Tight bound functions for multivariate polynomials with application to the reliable analysis of structural frames

J. Garloff, A. P. Smith

Faculty of Computer Science, HTWG (University of Applied Sciences) Konstanz, Germany,
{garloff,smith}@htwg-konstanz.de

We consider the computation of tight bounds for multivariate polynomials based on the expansion of such a polynomial into Bernstein polynomials. Bound functions may be employed for a solution to a constrained global optimization problem, where the objective function and the functions describing the inequalities are all multivariate polynomials and the optimization problem is solved by a relaxation method in a branch and bound framework.

A method is presented for the computation of constant bound functions which uses an implicit representation of the Bernstein control points so that the computational complexity becomes nearly linear w.r.t. the number of the terms in the polynomial instead of exponential w.r.t. the number of the variables. The bound functions can be guaranteed also in the presence of data uncertainties and rounding errors.

We apply the approach to the enclosure of the solution set of a system of linear equations where the coefficients of the system are rational functions of parameters varying within given intervals, and present an example from the analysis of structural frames, where such parametric systems appear.

1 The Bernstein Form

We first recall some fundamental properties of the Bernstein expansion, cf. [2].

We define multiindices $i = (i_1, \ldots, i_n)^T$ as vectors, where the n components are nonnegative integers. The vector 0 denotes the multiindex with all components equal to 0. Comparisons are used entrywise. Also the arithmetic operators on multiindices are defined componentwise such that $i \odot l := (i_1 \odot l_1, \ldots, i_n \odot l_n)^T$, for $\odot = +, -, \times$, and $/$ (with $l > 0$). For $x \in \mathbb{R}^n$ its monomials are

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

For the n-fold sum we use the notation

$$\sum_{i=0}^{l} := \sum_{i_1=0}^{l_1} \ldots \sum_{i_n=0}^{l_n} . \tag{2}$$

The generalised binomial coefficient is defined by

$$\binom{l}{i} := \prod_{\mu=1}^{n} \binom{l_\mu}{i_\mu}. \tag{3}$$

An n-variate polynomial p,

$$p(x) = \sum_{i=0}^{l} a_i x^i, \quad x = (x_1, \ldots, x_n), \tag{4}$$

can be represented over

$$[x] := [\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n], \tag{5}$$
$$\underline{x} = (\underline{x}_1, \ldots, \underline{x}_n),\ \overline{x} = (\overline{x}_1, \ldots, \overline{x}_n),$$

as

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

where B_i is the *i-th Bernstein polynomial of degree* $l = (l_1, \ldots, l_n)$,

$$B_i(x) = \binom{l}{i} \frac{(x - \underline{x})^i (\overline{x} - x)^{l-i}}{(\overline{x} - \underline{x})^l}, \tag{7}$$

and the so-called *Bernstein coefficients* b_i of the same degree are given by

$$b_i = \sum_{j=0}^{i} \frac{\binom{i}{j}}{\binom{l}{j}} (\overline{x} - \underline{x})^j \sum_{\kappa=j}^{l} \binom{\kappa}{j} \underline{x}^{\kappa-j} a_\kappa, \quad 0 \leq i \leq l. \tag{8}$$

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:

$$\min_i \{b_i\} \leq p(x) \leq \max_i \{b_i\}, \ \ x \in [x]. \tag{9}$$

The traditional approach (see, for example, [2]) requires that all of the Bernstein coefficients are computed, and their minimum and maximum is determined. By use of de Casteljau's algorithm, this computation can be made efficient, with time complexity $\mathrm{O}(n\hat{l}^{n+1})$ and space complexity (equal to the number of Bernstein coefficients) $\mathrm{O}((\hat{l}+1)^n)$, where $\hat{l} = \max_{i=1}^n l_i$. This exponential complexity is a drawback of the traditional approach, rendering it infeasible for polynomials with moderately many (typically, 10 or more) variables.

In the following we consider a new method for the representation and computation of the Bernstein coefficients, which is especially well suited to sparse polynomials. For details and examples the reader is referred to [4].

1.1 Bernstein Coefficients of Monomials

Let $q(x) = x^r$, $x = (x_1, \ldots, x_n)$, for some $0 \leq r \leq l$. Then the Bernstein coefficients of q (of degree l) over $[x]$ (5) are given by

$$b_i = \prod_{m=1}^{n} b_{i_m}^{(m)}, \tag{10}$$

where $b_{i_m}^{(m)}$ is the i_mth Bernstein coefficient (of degree l_m) of the univariate monomial x^{r_m} over $[\underline{x}_m, \overline{x}_m]$. If the box $[x]$ is restricted to a single orthant of $\mathbb{R}^n$ then the Bernstein coefficients of q over $[x]$ are monotone with respect to each variable x_j, $j = 1, \ldots, n$.

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. 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 $\mathrm{O}(n(\hat{l}+1)^2)$. Given the exponent r 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, and then evaluate the monomial at these two vertices.

1.2 The Implicit Bernstein Form

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

$$p(x) = \sum_{j=1}^{t} a_{i_j} x^{i_j}, \quad 0 \leq i_j \leq l, \quad x = (x_1, \ldots, x_n), \tag{11}$$

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

$$b_i = \sum_{j=1}^{t} b_i^{\{j\}}, \quad 0 \leq i \leq l, \tag{12}$$

where $b_i^{\{j\}}$ are the Bernstein coefficients of the jth term of p.

Therefore one may implicitly store the Bernstein coefficients of each term, and compute the Bernstein coefficients as a sum of t products, only as needed. 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 $\mathrm{O}(nt(\hat{l}+1)^2)$ and space complexity $\mathrm{O}(nt(\hat{l}+1))$, as opposed to $\mathrm{O}((\hat{l}+1)^n)$ for the explicit form. Computing a single Bernstein coefficient from the implicit form requires $(n+1)t-1$ arithmetic operations.

1.3 Determination of the Bernstein Enclosure for Polynomials

We consider the determination of the minimum Bernstein coefficient; the determination of the maximum Bernstein coefficient is analogous. For simplicity we assume that $[x]$ is restricted to a single orthant.

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 the monotonicity of the Bernstein coefficients of monomials and employ uniqueness, monotonicity, and dominance tests, cf. [4] for details. As the examples therein show, it is often possible in practice to dramatically reduce the number of Bernstein coefficients that have to be computed.

2 Application to Structural Mechanics

A standard method for solving problems in structural mechanics, such as linear static problems, is the finite element method (FEM). In the case of linearised geometric displacement equations and linear elastic material behaviour, the method leads to a system of linear equations which in the presence of uncertain parameters becomes a parametric system. Treating the parametric system as an interval system and using a typical interval method in general results in solution intervals which are too wide for practical purposes.

We illustrate the usage of a new parametric solver based on bounding polynomial ranges by the implicit Bernstein form as described above. The improved efficiency is demonstrated by comparing both the computing time and the quality of the enclosure of the parametric solution set for the new solver and a previous solver which is based on the combination of the parametric residual iteration with the method for bounding the range of a rational function presented in [3]. To compare the quality of two enclosures $[a]$ and $[b]$ with $[a] \subseteq [b]$ we employ a measure $\mathcal{O}_\omega$ for the overestimation of

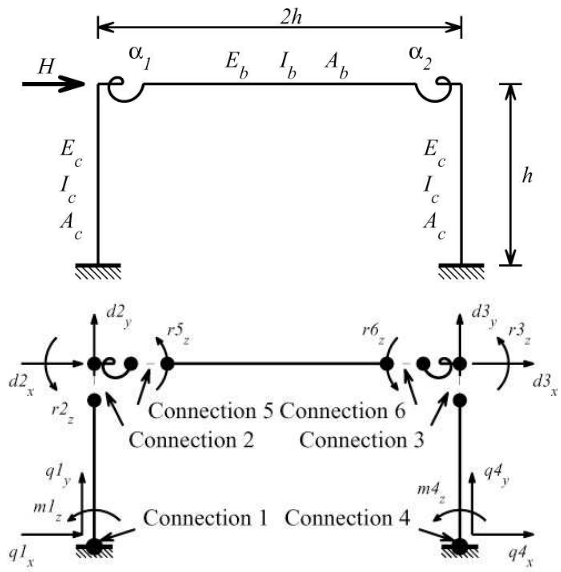

Figure 1: One-bay structural steel frame [1].

$[a]$ by $[b]$ which is defined by

$$\mathcal{O}_\omega([a],[b]) \;:=\; 100(1-\omega([a])/\omega([b])), \qquad (13)$$

where ω denotes the width of an interval.

2.1 One-Bay Steel Frame

We consider a simple one-bay structural steel frame, as shown in Figure 1, which was initially studied by interval methods in [1]. Following standard practice, the authors have assembled a parametric linear system of order eight and involving eight uncertain parameters. The typical nominal parameter values and the corresponding worst case uncertainties, as proposed in [1] are shown in Table 1, in SI-units.

As in [1], we solved the system first with parameter uncertainties which are 1% of the values presented in the last column of Table 1.

The example was run on a PC with an AMD Athlon-64 3GHz processor. The previous parametric solver finds an enclosure for the solution set in about 0.34 s, whereas the new solver needs only 0.05 s. The quality of the enclosures provided by both solvers is comparable. As shown in [3], the solution enclosure obtained by the para-

Table 1: Parameters involved in the steel frame example.

parameter		nominal value	uncertainty
Young modulus	E_b	$1.999 * 10^8$ kN/m^2	$\pm 2.399 * 10^7$ kN/m^2
	E_c	$1.999 * 10^8$ kN/m^2	$\pm 2.399 * 10^7$ kN/m^2
Second moment	I_b	$2.123 * 10^{-4}$ m^4	$\pm 2.123 * 10^{-5}$ m^4
	I_c	$1.132 * 10^{-4}$ m^4	$\pm 1.132 * 10^{-5}$ m^4
Area	A_b	$6.645 * 10^{-3}$ m^2	$\pm 6.645 * 10^{-4}$ m^2
	A_c	$9.290 * 10^{-3}$ m^2	$\pm 9.290 * 10^{-4}$ m^2
External force	H	23.600 kN	± 9.801 kN
Joint stiffness	α	$3.135 * 10^5$ kNm/rad	$\pm 1.429 * 10^5$ kNm/rad
Length	L_c	3.658 m, L_b 7.316 m	

Table 2: One-bay steel frame example with worst-case parameter uncertainties (Table 1). Interval end-points are multiplied by 10^5. The enclosure $[u]$ is compared to the combinatorial solution $[\tilde{h}]$.

	10^5* solution enclosure $[u]$	$\mathcal{O}_\omega([\tilde{h}],[u])$
$d2_x$:	[138.54954, 627.59325]	12.5
$d2_y$:	[0.29323100, 2.1529384]	8.0
$r2_z$:	[-129.02428, -22.381136]	23.7
$r5_z$:	[-113.21399, -17.95789]	25.6
$r6_z$:	[-105.9681, -17.64526]	25.0
$d3_x$:	[135.25570, 616.85513]	12.7
$d3_y$:	[-3.7624791, -0.41629803]	13.2
$r3_z$:	[-122.3362, -21.69878]	23.5

metric solver is better by more than one order of magnitude than that obtained in [1].

We next solve the same parametric linear system for the worst case parameter uncertainties in Table 1 ranging between about 10% and 46%. Firstly, we notice that the parametric solution depends linearly on the parameter H, so that we can obtain a better solution enclosure if we solve two parametric systems with the corresponding end-points for H. Secondly, enclosures of the hull of the solution set are obtained by subdivision of the worst case parameter intervals $(E_b, E_c, I_b, I_c, A_b, A_c, \alpha)^\top$ into $(2, 2, 2, 2, 1, 1, 6)^\top$ subintervals of equal width, respectively. We use more subdivision with respect to α since α is subject to the greatest uncertainty. The solution enclosure, obtained within 11 s, is given in Table 2. Moreover, the quality of the solution enclosure $[u]$ of the respective eight quantities is compared to the combinatorial solution $[\tilde{h}]$, i.e. the convex hull of the solutions to the point linear systems obtained when the parameters take all possible combinations of the interval end-points, which serves as an *inner* estimation of the solution enclosure. A good solution enclosure is quickly obtained for the worst-case parameter uncertainties.

2.2 Two-Bay Two-Story Frame

We consider a two-bay two-story steel frame subjected to lateral static forces and vertical uniform loads, where a system of 18 linear equations is obtained. In the first case we run this example with 13 uncertain parameters and then, allowing each element to adopt independent material parameters, it is extended to 37 interval parameters.

References

[1] G. Corliss, C. Foley and R. B. Kearfott, *Formulation for reliable analysis of structural frames.* Reliab. Comput. **13**, 125–147 (2007)

[2] J. Garloff, *Convergent bounds for the range of multivariate polynomials.* In K. Nickel (ed.), Interval Mathematics 1985, Lect. Notes in Comp. Sci., Vol. 212, pp. 37–56, Springer, Berlin (1986)

[3] E. D. Popova, *Solving linear systems whose input data are rational functions of interval parameters.* In T. Boyanov, et al. (eds.), NMA 2006, Lect. Notes in Comp. Sci., Vol. 4310, pp. 345–352, Springer, Berlin, Heidelberg, New York (2007)

[4] A. P. Smith, *Fast construction of constant bound functions for sparse polynomials.* J. Global Optim. **43(2–3)**, 445–458 (2009)