Robust Approximate Zeros

Vikram Sharma, Zilin Du, and Chee K. Yap[1]

Courant Institute of Mathematical Sciences
New York University
New York, NY 10012, USA
sharma,zilin,yap@cs.nyu.edu

December 9, 2005

Abstract. Smale's notion of an approximate zero of an analytic function $f : \mathbb{C} \to \mathbb{C}$ is extended to take into account the errors incurred in the evaluation of the Newton operator. Call this stronger notion a **robust approximate zero**. We develop a corresponding **robust point estimate** for such zeros: we prove that if $z_0 \in \mathbb{C}$ satisfies $\alpha(f, z_0) < 0.02$ then z_0 is a robust approximate zero, with the associated zero z^* lying in the closed disc $\overline{B}(z_0, \frac{0.07}{\gamma(f,z_0)})$. Here $\alpha(f, z), \gamma(f, z)$ are standard functions in point estimates.
Suppose $f(z)$ is an L-bit integer square-free polynomial of degree d. Using our new algorithm, we can compute an n-bit absolute approximation of $z^* \in \mathbb{R}$ starting from a robust approximate zero z_0, in time $O(d \lg(dL) M(d^2(L + \lg d)) + dM(dn))$, where $M(n)$ is the complexity of multiplying n-bit integers.

1 Introduction

The Newton-Raphson method has been studied extensively in many settings. Given an analytic function $f : \mathbb{C} \to \mathbb{C}$ and a point $z_0 \in \mathbb{C}$, we consider the iteration $z_{i+1} = N_f(z_i)$ for $i \geq 0$ where $N_f(z) := z - f(z)/f'(z)$. This sequence is well-defined provided $f'(z_i) \neq 0$ for all $i \geq 0$. Kantorovich [KA64] developed convergence criteria for $(z_i)_{i \geq 0}$ that are applicable when points in an entire neighborhood of z_0 satisfy certain bounds. Yamamoto [Yam85,Yam86] gives sharp bounds of this sort. A basic technique in Kantorovich's approach is the use of majorant sequences. Unfortunately, Kantorovich's criteria are sometimes inconvenient to use. Smale [Sma86,BCSS98] developed convergence criteria that are applicable to a single point $z_0 \in \mathbb{C}$. Such criteria are called **point estimates**. Following [BCSS98, p. 155], call z_0 an **approximate zero** of $f(z)$ if the sequence $z_{i+1} = N_f(z_i)$ is well defined for all natural numbers i and there exists a root z^* of $f(z)$ such that for all $i \geq 0$,

$$|z_i - z^*| \leq 2^{1-2^i} |z_0 - z^*|;$$

[1] This research is supported by NSF Grant #CCF-043836. The work of Yap is partially carried out at the Korea Institute for Advanced Study (KIAS). To appear, Proc. 13th European Symposium on Algorithms, 2005.

z^* is called the associated zero. One such point estimate [DF95] says that if $\alpha(f, z_0) < 3 - 2\sqrt{2} \sim 0.17157$ then z_0 is an approximate zero. Here $\alpha(f, z)$ is an easily computed function defined in the next Section.

Variations, improvements and extensions are known. Kim [Kim86,Kim88] derived comparable point estimates for slightly different[1] notions of approximate zeros than the one defined above. Shub and Smale [SS85,SS93] and Malajovich [Mal93,Mal94] have developed such criteria for multivariate Newton methods in affine and projective spaces. Malajovich further extended this to pseudo Newton iteration, i.e., Newton iteration using the Moore-Penrose inverse. Wang and Zhao [DF95] improved Smale's point estimate using Kantorovich's approach, and extended it to the Weierstrass method [Dur60,Ker66]. Petkovic and others [PCT95,PHI98,Bat98] also obtained point estimates for the Weierstrass method.

The above results are developed in a setting where the operations are assumed to be exact, i.e., $N_f(z)$ can be computed without error. Even when this is possible, such as the case where z is rational and $f(z)$ an integer polynomial, it may be undesirable because of inefficiency. In practice, the z_i's will be represented by floating point numbers. In this paper, we assume the use of **bigfloats**, i.e., floating point numbers whose exponent and mantissas are arbitrary precision integers. Since $N_f(z)$ involves division, the use of approximation is essential in bigfloat computation. Indeed, Newton iteration is uniquely suited for approximation because of its known self-correcting behavior.

Bigfloat arithmetic is basically the multiple-precision arithmetic of Brent [Bre76a,Bre76b]. Here, the fundamental results have been achieved by Brent over 30 years ago. In particular, he shows that if $f(x)$ is the zero of a nonlinear equation $F(y) = x$, i.e., $F(f(c)) = c$, and if $F(y)$ can be evaluated to n-bits of relative precision in time $O(M(n)\phi(n))$ where $\phi(n)$ is a positive monotone increasing function, then $f(x)$ can be approximated to s-bits of relative precision in time $O(M(s)\phi(s))$. Here $M(n)$ is the complexity of multiplying two n-bit integers. An important restriction on Brent's complexity results is that the input x as well as all as intermediate approximate values must come from a bounded range. When $F(x)$ is a polynomial, we shall prove a global complexity bound for approximating $f(x)$ to absolute s-bits (Theorem 6).

Although there is a large literature on the error analysis of Newton iteration [Ypm83,Ypm84,Tis01,Hig96], these results do not address the point estimate setting. To our best knowledge, the only such results are by Malajovich [Mal94]. There are several differences between our work and Malajovich's.

– We focus on the univariate case while Malajovich address the more general case of multi-variate Newton. Consequently, our complexity bounds for the univariate case are much stronger than Malajovich's bound (when specialized to the univariate case). Indeed, Malajovich's complexity statements (see [Mal93, p. 2, Main Theorem and p. 79-80] or [Mal94, p. 2, and p. 8, Theorem 10]) contain terms that have no explicit complexity bounds.
– Malajovich assumes that each Newton step is computed to a fixed precision s. In contrast, we follow Brent's approach of doubling the precision at each

[1] For that matter, Smale has used more than one variant in his papers.

iteration. This has the advantage that the overall complexity is essentially determined by the last iteration step (see [Bre76a,Bre76b]).
- Finally, Malajovich's robust point estimate involves an extra parameter s (the precision of the bigfloat computations steps above). In particular, he shows that z_0 and s should satisfy $\alpha(f, z_0) < 0.05$ and $\gamma(f, z_0)s < 1/384$. Since s has to be at least the precision with which we want to approximate the zero, this criterion imposes additional constraints on the procedure for finding z_0. In contrast, our robust point estimate only requires $\alpha(f, z_0) \leq 0.02$, which is independent of the desired final precision. Our approach guarantees convergence to the root, unlike Malajovich's approach where the distance between the iterates and the root can only be bounded above by 2^{-6s}.

Contributions of this paper. Our main results are as follows:

1. We introduce a notion of **robust approximate zero** of an analytic function $f : \mathbb{C} \to \mathbb{C}$ and give a corresponding **robust point estimate** for a value $z_0 \in \mathbb{C}$ to be a such a zero. This is shown in Section 3.
2. In Section 4, we derive explicit bounds on the precision necessary to carry out the steps of a robust Newton iteration.
3. In Section 6, we give explicit complexity bounds for approximating a zero of a square-free integer polynomial starting from a robust approximate zero. This can be viewed as an extension of Brent's complexity bound (for algebraic roots) to the unbounded case.

Error Notation. We use two convenient notations for error bounds: we shall write

$$[z]_t \qquad \text{(resp., } \langle z \rangle_t) \tag{1}$$

for *any* relative (resp., absolute) t-bit approximation of z.

The following meta-notation is convenient: whenever we write "$z = \widetilde{z} \pm \varepsilon$" it means "$z = \widetilde{z} + \theta\varepsilon$" for some $\theta \in [-1, 1]$. More generally, the sequence "$\pm h$" is always to to be rewritten as "$+\theta h$" where θ is an implicit real variable satisfying $|\theta| \leq 1$. Unless the context dictates otherwise, different occurrences of $\pm$ will introduce distinct θ-variables. E.g., $x(1 \pm u)(1 \pm v)$ means that $x = x(1+\theta u)(1+\theta' v)$ for some $\theta, \theta' \in [-1, 1]$. The effect of this notation is to replace inequalities by equalities, and to remove the use of absolute values.

BigFloat Model of Computation. As in Brent [Bre76b,Bre76a], we use bigfloat numbers to approximate real or complex numbers. If f is an integer, write $\langle f \rangle$ for the value $f2^{-\lfloor \lg |f| \rfloor}$; thus $\langle f \rangle \in [1, 2)$. A (binary) **bigfloat** is a rational number of the form $x = n2^m$ where $n, m \in \mathbb{Z}$. We say x has **precision** t if $|n| < 2^t$.

We represent $x = n2^m$ by a pair (e, f) of binary integers where $e = m - \lfloor \lg n \rfloor$ is the **exponent** and $f = n$ represents the **fraction** $n2^{-\lfloor \lg n \rfloor}$. We write $\langle e, f \rangle$ for $f2^{e - \lfloor \lg f \rfloor}$. The **bit size** of $\langle e, f \rangle$ is the pair $(\lg(2 + |e|), \lg(2 + |f|))$.

Functions used in Error Analysis. Let $f : \mathbb{C} \to \mathbb{C}$ be any analytic function with a simple root at z^*. We may assume f is fixed in this paper and $N_f(z) = z - \frac{f(z)}{f'(z)}$ is its Newton iterator. For any $z \in \mathbb{C}$ we define the following functions:

- $\gamma(f, z) := \sup_{k \geq 2} \left| \frac{f^{(k)}(z)}{k! f'(z)} \right|^{1/(k-1)}$. We use γ_* for $\gamma(f, z^*)$, where z^* is a simple root of $f(z)$.
- $\beta(f, z) := \left| \frac{f(z)}{f'(z)} \right|$.
- $\alpha(f, z) := \beta(f, z)\gamma(f, z)$.
- $\psi(x) := 1 - 4x + 2x^2$. The roots of ψ are $(2 \pm \sqrt{2})/2$.
- $u(z, w) := \gamma(f, z)|z - w|$. For the special case where $z = z^*$, a root of f, we use the succinct notation u_w.

2 Weak and Strong Models of BigFloat Computation

Our algorithms use bigfloat numbers to approximate real or complex numbers. This arithmetic model was extensively studied by Brent [Bre76b,Bre76a].

We may distinguish two **modes** of using bigfloats. In the **weak (bigfloat) mode**, one chooses some arbitrary but fixed precision bound on all the bigfloats to be used in the computation. Thus a weak mode of computation can be regarded as a generalization of the IEEE model implemented in hardware in modern computers. Malajovich's algorithms operate in this weak mode. In the **strong (bigfloat) mode**, we use bigfloats without a priori precision bounds, and the algorithms can actively manage the precision of each computation step. Brent's complexity results, as well as ours, are achieved in this strong mode.

Consider the standard error analysis for Newton iteration according to which one can show ([BCSS98, p. 157, Prop. 1(a)]) that for some $z \in \mathbb{C}$ if $u_z < 1 - \frac{1}{\sqrt{2}}$ then we have:

$$|N_f(z) - z^*| \leq \frac{\gamma_*}{\psi(u_z)} |z - z^*|^2. \tag{2}$$

Now consider the case when the division in Newton iteration is carried to some fixed relative precision ε, i.e. define $\widetilde{N}_f(z) := z - \frac{f(z)}{f'(z)}(1 + \varepsilon)$. Then, assuming $u_z < 1 - \frac{1}{\sqrt{2}}$, we have the following:

$$\begin{aligned} |\widetilde{N}_f(z) - z^*| &\leq \tfrac{\gamma_*}{\psi(u_z)} |z - z^*|^2 + \tfrac{|z - z^*|(1 - u_z)\varepsilon}{\psi(u_z)} \\ &= \tfrac{|z - z^*|}{\psi(u_z)} (\varepsilon + u_z(1 - \varepsilon)). \end{aligned}$$

If $|\varepsilon| \leq |z - z^*|$ then we have

$$|\widetilde{N}_f(z) - z^*| \leq K|z - z^*|^2, \tag{3}$$

for some constant $K \geq 0$.

Consider the sequence of iterates $\widetilde{z}_i := \widetilde{N}(\widetilde{z}_{i-1})$, $i \in \mathbb{N}$, starting from z_0. For such a sequence we have the following:

Theorem 1. *If* $u_{z_0} \leq \frac{1}{4}$ *and* $\varepsilon \leq \frac{1}{8}$ *then the sequence* $(\widetilde{z}_i)$ *satisfies the following:*

$$|\widetilde{z}_i - z^*| \leq 2^{-i}|z_0 - z^*|,$$

i.e. they converge linearly to the root z^*.

Proof. The proof is by induction. The base case $i = 0$ is clearly true. Thus supposing the hypothesis holds for $i-1$ then we have $u_{\widetilde{z}_{i-1}} < u_{z_0} < 1 - \frac{1}{\sqrt{2}}$ and hence $\psi(u_{\widetilde{z}_{i-1}}) > \psi(u_{z_0}) \geq \frac{1}{2}$. Then from above we know that

$$|\widetilde{z}_i - z^*| \leq \frac{|\widetilde{z}_{i-1} - z^*|}{\psi(u_{z_0})}(\varepsilon + u_{z_0}(1-\varepsilon)).$$

Moreover, the restrictions on u_{z_0} and ε imply that $\frac{\varepsilon + u_{z_0}(1-\varepsilon)}{\psi(u_{z_0})} \leq \frac{1}{2}$. Along with the induction hypothesis we get the desired result. **Q.E.D.**

The import of (3) and the above theorem is the following: Suppose we are given a $n \in \mathbb{N}_{\geq 0}$, such that $2^{-n} \ll |\varepsilon|$, and we want to compute $\langle z^* \rangle_n$, then the sequence of iterates $\widetilde{z}_i$ have quadratic convergence till the time $|\widetilde{z}_i - z^*| \sim |\varepsilon|$ and subsequently only linear convergence; thus to guarantee quadratic convergence throughout we should choose $\varepsilon := 2^{-n}$, the final precision with which we want to approximate the root z^*. The following algorithm gives the implementation details of this idea.

```
Algorithm A'
    Input:     f(z) ∈ ℤ[z], n ≥ 0 and z_0 such that u_{z_0} ≤ 0.25.
    Output:    ⟨z*⟩_n, z* is the associated root of z_0
    1    Let z̃_0 := z_0 and i = 0.
    2    do
              δ := f(z)/f'(z) (1 ± 2^{-n}).
              z̃_{i+1} := z̃_i − δ; ⟨z̃_{i+1}⟩_n.
         while(|δ| ≤ 2^{-n} or δ ≠ 0)
    3    Return z̃_{i+1}.
```

3 Robust Newton Iteration

Let $f(z)$ and the corresponding $N_f(z)$ be as defined above. Given $z \in \mathbb{C}$ and $C \in \mathbb{R}$, let

$$N_{f,i,C}(z) := \langle N_f(z) \rangle_{2^i + C}\,, \tag{4}$$

Equation (4) uses our error notation of (1): this means $|N_{f,i,C}(z) - N_f(z)| \leq 2^{-2^i - C}$. For any $z_0 \in \mathbb{C}$ and $C \in \mathbb{R}$, a **robust iteration sequence of** z_0 **(relative to** C **and** f**)** is an infinite sequence

$$(\widetilde{z}_i)_{i \geq 0} \tag{5}$$

such that $\widetilde{z}_0 = z_0$, and for all $i \geq 1$,

$$\widetilde{z}_i = N_{f,i,C}(\widetilde{z}_{i-1}). \tag{6}$$

We assume each $\widetilde{z}_i \in \mathbb{C} \cup \{\infty\}$, and the relation (6) must be understood in the following way: if $\widetilde{z}_{i-1} = \infty$ or $\widetilde{z}_{i-1}$ is a critical point of f (i.e., $f'(\widetilde{z}_{i-1}) = 0$), then $\widetilde{z}_i = \infty$. We call the iteration sequence **finite** if each $\widetilde{z}_i \neq \infty$.

Our key definition is as follows: z_0 is a **robust approximate zero** of f if, there exists a zero z^* of f, such that for all C satisfying

$$2^{-C} \leq |z_0 - z^*|, \tag{7}$$

whenever $(\widetilde{z}_i)_{i\geq 0}$ is any robust iteration sequence of z_0 (relative to C and f), then the sequence is finite and for all $i \geq 0$,

$$|\widetilde{z}_i - z^*| \leq 2^{1-2^i}|z_0 - z^*|. \tag{8}$$

Call z^* the **associated zero** of z_0.

Smale et al. [BCSS98, p. 156, Thm. 1] have shown the following:

Proposition 1. *If z^* is a simple zero of $f(z)$, then $z_0 \in \mathbb{C}$ is an approximate zero of f with associated zero z^* if*

$$|z_0 - z^*| \leq \frac{3-\sqrt{7}}{2\gamma(f, z^*)}.$$

Here is our robust analogue:

Theorem 2. *If z^* is a simple zero of $f(z)$, then $z_0 \in \mathbb{C}$ is a robust approximate zero of f with associated zero z^* if*

$$|z_0 - z^*| \leq \frac{4-\sqrt{14}}{2\gamma(f, z^*)}.$$

Proof. Let $u_z = \gamma(f, z^*)|z - z^*|$ as above. We prove (8) by induction on $i \geq 0$. The result is clearly true for $i = 0$. Inductively, assume that $\widetilde{z}_i$ satisfies (8). Then $u_{\widetilde{z}_i} \leq 2^{1-2^i}u_{z_0}$. Since $u_{z_0} \leq \frac{4-\sqrt{14}}{2}$, it is smaller than the both roots of $\psi(x)$. Hence

$$\psi(u_{\widetilde{z}_i}) \geq \psi(u_{z_0}). \tag{9}$$

Thus,

$$\begin{aligned} |\widetilde{z}_{i+1} - z^*| &= |N_{f,i+1,C}(\widetilde{z}_i) - z^*| \\ &\leq |N_f(\widetilde{z}_i) - z^*| + 2^{-2^{i+1}}|z_0 - z^*| \text{ (from (7))}. \end{aligned}$$

From [BCSS98, p. 157, Prop. 1] we further get

$$\begin{aligned} |N_f(\widetilde{z}_i) - z^*| &\leq \tfrac{\gamma(f,z^*)}{\psi(u_{\widetilde{z}_i})}|\widetilde{z}_i - z^*|^2 \\ &\leq \tfrac{\gamma(f,z^*)}{\psi(u_{z_0})}|\widetilde{z}_i - z^*|^2 \text{ (from (9))}. \end{aligned}$$

From the inductive hypothesis we thus get,

$$\begin{aligned} |\widetilde{z}_{i+1} - z^*| &\leq \tfrac{\gamma(f,z^*)}{\psi(u_{z_0})}2^{2-2^{i+1}}|z_0 - z^*|^2 + 2^{-2^{i+1}}|z_0 - z^*| \\ &= \tfrac{u_{z_0}}{\psi(u_{z_0})}2^{2-2^{i+1}}|z_0 - z^*| + 2^{-2^{i+1}}|z_0 - z^*| \\ &\leq 2^{1-2^{i+1}}|z_0 - z^*|, \end{aligned}$$

since the assumption $u_{z_0} \leq \frac{4-\sqrt{14}}{2}$ implies $\frac{u_{z_0}}{\psi(u_{z_0})} \leq \frac{1}{4}$. **Q.E.D.**

Let the continuous function $\Gamma : S \to S$ be a contraction map on $S \subseteq \mathbb{C}$ with contraction constant $K < 1$; this implies that there is a unique fixed point $z^* \in S$ of Γ such that for all $z \in S$, the sequence $(\Gamma^n(z))_{n\geq 0}$ converges to z^*. We consider the inexact analogue of $\Gamma^n(z)$:

Lemma 1. *Let $\Gamma_{i,C}(z) := \langle \Gamma(z) \rangle_{i+C}$ (for $C \in \mathbb{R}$ and $i \geq 0$). If $C \geq -\lg(|z - z^*|)$, then the sequence*

$$\widetilde{z}_{i+1} := \Gamma_{i+1,C}(\widetilde{z}_i),$$

starting from $\widetilde{z}_0 := z_0$, converges to $z^* \in S$, assuming $\widetilde{z}_i \in S$ for each i.

Proof. Consider

$$\begin{aligned}
|\widetilde{z}_{i+1} - z^*| &\leq |\Gamma(\widetilde{z}_i) - z^*| + 2^{-(i+1)}2^{-C} \\
&\leq |\Gamma(\widetilde{z}_i) - z^*| + 2^{-(i+1)}|z_0 - z^*| && (2^{-C} \leq |z_0 - z^*|) \\
&\leq K|\widetilde{z}_i - z^*| + 2^{-(i+1)}|z_0 - z^*| && (\widetilde{z}_i \in S) \\
&\leq K^2|\widetilde{z}_{i-1} - z^*| + K2^{-i}|z_0 - z^*| + 2^{-(i+1)}|z_0 - z^*| \\
&\;\;\vdots \\
&\leq (K^{i+1} + K^i 2^{-1} + \cdots + K2^{-i} + 2^{-(i+1)})|z_0 - z^*| \\
&\leq (i+2)G^{i+1}|z_0 - z^*|,
\end{aligned}$$

where $G := \max\{K, 2^{-1}\} < 1$. Thus $|\widetilde{z}_i - z^*|$ tends to zero as i tends to infinity and hence the sequence $(\widetilde{z}_i)_{i\geq 0}$ tends to $z^* \in S$, the fixed point of Γ. **Q.E.D.**

It is clear that the above lemma holds even if $\Gamma_{i,C}(z)$ is defined as $\langle \Gamma(z) \rangle_{2^i+C}$. The following shows that under suitable restrictions on z_0 the robust iteration sequence defined in (6) converges to a root z^* of f. Let $\overline{B}(z, R)$ denote the closed disc with center $z \in \mathbb{C}$ and radius R.

Lemma 2. *Suppose there exist constants α_0, u_0 and $C_0 := \frac{2(\alpha_0+u_0)}{\psi(u_0)^2}$ which satisfy the following criteria:*

1. $0 \leq u_0 < 1 - 1/\sqrt{2}$,
2. $C_0 < \frac{3}{4}$,
3. $\alpha_0 \leq (\frac{3}{4} - C_0)u_0$, *and*
4. $\frac{u_0}{\psi(u_0)(1-u_0)} \leq \frac{4-\sqrt{14}}{2}$.

If $z_0 \in \mathbb{C}$ is such that $\alpha(f, z_0) < \alpha_0$ we have the following:
(a) N_f is a contraction map on $\overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$ with contraction constant C_0.
(b) z_0 is a robust approximate zero of f, the associated zero $z^* \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$.

Proof. Part (a) is from [BCSS98, p. 164, Cor. 2]. To show part (b), consider the robust iteration sequence $(\widetilde{z}_i)_{i\geq 0}$ defined in (6). We show by induction on $i \geq 0$ that $\widetilde{z}_i \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$; then applying Lemma 1 for $\Gamma := N_f$, we know that there exists a root $z^* \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$ of $f(z)$ to which the sequence $(\widetilde{z}_i)$ converges. The base case $i = 0$ follows since $z_0 \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$. Inductively suppose

$$\widetilde{z}_{i-1} \in \overline{B}(z_0, \frac{u_0}{\gamma(f, z_0)}), \tag{10}$$

then we want

$$\begin{array}{rll} & |\widetilde{z}_i - z_0| & \leq \frac{u_0}{\gamma(f,z_0)} \\ \Leftarrow & |N_f(\widetilde{z}_{i-1}) - z_0| + 2^{-2^i} 2^{-C} & \leq \frac{u_0}{\gamma(f,z_0)} \\ \Leftarrow & |N_f(\widetilde{z}_{i-1}) - z_0| + 2^{-2^i} |z_0 - z^*| & \leq \frac{u_0}{\gamma(f,z_0)} \\ \Leftarrow & |N_f(\widetilde{z}_{i-1}) - N_f(z_0)| + |N_f(z_0) - z_0| + 2^{-2^i} |z_0 - z^*| & \leq \frac{u_0}{\gamma(f,z_0)}. \end{array}$$

Using the contraction property of part (a), the above statement follows if

$$C_0 |\widetilde{z}_{i-1} - z_0| + \beta(f, z_0) + 2^{-2^i} |z_0 - z^*| \leq \frac{u_0}{\gamma(f, z_0)},$$

which holds if

$$\beta(f, z_0) \leq (1 - C_0 - 2^{-2^i}) \frac{u_0}{\gamma(f, z_0)},$$

since $\widetilde{z}_{i-1}, z^* \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$. But $\alpha(f, z_0) = \gamma(f, z_0)\beta(f, z_0)$, thus the above statement follows if $\alpha(f, z_0) \leq (\frac{3}{4} - C_0)u_0$, since $i \geq 1$. This is true since by assumption $C_0 < \frac{3}{4}$ and $\alpha(f, z_0) < \alpha_0 \leq (\frac{3}{4} - C_0)u_0$.

From Thm. 2 we know that any $z \in \overline{B}(z^*, \frac{4-\sqrt{14}}{2\gamma(f,z^*)})$ is a robust approximate zero. Thus z_0 satisfying the condition $\alpha(f, z_0) < \alpha_0$ is a robust approximate zero if

$$\begin{array}{rll} & |z_0 - z^*| & \leq \frac{4-\sqrt{14}}{2\gamma(f,z^*)} \\ \Leftarrow & \frac{u_0}{\gamma(f,z_0)} & \leq \frac{4-\sqrt{14}}{2\gamma(f,z^*)} \\ \Leftarrow & \frac{u_0}{\psi(u_0)(1-u_0)} & \leq \frac{4-\sqrt{14}}{2}, \end{array}$$

where the last step follows from [BCSS98, p. 160, Prop. 3], $u(z_0, z^*) \leq u_0$ whence $\psi(u(z_0, z^*)) \leq \psi(u_0)$; $\psi(x)$ is monotonically decreasing for $x < 1 - \frac{1}{\sqrt{2}}$. **Q.E.D.**

One choice of constants that satisfy the above criteria is $u_0 = 0.07$ and $\alpha_0 = 0.02$.

Theorem 3 (Point estimate for robust approximate zero). *Any* $z_0 \in \mathbb{C}$ *for which* $\alpha(f, z_0) < 0.02$ *is a robust approximate zero of* f*, with the associated zero* $z^* \in \overline{B}(z_0, \frac{0.07}{\gamma(f,z_0)})$.

4 Approximate Evaluation of Newton Iterator

In this section we determine the absolute precision with which to evaluate f and f', and the relative precision with which to carry out the division at each iteration step; let these be e_i, E_i, and ϱ_i, respectively.

We will have recourse to the next two lemmas which apply to an analytic f.

Lemma 3. *Let $u = \gamma(f,z)|z-w| < 1 - \frac{1}{\sqrt{2}}$. Then we have*

$$\frac{\psi(u)}{(1-u)^2} \leq \frac{|f'(w)|}{|f'(z)|} \leq \frac{1}{(1-u)^2}.$$

Proof. The lower bound is proved in [BCSS98, p. 156]. For the upper bound:

$$\begin{aligned} \frac{|f'(z)|}{|f'(w)|} &= \left| \frac{1}{1+\sum_{k=2}^{\infty} \frac{f^{(k)}(z)}{f'(z)(k-1)!}(w-z)^{k-1}} \right| \\ &\geq \frac{1}{1+\sum_{k=2}^{\infty} k \left| \frac{f^{(k)}(z)}{f'(z)k!}(w-z)^{k-1} \right|} \\ &\geq \frac{1}{1+\sum_{k=2}^{\infty} k u^{k-1}} \\ &= (1-u)^2, \end{aligned}$$

since $u < 1$. **Q.E.D.**

Lemma 4. *Let z be such that $u_z = \gamma(f,z^*)|z-z^*| < 1$, where z^* is a simple root of f. Then*

$$\frac{|z-z^*|(1-2u_z)}{1-u_z} \leq \left| \frac{f(z)}{f'(z^*)} \right| \leq \frac{|z-z^*|}{1-u_z}.$$

Proof. For the upper bound, see [BCSS98, p. 161, Lem. 4(b)]. For the lower bound,

$$\begin{aligned} \left| \frac{f(z)}{f'(z^*)} \right| &\geq |z-z^*| - \sum_{j=2}^{\infty} \left| \frac{f^{(j)}(z^*)(z-z^*)^j}{f'(z^*)j!} \right| \\ &\geq |z-z^*| \frac{1-2u_z}{1-u_z}. \end{aligned}$$

Q.E.D.

Let $z_0 \in \mathbb{C}$ such that $\alpha(f,z_0) < 0.02$; from Thm. 3 we know that z_0 is a robust approximate zero with an associated root z^* and $u_{z_0} \leq \frac{4-\sqrt{14}}{2}$, and hence $\psi(u_{z_0}) \geq \frac{1}{2}$. Let $(\widetilde{z}_i)_{i\geq 0}$ be a robust approximate sequence starting from z_0, relative to a constant C satisfying (7). Writing $\delta_i = \widetilde{z}_i - z^*$, we know $|\delta_i| \leq 2^{-2^i+1}|\delta_0|$.

The main result of this section is:

Theorem 4. *Let $z_0 \in \mathbb{C}$ and $f : \mathbb{C} \to \mathbb{C}$ be an analytic function. To compute $\langle N_f(\widetilde{z}_i) \rangle_{2^{i+1}+C}$ it suffices to*
(i) evaluate $f(\widetilde{z}_i)$ to $(\kappa + 2^{i+1} + 4 + C)$ absolute bits,
(ii) evaluate $f'(\widetilde{z}_i)$ to $(\kappa' + 2^i + 3 + C)$ absolute bits,
(iii) and perform the division in N_f to $(\kappa'' + 2^i + 1 + C)$ relative bits.
Here, $\kappa \geq -\lg|f'(z_0)|$, $\kappa' \geq -\lg|f'(z_0)|\gamma(f,z_0)$ and $\kappa'' \geq 3 - \lg\gamma(f,z_0)$.

Proof. Let

$$\widetilde{z}_{i+1} = \widetilde{z}_i - \frac{f(\widetilde{z}_i)+e_i}{f'(\widetilde{z}_i)+E_i}(1+\varrho_i).$$

But we want $\widetilde{z}_{i+1} = \langle N_f(\widetilde{z}_i)\rangle_{2^{i+1}+C}$, where C is an upper bound on $-\lg|z_0 - z^*|$. Thus these errors should satisfy

$$\left|\frac{f(\widetilde{z}_i)+e_i}{f'(\widetilde{z}_i)+E_i}(1+\varrho_i) - \frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}\right| \le 2^{-2^{i+1}}2^{-C}.$$

This follows if

$$\left|\frac{f(\widetilde{z}_i)+e_i}{f'(\widetilde{z}_i)+E_i} - \frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}\right| \le 2^{-2^{i+1}-1}2^{-C} \tag{11}$$

and

$$\left|\frac{f(\widetilde{z}_i)+e_i}{f'(\widetilde{z}_i)+E_i}\varrho_i\right| \le 2^{-2^{i+1}-1}2^{-C} \tag{12}$$

are true. The first of these two can be reduced to

$$\begin{aligned} &|f'(\widetilde{z}_i)e_i - f(\widetilde{z}_i)E_i| \le 2^{-2^{i+1}-1}2^{-C}|(f'(\widetilde{z}_i)+E_i)f'(\widetilde{z}_i)| \\ \Leftarrow\ &|f'(\widetilde{z}_i)e_i| + |E_i f(\widetilde{z}_i)| \le 2^{-2^{i+1}-1}2^{-C}|(f'(\widetilde{z}_i)+E_i)f'(\widetilde{z}_i)|. \end{aligned}$$

The above statement follows if

$$|e_i| \le 2^{-2^{i+1}-2-C}|f'(\widetilde{z}_i)+E_i| \tag{13}$$

and

$$|E_i| \le 2^{-2^{i+1}-2-C}|f'(\widetilde{z}_i)+E_i|\frac{|f'(\widetilde{z}_i)|}{|f(\widetilde{z}_i)|}. \tag{14}$$

From Lemma 3 and Lemma 4, (14) follows if:

$$\begin{array}{lll}
|E_i| & \le 2^{-2^{i+1}-2-C}|f'(\widetilde{z}_i)+E_i|\frac{\psi(u_{\widetilde{z}_i})}{|\delta_i|(1-u_{\widetilde{z}_i})} & \\
\Leftarrow |E_i| & \le 2^{-2^{i+1}-3-C}|f'(\widetilde{z}_i)+E_i|\frac{1}{|\delta_i|} & (\frac{1}{2} \le \psi(u_{z_0}) \le \psi(u_{\widetilde{z}_i})) \\
\Leftarrow |E_i| & \le 2^{-2^i-4-C}|f'(\widetilde{z}_i)+E_i|\frac{1}{|\delta_0|} & (\text{since } |\delta_i| \le 2^{-2^i+1}|\delta_0|) \\
\Leftarrow |E_i|(1+\frac{2^{-2^i-4-C}}{|\delta_0|}) & \le \frac{2^{-2^i-4-C}}{|\delta_0|}|f'(\widetilde{z}_i)| & \\
\Leftarrow 2|E_i| & \le \frac{2^{-2^i-4-C}}{|\delta_0|}|f'(\widetilde{z}_i)| & (\text{from (7)}) \\
\Leftarrow |E_i| & \le \frac{2^{-2^i-6-C}}{|\delta_0|}|f'(z_0)| & (\text{from Lemma 3}) \\
\Leftarrow |E_i| & \le 2^{-2^i-3-C}|f'(z_0)|\gamma(f,z_0), & (*)
\end{array}$$

since $z^* \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$ where $u_0 = 0.07$.

Now consider (13). The inequality there follows if

$$\begin{array}{ll}
|e_i| \le 2^{-2^{i+1}-2-C}(|f'(\widetilde{z}_i)| - |E_i|) & \\
\Leftarrow |e_i| \le 2^{-2^{i+1}-2-C}(|f'(\widetilde{z}_i)| - 2^{-2^i-3-C}|f'(z_0)|\gamma(f,z_0)) & (\text{from } (*)) \\
\Leftarrow |e_i| \le 2^{-2^{i+1}-3-C}(|f'(z_0)| - 2^{-2^i-2-C}|f'(z_0)|\gamma(f,z_0)) & (\text{from Lemma 3}) \\
\Leftarrow |e_i| \le 2^{-2^{i+1}-3-C}|f'(z_0)|(1 - 2^{-2^i-2-C}\gamma(f,z_0)) & \\
\Leftarrow |e_i| \le 2^{-2^{i+1}-3-C}|f'(z_0)|(1 - 2^{-2^i-2}|z_0 - z^*|\gamma(f,z_0)) & (\text{from (7)}) \\
\Leftarrow |e_i| \le 2^{-2^{i+1}-3-C}|f'(z_0)|(1 - \frac{u_0}{8}). &
\end{array}$$

The last step follows because $z^* \in \overline{B}(z_0, \frac{u_0}{\gamma(f,z_0)})$ where $u_0 = 0.07$. Thus

$$|e_i| \le 2^{-2^{i+1}-4-C}|f'(z_0)|.$$

Now we consider (12) . Choose $|\varrho_i| \le 2^{-2^i-C-\kappa'-1}$, which reduces the inequality to

$$\begin{aligned} 2^{-\kappa''}|f(\widetilde{z}_i)+e_i| &\le 2^{-2^i}|f'(\widetilde{z}_i)+E_i| \\ \Leftarrow 2^{-\kappa''}|f(\widetilde{z}_i)|+2^{-\kappa''}|e_i| &\le 2^{-2^i}|f'(\widetilde{z}_i)+E_i|. \end{aligned}$$

This follows if

$$2^{-\kappa''}|f(\widetilde{z}_i)| \le 2^{-2^i-1}|f'(\widetilde{z}_i)+E_i| \tag{15}$$

and

$$2^{-\kappa''}|e_i| \le 2^{-2^i-1}|f'(\widetilde{z}_i)+E_i|. \tag{16}$$

From (13) it is clear that (16) is true for any κ''. Thus we focus on (15), which follows if

$$\begin{aligned} 2^{-\kappa''}|f(\widetilde{z}_i)|+2^{-2^i-1}|E_i| &\le 2^{-2^i-1}|f'(\widetilde{z}_i)| \\ \Leftarrow 2^{-\kappa''}|f'(z^*)||\delta_i|+2^{-2^{i+1}-4-C}|f'(z_0)|\gamma(f,z_0) &\le 2^{-2^i-2}|f'(z_0)| \text{ (from Lemma 4 and } (*)) \\ \Leftarrow 2^{-\kappa''}|f'(z_0)|2^{-2^i+1}|\delta_0|+2^{-2^{i+1}-4-C}|f'(z_0)|\gamma(f,z_0) &\le 2^{-2^i-2}|f'(z_0)| \text{ (from Lemma 3 and (8))} \\ \Leftarrow 2^{-\kappa''+1}|\delta_0|+2^{-2^i-4-C}\gamma(f,z_0) &\le 2^{-2}. \end{aligned}$$

Since $2^{-C} \le |\delta_0| \le \frac{u_0}{\gamma(f,z_0)}$ and $u_0 = 0.07$ the above inequality follows if

$$\frac{2^{-\kappa''}}{\gamma(f,z_0)} + 2^{-2^i-4} \le 2^{-2}$$

which clearly follows if

$$2^{-\kappa''} \le \frac{\gamma(f,z_0)}{8}.$$

Q.E.D.

Since we want to choose minimum values for κ, κ' and κ'', we let $\kappa = -\lg|f'(z_0)|$, $\kappa' = -\lg|f'(z_0)|\gamma(f,z_0)$ and $\kappa'' = 3 - \lg\gamma(f,z_0)$.

5 Estimating the Distance between an Approximate Zero and its Associated Root

Let z_0 be a robust approximate zero with the associated zero z^*. To construct a robust iteration sequence (6) converging to z^*, we need to determine a $C \in \mathbb{Z}$ satisfying (7), or equivalently, $C \ge -\lg|z_0 - z^*|$. In this section we compute tight bounds on $|z_0 - z^*|$ where z_0 is an approximate zero (not just a robust approximate zero). We assume that $\alpha(f,z_0) < 0.03$. Then from [BCSS98, p. 160, Thm. 2] and [BCSS98, p. 166, Remark 6], we know that z_0 is an approximate zero satisfying Prop. 1.

We can use an inequality from Kalantari [Kal05]: for any $z_0 \in \mathbb{C}$,

$$|z_0 - z^*| \geq \frac{1}{2\gamma_2(f, z_0)} \tag{17}$$

where

$$\gamma_2(f, z_0) := \sup_{k\geq 1} \left| \frac{f^{(k)}(z_0)}{k! f(z_0)} \right|^{1/k}. \tag{18}$$

Hence it suffices to choose any C satisfying

$$C \geq 1 + \lg \gamma_2(f, z_0). \tag{19}$$

The Kalantari function $\gamma_2(f, z_0)$ is easily approximated in practice.

Since C controls the number of bits used in our robust iteration, it is desirable for C to be as small as possible. We pose the problem of computing C up to some additive constant $K > 0$. More precisely, compute any C which satisfies

$$0 \leq C + \lg |z_0 - z^*| \leq K. \tag{20}$$

Kalantari's estimate (19) is not known to satisfy (20). In short, we want a tight estimate of the distance $|z_0 - z^*|$ between z_0 and its associated zero z^*. We could use Turan's proximity test [Pan97] to approximate the minimum and maximum distances from any complex number to the zeros of a polynomial $f(z)$ within a constant factor, at the cost of $O(d \lg d)$ arithmetic operations, where $d = \deg f(z)$. We do not use this test because it is limited to polynomials, and , more importantly, it does not leverage the fact that z_0 is an approximate zero.

Our solution exploits the property of approximate zeros, based on a tight relationship between $\delta := \left| \frac{f(z_0)}{f'(z_0)} \right|$ $(= \beta(z_0))$ and $|z_0 - z^*|$:

Lemma 5. *Let $z \in \mathbb{C}$ satisfy $u = \gamma(f, z^*)|z - z^*| < 1 - \frac{1}{\sqrt{2}}$, where z^* is a simple root of f. Then*

$$|z - z^*|(1 - 2u)(1 - u) \leq \left| \frac{f(z)}{f'(z)} \right| \leq \frac{|z - z^*|(1 - u)}{\psi(u)}.$$

Proof. The upper bound follows since

$$\left| \frac{f(z)}{f'(z)} \right| = \left| \frac{f(z)}{f'(z^*)} \right| \left| \frac{1}{\frac{f'(z)}{f'(z^*)}} \right| \leq \frac{|z - z^*|(1 - u)}{\psi(u)},$$

the last step uses the upper bound in Lemma 4 and the lower bound in Lemma 3. Similarly, the lower bound follows from the lower bound in Lemma 4 and the upper bound in Lemma 3. **Q.E.D.**

We now describe our algorithm:

ALGORITHM D
INPUT: f, z_0 where $\alpha(f, z_0) < 0.03$
OUTPUT: n such that $|f(z_0)/f'(z_0)| = C' \cdot 2^{-n}$
for some $0.5 \leq C' \leq 3$.
$n = 0$.
Do
$w \leftarrow \left\langle \frac{f(z_0)}{f'(z_0)} \right\rangle_n$
$n \leftarrow n + 1$
while ($|w| \leq 2^{-n+1}$)
Return $(n - 1)$

Note that the value n returned by the algorithm satisfies the inequalities

$$\left\langle \frac{f(z_0)}{f'(z_0)} \right\rangle_n \leq 2^{-n+1} \tag{21}$$

and

$$\left\langle \frac{f(z_0)}{f'(z_0)} \right\rangle_{n+1} > 2^{-n}. \tag{22}$$

The correctness of this Algorithm follows from the following lemma:

Lemma 6. *If n satisfies (21) and (22) then*

$$2^{-n-1} < \delta \leq 3 \cdot 2^{-n}.$$

Proof. The inequality (21) implies $\left|\frac{f(z_0)}{f'(z_0)}\right| - 2^{-n} \leq 2 \cdot 2^{-n}$ or $\delta \leq 3 \cdot 2^{-n}$. The inequality (22) implies $\left|\frac{f(z_0)}{f'(z_0)}\right| + 2^{-n-1} > 2 \cdot 2^{-n}$ or $\delta > 2^{-n-1}$. **Q.E.D.**

Note that $\alpha(f, z_0) < 0.03$ implies $u_{z_0} \leq \frac{3-\sqrt{7}}{2}$. Hence $\psi(u_{z_0}) \geq \frac{1}{2}$, and the above lemma gives us

$$\frac{\delta}{2} \leq |z_0 - z^*| \leq 2\delta. \tag{23}$$

We then conclude that Algorithm D produces the necessary constant C for robust iteration:

Lemma 7. *Let $C := n + 2$, where $n - 1$ is the value returned by Algorithm D on an approximate zero z_0, $\alpha(f, z_0) < 0.03$, with z^* as the associated root. Then*

$$2^{-C} \leq |z_0 - z^*| \leq 6.2^{-C+2}.$$

Basically, Algorithm D is converting absolute precision into relative precision. Algorithm D takes $(-\lg \delta) + O(1)$ steps of evaluation. But using the geometric searching method in [AKY04], we can further reduce the number of evaluation steps to $2 \lg \lg(1/\delta) + O(1)$. For the purposes of this exposition we present the simpler version, however the complexity result below is based upon the geometric search method.

6 Complexity of Approximating a Real Zero of a Real Polynomial

Our results in the previous sections assume $f(z)$ is an analytic function. In this section we focus on the special case when $f(z) = \sum_{i=0}^{d} a_i z^i \in \mathbb{R}[z]$ is a square-free polynomial that satisfies the following properties:

- $a_d = 1$ and in general $|a_i| \le 2^L - 1$, for some $L \ge 0$. Thus the exponents of a_i are bounded by L.
- The coefficients of $f(z)$ are represented as "blackbox" numbers that output a desired approximation. More precisely, we assume that given a blackbox number α, we can compute $[\alpha]_n$, a bigfloat, in time $B(n)$. For instance, if α is a bigfloat then we know from the appendix that $B(n) = C_0(n + \lg(2 + |\lg \alpha|))$, where $C_0 > 0$ is independen of α; in case α is an algebraic number $B(n) = O(M(n))$ – in fact this is what Brent has shown – however, the constant in O depends upon α.

Thus the problem is: given a bigfloat z_0, such that $\alpha(f, z_0) < 0.02$, compute a n-bit absolute approximation to $z^* \in \mathbb{R}$, the associated root of z_0.

Our assumptions imply that z_0 is a robust approximate zero. Thus starting from z_0 we can apply robust Newton iteration till $\widetilde{z}_i$ does not satisfy $|\widetilde{z}_i - z^*| \le 2^{-n}$; from (8) we know that this is guaranteed once $i \ge \lg(n + 1 + \lg|z_0 - z^*|)$. From Cauchy's bound [Yap00, p. 148] we know that $|z^*| \le 2^L$ and without loss of generality we assume that $|z_0| \le 2^L$; thus the exponent of z_0 is bounded by L. The assumption implies that $|z_0 - z^*| \le 2^{L+1}$ and hence we require at most $\lg(n + L + 2)$ steps of Newton iteration. The complete algorithm which computes $\langle z^* \rangle_n$, given z_0, is as follows:

ALGORITHM B
INPUT: $f(z) \in \mathbb{R}[z]$, $n \ge 0$ and z_0 where $\alpha(f, z_0) < 0.02$
OUTPUT: $\langle z^* \rangle_n$, z^* is the associated root of z_0
1 Compute C satisfying (7) using Algorithm D above.
Let $\widetilde{z}_0 := z_0$.
2 For $i = 1, \ldots, \lg(n + L + 2)$ do the following:
$x := \langle f(\widetilde{z}_i) \rangle_{2^{i+1}+C+4+\kappa}$.
$y := \langle f'(\widetilde{z}_i) \rangle_{2^i+3+C+\kappa'}$.
$\widetilde{z}_{i+1} := \widetilde{z}_i - \left[\frac{x}{y}\right]_{2^i+1+C+\kappa''}$.
3 Return $\widetilde{z}_i$.

To bound the complexity of the above algorithm we need to bound the complexity of Algorithm D.

Lemma 8. *Let the bigfloat z_0 be an approximate zero such that $\alpha(f, z_0) < 0.03$, and $|f(z_0)| \ge 2^{-\Delta}$, for some Δ. Then the geometric version of Algorithm D has complexity*

$$O(dM(\Delta) + d\lg(dL + d\lg d)M(dL + d\lg d)) + O(d\lg(dL + \Delta)B(dL + d\lg d + \Delta))$$

Proof. Let $\delta := \left|\frac{f(z_0)}{f'(z_0)}\right|$. As stated earlier the number of steps of the algorithm are $O(\lg\lg\frac{1}{\delta})$. From (23) and (17) we know that $\delta \ge \frac{1}{4\gamma_2(f, z_0)}$. Thus to bound $\lg\frac{1}{\delta}$

we compute an upper bound on $\gamma_2(f, z_0)$. Since we have assumed that $|z_0| \le 2^L$, we can show that $|f^{(k)}(z_0)| \le \frac{d!}{(d-k)!} 2^{L(d+1)}$. Thus from (18) we get:

$$\begin{aligned}
\gamma_2(f, z_0) &\le \sup_{k\ge 1} \left(\frac{d!}{k!(d-k)!} 2^{L(d+1)} 2^{\Delta} \right)^{1/k} \\
&\le 2^{L(d+1)+\Delta} \sup_{k\ge 1} \binom{d}{k}^{1/k} \\
&\le 2^{L(d+1)+\Delta} d.
\end{aligned}$$

Thus $\lg \frac{1}{\delta} = O(dL + \Delta)$; since $C = O(|\lg \delta|)$ we know that $C = O(dL + \Delta)$ and the same bound holds for κ in Thm. 4.

Consider the i'th iteration where we have to compute $\left\langle \frac{f(z_0)}{f'(z_0)} \right\rangle_{2^i}$; which can be achieved ([Yap04, Lem. 11,p. 24]) if we compute $\langle f(z_0)\rangle_{k_1}$ and $\langle f'(z_0)\rangle_{k_2}$, where $k_1 \ge 2^i + 2 - \lg |f'(z_0)|$ and $k_2 \ge 2^i + 2 - 2\lg |f'(z_0)| + \lg |f(z_0)|$, and do the division to $2^i + 1$ relative bits. But since $\psi(z_0) \ge \frac{1}{2}$, from Lemma 3 we know that $|f'(z_0)| \ge \frac{|f'(z^*)|}{2}$; from [Yap00, p. 183] we also know that $|f'(z^*)| \ge \frac{1}{d^{d-1.5} 2^{Ld}}$, thus $-\lg |f'(z_0)| = O(dL + d \lg d)$. Since $|z_0| \le 2^L$, we can also show that $\lg |f(z_0)| = O(dL)$. It is not hard to see that the complexity of the i'th iteration is dominated by the complexity of computing $\langle f(z_0)\rangle_{k_1}$, $k_1 = 2^i + O(dL + d \lg d)$, which is (see Appendix)

$$O(dM(2^i + dL + d \lg d) + dB(2^i + dL + d \lg d));$$

the complexity of evaluating $\langle f'(z)\rangle_{k_2}$ is asymptotically the same and the division takes $O(M(2^i + dL + d \lg d))$. Hence the total complexity of Algorithm D is

$$\sum_{i=0}^{O(\lg(dL+\Delta))} O(dM(2^i + dL + d \lg d) + dB(2^i + dL + d \lg d)), \tag{24}$$

that is

$$O(dM(dL+\Delta)) + \lg(dL+d \lg d) O(dM(dL+d \lg d)) + O(d \lg(dL+\Delta) B(dL+d \lg d+\Delta)),$$

which can be simplified to the result mentioned in the lemma. **Q.E.D.**

Now we can bound the running time of Algorithm B:

Theorem 5. *Let $f(z) = \sum_{i=0}^{d} a_i z^i \in \mathbb{R}[z]$ be a monic square-free polynomial such that $|a_i| \le 2^L - 1$. Suppose we are given a bigfloat z_0 such that $\alpha(f, z_0) \le 0.02$ and $|f(z_0)| \ge 2^{-\Delta}$; thus z_0 is a robust approximate zero with an associated root z^*. Then we can compute $\langle z^*\rangle_n$ in time*

$$\begin{aligned}
&O[dM(n) + dM(\Delta) + d \lg(dL + d \lg d) M(dL + d \lg d)] + \\
&O[d \lg(n + L) B(n + dL + d \lg d) + d \lg(dL + \Delta) B(dL + d \lg d + \Delta)].
\end{aligned} \tag{25}$$

If d, L are bounded then the complexity is $O(M(n))$.

Proof. We bound the running time of step 2 in the algorithm. It is clear that the dominating complexity in step 2 is to compute

$$\langle f(\widetilde{z}_i)\rangle_{2^{i+1}+C+4+\kappa}$$

at each i. From Lemma 8 we know that both C and κ are $O(dL + d\lg d)$. Let $\mathrm{sep}(f, z^*)$ denote the distance from z^* to the nearest root of f apart from itself. Since z_0 is a robust approximate zero, from Lemma 2(a) we know that $\widetilde{z}_{i-1} \in \overline{B}(z_0, \frac{0.07}{\gamma(f,z_0)})$. Thus

$$\begin{aligned}
|\widetilde{z}_i| &\le |z_0| + \left|\frac{0.07}{\gamma(f,z_0)}\right| \\
&\le |z_0| + \left|\frac{0.07}{\psi(u_{z_0})(1-u_{z_0})\gamma(f,z^*)}\right| && \text{([BCSS98, p. 160, Proposition 3])} \\
&\le |z_0| + \frac{1}{\gamma(f,z^*)} && \text{(since } u_{z_0} \le \tfrac{4-\sqrt{14}}{2},\ \psi(u_{z_0}), (1-u_{z_0}) \ge \tfrac{1}{2}) \\
&\le |z_0| + 2\mathrm{sep}(f,z^*) && \text{(from [Kal05, p. 846], } \mathrm{sep}(f,z^*) \ge \tfrac{1}{2\gamma(f,z^*)}) \\
&\le |z_0| + 2^{L+2} && \text{(from Cauchy's root bound).}
\end{aligned}$$

Since by assumption $|z_0| \le 2^L$, we have $|\widetilde{z}_i| \le 2^{L+3}$; thus the exponent of $\widetilde{z}_i$, for all i's, is bounded by $L + 3$.

Thus, from the appendix we get the complexity of computing the desired absolute approximation to $f(\widetilde{z}_i)$ as $O(dM(2^{i+1} + K) + dB(2^{i+1} + K))$, where $K = O(dL + d\lg d)$, and hence the total complexity of step 2 is

$$\begin{aligned}
\sum_{i=0}^{\lg(n+L+2)} O(dM(2^{i+1}+K) + dB(2^{i+1}+K)) &= O(dM(n+L+2)) + \lg K O(dM(K)) \\
&\quad + \lg(n+L+2) O(dB(n+dL+d\lg d)),
\end{aligned} \tag{26}$$

or $O[dM(n) + d\lg K M(K)] + O(d\lg(n+L)B(n+dL+d\lg d))$.

Combining this result with Lemma 8 we get the overall complexity of Algorithm B as stated in (25). **Q.E.D.**

This result may be regarded as a generalization of Brent's bounded precision bound [Bre76a, Lem. 3.1]. For the special case when the coefficients of $f(z)$ are real algebraic numbers this lemma states that $B(n) = O(M(n))$. This observation along with (24) and (26) gives us the following:

COROLLARY 9 *Let $f(z) = \sum_{i=0}^{d} a_i z^i$ be a monic square-free polynomial whose coefficients are real algebraic numbers satisfying $|a_i| \le 2^L - 1$. Given z_0 satisfying $\alpha(f, z_0) < 0.02$ with associated root z^*, for any n, we can compute $\langle z^*\rangle_n$ in time*

$$O[dM(n) + dM(\Delta) + d\lg(dL + d\lg d)M(dL + d\lg d)]$$

where Δ satisfies $|f(z_0)| \ge 2^{-\Delta}$.

6.1 Complexity in case of an integer polynomial

If $f(z)$ is an integer polynomial it is possible to simplify the results in Thm. 4 since now $f(z)$ can be evaluated *exactly* at any bigfloat z, and hence we can compute a C satisfying a tight inequality like that in Lemma 7. In particular, we compute $\left[\left|\frac{f(z_0)}{f'(z_0)}\right|\right]_2$ and use an upper bound, say by two extra bits, on its most significant bit to determine C; in fact, assuming that $\alpha(f, z_0) < 0.02$, which implies $u_{\widetilde{z}_0} \leq \frac{4-\sqrt{14}}{2}$ and hence $\psi(u_{\widetilde{z}_0}) \geq \frac{1}{2}$, we can show that

$$\frac{|z_0 - z^*|}{8} \leq 2^{-C} \leq |z_0 - z^*|. \tag{27}$$

Using this result we get the following simplification to Thm. 4.

Lemma 10. *Let $f(z) \in \mathbb{Z}[z]$ and z_0 be any bigfloat such that $\alpha(f, z_0) \leq 0.02$. To compute $\langle N_f(\widetilde{z}_i)\rangle_{2^{i+1}+C}$, $i \geq 0$, it suffices to perform the division in N_f to $2^i + 6$ relative bits.*

Proof. Since we can now evaluate $f(\widetilde{z}_i)$ and $f'(\widetilde{z}_i)$ exactly, we define

$$\widetilde{z}_{i+1} := \widetilde{z}_i - \frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}(1 + \varrho_i),$$

where $\varrho_i \in \mathbb{R}$. But we want

$$|\widetilde{z}_{i+1} - N_f(\widetilde{z}_i)| \leq 2^{-2^{i+1}-C-1},$$

which follows if $\left|\frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}\varrho_i\right| \leq 2^{-2^{i+1}-C}$ and we compute $\langle \widetilde{z}_i\rangle_{2^{i+1}+C+1}$; the second computation would be useful only in the first step since in the subsequent steps the precision only increases. Applying Lemma 5, (27) and (8), along with the observation that $\psi(u_{\widetilde{z}_i}) \geq \psi(u_{\widetilde{z}_0})$, we get $|\varrho_i| \leq 2^{-2^i-6}$ as a sufficient criterion to guarantee $\langle N_f(\widetilde{z}_i)\rangle_{2^{i+1}+C}$. **Q.E.D.**

To make the algorithm more adaptive we will use a better stopping criterion rather than using the for loop. Let $\delta_i := \left[\frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}\right]_{2^i+6}$.

Lemma 11. *Let z_0 be such that $\alpha(f, z_0) \leq 0.02$. If $|\delta_i| \leq 2^{-(n+2)}$ then $|\widetilde{z}_i - z^*| \leq 2^{-n}$.*

Proof. Let $\delta = \frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}$. Then from the definition of δ_i we know that $|\delta| \leq 2|\delta_i|$. From lemma 5 we also know that $|\widetilde{z}_i - z^*| \leq \frac{|\delta|}{(1-2u_{\widetilde{z}_i})(1-u_{\widetilde{z}_i})}$; but the right hand side is less than $2|\delta|$ since $\alpha(f, z_0) \leq 0.02$ implies $u_{\widetilde{z}_i} \leq u_{z_0} \leq \frac{4-\sqrt{14}}{2} < 0.13$. Thus $|\widetilde{z}_i - z^*| \leq 4|\delta_i| \leq 2^{-n}$. **Q.E.D.**

Now we can simplify Algorithm B to the following.

Algorithm A

Input: $f(z) \in \mathbb{Z}[z]$, $n \geq 0$ and z_0 where $\alpha(f, z_0) < 0.02$

Output: $\langle z^* \rangle_n$, z^* is the associated root of z_0

1 Compute C, an upper bound by two extra bits on the precision of $\left[\left|\frac{f(z_0)}{f'(z_0)}\right|\right]_2$.
Let $\widetilde{z}_0 := \langle z_0 \rangle_{C+3}$, $i = 0$.

2 do

$\delta_i := \left[\frac{f(\widetilde{z}_i)}{f'(\widetilde{z}_i)}\right]_{2^i+6}$.

$\widetilde{z}_{i+1} := \widetilde{z}_i - \delta_i$.

$i := i + 1$.

while($\delta_i \neq 0$ and $|\delta_i| \geq 2^{-n-2}$).

3 Return $\widetilde{z}_i$.

Let us assume that the coefficients of $f(z)$ are L-bit integers. Let L_i denote the bit size of $\widetilde{z}_i$. We can further assume that L_0, the bit size of the starting point z_0, is $O(dL + d \lg d)$ since once we have isolated the root from its nearest critical point, which is at least $\frac{\mathrm{sep}(f,z^*)}{d}$ from z^*, we are sure that Newton iteration will converge from z_0. This also implies that $C = O(dL + d \lg d)$.

It is clear that the most expensive step in Algorithm B is computing $f(\widetilde{z}_i)$. From the Appendix we know that this can be done in $O(dM(dL_i + L))$. Moreover, the bit size of the result, that is L_{i+1}, is $O(2^i + C)$; since by assumption C already dominates L we do not consider it. Thus the overall complexity of the algorithm is

$$\sum_{i=0}^{\lg(n+L+2)} O(dM(d(2^i + C))) = O(d \lg C M(dC) + dM(dn)).$$

Since $C = O(dL + d \lg d)$, the overall complexity is $O(d \lg(dL) M(d^2(L + \lg d)) + dM(dn))$.

7 Experiments

We compare the running times of the following two implementations of Newton's method:

– **Full precision version** where all the operations are done to a fixed precision, namely, the final precision with which we want to approximate the root. This is the implementation of Algorithm A' in Section 2. It is essentially Malajovich's algorithm.
– **Robust version**. This is the implementation of Algorithm A in the previous Section.

For each of the ten polynomials below we will approximate a fixed root of that polynomial to precision $n = 1000, 5000, 10000, 20000$ using the above two versions. The starting point is chosen such that empirically it is a robust approximate zero and at the same time guarantees that the full version converges quadratically. We did not apply the point estimate mentioned in Thm. 3 because these polynomials have very large $\gamma(f, z)$, thus forcing a very accurate precision for the starting point. This shows that there is still a gap between the theory of point estimates and their use in practise.

The initial approximation, around 20 digits of accuracy, for each of the roots was obtained using the Mpsolve package of Bini and Fiorentino [BF00]; the polynomials are also borrowed from the same resource.

Table 1 shows the time in seconds taken by the two versions. Note that for wilk40 the robust version always take the same time; this is because due to rounding it reaches the exact root. The last column shows the relative running times of the two algorithms: theoretically, this should grow as $\lg(n)$. Although this ratio is increasing with n, it seems to be somewhat smaller than expected.

The implementations were done using the Bigfloat package of Core Library [KLPY99]. The code is available under progs/newton directory in the file newton.h; the sequence of tests is available in the file test.h in the same place. Our implementation exploits a particular property of the BigFloat package in the Core Library, viz., the ring operations $(+, -, \times)$ are error-free. This is in contrast to certain bigfloat packages like `gmp`'s `mpfr` where each operation is guaranteed up to some arbitrarily specified precision. The workstation is Sun Blade 1000, 2x750 MHz UltraSPARC III CPU, 8 MB Cache each, with 2 GB of RAM.

8 Conclusion

The key contribution of this paper is the development of the concept of robust approximate zero and robust point estimates. We improve on Malajovich's work by obtaining explicit complexity bounds and a stronger point estimate in the univariate case.

We plan to extend the above work in the following directions: to multi-variate Newton iteration, and to multiple zeros. For the latter problem, Yakoubsohn [Yak03] has obtained results under the exact arithmetic setting. Brent has given the complexity of approximating a simple zero of a non-linear equation in the bounded bigfloat setting. We also plan to extend this to the unbounded robust setting.

Acknowledgements The authors would like to thank an anonymous referee for meticulous and invaluable feedback.

References

[AKY04] Tetsuo Asano, David Kirkpatrick, and Chee Yap. Pseudo approximation algorithms, with applications to optimal motion planning. *Discrete and Computational Geometry*, 31(1):139–171, 2004. Special Conference Issue from 18th ACM Symp. of Comput. Geom., 2002.

[Bat98] Prashant Batra. Improvement of a convergence condition for the Durand-Kerner iteration. *J. of Comp. and Appl. Math.*, 96:117–125, 1998.

[BCSS98] Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. *Complexity and Real Computation.* Springer-Verlag, New York, 1998.

[BF00] Dario Andrea Bini and Giuseppe Fiorention. *Numerical Computation of Polynomial Roots Using MPSolve Version 2.2.* Dipartimento di Matematica, Universitaá di Pisa, Via Bonarroti 2, 56127

Pisa, January 2000. Manual for the Mpsolve package. Available at ftp://ftp.dm.unipi.it/pub/mpsolve/MPSolve-2.2.tgz.

[BFMS99] C. Burnikel, R. Fleischer, K. Mehlhorn, and S. Schirra. Exact efficient geometric computation made easy. In *Proc. 15th ACM Symp. Comp. Geom.*, pages 341–450, New York, 1999. ACM Press.

[Bre76a] Richard P. Brent. Fast multiple-precision evaluation of elementary functions. *J. of the ACM*, 23:242–251, 1976.

[Bre76b] Richard P. Brent. Multiple-precision zero-finding methods and the complexity of elementary function evaluation. In J. F. Traub, editor, *Proc. Symp. on Analytic Computational Complexity*, pages 151–176. Academic Press, 1976.

[DF95] Wang Deren and Zhao Fengguang. The theory of Smale's point estimation and its applications. *J. of Comp. and Appl. Math.*, 60:253–269, 1995.

[Dur60] E. Durand. *Solutions Numériques des Équations Algébriques, Tome I: Equations du Type F(x) = 0*. Racines d'un Polyn̂ome, Masson, Paris, 1960.

[Hig96] Nicholas J. Higham. *Accuracy and stability of numerical algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, 1996.

[KA64] L.V. Kantorovich and G.P. Akilov. *Functional Analysis in Normed Spaces*. New York, MacMillan, 1964.

[Kal05] Bahman Kalantari. An infinite family of bounds on zeros of analytic functions and relationship to Smale's bound. *Mathematics of Computation*, 74(250):841–852, 2005.

[Ker66] I.O. Kerner. Ein Gesamtschrittverfahren zur Berechnung der Nullstellen von Polynomen. *Numer. Math.*, 8:290–294, 1966.

[Kim86] Myong-Hi Kim. *Computational Complexity of the Euler Type Algorithms for the Roots of polynomials*. PhD thesis, City University of New York, January 1986.

[Kim88] Myong-Hi Kim. On approximate zeroes and root finding algorithms for a complex polynomial. *Math. Comp.*, 51:707–719, 1988.

[KLPY99] V. Karamcheti, C. Li, I. Pechtchanski, and C. Yap. A Core library for robust numerical and geometric computation. In *15th ACM Symp. Computational Geometry*, pages 351–359, 1999.

[Mal93] Gegorio Malajovich. *On the complexity of path-following Newton algorithms for solving systems of polynomial equations with integer coefficients*. PhD thesis, Berkeley, 1993.

[Mal94] Gregorio Malajovich. On generalized Newton algorithms: Quadratic convergence, path-following and error analysis. *Theoretical Computer Science*, 133:65–84, 1994.

[MS01] Kurt Mehlhorn and Stefan Schirra. Exact computation with `leda_real` – theory and geometric applications. In G. Alefeld, J. Rohn, S. Rump, and T. Yamamoto, editors, *Symbolic Algebraic Methods and Verification Methods*, pages 163–172, Vienna, 2001. Springer-Verlag.

[Pan97] Victor Y. Pan. Solving a polynomial equation: some history and recent progress. *SIAM Review*, 39(2):187–220, 1997.

[PCT95] Miodrag S. Petković, Carsten Carstensen, and Miroslav Trajkovíc. Weierstrass formula and zero-finding methods. *Numer. Math.*, 69:353–372, 1995.

[PHI98] Miodrag S. Petković, Dorde Herceg, and Snežana Ilić. Safe convergence of simultaneous methods for polynomial zeros. *Numerical Algorithms*, 17:313–331, 1998.

[Sch80] A. Schönhage. Storage modification machines. *SIAM J. Computing*, 9:490–508, 1980.

[Sma86] S. Smale. Newton's method estimates from data at one point. In R. Ewing, K. Gross, and C. Martin, editors, *The Merging of Disciplines: New Directions in Pure, Applied, and Computational Mathematics*. Springer-Verlag, 1986.

[SS85] Mike Shub and Steven Smale. Computational Complexity: On the Geometry of Polynomials and a Theory of Cost. I. *Annales Scientifiques De L'É.N.S.*, 4(18):107–142, 1985.

[SS93] Mike Shub and Steven Smale. Complexity of Bezout's Theorem I: Geometric aspects. *J. of Amer. Math. Soc.*, 6(2):459–501, 1993.

[Tis01] Françoise Tisseur. Newton's method in floating point arithmetic and iterative refinement of generalized eigenvalue problems. *SIAM J. on Matrix Anal. and Appl.*, 22(4):1038–1057, 2001.

[Yak03] Jean-Claude Yakoubsohn. Numerical Elimination, Newton Method and Multiple Roots. In Frédéric Chyzak, editor, *Algorithms Seminar, 2001-2002*, number 5003 in Rapport de recherche, pages 49–54. INRIA, Nov. 2003.

[Yam85] T. Yamamoto. A unified derivation of several error bounds for Newton's process. *Journal of Comp. and Appl. Mathematics*, 12&13:179–191, 1985.

[Yam86] T. Yamamoto. Error bounds for Newton's method under the Kantorovich assumptions. In R. Ewing, K. Gross, and C. Martin, editors, *The Merging of Disciplines: New Directions in Pure, Applied, and Computational Mathematics*. Springer-Verlag, 1986.

[Yap00] Chee K. Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, 2000.

[Yap04] Chee K. Yap. On guaranteed accuracy computation. In Falai Chen and Dongming Wang, editors, *Geometric Computation*, chapter 12, pages 322–373. World Scientific Publishing Co., Singapore, 2004.

[YD95] Chee K. Yap and Thomas Dubé. The exact computation paradigm. In D.-Z. Du and F. K. Hwang, editors, *Computing in Euclidean Geometry*, pages 452–492. World Scientific Press, Singapore, 2nd edition, 1995.

[Ypm83] T.J. Ypma. The effect of rounding errors on Newton-like methods. *IMA J. of Numerical Analysis*, 3:109–118, 1983.

[Ypm84] T.J. Ypma. Local convergence of inexact Newton methods. *SIAM J. of Numer. Anal.*, 21(3):583–590, 1984.

Appendix A: Big Float Computation

We review some basic facts about bigfloats The name "bigfloat" serves to distinguish this from the usual programming concept of "floats" which has fixed precision. For a survey on bigfloat computation, see [YD95]. Our bigfloat model is essential Brent's multi-precision arithmetic model.

Consider a bigfloat number

$$x = \langle e_x, f_x \rangle = f_x 2^{e_x - \lfloor \lg |f_x| \rfloor} = \langle f_x \rangle 2^{e_x}.$$

A restriction in Brent's complexity model is that all bigfloats x used in a given computation are **bounded**, i.e., $e_x = O(1)$ for any bigfloat $x = \langle e_x, f_x \rangle$. We are however interested in unbounded bigfloats. For unbounded bigfloats, we found it to be essential to adopt a more flexible computational model based on the Pointer machines of Schönhage [Sch80] rather than Turing machines.

Theorem 6. *Let $x = \langle e_x, f_x \rangle, y = \langle e_y, f_y \rangle$ be unbounded bigfloats, and n be a positive natural number. Also, $f_x f_y \neq 0$.*

1. *We can compute $[x]_n$ in $C_0(n + \lg(2 + |e_x|))$ time.*
2. *We can compute $[xy]_n$ in $C_0(M(n) + \lg(2 + |e_x e_y|))$ time.*
3. *We can compute $[x + y]_n$ in $C_0(n + \lg(2 + |e_x e_y|))$ time provided $xy \geq 0$ or $|x| > 2|y|$ or $|x| < |y|/2$. In general, computing $[x + y]_n$ can be done in time $O(\lg(2 + |f_x f_y e_x e_y|))$.*
4. *An analogous statement holds for $[x - y]_n$, where we replace $xy \geq 0$ by $xy \leq 0$.*

C_0 is a constant that is independent of x and y.

Proof.

1. **Truncation**: To compute $[x]_n$ in $O(n+\lg(2+|e_x|))$ time on a pointer model: given the input n in binary and $x = \langle e_x, f_x \rangle$, we simply treat n as a binary counter and count down to 0, it is well-known that this takes $O(n)$ steps; simultaneously, we output the most significant n-bits of f_x. In other words, this complexity does not depend on $\lg |f_x|$. We can also output e_x in $O(\lg(2+|e_x|))$ time.
2. **Addition**: We can easily check that $xy \geq 0$ and $|x| > 2|y|$ or $2|x| \leq |y|$ in $O(2 + \lg |e_x e_y|))$ time. If so, we carry out
   (a) Compute $[x]_{n+2}$ and $[y]_{n+2}$. This takes time $O(n + \lg(2 + |e_x e_y|))$.
   (b) Compare e_x and e_y. This takes $O(\lg(2 + |e_x e_y|))$. Let $e_x \geq e_y$.
   (c) Compute $e_x - e_y$. This takes $O(\lg(2 + |e_x e_y|))$. Shift the decimal point of y by $\min\{e_x - e_y, n\}$ bits; this takes $O(n)$.
   (d) Add the two fractional parts; this takes $O(n)$. Since by assumption either both the fractional parts have the same sign, in which case no cancellation occurs, and if not then the most significant bit of $x + y$ is to the right of x or y, depending upon whether $|x| \geq 2|y|$ or vice versa.

Thus the total complexity is $O(n + \lg(2 + |e_x e_y|))$.
In general, i.e., when the above assumptions fail, the complexity will be $O(\lg|f_x f_y| + \lg(2 + |e_x e_y|))$. this is because the mantissas may be equal and catastrophic cancellation may occur.
3. **Subtraction**: Has the same complexity as addition, except that the assumption $xy \geq 0$ should be $xy \leq 0$.
4. **Multiplication**: We carry these steps.
   (a) Compute $[x]_{n+2}$ and $[y]_{n+2}$.
   (b) Multiply the fractional parts of the truncations.
   (c) Add the two exponents.
   Thus the total complexity is $O(M(n) + \lg|e_x e_y|)$.

Q.E.D.

It is clear from the above arguments that the constants in the above results are independent of the choice of x, y.

Evaluating a polynomial to absolute precision. Given $f(x) = \sum_{i=0}^{d} a_i x^i$, $a_i \in \mathbb{R}$, and $s \in \mathbb{Z}$, let $\widetilde{f}$ be the result of evaluating $f(x)$ at $x \in \mathbb{R}$ using Horner's rule where each operation is carried out with *relative* precision s. Given $n \in \mathbb{Z}$, we want to determine $s = s(n)$ such that $\langle f(x) \rangle_n = \widetilde{f}$. Here we assume that the coefficients and x are blackbox numbers (see Section 6) and can be truncated in time $B(n)$. Let e_i be such that $2^{e_i} \leq a_i < 2^{e_i+1}$ and e_x such that

$$2^{e_x} \leq x < 2^{e_x+1}; \tag{28}$$

let

$$e := \max_{\{i=0,\ldots,d\}} e_i. \tag{29}$$

From Higham [Hig96, p. 105], we have

$$\begin{aligned} |\widetilde{f} - f(x)| &\leq \gamma_{2d} \sum_{i=0}^{d} |a_i||x|^i \\ &\leq \gamma_{2d} 2^{e+1} \sum_{i=0}^{d} |x|^i. \end{aligned}$$

where $\gamma_k := \frac{k2^{-s}}{1-k2^{-s}}$. To choose s such that the right hand side in the above equation is less than 2^{-n}, we consider two cases:

1. When $e_x \geq 0$, i.e., $|x| \geq 1$. Then we have

$$\begin{array}{lll} \gamma_{2d} \sum_{i=0}^{d} |a_i||x|^i & \leq 2^{-n} & \\ \Leftarrow 2^{e+3+d(e_x+1)} d(d+1) 2^{-s} & \leq 2^{-n} & \text{if } s \geq 2 + \lg d \\ \Leftarrow s & \geq n + e + d(e_x + 3) + 4 \ (*). & \end{array}$$

2. When $e_x < 0$, i.e, $|x| \leq 1$. Then

$$\begin{array}{lll} \gamma_{2d} \sum_{i=0}^{d} |a_i||x|^i & \leq 2^{-n} & \\ \Leftarrow \gamma_{2d} 2^{e+1}(d+1) & \leq 2^{-n} & \\ \Leftarrow 2^{e+3}(d+1)2^{-s} & \leq 2^{-n} & \text{if } s \geq 2 + \lg(1+d) \\ \Leftarrow s & \geq n + e + \lg d + 4 \ (**). & \end{array}$$

The complexity of evaluation is evident from the following:

1. Compute $[a_i]_s$ for $i = 0, \ldots, d$ and $[x]_s$. This takes $O(dB(s))$ by our assumption on a_i and x.
2. Let us represent Horner's evaluation recursively as $P_{i-1} = P_i x + a_{i-1}$, where $P_d := a_d$. The most expensive step of this computation is the multiplication of P_i and x, both of which have at most s bits of relative accuracy, which costs $O(M(s) + \lg|e_{P_i} e_x|)$. It is straightforward that $|e_{P_{i-1}}| \leq |e_{P_i}| + |e_x| + |e|$, which yields that $|e_{P_i}| \leq (d-i+1)(|e_x|+|e|)$. Thus the complexity of the i^{th} step is $O(M(s) + \lg[|e_x|(d-i+1)(|e_x|+|e|)])$. Summing this for $i = 0, \ldots, d$ we get the complexity to be $O(dM(s) + d\lg d + d\lg(|e_x|^2 + |ee_x|))$, which dominates the complexity of the first step.

Lemma 12. *The complexity of evaluating $f(x) \in \mathbb{R}[x]$ at a point $x \in \mathbb{R}$ to absolute precision n is*

$$O(d[M(n + e + d\max\{1, e_x\}) + B(n + e + d\max\{1, e_x\})]),$$

where e and e_x are defined in (28) and (29) respectively.

NOTE:
(1) The complexity of computing $f'(x)$ is the same, since only the bit-size of the coefficients is increased to $e + \lg d$, which can be subsumed by $e + d\max\{1, e_x\}$.
(2) For the purposes of getting a priori bounds on the complexity, we need only upper bounds on e and e_x.

Evaluating an integer polynomial at a bigfloat.

Lemma 13. *Given $f(x) = \sum_{i=0}^{d} a_i x^i$, where $a_i \in \mathbb{Z}$ are L-bit integers, and x a bigfloat, we can evaluate $f(x)$ in time $O(dM(dL' + dL))$ where L' is the bit size of x.*

Proof. There are d algebraic operations involved in Horner's method. At each such operation the bit size increases by a factor of $L' + 2L$ and hence the overall complexity is $\sum_{i=1}^{d} O(M(i(L' + 2L))) = O(dM(dL' + dL))$. **Q.E.D.**

Polynomial	Initial Root	n	Time by Robust (t)	Time by Full (T)	Ratio (T/t)
chebyshev40	1000	0.08	0.26		
	5000	1.32	3.00		
	10000	3.67	10.82		
	20000	9.59	35.00		
chebyshev80	1000	0.33	0.75		
	5000	5.27	15.01		
	10000	14.83	47.00		
	20000	38.22	150.00		
hermite40	1000	0.09	0.17		
	5000	1.31	3.00		
	10000	3.69	11.33		
	20000	9.07	35.00		
hermite80	1000	0.33	0.70		
	5000	5.13	14.77		
	10000	14.58	47.00		
	20000	38.91	148.00		
laguerre40	1000	0.10	0.19		
	5000	1.32	3.00		
	10000	3.55	11.43		
	20000	9.41	36.00		
laguerre80	1000	0.33	0.72		
	5000	5.28	14.50		
	10000	14.45	47.00		
	20000	39.10	143.00		
mand31	1000	0.06	0.11		
	5000	0.77	2.00		
	10000	2.18	6.88		
	20000	5.62	20.00		
mand63	1000	0.21	0.46		
	5000	3.51	9.09		
	10000	9.24	28.00		
	20000	24.23	88.00		
wilk40	1000	0.03	0.38		
	5000	0.03	5.30		
	10000	0.03	15.97		
	20000	0.03	46.00		

Table 1.