Determining the Real Roots of Real Polynomials

Kurt Mehlhorn
Michael Sagraloff

max planck institut
informatik

March 2015

Overview

- The problem: Isolating Roots of Real Polynomials
- How I got interested in the problem: Computational geometry for curves and surfaces.
- The state of the art.
- The Descartes method.
- The new algorithm.
- Summary.

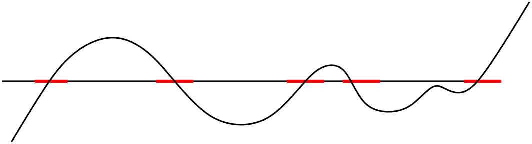
- M. Sagraloff and KM: Computing Real Roots of Real Polynomials – An Efficient Method Based on Descartes' Rule of Signs and Newton Iteration, J. Symbolic Computation, 2015
- KM, M. Sagraloff and P. Wang: From Approximate Factorization to Root Isolation with Application to Cylindrical Algebraic Decomposition, J. Symbolic Computation, 2015.

Warning: Some of my statements will be incorrect for the sake of simplicity of the presentation.

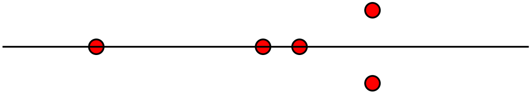
Slides and papers are available on my homepage.

The Real Root Isolation Problem

Given a polynomial with real coefficients (a real polynomial) determine its real roots, i.e., compute isolating intervals for its real roots. An interval is isolating if it contains exactly one root.



A polynomial with 5 real roots; isolating intervals are shown in red.



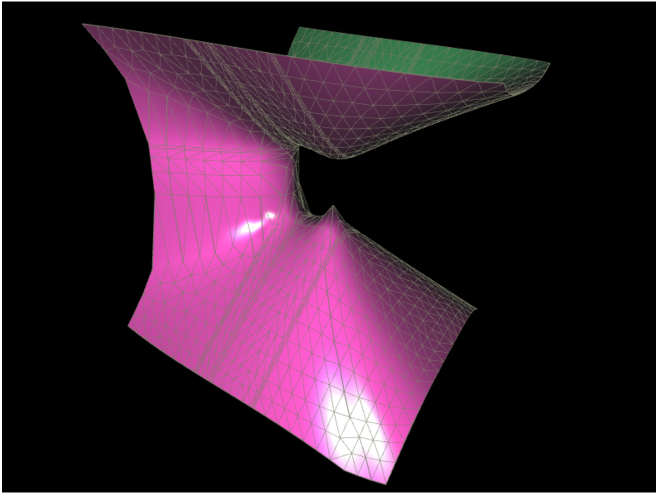
A polynomial of degree n has n complex roots. For a real polynomial, the complex roots come in pairs.

Motivation: Nonlinear Computational Geometry

an arrangement of four curves of degree 6

picture, courtesy of Michael Kerber

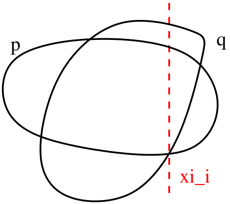
Motivation: $-z^4 + z^3 + y^4 + y^2 - x^3 + x^2 = 0$



Courtesy of Eric Berberich, Pawel Emiliyanenko, Michael Kerber, and Michael Sagraloff: EXACUS

How to intersect the curves $p(x,y)=0$ and $q(x,y)=0$?

- eliminate y and obtain a polynomial $R(x)$ of degree $d = \deg(p) \cdot \deg(q)$ compute resultant
- compute the real zeros $\xi_1, \xi_2, \ldots$ of $R(x)$
- analyse the situation at $x = \xi_i$:
  - this amounts to computing the real zeros of $p(\xi_i, y)$ and $q(\xi_i, y)$



Key task: compute the real roots of a univariate polynomial with real coefficients

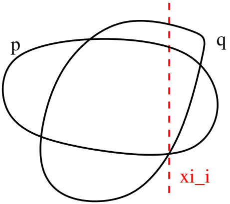
We want an algorithm that

- works for polynomials with real coefficients,
- is exact, and
- handles "easy cases" fast.

A Glimpse at the Arrangement Computation

How to intersect the curves $p(x, y) = 0$ and $q(x, y) = 0$?

- eliminate y and obtain a polynomial $R(x)$ of degree $d = \deg(p) \cdot \deg(q)$ — compute resultant
- compute the real zeros $\xi_1, \xi_2, \ldots$ of $R(x)$
- analyse the situation at $x = \xi_i$:
  - this amounts to computing the real zeros of $p(\xi_i, y)$ and $q(\xi_i, y)$



Key task: compute the real roots of a univariate polynomial with real coefficients

We want an algorithm that

- works for polynomials with real coefficients,
- is exact, and
- handles “easy cases” fast.

The State of the Art

All complex roots

- Numerical methods: usually fast, no global convergence proof.
- Splitting circle method: Schönhage (82), Pan (02), computes approximate factorization; almost optimal; not implemented yet.
- Root isolation with same complexity, M/Sagraloff/Wang (03/05), Pan/Tsigaridas(03)

Real roots or real roots restricted to an interval

- Subdivision methods: Descartes' Method, Sturm Sequences, Continued Fractions,
- Simple, however, worst-case running time much worse than Pan.
- Excellent implementations, e.g., F. Rouillier's algorithm RS.
- RS is the solver in MAPLE
- Today: A variant of Descartes, simple and competitive with Pan in the worst-case. First experiments are promising (Kobel, Rouillier, Sagraloff).

A Hard Example: Mignotte Polynomials

Mignotte polynomial, $p(x) = x^n - 2(ax-1)^2$, $a \geq 2$ integral

- Three real roots.
- Let $\tau = \log|a|$. Two of the roots have distance $\approx 2^{-\Omega(\tau n)}$.
  - polynomial is positive at $x = 1/a$.
  - $p(x)$ is negative for $x = 1/a \pm h$, where $h = (1/a)^{(n+2)/2}$.

$$p(1/a \pm h) = (1/a \pm h)^n - 2a^2h^2 < 1/a^n - a^2h^2 < 0.$$

- $a \approx 2^{20}$, $n = 20$, distance 2^{-200}.

Remarks

- Mignotte polynomials have worst-case root separation among polynomials with integer coefficients.
- Let sep(p) be the smallest distance between two roots. Then $\text{sep}(p) \geq 2^{-n\tau}$.

A Hard Example: Mignotte Polynomials

Mignotte polynomial, $p(x) = x^n - 2(ax-1)^2$, $a \geq 2$ integral

- Three real roots.
- Let $\tau = \log|a|$. Two of the roots have distance $\approx 2^{-\Omega(\tau n)}$.
  - polynomial is positive at $x = 1/a$.
  - $p(x)$ is negative for $x = 1/a \pm h$, where $h = (1/a)^{(n+2)/2}$.

$$p(1/a \pm h) = (1/a \pm h)^n - 2a^2h^2 < 1/a^n - a^2h^2 < 0.$$

- $a \approx 2^{20}$, $n = 20$, distance 2^{-200}.

Remarks

- Mignotte polynomials have worst-case root separation among polynomials with integer coefficients.
- Let sep(p) be the smallest distance between two roots. Then sep(p) $\geq 2^{-n\tau}$.

The Descartes Method for Real Root Isolation

- Descartes proved the underlying theorem: Descartes' rule of sign.
- Algorithm is due to Collins/Akritas (76) and Lane/Riesenfeld(81).

Number of Sign Changes

$p = \sum_{0 \le i \le n} p_i x^i$ with $p_n \neq 0 \neq p_0$. Let $v(p)$ be the number of sign changes in coefficient sequence, e.g., $v(-3, 0, -2, 2, -1) = 2$.

Theorem (Descartes' Rule of Sign)

- Number of real zeros of p in $(0, \infty)$ is at most $v(p)$.
- Both numbers have the same parity.
- $v_I(p)$ for interval $I = (a, b)$; consider $(b-x)^n \cdot p(\frac{x-a}{b-x})$.

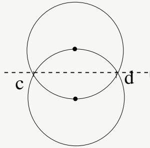
Corollary

$v_I(p) = 0 \quad \Rightarrow \quad$ no root

$v_I(p) = 1 \quad \Rightarrow \quad$ exactly one root

$w(I) \le \text{sep}(p)/4 \implies v_I(p) \le 1.$

Partial Converse (Landau, Obreshkoff)

If 

contains at most one root, then $v_I(p) \le 1$.

Number of Sign Changes

$p = \sum_{0 \le i \le n} p_i x^i$ with $p_n \neq 0 \neq p_0$. Let $v(p)$ be the number of sign changes in coefficient sequence, e.g., $v(-3, 0, -2, 2, -1) = 2$.

Theorem (Descartes' Rule of Sign)

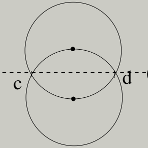
- Number of real zeros of p in $(0, \infty)$ is at most $v(p)$.
- Both numbers have the same parity.
- $v_I(p)$ for interval $I = (a, b)$; consider $(b - x)^n \cdot p(\frac{x-a}{b-x})$.

Corollary

$v_I(p) = 0$	$\Rightarrow$	no root
$v_I(p) = 1$	$\Rightarrow$	exactly one root

$w(I) \le \text{sep}(p)/4 \implies v_I(p) \le 1$.

Partial Converse (Landau, Obreshkoff)

If  contains at most one root, then $v_I(p) \le 1$.

Descartes Method: isolate roots of $p(x)$ in $I = (c, d)$

- Compute $v_I(p)$;
- If $v_I = 0$ return;
- If $v_I = 1$, return and report (c, d) as an isolating interval
- Let $m = (c + d)/2$. (more generally $m = \alpha c + (1 - \alpha)d$)
  - If $p(m) = 0$ report $[m, m]$ as an isolating interval.
  - Recurse on both sub-intervals.

To isolate all real roots

Start with $(-M, +M)$, where $M = \max_i 2 \cdot |p_i|/|p_n|$.

Subadditivity of Sign Variations

$$v_{(c,m)} + v_{(m,d)} \leq v_{(c,d)}.$$

Descartes Method: isolate roots of $p(x)$ in $I = (c, d)$

- Compute $v_I(p)$;
- If $v_I = 0$ return;
- If $v_I = 1$, return and report (c, d) as an isolating interval
- Let $m = (c + d)/2$. (more generally $m = \alpha c + (1 - \alpha)d$)
  - If $p(m) = 0$ report $[m, m]$ as an isolating interval.
  - Recurse on both sub-intervals.

To isolate all real roots

Start with $(-M, +M)$, where $M = \max_i 2 \cdot |p_i|/|p_n|$.

Subadditivity of Sign Variations

$$v_{(c,m)} + v_{(m,d)} \leq v_{(c,d)}.$$

Analysis of Descartes Method

- Asssume nonzero coeffs are in $[1, 2^\tau]$ in absolute value.
- Start with $[-M, +M]$, where $M = 2 \cdot 2^\tau$.
- Alg stops at intervals of length $\approx \text{sep}(p)$, maybe earlier.
- Depth of the recursion tree is $\leq \tau + \log 1/\text{sep}(p)$.
- Width of the recursion tree is $\leq n$, because of subadditivity.
- Number of nodes in the tree is $n(\tau + \log 1/\text{sep}(p))$.
- n arithmetic ops/node for computing v_I and for evaluating $p(m)$.
- Assume integer coeffs:
  - $\log 1/\text{sep}(p) = O(n\tau)$. Thus depth $= O(n\tau)$ and # nodes $= O(n^2\tau)$.
  - Numbers grow by n bits in every node of the recursion tree and hence grow to $\tau + n^2\tau$ bits.
  - Number of bit operations: $O(n^2\tau \cdot n \cdot n^2\tau) = O(n^5\tau^2)$.
- We are now in the year 2005.

Two Questions

Can we handle real coefficients, e.g., $\sqrt{2}$, π, $\ln 2$?

We assume them to be given by oracles that can be asked for arbitrary good approximations.

Coefficients are potentially infinite bitstreams.

How can we handle polynomials with bitstream coefficients?

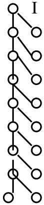
Can we improve complexity so that it matches Pan's?

For a polynomial with integer coefficients bounded by 2^τ in absolute values

- Descartes method uses $O(n^5\tau^2)$ bit operations, but
- Pan's alg uses only $\widetilde{O}(n^2\tau)$ bit operations to isolate all roots.

Analysis of Descartes Method Revisited

- Depth of recursion tree is $\leq \tau + \log 1/\text{sep}(p)$.
- Width of recursion tree is $\leq n$, because of subadditivity.
- Number of nodes in tree is $n^2\tau$.
- n arithmetic ops/node for computing v_I and for evaluating $p(m)$.
- Assume integer coeffs:
  - $\log 1/\text{sep}(p) = O(n\tau)$, depth $= n\tau$, # nodes $= O(n^2\tau)$
  - Numbers grow by n bits in every node of the recursion tree.
  - So numbers grow to $\tau + n^2\tau$ bits.
  - Number of bit operations: $O(n^2\tau \cdot n \cdot n^2\tau) = O(n^5\tau^2)$.
- Potential for improvement:
  - Why precision $n^2\tau$ if $\log 1/\text{sep}(p) = O(n\tau)$?
  - Tree has only n nodes where both children have non-zero sign variations. Can we traverse long chains faster? (ideally, with a logarithmic number of iterations)
  - # of bit operations would reduce to $\tilde{O}(n \cdot n \cdot n\tau)$.



Algorithm ANewDsc

Approximate Newton Descartes

A New Descartes

Bitstream Coefficients

Approximate Coefficients

We represent coefficients by intervals; these interval can be refined as needed.

Arithmetic becomes interval arithmetic, i.e.
$[a,b]+[c,d]=[a+c,b+d]$ and
$[a,b]\cdot[c,d]=[\min(ac,ad,bc,bd),\ldots]$.

Polynomials become interval polynomials.

Descartes becomes Interval-Descartes (Johnson-Krandick), but what is the sign of an interval and how does one compute sign changes?

Sign Variations in Sequences of Intervals

Set of potential sign variations in a sequence of intervals

$$
\begin{aligned}
v(([2,3],[-1,1])) &= \{0,1\}\,,\\
v(([2,3],[-1,1],[2,3])) &= \{0,2\}\,,\\
v(([2,3],[-1,1],[-2,-1])) &= \{1\}\,.
\end{aligned}
$$

We now have $\tilde{v}_I$ instead of v_I.

Capabilities needed for the Descartes method

1. For all nodes in the recursion tree: Does the Descartes test yield 0, 1, or at least 2 sign variations?
2. For internal nodes of the recursion tree: Does the polynomial vanish at the split point?

With an interval polynomial, making either decision may be impossible and hence the approach seems doomed.

A First Idea: Eigenwillig/Kettner/Krandick/KM/Schmitt/Wolpert (05)

Careful Choice of Subdivision Point

Choose $2n$ equidistant points in middle part of $[c, d]$; call them M

"Evaluate" $p(x)$ on all of them and choose $m \in M$ such that $|p(m)| \geq \max_{x \in M} |p(x)|/2$.

Then $p(m) \neq 0$ for all subdivision points.

Details:

- Precision required for the computation is determined by $|p(m)|$. Double precision until $p(m)$ can be computed up to factor two.
- Evaluation of $2n$ equidistant points does not cost more than evaluation on a single point (Kobel/Sagraloff).
- We can estimate $p(m)$ from below because a polynomial can be small only close to one of its roots (Smith bound).

Interval Descartes Method: Isolate roots of $p(x)$ in $I = (c, d)$

- Compute $\tilde{v}_I(p)$;
- If $\tilde{v}_I = \{0\}$ return;
- If $\tilde{v}_I = \{1\}$, return and report (c, d) as an isolating interval
- Choose a good midpoint, namely, $m \in M$ with

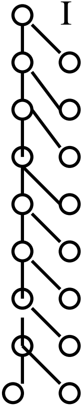
  $$|p(m)| \geq \max_{x \in M} |p(x)|/2,$$

  where M is a nice set of $2n$ points in the middle part of I.
- Recurse on both sub-intervals.

Amazing fact (Eigenwillig et al)

Recursion depth is essentially the same as for exact Descartes, more precisely, $+O(1)$. Proof hinges crucially on the fact that we choose midpoints where the polynomial is large.

Second Idea: Sagraloff (2014)



The shape of the recursion tree

Tree may have depth $O(n\tau)$, but only $O(n)$ nodes where both children are subdivided further.

There must be long chains, say length L, where we split off an interval with no sign change, i.e., we have a tiny interval of length $2^{-L}|I|$ containing all roots in I.

Bisection versus Newton

Bisection converges linearly, i.e., one additional correct bit per iteration.

Newton converges quadratically, i.e., number of correct bits doubles per iteration.

Can we use Newton-iteration to traverse chains of length L in $\log L$ steps?

Clusters of roots (a bit of wishful thinking)

I 

The red interval contains k roots. If red interval is small enough, we may treat the k roots as a k-fold root.

Newton iteration for a k-fold root: $$x_{n+1} = x_n - k\frac{p(x_n)}{p'(x_n)}.$$

Problem: We do not know k.

In interval (c, d), we tentatively use $x_n = c$ and $x_n = d$, equate the two values for x_{n+1} and solve for k, i.e.,

$$c - \hat{k}\frac{p(c)}{p'(c)} \stackrel{!}{=} d - \hat{k}\frac{p(d)}{p'(d)}.$$

Let $\xi = c - \hat{k}p(c)/p'(c)$ and consider an interval $[c', d']$ around ξ.

If $[c, c']$ and $[d', d]$ have zero sign-changes, continue with $[c', d']$ else use bisection, i.e., continue with $[c, m]$ and $[m, d]$.

Clusters of roots: The situation

Let $\xi = c - \hat{k}p(c)/p'(c)$ and consider an interval $[c', d']$ around ξ.

If $[c, c']$ and $[d', d]$ have zero sign-changes, continue with $I' = [c', d']$

else use bisection, i.e., continue with $[c, m]$ and $[m, d]$

Clusters of roots: The situation

Let $\xi = c - \hat{k}p(c)/p'(c)$ and consider an interval $[c', d']$ around ξ.

If $[c, c']$ and $[d', d]$ have zero sign-changes, continue with $I' = [c', d']$ and level of agressiveness N_I^2 (Success).

else use bisection, i.e., continue with $[c, m]$ and $[m, d]$ and level of agressiveness $\max(4, \sqrt{N_I})$.

Quadratic Interval Refinement (Abott)

Each interval I has a level of agressiveness N_I.

We choose $[c', d']$ such that $d' - c' = w(I)/N_I$.

Clusters of roots: The situation

Let $\xi = c - \hat{k}p(c)/p'(c)$ and consider an interval $[c', d']$ around ξ.

If $[c, c']$ and $[d', d]$ have zero sign-changes, continue with $I' = [c', d']$ and level of agressiveness N_I^2 (Success).

else use bisection, i.e., continue with $[c, m]$ and $[m, d]$ and level of agressiveness $\max(4, \sqrt{N_I})$.

Success Lemma

If red interval is tiny with respect to black interval, circumcircle of I' contains k roots, enlarged circumcircle of I contains no other roots, and agressiveness is not too high, we have success.

Consequence for chain traversal.

Summary

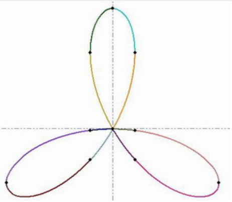
Algorithm ANewDsc for Real Root Isolation

- has a worst-case complexity similar to Pan's alg and a much better observed complexity,
- can be asked to isolate roots in an interval,
- is simple enough to be implemented (Kobel/Sagraloff/Rouillier)
  - for integer polynomials: for simple cases, same speed as Rouillier's RS, for difficult cases, much faster.
  - for polynomials with bitstream coefficients: to be done.

Curve Topology Computation (MSW)

Determine topology of zero-set of a polynomial of degree n in two variables.

Dependency on n reduces from n^{10} to n^6.



$$y^4 - y^3 + 2x^2y^2 + 3x^2y + x^4 = 0.$$