

to appear in *Symbolic Algebraic Methods and Verification Methods* edited by
G. Alefeld, S. Rump, J. Rohn, and T. Yamamoto, Springer

Solution of Systems of Polynomial Equations by Using Bernstein Expansion*

Jürgen Garloff and Andrew P. Smith

1 Introduction

Systems of polynomial equations appear in a great variety of applications, e.g., in geometric intersection computations (Hu et al. 1996), chemical equilibrium problems, combustion, and kinematics, to name only a few. Examples can be found in the monograph Morgan (1987). Following Sherbrooke and Patrikalakis (1993), most of the methods for the solution of such a system can be classified as techniques based on elimination theory, continuation, and subdivision. Elimination theory-based methods for constructing Gröbner bases rely on symbolic manipulations, making those methods seem somewhat unsuitable for larger problems. This class and also the second of the methods based on continuation frequently give us more information than we need since they determine all complex solutions of the system, whereas in applications often only the solutions in a given area of interest - typically a box - are sought. In the last category we collect all methods which apply a domain-splitting approach: Starting with the box of interest, such an algorithm sequentially splits it into subboxes, eliminating infeasible boxes by using bounds for the range of the polynomials under consideration over each of them, and ending up with a union of boxes that contains all solutions to the system which lie within the given box. Methods utilising this approach include interval computation techniques as well as methods which apply the expansion of a multivariate polynomial into Bernstein polynomials. In principle, each interval computation method for solving a system of nonlinear equations, cf. the monographs Kearfott (1996) and Neumaier (1990), can be applied to a polynomial system. Not surprisingly, techniques specially designed for polynomial systems are often more efficient in computing time. So we concentrate here on these methods. Jäger and Ratz (1995) combine the method of Gröbner bases with interval computations. Van Hentenryck et al. (1997) present a branch and prune approach which can be characterised as a

*Support from the Ministry of Education, Science, Research, and Technology of the Federal Republic of Germany under contract no.1706998 is gratefully acknowledged.

global search method using intervals for numerical correctness and for pruning the search space early.

As in the method to be presented in this paper, Sherbrooke and Patrikalakis (1993) use Bernstein expansion. Sequences of bounding boxes for the solutions to the polynomial system are generated by two different approaches: the first method projects control polyhedra onto a set of coordinate planes and the second exploits linear programming. But no use of the relationship between the Bernstein coefficients on neighbouring subboxes, cf. Subsection 2.2 below, is made and no existence test for a box to contain a solution, cf. Section 3, is provided.

Other applications of Bernstein expansion include applications to Computer Aided Geometric Design (e.g., Hu et al. 1996), to robust stability problems, cf. the survey article Garloff (2000), and to the solution of systems of polynomial inequalities (Garloff and Graf 1999).

The organisation of this paper is as follows: In the next section we briefly recall the Bernstein expansion, cf. Garloff (1993) and Zettler and Garloff (1998) and the references therein. The method is presented in Section 3. Examples are given in Section 4.

We concentrate here on real solutions, but we note that complex solutions can be found simply by separating each variable and each polynomial into their real and imaginary parts, doubling the order of the system.

2 Bernstein expansion

For compactness, we will use multi-indices $I = (i_1, \dots, i_l)$ and multi-powers $\mathbf{x}^I = x_1^{i_1} x_2^{i_2} \dots x_l^{i_l}$ for $\mathbf{x} \in \mathbf{R}^l$. Inequalities $I \leq N$ for multi-indices are meant componentwise, where $0 \leq i_k, k = 1, \dots, l$, is implicitly understood. With $I = (i_1, \dots, i_{r-1}, i_r, i_{r+1}, \dots, i_l)$ we associate the index $I_{r,k}$ given by $I_{r,k} = (i_1, \dots, i_{r-1}, i_r + k, i_{r+1}, \dots, i_l)$, where $0 \leq i_r + k \leq n_r$. We can then write an l -variate polynomial p in the form

$$p(\mathbf{x}) = \sum_{I \leq N} a_I \mathbf{x}^I, \quad \mathbf{x} \in \mathbf{R}^l, \quad (1)$$

and refer to N as the *degree* of p . Also, we write $\binom{N}{I}$ for $\binom{n_1}{i_1} \dots \binom{n_l}{i_l}$.

2.1 Bernstein transformation of a polynomial

In this subsection we expand a given l -variate polynomial (1) into Bernstein polynomials to obtain bounds for its range over an l -dimensional box. Without

loss of generality we consider the unit box $\mathbf{U} = [0, 1]^l$ since any nonempty box of \mathbf{R}^l can be mapped affinely onto this box.

For $\mathbf{x} = (x_1, \dots, x_l) \in \mathbf{R}^l$, the I th *Bernstein polynomial* of degree N is defined as

$$B_{N,I}(\mathbf{x}) = b_{n_1, i_1}(x_1) b_{n_2, i_2}(x_2) \cdots b_{n_l, i_l}(x_l),$$

where for $i_j = 0, \dots, n_j$, $j = 1, \dots, l$

$$b_{n_j, i_j}(x_j) = \binom{n_j}{i_j} x_j^{i_j} (1 - x_j)^{n_j - i_j}.$$

The transformation of a polynomial from its power form (1) into its *Bernstein form* results in

$$p(\mathbf{x}) = \sum_{I \leq N} b_I(\mathbf{U}) B_{N,I}(\mathbf{x}),$$

where the *Bernstein coefficients* $b_I(\mathbf{U})$ of p over \mathbf{U} are given by

$$b_I(\mathbf{U}) = \sum_{J \leq I} \frac{\binom{I}{J}}{\binom{N}{J}} a_J, \quad I \leq N. \quad (2)$$

We collect the Bernstein coefficients in an array $B(\mathbf{U})$, i.e., $B(\mathbf{U}) = (b_I(\mathbf{U}))_{I \leq N}$. In analogy to Computer Aided Geometric Design we call $B(\mathbf{U})$ a *patch*. For an efficient calculation of the Bernstein coefficients, which does not use (2), see Garloff (1986). All rounding errors appearing in the computation of the Bernstein coefficients can be taken into account similarly as in Fischer (1990).

In the following lemma, we list some useful properties of the Bernstein coefficients. Property (i) was given by Cargo and Shisha (1966) and property (ii) by Farouki and Rajan (1988).

Lemma 1 *Let p be a polynomial (1) of degree N . Then the following properties hold for its Bernstein coefficients $b_I(\mathbf{U})$ (2):*

i) Range enclosing property:

$$\forall \mathbf{x} \in \mathbf{U} : \min_{I \leq N} b_I(\mathbf{U}) \leq p(\mathbf{x}) \leq \max_{I \leq N} b_I(\mathbf{U}) \quad (3)$$

with equality in the left (resp., right) inequality if and only if $\min_{I \leq N} b_I(\mathbf{U})$ (resp., $\max_{I \leq N} b_I(\mathbf{U})$) is attained at a Bernstein coefficient $b_{i_k}(\mathbf{U})$ with $i_k \in$

$\{0, n_k\}$, $k = 1, \dots, l$.

ii) *Partial derivative:*

$$\frac{\partial p}{\partial x_r}(\mathbf{x}) = n_r \sum_{I \leq N_{r,-1}} [b_{I_{r,1}}(\mathbf{U}) - b_I(\mathbf{U})] B_{N_{r,-1}, I}(\mathbf{x}). \quad (4)$$

Lemma 2 *Let p be an l -variate polynomial and let $B(\mathbf{U})$ be the patch of its Bernstein coefficients on \mathbf{U} . Then the Bernstein coefficients of p on the m -dimensional faces of \mathbf{U} are just the coefficients on the respective m -dimensional faces of the patch $B(\mathbf{U})$, $0 \leq m \leq l - 1$.*

Proof: It is sufficient to prove the statement only for $m = l - 1$. Let $k \in \{1, \dots, l\}$. In the sequel we indicate by (k) that the quantity under consideration is taken without the contribution of the k th component, e.g., $I_{(k)} = (i_1, \dots, i_{k-1}, i_{k+1}, \dots, i_l)$. The I th Bernstein coefficient of

$$p(x_1, \dots, x_{k-1}, 0, x_{k+1}, \dots, x_l) = \sum_{I \leq N, i_k=0} a_I(\mathbf{x}^I)_{(k)}$$

considered as a polynomial in $l - 1$ variables is given by

$$\sum_{J \leq I, j_k=0} \frac{\binom{I}{J}_{(k)}}{\binom{N}{J}_{(k)}} a_J$$

which coincides with $b_{i_1 \dots i_{k-1} 0 i_{k+1} \dots i_l}$. Similarly, we obtain for $x_k = 1$

$$p(x_1, \dots, x_{k-1}, 1, x_{k+1}, \dots, x_l) = \sum_{I_{(k)} \leq N_{(k)}} c_{I_{(k)}}(\mathbf{x}^I)_{(k)},$$

where $c_{I_{(k)}} = \sum_{i_k=0}^{n_k} a_I$. The I th Bernstein coefficient of this polynomial is given by

$$\sum_{J_{(k)} \leq I_{(k)}} \frac{\binom{I}{J}_{(k)}}{\binom{N}{J}_{(k)}} c_{J_{(k)}}. \quad (5)$$

On the other hand, we have

$$b_{i_1 \dots i_{k-1} n_k i_{k+1} \dots i_l} = \sum_{J \leq I, i_k=n_k} \frac{\binom{I}{J}}{\binom{N}{J}} a_J$$

which coincides with (5).

Remark: Application of Lemma 2 for bounding the range of p over an edge of \mathbf{U} was given in the Edge Lemma in Zettler and Garloff (1998).

2.2 Sweep procedure

The bounds obtained by the inequalities (3) can be tightened if the unit box \mathbf{U} is bisected into subboxes and Bernstein expansion is applied to the polynomial p on these subboxes, i.e., to the polynomial shifted from each subbox back to \mathbf{U} . A sweep in the r th direction ($1 \leq r \leq l$) is a bisection perpendicular to this direction and is performed by recursively applying a linear interpolation. Let

$$\mathbf{D} = [\underline{d}_1, \bar{d}_1] \times \dots \times [\underline{d}_l, \bar{d}_l]$$

be any subbox of \mathbf{U} generated by sweep operations (at the beginning, we have $\mathbf{D} = \mathbf{U}$). Starting with $B^{(0)}(\mathbf{D}) = B(\mathbf{D})$ we set for $k = 1, \dots, n_r$

$$b_I^{(k)}(\mathbf{D}) = \begin{cases} b_I^{(k-1)}(\mathbf{D}) : i_r < k \\ (b_{I_{r,-1}}^{(k-1)}(\mathbf{D}) + b_I^{(k-1)}(\mathbf{D}))/2 : k \leq i_r. \end{cases}$$

To obtain the new coefficients, this is applied for $i_j = 0, \dots, n_j$, $j = 1, \dots, r-1, r+1, \dots, l$. Then the Bernstein coefficients on \mathbf{D}_0 , where the subbox \mathbf{D}_0 is given by

$$\mathbf{D}_0 = [\underline{d}_1, \bar{d}_1] \times \dots \times [\underline{d}_r, \hat{d}_r] \times \dots \times [\underline{d}_l, \bar{d}_l],$$

with \hat{d}_r denoting the midpoint of $[\underline{d}_r, \bar{d}_r]$, are obtained as $B(\mathbf{D}_0) = B^{(n_r)}(\mathbf{D})$. The Bernstein coefficients $B(\mathbf{D}_1)$ on the neighbouring subbox \mathbf{D}_1

$$\mathbf{D}_1 = [\underline{d}_1, \bar{d}_1] \times \dots \times [\hat{d}_r, \bar{d}_r] \times \dots \times [\underline{d}_l, \bar{d}_l]$$

are obtained as intermediate values in this computation, since for $k = 0, \dots, n_r$ the following relation holds (Garloff 1993):

$$b_{i_1, \dots, n_r-k, \dots, i_l}(\mathbf{D}_1) = b_{i_1, \dots, n_r, \dots, i_l}^{(k)}(\mathbf{D}).$$

A sweep needs $O(\hat{n}^{l+1})$ additions and multiplications, where $\hat{n} = \max\{n_i : i = 1, \dots, l\}$, cf. Zettler and Garloff (1998). Note that by the sweep procedure the explicit transformation of the subboxes generated by the sweeps back to \mathbf{U} is avoided. Fig. 1 illustrates the sweeping process for $l = 2$ and $r = 1$.

3 The method

Let n polynomials p_i , $i = 1, \dots, n$, in the real variables x_1, \dots, x_n and a box \mathbf{Q} in the \mathbf{R}^n be given. We want to know the set of all solutions to the equations

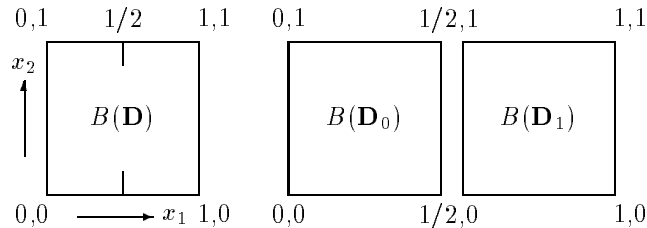


Fig. 1. Two new patches are obtained by a sweep in the first direction

$p_i(x) = 0, i = 1, \dots, n$, within \mathbf{Q}^1 . Without loss of generality we can assume that \mathbf{Q} is the unit box.

Our procedure is very simple: We take away from \mathbf{Q} all subboxes generated by sweeps for which there is a polynomial p_i being (strictly) positive or negative over the subbox. We check the sign of the polynomials by their Bernstein coefficients according to Lemma 1: If all Bernstein coefficients of a polynomial p_i are either positive or negative over a box, this box can not contain a solution.

After this pruning step we end up with a set of boxes of sufficiently small volume. All these boxes now undergo an existence test. In a first attempt we exploit the existence test given by Miranda (1941) which provides a generalisation of the fact that if a univariate continuous function f has a sign change at the endpoints of an interval then this interval contains a zero of f :

Theorem 1 (Miranda) *Let $\mathbf{X} = [\underline{x}_1, \bar{x}_1] \times \dots \times [\underline{x}_n, \bar{x}_n]$. Denote by $\mathbf{X}_i^- := \{\mathbf{x} \in \mathbf{X} \mid x_i = \underline{x}_i\}$ and $\mathbf{X}_i^+ := \{\mathbf{x} \in \mathbf{X} \mid x_i = \bar{x}_i\}$ the pair of opposite parallel faces of \mathbf{X} perpendicular to the i th coordinate direction.*

Let $\mathbf{F} = (f_1, \dots, f_n)^T$ be a continuous function defined on \mathbf{X} . If there is a permutation (v_1, \dots, v_n) of $(1, \dots, n)$ such that

$$f_i(\mathbf{x})f_i(\mathbf{y}) \leq 0 \text{ for all } \mathbf{x} \in \mathbf{X}_{v_i}^-, \mathbf{y} \in \mathbf{X}_{v_i}^+, i = 1, \dots, n, \quad (6)$$

then the equation $\mathbf{F}(\mathbf{x}) = 0$ has a solution in \mathbf{X} .

A short proof of Miranda's Theorem was given by Vrahatis (1989). An efficient method for checking all permutations is to be presented in Garloff and Smith (2001). Kioustelidis (1978), cf. Moore and Kioustelidis (1980) and Zuhe and Neumaier (1988), argued that the system $\mathbf{F}(\mathbf{x}) = 0$ should be preconditioned, i.e., it should be replaced by $A\mathbf{F}(\mathbf{x}) = 0$ with a suitably chosen matrix A . If \mathbf{F} is differentiable on \mathbf{X} , a reasonable choice for A is to take an approximation to the inverse of the Jacobian of \mathbf{F} at the midpoint of \mathbf{X} . If we apply

¹If the number of the equations does not coincide with the number of the variables we could indeed find an enclosure for the set of the solutions. This enclosure would consist of a union of boxes. But we would not be able to check easily whether such a box contains a solution.

Miranda's Theorem to the given polynomial system and use Bernstein expansion we can then make use of the easy calculation of the Bernstein form of the partial derivatives of a polynomial from its Bernstein form, cf. (4). Furthermore, the test required in (6) costs nearly nothing since the Bernstein coefficients of p on the faces of \mathbf{X} are known once the Bernstein coefficients of p on \mathbf{X} are computed, cf. Lemma 2.

We employ a heuristic sweep direction selection rule in an attempt to minimise the total number of subboxes which need to be processed. Such a rule may favour directions in which polynomials have large partial derivatives and in which the box edge lengths are larger, to avoid repetitive sweeps in a single direction. The method is tested with the following direction selection rule variants:

- **C**: The direction is set equal to the subdivision depth modulo the number of variables, plus one, viz. each direction is afforded an equal bias and is chosen in sequence. This is used as a control rule.
- **D1**: We compute an upper bound for the absolute value of the partial derivative (from its Bernstein form, cf. Zettler and Garloff (1998)) for each direction on each polynomial (patch). In each direction, we sum these values over all polynomials, and select the direction for which the product of box edge length and partial derivative sum is maximal.
- **D2**: As **D1**, except that we take the maximum of the upper bounds for the absolute value of the partial derivatives over all polynomials for each direction, and then multiply by the box edge length, as before.

4 Examples

The method was tested for some of the sample problems from Sherbrooke and Patrikalakis (1993) (**SP**) and Jäger and Ratz (1995) (**JR**), see Table 1.

The maximum subdivision depth is chosen to achieve the same tolerance as used in **SP** and **JR**, respectively. In each case, we record in Table 2 the total number of boxes processed (which is equal to twice the number of sweep operations, plus one), the number of Miranda tests performed, and the execution time (averaged over 5 repeat runs). All examples were run on a PC equipped with a 450MHz Pentium III processor.

Some categories of problems seem to require subdivision in all directions equally; for these cases we observe no appreciable difference in the output data between the control and the derivative methods. In other cases we notice that

Table 1. Example problems

Name	Q	Tolerance	Max subdivision depth	#Solutions
SP1	$[0, 1]^2$	10^{-8}	53	1
SP2	$[0, 21]$	10^{-7}	28	20
SP3	$[0, 1]^2$	10^{-8}	53	9
SP4	$[0, 1]^6$	10^{-8}	159	4
SP5	$[0, 1]^2$	10^{-14}	93	1
JR2	$[-1, 1]^3$	10^{-12}	123	2
JR4	$[-1, 1]^3$	10^{-12}	123	8

Table 2. Results for example problems

Example	Method	C	D1	D2
SP1	Number of boxes	205	183	183
	Miranda tests	2	1	1
	Time	0.02	0.02	0.02
SP2	Number of boxes	983	identical results	
	Miranda tests	20	(sweep in one direction only)	
	Time	0.12		
SP3	Number of boxes	2493	2245	2245
	Miranda tests	28	20	20
	Time	0.29	0.27	0.29
SP4	Number of boxes	6901	6315	6789
	Miranda tests	15	12	9
	Time	11.99	10.15	8.69
SP5	Number of boxes	5141	5131	5131
	Miranda tests	31	32	32
	Time	0.80	0.80	0.85
JR2	Number of boxes	3705	3571	3655
	Miranda tests	16	16	16
	Time	0.77	0.77	0.82
JR4	Number of boxes	8759	5895	6173
	Miranda tests	27	22	22
	Time	3.05	2.23	2.49

the choice of the sweep direction based on the absolute value of the partial derivative is effective in reducing the overall number of boxes that are processed and the number of Miranda tests required. There is very little difference between the sum and maximum variants. The methods were also tested for a range of subdivision depths, and it is worth noting that by making a small change, a greater variance in the number of Miranda tests (and the time taken) between them may be observed. We do not present the results here, since they would require tolerances which would not coincide with those used in **SP** and **JR**. The timings compare mostly favourably to those reported by **SP** and **JR**, but we should note that the processor capability available to us is approximately an order of magnitude faster.

5 Conclusions

In this paper, we have presented a further application of Bernstein expansion. It is an advantage of this approach that continua of solutions can also be enclosed. With its range enclosing property, the Bernstein form provides an alternative to the narrowing operators used by Van Hentenryck et al. (1997), cf. Granvilliers (2000), most likely speeding up the algorithm presented therein. On the other hand, a preprocessing step as used by Van Hentenryck et al. (1997) seems to be required in order to avoid unnecessarily many bisections and in order to approach the vicinity of the solutions earlier.

References

- Cargo, G.T., Shisha, O. (1966): The Bernstein form of a polynomial. *J. Res. Nat. Bur. Standards Sect. B*, vol. 70B (Math. Sci.), 1: 79-81
- Farouki, R.T., Rajan, V.T. (1988): Algorithms for polynomials in Bernstein form. *Computer Aided Geometric Design* 5: 1-26
- Fischer, H.C. (1990): Range computations and applications. In: Ullrich, C. (ed.): *Contributions to computer arithmetic and self-validating numerical methods*. J.C.Baltzer, Amsterdam, pp. 197-211
- Garloff, J. (1986): Convergent bounds for the range of multivariate polynomials. In: Nickel, K. (ed.): *Interval mathematics 1985*. Springer, Berlin Heidelberg New York, pp. 37-56 (Lecture notes in computer science, vol. 212)
- Garloff, J. (1993): The Bernstein algorithm. *Interval Comp.* 2: 154-168

- Garloff, J. (2000): Applications of Bernstein expansion to the solution of control problems. *Reliable Comp.* 6: 303-320
- Garloff, J., Graf, B. (1999): Solving strict polynomial inequalities by Bernstein expansion. In: Munro, N. (ed.): *The use of symbolic methods in control system analysis and design*. IEE, London, pp. 329-352
- Garloff, J., Smith, A.P. (2001): Improvements of a subdivision-based algorithm for solving systems of polynomial equations, to appear in the *Proceedings of the 3rd World Congress of Nonlinear Analysts, July 19-26 2000, Catania, Italy, special series of the Journal of Nonlinear Analysis, Elsevier Sci. Publ.*
- Granvilliers, L. (2000): Towards cooperative interval narrowing. In: *Proceedings 3rd Intern. Workshop on Frontiers of Combining Systems, FroCoS'2000, Nancy, France, Springer, Berlin Heidelberg New York (Lecture notes in artificial intelligence, vol. 1794)*
- Hu, Chun-Yi, Maekawa, T., Sherbrooke, E.C., Patrikalakis, N.M. (1996): Robust interval algorithm for curve intersections. *Computer-Aided Design* 28:495-506
- Jäger, C., Ratz, D. (1995): A combined method for enclosing all solutions of nonlinear systems of polynomial equations, *Reliable Comp.* 1: 41-64
- Kearfott, R.B. (1996): *Rigorous global search: continuous problems*, Kluwer Acad. Publ., Dordrecht Boston London
- Kioustelidis, J.B. (1978): Algorithmic error estimation for approximate solutions of nonlinear systems of equations. *Computing* 19: 313-320
- Miranda, C. (1941): Un'osservazione su un teorema di Brouwer. *Boll. Un. Mat. Ital. Ser.2, 3: 5-7*
- Moore, R.E., Kioustelidis, J.B. (1980): A simple test for accuracy of approximate solutions to nonlinear (or linear) systems. *SIAM J. Numer. Anal.* 17:521-529
- Morgan, A.P.(1987): *Solving polynomial systems using continuation for engineering and scientific problems*. Prentice Hall, Englewood-Cliffs, N.J.
- Neumaier, A. (1990): *Interval methods for systems of equations*. Cambridge University Press, Cambridge (*Encyclopedia of mathematics and its applications*)
- Sherbrooke, E.C., Patrikalakis, N.M. (1993): Computation of the solutions of nonlinear polynomial systems. *Computer Aided Geometric Design* 10: 379-405
- Van Hentenryck, P., McAllester, D., Kapur, D. (1997): Solving polynomial systems using a branch and prune approach. *SIAM J. Numer. Anal.* 34:797-827

Vrahatis, M.N. (1989): A short proof and a generalization of Miranda's existence theorem. Proc. Amer. Math. Soc. 107: 701-703

Zettler, M., Garloff, J. (1998): Robustness analysis of polynomials with polynomial parameter dependency using Bernstein expansion. IEEE Trans. Automat. Contr. 43:425-431

Zuhe, S., Neumaier, A. (1988): A note on Moore's interval test for zeros of nonlinear systems. Computing 40: 85-90