J Glob Optim (2009) 43:445–458
DOI 10.1007/s10898-007-9195-4

Fast construction of constant bound functions for sparse polynomials

Andrew Paul Smith

Received: 18 May 2007 / Accepted: 9 June 2007 / Published online: 31 July 2007

Abstract A new method for the representation and computation of Bernstein coefficients of multivariate polynomials is presented. It is known that the coefficients of the Bernstein expansion of a given polynomial over a specified box of interest tightly bound the range of the polynomial over the box. The traditional approach requires that all Bernstein coefficients are computed, and their number is often very large for polynomials with moderately-many variables. The new technique detailed represents the coefficients implicitly and uses lazy evaluation so as to render the approach practical for many types of non-trivial sparse polynomials typically encountered in global optimization problems; the computational complexity becomes nearly linear with respect to the number of terms in the polynomial, instead of exponential with respect to the number of variables. These range-enclosing coefficients can be employed in a branch-and-bound framework for solving constrained global optimization problems involving polynomial functions, either as constant bounds used for box selection, or to construct affine underestimating bound functions. If such functions are used to construct relaxations for a global optimization problem, then sub-problems over boxes can be reduced to linear programming problems, which are easier to solve. Some numerical examples are presented and the software used is briefly introduced.

Keywords Constrained global optimization · Bernstein polynomials · Multivariate polynomials · Lazy evaluation · Interval arithmetic · Relaxation · Bound functions

1 Introduction

Whenever a branch-and-bound approach is used to solve a constrained global optimization problem, it is crucial to be able to compute tight bounds for the ranges of the objective and constraint functions over subboxes, in order to efficiently resolve the associated

A. P. Smith (✉)
Institute of Applied Research, University of Applied Sciences (HTWG) Konstanz,
Postfach 100543, 78405 Konstanz, Germany
e-mail: smith@htwg-konstanz.de

subproblems. Often, relaxations are generated by replacing the objective and constraint functions by corresponding lower bound functions. The relaxed subproblem with its set of feasible solutions constitutes a simpler type of problem (for example, a linear programming problem) whose solution provides a lower bound for the solution of the subproblem. In all cases, the computation of a good-quality convex lower bound function is important. The uniformly best underestimating convex functions are convex envelopes, cf. [1,5,22]. Affine lower bound functions are simpler to compute and work with, preserving basic shape information and yielding linear programming problems. A sequence of diverse methods for computing such affine bound functions for polynomials based upon Bernstein expansion has been proposed, cf. [7–10] . Constant bound functions, the simplest of the three types, are used frequently when interval computation techniques are applied to global optimization, cf. [12,14,18].

At a typical point during the execution of a branch-and-bound method to solve a global optimization problem involving polynomial functions, we have a polynomial

$$p(x) = \sum_{i=0}^{l} a_i x^i, \quad x = (x_1, \ldots, x_n), \tag{1}$$

in n variables, $x_1, \ldots, x_n$, of degree $l = (l_1, \ldots, l_n)$, and a box

$$X := [\underline{x}_1, \overline{x}_1] \times \cdots \times [\underline{x}_n, \overline{x}_n]. \tag{2}$$

This paper addresses the question of how to determine a tight outer approximation for $p(X)$, the range of p over X, in a timely fashion. Such bounds can be determined by utilising the coefficients of the expansion of the given polynomial into Bernstein polynomials.

The organisation of the paper is as follows: properties of Bernstein polynomials are introduced in Sect. 2, with a brief discussion of complexity, which motivates this work. Section 3 establishes some important results concerning the Bernstein coefficients of monomials. The implicit Bernstein form is introduced in Sect. 4, together with an efficient range computation algorithm. A brief discussion of the software and some numerical results are in Sect. 5. Directions for future work conclude the paper.

1.1 Notation

A shorthand notation for multiindices is used: a vector $(i_1, \ldots, i_n)^T$, where the n components are non-negative integers, is abbreviated to i. The vector 0 denotes the multiindex with all components equal to 0. Comparisons are used entrywise. The arithmetic operators on multiindices are also defined componentwise such that $i \odot l := (i_1 \odot l_1, \ldots, i_n \odot l_n)^T$, for $\odot = +, -, \times$, and / (with $l > 0$). Likewise, the minimum or maximum of two multiindices is formed componentwise. For $x \in \mathbf{R}^n$ its multipowers are

$$x^i := \prod_{\mu=1}^{n} x_\mu^{i_\mu}. \tag{3}$$

For the sum we use the notation

$$\sum_{i=0}^{l} := \sum_{i_1=0}^{l_1} \cdots \sum_{i_n=0}^{l_n} . \tag{4}$$

The generalised binomial coefficient is defined by

$$\binom{l}{i} := \prod_{\mu=1}^{n} \binom{l_\mu}{i_\mu}. \tag{5}$$

2 Bernstein expansion

An n-variate polynomial p (1) can be represented over $I = [0, 1]^n$ as

$$p(x) = \sum_{i=0}^{l} b_i B_i(x), \tag{6}$$

where

$$B_i(x) = \binom{l}{i} x^i (1 - x)^{l-i} \tag{7}$$

and the so-called *Bernstein coefficients* b_i (of degree l) are given by

$$b_i = \sum_{j=0}^{i} \frac{\binom{i}{j}}{\binom{l}{j}} a_j, \quad 0 \le i \le l. \tag{8}$$

In Sect. 3 we will allow that the degree of p is given by some r, where $r < l$. In this case, formulae (6)–(8) remain in force with the convention that $a_j = 0$ if $j \not\le r$.

2.1 Generalised Bernstein coefficients

Although the case of the unit box I may be considered without loss of generality, since any non-empty box in $\mathbf{R}^n$ can be mapped affinely thereupon, we consider here the general case. The Bernstein coefficients b_i of degree $l = (l_1, \ldots, l_n)$ over a box X (2) are given by

$$b_i = \sum_{j=0}^{i} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \sum_{k=j}^{l} \binom{k}{j} \underline{x}^{k-j} a_k, \quad 0 \le i \le l. \tag{9}$$

2.2 Properties

The essential property of the Bernstein expansion is the *range enclosing property*, namely that the range of p over X is contained within the interval spanned by the minimum and maximum Bernstein coefficients:

$$\min_i \{b_i\} \le p(x) \le \max_i \{b_i\}, \quad x \in X. \tag{10}$$

This property is in fact merely a corollary of the *convex hull property*

$$\left\{ \begin{pmatrix} x \\ p(x) \end{pmatrix} : x \in I \right\} \subseteq conv \left\{ \begin{pmatrix} i/l \\ b_i \end{pmatrix} : 0 \le i \le l \right\}, \tag{11}$$

where the convex hull is denoted by $conv$.

Figure 1 illustrates the convex hull property for a univariate polynomial of degree 5 over the unit interval.

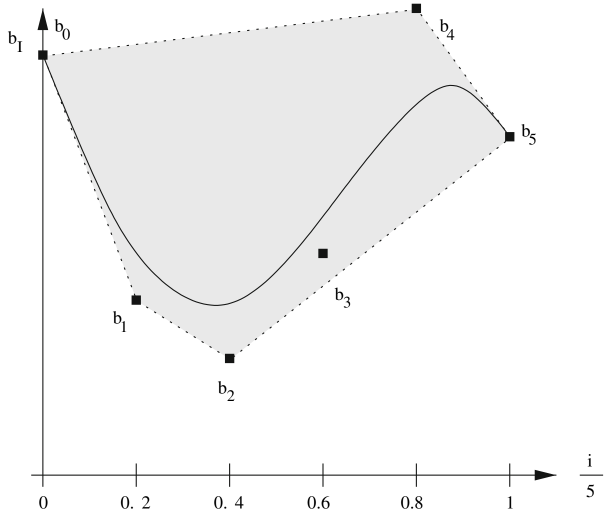

Fig. 1 The curve of a polynomial of fifth degree (bold) and the convex hull (shaded) of its control points (marked by squares)

It is also worth noting that the values attained by the polynomial at the vertices of X are identical to the corresponding vertex Bernstein coefficients. For example $b_0 = p(\underline{x})$ and $b_l = p(\overline{x})$. The *sharpness property* states that the lower (resp. upper) bound provided by the minimum (resp. maximum) Bernstein coefficient is sharp, i.e., there is no underestimation (resp. overestimation), if and only if this coefficient occurs at a vertex of X.

For a further introduction to the subject of Bernstein expansion, the reader is also referred to [3,6,17,24]. It is known [21] that the bounds provided by the Bernstein expansion are in general tighter than those given by interval arithmetic and many centered forms.

2.3 Complexity

The traditional approach (see, for example, [6,7,24]) assumes that all of the Bernstein coefficients are computed, and their minimum and maximum is determined. By use of an algorithm (cf. [6,24]) which is similar to de Casteljau's algorithm (see, for example, [17]), this computation can be made efficient, with time complexity $O(n\hat{l}^{n+1})$ and space complexity (equal to the number of Bernstein coefficients) $O((\hat{l}+1)^n)$, where $\hat{l} = \max_{i=1}^{n} l_i$. This exponential complexity is a drawback of the method, rendering it infeasible for polynomials with moderately many (typically, 10 or more) variables.

The main motivation of this work is therefore to exploit the range enclosing property of the Bernstein expansion without recourse to the exhaustive computation of all the Bernstein coefficients.

3 Bernstein coefficients of monomials

We begin by deriving some fundamental properties of the Bernstein coefficients of multivariate monomials. Let us consider the case of a polynomial comprising a single term

$$q(x) = a_k x^k, \quad x = (x_1, \ldots, x_n), \quad \text{for some } 0 \le k \le l. \tag{12}$$

Without loss of generality, we can take $a_k = 1$, since the Bernstein form is linear, i.e., the Bernstein coefficients for general a_k may be obtained by multiplying these Bernstein coefficients by a_k:

$$\begin{aligned} b_i &= \sum_{j=0}^{\min\{i,k\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} a_k \\ &= a_k \sum_{j=0}^{\min\{i,k\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} \end{aligned}$$

Theorem 1 *Let*

$$q(x) = x^k, \quad x = (x_1, \ldots, x_n), \quad \textit{for some } 0 \le k \le l, \tag{13}$$

where $l = (l_1, \ldots, l_n)$. The Bernstein coefficients of q (of degree l) over X (2) are given by

$$b_i = \prod_{m=1}^{n} b_{i_m}^{(m)}, \tag{14}$$

where $b_{i_m}^{(m)}$ is the i_mth Bernstein coefficient (of degree l_m) of the (univariate) monomial x^{k_m} over the interval $[\underline{x}_m, \overline{x}_m]$.

Proof The Bernstein coefficients of q (of degree l) over X are given by

$$\begin{aligned} b_i &= \sum_{j=0}^{\min\{i,k\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} \\ &= \sum_{j_1=0}^{\min\{i_1,k_1\}} \cdots \sum_{j_n=0}^{\min\{i_n,k_n\}} \frac{\binom{i_1}{j_1} \ldots \binom{i_n}{j_n}}{\binom{l_1}{j_1} \ldots \binom{l_n}{j_n}} (\overline{x}_1 - \underline{x}_1)^{j_1} \ldots (\overline{x}_n - \underline{x}_n)^{j_n} \binom{k_1}{j_1} \ldots \binom{k_n}{j_n} \underline{x}_1^{k_1-j_1} \\ &\quad \ldots \underline{x}_n^{k_n-j_n} \\ &= \sum_{j_1=0}^{\min\{i_1,k_1\}} \frac{\binom{i_1}{j_1}}{\binom{l_1}{j_1}} (\overline{x}_1 - \underline{x}_1)^{j_1} \binom{k_1}{j_1} \underline{x}_1^{k_1-j_1} \ldots \sum_{j_n=0}^{\min\{i_n,k_n\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} \\ &= \prod_{m=1}^{n} \sum_{j_m=0}^{\min\{i_m,k_m\}} \frac{\binom{i_m}{j_m}}{\binom{l_m}{j_m}} (\overline{x}_m - \underline{x}_m)^{j_m} \binom{k_m}{j_m} \underline{x}_m^{k_m-j_m} \\ &= \prod_{m=1}^{n} b_{i_m}^{(m)}. \end{aligned}$$

□

Example Let $n := 2$, $q(x) := x_1^3 x_2^2$, $l := (3, 2)$, and the box $X := [1, 2] \times [2, 4]$. The Bernstein coefficients are

$$\{b_i\} = \begin{pmatrix} 4 & 8 & 16 \\ 8 & 16 & 32 \\ 16 & 32 & 64 \\ 32 & 64 & 128 \end{pmatrix}. \tag{15}$$

Instead of calculating and storing all 12 Bernstein coefficients, we might instead represent them by

$$\begin{matrix} \mathbf{1} & \begin{pmatrix} 4 & 8 & 16 \end{pmatrix} \\ \begin{pmatrix} 1 \\ 2 \\ 4 \\ 8 \end{pmatrix} & \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \cdots \end{pmatrix} \end{matrix}, \tag{16}$$

i.e., the coefficient a_k, plus the Bernstein coefficients of x^3 over $[1, 2]$, plus the Bernstein coefficients of x^2 over $[2, 4]$. Any Bernstein coefficient of q over X can be computed as required as a simple product.

3.1 Bernstein coefficients of univariate monomials

In this section we consider the Bernstein coefficients $b_i^{(m)}$, $0 \leq i \leq l_m$, of degree l_m, of the univariate monomial x^{k_m} over the interval $[\underline{x}_m, \overline{x}_m]$, for some $1 \leq m \leq n$. For the remainder of the section, we omit the subscript m and the superscript (m), for simplicity.

The Bernstein coefficients are given by

$$b_i = \sum_{j=0}^{\min\{i,k\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} \tag{17}$$

We wish to simplify this computation as much as possible.

Case $k = l$

In this case, the formula can be simplified as follows:

$$\begin{aligned} b_i &= \sum_{j=0}^{\min\{i,k\}} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \binom{k}{j} \underline{x}^{k-j} \\ &= \sum_{j=0}^{i} \binom{i}{j} (\overline{x} - \underline{x})^j \underline{x}^{k-j} \\ &= \underline{x}^{k-i} \sum_{j=0}^{i} \binom{i}{j} (\overline{x} - \underline{x})^j \underline{x}^{i-j} \\ &= \underline{x}^{k-i} (\overline{x} - \underline{x} + \underline{x})^i \\ &= \underline{x}^{k-i} \overline{x}^i \end{aligned}$$

Case $k < l$

In this case, we can start with the Bernstein coefficients of degree k, and use degree elevation. It is known, e.g. [4], that the coefficients of higher degree can be expressed as a simple weighted sum of lower degree coefficients, as follows:

$$b_i^{[k+1]} = \frac{i b_{i-1}^{[k]} + (k + 1 - i) b_i^{[k]}}{k + 1}, \quad \text{with } b_{-1}^{[k]} = b_{k+1}^{[k]} = 0, \quad i = 0, \ldots, k + 1, \tag{18}$$

where the superscript in square brackets denotes the degree of the coefficient.

Repeated degree elevation yields the following expression (cf. [4]) for the coefficients of degree l:

$$b_i^{[l]} = \sum_{j=\max(0,i-m)}^{\min(k,i)} \frac{\binom{m}{i-j}\binom{k}{j}}{\binom{k+m}{i}} b_j^{[k]},$$
$$= \sum_{j=\max(0,i-m)}^{\min(k,i)} \frac{\binom{m}{i-j}\binom{k}{j}}{\binom{k+m}{i}} \underline{x}^{k-j}\overline{x}^{j},$$

where $m = l - k$. This computation can be simplified (cf. [23]) by observing that the part of the formula consisting of the two binomial coefficients where m appears can always be expressed as a product of k factors.

3.2 Monotonicity of the Bernstein coefficients of monomials

Theorem 2 *Let* $q(x) = a_k x^k$, $x = (x_1, \ldots, x_n)$, *for some* $0 \le k \le l$ *and let* $X = [\underline{x}_1, \overline{x}_1] \times \cdots \times [\underline{x}_n, \overline{x}_n]$ *be a box which is restricted to a single orthant of* $\mathbf{R}^n$. *Then the Bernstein coefficients* b_i *of* q *(of degree* l*) over* X *are monotone with respect to each variable* x_j, $j = 1, \ldots, n$.

Proof Given that the Bernstein coefficients of q over X can be expressed as a simple product of Bernstein coefficients of univariate monomials (cf. Theorem 1), it suffices to show monotonicity in the univariate case. □

Lemma 3 *Let* $q(x) = x^k$, *for some* $0 \le k \le l$, *be a univariate monomial and let* $X = [\underline{x}, \overline{x}]$ *be an interval where* $\underline{x}$ *and* $\overline{x}$ *have the same sign or vanish. Then the Bernstein coefficients* b_i *of* q *(of degree* l*) over* X *are monotone with respect to* i.

Proof We will assume that $0 \le \underline{x} < \overline{x}$; the negative case is entirely analogous. The result follows by induction on the degree of the Bernstein coefficients.

Case $l = k$*:* From the case $k = l$ above we have

$$b_i^{[k]} = \underline{x}^{k-i}\overline{x}^{i}, \tag{19}$$

from which it is clear that $b_i^{[k]} \le b_{i+1}^{[k]}$, $i = 0, \ldots, k-1$.

Case $l = k + m$, $m \ge 1$*:* Assume that the Bernstein coefficients of degree $k + j$, $b_i^{[k+j]}$, $i = 0, \ldots, k+j$, $0 \le j < m$, are increasing. The coefficients of degree $k + j + 1$, from Sect. 3.1, may be expressed as

$$b_i^{[k+j+1]} = \frac{i b_{i-1}^{[k+j]} + (k+j+1-i) b_i^{[k+j]}}{k+j+1}, \quad i = 0, \ldots, k+j+1, \tag{20}$$

with $b_{-1}^{[k+j]} = b_{k+j+1}^{[k+j]} = 0$. We observe that each $b_i^{[k+j+1]}$ is an affine combination of $b_{i-1}^{[k+j]}$ and $b_i^{[k+j]}$, from which

$$b_{i-1}^{[k+j]} \le b_i^{[k+j+1]} \le b_i^{[k+j]}, \quad i = 0, \ldots, k+j+1. \tag{21}$$

Therefore we have

$$b_i^{[k+j+1]} \leq b_{i+1}^{[k+j+1]}, \quad i = 0, \ldots, k + j, \tag{22}$$

□

and the result follows by induction.

With this property, for a single-orthant box, the minimum and maximum Bernstein coefficients must occur at a vertex of the array of Bernstein coefficients. This also implies that the bounds provided by these coefficients are sharp; see the sharpness property (Sect. 2.2). Finding the minimum and maximum Bernstein coefficients is therefore straightforward; it is not necessary to explicitly compute the whole set of Bernstein coefficients. Computing the component univariate Bernstein coefficients for a multivariate monomial has time complexity $O(n(\hat{l} + 1)^2)$. Of course, one can readily calculate the exact range of a multivariate monomial over a single-orthant box without recourse to any Bernstein coefficients: Given the exponent k and the orthant in question, one can determine whether the monomial (and its Bernstein coefficients) is increasing or decreasing with respect to each coordinate direction. This allows one to determine in advance at which vertex of the box the minimum or maximum is attained; one then merely needs to evaluate the monomial at these two vertices.

Without the single orthant assumption, monotonicity does not necessarily hold, and the problem of determining the minimum and maximum Bernstein coefficients is more complicated. For boxes which intersect two or more orthants of $\mathbf{R}^n$, the box can be bisected, and the Bernstein coefficients of each single-orthant sub-box can be computed separately. The complexity of computing the minimum or maximum Bernstein coefficient will often still be much less than $O((\hat{l} + 1)^n)$. It is already known [21] that a bisection performed around zero will yield an improvement of the bounds, unless they are already sharp.

4 The implicit Bernstein form

In this section, a new method of storing and representing the Bernstein coefficients of multivariate polynomials is proposed, which is referred to here as the “implicit Bernstein form”.

First we can observe that since the Bernstein form is linear, if a polynomial p consists of t terms, as follows

$$p(x) = \sum_{j=1}^{t} a_{i_j} x^{i_j}, \quad 0 \leq i_j \leq l, \quad x = (x_1, \ldots, x_n), \tag{23}$$

then each Bernstein coefficient is equal to the sum of the corresponding Bernstein coefficients of each term, as follows:

$$b_i = \sum_{j=1}^{t} b_i^{(j)}, \quad 0 \leq i \leq l, \tag{24}$$

where $b_i^{(j)}$ are the Bernstein coefficients of the jth term of p. (Hereafter, a superscript in brackets specifies a particular term of the polynomial. The use of this notation to indicate a particular coordinate direction, as in the previous section, is no longer required.)

Therefore one may implicitly store the Bernstein coefficients of each term, as in Sect. 3.1, and compute the Bernstein coefficients as a sum of t products, only as needed. Computing and storing the whole set of Bernstein coefficients, should, in general, not be required.

The implicit Bernstein form thus consists of computing and storing the n sets of univariate Bernstein coefficients (one set for each component univariate monomial) for each of t terms. Computing this form has time complexity $O(nt(\hat{l}+1)^2)$ and space complexity $O(nt(\hat{l}+1))$, as opposed to $O((\hat{l}+1)^n)$ for the explicit form. Computing a single Bernstein coefficient from the implicit form requires $(n+1)t-1$ arithmetic operations.

Example We extend the example presented in Sect. 3. Let $n := 2$, $p(x) := x_1^3x_2^2 - 30x_1x_2$, $l := (3, 2)$, and the box $X := [1, 2] \times [2, 4]$. The sum of the corresponding Bernstein coefficients of each term gives the Bernstein coefficients of p:

$$\{b_i\} = \begin{pmatrix} 4 & 8 & 16 \\ 8 & 16 & 32 \\ 16 & 32 & 64 \\ 32 & 64 & 128 \end{pmatrix} + \begin{pmatrix} -60 & -90 & -120 \\ -80 & -120 & -160 \\ -100 & -150 & -200 \\ -120 & -180 & -240 \end{pmatrix}$$

$$= \begin{pmatrix} -56 & -82 & -104 \\ -72 & -104 & -128 \\ -84 & -118 & -136 \\ -88 & -116 & -112 \end{pmatrix}. \tag{25}$$

The implicit form of these coefficients can be represented as

$$\begin{matrix} \mathbf{1} \\ \begin{pmatrix} 1 \\ 2 \\ 4 \\ 8 \end{pmatrix} \end{matrix} \begin{matrix} \begin{pmatrix} 4 & 8 & 16 \end{pmatrix} \\ \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \cdots \end{pmatrix} \end{matrix} + \begin{matrix} \mathbf{-30} \\ \begin{pmatrix} 1 \\ \frac{4}{3} \\ \frac{5}{3} \\ 2 \end{pmatrix} \end{matrix} \begin{matrix} \begin{pmatrix} 2 & 3 & 4 \end{pmatrix} \\ \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \cdots \end{pmatrix} \end{matrix}. \tag{26}$$

4.1 Determination of the Bernstein enclosure for polynomials

This section considers the determination of the minimum Bernstein coefficient; the determination of the maximum Bernstein coefficient is analogous. The Bernstein enclosure is given as the interval spanned by the minimum and maximum Bernstein coefficients.

If the box X (2) spans multiple orthants of $\mathbf{R}^n$, then it should be subdivided around zero into two or more subboxes, and the Bernstein enclosure for each subbox computed separately. The remainder of this section thus assumes that X is restricted to a single orthant. It should be noted that, for branch-and-bound methods for constrained global optimization, the vast majority of the computational effort is typically occupied with small subboxes which lie within a single orthant.

Clearly, the determination of the minimum Berstein coefficient is not so simple as for a polynomial comprising a single term; the minimum is not guaranteed to occur at a vertex of the array, although that may often be the case. For polynomials in general, it is doubtful that a *universal* method more efficient than simply computing all of the Bernstein coefficients exists. However, when the number of terms of the polynomial is much less than the number of Bernstein coefficients (which is typically the case for many real-world problems), it is often possible in practice to dramatically reduce the number of coefficients which have to be computed, by reducing the number of Bernstein coefficients which have to be searched.

The minimum Bernstein coefficient is referenced by a multiindex, which we label $i_{\min}$, $0 \leq i_{\min} \leq l$. We wish to determine the value of the multiindex of the minimum Bernstein coefficient in each direction. In order to reduce the search space (among the $(\hat{l}+1)^n$ Bernstein coefficients) we can exploit Theorem 2 and employ the following tests:

- *Uniqueness*: If a variable x_j appears in only one term of the polynomial, then the Bernstein coefficients of the term in which it appears determines $i_{\min_j}$, which is thus either 0 or l_j.
- *Monotonicity*: If the Bernstein coefficients of all terms containing x_j are likewise monotone with respect to x_j, then $i_{\min_j} = 0$ (if all are increasing) or l_j (if decreasing).
- *Dominance*: Otherwise, all the terms containing x_j can be split into two sets, depending on whether they are increasing or decreasing with respect to x_j. If the width of the Bernstein enclosure of one set is less than the minimum difference between Bernstein coefficients among the terms of the other set, then the first set can make no contribution to the determination of $i_{\min_j}$, and the monotonicity clause applies.

Theorem 4 (Location of minimum Bernstein coefficient under uniqueness/monotonicity) *For a polynomial p given as per (23), the multiindex of the minimum Bernstein coefficient of p over a single-orthant box X,* $i_{\min}$*, must satisfy*

$$\min_{j=1}^{t}\{i_{\min}^{(j)}\} \leq i_{\min} \leq \max_{j=1}^{t}\{i_{\min}^{(j)}\}. \tag{27}$$

Proof Suppose there is some k, $k \in \{1, \dots, n\}$, for which

$$\min_{j=1}^{t}\{i_{\min_k}^{(j)}\} > i_{\min k}. \tag{28}$$

The case of $i_{\min k}$ exceeding the maximum of the $i_{\min_k}^{(j)}$ is entirely analogous. Assume $0 \leq \underline{x}_k < \overline{x}_k$; the negative case is analogous. Then there is no m, $m \in \{1, \dots, t\}$, for which $i_{\min_k}^{(m)} = 0$ and therefore the $b_i^{(m)}$ are decreasing with respect to i_k for all $m \in \{1, \dots, t\}$. Therefore the $b_i = \sum_{j=1}^{t} b_i^{(j)}$ are decreasing with respect to i_k and so $i_{\min k} = l_k$, which is a contradiction of the initial supposition, and so the result follows. □

Theorem 5 (Location of minimum Bernstein coefficient under dominance) *Given a polynomial p as per (23) and a single-orthant box X, for some* $j \in \{1, \dots, n\}$*, let* p^{inc} *be the polynomial comprising the sum of the terms of p which are increasing with respect to* x_j*, and let* p^{dec} *be the polynomial comprising the sum of the terms of p which are decreasing with respect to* x_j*, with Bernstein coefficients* b_i^{inc} *and* b_i^{dec}*, respectively,* $0 \leq i \leq l$*. If*

$$\forall i = 0, \dots, l, i_j \neq l_j : b^{inc}_{i_1,\dots,i_j+1,\dots,i_l} - b^{inc}_{i_1,\dots,i_j,\dots,i_l} > b^{dec}_{i_1,\dots,0,\dots,i_l} - b^{dec}_{i_1,\dots,l_j,\dots,i_l} \tag{29}$$

then $i_{\min j} = 0$*. If*

$$\forall i = 0, \dots, l, i_j \neq l_j : b^{dec}_{i_1,\dots,i_j,\dots,i_l} - b^{dec}_{i_1,\dots,i_j+1,\dots,i_l} > b^{inc}_{i_1,\dots,l_j,\dots,i_l} - b^{inc}_{i_1,\dots,0,\dots,i_l} \tag{30}$$

then $i_{\min j} = l_j$*.*

Proof The proof is presented for the first result (29); the proof of the second (30) is entirely analogous. For all $i = 0, \dots, l$, $i_j \neq l_j$ we have

$$\begin{aligned} b_{i_1,\dots,i_j+1,\dots,i_l} &= b^{inc}_{i_1,\dots,i_j+1,\dots,i_l} + b^{dec}_{i_1,\dots,i_j+1,\dots,i_l} \\ &\geq b^{inc}_{i_1,\dots,i_j+1,\dots,i_l} + b^{dec}_{i_1,\dots,l_j,\dots,i_l} \\ &> b^{inc}_{i_1,\dots,i_j,\dots,i_l} + b^{dec}_{i_1,\dots,0,\dots,i_l} \\ &\geq b^{inc}_{i_1,\dots,i_j,\dots,i_l} + b^{dec}_{i_1,\dots,i_j,\dots,i_l} \\ &= b_{i_1,\dots,i_j,\dots,i_l}, \end{aligned}$$

i.e., that the b_i are increasing with respect to x_j, and the result follows. □

4.2 Example

Consider the polynomial

$$p(x) = 3x_1x_2^5 + 2x_1^4x_2 - 8x_1^2x_3^6x_4^2 - x_1x_4^8 + 3x_2^3x_5 - 10x_4^5x_5^5x_6^5 + 0.01x_5^2x_6^2 + 4x_5^3x_7^4 \tag{31}$$

over the box

$$X = [1, 2]^7. \tag{32}$$

The degree, l, is $(4, 5, 6, 8, 5, 5, 4)$ and the number of Bernstein coefficients is thus 3, 40, 200 $(5 \times 6 \times 7 \times 9 \times 6 \times 6 \times 5)$. We can make the following observations:

- *Uniqueness*: x_3 appears only in term 3, which is decreasing wrt it. Therefore $i_{\min 3} = 6$.
- *Uniqueness*: x_7 appears only in term 8, which is increasing wrt it. Therefore $i_{\min 7} = 0$.
- *Monotonicity*: x_2 appears in terms 1 and 2, both of which are increasing wrt it. Therefore $i_{\min 2} = 0$.
- *Monotonicity*: x_4 appears in terms 3, 4, and 6, all of which are decreasing wrt it. Therefore $i_{\min 4} = 8$.
- *Dominance*: x_6 appears in terms 6 and 7, one of which is decreasing and one of which is increasing wrt it. However, term 6 dominates term 7 to such an extent that term 7 plays no role in determining $i_{\min 6}$. Therefore $i_{\min 6} = 5$, since term 6 is decreasing wrt x_6.

Variable x_1 appears in terms 1, 2, 3, and 4, and x_5 appears in terms 6, 7, and 8. A determination of $i_{\min 1}$ and $i_{\min 5}$ thus seems to be non-trivial.

So far, we have determined that $i_{\min} = (?, 0, 6, 8, ?, 5, 0)$. The dimensionality of the search space has thus been reduced from 7 to 2. The number of Bernstein coefficients to compute is consequently reduced from 3, 40, 200 to 30 (5×6), plus those needed for the implicit Bernstein form, 78 $(8 + 7 + 13 + 11 + 6 + 18 + 6 + 9)$, 108 total.

4.3 Algorithm for the efficient calculation of the Bernstein enclosure of polynomials

An algorithm for the determination of the minimum Bernstein coefficient is given here; the procedure for the determination of the maximum Bernstein coefficient is analogous.

We are given a polynomial p consisting of t terms, whose degree is $l = (l_1, \ldots, l_n)$, and a box X, as before. We seek to find a multiindex $i_{\min}$ which references the minimum Bernstein coefficient $b_{i_{\min}}$.

1. If X is not restricted to a single orthant of $\mathbf{R}^n$, i.e., there is one or more component intervals of X, $[\underline{x}_m, \overline{x}_m]$, $1 \leq m \leq n$, which contain both positive and negative numbers, then subdivide X around 0, perform steps 2–5 below for each subbox, and take the minimum of the minimum Bernstein coefficients for each subbox.
2. Compute the implicit Bernstein form of p over X, consisting of the Bernstein coefficients of the component univariate monomials of each term.
3. Initialise the search space for $i_{\min}$, S, as the set of all possible multiindices $\{(0, \ldots, 0), \ldots, (l_1, \ldots, l_n)\}$.
4. For each variable x_j, $j = 1, \ldots, n$:

   (a) *Uniqueness test*: Count the number of terms for which the Bernstein coefficients are non-constant with respect to x_j. If the number is one, then restrict S so that the jth index corresponds to the minimum Bernstein coefficient of the non-constant term; it is either 0 or l_j.

(b) *If the uniqueness test fails, then proceed with the monotonicity test*: sort the terms into those which are increasing, decreasing, and constant with respect to x_j. If the set of increasing terms is empty, then restrict S so that the jth index is l_j. If the set of decreasing terms is empty, then restrict S so that the jth index is 0.

(c) *If the uniqueness and monotonicity tests fail, then proceed with the dominance test*: using the two non-empty sets of increasing and decreasing terms from the monotonicity test, compute the width of the Bernstein enclosures of each set, and the minimum absolute difference between Bernstein coefficients of each set. If the minimum absolute difference of the decreasing terms is greater than the width of the increasing terms, then restrict S so that the jth index is l_j. If the minimum absolute difference of the increasing terms is greater than the width of the decreasing terms, then restrict S so that the jth index is 0.

5. Explicitly compute the Bernstein coefficients corresponding to the remaining multiindices in S, from the implicit form, and determine their minimum.

5 Numerical results

The implicit Bernstein form and the algorithm presented in the previous section is tested here, and compared to the usual Bernstein form. For each test problem, consisting of a polynomial and a starting box, the Bernstein enclosure is computed, using each method. The box is then bisected in each variable direction in turn, providing a crude simulation of a branch-and-bound environment. One of the two resulting subboxes (chosen randomly) is retained and the other is discarded, although in practice, various criteria may be used for subbox selection. After each bisection, the Bernstein enclosure is recomputed over the new box. This process is iterated 100 times, so that the final subbox is very small.

With normal floating-point arithmetic, inaccuracies may be introduced into the calculation of the Bernstein coefficients and the corresponding bounds, due to rounding errors. As a result, the bounds may not be guaranteed to enclose the range of the polynomial over the box. Therefore interval arithmetic has been used; all Bernstein coefficients are computed and stored as intervals. The bounds provided are thus guaranteed; see also [2,13,16].

The first test problem consists of the polynomial

$$p(x) = 3x_1^2x_2^3x_3^4 + 1x_1^3x_2x_3^4 - 5x_1x_2x_4^5 + 1x_3x_4x_5^3 \tag{33}$$

over the box

$$X = [1, 2] \times [2, 3] \times [4, 6] \times [-5, -2] \times [2, 10]. \tag{34}$$

The second test problem is the example given in Sect. 4.2. The remaining test problems are polynomial objective functions drawn from GLOBALLib [11]. Where unspecified, a suitable single-orthant starting box of unit width was chosen.

The results are given in Table 1; n is the number of variables, t the number of terms, and l the degree, in each case. The number of Bernstein coefficients refers to the number that have to be computed explicitly; for test2, for example (the polynomial and box given in Sect. 4.2), 30 Bernstein coefficients are required to determine the minimum, and 30 to determine the maximum. The timings in the table are the mean computation times for a single iteration. These numbers should be multiplied by 100 to get the computation times for all iterations. The results were produced with C++ on a 2.4 GHz PC running Linux; the BeBP software package [20] and the interval library filib++ [15] were employed. The algorithm may be executed with floating-point arithmetic in place of interval arithmetic; this speeds

Table 1 Number of Bernstein coefficients calculated and computation time for some example polynomials

Name	n	t	l	Bernstein form			Implicit Bernstein form		
				Iterations	No. of BCs	Time (s)	Iterations	No. of BCs	Time (s)
test1	5	4	(3, 3, 4, 5, 3)	1–100	1,920	0.01	1–3	60	
							4–5	12	0.0001
							6–100	2	
test2	7	8	(4, 5, 6, 8, 5, 5, 4)	1–100	3,40,200	6.05	1–9	60	
							10–100	12	0.0004
mhw4d	5	17	(2, 3, 4, 4, 4)	1–100	1,500	0.04	1–3	1,000	
							4–100	200	0.0068
meanvar	7	49	(2, 2, 2, 2, 2, 2, 2)	1–100	2,187	0.24	1–100	2	0.0008
ex2-1-5	10	16	(2, 2, 2, 2, 2, 2, 2, 1, 1, 1)	1–100	17,496	0.83	1–100	2	0.0003
harker	20	40	(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 2)	1–100	1.96×10^{11}	$> 10^5$	1–100	2	0.0019

up the process (by approximately one order of magnitude), but the resultant bounds are no longer guaranteed.

It is clear that the use of the implicit Bernstein form can dramatically reduce the number of Bernstein coefficients that need to be computed explicitly, thereby speeding up the computation of the Bernstein enclosure by up to several orders of magnitude.

It should be noted that almost all of the polynomials in [11] are sparse and of low degree, with few or no terms involving more than a single variable. This seems to be typical of the types of polynomials encountered in global optimization problems. In such cases, the uniqueness, monotonicity, and dominance tests are much more likely to succeed, compared to a polynomial where each variable appears in many terms.

6 Future work

A follow-up paper will report on algorithms for the construction of affine bound functions for polynomials, cf. [9,10], based upon the implicit Bernstein form presented here. These bound functions, and the constant bound functions here, will be tested as a black-box component of the COCONUT software environment [19], a general-purpose package for the solution of global optimization and continuous constraint satisfaction problems.

In principle, this approach may be extended to the construction of tight constant bound functions for arbitrary sufficiently differentiable functions, by using Taylor expansion. A high-degree Taylor polynomial can be calculated, for which the implicit Bernstein form and the resulting bounds can be computed, as before. The remainder of the Taylor expansion can be enclosed in an interval, by using established methods from interval analysis. Subtracting this interval from the lower bound of the Taylor polynomial provides the lower bound for the given function. However, such Taylor polynomials are in general dense, for which the computational advantage of the implicit Berstein form is negated.

Acknowledgements The author would like to thank Jürgen Garloff for his careful reading of the manuscript and his constructive comments. Support from the German Research Council (DFG) is gratefully acknowledged.

References

1. Adjiman, C.S., Floudas, C.A.: Rigorous convex underestimators for general twice-differentiable problems. J. Glob. Optim. **9**(1), 23–40 (1996)
2. Borradaile, G., Van Hentenryck, P.: Safe and tight linear estimators for global optimization. Math. Program. **102**(3), 495–518 (2005)
3. Cargo, G.T., Shisha, O.: The Bernstein form of a polynomial. J. Res. Nat. Bur. Stand. **70B**, 79–81 (1966)
4. Farouki, R.T., Rajan, V.T.: Algorithms for polynomials in Bernstein form. Comput. Aided Geom. Design **5**, 1–26 (1988)
5. Floudas, C.A.: Deterministic Global Optimization: Theory, Methods, and Applications. Series Nonconvex Optimization and its Applications. Vol. 37. Kluwer Acad. Publ., Dordrecht (2000)
6. Garloff, J.: Convergent bounds for the range of multivariate polynomials. In: Nickel K. (eds.) Interval Mathematics 1985. Lecture Notes in Computer Science, vol. 212, pp. 37–56, Springer, Berlin (1986)
7. Garloff, J., Smith, A.P. : An improved method for the computation of affine lower bound functions for polynomials. In: Floudas, C.A., Pardalos, P.M. (eds.) Frontiers in Global Optimization Series Nonconvex Optimization with its Applications, pp. 135–144. Kluwer Acad. Publ., Dordrecht (2004)
8. Garloff, J., Smith, A.P.: A comparison of methods for the computation of affine lower bound functions for polynomials. In: Jermann C., Neumaier A., Sam D. (eds.) Global Optimization and Constraint Satisfaction. Lecture Notes in Computer Science, No. 3478, pp. 71–85. Springer-Verlag, Berlin (2005)
9. Garloff, J., Jansson, C., Smith, A.P.: Lower bound functions for polynomials. J. Computat. Appl. Math. **157**, 207–225 (2003)
10. Garloff, J., Jansson, C., Smith, A.P.: Inclusion isotonicity of convex-concave extensions for polynomials based on Bernstein expansion. Computing **70**, 111–119 (2003)
11. GLOBAL Library, available under http://www.gamsworld.org/global/globallib.htm
12. Hansen, E.R.: Global Optimization using Interval Analysis, 2nd edn. Marcel Dekker, Inc., New York (2003)
13. Hongthong, S., Kearfott, R.B.: Rigorous linear overestimators and underestimators. Tech. rep., University of Louisiana at Lafayette (2004)
14. Kearfott, R.B.: Rigorous Global Search: Continuous Problems. Series Nonconvex Optimization and its Applications. Vol. 13, Kluwer Acad. Publ., Dordrecht (1996)
15. Lerch, M., Tischler, G., Wolff von Gudenberg J.: filib++ - Interval library specification and reference manual. Technical Report 279, University of Wuerzburg (2001)
16. Neumaier, A., Shcherbina, O.: Safe bounds in linear and mixed-integer programming. Math. Program. A **99**, 283–296 (2004)
17. Prautzsch, H., Boehm, W., Paluszny, M.: Bézier and B-Spline Techniques. Springer, Berlin (2002)
18. Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization. Ellis Horwood Ltd., Chichester (1988)
19. Schichl, H.: Mathematical Modeling and Global Optimization. Habilitation Thesis, University of Vienna, (2003)
20. Smith, A.P.: BeBP: a software package for the rigorous computation of affine bound functions for polynomials. ACM Trans. Math. Software (2007) (submitted)
21. Stahl V.: Interval methods for bounding the range of polynomials and solving systems of nonlinear equations. Ph.D. thesis, University of Linz (2005)
22. Tawarmalani, M., Sahinidis, N.V.: Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications.Series Nonconvex Optimization and its Applications. Vol. 65, Kluwer Acad. bl., Dordrecht (2002)
23. Trump, W., Prautzsch, H.: Arbitrarily high degree elevation of Bézier representations. Computer Aided Geometric Design **13**(5), 387–398 (1996)
24. Zettler, M., Garloff, J.: Robustness analysis of polynomials with polynomial parameter dependency using Bernstein expansion. IEEE Trans. Automat. Contr. **43**, 425–431 (1998)