Polytope-Based Computation of Polynomial Ranges

Christoph Fünfzig[a], Dominique Michelucci[a], Sebti Foufou[a,b]

[a] *LE2I (UMR CNRS 5158), Univ. Bourgogne, 9 av Alain Savary, 21078 Dijon, France*
[b] *CSE, Univ. Qatar, P.O.B. 2713, Doha, Qatar*

Abstract

Polynomial ranges are commonly used for numerically solving polynomial systems with interval Newton solvers. Often ranges are computed using the convex hull property of the tensorial Bernstein basis, which is exponential size in the number n of variables. In this paper, we consider methods to compute tight bounds for polynomials in n variables by solving two linear programming problems over a polytope. We formulate a polytope defined as the convex hull of the coefficients with respect to the tensorial Bernstein basis, and we formulate several polytopes based on the Bernstein polynomials of the domain. These Bernstein polytopes can be defined by a polynomial number of halfspaces. We give the number of vertices, the number of hyperfaces, and the volume of each polytope for $n = 1, 2, 3, 4$, and we compare the computed range widths for random n-variate polynomials for $n \leq 10$. The Bernstein polytope of polynomial size gives only marginally worse range bounds compared to the range bounds obtained with the tensorial Bernstein basis of exponential size.

Keywords: polynomial ranges, Bernstein polynomials, multivariate

Email addresses: `c.fuenfzig@gmx.de` (Christoph Fünfzig), `dmichel@u-bourgogne.fr` (Dominique Michelucci), `sfoufou@u-bourgogne.fr`, `sfoufou@qu.edu.qa` (Sebti Foufou)

polynomials, polytopes

1. Introduction

Ranges of polynomials and properties of tensorial Bernstein bases are typically used for numerically solving polynomial systems with interval Newton solvers. These solvers require tight ranges of the Newton map $N(\mathbf{x}) = \mathbf{x} - P(\mathbf{x})P'(m(D))^{-1}$, containing multivariate polynomials over an n-dimensional domain D with center $m(D)$. Popular methods in CAD-CAM for intersecting Bézier or piecewise algebraic curves and surfaces are examples for them.

We consider the problem of computing a tight range approximation, *i.e.,* a constant lower bound and a constant upper bound, of a given multivariate polynomial $p(\mathbf{x})$, $\mathbf{x} \in D$. Further bounding problems, *i.e.,* an affine lower bound function, have been considered in Garloff et al. (2003) and how to compute them by linear programming.

In Moore et al. (2009), a branch-and-bound algorithm (*Moore-Skelboe algorithm*) for an ϵ-exact range is presented. It can use any approximation $[p_{\min}, p_{\max}]$ for the range of the polynomial on a domain D. The time complexity of this algorithm comes from the complexity of subdivisions in $\mathbb{R}^n$. Uniform subdivision of D_i into N subintervals in each of the n variables requires the evaluation of N^n range approximations, where N is chosen based on the given ϵ. In contrast the branch-and-bound algorithm, explores a node front in the tree of all the N^n leaf nodes. If the node front consists only of a constant number of leaf nodes then the method requires $O(n \log_2 N)$ range approximations (root-to-leaf-paths have length $\log_2 N^n$). Of course, problems exist where the branch-and-bound method needs to explore much larger parts of the tree.

Several authors proposed to use properties of the tensorial Bernstein basis for computing tight range approximations of polynomials $p(\mathbf{x})$ with $\mathbf{x} \in [0,1]^n$. The general case $\mathbf{x}^* \in D$ can be transferred to the case with $\mathbf{x} \in [0,1]^n$ by an affine map of the variables $x_i^* = \underline{D_i} + x_i(\overline{D_i} - \underline{D_i})$. Using the affine map of variables, has the disadvantage that it includes further monomials into the polynomial. The polynomial is expressed in the tensorial Bernstein basis, and due to the convex hull property, the smallest coefficient is a lower bound, and the largest is an upper bound for the range. In Smith (2009), a recursive method for constructing the coefficients of the tensorial Bernstein basis by exploiting the fact that the univariate monomials x_i and x_i^d used in the construction are monotonous on any interval $D_i \subseteq \mathbb{R}^{\geq 0}$ or $D_i \subseteq \mathbb{R}^{\leq 0}$. For the case of an arbitrary domain D with point $\mathbf{0}$ in the interior, the domain D must be subdivided to make $\mathbf{0}$ a vertex and compute range approximations then for the 2^n sub-domains with significant speed-up using the method from Smith (2009). Similar optimizations concerning monotonicity of terms have been considered by Araya et al. (2009).

In this article, we analyze enclosures of the monomials of the canonical basis on an arbitrary domain D into a polytope. A polytope is a bounded, convex set containing the feasible points for a linear programming problem. It can be defined by the list of its halfspaces, or by the list of its vertices, see Avis (2000). These polytopes based on the Bernstein polynomials serve for computing the lower bound or the upper bound of a multivariate polynomial by linear programming. As far as we know, it is the first article, which discusses polytopes related to Bernstein polynomials. Polytopes related to

Chebyshev polynomials have been introduced in Beaumont (1999).

1.1. Notation

Concerning notation, we use multi-indices $\mathbf{a} = (a_1, \ldots, a_n) \in \mathbb{Z}^n$ and componentwise comparisons $\mathbf{a} \leq \mathbf{d}$ iff $a_i \leq d_i$ for all i. We denote by $|\mathbf{a}|$ the sum $\sum_i a_i$ of indices. For $\mathbf{x} = (x_1, \ldots, x_n)$ and $D = \otimes_{i=1}^n D_i$, $D_i \in \mathbb{I}R$, $\mathbf{x}^\mathbf{a} = x_1^{a_1} \cdots x_n^{a_n}$ and $D^\mathbf{a} = D_1^{a_1} \cdots D_n^{a_n}$. A similar notation also applies to Bernstein polynomials introduced in Section 3.2, $B_\mathbf{a}^{(\mathbf{d})}(\mathbf{x}) = B_{a_1}^{(d_1)}(x_1) \cdots B_{a_n}^{(d_n)}(x_n)$. Furthermore, $\mathbf{a}/\mathbf{d} = (a_1/d_1, \ldots, a_n/d_n)$ and $\mathrm{supp}_\mathbf{d}(\mathbf{a}) = \{i : 0 < a_i, a_i < d_i\}$. We use $\underline{D}$ for the lower bound of an interval D and $\overline{D}$ for the upper bound. We denote by $w(D)$ the width $\max_{1 \leq i \leq n}(\overline{D_i} - \underline{D_i})$, and by $m(D)$ the center $(1/2(\underline{D_i} + \overline{D_i}))_{i=1,\ldots,n}$ of the interval D.

1.2. Degree Reduction

We consider polytope-based bounds for non-linear functions of degree 2 and degree 3 in this article. Large classes of geometric constraint systems are quadratic. At the expense of increasing the dimension of the problem, it is possible to reduce the degree. We call reduction to total degree-2 *binarization*. Binarization can be defined inductively: The binarization of a term $x_1 \ldots x_l$ is $x_{1\ldots l} = x_{12} x_{3\ldots l}$ and of a term $x_1 x_2$ is $x_{12} = x_1 x_2$. This reduction generates at most $n-1$ additional variables and $4(n-1)$ additional equalities per term.

We call reduction to total degree-3 *ternarization*. Both problems of binarization and ternarization with a minimum number of variables are instances of the *Exact Cover* problem in Garey and Johnson (1979). For ternarization, the covering sets are all three-element subsets $\binom{n}{3}$ of the variables

$\{x_1, \ldots, x_n\}$. A greedy approach for this problem would be to count the occurrences of all these $\binom{n}{3}$ subsets in the monomials $\mathbf{x}^{\mathbf{a}^{1}} \ldots, \mathbf{x}^{\mathbf{a}^{l}}$ to be covered. Always the subset $\{x_{i_1}, x_{i_2}, x_{i_3}\}$ of maximum count is defined as a new variable $x_{i_1,i_2,i_3} = x_{i_1} x_{i_2} x_{i_3}$ and replaced in the monomials. After each replacement $n' := n+1$, $\binom{n'}{2}$ new 3-element subsets are added, but the number of non-zero counts are bounded by the number of initial monomials. Finally, remaining two-element subsets of the variables $\{x_1, \ldots, x_n\}$ are covered.

1.3. Paper Organization

The paper is organized as follows. Section 2 introduces the reader to the theoretical complexity of range computation. In Section 3, we review the previous work on tight range approximation (Section 3.1), and Section 3.2 presents the tensorial Bernstein basis with its convex hull property. Section 4 uses the convex hull of the control points of total degree 2 in the canonical basis. All these polytopes have an exponential complexity in terms of n: both their number of vertices and their number of hyperfaces is at least exponential in the number n of variables. Section 5 presents all possible polytopes based on the Bernstein polynomials and examines several examples with a polynomial number of halfspaces. Using linear programming in polynomial time, these polytopes permit to compute tight ranges of polynomials, which are better than the ones provided by interval analysis. We show polytope projections in Section 6, which are significantly different, and we compare the ranges obtained by linear programming in Section 7.

2. Complexity of Exact Range Computation

According to a theorem by Gaganov (1985), computing an ϵ-accurate range of a polynomial is NP-hard. A consequence is that there is no geometric basis for quadratic polynomials providing tight ranges: By definition, the polynomials of such a geometric basis are non-negative for $\mathbf{x} \in [0,1]^n$, and their sum equals 1 for any $\mathbf{x}$. We sketch how to derive a geometric basis from a simplex enclosing the patch

$$Q_n := (x_1, \ldots, x_n, x_1^2, \ldots, x_n^2, x_1 x_2, \ldots, x_{n-1} x_n), \quad \mathbf{x} \in [0,1]^n. \tag{1}$$

Assume the smallest simplex enclosing Q_n is known. Then the minimum and maximum coefficient of a quadratic polynomial in the basis derived from this smallest simplex gives a range of the polynomial. Moreover, this method is polynomial time since there are only $O(n^2)$ coefficients to compute. The obtained range can not be tight, because it would contradict Gaganov's theorem, which establishes the NP-hardness of computing an ϵ-accurate range of a polynomial.

In terms of computational complexity, it has been shown that linear programming is weakly polynomial time (Schrijver (1986)). *Weakly polynomial time* refers to problems and algorithms which perform a polynomial number of operations in the unit-cost model of computation (similar to floating point computation with fixed length). Kamarkar was the first to propose a weakly polynomial time algorithm for linear programming (*ellipsoid method* Schrijver (1986)). In practice, interior-point methods and simplex methods are polynomial time. In special cases shown by Klee-Minty, the number of iterations can be exponential for both kinds of methods, and we refer the reader to

Deza et al. (2008). Under the arithmetic model of computation over the integers, a strongly polynomial-time algorithm is currently not known. *Strongly polynomial time* refers to algorithms which perform a number of operations bounded by a polynomial in the number of integers in the input, and the used space is bounded by a polynomial in the length of integers in the input.

3. Algorithms for Range Computation

3.1. Interval Extensions

The range interval $p(D)$ of a function $p : \mathbb{R}^n \to \mathbb{R}$ on a domain $D \in \mathbb{I}R^n$ is the minimal interval $p(D) \in \mathbb{I}R$, which contains $p(\mathbf{x})$ for all $\mathbf{x} \in D$. An interval function $[p] : \mathbb{I}R^n \to \mathbb{I}R$ is called an *inclusion function* or *interval extension* of a function $p : \mathbb{R}^n \to \mathbb{R}$ if $p(D) \subseteq [p](D)$.

The simplest one is the *natural inclusion function* $[p]_{\text{natural}}$, which can be defined inductively by the interval evaluation of an expression defining $p(\mathbf{x})$. We assume that $p(\mathbf{x})$ is given in the canonical basis

$$[p]_{\text{natural}}(D) := \sum_{\mathbf{a} \leq \mathbf{d}} c_{\mathbf{a}} \, D^{\mathbf{a}}$$

The *centered inclusion function* $[p]_{\text{mean}}$ is given by

$$[p]_{\text{mean}}(D) := p(\mathbf{m}) + [p'](D) \cdot (D - m(D)) \text{ with the domain center } m(D) \in D$$

This function uses an inclusion function $[p']$ for the gradient function p'. Note that for a quadratic polynomial p, the gradient function p' is linear, and $[p'](D)$ encloses the gradient range tightly. In Moore et al. (2009), it is shown that the natural inclusion function is $w([p]_{\text{natural}}(D)) \in O(w(D))$ for any Lipschitz-continuous function p, *i.e.,* it approximates the range to the first order. The centered inclusion function is $w([p]_{\text{mean}}(D)) \in O(w^2(D))$, *i.e.,* it approximates the range to the second order.

3.2. Bernstein Basis Representation

The $d+1$ Bernstein polynomials $B_i^{(d)}$, $i = 0, \ldots, d$, of degree d are a basis for univariate polynomials of degree d

$$B_i^{(d)}(x) = \binom{d}{i} x^i (1-x)^{d-i}.$$

In the following, we denote the Bernstein polynomials of degree 2 by $B_0^{(2)}(x) \in [0,1]$, $B_1^{(2)}(x) \in [0,1/2]$, $B_2^{(2)}(x) \in [0,1]$ for $x \in [0,1]$. If the degree superscript is omitted, we assume degree 2 in this article. Note that the Bernstein polynomials can also be defined on an arbitrary interval D different from $[0,1]$.

Univariate Bernstein Representation. The conversion with the canonical basis $(x^0, x^1, \ldots, x^d)$ is a linear map. Classical identities are according to Farin (2002)

$$x^k = \binom{d}{k}^{-1} \sum_{i=k}^{d} \binom{i}{k} B_i^{(d)}(x)$$

$$x = \frac{1}{d} \sum_{i=1}^{d} i \, B_i^{(d)}(x)$$

$$x^0 = 1 = \sum_{i=0}^{d} B_i^{(d)}(x) \text{ } \textit{i.e.,} \text{ their sum equals 1.}$$

However, Farouki (1991) shows that the conversion matrix with the canonical basis has large condition number for high degrees. Therefore the conversion should be used with caution. Note that the explicit conversion is not necessary in this article.

Definition 1. *For each univariate polynomial* $p(x)$ *of degree* d*, there exists* **an univariate Bernstein representation***, i.e., coefficients* $p_i \in \mathbb{R}$ *so that* $p(x) = \sum_i p_i B_i^{(d)}(x)$ *is a linear combination of the Bernstein basis functions.*

Properties of Bernstein Polynomials. The Bernstein polynomials form a partition of unity, and every $B_i^{(d)}(x)$ is non-negative for $x \in [0,1]$. These properties imply that for $x \in [0,1]$, $p(x) = \sum_i p_i\, B_i^{(d)}(x)$ is a convex combination of the coefficients p_i. For a polynomial $p(x)$ over $x \in [0,1]$, its value $p(x)$ lies in the convex hull of the control points $p_i \in \mathbb{R}$, which is just the interval $[\min p_i, \max p_i]$ and this enclosure is tight. The stability of operations with polynomials in the Bernstein representation is analyzed in Farouki and Rajan (1987).

Since $x = 0 \cdot B_0^{(d)}(x) + 1/d \cdot B_1^{(d)}(x) + 2/d \cdot B_2^{(d)}(x) + \ldots d/d \cdot B_d^{(d)}(x)$, the function graph $(x, y = p(x) = \sum_i p_i\, B_i^{(d)}(x))$ for $x \in [0,1]$, lies in the convex hull of its control points $(i/d, p_i) \in \mathbb{R}^2$.

For $i = 1, \ldots, d-1$, the maximum of $B_i^{(d)}(x)$ occurs at $x = i/d$ and equals $B_i^{(d)}(i/d)$. Later in this article, the non-negativity and the maximum property are used to generate halfspaces of the polytopes.

Multivariate Bernstein Representation. For multivariate polynomials, a Bernstein basis can be constructed using the tensorial product (TBB) of univariate Bernstein basis functions

$$(B_0^{(d)}(x_1), \ldots, B_d^{(d)}(x_1)) \times \ldots \times (B_0^{(d)}(x_n), \ldots, B_d^{(d)}(x_n))$$

Definition 2. *For each multivariate polynomial* p *of maximum degree* d*, there exists* **a multivariate Bernstein representation***, i.e., coefficients*

$\mathbf{0} \le p_\mathbf{a} \in \mathbb{R}$, $\mathbf{a} \le \mathbf{d}$ *such that* $p(\mathbf{x}) = \sum_{\mathbf{a}\le\mathbf{d}} p_\mathbf{a}\, B^{(d)}_\mathbf{a}(\mathbf{x})$ *is a linear combination of the Bernstein basis functions.*

The convex hull property, *i.e.,* $(\mathbf{x}, p(\mathbf{x})) \in \mathrm{conv}(\frac{\mathbf{a}}{\mathbf{d}}, p_\mathbf{a})$, extends to the TBB, which provides tight enclosures of a multivariate polynomial $p(\mathbf{x}) = \sum_{0\le\mathbf{a}\le\mathbf{d}} p_\mathbf{a}\, B_\mathbf{a}(\mathbf{x})$ for $\mathbf{x} \in [0,1]^n$ Garloff and Smith (2001). Bounds for the pointwise distance of a function $p(\mathbf{x})$ to the piecewise linear Bézier control mesh $M(p, D)(\mathbf{x})$ have been shown in Dahmen (1986)

$$|p(\mathbf{x}) - M(p,D)(\mathbf{x})| \le \max_{\mathbf{x}\in D, 1\le i,j\le n} |\delta_i\delta_j p(\mathbf{x})| w(D)^2$$

which means it approximates the range to the second order: It exist two points $\mathbf{x}, \mathbf{y} \in D$ such that

$$\begin{aligned} w(M(p,D)(D)) &= \overline{M(p,D)(D)} - \underline{M(p,D)(D)} = M(p,D)(\mathbf{y}) - M(p,D)(\mathbf{x}) \\ &\le |M(p,D)(\mathbf{y}) - p(\mathbf{y})| + |p(\mathbf{y}) - p(\mathbf{x})| + |p(\mathbf{x}) - M(p,D)(\mathbf{x})| \\ &\le 2\max\nolimits_{\mathbf{x}\in D, 1\le i,j\le n} |\delta_i\delta_j p(\mathbf{x})| \cdot w(D)^2 \\ &+ \max\nolimits_{\mathbf{x}\in D, 1\le i\le n} |\delta_i p(\mathbf{x})| \cdot \max\nolimits_{\mathbf{x}\in D, 1\le i,j\le n} |\delta_i\delta_j p(\mathbf{x})| \cdot w(D)^2 \end{aligned}$$

Degree of Multivariate Polynomial.

Definition 3. *Let* $p(\mathbf{x}) = \sum_{\mathbf{a}\le\mathbf{d}} p_\mathbf{a}\, \mathbf{x}^\mathbf{a}$ *be a multivariate polynomial in* $\mathbf{x} = (x_1 \ldots, x_n)$. *If* $a_i \le d$ *for all* $\mathbf{a}$ *with* $p_\mathbf{a} \ne 0$, *we say* **the multivariate polynomial** p **has maximum degree** d. *We call the space of these polynomials* $\Pi^{(d)}[x_1, \ldots, x_n]$. *In case* $p_\mathbf{a} = 0$ *for all* $\mathbf{a}$, $|\mathbf{a}| > d$, *we say* **the multivariate polynomial** p **has total degree** d. *We call the space of these polynomials* $\pi^{(d)}[x_1, \ldots, x_n]$.

Note that the vector space $\pi^{(d)}[x_1, \ldots, x_n]$ has dimension $(n+1)^d \in \Theta(n^d)$. But the smallest polynomial space with a TBB, containing it, is the space

$\Pi^{(d)}[x_1, \dots, x_n]$. Unfortunately, this space (and the tensorial Bernstein basis) has dimension $\Theta((d{+}1)^n)$, which is exponential in the number n of unknowns, even for linear systems ($d = 1$). This is not a problem when n is small, *e.g.,* when computing the intersection of three surfaces in 3-space, but the exponential size is problematic with a large number n of variables *e.g.,* when computing the coordinates of points with specified distances.

4. Polytope with Exponential Number of Non-Negative Variables

As stated in Section 3.2, the range of $p(\mathbf{x})$, $\mathbf{x} \in D$ is given by the convex hull of the coefficients in the TBB representation. This method is classical and used for example in Garloff and Smith (2001); Martin et al. (2002); Sherbrooke and Patrikalakis (1993).

The method computes all coefficients $p_{\mathbf{a}}$, $0 \leq \mathbf{a} \leq \mathbf{d}$ of the polynomial p in the TBB. The minimum coefficient $p_{\mathbf{a}}$ is a lower bound, and the maximum coefficient $p_{\mathbf{a}}$ is an upper bound for $p(\mathbf{x})$, $\mathbf{x} \in D$.

Using the linearity of the Bézier form, the coefficients of a polynomial $p \in \pi^d[x_1, \dots, x_n]$ can be obtained from the coefficients of the monomials in Q_n. For each monomial x_i, x_i^2, and $x_i x_j$ ($1 \leq i < j \leq n$), we have a component x_i, x_{ii}, and x_{ij} storing the coefficient of the corresponding Bézier basis function. The control points of the univariate monomial x_i are 0, 1/2, and 1; and for the multivariate monomial x_i^2, 0 for $x_i = 0$ and $x_i = 1/2$, and 1 for $x_i = 1$. Figure 1 shows the control points for the parabola $(x_i, x_{ii} = x_i^2)$. The multivariate monomial $x_i x_j$ can be represented as the tensor product of three line segments. Figure 1 shows the nine control points of the surface patch $(x_i, x_j, x_i x_j)$.

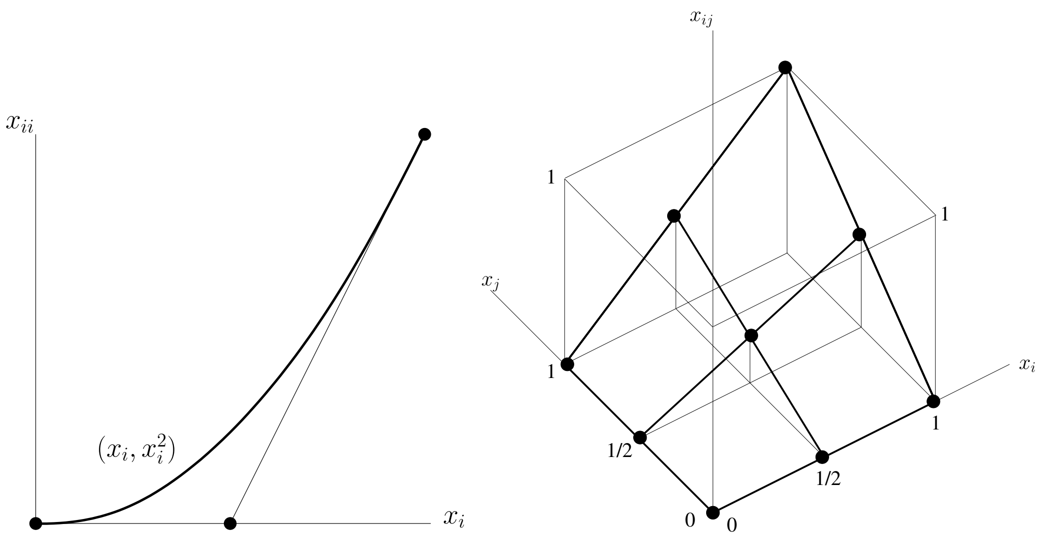

Figure 1: Left: Control points of the curve $(x_i, x_{ii} = x_i^2)$. Right: Control points of the patch $(x_i, x_j, x_{ij} = x_i x_j)$.

Example. For $n = 2$, the 3^2 control points p_i are

	x_1	x_2	x_{11}	x_{22}	x_{12}
$\mathbf{b}_1$	0	0	0	0	0
$\mathbf{b}_2$	0	1/2	0	0	0
$\mathbf{b}_3$	0	1	0	1	0
$\mathbf{b}_4$	1/2	0	0	0	0
$\mathbf{b}_5$	1/2	1/2	0	0	1/4
$\mathbf{b}_6$	1/2	1	0	1	1/2
$\mathbf{b}_7$	1	0	1	0	0
$\mathbf{b}_8$	1	1/2	1	0	1/2
$\mathbf{b}_9$	1	1	1	1	1

Definition 4. *The polytope $\mathcal{Q}_n^{(2)}$ is defined as the convex hull of the 3^n control*

points $\mathbf{b}_i$*, i.e., the set of points* $\sum_{i=1}^{3^n} \lambda_i b_i$ *subject to* $\lambda_i \geq 0$ *and* $\sum_{i=1}^{3^n} \lambda_i = 1$.

We computed properties of the polytopes $\mathcal{Q}_n^{(2)}$ for $n \leq 4$ with the method *lrs* (Avis (2000)).

	$n =$ 1	2	3	4
nb. coordinates	3	6	10	15
nb. vertices	3	$9 = 3^2$	$27 = 3^3$	$81 = 3^4$
nb. hyperfaces	3	18	173	46068
volume	$\frac{1}{4}$	$\frac{1}{96}$	$\frac{47}{645120}$	$\frac{375533}{4637432217600}$
	$=$	$=$	$\approx$	$\approx$
	0.25	$1.041\overline{6}$e-2	7.285e-5	8.098e-8

For the range of

$$p(\mathbf{x}) = \sum_i a_i x_i^2 + \sum_{\{(i,j)|i<j\}} b_{ij} x_i x_j + \sum_i c_i x_i + d,$$

we determine the vertex of the polytope $\mathcal{Q}_n^{(2)}$, which minimizes, and respectively maximizes, the corresponding linear objective function

$$\sum_i a_i x_{ii} + \sum_{\{(i,j)|i<j\}} b_{ij} x_{ij} + \sum_i c_i x_i + d$$

with variables λ_i, $(x_1, \ldots, x_n, x_{11}, \ldots x_{nn}, x_{12}, \ldots, x_{n-1,n}) = \sum_{i=1}^{3^n} \lambda_i \mathbf{b}_i$ and constraints $0 \leq \lambda_i$, $\sum_{i=1}^{3^n} \lambda_i = 1$.

5. Bernstein Polytopes

In this section, we consider the definition of polytopes by halfspaces derived from the Bernstein basis functions. For the multivariate monomial

$$\mathbf{x}^{(d_1,d_2,\ldots,d_k)} = x_1^{(d_1)} x_2^{(d_2)} \ldots x_k^{(d_k)} \tag{2}$$

the following inequalities of polynomials in $\Pi^{(d)}[x_1,\dots,x_k]$ are valid for $\mathbf{x} \in D$

$$0 \leq \quad B_{a_1}^{(d_1)}(x_1)\dots B_{a_k}^{(d_k)}(x_k) \tag{3}$$

$$B_{a_1}^{(d_1)}(x_1)\dots B_{a_k}^{(d_k)}(x_k) \quad \leq B_{a_1}^{(d_1)}(a_1/d_1)\dots B_{a_k}^{(d_k)}(a_k/d_k) \tag{4}$$

$$\text{for all } \mathbf{0} \leq \mathbf{a} \leq \mathbf{d}$$

Note that the right-hand side is the product of the maxima of the Bernstein functions. A new variable can be associated to each monomial of the tensorial canonical basis, and (3) and (4) provide two halfspaces for each $\mathbf{a} \leq \mathbf{d}$.

Example. Polytopes with $k = 1$, $d_1 = 2$ and $k = 2$, $|\mathbf{d}| = 2$ can be used for $\pi^2[x_1,\dots,x_n]$. The monomials are x_1^2 and x_1x_2, of which there is a number of n and $\binom{n}{2}$.

Definition 5. *The polytope resulting from $k = 1$, $d_1 = 2$, and $k = 2$, $d_1 = d_2 = 1$ in inequality (3) is denoted by $\mathcal{P}_n^{(2)}$ (Fuenfzig et al. (2009)). The polytope resulting from $k = 1$, $d_1 = 2$, and $k = 2$, $d_1 = d_2 = 1$ in inequalities (3) and (4) is denoted by $\mathcal{P}_n'^{(2)}$.*

The number of inequalities is $2 \cdot 2^2$ and $2 \cdot 3$. The polytope in (x_i, x_{ii}) for $k = 1$, $|\mathbf{d}| = 2$ is a triangle, shown in Figure 2. The polytope in (x_i, x_j, x_{ij}) for $k = 2$, $|\mathbf{d}| = 2$ is a tetrahedron, shown in Figure 3. This tetrahedron is the minimal convex set containing (x_i, x_j, x_ix_j).

Example. Similarly, polytopes with $k = 1$, $d_1 = 3$ and $k = 2$, $|d| = 3$, and $k = 3$, $d_1 = d_2 = d_3 = 1$ can be used for $\pi^3[x_1,\dots,x_n]$.

Definition 6. *The polytope resulting from $k = 1$, $d_1 = 3$ and $k = 2$, $|d| = 3$, and $k = 3$, $d_1 = d_2 = d_3 = 1$ in inequality (3) is denoted by $\mathcal{P}_n^{(3)}$. The*

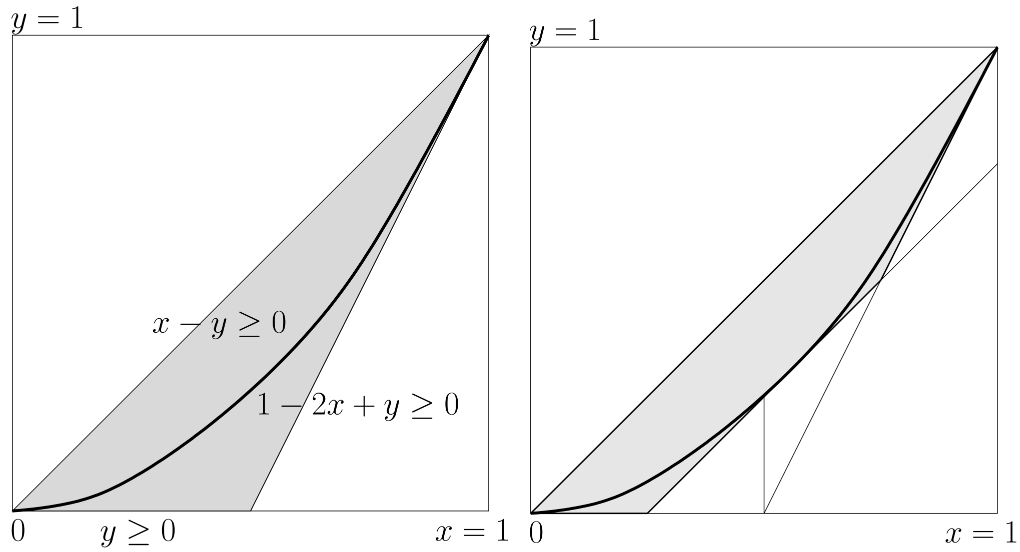

Figure 2: Left: The Bernstein polytope encloses the curve $(x, y = x^2)$ for $(x, y) \in [0, 1]^2$. Its bounding halfspaces are $B_0(x) = (1-x)^2 = y-2x+1 \geq 0$, $B_1(x) = 2x(1-x) = 2x-2y \geq 0$, $B_2(x) = x^2 = y \geq 0$. Right: A fourth constraint $(x - 1/2)^2 = x^2 - x + 1/4 \geq 0 \rightarrow x - y \leq 1/4$.

polytope resulting from $k = 1$, $d_1 = 3$ *and* $k = 2$, $|d| = 3$, *and* $k = 3$, $d_1 = d_2 = d_3 = 1$ *in inequalities (3) and (4) is denoted by* $\mathcal{P}_n'^{(3)}$.

The monomials are x_1^3, $x_1^2 x_2$, $x_1 x_2^2$ and $x_1 x_2 x_3$, of which there is a number of n, $\binom{n}{2}$, and $\binom{n}{3}$, respectively. The number of inequalities is $2 \cdot 4$, $4 \cdot 3 \cdot 2$, and $2 \cdot 2^3$. The polytope in $(x, y = x^2, z = x^3)$ for $k = 1$, $d_1 = 3$ is again a tetrahedron, shown in Figure 4.

Example. Polytope with $k = n$, $\mathbf{d} = \{2\}^n$. The only monomial is $\mathbf{x^d}$. The number of inequalities is $2 \cdot 3^n$, and the number of variables is 3^n: $\mathbf{x^a}$, $\mathbf{a} \leq (d_1, \ldots, d_n)$. For a factor $\mathbf{x^e} \in \pi^l[x_1, \ldots, x_n]$ of $\mathbf{x^d}$, its representation in the

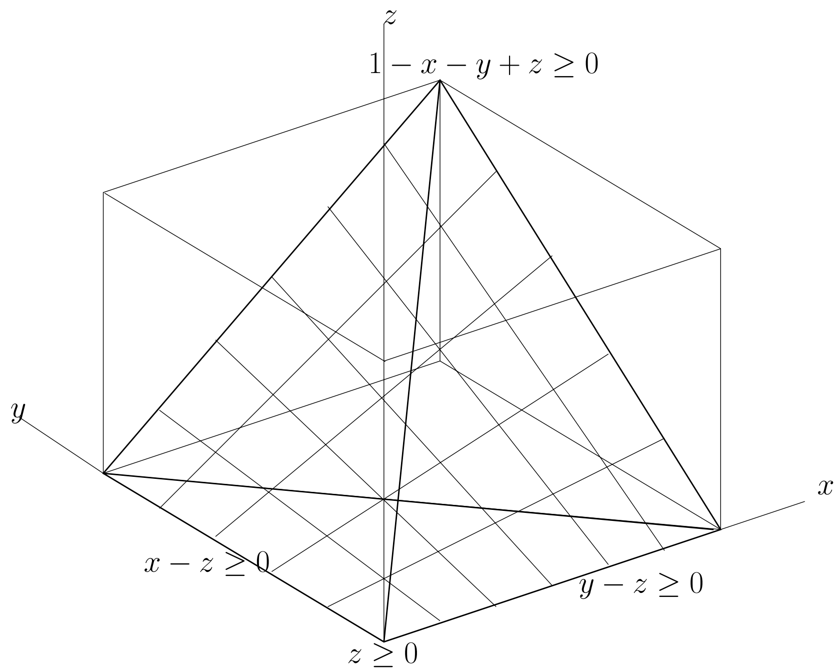

Figure 3: The Bernstein polytope enclosing the surface patch $(x, y, z = xy)$. Then the inequalities of bounding halfspaces are linearizations of $B_i^{(1)}(x)B_j^{(1)}(y) \geq 0$ with $i = 0, 1$, $j = 0, 1$: $B_0^{(1)}(x)B_0^{(1)}(y) = (1-x)(1-y) = 1-x-y+z \geq 0$, $B_0^{(1)}(x)B_1^{(1)}(y) = y - z \geq 0$, $B_1^{(1)}(x)B_0^{(1)}(y) = x - z \geq 0$, and $B_1^{(1)}(x)B_1^{(1)}(y) = z \geq 0$.

TBB is

$$\mathbf{x}^{\mathbf{e}} = \sum\nolimits_{0\leq\mathbf{b}\leq\mathbf{e}} b_{\mathbf{b}} B_{\mathbf{b}}^{\mathbf{d}}(\mathbf{x}) \cdot \prod\nolimits_{\{i:b_i=0\}} B_0^{d_i}(x_i) + \ldots + B_{d_i}^{d_i}(x_i)$$
$$\text{where } B_0^{d_i}(x_i) + \ldots + B_{d_i}^{d_i}(x_i) = 1.$$

If we substitute the polynomial $B_{\mathbf{b}}^{\mathbf{d}}(\mathbf{x})$ by a variable $B_{\mathbf{b}}^{\mathbf{d}} := \max_{\mathbf{x}} B_{\mathbf{b}}^{\mathbf{d}}(\mathbf{x}) = \prod_i \max B_{b_i}^{d_i}(x_i)$ giving its maximum value, then it is $B_{\mathbf{e}}^{\mathbf{d}} \geq B_{\mathbf{a}}^{\mathbf{d}}$ for $\mathbf{e}$ with $\mathrm{supp}_{\mathbf{d}}(\mathbf{e}) \subseteq \mathrm{supp}_{\mathbf{d}}(\mathbf{a})$. Because of this property, the inequalities with respect to $\mathbf{x}^{\mathbf{d}}$ are sharper than the ones for a factor $\mathbf{x}^{\mathbf{e}}$. In consequence, the polytope $\mathcal{Q}_n^{(2)}$ is contained in the polytope $\mathcal{P}_n^{(2)}$.

For small $n \leq 4$, the method *lrs* can explicitly compute a vertex repre-

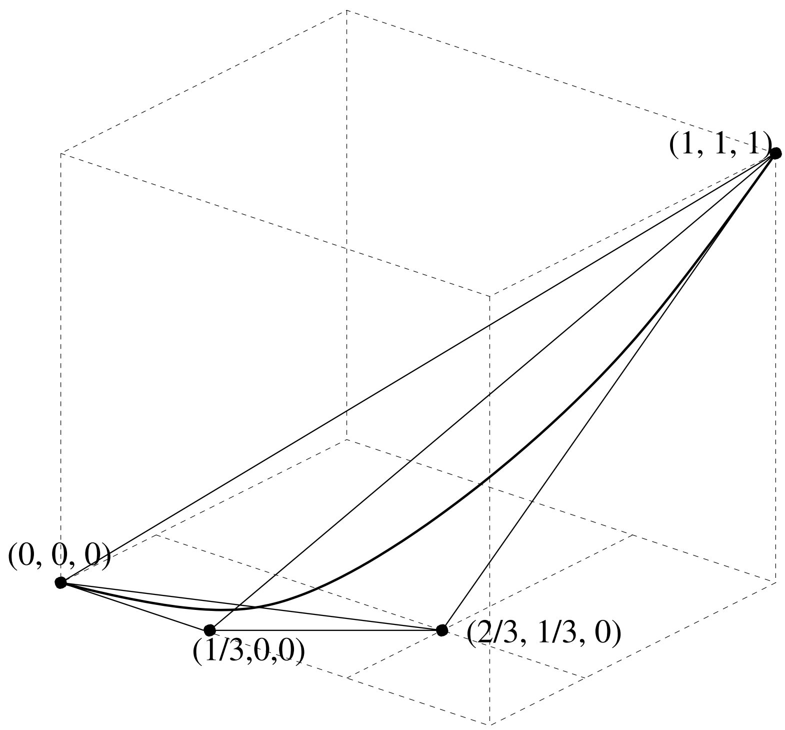

Figure 4: The Bernstein polytope, a tetrahedron, enclosing the curve $(x, y = x^2, z = x^3)$ with $x \in [0, 1]$. Its vertices are $v_0 = (0, 0, 0)$, $v_1 = (1/3, 0, 0)$, $v_2 = (2/3, 1/3, 0)$ and $v_3 = (1, 1, 1)$. v_0 lies on $B_1 = B_2 = B_3 = 0$, v_1 on $B_0 = B_2 = B_3 = 0$, etc. $B_0(x) = (1-x)^3 \geq 0 \Rightarrow 1-3x+3y-z \geq 0$, $B_1(x) = 3x(1-x)^2 \geq 0 \Rightarrow 3x-6y+3z \geq 0$, $B_2(x) = 3x^2(1-x) \geq 0 \Rightarrow 3y - 3z \geq 0$, $B_3(x) = x^3 \geq 0 \Rightarrow 3z \geq 0$.

sentation and the properties of the Bernstein polytope $\mathcal{P}_n^{(2)}$, $\mathcal{P}'^{(2)}_n$, and $\mathcal{P}''^{(2)}_n$ (with inequalities (3) and $(x_i - x_j)^2 = x_i^2 - 2x_i x_j + x_j^2 \geq 0$, $i < j$). The following table gives the projective dimension (*i.e.,* the number of homogeneous coordinates, subtract one for the affine dimension), the number of hyperfaces, the number of vertices, and the volume. An entry "?" denotes that the computation was stopped after considerable time.

	$n=$			
	1	2	3	4
$\mathcal{P}_n^{(2)}$				
nb. coordinates	3	6	10	15
nb. hyperfaces	3	10	21	36
nb. vertices	3	14	116	1688
volume	$\frac{1}{4}$	$\frac{1}{60}$	$\frac{1}{2688}$	?
	$=$	$=$	$\approx$	
	0.25	$0.01\overline{6}$	0.000372	
$\mathcal{P}'^{(2)}_n$				
nb. coordinates	3	6	10	15
nb. hyperfaces	4	12	24	40
nb. vertices	4	28	464	17744
volume	$\frac{3}{16}$	$\frac{1}{120}$	$\frac{389}{3440640}$	?
	$=$	$=$	$\approx$	
	0.1875	$0.008\overline{3}$	0.000113	
$\mathcal{P}''^{(2)}_n$				
nb. coordinates	3	6	10	15
nb. hyperfaces	4	13	27	46
nb. vertices	4	26	525	42307
volume	$\frac{3}{16}$	$\frac{109}{15360}$	?	?
	$=$	$\approx$		
	0.1875	0.007096		

6. Polytope Projections

To illustrate the polytope definitions, we show projections of the polytopes for $|\mathbf{d}| = 2$, $n = 2$ on a plane, where they are significantly different. In Figure 5, the polytope $\mathcal{Q}_2^{(2)}$ corresponding to the tensorial Bernstein basis is the smallest. Subtle differences to the polytope $\mathcal{P}_2^{(2)}$ result from the additional inequalities in $\mathcal{P}'^{(2)}_2$ and $\mathcal{P}''^{(2)}_2$.

Furthermore, it is possible to compute range bounds for a multivariate polynomial p by interval inclusion functions, introduced in Section 3.1. A polytope representation of the quadratic patch Q_n using the natural inclusion function is given by the boxes $x_i \in [0,1]$, $x_i^2 \in [0,1]$ and $x_i x_j \in [0,1]$ for $x_i, x_j \in [0,1]$. We denote the polytope resulting from the variable intervals by N. The natural inclusion function has also been used in Yamamura and Fujioka (2003) for solving nonlinear equations using the dual simplex method. Similarly, a polytope representation of the quadratic patch Q_n using the centered inclusion function is given by the boxes $x_i \in [0,1]$, $x_i^2 \in \frac{1}{4} + 2[0,1][-\frac{1}{2}, \frac{1}{2}] = [-0.75, 1.25]$ and $x_i x_j \in \frac{1}{4} + 2[0,1][-\frac{1}{2}, \frac{1}{2}] = [-0.75, 1.25]$ for $x_i, x_j \in [0,1]$. We denote the polytope resulting from the variable intervals by $\mathcal{C}$. Figure 5 shows the polytopes resulting from interval arithmetic in the last row. In comparison, these are the largest.

In Figure 6, we show the range bounds computed for an example in the bivariate case $n = 2$. The polynomial $p(x,y) = (1 - x - y - xy)^2 - 2 = x^2 + y^2 - 2x - 2y + x^2y + xy^2 + x^2y^2 - 1$ can be written of total degree-2 as $p_2 = -2x - 2y + x^2 + 2xz_1 + y^2 + 2z_1y + z_1^2 - 1$, $z_1 = xy$ and of total degree-3 as $p_3 = -2x - 2y + x^2 + 2z_1 + y^2 + 2xy^2 + z_1y - 1$, $z_1 = x^2y$. The exact range is $[-2, 2]$. On the polytopes $\mathcal{P}_2^{(2)}$ gives the range bound $[-3, 2]$. $\mathcal{P}_2^{(3)}$

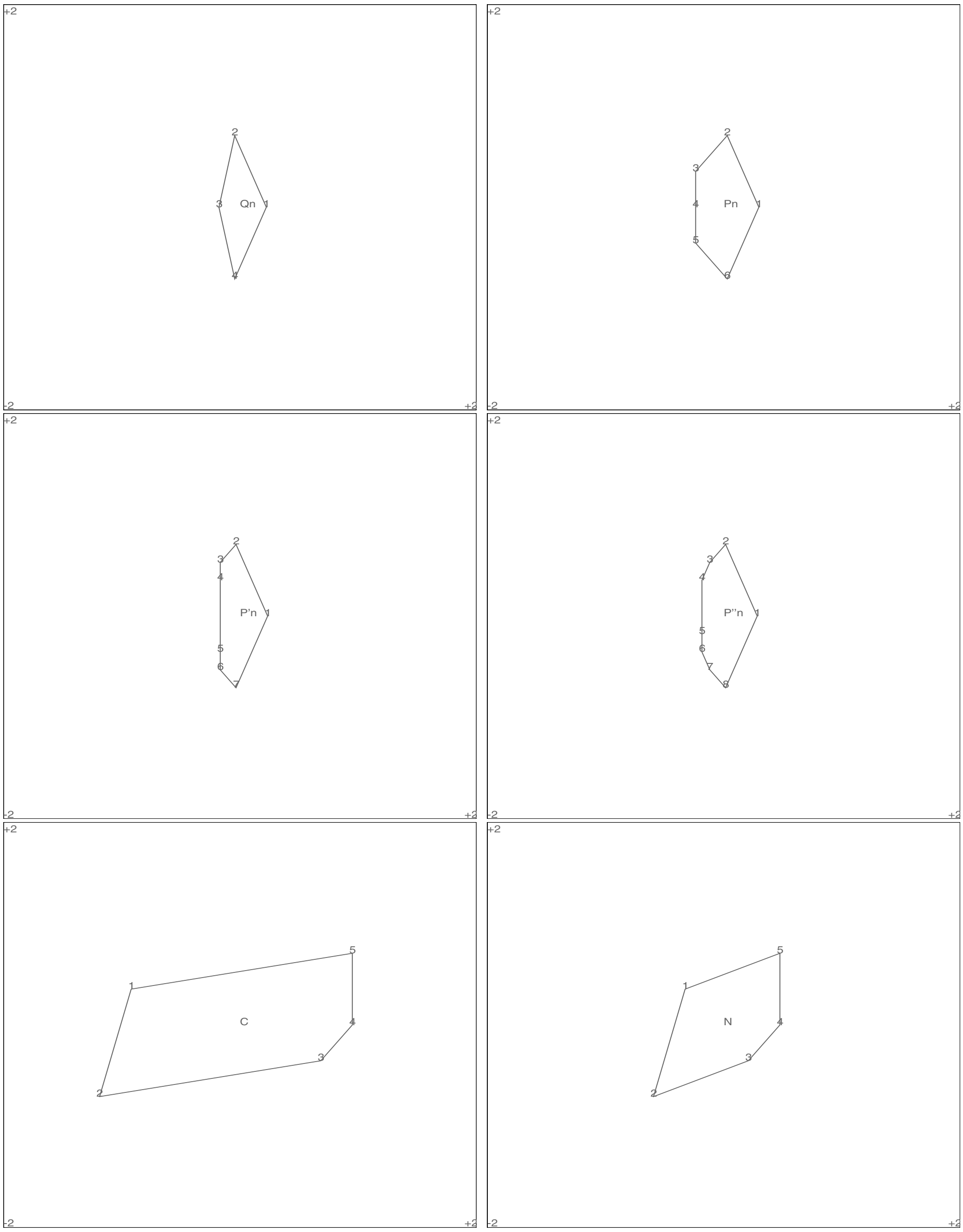

Figure 5: Projection of the polytopes $\mathcal{Q}_2^{(2)}$, $\mathcal{P}_2^{(2)}$, ${\mathcal{P}'}_2^{(2)}$, ${\mathcal{P}''}_2^{(2)}$, $\mathcal{C}$, and $\mathcal{N}$ (left to right and top to bottom) for $n = 2$, $N = 5$ on the plane given by the directions $(-2, -2, 1, 1, 2)/\sqrt{14}$ and $(-\frac{1}{2}, \frac{1}{2}, 0, 0, 0)/\sqrt{\frac{1}{2}}$.

on p_2 gives the range bound $[-2.\overline{6}, 2]$, and $\mathcal{P}_2^{(3)}$ on p_3 gives the range bound $[-3, 3]$. The two inclusion functions using interval arithmetic give the range bounds $[-6, 5]$ and $[-4.0625, 2.0625]$ respectively.

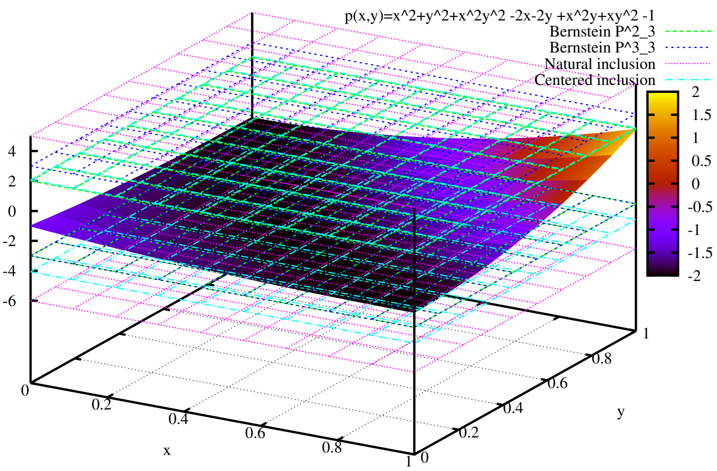

Figure 6: Range approximations for polynomial $p(x, y) = (1 - x - y - xy)^2 - 2 = x^2 + y^2 - 2x - 2y + x^2y + xy^2 + x^2y^2 - 1$, $x, y \in [0, 1]$. The exact range is $[-2, 2]$.

7. Comparison of Range Bounds

We measure the average width of the ranges of 100 random polynomials of total degree-2 by linear programming on each of the polytopes and by interval arithmetic.

As random polynomials, we consider two classes of polynomials. For general polynomials $p(\mathbf{x}) = \sum_i a_i x_i^2 + \sum_{\{(i,j)|i<j\}} b_{ij} x_i x_j + \sum_i c_i x_i + d$, we choose all coefficients a_i, b_{ij}, c_i, d equally random distributed in the interval $[-100, 100]$. Note that polynomials encountered in practice are sparse, *i.e.*,

most of the $\Theta(n^2)$ coefficients are zero.

In interval Newton solvers, preconditioning yields to special polynomials for the Newton map $N(\mathbf{x}) = \mathbf{x} - P(\mathbf{x})(P'(\mathbf{m}))^{-1}$. At the domain center $m_i = 1/2$, $i = 1, \ldots, n$, all derivatives of the polynomials in $N(\mathbf{x})$ vanish. In this class of polynomials $p(\mathbf{x}) = \sum_i a_i(x_i - 1/2)^2 + \sum_{\{(i,j)|i<j\}} b_{ij}(x_i - 1/2)(x_j - 1/2) + \sum_i c_i(x_i - 1/2) + d$, we choose coefficients a_i, b_{ij}, c_i, d equally random distributed in the interval $[-100, 100]$.

Figures 7 and 8 for degree $\mathbf{d} = 2$ (quadratic), and Figure 9 and Figure 10 for degree $|\mathbf{d}| = 3$ (cubic) give the average width of the polynomial ranges with its standard deviation as a function of the number n of variables. For small n and $|\mathbf{d}| = 2$, the differences between the range widths of all methods are small. For large n, the range widths provided by the Bernstein polytope $\mathcal{P}_n$ or one of its variants are significantly better than the ones provided by interval inclusion functions on the domain. For higher degree $|\mathbf{d}| = 3$, the interval inclusion functions on the domain $[0, 1]^n$ already have significant over-estimation but are fast to compute by evaluation of the polynomial. In comparison to the minimum/maximum-bounds with the TBB polytope, the range widths provided by the Bernstein polytopes are only slightly larger (15% for $n = 10$).

In the Bernstein polytope $\mathcal{P'}_n^{(2)}$ (Figure 2, right), the widths' improvements are not large, the average widths for $n = 10$ decreased by 6% only. The running times are also very similar. The runtime of linear programming by the simplex algorithm on the polytopes is difficult to state precisely, as it depends on the number of inequalities and the start basis.

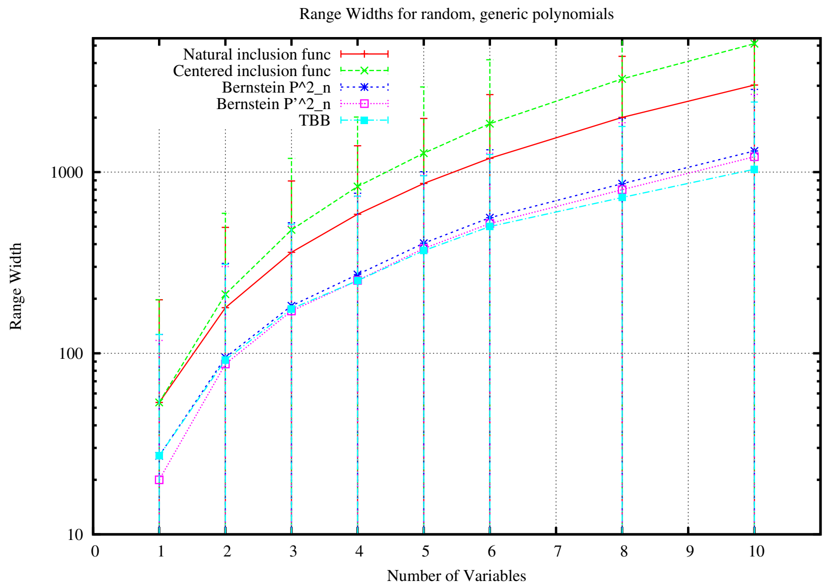

Figure 7: Statistics of range widths for random, quadratic polynomials (average widths with its standard deviation).

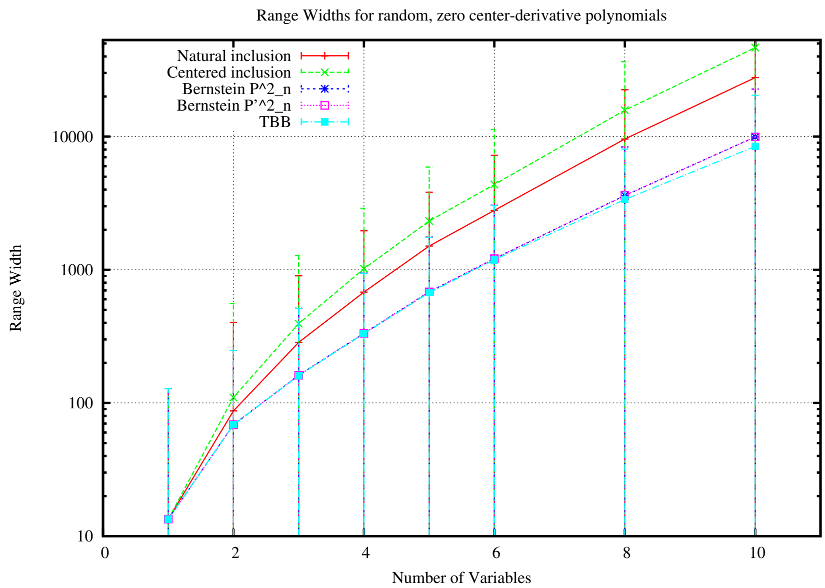

Figure 8: Statistics of range widths for random, quadratic polynomials with zero center-derivatives (average widths with its standard deviation).

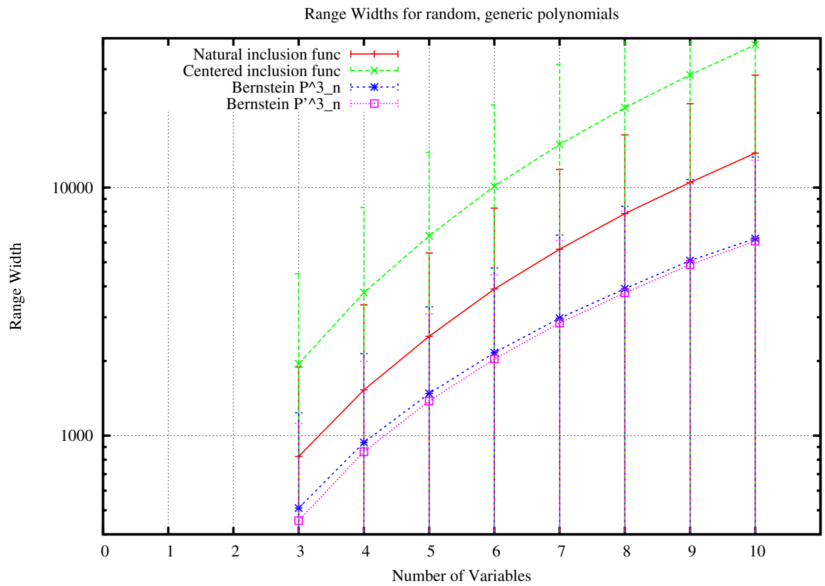

Figure 9: Statistics of range widths for random, cubic polynomials (average widths with its standard deviation).

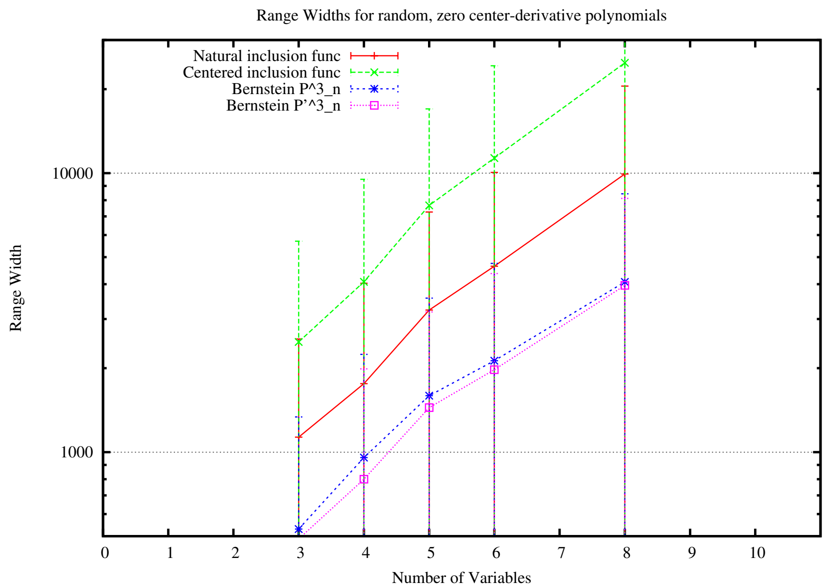

Figure 10: Statistics of range widths for random, cubic polynomials with zero center-derivatives (average widths with its standard deviation).

8. Conclusion

In this paper, we presented polytopes defined by halfspaces based on the Bernstein basis functions for range bound computation of polynomials. We also included a polytope, defined by the coefficients of the TBB, which has an exponential number of vertices in the number n of variables. For the Bernstein polytopes, we looked in detail at the case of total degree-2 and total degree-3 with a polynomial number of hyperfaces (in the number n of variables). We computed the number of vertices, the number of hyperfaces, and the volume of all polytopes as far as possible for small n. An empirical comparison of the average range widths for random polynomials computed using these polytopes shows that these polynomial-size Bernstein polytopes provide only slightly worse bounds for $n \leq 10$. Although this difference will increase with even larger n, the method using the Bernstein polytope is the only one with acceptable runtime for range computation in practice. For using the polytopes in practice, a linear program solver using a simplex method or an interior-point method is required.

9. Acknowledgments

This research work has been funded in part by Conseil Régional de Bourgogne and by NPRP grant number NPRP $09-906-1-137$ from the Qatar National Research Fund (a member of The Qatar Foundation).

References

Araya, I., Neveu, B., Trombettoni, G., 2009. An interval constraint propagation algorithm exploiting monotonicity. Int. Workshop IntCP , 65–83.

Avis, D., 2000. lrs: A revised implementation of the reverse search vertex enumeration algorithm. Polytopes - Combinatorics and Computation (G. Kalai & G. Ziegler eds.) , 177–198.

Beaumont, O., 1999. Algorithmique pour les intervalles. Ph.D. thesis. Univ. Rennes 1, France.

Dahmen, W., 1986. Subdivision algorithms converge quadratically. Journal of Computational and Applied Mathematics 16, 145 – 158.

Deza, A., Nematollahi, E., Terlaky, T., 2008. How good are interior point methods? Klee-Minty cubes tighten iteration-complexity bounds. Mathematical Programming 113, 1–14.

Farin, G., 2002. Curves and Surfaces for Computer-Aided Geometric Design — A Practical Guide. The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling, Morgan Kaufmann Publishers (Academic Press). 5^{th} edition. 499 pages.

Farouki, R., 1991. On the stability of transformations between power and Bernstein polynomial forms. Comput. Aided Geom. Des. 8, 29–36.

Farouki, R., Rajan, V., 1987. On the numerical condition of polynomials in Bernstein form. Comput. Aided Geom. Des. 4, 191–216.

Fuenfzig, C., Michelucci, D., Foufou, S., 2009. Nonlinear systems solver in floating point arithmetic using lp reduction, in: SPM 2009: ACM Symp. Solid and Physical Modeling, pp. 123–134.

Gaganov, A., 1985. Computational complexity of the range of the polynomial in several variables. Cybernetics , 418–425.

Garey, M., Johnson, D., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Co.

Garloff, J., Jansson, C., Smith, A., 2003. Lower bound functions for polynomials. Journal of Computational and Applied Mathematics .

Garloff, J., Smith, A., 2001. Investigation of a subdivision based algorithm for solving systems of polynomial equations. Journal of Nonlinear Analysis : Series A Theory and Methods 47, 167–178.

Martin, R., Shou, H., Voiculescu, I., Bowyer, A., Wang, G., 2002. Comparison of interval methods for plotting algebraic curves. Comput. Aided Geom. Des. 19, 553–587.

Moore, R.E., Kearfott, R.B., Cloud, M.J., 2009. Introduction to Interval Analysis. SIAM Philadelphia.

Schrijver, A., 1986. Theory of Linear and Integer Programming. J. Wiley and Sons.

Sherbrooke, E., Patrikalakis, N., 1993. Computation of the solutions of nonlinear polynomial systems. Comput. Aided Geom. Des. 10, 379–405.

Smith, A., 2009. Fast construction of constant bound functions for sparse polynomials. Journal of Global Optimization 43, 445–458.

Yamamura, K., Fujioka, T., 2003. Finding all solutions of nonlinear equations using the dual simplex method. Journal of Computational and Applied Mathematics 1-2, 587–595.