Testing polynomial primality with pseudozeros

Stef Graillat, Philippe Langlois

Laboratoire MANO, Université de Perpignan
52, avenue de Villeneuve
66860 Perpignan Cedex

Abstract

When polynomials have limited accuracy coefficients or are computed in finite precision, classical algebraic problems such that GCD, primality, divisibility have to be redefined. Such approximate algebraic problems are still challenging open questions in the symbolic computation community. In this paper, we show how a numerical tool, the pseudozero set, may provide solutions to some approximate algebraic problems. We propose a graphical answer to test polynomial primality.

Key words: pseudozeros, approximate GCD, approximate primality.
1991 MSC: 65F35, 68W30

1 Introduction

An open challenge in symbolic computation is to define reliable finite precision computations to solve exact problems. Except in Mathematics, polynomials have coefficients known to a limited accuracy. Such uncertainty may come from measured or observed data or previous computations performed in finite precision, *i.e.* in floating point arithmetic. Polynomials we consider in this paper suffer from such an uncertainty. Algebraic computation with uncertain polynomials occur in robotic, CAGD [39], molecular biology, etc. Classical polynomial problems like GCD, divisor or primality have to be redefined to take into account the limited accuracy of the polynomial coefficients. Numerous papers considering these questions are proposed in the symbolic computation literature, *e.g.* [1,2,38].

Email addresses: `graillat@univ-perp.fr` (Stef Graillat), `langlois@univ-perp.fr` (Philippe Langlois).

On the other hand, the scientific computing community is used to manage the effects of finite precision computation to the stability of numerical algorithm and the accuracy of computed results, particularly in numerical linear algebra. Very less results are proposed for problems that involve polynomial computation. This gap may be justified since polynomial problems can be transformed in linear algebra problems (thanks to *ad hoc* matrices such that companion matrices, Sylvester matrix, . . .) *in real arithmetic*.

The reliability of this transformation when problems are solved with finite precision arithmetic is not clear and motivates the kind of approach we describe herein. Two well known papers illustrate this difficulty for polynomial zerofinding : Toh and Trefethen report in [42, p.404] that "finding zeros via eigenvalues of companion matrices, the method used by the MATLAB `roots` command, is a stable algorithm" whereas Edelman and Murakami "construct examples for which a small componentwise relative backward error is neither predicted nor obtained in practice" [7, p.763]. This paper aims to illustrate that specific tools for polynomials exist and help to understand and solve some polynomial problems. We focus here the set of pseudozeros and some application of this tool introduced by Mosier [28] and, in our point of view, that have not been exploited enough. Main published results discuss the equivalence between the pseudozero set and the pseudospectra of the companion matrix [42,7]. We propose to test the primality of two univariate polynomials having coefficients known to a limited accuracy with these pseudozeros.

The paper is organized as follows. Approximate polynomial problems are presented in Section 2. Next two Sections 3 and 4 are respectively devoted to polynomial pseudozeros and first applications. The test of the polynomial primality is proposed in Section 5.

2 Approximate GCD and primality

The classic definition of the polynomial GCD does not fit the finite precision field. For example, let p and q be two unitary polynomials such that $\deg p > 1$ and p divides q. It yields that $\gcd(p, q) = p$. Nevertheless, for any number $\varepsilon > 0$, we have $\gcd(p + \varepsilon, q) = 1$: any small perturbation of the polynomial p critically affects the GCD. Since polynomial GCD does not depend continuously of the perturbation of its coefficients, computing a polynomial GCD is an ill-posed problem in the sense of Hadamard.

We have the same difficulty with the easiest problem of *primality*. The following example from [2] is significant. As soon as the coefficients of the following polynomials

$$p(z) = (z - \frac{1}{3})(z - \frac{5}{3}) = z^2 - 2z + \frac{5}{9}, \quad \text{and} \quad q(z) = z - \frac{1}{3}.$$

are represented with binary floating point numbers, p et q become *coprime* whereas they have a common root in exact arithmetic. On the contrary, polynomials

$$p(z) = 50z - 7 \quad \text{and} \quad q(z) = z - \frac{1}{7},$$

are coprime in exact arithmetic whereas they share a common root if we seek it with two decimal digit numbers since $1/7 = 0.14285714$ and $7/50 = 0.14$.

The first definition of an *approximate GCD* is proposed by Schönhage [38] in 1985 but is not appropriated since its computation involved coefficients known up to an arbitrary precision. The following more standard definition introduces ε-divisors and the ε-GCD.

Definition 1 *Given two polynomials p and q of degree respectively n and m, and ε a positive real, an ε-*divisor *(or* approximate divisor*) of p and q is every divisor of perturbed polynomials $\widehat{p}$ and $\widehat{q}$ satisfying $\|p - \widehat{p}\| \leqslant \varepsilon$, $\|q - \widehat{q}\| \leqslant \varepsilon$ and $\deg(p - \widehat{p}) \leqslant n$, $\deg(q - \widehat{q}) \leqslant m$.*
*An ε-*GCD *of p and q is an ε-divisor of highest degree.*

One can verify that the ε-GCD is not unique in general.

Polynomials are defined either by its roots or its coefficients. The study of the ε-GCD when the roots are perturbed is proposed by Pan in [32,34]. When coefficients suffer from uncertainty, two approaches have been proposed.

- The Euclidean algorithm is modified by changing the tests and the stopping criterion [11,23]. This kind of algorithms is efficient since it is similar to the classic Euclidean algorithm but does not work in finite precision. Alas, it only yields an ε-divisor and no, in general, an ε-GCD. Since the degree of the GCD equals the deficient rank of the Sylvester matrix, [6,11,10] compute the numerical deficiency rank that corresponds to the degree of an ε-GCD. This computation is reliably performed in finite precision using the singular value decomposition (SVD).
- The second approach formulates the ε-GCD problem as an optimization problem [24,25]. Karmarkar and Lakshman compute an ε-GCD together with the perturbed polynomial. A part of the algorithm can be executed in finite precision. The drawback of this algorithm is that it is exponential in the degree of the GCD.

The corresponding *ε-primality* problem consists in proving whether ε-GCD(p, q) equals 1 or not.

Definition 2 *Let two polynomials p and q of degree respectively n and m and ε a positive real. Polynomials p and q are ε-coprime if ε-GCD$(p,q) = 1$.*

Of course, computing an ε-GCD and compare it to 1 suffers from a too expensive complexity. Beckermann and Labahn propose a new algorithm to deal with primality without computing an ε-GCD in [2].

Using the norm $\|p\| = \sum_j |p_j|$ defined on $\mathbb{C}[z]$ and

$$\|(p,q)\| = \max\{\|p\|, \|q\|\} = \max\{\sum |p_i|, \sum |q_j|\},$$

they define $\epsilon(p,q)$ to be the minimum distance between two given polynomials and not coprime ones ; that is for $p, q \in \mathbb{C}[z]$,

$$\epsilon(p,q) = \inf\{\|(p - \widehat{p}, q - \widehat{q})\| : (\widehat{p}, \widehat{q}) \text{ have a common root and } \deg \widehat{p} \leqslant n, \deg \widehat{q} \leqslant m\}.$$

Beckermann and Labahn compute a lower bound for $\epsilon(p,q)$ and so guaranty a primality neighborhood around p and q. This algorithm costs $\mathcal{O}((n+m)^2)$ operations but does not always yield sharp bound.

We propose to answer to the polynomial ε-primality problem thanks to the set of *pseudozeros*. In the next section, we propose a uniform presentation of pseudozeros that gather definition and properties from Mosier [28], Trefethen and Toh [42], Chatelin and Frayssé [4] and Stetter [41].

3 Definition and computation of the ε-pseudozero set

3.1 Definition of the ε-pseudozero set

$\mathbb{P}_n$ denotes the set of polynomials with complex coefficients and degree at most n. Let $p \in \mathbb{P}_n$ given by

$$p(z) = p_0 + p_1 z + \cdots + p_n z^n. \tag{1}$$

Representing p by the vector of its coefficients, we define a norm $\|\cdot\|$ on $\mathbb{P}_n$ as the norm on $\mathbb{C}^{n+1}$ of the vector of the polynomial coefficients.
For this norm, we define an ε-neighborhood of p to be the set of every polynomial of degree at most n, closed enough to p, that is,

$$N_\varepsilon(p) = \{\widehat{p} \in \mathbb{P}_n : \|p - \widehat{p}\| \leqslant \varepsilon\}. \tag{2}$$

Then the ε-pseudozero set is defined to gather all the zeros of the ε-neighborhood. A non constructive definition of this set is

$$Z_\varepsilon(p) = \{z \in \mathbb{C} : \widehat{p}(z) = 0 \text{ for } \widehat{p} \in N_\varepsilon(p)\}\,. \tag{3}$$

3.2 A computable form of the ε-pseudozero set

The following theorem prove that the ε-pseudozero set can be obtain as a level contour of an easily computable function.

Theorem 3 *The ε-pseudozero set satisfies*

$$Z_\varepsilon(p) = \left\{ z \in \mathbb{C} : |g(z)| = \frac{|p(z)|}{\|\underline{z}\|_*} \leqslant \varepsilon \right\}, \tag{4}$$

where $\underline{z} = (1, z, \ldots, z^n)$ and $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$.

PROOF. We remind that the dual norm $\|\cdot\|_*$ on $\mathbb{C}^{n+1}$ is defined by

$$\|x\|_* = \max_{z\neq 0} \frac{|z^t x|}{\|z\|} = \max_{\|z\|=1} |z^t x|.$$

If $z \in Z_\varepsilon(p)$ then it exists $\widehat{p} \in \mathbb{P}_n$ such that $\widehat{p}(z) = 0$ et $\|p - \widehat{p}\| \leqslant \varepsilon$. From Hölder's inequality $|x^t y| \leqslant \|x\|\|y\|_*$, we get

$$|p(z)| = |p(z) - \widehat{p}(z)| = |\sum_{i=0}^{n} (p_i - \widehat{p}_i) z^i| \leqslant \|p - \widehat{p}\|\|\underline{z}\|_*.$$

It follows $|p(z)| \leqslant \varepsilon\|\underline{z}\|_*$.

To prove the reciprocal, let $u \in \mathbb{C}$ be such that $|p(u)| \leqslant \varepsilon\|\underline{u}\|_*$. A classical result enable us to assert the existence of a vector $d = (d_i) \in \mathbb{C}^{n+1}$ with norm 1 satisfying $d^*\underline{u} = \|\underline{u}\|_*$ ([17, p.107] or [22, p. 278]). This vector d is called the dual vector of $\underline{u}$. Let us introduce the polynomials r and p_u defined by

$$r(z) = \sum_{k=0}^{n} r_k z^k \text{ with } r_k = d_k, \tag{5}$$

$$p_u(z) = p(z) - \frac{p(u)}{r(u)} r(z). \tag{6}$$

Such polynomial p_u is the nearest polynomial of p, in the sense of the norm $\|\cdot\|$, with u as root.

It is clear that $r(u) = d^t \underline{u} = \|\underline{u}\|_*$ and $p_u(u) = 0$. So we have

$$\|p - p_u\| = \frac{|p(u)|}{|r(u)|}\|r\| \leqslant \|d\|\varepsilon.$$

As $\|d\| = 1$, we get

$$\|p - p_u\| \leqslant \varepsilon.$$

Hence $u \in Z_\varepsilon(p)$. □

3.3 Computing the ε-pseudozero set

Theorem 3 yields a computable expression for the ε-pseudozero set. It consists in evaluating a normalized form of polynomial p on a grid of the complex plane and comparing its value to the ε parameter.

Matlab software, for example, provides primitives that allow us to plot pseudozeros with the following very simple Algorithm 1. Such an implementation is very similar to existing pseudospectra software [8].

Algorithm 1 Computation of ε-pseudozero set

Require: polynomial p and precision ε

Ensure: pseudozero set layout in the complex plane

1: We grid a square containing the whole roots of p with the Matlab command `meshgrid`.
2: We compute $g(z)$ for the whole points z on the grid.
3: We draw the level line $|g(z)| = \varepsilon$ with the Matlab command `contour`.

Let L be the length of the square and h the step of discretisation. Evaluating of $g(u)$ needs the evaluation of a polynomial, that costs $\mathcal{O}(n)$ operations, the computation of the norm of a vector whose complexity depends on the norm. For example, the computation of the $\|\cdot\|_1$ requires $n-1$ operations and $\|\cdot\|_2$ requires $2n$ operations. Let us denote $\mathcal{O}(\|\cdot\|_*)$ this complexity. The complexity of the whole algorithm is in $\mathcal{O}((L/h)^2(n + \|\cdot\|_*))$.

Now, we expand the computable form of pseudozeros considering the two main perturbation types : *normwise* perturbations and *componentwise* perturbations. Componentwise perturbations describe every coefficient perturbations whereas normwise perturbations globally apply to the vector of coefficients.

3.4 Pseudozeros for normwise perturbations

Let p be defined by Relation (1) and $\widehat{p}$ a perturbed polynomial of p. We define the *normwise* norm by

$$\|p - \widehat{p}\|^{\mathcal{N}} = \frac{\|p - \widehat{p}\|}{\beta},$$

where $\|\cdot\|$ is a norm on the polynomials and β is a real. We usually choose $\beta = \|p\|$ to have a relative norm.

For such normwise perturbation, Theorem 3 gives the following result.

Corollary 4 *The ε-pseudozero set with* normwise *perturbations satisfies*

$$Z_\epsilon^{\mathcal{N}}(p) = \left\{ z \in \mathbb{C} : \frac{|p(z)|}{\|\underline{z}\|_* \beta} \leqslant \epsilon \right\}, \tag{7}$$

where $\underline{z} = (1, z, \dots, z^n)$ and $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$.

3.5 Pseudozeros for componentwise perturbations

We define the *componentwise* norm by

$$\|p - \widehat{p}\|^{\mathcal{C}} = \max_i \frac{|p_i - \widehat{p}_i|}{f_i},$$

where $(f_i)_{i=0,\dots,n}$ are non-negative real numbers. Usually, we take $f_i = |p_i|$ in order to have a relative norm. This perturbation provides a detailed description of the finite precision effect when the polynomial coefficient are represented with floating point numbers.

Theorem 3 now gives the following result.

Corollary 5 *The ε-pseudozero set with* componentwise *perturbations satisfies*

$$Z_\epsilon^{\mathcal{C}}(p) = \left\{ z \in \mathbb{C} : \frac{|p(z)|}{\sum_{i=0}^{n} |f_i||z|^i} \leqslant \epsilon \right\}. \tag{8}$$

4 First answers to approximate algebraic problems with pseudozeros

Drawing an ε-pseudozero set gives a "better understanding" of the behavior of the polynomial when we compute with it. It can help us to analyse the result of a computation. For instance, Figure 1 is the ε-pseudozero set of the Wilkinson polynomial

$$W_{20}(z) = (z-1)(z-2)\cdots(z-20),$$

when the coefficient of z^{19} is perturbed with $\varepsilon = 2^{-23}$.

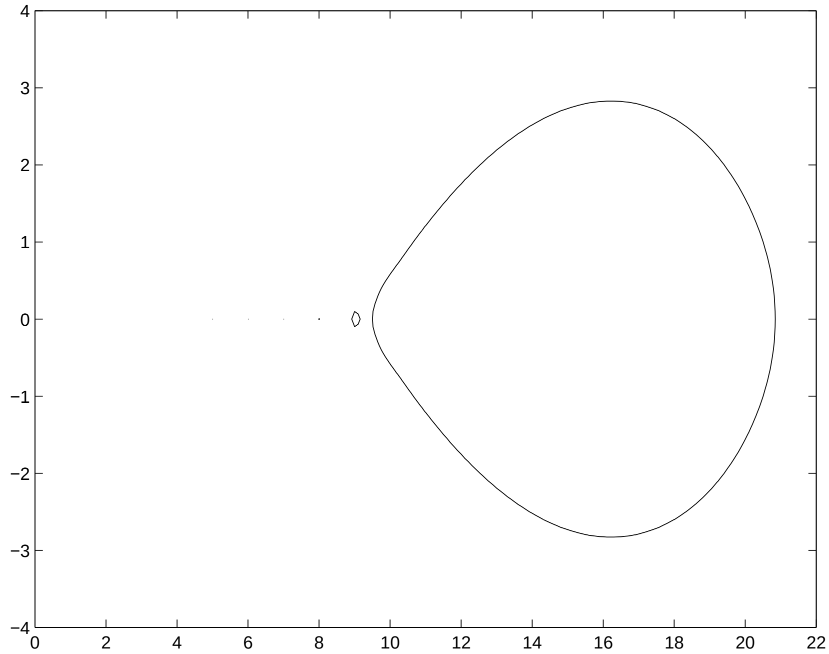


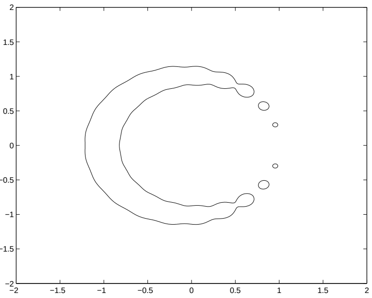
Fig. 1. ε-pseudozero set of Wilkinson polynomial for componentwise perturbation $\varepsilon = 2^{-23}$ to the coefficient z^{19}.

This plotting proves that no backward stable algorithm performed in IEEE-754 single precision can isolate the roots $10, 11, \ldots, 20$ of W_{20}.

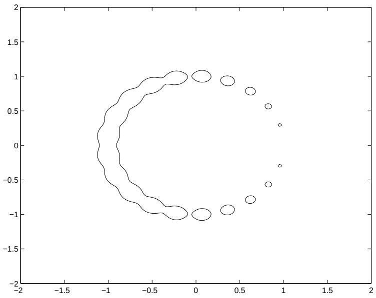
Figure 2 shows the evolution of the ε-pseudozero set when refining the precision ε. This can help us to choose the minimum computing precision necessary to isolate the roots.

The ε-pseudozero set can be used to decide the stability of system. Sometimes, it is important to know if it exists a root of modulus greater or less than 1. When the polynomial coefficients are known with a tolerance ε, it is difficult to compute all the roots of every polynomial in its neighborhood. Of course, sensitivity analysis that uses the condition number of the polynomial with respect to its coefficients can be performed and yields a first order criteria. Pseudozeros provide an alternative answer to this question without neglecting higher order effects of coefficient uncertainties. It suffices to draw the ε-pseudozero set and verify if it is included in the unit circle.

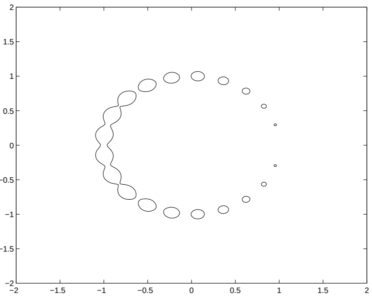
Figure 3 shows the ε-pseudozero set of polynomial $p(z) = (z-0.8)^2$ with two coefficient uncertainties $\varepsilon = 0.1$ and $\varepsilon = 0.01$.



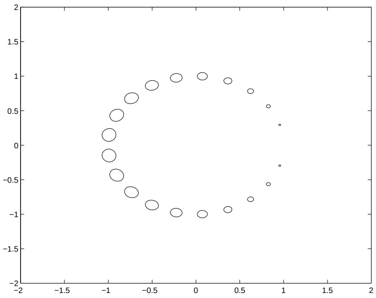
(a) $\epsilon = 10^{-1}$



(b) $\varepsilon = 10^{-1.2}$



(c) $\varepsilon = 10^{-1.\text{''}}$



(d) $\varepsilon = 10^{-1.4}$

Fig. 2. Pseudozero set of the polynomial $p(z) = 1 + z + \cdots + z^{20}$ for different values of ε.

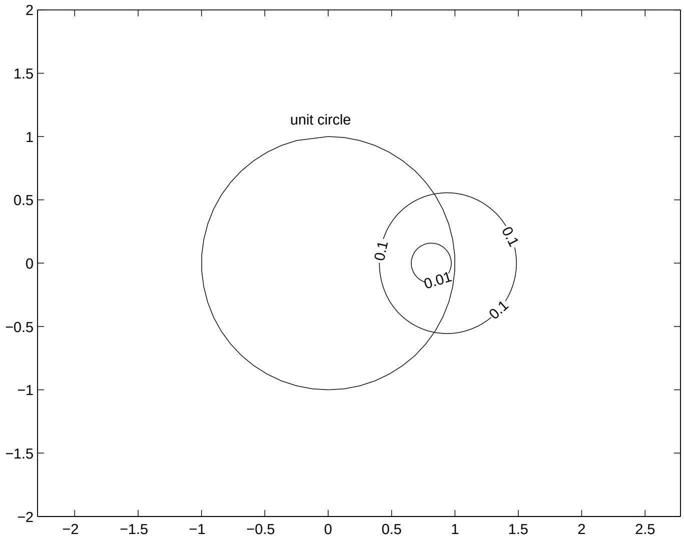


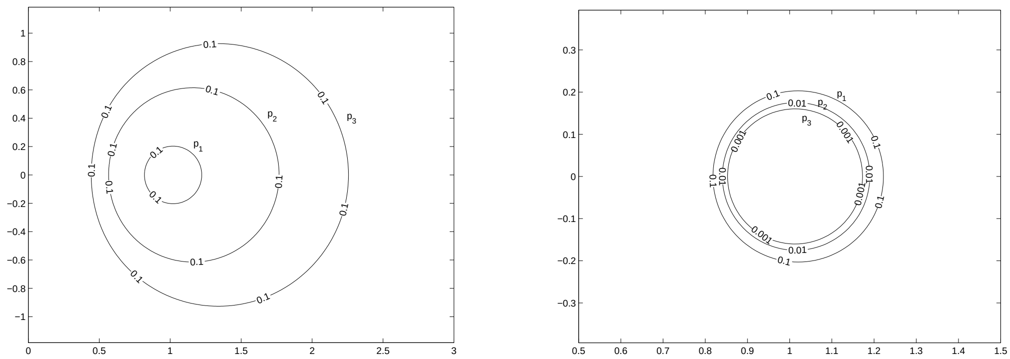
Fig. 3. ε-pseudozero set of the polynomial $p(z) = (z-0.8)^2$ with $\varepsilon = 0.1$ and $\varepsilon = 0.01$

From the picture, we cannot decide the stability for $\varepsilon = 0.1$ because some 0.1-pseudozeros have modulus greater than 1. On the other hand, we can see that all the 0.01-pseudozeros have modulus less than 1 and so conclude for stability.

Pseudozeros illustrate the well-known "rule of thumb" that describes the attainable accuracy of a multiple root computed in precision ε: this accuracy is of the order of ε^m where m is the multiplicity of the root ($\varepsilon < 1$). Another interpretation of this "rule of thumb" is that a backward stable algorithm cannot compute separate roots of any polynomial that admits a root in the ε^m-pseudozero set of a polynomial with a root of multiplicity equals to m. We verify this property computing for example the ε-pseudozeros of polynomials:

$$p_1(z) = z - 1,$$
$$p_2(z) = (z - 1)^2,$$
$$p_3(z) = (z - 1)^3,$$

with, respectively, $\varepsilon_1 = \varepsilon$, $\varepsilon_2 = \varepsilon^2$, $\varepsilon_3 = \varepsilon^3$ and $\varepsilon = 10^{-1}$. Figure 4 exhibits that the three sets $Z_{\varepsilon_1}(p_1)$, $Z_{\varepsilon_2}(p_2)$, and $Z_{\varepsilon_3}(p_3)$ are very similar (right side) compared to the ε-pseudozeros Z_ε of polynomials p_1, p_2 and p_3 (left side).



(a) Z_ε of p_1, p_2, p_3 and $\varepsilon = 10^{-1}$

(b) Pseudozero sets $Z_\varepsilon(p_1)$, $Z_{\varepsilon^2}(p_2)$, $Z_{\varepsilon^3}(p_3)$ for $\varepsilon = 10^{-1}$

Fig. 4. Comparison of different pseudozeros in terms of multiplicity

5 Polynomial primality and ε-pseudozero set

Let p and q belonging respectively to $\mathbb{P}_n$ and $\mathbb{P}_m$. It follows from the definition of coprimeness that p and q are ε-coprime if and only if for all $\widehat{p} \in N_\varepsilon(p)$, $\widehat{q} \in N_\varepsilon(p)$, the polynomials $\widehat{p}$ and $\widehat{q}$ are coprime.

The ε-pseudozero set provides an answer to the study of ε-primality. From the definition of the ε-pseudozero set, we derive that:

- if the intersection of the ε-pseudozero sets of p and q is empty then the two polynomials are ε-coprime,
- if the intersection is not empty then they are not ε-coprime.

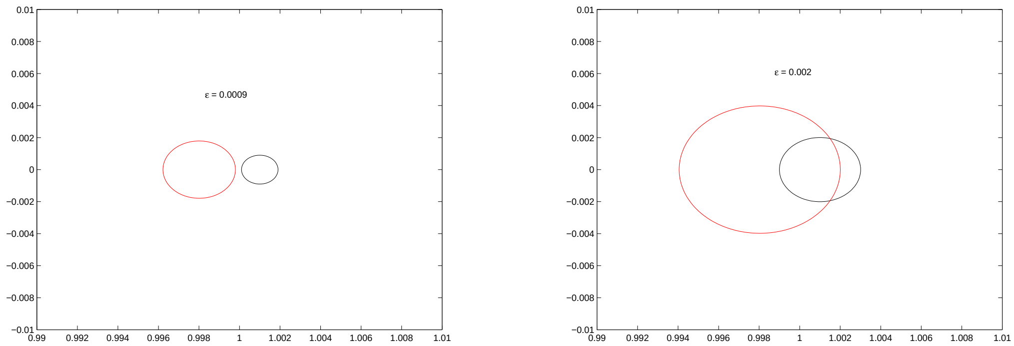
Let us prove these assertions.

Let p and q be two polynomials with complex coefficients. If $Z_\varepsilon(p) \cap Z_\varepsilon(q) = \emptyset$ then from the ε-pseudozero set definition, we cannot find $\widehat{p} \in N_\varepsilon(p)$ and $\widehat{q} \in N_\varepsilon(q)$ having a common root. It means that p and q are ε-coprime. If now $Z_\varepsilon(p) \cap Z_\varepsilon(q) \neq \emptyset$, then let us take $a \in Z_\varepsilon(p) \cap Z_\varepsilon(q)$. It means that it exists $\widehat{p} \in N_\varepsilon(p)$ and $\widehat{q} \in N_\varepsilon(q)$ such that $\widehat{p}(a) = 0$ and $\widehat{q}(a) = 0$. Hence the polynomial $(z - a)$ divide $\widehat{p}$ and $\widehat{q}$. Therefore p et q are not ε-coprime.

We illustrate this property considering, for example, p and q, where

$$p(z) = z^2 - 3.999z + 3.001 \quad \text{and} \quad q(z) = z^2 - 3.001z + 1.999.$$

We draw the ε-pseudozero set of these two polynomials for two values of ε (0.0009 and 0.002) in Figure 5. On the left hand side picture, the intersection is empty so the two polynomial p and q are 0.0009-coprime. On the contrary, the intersection is not empty on the right hand picture, so p and q are not 0.002-coprime.



(a) p and q are 0.0009-coprime

(b) p and q have a common 0.002-divisor

Fig. 5. ε-pseudozero set for different values of ε of the polynomials p and q.

6 Computing ε-pseudozero set in finite precision

In this section, we discuss two aspects of the finite precision computation of ε-pseudozero set.

6.1 How can we a priori choose the grid?

The initial grid must satisfy the two following properties:

a) the zeros and pseudozeros are included in its coverage;

b) the roots are isolated by the grid discretisation.

a) We choose to grid the square $[-R, R] \times [-R, R]$ where

$$R = \max\{1, \sum_{i=1}^{n} |p_i| + n\epsilon\}.$$

Let p be a unitary polynomial of degree n and $\{z_i\}$ the set of its n roots. For $r = \max_{i=1,\ldots,n} |z_i|$, we verify [27, p. 154] that

$$r \leqslant \max\{1, \sum_{k=1}^{n} |p_k|\}.$$

This enables us to build a square containing the set of the pseudozeros. Indeed, let z be in $Z_\epsilon(p)$. Then we know it exists $\widehat{p} \in N_\epsilon(p)$ such that $\widehat{p}(z) = 0$. The complex number z being a root of $\widehat{p}$, it is less than the greatest root of $\widehat{p}$. So we have the inequality

$$|z| \leqslant \max\{1, \sum_{k=1}^{n} |\widehat{p}_k|\}.$$

Assuming that the perturbation norm is an Hölder p-norm $\|\cdot\|_p$, we know that $\|p - \widehat{p}\|_p \leqslant \epsilon$. Since $\|\cdot\|_\infty \leqslant \|\cdot\|_p$, $\|p - \widehat{p}\|_\infty \leqslant \epsilon$. Then $|p_i - \widehat{p}_i| \leqslant \epsilon$ and we have $|\widehat{p}| \leqslant |p_i| + \epsilon$ for all $i = 1, \ldots, n$. Hence,

$$|z| \leqslant \max\{1, \sum_{i=1}^{n} |p_i| + n\epsilon\} = R.$$

It yields that $Z_\epsilon(p) \subset B(0, R)$, where $B(0, R)$ is the closed ball of center 0 and of radius R.

The drawbacks of this method is that if the polynomial coefficients are large then the grid can be very large even if the roots are very small. A solution would be using other bounds for the roots.

b) We need a grid that provides the isolations of the roots of p. The discretisation step of the grid must be chosen consequently. It exists numerous results about the separation of the roots. Current implementation of our pseudozero package does not automatically manage the separation of the roots since we choose the discretisation step by successive experimentations.

6.2 Accuracy limitation due to the finite precision evaluation of $p(z)$

The computation of the pseudozero set consists in the evaluation of the function $g(z) = p(z)/f(z)$ (where p is a polynomial and f a norm) performed at every node of the chosen grid. For usual norm, we have $f(z) \geqslant 1$ and the associated computing error is negligible. The error in the evaluation of polynomial p has to be considered.

Let y be the evaluation of $p(z)$ using the Hörner's scheme. It is well known [17, p. 95] that

$$|y - p(z)| \leqslant 2n\mathbf{u} \sum_{i=0}^{n} |p_i||z|^i =: \eta,$$

where $\mathbf{u}$ is the computing precision. The more precise following bound has been proposed by Kahan

$$|y - p(z)| \leqslant 8n\mathbf{u} \sum_{i=0}^{n} |s_i z^i| \quad \text{with} \quad s_i = \sum_{j=i}^{n} p_j z^{j-i}.$$

Then no reliable interpretation of ε-pseudozeros can be proposed when $\varepsilon < \eta$. In this case, increasing the computing precision $\mathbf{u}$ yields a reliable evaluation of ε-pseudozeros.

7 Conclusion and future directions

We have shown that plotting pseudozero can give qualitative and quantitative interesting informations about the behavior of polynomials used in a finite precision environment. Future work consists in developing a software that will automatize all the procedures described in the article. We hope that pseudozeros will be used as much as the pseudospectra because it seems to us that it could be useful for some application fields as CGAD, control and network theory for example.

References

[1] Bernhard Beckermann and George Labahn. A fast and numerically stable Euclidean-like algorithm for detecting relatively prime numerical polynomials. *J. Symbolic Comput.*, 26(6):691–714, 1998.

[2] Bernhard Beckermann and George Labahn. When are two numerical polynomials relatively prime? *J. Symbolic Comput.*, 26(6):677–689, 1998.

[3] Stan Cabay, Anthony R. Jones, and George Labahn. Algorithm 766: Experiments with a weakly stable algorithm for computing Pade-Hermite and simultaneous Pade approximants. *ACM Trans. Math. Softw.*, 23(1):91–110, 1997.

[4] Françoise Chaitin-Chatelin and Valérie Frayssé. *Lectures on finite precision computations*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[5] Paulina Chin, Robert M. Corless, and George F. Corliss. Optimization strategies for the approximate GCD problem. In Oliver Gloor, editor, *Proceedings of the 1998 international symposium on symbolic and algebraic computation, ISSAC '98*, pages 228–235, Août 1998.

[6] Robert M. Corless, Patrizia M. Gianni, Barry M. Trager, and Stephen M. Watt. The singular value decomposition for polynomial systems. In A. H. M. Levelt, editor, *Proceedings of the 1995 international symposium on symbolic and algebraic computation, ISSAC '95*, pages 195–207. ACM Press, Juillet 1995.

[7] Alan Edelman and H. Murakami. Polynomial roots from companion matrix eigenvalues. *Math. Comp.*, 64(210):763–776, 1995.

[8] Mark Embree and Nick Trefethen. *Pseudospectra Gateway*. http://web.comlab.ox.ac.uk/projects/pseudospectra/.

[9] Ioannis Z. Emiris. Symbolic-numeric algebra for polynomials. Rapport de recherche, Institut National de Recherche en Informatique et en Automatique (INRIA), 1997.

[10] Ioannis Z. Emiris, André Galligo, and Henri Lombardi. Numerical univariate polynomial GCD. In *The mathematics of numerical analysis (Park City, UT, 1995)*, pages 323–343. Amer. Math. Soc., Providence, RI, 1996.

[11] Ioannis Z. Emiris, André Galligo, and Henri Lombardi. Certified approximate univariate GCDs. *J. Pure Appl. Algebra*, 117/118:229–251, 1997.

[12] Walter Gautschi. Questions of numerical condition related to polynomials. In *Studies in numerical analysis*, pages 140–177. Math. Assoc. America, Washington, DC, 1984.

[13] K. O. Geddes, S. R. Czapor, and G. Labahn. *Algorithms for computer algebra*. Kluwer Academic Publishers, Boston, MA, 1992.

[14] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[15] Desmond J. Higham and Nicholas J. Higham. *MATLAB guide*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[16] Nicholas J. Higham. *Handbook of writing for the mathematical sciences*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 1998.

[17] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.

[18] D. Hinrichsen and B. Kelb. Spectral value sets: a graphical tool for robustness analysis. *Systems Control Lett.*, 21(2):127–136, 1993.

[19] Markus A. Hitz. *Efficient algorithms for computing the nearest polynomial with constrained roots*. PhD thesis, Rensselaer Polytechnic Institute, Avril 1998.

[20] Markus A. Hitz and Erich Kaltofen. Efficient algorithms for computing the nearest polynomial with constrained roots. In *Proceedings of the 1998 International Symposium on Symbolic and Algebraic Computation (Rostock)*, pages 236–243 (electronic), New York, 1998. ACM.

[21] Markus A. Hitz, Erich Kaltofen, and Y. N. Lakshman. Efficient algorithms for computing the nearest polynomial with a real root and related problems. In *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation (Vancouver, BC)*, pages 205–212 (electronic), New York, 1999. ACM.

[22] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 1990.

[23] V. Hribernig and H. J. Stetter. Detection and validation of clusters of polynomial zeros. *J. Symbolic Comput.*, 24(6):667–681, 1997.

[24] N. Karmarkar and Y.N. Lakshman. Approximate polynomial greatest common divisors and nearest singular polynomials. In Y. N. Lakshman, editor, *Proceedings of the 1996 international symposium on symbolic and algebraic computation, ISSAC '96*, pages 35–39, Juillet 1996.

[25] N. K. Karmarkar and Y. N. Lakshman. On approximate GCDs of univariate polynomials. *J. Symbolic Comput.*, 26(6):653–666, 1998. Symbolic numeric algebra for polynomials.

[26] Donald E. Knuth. *The art of computer programming. Vol. 2*. Addison-Wesley Publishing Co., Reading, Mass., second edition, 1981. Seminumerical algorithms, Addison-Wesley Series in Computer Science and Information Processing.

[27] Maurice Mignotte. *Mathematiques pour le calcul formel*. Presses Universitaires de France, Paris, 1989.

[28] Ronald G. Mosier. Root neighborhoods of a polynomial. *Math. Comp.*, 47(175):265–273, 1986.

[29] Matu-Tarow Noda and Tateaki Sasaki. Approximate GCD and its application to ill-conditioned algebraic equations. In *Proceedings of the International Symposium on Computational Mathematics (Matsuyama, 1990)*, volume 38, pages 335–351, 1991.

[30] Masa-aki Ochi, Matu-Tarow Noda, and Tateaki Sasaki. Approximate greatest common divisor of multivariate polynomials and its application to ill-conditioned systems of algebraic equations. *J. Inform. Process.*, 14(3):292–300, 1991.

[31] A. M. Ostrowski. *Solution of equations and systems of equations*. Academic Press, New York, 1966.

[32] Victor Y. Pan. Numerical computation of a polynomial gcd and extensions. Rapport de Recherche 2969, Institut National de Recherche en Informatique et en Automatique (INRIA), Août 1996.

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

[34] Victor Y. Pan. Computation of approximate polynomial GCDs and an extension. *Inform. and Comput.*, 167:71–85, 2001.

[35] David Rupprecht. *Elément de Géométrie Approchée : Etude du pgcd et de la factorisation*. PhD thesis, Université de Nice-Sophia Antipolis, Janvier 2000.

[36] Tateaki Sasaki and Matu-Tarow Noda. Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations. *J. Inform. Process.*, 12(2):159–168, 1989.

[37] Reiner Schätzle. On the perturbation of the zeros of complex polynomials. *IMA J. Numer. Anal.*, 20(2):185–202, 2000.

[38] Arnold Schönhage. Quasi-gcd computations. *J. Complexity*, 1(1):118–137, 1985.

[39] T. W. Sederberg and G.-Z. Chang. Best linear common divisors for approximate degree reduction. *Computer-aided design*, 25(3):163–168, march 1993.

[40] Hans J. Stetter. The nearest polynomial with a given zero, and similar problems. *SIGSAM Bulletin (ACM Special Interest Group on Symbolic and Algebraic Manipulation)*, 33(4):2–4, décembre 1999.

[41] Hans J. Stetter. Polynomials with coefficients of limited accuracy. In *Computer algebra in scientific computing—CASC'99 (Munich)*, pages 409–430. Springer, Berlin, 1999.

[42] Kim-Chuan Toh and Lloyd N. Trefethen. Pseudozeros of polynomials and pseudospectra of companion matrices. *Numer. Math.*, 68(3):403–425, 1994.

[43] Lloyd N. Trefethen. Computation of pseudospectra. In *Acta numerica, 1999*, pages 247–295. Cambridge Univ. Press, Cambridge, 1999.

[44] J. H. Wilkinson. *Rounding errors in algebraic processes*. Prentice-Hall Inc., Englewood Cliffs, N.J., 1963.

[45] Hong Zhang. Numerical condition of polynomials in different forms. *ETNA*, 12:66–87, 2001.