{si} such that the $\bar{\varphi}_i := \sum_{s \leq b} c_{si} \chi_s$ are the best possible approximations of the spatial orbitals. Thus, the decision variables of this mathematical programming problem are the coefficients c_{si}. The objective function (quality of the approximation) is given by a

Table 4 Relative containments of the enveloping polytopes corresponding to S_1, S_2, S_3, S_4 projected onto $\mathbb{R}^5$

#	x_1		x_2		x_3		x_4		Containments
	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	
i_1	2	$+,+$	2	$+,+$	2	$-,+$	2	$-,+$	$S_2 \equiv S_4 \subseteq S_1,\ S3 \subseteq S1$
i_2	2	$+,+$	2	$+,+$	2	$-,+$	2	$-,-$	$S3 \subseteq S1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_3	2	$+,+$	2	$-,+$	2	$-,+$	2	$-,+$	$S_3 \subseteq S_1 \equiv S_2 \equiv S_4$
i_4	2	$+,+$	2	$-,+$	2	$-,+$	2	$-,-$	$S_3 \subseteq S_1 \equiv S_4 \subseteq S_2$
i_5	2	$-,+$	2	$-,+$	2	$-,+$	2	$-,+$	$S_1 \equiv S_2 \equiv S_3 \equiv S_4$
i_6	2	$-,+$	2	$-,+$	2	$-,+$	2	$-,-$	$S_1 \equiv S_3 \equiv S_4 \subseteq S_2$
i_7	2	$+,+$	2	$-,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_8	2	$+,+$	2	$-,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_9	2	$-,+$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_10	2	$-,+$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_11	2	$-,+$	2	$-,+$	2	$-,-$	2	$-,+$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_12	2	$-,+$	2	$-,+$	2	$-,-$	2	$-,-$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_13	2	$+,+$	2	$+,+$	2	$+,+$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_14	2	$+,+$	2	$+,+$	2	$+,+$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_15	2	$+,+$	2	$+,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_16	2	$+,+$	2	$+,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_17	2	$+,+$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_18	2	$+,+$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_19	2	$-,-$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_20	2	$-,-$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_21	1	$+,+$	2	$+,+$	2	$-,+$	2	$-,+$	$S_2 \equiv S_4 \subseteq S_1,\ S3 \subseteq S1$
i_22	1	$+,+$	2	$+,+$	2	$-,+$	2	$-,-$	$S3 \subseteq S1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_23	1	$+,+$	2	$-,+$	2	$-,+$	2	$-,+$	$S_3 \subseteq S_1 \equiv S_2 \equiv S_4$
i_24	1	$+,+$	2	$-,+$	2	$-,+$	2	$-,-$	$S_3 \subseteq S_1 \equiv S_4 \subseteq S_2$
i_25	1	$-,+$	2	$-,+$	2	$-,+$	2	$-,+$	$S_1 \equiv S_2 \equiv S_3 \equiv S_4$
i_26	1	$-,+$	2	$-,+$	2	$-,+$	2	$-,-$	$S_1 \equiv S_3 \equiv S_4 \subseteq S_2$
i_27	1	$+,+$	2	$-,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_28	1	$+,+$	2	$-,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_29	1	$-,+$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_30	1	$-,+$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_31	1	$-,+$	2	$-,+$	2	$-,-$	2	$-,+$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_32	1	$-,+$	2	$-,+$	2	$-,-$	2	$-,-$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_33	1	$+,+$	2	$+,+$	2	$+,+$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_34	1	$+,+$	2	$+,+$	2	$+,+$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_35	1	$+,+$	2	$+,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_36	1	$+,+$	2	$+,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_37	1	$+,+$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_38	1	$+,+$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_39	1	$-,-$	2	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_40	1	$-,-$	2	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$

Table 4 continued

#	x_1		x_2		x_3		x_4		Containments
	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	$\lvert\cdot\rvert$	$sign$	
i_41	1	$+,+$	1	$+,+$	2	$-,+$	2	$-,+$	$S_2 \equiv S_4 \subseteq S_1,\ S3 \subseteq S1$
i_42	1	$+,+$	1	$+,+$	2	$-,+$	2	$-,-$	$S3 \subseteq S1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_43	1	$+,+$	1	$-,+$	2	$-,+$	2	$-,+$	$S_3 \subseteq S_1 \equiv S_2 \equiv S_4$
i_44	1	$+,+$	1	$-,+$	2	$-,+$	2	$-,-$	$S_3 \subseteq S_1 \equiv S_4 \subseteq S_2$
i_45	1	$-,+$	1	$-,+$	2	$-,+$	2	$-,+$	$S_1 \equiv S_2 \equiv S_3 \equiv S_4$
i_46	1	$-,+$	1	$-,+$	2	$-,+$	2	$-,-$	$S_1 \equiv S_3 \equiv S_4 \subseteq S_2$
i_47	1	$+,+$	1	$-,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_48	1	$+,+$	1	$-,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_49	1	$-,+$	1	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_50	1	$-,+$	1	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_51	1	$-,+$	1	$-,+$	2	$-,-$	2	$-,+$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_52	1	$-,+$	1	$-,+$	2	$-,-$	2	$-,-$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_53	1	$+,+$	1	$+,+$	2	$+,+$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_54	1	$+,+$	1	$+,+$	2	$+,+$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_55	1	$+,+$	1	$+,+$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_56	1	$+,+$	1	$+,+$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_57	1	$+,+$	1	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_58	1	$+,+$	1	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_59	1	$-,-$	1	$-,-$	2	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_60	1	$-,-$	1	$-,-$	2	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_61	1	$+,+$	1	$+,+$	1	$-,+$	2	$-,+$	$S_2 \equiv S_4 \subseteq S_1,\ S3 \subseteq S1$
i_62	1	$+,+$	1	$+,+$	1	$-,+$	2	$-,-$	$S3 \subseteq S1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_63	1	$+,+$	1	$-,+$	1	$-,+$	2	$-,+$	$S_3 \subseteq S_1 \equiv S_2 \equiv S_4$
i_64	1	$+,+$	1	$-,+$	1	$-,+$	2	$-,-$	$S_3 \subseteq S_1 \equiv S_4 \subseteq S_2$
i_65	1	$-,+$	1	$-,+$	1	$-,+$	2	$-,+$	$S_1 \equiv S_2 \equiv S_3 \equiv S_4$
i_66	1	$-,+$	1	$-,+$	1	$-,+$	2	$-,-$	$S_1 \equiv S_3 \equiv S_4 \subseteq S_2$
i_67	1	$+,+$	1	$-,+$	1	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_68	1	$+,+$	1	$-,+$	1	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_69	1	$-,+$	1	$-,-$	1	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_70	1	$-,+$	1	$-,-$	1	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_71	1	$-,+$	1	$-,+$	1	$-,-$	2	$-,+$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_72	1	$-,+$	1	$-,+$	1	$-,-$	2	$-,-$	$S_4 \subseteq S_1 \equiv S_3,\ S_4 \subseteq S_2$
i_73	1	$+,+$	1	$+,+$	1	$+,+$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_74	1	$+,+$	1	$+,+$	1	$+,+$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_75	1	$+,+$	1	$+,+$	1	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_76	1	$+,+$	1	$+,+$	1	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_77	1	$+,+$	1	$-,-$	1	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_78	1	$+,+$	1	$-,-$	1	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_79	1	$-,-$	1	$-,-$	1	$-,-$	2	$-,+$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$
i_80	1	$-,-$	1	$-,-$	1	$-,-$	2	$-,-$	$S_3 \subseteq S_1,\ S_4 \subseteq S_1,\ S_4 \subseteq S_2$

For each x_i, $|\cdot| = x_i^U - x_i^L$ and $sign$ is the sign of x_i^L, x_i^U

suitable energy function E associated with the approximating set $\{\bar{\varphi}_i\}$, which is constrained to be an orthonormal set:

$$\begin{aligned} \min\ & E(c) \\ \text{s.t.}\ & \left\langle \sum_{s\le b} c_{si}\chi_s\ ,\ \sum_{s\le b} c_{sj}\chi_s \right\rangle = \delta_{ij} \quad \forall i \le j \le n \\ & c^L \le c \le c^U\ , \end{aligned}$$

where δ is the Kronecker delta function. We can readily see that the orthonormality constraints are quadratic in the decision variables c. Moreover, it turns out that for closed-shell atomic systems, the objective function is a quartic expression of the decision variables c.

5.3 Bound evaluation algorithm

The natural application of tight lower bounds computed through a convex relaxation is within the sBB algorithm. In order to quickly assess the quality of our proposed alternative bound for quadrilinear terms on the MDGP and HFP without having to implement a full sBB framework, we implemented (using AMPL [8]) a simplified "partial sBB" algorithm which, at each branching step, only records the most promising node and discards the other, thus exploring a single branch up to a leaf. This corresponds to well-known "diving heuristics" employed in integer linear programming. In Algorithm 1, P is a mathematical program defined as $\min\{f(x) \mid g(x) \le 0,\ x^L \le x \le x^U\}$ with decision variables $x \in \mathbb{R}^n \cap [x^L, x^U]$, objective function $f : \mathbb{R}^n \to \mathbb{R}$ and constraints $g : \mathbb{R}^n \to \mathbb{R}^m$.

Algorithm 1 can be considered as a heuristic solution algorithm for nonconvex NLPs, whose purpose is that of assessing the quality of a lower bound instead of that of the incumbent. We use a very simple branching strategy (the variable index i maximizing $|x_i^* - \bar{x}_i|$) and terminate either on iteration limit or on reaching a node that is infeasible or that contains the global optimum.

P is solved by SNOPT [10], and its (linear) convex relaxations R_0, R_1 by CPLEX [13]. R_0, R_1 are constructed automatically (by the ROSE software [24]) in the four different ways corresponding to S_1–S_4. The algorithm implemented in ROSE for constructing the convex relaxation is similar to the symbolic reformulation algorithm in [38]. First, each nonlinear term is replaced by an additional variable, and a defining constraint "additional variable = nonlinear term" is added to the problem. In a second stage, each defining constraint is replaced by a convex relaxation. The different associativity precedences in S_1–S_4 yield different defining constraints and in turn different convex relaxations. For the sake of completeness, in the following we report the convex envelopes for piecewise/convex/concave monomials with the respective linear relaxations.

– The concave univariate function $f(x_j)$ is replaced by a variable x_i and two inequalities are added to the problem relaxation: the function itself and the secant:

$$x_i \le f(x_j) \tag{4}$$

$$x_i \ge f(x_j^L) + \frac{f(x_j^U) - f(x_j^L)}{x_j^U - x_j^L}(x_j - x_j^L). \tag{5}$$

Constraint (4) is a nonlinear over-estimator which is replaced in our implementation by a pre-determined number of tangents to f at various given points.

Algorithm 1 The partial depth-first sBB algorithm

Input an NLP P and an iteration `limit`
Let `counter` $\leftarrow 0$
Solve P locally to find x^* with objective function value f^* (*incumbent*)
Construct a (linear) convex relaxation R of P
Solve R to find an optimum $\bar{x}$ with function value $\bar{f}$ (*bound*)
Choose the branching variable i with branching point $\bar{x}_i$
Let `termination` $\leftarrow$ `false`
while `!termination` **do**
  Let P_0 be defined as P with the added constraint $x_i^L \leq x_i \leq \bar{x}_i$
  Let P_1 be defined as P with the added constraint $\bar{x}_i \leq x_i \leq x_i^U$
  For $k \in \{0, 1\}$, let R_k be the convex relaxation of P_k
  For $k \in \{0, 1\}$, let $\bar{x}^k$ be the optimum of R_k with value $\bar{f}_k$
  Let $\ell = \text{argmin}(\bar{f}_0, \bar{f}_1)$ (*best lower bound*)
  if $\bar{f}_\ell > f^*$ **then**
    Let `termination` $\leftarrow$ `true` (*node cannot improve*)
  else
    if $\bar{f}_\ell > \bar{f}$ **then**
      Let $\bar{f} \leftarrow \bar{f}_\ell$ (*overall bound improvement*)
    end if
  end if
  if x^* is infeasible in P_ℓ **then**
    Solve P_ℓ locally to find x' with value f'
    if $f' < f^*$ **then**
      Let $f^* \leftarrow f'$ and $x^* \leftarrow x'$ (*incumb. improv.*)
    end if
  end if
  if $|f^* - \bar{f}| < \varepsilon$ or `counter` $>$ `limit` **then**
    Let `termination` $\leftarrow$ `true` (*global optimum*)
  end if
  if $\bar{f}_\ell = \infty$ **then**
    Let `termination` $\leftarrow$ `true` (*infeasible node*)
  end if
  Let $P \leftarrow P_\ell$ and $\bar{x} \leftarrow \bar{x}^\ell$
  Update the branching variable i and branching point $\bar{x}_i$
  Increase `counter`
end while

– The convex univariate function $f(x_j)$ is replaced by a variable x_i and two inequalities are added to the problem relaxation:

$$x_i \leq f(x_j^L) + \frac{f(x_j^U) - f(x_j^L)}{x_j^U - x_j^L}(x_j - x_j^L) \tag{6}$$

$$x_i \geq f(x_j). \tag{7}$$

Constraint (7) is a nonlinear under-estimator which is replaced in our implementation by a pre-determined number of tangents to f at various given points.

– The term x_j^{2k} for any $k \in \mathbb{N}$ is replaced by a variable x_i and treated as a convex univariate function.
– The term x_j^{2k+1} for any $k \in \mathbb{N}$ is replaced by a variable x_i and can be convex, concave, or piecewise convex and concave with a turning point at 0. If the range of x_j does not include 0, the function is convex or concave and falls into a category described above.

Table 5 Results obtained by running Algorithm 1 on MDGP and HFP instances

Instance	S_1	S_2	S_3	S_4
lavor6	−1006.75	−1839.21	−1006.75	**−990.167**
lavor7	−1285.67	−1279.88	**−1175.95**	−1216.91
lavor8	−1711.27	−1694.56	−1718.41	**−1671.09**
lavor10	−3149.29	−3172.05	−3007.41	**−2755.04**
beryllium	−24.2038	**−22.4639**	−24.2038	−23.8677
neon	−683.034	−619.238	−651.045	**−508.061**

The best values are reported in bold face

Otherwise, the convex/concave envelope is given in [23]; a tight linear relaxation is given by:

$$x_i \geq (x_j^L)^{2k+1}\left(1 + T_k\left(\frac{x_j}{x_j^L} - 1\right)\right) \tag{8}$$

$$x_i \leq (x_j^U)^{2k+1}\left(1 + T_k\left(\frac{x_j}{x_j^U} - 1\right)\right) \tag{9}$$

$$x_i \geq (2k+1)(x_j^U)^{2k}x_j - 2k(x_j^U)^{2k+1} \tag{10}$$

$$x_i \leq (2k+1)(x_j^L)^{2k}x_j - 2k(x_j^L)^{2k+1}, \tag{11}$$

where $T_k = \frac{t_k^{2k+1}-1}{t_k-1}$ and the coefficients t_k are given for the first few values of k's in the table below (see [23] for details).

k	t_k	k	t_k
1	−0.5000000000	6	−0.7721416355
2	−0.6058295862	7	−0.7921778546
3	−0.6703320476	8	−0.8086048979
4	−0.7145377272	9	−0.8223534102
5	−0.7470540749	10	−0.8340533676

5.4 Computational results

Table 5 shows the results obtained by running Algorithm 1 on four MDGP and two HFP instances. We report the lower bounds obtained with the four relaxations as per S_1–S_4. On all the MDGP instances the best lower bound is that obtained with a relaxation involving a trilinear envelope. In particular, S_4 gives the tightest bound for most cases, and this bound is significantly better than the values obtained with bilinear relaxations on the largest MDGP instance. On the first HFP instance, we found a good bound using a relaxation based on bilinear envelopes. We never found that the first relaxation, which is currently the most used in sBB implementations, gives the best bounds. The CPU time taken to solve each of the different relaxations is about the same.

These results confirm the results of the previous sections and suggest that they can be used to configure a sBB algorithm to be efficiently applied to mathematical programs containing quadrilinear terms.

6 Conclusion

This paper focuses on convex relaxations of quadrilinear terms $x_1x_2x_3x_4$. We computationally and mathematically evaluated four linear relaxations, showing that the tightest one can be obtained by combining the convex envelope of a trilinear term and that of a bilinear term. Our results can be exploited in a sBB algorithm to compute tight bounds.

Acknowledgments We thank Komei Fukuda for helping us to use his code cdd to compute, in exact arithmetic, projections of polytopes and extreme point representations of polytopes given via inequalities. We thank Jesús De Loera for advising us to use David Avis' code lrs, and in turn we thank David Avis for lrs, which we used to compute exact volumes of polytopes given via extreme points. S. Cafieri and L. Liberti gratefully acknowledge financial support under ANR grant 07-JCJC-0151.

References

1. Adjiman, C.S.: Global optimization techniques for process systems engineering. PhD thesis, Princeton University (1998)
2. Adjiman, C.S., Dallwig, S., Floudas, C.A., Neumaier, A.: A global optimization method, αBB, for general twice-differentiable constrained NLPs: I. Theoretical advances. Comput. Chem. Eng. **22**(9), 1137–1158 (1998)
3. Al-Khayyal, F.A., Falk, J.E.: Jointly constrained biconvex programming. Math. Oper. Res. **8**(2), 273–286 (1983)
4. Avis, D.: lrs. cgm.cs.mcgill.ca/~avis/C/lrs.html
5. Belotti, P., Lee, J., Liberti, L., Margot, F., Wächter, A.: Branching and bounds tightening techniques for non-convex MINLP. Optim. Methods Softw. **24**(4), 597–634 (2009)
6. Crippen, G.M., Havel, T.F.: Distance Geometry and Molecular Conformation. Wiley, New York (1988)
7. Floudas, C.: Personal communication (2007)
8. Fourer, R., Gay, D.: The AMPL Book. Duxbury Press, Pacific Grove (2002)
9. Fukuda, K.: cdd. www.ifor.math.ethz.ch/~fukuda/cdd_home/cdd.html
10. Gill, P.E.: User's guide for SNOPT version 7. Systems Optimization Laboratory, Stanford University, California (2006)
11. Gounaris, C.E., Floudas, C.A.: Tight convex underestimators for C^2-continuous problems: I. Multivariate functions. J. Glob. Optim. **42**(1), 69–89 (2008)
12. Gounaris, C.E., Floudas, C.A.: Tight convex underestimators for C^2-continuous problems: I. Univariate functions. J. Glob. Optim. **42**(1), 51–67 (2008)
13. ILOG: *ILOG CPLEX 11.0 User's Manual*. (ILOG S.A. Gentilly, France 2008)
14. Jach, M., Michaels, D., Weismantel, R.: The convex envelope of (n-1)-convex functions. SIAM J. Optim. **19**(3), 1451–1466 (2008)
15. Lavor, C.: On generating instances for the molecular distance geometry problem. In Liberti and Maculan [22], pp 405–414
16. Lavor, C., Liberti, L., Maculan, N. : Computational experience with the molecular distance geometry problem. In: Pintér, J. (ed.) Global Optimization: Scientific and Engineering Case Studies, pp. 213–225. Springer, Berlin (2006)
17. Lavor, C., Liberti, L., Maculan, N. : Molecular distance geometry problem. In: Floudas, C., Pardalos, P. (eds.) Encyclopedia of Optimization, pp. 2305–2311. Springer, New York (2008)
18. Lavor, C., Liberti, L., Maculan, N., Chaer Nascimento, M.A.: Solving Hartree–Fock systems with global optimization metohds. Europhys. Lett. **5**(77), 50006p1–50006p5 (2007)
19. Lee, J., Morris, W.D. Jr.: Geometric comparison of combinatorial polytopes. Discrete Appl. Math. **55**(2), 163–182 (1994)
20. Liberti, L.: Comparison of convex relaxations for monomials of odd degree. In: Tseveendorj, I., Pardalos, P.M., Enkhbat, R. (eds.) Optimization and Optimal Control, World Scientific, Singapore (2003)
21. Liberti, L.: Writing global optimization software. In Liberti and Maculan [22], pp 211–262 (2006)
22. Liberti, L., Maculan, N. (eds.): Global Optimization: From Theory to Implementation. Springer, Berlin (2006)
23. Liberti, L., Pantelides, C.C.: Convex envelopes of monomials of odd degree. J. Glob. Optim. **25**, 157–168 (2003)

24. Liberti, L., Cafieri, S., Tarissan, F.: Reformulations in mathematical programming: a computational approach. In: Abraham, A., Hassanien, A.-E., Siarry, P., Engelbrecht, A. (eds.) Foundations on Computational Intelligence, vol. 3, Studies in Computational Intelligence volume 203, pp. 153–234. Springer, Berlin (2009)
25. Liberti, L., Lavor, C., Chaer Nascimento, M.A., Maculan, N.: Reformulation in mathematical programming: an application to quantum chemistry. Discrete Appl. Math. **157**(6), 1309–1318 (2009)
26. Liberti, L., Lavor, C., Maculan, N., Marinelli, F.: Double variable neighbourhood search with smoothing for the molecular distance geometry problem. J. Glob. Optim. **43**, 207–218 (2009)
27. Liberti, L., Tsiakis, P., Keeping, B., Pantelides, C.C.: $oo\mathcal{OPS}$. Centre for Process Systems Engineering, Chemical Engineering Department, Imperial College, London, UK (2001)
28. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I — Convex underestimating problems. Math. Program. **10**, 146–175 (1976)
29. Meyer, C.A., Floudas, C.A.: Trilinear monomials with positive or negative domains: Facets of the convex and concave envelopes. In: Floudas, C.A., Pardalos, P.M. (eds.) Frontiers in Global Optimization., pp. 327–352. Kluwer, Amsterdam (2003)
30. Meyer, C.A., Floudas, C.A.: Trilinear monomials with mixed sign domains: facets of the convex and concave envelopes. J. Glob. Optim. **29**, 125–155 (2004)
31. Mosses, P.: Denotational semantics. In: van Leeuwen, J. (ed.) Handbook of Theoretical Computer Science, vol. B, pp. 575–631. Elsevier, Amsterdam (1990)
32. Neumaier, A.: Molecular modeling of proteins and mathematical prediction of protein structure. SIAM Rev **39**, 407–460 (1997)
33. Rikun, A.: A convex envelope formula for multilinear functions. J. Glob. Optim. **10**(4), 425–437 (1997)
34. Ryoo, H.S., Sahinidis, N.V.: A branch-and-reduce approach to global optimization. J. Glob. Optim. **8**(2), 107–138 (1996)
35. Ryoo, H.S., Sahinidis, N.V.: Analysis of bounds for multilinear functions. J. Glob. Optim. **19**, 403–424 (2001)
36. Sahinidis, N.V., Tawarmalani, M.: BARON 7.2.5: Global Optimization of Mixed-Integer Nonlinear Programs, User's Manual (2005)
37. Smith, E.M.B.: on the optimal design of continuous processes. PhD thesis, Imperial College of Science, Technology and Medicine, University of London, October (1996)
38. Smith, E.M.B., Pantelides, C.C.: A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex MINLPs. Comput. Chem. Eng. **23**, 457–478 (1999)
39. Tardella, F.: Existence and sum decomposition of vertex polyhedral convex envelopes. Optim. Lett. **2**, 363–375 (2008)
40. Tawarmalani, M., Sahinidis, N.: Convex extensions and envelopes of semi-continuous functions. Math. Program. **93**(2), 247–263 (2002)
41. Tawarmalani, M., Sahinidis, N.V.: Global optimization of mixed integer nonlinear programs: a theoretical and computational study. Math. Program. **99**, 563–591 (2004)
42. Yoon, J.-M., Gad, Y., Wu, Z.: Mathematical modeling of protein structure using distance geometry. Technical Report TR00-24, Dept. Comput. Applied Maths, Rice University, Houston (2000)