A Simple But Exact and Efficient Algorithm for Complex Root Isolation*

Michael Sagraloff
Max Planck Institute for Informatics
Saarbrücken, Germany
msagralo@mpi-inf.mpg.de

Chee K. Yap†
Courant Institute of Mathematical Sciences
New York University, New York
yap@cs.nyu.edu

ABSTRACT

We present a new exact subdivision algorithm CEVAL for isolating the complex roots of a square-free polynomial in any given box. It is a generalization of a previous real root isolation algorithm called EVAL. Under suitable conditions, our approach is applicable for general analytic functions. CEVAL is based on the simple Bolzano Principle and is easy to implement exactly. Preliminary experiments have shown its competitiveness.

We further show that, for the "benchmark problem" of isolating all roots of a square-free polynomial with integer coefficients, the asymptotic complexity of both algorithms EVAL and CEVAL matches (up a logarithmic term) that of more sophisticated real root isolation methods which are based on Descartes' Rule of Signs, Continued Fraction or Sturm sequences. In particular, we show that the tree size of EVAL matches that of other algorithms. Our analysis is based on a novel technique called δ-clusters from which we expect to see further applications.

Categories and Subject Descriptors

G.1.5 [**Numerical Analysis**]: Roots of Nonlinear Equations; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems

General Terms

Algorithms, Reliability, Theory

Keywords

exact root isolation, complexity of complex root isolation, Bolzano methods, subdivision algorithms, evaluation-based root isolation

*The full paper is available from `http://www.mpi-inf.mpg.de/~msagralo/` or `http://cs.nyu.edu/exact/`.

†Chee's work is supported by NSF Grants CCF-0917093 and CCF-0728977.

ISSAC'11, June 8–11, 2011, San Jose, California, USA.

1. INTRODUCTION

Root finding might be called the *Fundamental Problem of Algebra*, after the Fundamental Theorem of Algebra [40, 42, 47]. The literature on root finding is extremely rich, with a large classical literature. The work of Schönhage [40] marks the beginning of complexity-theoretic approaches to the Fundamental Problem. Pan [33] provides a history of root-finding from the complexity view point; see McNamee [23] for a general bibliography. The root finding problem can be studied as two distinct problems: root isolation and root refinement. In the complexity literature, the main focus is on what we call the **benchmark problem**, that is, isolating all the complex roots of a polynomial f of degree n with integer coefficients of at most L bits. Let $T(n, L)$ denote the (worst case) bit complexity of this problem. There are three variations on this benchmark problem:

- We can ask for only the real roots. Special techniques apply in this important case. E.g., Sturm [12, 21, 36], Descartes [9, 13, 15, 20, 28, 37], and continued fraction methods [1, 41, 44].
- We can seek the arithmetic complexity of this problem, that is, we seek to optimize the number $T_A(n, L)$ of arithmetic operations.
- We can add another parameter $p > 0$, and instead of isolation, we may seek to approximate each of the roots to p relative or absolute bits.

Schönhage achieved a bound of $T(n, L) = \widetilde{O}(n^3 L)$ for the benchmark isolation problem where $\widetilde{O}$ indicates the omission of logarithmic factors. This bound has remained intact. Pan and others [33] gave theoretical improvements in the sense of achieving $T_A(n, L) = \widetilde{O}(n^2 L)$ and $T(n, L) = T_A(n, L) \cdot \widetilde{O}(n)$, thus achieving record bounds simultaneously in both bit complexity and arithmetic complexity. Theoretical algorithms designed to achieve record bounds for the benchmark problem have so far not been used in practice. Moreover, the benchmark problem is inappropriate for some applications. For instance, we may only be interested in the first positive root (as in ray shooting in computer graphics), or in the roots in some specified neighborhood. In the numerical literature, there are many algorithms that are widely used and effective in practice but lack a guarantee on the global behavior (cf. [33] for discussion). Some "global methods" such as the Weierstrass or Durant-Kerner method that simultaneously approximate all roots seem ideal for the benchmark problem and work well in practice, but their convergence and/or complexity analysis are open. Thus, the benchmark complexity, despite its theoretical usefulness, has limitation as sole criterion in evaluating the usefulness of root isolation algorithms.

There are two sub-literature on “practical” root isolation algorithms: (1) One is the exact computation literature, providing algorithms used in various algebraic applications and computer algebra systems. Such exact algorithms have a well-developed complexity analysis and there is considerable computational experience especially in the context of cylindrical algebraic decomposition. The favored root isolation algorithms here, applied to the benchmark problem, tend to lag behind the theoretical algorithms by a factor of nL. Nevertheless, current experimental data justify their use [17, 37]. (2) The other is the numerical literature mentioned above. Although numerical algorithms traditionally lack any exactness guarantees, they have many advantages that practitioners intuitively understand: compared to algebraic methods, they are easier to implement and their complexity is more adaptive. Hence, there is a growing interest in constructing numerical algorithms that are exact and efficient.

§1. The Subdivision Approach.

Among the exact root isolation algorithms, the subdivision paradigm is widely used. It is a generalization of binary search in which we search for roots in a given domain (say a box $B_0 \subseteq \mathbb{C}$). Its principle action is a simple **subdivision phase** where we keep subdividing boxes into 4 congruent subboxes until each box B satisfies a predicate $C_{stop}(B)$. Typically, $C_{stop}(B) \equiv C_{out}(B) \vee C_{in}(B)$ where $C_{out}(B)$ is an **exclusion predicate** whose truth implies that B has no roots, and $C_{in}(B)$ is an **inclusion predicate** whose truth implies that B contains a unique root. Subdivision methods have the advantage of being “local”: they can restrict computational effort to the given box B_0, and may terminate quickly if there few or no roots in B_0.

Exact implementation of $C_{stop}(B)$ can be based on algebraic properties such as generalized Sturm sequences [47, Chap. 7]. Unfortunately, algebraic predicates are expensive. Since finding a root is metaphorically like “finding a needle in a hay stack”, an efficient exclusion predicate C_{out} can be highly advantageous. Numerical exclusion predicates have been used in Dedieu, Yakoubsohn and Taubin [11, 43, 46] but the inclusion predicate in these papers are inexact, based on an arbitrary ϵ-cutoff: $C_{in}(B) \equiv size(B) < \epsilon$. *Our paper will exploit numerical exclusion and inclusion predicates to yield exact subdivision algorithms.*

§2. Three Principles for Subdivision.

We compare three general principles used in subdivision algorithms for real root isolation: theory of Sturm sequences, Descartes’ rule of sign, and the Bolzano principle. The latter principle is simple and intuitive: *if a continuous real function $f(x)$ satisfies $f(a)f(b) < 0$, then there is a point c between a and b such that $f(c) = 0$. Furthermore, if f is differentiable and f' does not vanish on (a, b), then this root is unique in (a, b).* Modern algorithmic treatment of the Descartes method began with Collins and Akritas [9]. In recent years, algorithms based on the first two principles have been called (respectively) **Sturm method** and the **Descartes method**. By analogy, algorithms based on the third principle may be classified under the **Bolzano method** [7, 8, 26]. Note that the Bolzano principle is an analytic one, while Sturm is algebraic (Descartes seems to have an intermediate status). Johnson [17] has shown empirically that the Descartes method is more efficient than Sturm. Rouillier and Zimmermann [37] implemented a highly efficient exact real root isolation algorithm based on Descartes method. Since their theoretical bounds are indistinguishable, any practical advantage of Descartes over Sturm must be derived from the fact that the predicates in the Descartes method are cheaper. We believe that Bolzano methods have a similar advantage over Descartes. Such evidence is provided in a recent empirical study of Kamath [18] where a version of Ceval is compared with several algorithms, including the well-known Mpsolve of Bini and Fiorentini [3, 4]. Bolzano methods also have the advantage of greater generality: *The Bolzano method is applicable to the much larger class of complex analytic functions. Our* Ceval *algorithm can be adapted to such functions under mild conditions.*

§3. Complexity Analysis.

All complexity analysis is for the benchmark problem of isolating all roots of a polynomial $f(z)$. There are two complexity measures for subdivision algorithms: the subdivision tree size $S(n, L)$ and the bit complexity $P(n, L)$ of the subdivision predicates. Clearly, $T(n, L) \le S(n, L)P(n, L)$. But the analysis in this paper shows that $T(n, L)$ may be smaller than $S(n, L)P(n, L)$ by a factor of n. For the Sturm method, Davenport [10] has shown that for isolating all real roots of $f(x)$, we have $S(n, L) = O(n(L + \log n))$. This is optimal if $L \ge \log n$ [15]. The tree size in the Descartes method was only recently proven to be $O(n(L+\log n))$ [15] matching the Sturm bound. In this paper, we will prove that the tree size in the Bolzano method is $\widetilde{O}(n(L+\log n))$ for real roots. Furthermore, in our extension of the Bolzano method for complex roots the corresponding tree size is $\widetilde{O}(n^2(L + \log n))$. Despite this larger tree size, we prove that both real and complex Bolzano have $\widetilde{O}(n^4L^2)$ bit complexity, matching Descartes and Sturm.

Our complexity analysis of Bolzano methods is novel, and it opens up the exciting possibility of analysis of similar subdivision algorithms as in meshing of algebraic surfaces [5, 22, 35]. Perhaps it is no surprise that Bolzano methods could outperform the more sophisticated algebraic methods in practice. *What seems surprising from our analysis is that Bolzano methods could also match (up to a logarithmic factor) the theoretical complexity of algebraic methods as well.*

§4. Contributions of this paper.

1. Our complex root isolation algorithm (Ceval) is a contribution to the growing literature on exact algorithms based on numerical techniques and subdivision. The algorithm is simple and practical. Preliminary implementation shows that it is competitive with the highly regarded `MPSOLVE`.
2. This paper provides a rather sharp complexity analysis of Eval. Somewhat surprisingly, the worst-case bit-complexity of this simple algorithm can match (up to logarithmic-factors) those of sophisticated methods like Sturm or Descartes.
3. We further show that the more general Ceval also achieves the same bit complexity as Eval (despite the fact that the tree size of Ceval may be quadratically larger).
4. Our analysis is based on the novel technique of δ-clusters. We expect to see other applications of cluster analysis. This is a contribution to the general challenge of analyzing the complexity of numerical subdivision algorithms.

§5. Overview of Paper.

Section 2 reviews related work. The algorithm is presented in Section 3. In Section 4, we sketch our approach of δ-cluster from which we derive the complexity analysis of Eval and Ceval. In this conference paper, we summarize the most important results and provide sketches of the main proofs and techniques. Full proofs and further details on our δ-cluster analysis technique and 8-point test may be found in the full paper [39] on our homepages. Our original paper [39] only has the 8-point version of Ceval, not the simplified version described below.

2. PRIOR WORK

The main distinction among the various subdivision algorithms is the choice[1] of tests or predicates. One approach is based on doing root isolation on the boundary of the boxes. Pinkert [34] and Wilf [45] (see also [47, Chap. 7]) use Sturm-like sequences, while Collins and Krandick [19] considered the Descartes method. Such approaches are related to topological degree methods [29], which go back to Brouwer (1924). But root isolation on boundary of subdivision boxes and topological degrees computations are relatively expensive and unnecessary: as shown in this paper, weaker but cheaper predicates may be more effective. This key motivation for our present work came from subdivision algorithms for curve approximation where a similar phenomenon occurs [22]. We next review several previous work that are most closely related to our paper.

§6. Work of Pan, Yakoubsohn, Dedieu and Taubin.

Pan [30, 31, 32, 33] describes a subdivision algorithm with the current record asymptotic complexity bound. Pan regards his work as a refinement of Weyl's Exclusion Algorithm (1924). Weyl is also the basis for Henrici and Gargantini (1969) and Renegar (1987) (see [33]). The predicates are based on estimating the distance from the midpoint of a box B to the nearest zero of the input polynomial $f(z)$. Turan (1968) provides such a bound up to a constant factor, say 5. Pan further reduces this factor to $(1+\epsilon)$ (for a small $\epsilon > 0$) by applying the Graeffe iteration to $f(z)$. Finally, he combines the exclusion test with Newton-like accelerations to achieve the bound of $O(n^2 \ln n \ln(hn))$, where h is the cut-off depth of subdivision. Pan noted that "*there remains many open problems on the numerical implementation of Weyl's algorithm and its modification*" [33, p. 216]; in particular, "*proximity tests should be modified substantially to take into account numerical problems ... and controlling the precision growth*" [33, p. 193].

The approach of Yakoubsohn and Dedieu [11, 46] is much simpler than Pan's. Their algorithm keeps subdividing boxes until each box B satisfies an exclusion predicate $C_{out}(B)$, or B is smaller than an arbitrary cut-off $\epsilon > 0$. For any analytic function f, their predicate $C_{out}(B)$ is "$M^f(z, r\sqrt{2}) > 0$" where B is a square centered at z of length $2r$, and

$$M^f(z,t) := |f(z)| - \sum_{k\geq 1} \frac{|f^{(k)}(z)|}{k!} t^k. \tag{1}$$

It is easy to see that if $C_{out}(B)$ holds, then B has no roots of f. Taubin [43] introduce exclusion predicates that can be viewed as the linearized form of $M^f(z,t)$ or a Newton correction term. He shows their effectiveness in approximating (rasterizing) surfaces. These algorithms are useful in practice, but the use of ϵ-cutoff does not constitute a true inclusion predicate in the sense on §1: at termination, we have a collection of non-excluded ϵ-boxes, none of which is guaranteed to isolate a root.

§7. The Eval Algorithm.

The starting point for this paper is a simple algorithm for real root isolation. Suppose we want to isolate the roots of a real analytic function $f : \mathbb{R} \to \mathbb{R}$ in the interval $I_0 = [a, b]$. Assume f has only simple roots in I_0. For any interval I with center $m = m(I)$ and width $w = w(I)$, we introduce two interval predicates using the function in (1):

$$\left.\begin{array}{lcl} C_0(I) & \equiv & M^f(m, w/2) > 0 \\ C_1(I) & \equiv & M^{f'}(m, w/2) > 0 \end{array}\right\} \tag{2}$$

Clearly, $C_0(I)$ is an exclusion predicate. Note that if $C_1(I)$ holds, then f has at most one zero in I. Thus, $C_1(I)$ in combination with the following **root confirmation test**

$$f(a)f(b) < 0, \qquad \text{where } I = [a, b], \tag{3}$$

constitutes an inclusion predicate. Here is the algorithm:

	Eval(I_0):
	Check the endpoints of I_0, and output them if they are zeros of f
	Let Q be a queue of intervals, initialized as $Q \leftarrow \{I_0\}$
	While Q is non-empty:
	Remove I from Q.
1.	If $C_0(I)$ holds, discard I.
2.	Else if $C_1(I)$ holds,
3.	If I passes the confirmation test (3), output I.
4.	Else, discard I.
5.	Else
6.	If $f(m) = 0$, output $[m, m]$ where $m = m(I)$.
7.	Split I at m and put both subintervals into Q.

Termination and correctness are easy to see (e.g., [7]). Output intervals either have the exact form $[m, m]$ or are regarded as open intervals (a, b). This algorithm is easy to implement exactly if we assume that all intervals are represented by dyadic numbers.

Mitchell [26] seems to be the first to explicitly describe Eval, but as he assumes approximate floating point arithmetic, he does not check if $f(m) = 0$ at the midpoint m. He attributes ideas to Moore [27]. The second author of the present paper initiated the complexity investigation of Eval (and its extension for multiple roots) as the 1-D analogue of the surface meshing algorithm of Plantinga-Vegter [5, 22, 35]. In [7], we succeeded in obtaining a bound of $O(n^3(L + \log n))$ when Eval is applied to the benchmark problem. The proof involves several highly technical tools, but the approach is based on the novel concept of **continuous amortization**. The idea is to bound the tree size in terms of an integral $\int_I \frac{dx}{F(x)}$ where $F(x)$ is a suitable "stopping function". Recently, Burr and Krahmer [6] simplified the choice of $F(x)$, obtaining a tree size bound $O(n(L + \ln n))$ for Eval. Such a bound is optimal for $L \geq \ln n$ (see [15]), and matches the bounds in the present paper, as well as those for Descartes and Sturm methods. But they require f' to be square-free. Our present paper uses a different analysis to obtain a slightly weaker bound of

[1] We use the terms "predicate" and "test" interchangeably.

$O(n(L+\ln n)(\ln L+\ln n))$, but we do not require the square-freeness of f'. Furthermore, our analysis extends to the complex root isolation algorithm CEVAL. Our upper bound for the bit complexity of CEVAL matches those of EVAL, Sturm and Descartes method.

3. THE COMPLEX ROOT ALGORITHM

In this section, we describe CEVAL, the complex analogue of EVAL. In fact, we describe two versions of CEVAL, and only prove the correctness of the simpler version here. The algorithm in described in way that allows a straightforward exact implementation.

Notation. For the rest of this paper, we fix a square-free polynomial $f \in \mathbb{C}[z]$ of degree n. For $m \in \mathbb{C}$ and $r > a$ real value, we denote $D_r(m)$ the disk of **radius** $r(D) = r$ centered at $m(D) = m$. For $\xi, \mu \in \mathbb{C}$, we write "$\xi \leq \mu$" if $\mathtt{Re}(\xi) \leq \mathtt{Re}(\mu)$ and $\mathtt{Im}(\xi) \leq \mathtt{Im}(\mu)$. A subset $B \subseteq \mathbb{C}$ is called a **box** if $B = B(\xi,\mu) := \{z \in \mathbb{C} : \xi \leq z \leq \mu\}$ for some $\xi \leq \mu$. We further define $m(B) := (\xi+\mu)/2$ the **midpoint** and $w(B) := \max\{|\mathtt{Re}(\mu-\xi)|, |\mathtt{Im}(\mu-\xi)|\}$ the **width** of B. Its **radius** $r(B)$ is defined as the radius of the smallest disk centered in $m(B)$ and containing B. Obviously, $\bar{r}(B) := 3w(B)/4$ is an upper bound on $r(B)$. We can split a box B into four congruent subboxes, called the **children** of B. The boundary of a region $R \subseteq \mathbb{C}$ is denoted ∂R (R is usually a disk or a box). A connected region R is said to be **isolating** if it contains exactly one zero of $f(z)$.

§8. Complex Analogues of C_0 and C_1 Predicates.

For $m \in \mathbb{C}$ and $K, r > 0$, we define the **test function** $t^f(m,r)$ and the **predicate** $T_K^f(m,r)$ as follows:

$$t^f(m,r) := \sum_{k\geq 1} \left|\frac{f^{(k)}(m)}{f(m)}\right| \frac{r^k}{k!} \tag{4}$$

$$T_K^f(m,r) \equiv t^f(m,r) < \frac{1}{K} \tag{5}$$

Since f is fixed in this paper, we simply write $T_K(m,r)$ for $T_K^f(m,r)$. When f' is used in place of f, we simply write $T'_K(m,r)$ for $T_K^{f'}(m,r)$. Moreover, for a disk D, we may write $T_K(D)$ for $T_K(m(D), r(D))$, etc. We remark that the success of $T_K^f(m,r)$ implies the success of $T_{K'}^f(m,r)$ for any $K' \leq K$ and $r' \leq r$, and $T_K^f(m,r)$ is equivalent to $T_K^{f(m+rz/\lambda)}(0,\lambda)$ with $\lambda \in \mathbb{R}$ an arbitrary positive real value.

LEMMA 1 (EXCLUSION-INCLUSION PROPERTIES). *Consider any disk $D = D_r(m)$:*
(i) If $T_1(D)$ holds, the closure $\overline{D}$ of D has no root of f.
(ii) If $T_1(D)$ fails, the disc $D_{2nr}(m)$ has some root of f.
(iii) If $T'_{\sqrt{2}}(D)$ holds, $\overline{D}$ has at most one root of f.

Proof. See [2, 39] for proofs of (i) and (iii). We show the contrapositive of (ii): let $z_1, \ldots, z_n$ denote the roots of f and suppose that $D_{2nr}(m)$ contains no root. Then,

$$\left|\frac{f^{(k)}(m)}{f(m)}\right| = \left|\sum\nolimits'_{i_1,\ldots,i_k} \frac{1}{(m-z_{i_1})\ldots(m-z_{i_k})}\right|$$
$$\leq \Sigma_k(m) := \left(\sum_{i=1}^{n}\left|\frac{1}{m-z_i}\right|\right)^k \leq \left(\frac{1}{2r}\right)^k, \tag{6}$$

where the prime means that the i_j's ($j = 1 \ldots k$) are chosen to be distinct. Hence, it follows that

$$\sum_{k\geq 1}\left|\frac{f^{(k)}(m)}{f(m)}\right|\frac{r^k}{k!} < \sum_{k\geq 1}\frac{1}{k!}\left(\frac{1}{2}\right)^k = e^{\frac{1}{2}} - 1 < 1$$

and, thus, $T_1(D)$ holds. **Q.E.D.**

Part (i) of the lemma shows that $T_1(D)$, in analogy to $C_0(I)$, is an exclusion predicate for $D = D_r(m)$. Part (ii) shows that the negation of $T_1(D)$ is a root confirmation test like (3), albeit for the enlarged disc $D^+ := D_{2nr}(m)$. Part (iii) shows that $T'_{\sqrt{2}}(D)$ plays the role of the predicate $C_1(I)$. From (ii) and (iii) we could derive an inclusion predicate.

The next lemma gives lower bounds on the size of discs that pass our tests. The bounds are in terms of the **separation** $\sigma(\xi) := \min_{j\neq i}|z_j - \xi|$ of a root $\xi := z_i$ of f, and the **separation** $\sigma(f) := \min_i \sigma(z_i)$ **of** f.

LEMMA 2. *Consider a disk D and a root $\xi := z_i$ of f:*
(i) If $r(D) \leq \sigma(f)/(4n^2)$, then $T_1(D)$ or $T'_{\sqrt{2}}(D)$ holds.
(ii) If D contains ξ and $r \leq \sigma(\xi)/(4n^2)$, then $T'_{\sqrt{2}}(D)$ holds.
(iii) If D contains ξ and $r \leq \sigma(\xi)/(8n^3)$, then D^+ is isolating.

Proof. For (i), suppose that $r(D) \leq \sigma(f)/(4n^2)$ and both $T_1(D)$ and $T'_{\sqrt{2}}(D)$ do not hold. Then, according to Lemma 1 (ii), $D_{2nr(D)}(m)$ must contain a root z of f. The same result applied to f' shows that $D_{2nr(D}(m)$ also contains a root z' of f'. It follows that $|z - z'| < 4nr(D) \leq \sigma(f)/n \leq \sigma(z)/n$ contradicting the fact [13, 47] that $D_{\sigma(z)/n}(z)$ does not contain any root of the derivative f'. Part (ii) follows from (i) since $\xi \in D$ implies that $T_1(D)$ does not hold. Part (iii) is a direct consequence of (ii). **Q.E.D.**

§9. Simplified Complex Root Isolation.

We are ready to present a complex version of EVAL. Call a disk $D_r(m)$ **well-isolating** if $D_r(m)$ and $D_{2r}(m)$ are both isolating. The property we exploit is that if D and D' are both well-isolating with non-empty intersection, then they share a common root in $D \cap D'$. Our algorithm produces well-isolated disks:

Simplified CEVAL(B_0, f):
Input: Box B_0, and square-free polynomial $f(z)$ of degree n.
Output: List $\mathcal{L}$ of disjoint well-isolating disks, centered in B_0.

$Q \leftarrow \{B_0\}$. $\mathcal{L} \leftarrow \emptyset$.
While Q is non-empty:
Remove B from Q. Let $m = m(B)$, $\bar{r} = \frac{3}{4}w(B) > r(B)$.
1. If $T_1(m,\bar{r})$ holds, discard B.
2. Else if $T'_{\sqrt{2}}(m, 4n\bar{r})$ holds:
2.1 If $D_{2n\bar{r}}(m)$ intersects any disk D' in $\mathcal{L}$,
2.2 replace D' by the smaller of $D_{2n\bar{r}}(m)$ and D'.
2.3 Else insert $D_{2n\bar{r}}(m)$ into $\mathcal{L}$.
3. Else
Split B into four children and insert them into Q.

Correctness of our algorithm is based on three claims:

THEOREM 3 (CORRECTNESS).
(i) The algorithm halts: indeed, no box of width less than $\sigma(f)/(12n^3)$ is subdivided.
(ii) $\mathcal{L}$ is a list of well-isolating disks, each centered in B_0.
(iii) Every root of $f(z)$ in B_0 is isolated by some disk in $\mathcal{L}$.

Proof. Claim (i) is true because Lemma 2(i) implies that the tests in Steps 1 or 2 must pass when $4n\bar{r} \le \sigma(f)/(4n^2)$, and by definition $\bar{r} = 3w(B)/4$. To see (ii), observe that the disc $D_{2n\bar{r}}(m)$ is inserted into $\mathcal{L}$ in Steps 2.2 or 2.3. The m and $\bar{r}$ in Step 2.1 have the properties that $T_1(m, \bar{r})$ fails and $T'_{\sqrt{2}}(m, 4n\bar{r})$ succeeds. Then, Lemma 1(ii,iii) implies that $D_{2n\bar{r}}(m)$ is well-isolating. To see (iii), observe that boxes $B \subseteq B_0$ are discarded in Steps 1 or 2.2 of the algorithm: Step 1 is justified by Lemma 1(i) and Step 2.2 is justified because of the above-noted property of well-isolating disks.

Q.E.D.

§10. The Eight Point Test.
Instead of relying on Lemma 1(ii) for root confirmation, we offer another root confirmation test that is closer in spirit to the sign-change idea in (3), and which could be generalized for analytic functions. The idea is to look at the 8 compass points (N,S,E,W, NE, SE, NW, SW) on the disk $D_{4r}(m)$ as illustrated in Figure 1. These compass points divide the boundary $\partial D_{4r}(m)$ of the disk into 8 arcs $A_0, \ldots, A_7$ where $A_j := \{m + 4re^{\mathbf{i}\theta} : j\pi/4 \le \theta < (j+1)\pi/4\}$.

We rewrite the function $f(z)$ as $f(x + \mathbf{i}y) = u(x, y) + \mathbf{i}v(x, y)$, where $z = x + \mathbf{i}y$, $\mathbf{i} = \sqrt{-1}$ and u and v are the real and imaginary part of f. So $f(x+\mathbf{i}y) = 0$ iff $u(x, y) = 0$ and $v(x, y) = 0$. Since the roots are simple, the u- and v-curves intersect at right angles. We say that there is an **arcwise u-crossing** at A_j if $u(m+4re^{\mathbf{i}j\pi/4}) \cdot u(m+4re^{\mathbf{i}(j+1)\pi/4}) < 0$ or $u(m + 4re^{\mathbf{i}j\pi/4}) = 0$.

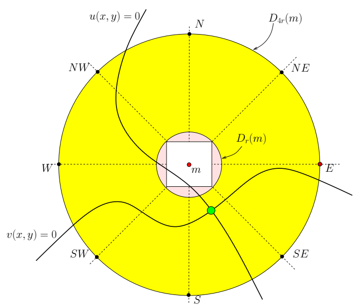

Figure 1: 8 compass points.

If r is sufficiently small, then we want to detect roots in $D_r(m)$ by arcwise u- and v-crossings. More precisely: we say $D_{4r}(m)$ passes the **8-Point test** if there are exactly two arcwise u-crossings at A_j, A_k, $(j < k)$ and exactly two arcwise v-crossings at $A_{j'}, A_{k'}$ $(j' < k')$, and these **interleave** in the sense that either $0 \le j < j' < k < k' < 8$ or $0 \le j' < j < k' < k < 8$.

We introduce the following novel test to confirm the existence of ordinary roots.

Theorem 4 (Success of 8-Point Test). *Suppose $T'_6(m, 4r)$ holds and the 8-point test is applied to $D_{4r}(m)$.*
(i) If $D_{4r}(m)$ fails the test, then $D_r(m)$ is non-isolating.
(ii) If $D_{4r}(m)$ passes the test, then $D_{4r}(m)$ is isolating.

Using the 8-point test, we devise an alternative to the simplified Ceval. This **8-point Ceval** is described in the full version [39] of this paper including the proof of Theorem 4 which is non-trivial. The cardinal points (N,S,E,W) are dyadic assuming the center and radius are dyadic; however the ordinal points (NE,SE,SW,NW) are irrational. Hence for exact implementation, we show how the correctness of the 8-Point test is preserved if we use rational points that are slightly perturbed versions of ordinal points. The 8-point test has independent interest: (a) For analytic functions, we no longer have Lemma 1(ii) for root confirmation, but some kind of 8-point test is applicable. More precisely, the tests $T^f_K(m, r)$ can be considered for arbitrary analytic function, and the same argumentation as in the case of polynomials shows the correctness of Lemma 1(i),(iii) and Theorem 4. (b) We can use it to "confirm" the output from pure-exclusion algorithms such as Yakoubsohn-Dedieu's in §6. The asymptotic complexity of these two forms of Ceval for the benchmark problem are the same. This is due to the fact that there exists a corresponding result to Lemma 2 for the 8-point test.

4. COMPLEXITY ANALYSIS

In this section, we analyze the complexity of Eval and the simplified Ceval. For this purpose, we use the benchmark problem of isolating all roots of a square-free polynomial of degree n with L-bit integer coefficients. The initial start box for Ceval may be assumed to be $B_0 = B(-2^L(1 + \mathbf{i}), 2^L(1 + \mathbf{i}))$. For Eval, we can start with the interval $I_0 = (-2^{-L}, 2^L)$. According to Cauchy's bound [47], B_0 contains all complex roots $z_1, \ldots, z_n \in \mathbb{C}$ of f (thus, I_0 all real roots of f). Throughout the following considerations, let $\mathcal{T}^{CE}$ and $\mathcal{T}^{EV}$ denote the subdivision trees induced by Ceval and Eval, respectively.

§11. Cluster Analysis and Tree Size.
In (6), we have already seen that $\Sigma_k(m) := (\sum_i \frac{1}{|m-z_i|})^k = (\Sigma_1(m))^k$ constitutes an upper bound on $\frac{|f^{(k)}(m)|}{|f(m)|}$ for all $k \ge 1$. Furthermore, $\Sigma_1(m) < \nu$ for a $\nu > 0$ implies that $\sum_{k\ge1} \left|\frac{f^{(k)}(m)}{f(m)}\right| \frac{r^k}{k!} < e^{\nu r} - 1$ and, thus,

$$T^f_K(m, r) \text{ holds if } \Sigma_1(m) < \frac{1}{r} \ln\left(1 + \frac{1}{K}\right). \tag{7}$$

Now let us consider an arbitrary box B of depth h in the subdivision process, that is, B has width $w(B) = w_h := 2^{L+1-h}$. Let $\bar{r} = \bar{r}(B) = 3w(B)/4$ be the upper bound on the radius of B used in the Ceval algorithm. If the midpoint $m(B)$ of B fulfills $|m(B)-z_i| > 2n\cdot\bar{r}$ for all $i = 1, \ldots, n$, then $\Sigma_1(m(B)) < \frac{1}{2\bar{r}} < \frac{\ln 2}{\bar{r}}$, thus $T_1(m(B), \bar{r})$ holds according to the above consideration and B is discarded. It follows that, for each root z_i, there exist at most $O(n^2)$ disjoint boxes B of the same size with $|m(B) - z_i| \le 2n\bar{r}$. Hence, in total, at most $O(n^3)$ boxes are retained at each subdivision level h. From this straightforward observation we immediately derive the upper bound $O(n^3)$ on the width of $\mathcal{T}^{CE}$. For Eval, a similar argumentation shows that $O(n^2)$ intervals are retained at each subdivision level. This consideration is based on a pretty rough estimation of $\Sigma_1(m)$ which assumes that, from a given point m, the distances to all roots z_i are nearly of the same minimal value. In order to improve the latter estimate, we introduce the concept of **δ-clusters** of roots, where δ is an arbitrary positive real value. We will show that, outside some "smaller" neighborhood of the roots of f, the sum $\Sigma_1(m)$ is sufficiently small to guarantee the success of our exclusion predicate T_1:

Theorem 5. *For arbitrary $\delta > 0$, there exist disjoint, axes-parallel, open boxes $B_1, \ldots, B_k \subset \mathbb{C}$ $(k \le n^2)$ such that:*
(i) $\mathcal{B} := \bigcup_{i=1,\ldots,k} B_i$ covers all roots $z_1, \ldots, z_n$.
(ii) $\mathcal{B}$ covers an area of less than or equal to $4n^2\delta^2$.
(iii) For each point $m \notin \mathcal{B}$, we have $\Sigma_1(m) \le \frac{2(1+\ln\lceil n/2\rceil)}{\delta}$.

Proof. We only provide a sketch of the proof and refer the reader to the full paper [39] for a complete reasoning. The roots $z_1, \ldots, z_n$ are first projected onto the real axes defining a multiset (elements may appear several times) $R_{\mathtt{Re}} = \{x_1, \ldots, x_n\}$ in $\mathbb{R}$. The latter points are now partitioned into disjoint multisets $R_1, \ldots, R_l$ such that the following properties are fulfilled:

(a) Each R_i is a so called **δ-cluster** which is defined as follows: The corresponding **δ-interval**

$$I_\delta(R_i) = (\mathrm{cg}(R_i) - \delta|R_i|, \mathrm{cg}(R_i) + \delta|R_i|),$$

with $\mathrm{cg}(R_i) = \frac{\sum_{x\in R_i} x}{|R_i|}$ the **center of gravity** of R_i, contains all elements of R_i. In addition, we can order the elements of R_i in way such that their distances to the right boundary of $I_\delta(R_i)$ are at least $\delta, 2\delta, \ldots, |R_i|\delta$, respectively, and the same for the left boundary of $I_\delta(R_i)$.

(b) The δ-intervals $I_\delta(R_i)$ are pairwise disjoint.

The construction of a partition of $R_{\mathtt{Re}}$ with the above properties is rather simple: We start with the trivial partition of $R_{\mathtt{Re}}$ into n δ-clusters each consisting of one element of $R_{\mathtt{Re}}$. An easy computation shows that the union of two δ-clusters for which (b) is not fulfilled is again a δ-cluster. Thus, we iteratively merge δ-clusters whose corresponding δ-intervals overlap until (b) is eventually fulfilled. It is now easy to see that, for each $x \in \mathbb{R}\setminus\bigcup_i I_\delta(R_i)$, the inequality $\sum_j \frac{1}{|x-x_j|} \le \frac{2(1+\ln\lceil n/2\rceil)}{\delta}$ holds.

In a second step, we project the roots of f onto the imaginary axes defining a multiset $R_{\mathtt{Im}}$ for which we proceed in exactly the same manner as for $R_{\mathtt{Re}}$. Let $S_1, \ldots, S_{l'}$ be the corresponding partition of $R_{\mathtt{Im}}$, then the overlapping of the stripes $\mathtt{Re}(z) \in I_\delta(R_i)$ and $\mathtt{Im}(z) \in I_\delta(S_j)$ defines $k \le n^2$ boxes $B_1, \ldots, B_k$ covering an area of total size $4n^2\delta^2$ or less. Now, for each $m \notin \mathcal{B} = \bigcup_i B_i$, either $\mathtt{Re}(m) \notin \bigcup_i I_\delta(R_i)$ or $\mathtt{Im}(m) \notin \bigcup_i I_\delta(S_i)$. In the first case, we have

$$\Sigma_1(m) \le \sum_{j=1}^{n} \frac{1}{|\mathtt{Re}(m) - \mathtt{Re}(z_j)|} \le \frac{2(1+\ln\lceil n/2\rceil)}{\delta}.$$

The case $\mathtt{Im}(m) \notin \bigcup_j I_\delta(S_j)$ is treated in exactly the same manner. **Q.E.D.**

We now apply the above theorem to

$$\delta := r \cdot \frac{(1+\ln\lceil n/2\rceil)}{\ln 2} = \frac{3w(B)(1+\ln\lceil n/2\rceil)}{4\ln 2}$$

and use (7). It follows that, for all m outside a union of boxes covering an area of size $w(B)^2 \cdot O((n\ln n)^2)$, we have $\Sigma_1(m) < \frac{1}{r}\ln 2$. Thus, at any level in the subdivision process, only $O((n\ln n)^2)$ boxes are retained. For Eval, we can apply the real counterpart of Theorem 5 which says that there exist $k \le n$ disjoint intervals $I_1, \ldots, I_k$ that cover the projections of all z_i onto the real axes, the total size of all intervals is $\le 2n\delta$, and $\Sigma_1(m) \le \frac{2(1+\ln\lceil n/2\rceil)}{\delta}$ for each m located outside all I_j. It follows that the width of $\mathcal{T}^{EV}$ can be bounded by $O(n\ln n)$. A more refined argument even shows that, at a subdivision level h, the width of the tree adapts itself to the number k_h of roots z_i with separation $\sigma(z_i) \le 16n^3 w_h = 2^{L+5-h}n^3$ related to the width $w_h = 2^{L+1-h}$ of the boxes at that level. We refer the reader to the full paper for the non-trivial proof. We fix this result:

Theorem 6. *Let $h \in \mathbb{N}_0$ be an arbitrary subdivision level and k_h be the number of roots z_i with $\sigma(z_i) \le 16n^3 w_h = 2^{L+5-h}n^3$. Then, the width of $\mathcal{T}^{CE}$ at level h is upper bounded by*

$$16k_{h-1}^2(17 + \ln\lceil k_{h-1}/2\rceil) = O(k_{h-1}^2(\ln k_{h-1})^2)$$

and the width of $\mathcal{T}^{EV}$ is upper bounded by

$$4k_{h-1}(17 + \ln\lceil k_{h-1}/2\rceil) = O(k_{h-1}\ln k_{h-1}).$$

In order to translate the above result on the treewidth into a bound on the treesize in terms of the degree n and the bitsize L, we have to derive an estimate for k_h. The main idea is to apply the generalized Davenport-Mahler bound [12, 13] to the roots of f. In a first step, we partition the set $R = \{z_1, \ldots, z_n\}$ of roots into disjoint sets $R_1, \ldots, R_l$ such that $|R_{i_0}| \ge 2$ for each $i_0 = 1, \ldots, n$ and $|z_i - z_j| \le 2^{L+5-h}n^3 \cdot |R_{i_0}| \le 2^{L+5-h}n^4$ for all pairs $z_i, z_j \in R_{i_0}$: Starting with the set $R_1 := \{z_1\}$, we can iteratively add roots to R_1 that have distance $\le 2^{L+5-h}n^3$ to at least one root within R_1. When there is no further root to add, we proceed with a root z_i not contained in R_1 and construct a set R_2 from $\{z_i\}$ in the same manner, etc; see [39].

In a second step, we consider a directed graph $\mathcal{G}_i$ on each R_i which connects consecutive points of R_i in ascending order of their absolute values. We define $\mathcal{G} := (R, E)$ as the union of all $\mathcal{G}_i$. Then $\mathcal{G}$ is a directed graph on R with the following properties:

1. each edge $(\alpha, \beta) \in E$ satisfies $|\alpha| \le |\beta|$,
2. $\mathcal{G}$ is acyclic, and
3. the in-degree of any node is at most 1.

Now, the generalized Davenport-Mahler bound applies:

$$\prod_{(\alpha,\beta)\in E} |\alpha - \beta| \ge \frac{1}{((n+1)^{1/2}2^L)^{n-1}} \cdot \left(\frac{\sqrt{3}}{n}\right)^{\#E} \cdot \left(\frac{1}{n}\right)^{n/2}$$

As each set R_i contains at least 2 roots, we must have $\#E \ge k_h/2$. Furthermore, for each edge $(\alpha, \beta) \in E$, we have $|\alpha - \beta| \le 16n^4 w_h = 2^{L+5-h}n^4$, thus,

$$\begin{aligned}(2^{L+5-h}n^4)^{\frac{k_h}{2}} &\ge \tfrac{1}{((n+1)^{1/2}2^L)^{n-1}} \cdot \left(\tfrac{\sqrt{3}}{n}\right)^{k_h} \cdot \left(\tfrac{1}{n}\right)^{n/2} \\ &> \tfrac{1}{(n+1)^n 2^{nL}} \cdot \left(\tfrac{3}{n^2}\right)^{k_h/2} > n^{-n-k_h}2^{-n(L+1)}.\end{aligned}$$

A simple computation then shows that

$$k_h < \frac{16n(L+\ln n)}{h-2L} \quad \forall h > h_0 := \max(2L, \lceil 64\ln n + L\rceil). \tag{8}$$

In particular, the bound $O(n(L+\ln n)))$ on the depth of the subdivision tree immediately follows. Namely, if $k_{h+1} < 1$, then $k_h = 0$ and, thus, $\sigma(f) < 2^{L+4-h}n^3 < 12w_h n^3$. But this implies that, at subdivision level h, no box is further subdivided (Theorem 3). For $h \le h_0$, the trivial inequality $k_h \le n$ holds. Now, we can derive our bound on the tree size by summing up the number of nodes over all subdivision levels, where we use Theorem 6 and the bound (8) for k_h. A similar computation also applies to the tree induced by the Eval algorithm; see [39] for details.

Theorem 7. *Let f be a square-free polynomial of degree n with integer coefficients of bit-size $\le L$. Then,*
(i) the subdivision tree $\mathcal{T}^{CE}$ has size $\widetilde{O}(n^2L)$.
(ii) the subdivision tree $\mathcal{T}^{EV}$ has size $\widetilde{O}(nL)$.

§12. Bit Complexity.

For the bit complexity analysis of CEVAL, we consider the computational costs at a node (box B) of depth h. So B has width $w(B) = w_h = 2^{L+1-h}$. In order to evaluate $T_1^f(m(B), \bar{r})$ and $T_{\sqrt{2}}^{f'}(m(B), 2n\bar{r})$, where $\bar{r} = \frac{3}{4}w(B)$ bounds the radius $r(B)$ of B, we compute

$$f_B(z) = f(m(B) + w(B) \cdot z)$$

and test whether $T_1^{f_B(z)}(0, 3/4)$ or $T_{\sqrt{2}}^{f'_B(z)}(0, 3n)$ holds. Notice that the latter two tests are equivalent to $T_1^f(m(B), \bar{r})$ and $T_{\sqrt{2}}^{f'}(m(B), 4n\bar{r})$, respectively. We first bound the costs for computing $f_B(z)$: For a polynomial $g(z) := \sum_{i=0}^n g_i z^i$ with binary fractions $g_i = m_i \cdot 2^{-\tau_i}$, $m_i \in \mathbb{Z}$ and $\tau_i \in \mathbb{N}_0$, as coefficients, we say that g **has bitsize** $\tau(g)$ if multiplication of g by the common denominator $2^{\max_i \tau_i}$ of all g_i leads to an integer polynomial with coefficients of at most $\tau(g)$ bits. For our starting box B_0, the polynomial $f_{B_0}(z) = f(2^{L+1}z)$ has bitsize $O(nL)$ because of the scaling operation $z \mapsto 2^{L+1}z$. We incrementally compute $f_{B'}$ from f_B via the substitution $z \mapsto (z \pm 1 \pm \mathbf{i})/2$, where B' is one of the four children of B. Hence, the bitsize of $f_{B'}$ increases by at most n compared to the bitsize of f_B. It follows that, for a box B at subdivision level h, f_B has bitsize $\tau_B = O(n(L+h))$. $f_{B'}$ is computed from f_B by first substituting z by $z/2$ followed by a Taylor shift by 1 and then by $\mathbf{i}$, that is, $z \mapsto z \pm 1 \pm \mathbf{i}$. A Taylor shift by $\mathbf{i}$ can be realized as a Taylor shift by 1 combined with two scalings by $\mathbf{i}$, using the identity $f(z + \mathbf{i}) = f(\mathbf{i}(-\mathbf{i}z + 1))$. The scalings by $\mathbf{i}$ are easy. Using asymptotically fast Taylor shift [16], each shift by 1 requires $\widetilde{O}(n(n + \tau_B)) = O(n^2(L + h))$ bit operations.

To evaluate the polynomials in the predicates $T_1^{f_B(z)}(0, 3/4)$ and $T_{\sqrt{2}}^{f'_B(z)}(0, 3n)$, we have to compute the value of a polynomial of bitsize $O(n(L+h))$ at a point of bit size $O(1)$ and $O(\log n)$, respectively. Therefore, $\widetilde{O}(n(L+h))$ bit operations suffice and so the overall bit complexity for a box of depth h is $\widetilde{O}(n^2(L + h))$. An analogous argument shows that, for an interval I at level h (i.e., $w(I) = 2^{L+1-h}$), EVAL requires $\widetilde{O}(n^2(L+h))$ bit operations as well. Thus, the bit complexity at each node is bounded by $\widetilde{O}(n^3 L)$ since $h = O(n(L+\ln n))$.

For EVAL, the claimed bit complexity of $\tilde{O}(n^4L^2)$ follows easily by multiplying the bound $\tilde{O}(nL)$ from Theorem 7 on the number of nodes with the bound $\widetilde{O}(n^3L)$ on the bit operations at each node. Furthermore, a simple computation (see [39]) combining our results on the width of $\mathcal{T}^{CE}$ and the costs at each node at any subdivision level h leads to the overall bit complexity of $\tilde{O}(n^4L^2)$ for CEVAL. It is worth noting that the larger tree size of $\mathcal{T}^{CE}$ (compared to $\mathcal{T}^{EV}$) does not effect the overall bit complexity. Intuitively, most of the nodes of $\mathcal{T}^{CE}$ are at subdivision levels where the computational costs are considerably smaller than the worst case bound $\tilde{O}(n^3L)$.

THEOREM 8. *For a square-free polynomial f of degree n with integer coefficients with absolute value bounded by 2^L, the algorithms* CEVAL *and* EVAL *isolate the complex (real) roots of f with a number of bit operations bounded by $\widetilde{O}(n^4L^2)$.*

5. CONCLUSION

This paper introduced CEVAL, a new complex root isolation algorithm, continuing a line of recent work to develop exact subdivision algorithms based on the Bolzano principle. The primitives in such algorithms are simple to implement and extendible to analytic functions. Our 8-Point CEVAL algorithm has been implemented in Kamath's thesis [18] using the `Core Library` [48], and compares favorably to Yakoubsohn's algorithm and MPSOLVE [3, 4].

The complexity of CEVAL is theoretically competitive with that of known exact practical algorithms for real root isolation. It is somewhat unexpected that our simple evaluation-based algorithms can match those based on sophisticated primitives like Descartes or Sturm methods. Another surprise is that the complex case has (up to $\widetilde{O}$-order) the bit complexity of the real case despite its larger subdivision tree.

Our complexity analysis introduces new ideas including a technique of root clusters which has proven to have other applications [24] as well. One open problem is to sharpen our complexity estimates (only logarithmic improvements can be expected).

The Descartes method had been successfully extended to the bitstream model [14, 25] in which the coefficients of the input polynomial are given by a bitstream on-demand. It has useful applications in situations where the coefficients are algebraic numbers (e.g., in cylindrical algebraic decomposition). Recent work [38] shows that the CEVAL algorithm also extends to bitstream polynomials.

6. REFERENCES

[1] A. G. Akritas and A. Strzeboński. A comparative study of two real root isolation methods. Nonlinear Analysis:Modelling and Control, 10(4):297–304, 2005.

[2] E. Berberich, P. Emeliyanenko, and M. Sagraloff. An elimination method for solving bivariate polynomial systems: Eliminating the usual drawbacks. ALENEX 2011, pp. 35-47, Jan 22, 2011. San Francisco.

[3] D. A. Bini. Numerical computation of polynomial zeroes by means of Aberth's method. Numerical Algorithms, 13:179–200, 1996.

[4] D. A. Bini and G. Fiorentino. Numerical Computation of Polynomial Roots Using MPSolve Version 2.2. Dipart. di Matematica, Univ. di Pisa. Jan 2000. `ftp://ftp.dm.unipi.it/pub/mpsolve/MPSolve-2.2.tgz`.

[5] M. Burr, S. Choi, B. Galehouse, and C. Yap. Complete subdivision algorithms, II: Isotopic meshing of singular algebraic curves. ISSAC'08, pp. 87–94. Accepted Special JSC Issue.

[6] M. Burr and F. Krahmer. SqFreeEVAL: An (almost) optimal real-root isolation algorithm. CoRR, abs/1102.5266, 2011.

[7] M. Burr, F. Krahmer, and C. Yap. Continuous amortization: A non-probabilistic adaptive analysis technique. ECCC, TR09(136), Dec 2009.

[8] M. Burr, V. Sharma, and C. Yap. Evaluation-based root isolation, Feb. 2009. In preparation.

[9] G. E. Collins and A. G. Akritas. Polynomial real root isolation using Descartes' rule of signs. In R. D. Jenks, ed., ACM Symp. on Symb. and Alg. Comp., pp. 272–275. ACM Press, 1976.

[10] J. H. Davenport. Computer algebra for cylindrical algebraic decomposition. Tech. Rep., Royal Inst. of Tech., Dept. of Numerical Analysis & Comp. Sci., Stockholm, Sweden, 1985.

[11] J.-P. Dedieu and J.-C. Yakoubsohn. Localization of an

algebraic hypersurface by the exclusion algorithm. AAECC, 2:239–256, 1992.

[12] Z. Du, V. Sharma, and C. Yap. Amortized bounds for root isolation via Sturm sequences. In D. Wang and L. Zhi, eds., Symbolic-Numeric Computation, pp. 113–130. Birkhäuser, Basel, 2007. Proc. SNC 2005.

[13] A. Eigenwillig. Real Root Isolation for Exact and Approximate Polynomials Using Descartes' Rule of Signs. Ph.D. thesis, Univ. of Saarland, May 2008.

[14] A. Eigenwillig, L. Kettner, W. Krandick, K. Mehlhorn, S. Schmitt, and N. Wolpert. A Descartes algorithm for polynomials with bit stream coefficients. In 8th Comp.Algebra in Sci.Computing (CASC), pp. 138–149. Springer, 2005. LNCS 3718.

[15] A. Eigenwillig, V. Sharma, and C. Yap. Almost tight complexity bounds for the Descartes method. ISSAC'06, pp. 71–78, 2006.

[16] J. Gerhard. Modular algorithms in symbolic summation and symbolic integration. LNCS 3218, Springer, 2004.

[17] J. Johnson. Algorithms for polynomial real root isolation. In B. Caviness and J. Johnson, eds., Quantifier Elimination and Cylindrical Algebraic Decomposition, pp. 269–299. Springer, 1998.

[18] N. Kamath. Subdivision algorithms for complex root isolation: Empirical comparisons. Master's thesis, Oxford Univ., Oxford Computing Lab, Aug. 2010.

[19] W. Krandick and G. E. Collins. An efficient algorithm for infallible polynomial complex root isolation. In ISSAC 97, pp. 189–194, 1992.

[20] W. Krandick and K. Mehlhorn. New bounds for the Descartes method. JSC, 41(1):49–66, 2006.

[21] T. Lickteig and M.-F. Roy. Sylvester-Habicht sequences and fast Cauchy index computation. J. Symbolic Comp., 31:315–341, 2001.

[22] L. Lin and C. Yap. Adaptive isotopic approximation of nonsingular curves: the parametrizability and non-local isotopy approach. In 25th SoCG, pp. 351–360, 2009. To appear, Special Issue of DCG.

[23] J. McNamee. A bibliography on roots of polynomials. J. Comput. Appl. Math., 47:391–394, 1993. Online at http://www.elsevier.com/homepage/sac/cam/mcnamee.

[24] K. Mehlhorn and R. Osbild and M. Sagraloff. A General Approach to the Analysis of Controlled Perturbation Algorithms. CGTA, 2011. To appear.

[25] K. Mehlhorn and M. Sagraloff. Isolating real roots of real polynomials. In ISSAC 09, 2009.

[26] D. P. Mitchell. Robust ray intersection with interval arithmetic. In Graphics Interface'90, pp. 68–74, 1990.

[27] R. E. Moore. Interval Analysis. Prentice Hall, Englewood Cliffs, NJ, 1966.

[28] B. Mourrain, F. Rouillier, and M.-F. Roy. The Bernstein basis and real root isolation. In Goodman, Pach, and Welzl, eds., Comb. and Comp. Geom., No. 52 MSRI Pub., pp. 459–478. Cambridge Press, 2005.

[29] B. Mourrain, M. N. Vrahatis, and J. C. Yakoubsohn. On the complexity of isolating real roots and computing with certainty the topological degree. J. Complexity, 18:612–640, 2002.

[30] V. Y. Pan. New techniques for approximating complex polynomial zeros. Proc. 5th ACM-SIAM Symp. on Discrete Algorithms (SODA94), pp. 260–270, 1994.

[31] V. Y. Pan. Optimal (up to polylog factors) sequential and parallel algorithms for approximating complex polynomial zeros. 27th STOC, pp. 741–750, 1995.

[32] V. Y. Pan. On approximating polynomial zeros: Modified quadtree (Weyl's) construction and improved Newton's iteration. RR No. 2894, INRIA Sophia-Antipolis, 1996.

[33] V. Y. Pan. Solving a polynomial equation: some history and recent progress. SIAM Review, 39(2):187–220, 1997.

[34] J. R. Pinkert. An exact method for finding the roots of a complex polynomial. ACM Trans. on Math. Software, 2:351–363, 1976.

[35] S. Plantinga and G. Vegter. Isotopic approximation of implicit curves and surfaces. In Eurographics Symp. on Geom. Processing, pp. 245–254, 2004. ACM Press.

[36] D. Reischert. Asymptotically fast computation of subresultants. In ISSAC 97, pp. 233–240, 1997.

[37] F. Rouillier and P. Zimmermann. Efficient isolation of [a] polynomial's real roots. J. Comput. and Applied Math., 162:33–50, 2004.

[38] M. Sagraloff. A general approach to isolating roots of a bitstream polynomial. Mathematics in Computer Science (MCS), 2011. To appear.

[39] M. Sagraloff and C. K. Yap. An efficient exact subdivision algorithm for isolating complex roots of a polynomial and its complexity analysis, July 2009. Full paper from http://cs.nyu.edu/exact/ or http://www.mpi-inf.mpg.de/˜msagralo/.

[40] A. Schönhage. The fundamental theorem of algebra in terms of computational complexity, 1982. Manuscript , Dept. of Math., U. of Tübingen. Updated 2004.

[41] V. Sharma. Complexity of real root isolation using continued fractions. Th. Comp. Sci., 409(2), 2008.

[42] S. Smale. The fundamental theorem of algebra and complexity theory. Bull. (N.S.) AMS, 4(1):1–36, 1981.

[43] G. Taubin. Rasterizing algebraic curves and surfaces. IEEE Comp. Graphics and Applic., 14(2):14–23, 1994.

[44] E. Tsigaridas and I. Emiris. On the complexity of real root isolation using continued fractions. Theor. Computer Science, 392:158–173, 2008.

[45] H. S. Wilf. A global bisection algorithm for computing the zeros of polynomials in the complex plane. J. ACM, 25(3):415–420, 1978.

[46] J.-C. Yakoubsohn. Numerical analysis of a bisection-exclusion method to find zeros of univariate analytic functions. J. of Complexity, 21:652–690, 2005.

[47] C. K. Yap. Fundamental Problems of Algorithmic Algebra. Oxford Univ. Press, 2000.

[48] J. Yu, C. Yap, Z. Du, S. Pion, and H. Bronnimann. Core 2: A library for Exact Numeric Computation in Geometry and Algebra. In 3rd ICMS, pp. 121–141. 2010. LNCS 6327.