Combinatorial and Computational Geometry
MSRI Publications
Volume **52**, 2005

The Bernstein Basis and Real Root Isolation

BERNARD MOURRAIN, FABRICE ROUILLIER,
AND MARIE-FRANÇOISE ROY

Abstract. In this mostly expository paper we explain how the Bernstein basis, widely used in computer-aided geometric design, provides an efficient method for real root isolation, using de Casteljau's algorithm. We discuss the link between this approach and more classical methods for real root isolation. We also present a new improved method for isolating real roots in the Bernstein basis inspired by Roullier and Zimmerman.

Introduction

Real root isolation is an important subroutine in many algorithms of real algebraic geometry [Basu et al. 2003] as well as in exact geometric computations, and is also interesting in its own right.

Our approach to real root isolation is based on properties of the Bernstein basis. We first recall Descartes' Law of Signs and give a useful partial reciprocal to it. Section 2 contains the definition and main properties of the Bernstein basis. In the third section, several variants of real root isolation based on the Bernstein basis are given. In the fourth section, the link with more classical real root isolation methods [Uspensky 1948] is established. We end the paper with a few remarks on the computational efficiency of the algorithms described.

1. Descartes' Law of Signs

The *number of sign changes*, $\mathrm{V}(a)$, in a sequence $a = a_0, \ldots, a_p$ of elements in $\mathbb{R} \setminus \{0\}$ is defined by induction on p by

$$\begin{aligned} \mathrm{V}(a_0) &= 0, \\ \mathrm{V}(a_0, \ldots, a_p) &= \begin{cases} \mathrm{V}(a_1, \ldots, a_p) + 1 & \text{if } a_0 a_1 < 0, \\ \mathrm{V}(a_1, \ldots, a_p) & \text{if } a_0 a_1 > 0. \end{cases} \end{aligned}$$

This definition extends to any finite sequence a of elements in $\mathbb{R}$ by considering the finite sequence b obtained by dropping the zeros in a and defining $\mathrm{V}(a) = \mathrm{V}(b)$, with the convention $\mathrm{V}(\varnothing) = 0$.

Let $P = a_p X^p + \cdots + a_0$ be a univariate polynomial in $\mathbb{R}[X]$. We write $\mathrm{V}(P)$ for the number of sign changes in $a_0, \ldots, a_p$ and $\mathrm{pos}(P)$ for the number of positive real roots of P, counted with multiplicity.

We state the famous Descartes' law of signs, of 1636. (Descartes' text appears in [Struik 1969, pp. 90–91]. See also [Basu et al. 2003], for example, for a proof.)

THEOREM 1.1 (DESCARTES' LAW OF SIGNS).

(i) $\mathrm{pos}(P) \leq \mathrm{V}(P)$.
(ii) $\mathrm{V}(P) - \mathrm{pos}(P)$ *is even.*

In general, it is not possible to conclude much about the number of roots on an interval using only Theorem 1.1.

An instance where Descartes' law of signs permits a sharp conclusion is the following.

THEOREM 1.2. *Let*

$$\mathcal{D} = \{(x+iy) \in \mathbb{R}[i] \mid x < -\tfrac{1}{2}, (x+1)^2 + y^2 < 1\}$$

be the part of the open disk with center $(-1, 0)$ *and radius* 1 *which is to the left of the line* $x = -\frac{1}{2}$ *in* $\mathbb{R}^2 = \mathbb{R}[i]$. *If* $P \in \mathbb{R}[X]$ *is square-free and has either no roots or exactly one simple root in* $(0, +\infty)$, *and all its complex roots in* $\mathcal{D}$, *then* $\mathrm{V}(P) = 0$ *or* $\mathrm{V}(P) = 1$ *and*

(i) P *has one root in* $(0, +\infty)$ *if and only if* $\mathrm{V}(P) = 1$,
(ii) P *has no root in* $(0, +\infty)$ *if and only if* $\mathrm{V}(P) = 0$.

The proof of the theorem relies on the following lemmas.

LEMMA 1.3. *For* $A, B \in \mathbb{R}[X]$

$$\mathrm{V}(A) = 0,\ \mathrm{V}(B) = 0 \implies \mathrm{V}(AB) = 0.$$

PROOF. Obvious. □

LEMMA 1.4. *For* $A, B \in \mathbb{R}[X]$

$$\mathrm{V}(A) = 1,\ B = X + b,\ b \geq 0 \implies \mathrm{V}(AB) = 1.$$

PROOF. If $b = 0$, $\mathrm{V}(AB) = \mathrm{V}(A) = 1$. Now, let $b > 0$. Let

$$A = a_d X^d + a_{d-1} X^{d-1} + \cdots + a_0,$$

and suppose, without loss of generality, that $a_d = 1$. Since $\mathrm{V}(A) = 1$ and $a_d = 1$, there exists k such that

$$a_i \begin{cases} \geq 0 & \text{if } i > k, \\ < 0 & \text{if } i = k, \\ \leq 0 & \text{if } i < k. \end{cases} \tag{1–1}$$

Letting c_i be the coefficient of X^i in AB and making the convention that $a_{d+1} = a_{-1} = 0$, we have

$$c_i = \begin{cases} a_{i-1} + a_i b \geq 0 & \text{if } k+1 < i \leq d, \\ a_{k-1} + a_k b < 0 & \text{if } i = k, \\ a_{i-1} + a_i b \leq 0 & \text{if } i < k, \end{cases}$$

and $c_{d+1} = a_d > 0$. So, whatever the sign of c_{k+1}, $\mathrm{V}(AB) = 1$. □

LEMMA 1.5. *If* $\mathrm{V}(A) = 1$, $B = X^2 + bX + c$ *with* $b > 1$, $b > c > 0$, *then* $\mathrm{V}(AB) = 1$.

PROOF. Let $A = a_d X^d + a_{d-1} X^{d-1} + \cdots + a_0$ and suppose without loss of generality that $a_d = 1$. Since $\mathrm{V}(A) = 1$ and $a_d = 1$, there exists k such that (1–1) is satisfied. Letting c_i be the coefficient of X^i in AB and making the convention that $a_{d+2} = a_{d+1} = a_{-1} = a_{-2} = 0$, we have

$$c_i = \begin{cases} a_{i-2} + a_{i-1} b + a_i c \geq 0 & \text{for } k+2 < i \leq d+2, \\ a_{k-2} + a_{k-1} b + a_k c < 0 & \text{for } i = k, \\ a_{i-2} + a_{i-1} b + a_i c \leq 0 & \text{for } i < k. \end{cases}$$

The only way to have $\mathrm{V}(AB) > 1$ would be to have $c_{k+1} > 0, c_{k+2} < 0$, but this is impossible since

$$c_{k+2} - c_{k+1} = a_{k+2} c + a_{k+1}(b-c) + a_k(1-b) - a_{k-1} > 0.$$ □

PROOF OF THEOREM 1.2. Notice first that by Theorem 1.1, $\mathrm{V}(P) = 1$ implies that P has one root in $(0, +\infty)$, and $\mathrm{V}(P) = 0$ implies that P has no root in $(0, +\infty)$. Note also that

- if $X + a$ has its root in $(0, +\infty)$, then $a < 0$ and $\mathrm{V}(X + a) = 1$,
- if $X + b$ has its root in $(-\infty, 0]$, then $b \geq 0$ and $\mathrm{V}(X + b) = 0$,
- if $X^2 + bX + c$ has its roots in $\mathcal{D}$, then $b > 1, b > c > 0$ and $\mathrm{V}(X^2 + bX + c) = 0$.

Now decompose P into irreducible factors of degree 1 and 2 over $\mathbb{R}$. If P has one root a in $(0, +\infty)$, $\mathrm{V}(X + a) = 1$. Starting from $X + a$ and multiplying successively by the other irreducible factors of P, we get polynomials with sign variations equal to 1, using Lemma 1.4 and Lemma 1.5. Finally, $\mathrm{V}(P) = 1$.

If P has no root in $(0, +\infty)$, starting from 1 and multiplying successively by the irreducible factors of P, we get polynomials with sign variations equal to 0, using Lemma 1.3. Finally, $\mathrm{V}(P) = 0$. □

2. The Bernstein Basis

The Bernstein basis is widely used in computer-aided design [Farin 1990]. We recall some of its main properties, in order to use them for real root isolation in the next section.

NOTATION 2.1. Let P be a polynomial of degree $\leq p$. The *Bernstein polynomials* of degree p for c, d are the

$$B_{p,i}(c,d) = \binom{p}{i} \frac{(X-c)^i (d-X)^{p-i}}{(d-c)^p},$$

for $i = 0, \ldots, p$.

REMARK 2.2. Note that $B_{p,i}(c,d) = B_{p,p-i}(d,c)$ and that

$$B_{p,i}(c,d) = \frac{(X-c)}{d-c} \frac{p}{i} B_{p-1,i-1}(c,d).$$

Since the multiplicity of the polynomial $B_{p,i}(c,d)$ at $x = c$ is i and $B_{p,i}(c,d)$ is a polynomial of degree p, we immediately deduce that the polynomials $B_{p,i}(c,d)$, $i = 0, \ldots, p$ are linearly independent and form a basis of the vector space of polynomials of degree $\leq p$.

Here are some simple transformations, useful to understand the connection between the Bernstein basis and the monomial basis.

Reciprocal polynomial in degree p: $\mathrm{Rec}_p(P(X)) = X^p P(1/X)$. The nonzero roots of P are the inverses of the nonzero roots of $\mathrm{Rec}(P)$.

Contraction by ratio λ: for every nonzero λ, $\mathrm{C}_\lambda(P(X)) = P(\lambda X)$. The roots of $\mathrm{C}_\lambda(P)$ are of the form x/λ, where x is a root of P.

Translation by c: for every c, $\mathrm{T}_c(P(X)) = P(X-c)$. The roots of $\mathrm{T}_c(P(X))$ are of the form $x+c$ where x is a root of P.

PROPOSITION 2.3. *Let* $P = \sum_{i=0}^p b_i B_{p,i}(d,c) \in \mathbb{R}[X]$ *be of degree* $\leq p$. *Let*

$$\mathrm{T}_{-1}\left(\mathrm{Rec}_p(\mathrm{C}_{d-c}(\mathrm{T}_{-c}(P)))\right) = \sum_{i=0}^p c_i X^i.$$

Then

$$\binom{p}{i} b_i = c_{p-i}.$$

PROOF. Performing the contraction of ratio $d-c$ after translating by $-c$ transforms

$$\binom{p}{i} \frac{(X-c)^i (d-X)^{p-i}}{(d-c)^p} \quad \text{into} \quad \binom{p}{i} X^i (1-X)^{p-i}.$$

Translating by -1 after taking the reciprocal polynomial in degree p transforms

$$\binom{p}{i} X^i (1-X)^{p-i} \quad \text{into} \quad \binom{p}{i} X^{p-i}. \qquad \square$$

Let P be of degree p. We denote by $b = b_0, \ldots, b_p$ the coefficients of P in the Bernstein basis of c, d. Let $n(P; (c,d))$ be the number of roots of P in (c,d) counted with multiplicities.

PROPOSITION 2.4. (i) $\mathrm{V}(b) \geq n(P; (c,d))$.
(ii) $\mathrm{V}(b) - n(P; (c,d))$ *is even.*

PROOF. This follows immediately from Descartes' law of signs (Theorem 1.1), using Proposition 2.3. Indeed, the image of (c,d) under the translation by $-c$ followed by the contraction of ratio $d-c$ is $(0,1)$. The image of $(0,1)$ under the inversion $z \mapsto 1/z$ is $(1,+\infty)$. Finally, translating by -1 gives $(0,+\infty)$. □

We now describe a special case where the number $\mathrm{V}(b)$ coincides with the number of roots of P on (c,d). Let $d > c$, and $\mathcal{C}(c,d)_0$ be the closed disk with center $(c,0)$ and radius $d-c$, and let $\mathcal{C}(c,d)_1$ be the closed disk with center $(d,0)$ and radius $d-c$.

THEOREM 2.5 (THEOREM OF 2 CIRCLES). *If P is square-free and has either no root or exactly one simple root in (c,d) and P has no complex root in $\mathcal{C}(c,d)_0 \cup \mathcal{C}(c,d)_1$, then*

(i) *P has one root in (c,d) if and only if $\mathrm{V}(b) = 1$,*
(ii) *P has no root in (c,d) if and only if $\mathrm{V}(b) = 0$.*

PROOF. We identify $\mathbb{R}^2$ with $\mathbb{C} = \mathbb{R}[i]$. The image of the complement of $\mathcal{C}(c,d)_0$ (resp. $\mathcal{C}(c,d)_1$) under the translation by $-c$ followed by the contraction of ratio $d-c$ is the complement of $\mathcal{C}(0,1)_0$ (resp. $\mathcal{C}(0,1)_1$). The image of the complement of $\mathcal{C}(0,1)_0$ under the inversion $z \mapsto 1/z$ is

$$\{(x+iy) \in \mathbb{R}[i] \mid 0 < x^2+y^2 < 1\}.$$

The image of the complement of $\mathcal{C}(0,1)_1$ under the inversion $z \mapsto 1/z$ is

$$\{(x+iy) \in \mathbb{R}[i] \mid x < \tfrac{1}{2}\}.$$

The image of the complement of $\mathcal{C}(0,1)_0 \cup \mathcal{C}(0,1)_1$ under $z \mapsto 1/z$ is

$$\{(x+iy) \in \mathbb{R}[i] \mid 0 < x^2+y^2 < 1,\ x < \tfrac{1}{2}\}.$$

Translating this region by -1, we get the region

$$\mathcal{D} = \{(x+iy) \mid x < -\tfrac{1}{2},\ (x+1)^2+y^2 < 1\}$$

defined in Theorem 1.2.

The statement then follows from Theorem 1.2 and Proposition 2.3. □

Notice that this result which is a weaker version of the two-circles theorem presented in [Mehlhorn 2001], and related to [Ostrowski 1950], is given for the sake of simplicity. Indeed, one can use instead the two-circles $D\big(\frac{1}{2} \pm \frac{\mathbf{i}}{2\sqrt{3}}, \frac{1}{\sqrt{3}}\big)$, as proved in the works cited.

The coefficients $b = b_0, \ldots, b_p$ of P in the Bernstein basis of c, d give a rough idea of the shape of the polynomial P on the interval c, d. The *control line of P on $[c,d]$* is the union of the segments $[M_i, M_{i+1}]$ for $i = 0, \ldots, p-1$, with

$$M_i = \left(\frac{i\,d + (p-i)\,c}{p}, b_i \right).$$

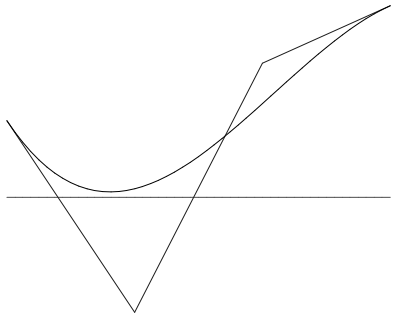

Figure 1. Graph of P on $[0,1]$ and the control line.

It is clear from the definitions that the graph of P goes through M_0 and M_p and that the line M_0, M_1 (resp. M_{p-1}, M_p) is tangent to the graph of P at M_0 (resp. M_p).

EXAMPLE 2.6. We take $p = 3$, and consider the polynomial P with coefficients $(4, -6, 7, 10)$ in the Bernstein basis for $0, 1$

$$(1-X)^3,\ 3(1-X)^2X,\ 3(1-X)X^2,\ X^3.$$

We draw the graph of P on $[0,1]$, the control line, and the X-axis in Figure 1.

The *control polygon of P on $[c,d]$* is the convex hull of the points M_i for $i = 1, \ldots, p$.

EXAMPLE 2.7. Continuing Example 2.6, we draw the graph of P on $[0,1]$ and the control polygon in Figure 2.

An important and well-known property of the Bernstein polynomials is the following:

PROPOSITION 2.8. *The graph of P on $[c,d]$ is contained in the control polygon of P on $[c,d]$.*

PROOF. In order to prove the proposition, it is enough to prove that any line L above (respectively under) all the points in the control polygon of P on $[c,d]$ is

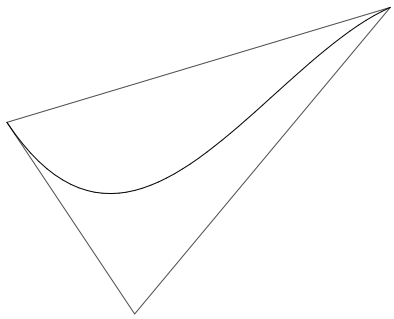

Figure 2. Graph of P on $[0,1]$ and the control polygon.

above (respectively under) the graph of P on $[c,d]$. If L is defined by $Y = aX+b$, let us express the polynomial $aX+b$ in the Bernstein basis. Since

$$1 = \left(\frac{X-c}{d-c} + \frac{d-X}{d-c}\right)^p,$$

the binomial formula gives

$$1 = \sum_{i=0}^{p} \binom{p}{i} \left(\frac{X-c}{d-c}\right)^i \left(\frac{d-X}{d-c}\right)^{p-i} = \sum_{i=0}^{p} B_{p,i}(c,d).$$

Since

$$X = \left(d\left(\frac{X-c}{d-c}\right) + c\left(\frac{d-X}{d-c}\right)\right)\left(\frac{X-c}{d-c} + \frac{d-X}{d-c}\right)^{p-1},$$

the binomial formula together with Remark 2.2 gives

$$\begin{aligned} X &= \sum_{i=0}^{p-1} \left(d\left(\frac{X-c}{d-c}\right) + c\left(\frac{d-X}{d-c}\right)\right) B_{p-1,i}(c,d) \\ &= \sum_{i=0}^{p} \left(\frac{id+(p-i)c}{p}\right) B_{p,i}(c,d). \end{aligned}$$

Thus,

$$aX+b = \sum_{i=0}^{p} \left(a\left(\frac{id+(p-i)c}{p}\right) + b\right) B_{p,i}(c,d).$$

It follows immediately that if L is above every M_i, that is, if

$$a\left(\frac{id+(p-i)c}{p}\right) + b \geq b_i$$

for every i, then L is above the graph of P on $[c,d]$, since $P = \sum_{i=0}^{p} b_i B_{p,i}(c,d)$ and the Bernstein polynomials of c,d are nonnegative on $[c,d]$. A similar argument holds for L under every M_i. □

The following algorithm computes the coefficients of P in the Bernstein bases of c,e and e,d from the coefficients of P in the Bernstein basis of c,d.

ALGORITHM 2.9 (DE CASTELJAU).

INPUT: a list $b = b_0,\ldots,b_p$ representing a polynomial P of degree $\leq p$ in the Bernstein basis of c,d, and a number $e \in \mathbb{R}$.

OUTPUT: the list $b' = b'_0,\ldots,b'_p$ representing P in the Bernstein basis of c,e and the list $b'' = b''_0,\ldots,b''_p$ representing P in the Bernstein basis of e,d.

1. Define $\alpha = (d-e)/(d-c)$ and $\beta = (e-c)/(d-c)$.
2. Initialization: $b_j^{(0)} := b_j$, $j = 0,\ldots,p$.
3. For $i = 1,\ldots,p$
   For $j = 0,\ldots,p-i$
   compute $b_j^{(i)} := \alpha b_j^{(i-1)} + \beta b_{j+1}^{(i-1)}$.
4. Output $b' = b_0^{(0)},\ldots,b_0^{(j)},\ldots,b_0^{(p)}$ and $b'' = b_0^{(p)},\ldots,b_j^{(p-j)},\ldots,b_p^{(0)}$.

De Casteljau's algorithm can be visualized by means of the triangle

$$\begin{array}{ccccccccccccc}
b_0^{(0)} & & b_1^{(0)} & & \cdots & & \cdots & & b_{p-1}^{(0)} & & b_p^{(0)} \\
& b_0^{(1)} & & \cdots & & \cdots & & \cdots & & b_{p-1}^{(1)} & \\
& & \cdots & & \cdots & & \cdots & & \cdots & & \\
& & & \cdots & & \cdots & & \cdots & & & \\
& & & & b_0^{(p-1)} & & b_1^{(p-1)} & & & & \\
& & & & & b_0^{(p)} & & & & &
\end{array}$$

where $b_j^{(i)} := \alpha b_j^{(i-1)} + \beta b_{j+1}^{(i-1)}$, $\alpha = (d-e)/(d-c)$ and $\beta = (e-c)/(d-c)$.

The coefficients of P in the Bernstein basis of c, d appear in the top side of the triangle and the coefficients of P in the Bernstein basis of c, e and e, d appear in the two other sides of the triangle.

NOTATION 2.10. We denote by $\tilde{a}$ the list obtained by reversing the list a.

PROOF OF CORRECTNESS OF DE CASTELJAU'S ALGORITHM. It is enough to prove the part of the claim concerning c, e. Indeed, by Remark 2.2, $\tilde{b}$ represents P in the Bernstein basis of d, c, and the claim is obtained by applying de Casteljau's Algorithm to $\tilde{b}$ at e. The output is $\tilde{b}''$ and $\tilde{b}'$ and the conclusion follows using again Remark 2.2.

Let $\delta_{p,i}$ be the list of length $p+1$ consisting of zeros except a 1 at the $i+1$-th place. Note that $\delta_{p,i}$ is the list of coefficients of $B_{p,i}(c, d)$ in the Bernstein basis of c, d. We will prove that the coefficients of $B_{p,i}(c, d)$ in the Bernstein basis of c, e coincide with the result of de Casteljau's Algorithm 2.9 performed with input $\delta_{p,i}$. The correctness of de Casteljau's Algorithm 2.9 for c, e then follows by linearity.

First notice that, since $\alpha = (d-e)/(d-c)$ and $\beta = (e-c)/(d-c)$,

$$\frac{d-X}{d-c} = \alpha \frac{X-c}{e-c} + \frac{e-X}{e-c},$$
$$\frac{X-c}{d-c} = \beta \frac{X-c}{e-c}.$$

Thus

$$\left(\frac{d-X}{d-c}\right)^{p-i} = \sum_{k=0}^{p-i} \binom{p-i}{k} \alpha^k \left(\frac{X-c}{e-c}\right)^k \left(\frac{e-X}{e-c}\right)^{p-i-k},$$
$$\left(\frac{X-c}{d-c}\right)^{i} = \beta^i \left(\frac{X-c}{e-c}\right)^{i}.$$

It follows that

$$B_{p,i}(c, d) = \binom{p}{i} \sum_{j=i}^{p} \binom{p-i}{j-i} \alpha^{j-i} \beta^i \left(\frac{X-c}{e-c}\right)^j \left(\frac{e-X}{e-c}\right)^{p-j}.$$

Since

$$\binom{p}{i}\binom{p-i}{j-i} = \binom{j}{i}\binom{p}{j},$$

$$B_{p,i}(c,d) = \sum_{j=i}^{p} \binom{j}{i}\alpha^{j-i}\beta^{i}\binom{p}{j}\left(\frac{X-c}{e-c}\right)^{j}\left(\frac{e-X}{e-c}\right)^{p-j}.$$

Finally,

$$B_{p,i}(c,d) = \sum_{j=i}^{p} \binom{j}{i}\alpha^{j-i}\beta^{i}B_{p,j}(c,e).$$

On the other hand, we prove by induction on p that de Casteljau's Algorithm with input $\delta_{p,i}$ outputs the list $\delta'_{p,i}$ starting with i zeros and with $(j+1)$-th element $\binom{j}{i}\alpha^{j-i}\beta^{i}$ for $j = i, \ldots, p$.

The result is clear for $p = i = 0$. If de Casteljau's Algorithm applied to $\delta_{p-1,i-1}$ outputs $\delta'_{p-1,i-1}$, the equality

$$\binom{j}{i}\alpha^{j-i}\beta^{i} = \alpha\binom{j-1}{i}\alpha^{j-i-1}\beta^{i} + \beta\binom{j-1}{i-1}\alpha^{j-i}\beta^{i-1}$$

proves by induction on j that the algorithm applied to $\delta_{p,i}$ outputs $\delta'_{p,i}$. So the coefficients of $B_{p,i}(c,d)$ in the Bernstein basis of c, e coincide with the output of the algorithm with input $\delta_{p,i}$. □

de Casteljau' Algorithm works both ways.

COROLLARY 2.11. *Let b, b' and b'' be the lists of coefficients of P in the Bernstein basis of (c,d), (c,e) and (e,d) respectively.*

(i) *De Casteljau's Algorithm applied to b with weights $\alpha = (d-e)/(d-c)$ and $\beta = (e-c)/(d-c)$ outputs b' and b''.*
(ii) *De Casteljau's Algorithm applied to b' with weights $\alpha = (e-d)/(e-c)$ and $\beta = (d-c)/(e-c)$ outputs b and $\tilde{b}''$.*
(iii) *De Casteljau's Algorithm applied to b'' with weights $\alpha = (d-c)/(d-e)$ and $\beta = (c-e)/(d-e)$ outputs $\tilde{b}'$ and b.*

De Casteljau's Algorithm gives a geometric construction of the control polygon of P on $[c,e]$ and on $[e,d]$ from the control polygon of P on $[c,d]$. The points of the new control polygons are constructed by taking iterated barycenters with weights α and β.

EXAMPLE 2.12. Continuing Example 2.7, de Casteljau's Algorithm gives

$$\begin{array}{ccccccc}
4 & & -6 & & 7 & & 10 \\
& -1 & & \frac{1}{2} & & \frac{17}{2} & \\
& & -\frac{1}{4} & & \frac{9}{2} & & \\
& & & \frac{17}{8} & & &
\end{array}$$

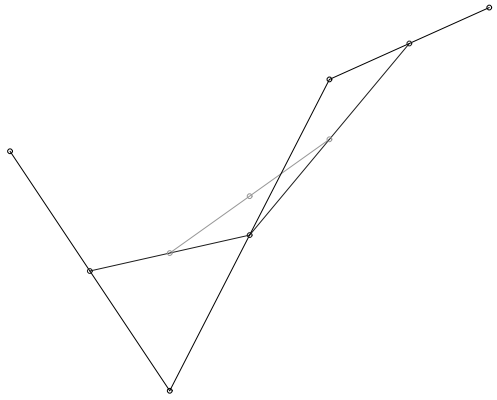

Figure 3. Control line of P on $[0, \frac{1}{2}]$.

We construct the control line of P on $[0, \frac{1}{2}]$ from the control line of P on $[0, 1]$ as explained in Figure 3.

We then draw the graph of P on $[0, 1]$ and the control line on $[0, \frac{1}{2}]$ in Figure 4.

3. Real Root Isolation in the Bernstein Basis

Let P be a polynomial of degree p in $\mathbb{R}[X]$. We are going to explain how to characterize the roots of P in $\mathbb{R}$, performing exact computations. The roots of P in $\mathbb{R}$ will be described by intervals with rational end points. Our method will be based on Descartes' law of signs (Theorem 1.1) and the properties of the Bernstein basis studied in the preceding section.

PROPOSITION 3.1. *Let b, b' and b'' be the lists of coefficients of P in the Bernstein basis of c, d; c, e; and e, d. If $c < e < d$,then*

$$\mathrm{V}(b') + \mathrm{V}(b'') \leq \mathrm{V}(b).$$

Moreover if $P(e) \neq 0$, $\mathrm{V}(b) - \mathrm{V}(b') - \mathrm{V}(b'')$ is even.

PROOF. The proof of the proposition is based on the following easy observations:

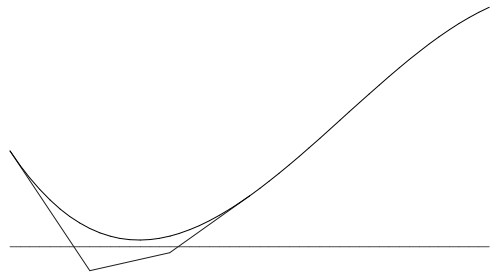

Figure 4. Graph of P on $[0, 1]$ and control line on $[0, \frac{1}{2}]$.

(i) Inserting in a list $a = a_0, \ldots, a_n$ a value x in $[a_i, a_{i+1}]$ if $a_{i+1} \geq a_i$ (resp. in $[a_{i+1}, a_i]$ if $a_{i+1} < a_i$) between a_i and a_{i+1} does not modify the number of sign variations.

(ii) Removing from a list $a = a_0, \ldots, a_n$ with first nonzero $a_k, k \geq 0$, and last nonzero $a_\ell, k \leq \ell \leq n$, an element a_i, $i \neq k, i \neq \ell$ decreases the number of sign variation by an even (possibly zero) natural number.

Indeed the lists $b^{(j)}$ defined from the values $b_j^{(i)}$ (see de Casteljau's algorithm), as follows:

$$
\begin{aligned}
b^{(0)} &= b_0^{(0)}, \ldots, \ldots, \ldots, \ldots, \ldots, b_p^{(0)} \\
b^{(1)} &= b_0^{(0)}, b_0^{(1)}, \ldots, \ldots, \ldots, \ldots, b_{p-1}^{(1)}, b_p^{(0)} \\
&\ldots \\
b^{(i)} &= b_0^{(0)}, \ldots, \ldots, b_0^{(i)}, \ldots, \ldots, b_{p-i}^{(i)}, \ldots, \ldots, b_p^{(0)} \\
&\ldots \\
b^{(p-1)} &= b_0^{(0)}, \ldots, \ldots, \ldots, \ldots, b_0^{(p-1)}, b_1^{(p-1)}, \ldots, \ldots, \ldots, \ldots, b_p^{(0)} \\
b^{(p)} &= b_0^{(0)}, \ldots, \ldots, \ldots, \ldots, \ldots, \quad b_0^{(p)} \quad , \ldots, \ldots, \ldots, \ldots, \ldots, b_p^{(0)}
\end{aligned}
$$

are successively obtained by inserting intermediate values and removing elements that are not end points, since when $c < e < d$, $b_j^{(i)}$ is between $b_j^{(i-1)}$ and $b_{j+1}^{(i-1)}$, for $i = 1, \ldots, p, j = 0, \ldots, p-i-1$. Thus $\mathrm{V}(b^{(p)}) \leq \mathrm{V}(b)$ and the difference is even. Since

$$
\begin{aligned}
b' &= b_0^{(0)}, \ldots, \ldots, \ldots, \ldots, \ldots, \quad b_0^{(p)}, \\
b'' &= b_0^{(p)} \quad , \ldots, \ldots, \ldots, \ldots, \ldots, b_p^{(0)},
\end{aligned}
$$

$\mathrm{V}(b') + \mathrm{V}(b'') \leq \mathrm{V}(b^{(p)})$, and $\mathrm{V}(b') + \mathrm{V}(b'') \leq \mathrm{V}(b)$. If $P(e) \neq 0$, it is clear that $\mathrm{V}(b^{(p)}) = \mathrm{V}(b') + \mathrm{V}(b'')$, since $b_0^{(p)} = P(e) \neq 0$. □

Example 3.2. Continuing Example 2.12, we observe, denoting by b, b' and b'', the lists of coefficients of P in the Bernstein basis of $0, 1$, $0, \frac{1}{2}$, and $\frac{1}{2}, 1$, that $\mathrm{V}(b) = 2$. This is visible on Figure 1: the control line for $[0, 1]$ cuts twice the X-axis. Similarly, $\mathrm{V}(b') = 2$. This is visible on Figure 4: the control line for $[0, \frac{1}{2}]$ also cuts twice the X-axis. Similarly, it is easy to check that $\mathrm{V}(b'') = 0$.

We cannot decide from this information whether P has two roots on $(0, \frac{1}{2})$ or no root on $(0, \frac{1}{2})$.

Suppose that $P \in \mathbb{R}[X]$ is a polynomial of degree p with all its real zeros in $(-2^\ell, 2^\ell)$ and is square-free. Consider natural numbers k and c such that $0 \leq c \leq 2^k$ and define

$$a_{c,k} = \frac{-2^{\ell+k} + c2^{\ell+1}}{2^k}.$$

It is clear that, for k big enough, the polynomial P has at most one root in $(a_{c,k}, a_{c+1,k})$ and has no other complex root in $\mathcal{C}(a_{c,k}, a_{c+1,k})_0 \cup \mathcal{C}(a_{c,k}, a_{c+1,k})_1$.

Let $b(P,c,k)$ be the list of coefficients of P in the Bernstein basis of $(a_{c,k}, a_{c+1,k})$. Note that $b(P,0,0)$, the list of coefficients of P in the Bernstein basis of $(-2^\ell, 2^\ell)$, can easily be computed from P, using Proposition 2.3.

Using Theorem 2.5, it is possible to decide, for k big enough, whether P has exactly one root in $(a_{c,k}, a_{c+1,k})$ or has no root on $(a_{c,k}, a_{c+1,k})$ by testing whether $\mathrm{V}(b(P,c,k))$ is zero or one.

Example 3.3. Continuing Example 3.2, let us study the roots of P on $(0,1)$, as a preparation to a more formal description of Algorithm 3.4 (B1 Real Root Isolation).

The Bernstein coefficients of P for $(0,1)$ are $4, -6, 7, 10$. There may be roots of P on $(0,1)$ as there are sign variations in its Bernstein coefficients.

As seen in Example 3.2, a first application of de Casteljau's Algorithm with weights $\alpha = \beta = \frac{1}{2}$ gives

$$\begin{array}{ccccccc} 4 & & -6 & & 7 & & 10 \\ & -1 & & \frac{1}{2} & & \frac{17}{2} & \\ & & -\frac{1}{4} & & \frac{9}{2} & & \\ & & & \frac{17}{8} & & & \end{array}$$

There may be roots of P on $(0, \frac{1}{2})$ as there are sign variations in the Bernstein coefficients of P which are $32, -8, -2, 17$. There are no roots of P on $(\frac{1}{2}, 1)$.

We apply once more de Casteljau's Algorithm with weights $\frac{1}{2}, \frac{1}{2}$:

$$\begin{array}{ccccccc} 4 & & -1 & & -\frac{1}{4} & & \frac{17}{8} \\ & \frac{3}{2} & & -\frac{5}{8} & & \frac{15}{16} & \\ & & \frac{7}{16} & & \frac{5}{32} & & \\ & & & \frac{19}{64} & & & \end{array}$$

There are no sign variations on the sides of the triangle so there are no roots of P on $(0, \frac{1}{4})$ and on $(\frac{1}{4}, \frac{1}{2})$.

An *isolating list for* P is a finite list L of rational points and disjoint open intervals with rational end points of $\mathbb{R}$ such that each point or interval of L contains exactly one root of P in $\mathbb{R}$ and every root of P in $\mathbb{R}$ belongs to an element of L.

Algorithm 3.4 (B1 Real Root Isolation).
Input: a square-free nonzero polynomial $P \in \mathbb{R}[X]$, an interval $(-2^\ell, 2^\ell)$ containing the roots of P in $\mathbb{R}$, the list $b(P,0,0)$ of the Bernstein coefficients of P for $(-2^\ell, 2^\ell)$.
Output: a list $L(P)$ isolating for P.

1. Initialization: Define Pos $:= \{b(P,0,0)\}$ and $L(P) := \varnothing$.
2. While Pos is nonempty:
   - Remove $b(P,c,k)$ from Pos.
   - If $\mathrm{V}(b(P,c,k)) = 1$ then insert $(a_{c,k}, a_{c+1,k})$ in $L(P)$.

- If $V(b(P,c,k)) > 1$ then
  - Compute $b(P, 2c, k+1)$ and $b(P, 2c+1, k+1)$ using de Casteljau's Algorithm with weights $(\frac{1}{2}, \frac{1}{2})$ and insert them in Pos.
  - If $P(a_{2c+1,k+1}) = 0$ then insert $a_{2c+1,k+1}$ in $L(P)$.

3. Output $L(P)$.

The hypotheses are not a real loss of generality since, given any polynomial Q, a square-free polynomial P having the same roots at Q can be computed using the gcd of Q and Q' (see for example [Basu et al. 2003]).

Moreover, setting

$$Q = c_p X^p + \ldots + c_0,$$
$$C(Q) = \sum_{0 \le i \le p} \left| \frac{c_i}{c_p} \right|,$$

the absolute value of any root of Q in $\mathbb{R}$ is smaller than $C(Q)$ [Mignotte and Ştefănescu 1999; Basu et al. 2003], so that it is easy, knowing Q, to compute ℓ such that $(-2^\ell, 2^\ell)$ contains the roots of Q in $\mathbb{R}$.

Since each subdivision yields after a scaling and a shift, new polynomials on $[0,1]$ for which the distance between the roots if doubled, by the two-circles theorem, the maximal number h of the subdivisions is bounded by

$$h \le \lceil \log_2(2/s) \rceil,$$

where s is the minimal distance between the complex roots of Q. Using classical bounds on this minimal distance between the roots of a polynomial Q with integer coefficients [Mignotte and Ştefănescu 1999; Basu et al. 2003], one can prove that

$$h \le (p-1)\log_2 \|Q\|_2 + \tfrac{1}{2}(p+2)\log_2 p \; + \; 1$$

(where $\|Q\|_2$ is the 2-norm of the coefficient vector of Q) and that the binary complexity of computing the square-free part P of Q, computing ℓ such that $(-2^\ell, 2^\ell)$ contains the roots of Q in $\mathbb{R}$, and performing Algorithm 3.4 (B1 Real Root Isolation) for P, is $O(p^6(\tau + \log_2 p)^2)$, where p is a bound on the degree of Q and τ a bound on the bitsize of the coefficients of Q [Basu et al. 2003]. The coefficients of the elements of the $b(P,c,k)$ computed in the algorithm are rational numbers of bitsize $O(p^2(\tau + \log_2 p))$ [Basu et al. 2003]. Since there are at most $2p$ values of $b(P,c,k)$ in Pos throughout the computation, and there are $p+1$ coefficients in each $b(P,c,k)$, the workspace of the algorithm is $O(p^4(\tau + \log_2 p))$.

An improved version of Algorithm 3.4 (B1 Real Root Isolation) is based on the following idea, inspired from [Rouillier and Zimmermann 2004]: since every $b(P,c,k)$ computed in the algorithm carries the whole information about P, it is not necessary to store the value of $b(P,c,k)$ at all the nodes, and the workspace of the algorithm can be improved.

It will be necessary to convert the Bernstein coefficients of P on an interval $(a_{d,m}, a_{d+1,m})$ into the Bernstein coefficients of P on an interval $(a_{c,k}, a_{c+1,k})$.

ALGORITHM 3.5 (CONVERT).
INPUT: (c,k), (d,m), and $b(P,d,m)$, Bernstein coefficients of P on $(a_{d,m}, a_{d+1,m})$.
OUTPUT: the Bernstein coefficients $b(P,c,k)$ of P on $(a_{c,k}, a_{c+1,k})$.

1. Initialize $b := b(P,d,m)$.
2. Let $c = c_0 + \cdots + c_{n-1}2^{n-1} + c_n 2^n + \cdots + c_{k-1}2^{k-1}$ and $d = d_0 + \cdots + d_{n-1}2^{n-1} + d_n 2^n + \cdots + d_{m-1}2^{m-1}$, with $c_i \in \{0,1\}$, $c_n \neq d_n$, $c_i = d_i$ for every $i < n$.
3. For i in $m-1, \ldots, n$:
   - If $d_i = 0$ then apply de Casteljau's Algorithm to b, weights $(-1,2)$ and output b', b''. Update $b := b'$.
   - If $d_i = 1$ then apply de Casteljau's Algorithm to b with weights $(2,-1)$ and output b', b''. Update $b := b''$.
4. For i in $n, \ldots, k-1$:
   - If $c_i = 0$ then apply de Casteljau's Algorithm to b with weights $(\frac{1}{2}, \frac{1}{2})$ and output b', b''. Update $b := b'$.
   - If $c_i = 1$ then apply de Casteljau's Algorithm to b with weights $(\frac{1}{2}, \frac{1}{2})$ and output b', b''. Update $b := b''$.
5. Output b.

The correctness of this algorithm clearly follows from that of de Casteljau's Algorithm.

It is now easy to describe the improved real root isolation method.

ALGORITHM 3.6 (B2 REAL ROOT ISOLATION).
INPUT: a square-free nonzero polynomial $P \in \mathbb{R}[X]$, an interval $(-2^\ell, 2^\ell)$ containing the roots of P in $\mathbb{R}$, the list $b(P,0,0)$ of the Bernstein coefficients of P for $(-2^\ell, 2^\ell)$.
OUTPUT: a list $L(P)$ isolating for P.

1. Initialization: Set Pos $:= \{(0,0)\}$, $L(P) := \varnothing$, $d := 0$, $m := 0$.
2. While Pos is nonempty:
   - Remove the first element (c,k) of Pos.
   - Compute $b(P,c,k)$ from $b(P,d,m)$ using Algorithm 3.5 (Convert).
   - If $\mathrm{V}(b(P,c,k)) = 1$ then insert $(a_{c,k}, a_{c+1,k})$ in $L(P)$.
   - If $\mathrm{V}(b(P,c,k)) > 1$ then:
     - Insert $(2c, k+1), (2c+1, k+1)$ at the beginning of Pos.
     - If $P(a_{2c+1,k+1}) = 0$ then insert $a_{2c+1,k+1}$ in $L(P)$.
   - Update $d := c$, $m := k$.
3. Output $L(P)$.

The next lemma is the key result for analyzing the complexity of this algorithm. (Note that the set of (c,k) such that $b(P,c,k)$ is computed in Algorithm 3.4 (B1 Real Root Isolation) is naturally equipped with a binary tree structure, denoted by T: (d,m) is a child of (c,k) if $d=2c$ or $d=2c+1$, and $m=k+1$.)

LEMMA 3.7. *In Algorithm* 3.6 (*B2 Real Root Isolation*), *the leaves of* T *are visited once*, *the nodes of* T *with one child are visited twice and the nodes of* T *with two children are visited three times*

PROOF. Easy by induction on the depth of T, noting that if (c,k) and (d,m) are nodes of T, $a_{c,k} < a_{d,m}$ if and only if every visit of (c,k) takes place before (d,m) is visited. □

The binary complexity of Algorithm 3.6 (B2 Real Root Isolation) for P, is $O(p^6(\tau+\log_2 p)^2)$, similarly to Algorithm 3.4 (B1 Real Root Isolation), since every node in the tree T is visited at most three times in by Lemma 3.7. However Algorithm 3.6 (B2 Real Root Isolation) uses only $O(p^3(\tau+\log_2 p))$ workspace, since only one vector of Bernstein coefficients is stored throughout the computation, rather than $O(p^4(\tau+\log_2 p))$ workspace in Algorithm 3.4 (B1 Real Root Isolation).

We can also perform the computation using interval arithmetic. The basic idea of interval arithmetic is that real numbers are represented by intervals with rational bounds encoded as floating point numbers with a fixed precision. The advantages of interval arithmetic is that it is much quicker than exact arithmetic, and it allows us to compute with polynomials known approximately.

The interval arithmetic we consider is indexed by two natural numbers u,n, defining the precision. The u,n-intervals are of the form $[\frac{i}{2^I},\frac{j}{2^J}]$, with i and j being integers between -2^u and 2^u, $i\le j$, and I and J being integers between -2^n and 2^n. A consistent interval arithmetic is compatible with the arithmetic operations: if α and β are two real numbers represented respectively by two intervals A and B, the result $A\odot B$ of any arithmetic operation $\odot$ will contain the real number $\alpha\odot\beta$. In the next paragraph, we assume working with a multi-precision interval arithmetic such as in [Revol and Rouillier 2002] (where u can be arbitrary fixed by the user). In order to perform Algorithm 3.6 (B2 Real Root Isolation) in this arithmetic, we only need to double an interval, subtract two intervals and compute the average of two intervals.

The sign of an interval $[a,b]$, where $a\le b$, is defined as follows:

$$\operatorname{sign}[a,b]=\begin{cases}0 & \text{if } a=b=0,\\ 1 & \text{if } a>0,\\ -1 & \text{if } b<0,\\ ? & \text{if } a\le 0\le b,\ a\neq 0,\ b\neq 0.\end{cases}$$

The number of sign variations in a list $A=[a_0,b_0],\dots,[a_p,b_p]$ of intervals with rational end points is defined as follows:

- If $\operatorname{sign}[a_i, b_i] \neq ?$ for all $i = 0, \ldots, p$, set
$$\mathrm{V}([a_0, b_0], \ldots, [a_p, b_p]) = \mathrm{V}(\operatorname{sign}[a_0, b_0], \ldots, \operatorname{sign}[a_p, b_p]).$$
- If, for every $i = 1, \ldots, p$ such that $\operatorname{sign}[a_i, b_i] = ?$, both $\operatorname{sign}[a_{i-1}, b_{i-1}]$ and $\operatorname{sign}[a_{i+1}, b_{i+1}]$ are defined and moreover $\operatorname{sign}[a_{i-1}, b_{i-1}] \operatorname{sign}[a_{i+1}, b_{i+1}] < 0$, then set
$$\mathrm{V}(A) = \mathrm{V}(B),$$
where B is obtained by removing from A all the $[a_i, b_i]$ such that $\operatorname{sign}[a_i, b_i] = ?$.
- Otherwise set $\mathrm{V}(A) = ?$.

Example 3.8. If $A = [1, 2], [-2, -1]$, $\mathrm{V}(A) = 1$. If $A = [1, 2], [-1, 1]$, $\mathrm{V}(A) = ?$. If $A = [1, 2], [-1, 1], [-2, -1]$, $\mathrm{V}(A) = 1$.

Algorithm 3.9 (B3 Real Root Isolation).
Input: an integer ℓ, a precision u, n, a list $\bar{b}(0, 0)$ with $p+1$ elements which are u, n-intervals, and whose first and last element do not contain 0.
Output: a list L and a list N of intervals such that for every polynomial P such that $(-2^\ell, 2^\ell)$ contains the roots of P in $\mathbb{R}$, and whose Bernstein coefficients for $(-2^\ell, 2^\ell)$ belong to $\bar{b}(0, 0)$, there exists one and only one root of P in each interval of L and all the other roots of P in $(-2^\ell, 2^\ell)$, belong to an interval of N.

1. Initialization: Compute $\mathrm{V}(\bar{b}(0, 0))$, using Proposition 2.3 and u, n-arithmetic define $\mathrm{Pos} := \{(0, 0)\}$, $L := \varnothing$, $N := \varnothing$, $d := 0$, $m := 0$.
2. While Pos is nonempty:
   - Remove the first element (c, k) of Pos.
   - Compute $\bar{b}(c, k)$ from $\bar{b}(d, m)$ by Algorithm 3.5 (Convert), using u, n-arithmetic.
   - If $\mathrm{V}(\bar{b}(c, k)) = 1$ then insert $(a_{c,k}, a_{c+1,k})$ in L.
   - If $\mathrm{V}(\bar{b}(c, k)) > 1$ then insert $(2c, k+1), (2c+1, k+1)$ at the beginning of Pos.
   - If $\mathrm{V}(\bar{b}(c, k)) = ?$ then insert $(a_{c,k}, a_{c+1,k})$ in N.
   - Update $d := c$, $m := k$.
3. Output L, N.

Interval arithmetic can be used as well when the polynomial P is known exactly. In this case we can compute the square-free part of P and it is easy to design a variant of Algorithm 3.9 and output a list of isolating intervals by augmenting precision, examining again the intervals where no decision has been taken yet.

Algorithm 3.10 (B4 Real Root Isolation).
Input: a square-free $P \in \mathbb{R}[X]$, and the list $b(P, 0, 0)$ of Bernstein coefficients of P for $(-2^\ell, 2^\ell)$, where $(-2^\ell, 2^\ell)$ contains the roots of P in $\mathbb{R}$.
Output: a list $L(P)$ isolating for P.

1. Initialization: u such that the elements of $b(P,0,0)$ belong to $(-2^u, 2^u)$, $n := 1$. Compute $\mathrm{V}(b(P,0,0))$, define Pos $:= \{(0,0)\}$, $L(P) := \varnothing$, $N(P) := \varnothing$, $d := 0$, $m := 0$.
2. While Pos is nonempty:
   - Remove the first element (c,k) of Pos.
   - Compute $b(P,c,k)$ from $b(P,d,m)$ by Algorithm 3.5 (Convert) using u, n-arithmetic.
   - If $\mathrm{V}(b(P,c,k)) = 1$ then insert $(a_{c,k}, a_{c+1,k})$ in $L(P)$.
   - If $\mathrm{V}(b(P,c,k)) > 1$ then:
     - Insert $(2c, k+1), (2c+1, k+1)$ at the beginning of Pos.
     - If $P(a_{2c+1,k+1}) = 0$ then insert $a_{2c+1,k+1}$ to $L(P)$.
   - If $\mathrm{V}(b(P,c,k)) = ?$ then insert $(a_{c,k}, a_{c+1,k})$ in $N(P)$.
   - Update $d := c$, $m := k$.
3. If $N(P) \neq \varnothing$ then update $n := n+1$, Pos $= N(P)$ and go to step 2.
4. Output $L(P)$.

4. Real Root Isolation in the Monomial Basis

The preceding methods for real root isolation are adapted to polynomials given in the Bernstein basis. However in many cases, the polynomials are given in the monomial basis, and the conversion to the Bernstein basis is computationally expensive. It is thus natural to look for real root isolation algorithms adapted to the case where the polynomials are expressed in the monomial basis.

Such algorithms for real root isolation in the monomial basis are very classical, and have been studied extensively, starting from [Uspensky 1948] (see [Rouillier and Zimmermann 2004] for a bibliography). We prove here that their correctness is an immediate consequence of the correctness of the corresponding algorithms in the Bernstein basis.

Rather than looking at the Bernstein coefficients of the same polynomial P on varying intervals, we are going to consider different polynomials closely related to P on each interval. We need some notation. Suppose as before that $P \in \mathbb{R}[X]$ is a polynomial of degree p with all its real zeros in $(-2^\ell, 2^\ell)$ and is square-free, consider natural numbers k and c such that $0 \leq c \leq 2^k$ and define

$$a_{c,k} = \frac{-2^{\ell+k} + c2^{\ell+1}}{2^k}.$$

We define

$$P_{c,k} := \mathrm{C}_{2^{\ell+1-k}}(\mathrm{T}_{-a_{c,k}}(P)).$$

$P_{c,k}$ is simply the result of the transformation operated on P when the segment $(a_{c,k}, a_{c+1,k})$ is sent to $(0,1)$ by a translation followed by a contraction.

The following lemma is the key result making the connection between the real root isolation in the monomial basis and the Bernstein basis.

LEMMA 4.1. *Let* $Q_{c,k} := \mathrm{T}_{-1}\left(\mathrm{Rec}_p(P_{c,k})\right)$. *Then* $\mathrm{V}(Q_{c,k}) = \mathrm{V}(b(P,c,k))$.

PROOF. Immediate by Proposition 2.3. □

The four algorithms of the preceding section have analogous versions in the monomial basis [Rouillier and Zimmermann 2004]. We describe only the algorithms corresponding to the conversion from one interval to another and the improved root isolation algorithm.

ALGORITHM 4.2 (M1 CHANGE INTERVAL).
INPUT: (c,k), (d,m) and the polynomial $P_{d,m}$.
OUTPUT: the polynomial $P_{c,k}$.

1. Let $c = c_0 + \cdots + c_{n-1}2^{n-1} + c_n 2^n + \cdots + c_{k-1}2^{k-1}$ and $d = d_0 + \cdots + d_{n-1}2^{n-1} + d_n 2^n + \cdots + d_{m-1}2^{m-1}$, with $c_i \in \{0,1\}$, $c_n \neq d_n$, $c_i = d_i$ for every $i < n$. and $R := P_{d,m}$.
2. For i from $m-1$ to n:
   - If $d_i = 0$, then $R := \mathrm{C}_2(R)$.
   - If $d_i = 1$, then $R := \mathrm{C}_2(\mathrm{T}_{-1}(R))$.
3. For i from n to $k-1$:
   - If $c_i = 0$, then $R := \mathrm{C}_{1/2}(R)$.
   - If $c_i = 1$, then $R := \mathrm{C}_{1/2}(\mathrm{T}_{-1/2}(R))$.
4. Output R.

The correctness of the algorithm follows clearly from the definition of $P_{c,k}$.

It is now easy to describe the improved real root isolation method in the monomial basis.

ALGORITHM 4.3 (M2 REAL ROOT ISOLATION).
INPUT: a square-free nonzero polynomial $P \in \mathbb{R}[X]$, and an interval $(-2^\ell, 2^\ell)$ containing the roots of P in $\mathbb{R}$.
OUTPUT: a list $L(P)$ isolating for P.

1. Initialization: Define Pos $:= \{(0,0)\}$, $L(P) := \varnothing$, $d := 0$, $m := 0$.
2. While Pos is nonempty:
   - Remove the first element (c,k) of Pos.
   - Compute $P_{c,k}$ from $P_{d,m}$ using Algorithm 3.5 (Change Interval). Take $Q_{c,k} := \mathrm{T}_{-1}\left(\mathrm{Rec}_p(P_{c,k})\right)$.
   - If $\mathrm{V}(Q_{c,k}) = 1$, then insert $(a_{c,k}, a_{c+1,k})$ in $L(P)$.
   - If $\mathrm{V}(Q_{c,k}) > 1$ then:
     - Insert $(2c, k+1), (2c+1, k+1)$ at the beginning of Pos.
     - If $P(a_{2c+1,k+1}) = 0$ then insert $a_{2c+1,k+1}$ in $L(P)$.
   - Update $d := c$, $m := k$.
3. Output $L(P)$.

The correctness of Algorithm 4.3 follows from the correctness of Algorithm 3.6 and Lemma 4.1.

The complexity analysis of the real root isolation method in the monomial basis and in the Bernstein basis are quite similar.

5. Efficiency of the Methods

The experimental behavior of Algorithms 4.3 [M2 Real Root Isolation in monomial basis], more precisely of its interval arithmetic variants is excellent, and real root isolation can be performed by this method for polynomials of degree several thousands and with coefficients of bit size several hundred (see [Rouillier and Zimmermann 2004] for details on these experimental results).

The experimental behavior in the case of the Bernstein basis has not been studied fully yet, but the first experiments indicate that the algorithms presented here are as efficient as the corresponding ones in the monomial basis, if the polynomial is initially given in the Bernstein basis.

Implementations of these algorithms are available in the libraries RS (see http://fgbrs.lip6.fr/salsa/Software/) and SYNAPS (see http://www-sop.inria.fr/galaad/software/synaps/).

References

[Basu et al. 2003] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in real algebraic geometry*, Algorithms and Computation in Mathematics **10**, Springer, Berlin, 2003.

[Farin 1990] G. Farin, *Curves and surfaces for computer aided geometric design*, Academic Press, Boston, 1990.

[Mehlhorn 2001] K. Mehlhorn, "A remark on the Sign Variation Method for real root isolation", preprint, 2001. Available at http://www.mpi-sb.mpg.de/~mehlhorn/ftp/Descartes.ps.

[Mignotte and Ştefănescu 1999] M. Mignotte and D. Ştefănescu, *Polynomials*, Series in Disc. Math. and Theoret. Computer Science, Springer-Verlag, Singapore, 1999.

[Ostrowski 1950] A. M. Ostrowski, "Note on Vincent's theorem", *Ann. of Math.* (2) **52** (1950), 702–707. Reprinted as pp. 728–733 in *Collected Mathematical Papers*, vol. 1, Birkhäuser, Basel, 1983.

[Revol and Rouillier 2002] N. Revol and F. Rouillier, "Motivations for an arbitrary precision interval arithmetic and the MPFI library", pp. 155–161 in *Proceedings of the Workshop on Validated Computing* (Toronto, 2002), 2002. Revised version to appear in *Reliable Computing*.

[Rouillier and Zimmermann 2004] F. Rouillier and P. Zimmermann, "Efficient isolation of polynomial's real roots", *J. Comput. Appl. Math.* **162**:1 (2004), 33–50.

[Struik 1969] D. J. Struik (editor), *A source book in mathematics,* 1200–1800, edited by D. J. Struik, Harvard University Press, Cambridge, MA, 1969.

[Uspensky 1948] J. V. Uspensky, *Theory of equations*, MacGraw-Hill, New York, 1948.

Bernard Mourrain
GALAAD, INRIA
BP 93
06902 Sophia Antipolis
France
mourrain@sophia.inria.fr

Fabrice Rouillier
SALSA, INRIA/LIP6
8, rue du Capitaine Scott
75015 Paris
France
Fabrice.Rouillier@inria.fr

Marie-Françoise Roy
IRMAR, CNRS
Université de Rennes I
Campus de Beaulieu
35042 Rennes cedex
France
marie-francoise.roy@univ-rennes1.fr