New Progress in Univariate Polynomial Root Finding

ISSAC 2020

Rémi Imbach[1,3] and Victor Y. Pan[2,4]

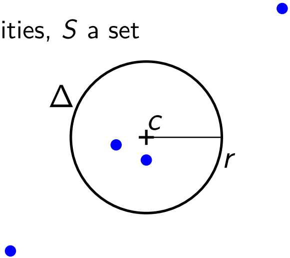
[1] Courant Institute of Mathematical Sciences, New York University, USA
[2] Lehman College, City University of New York, USA
[3] NSF Grants # CCF-1563942 and # CCF-1564132
[4] NSF Grants # CCF-1116736 and # CCF-1563942 and PSC CUNY Award 698130048.

Root Counters

Complex disk: $\Delta = D(c, r) := \{z \text{ s.t. } |z - c| \leq r\}$

Polynomial: $p \in \mathbb{C}[z]$ of degree d
$\#(S, p) :=$ nb of roots of p in S counted with multiplicities, S a set

Counting test:

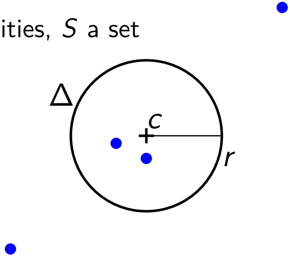


Exclusion test:

Root Counters

Complex disk: $\Delta = D(c,r) := \{z \text{ s.t. } |z-c| \leq r\}$

Polynomial: $p \in \mathbb{C}[z]$ of degree d
$\#(S,p) :=$ nb of roots of p in S counted with multiplicities, S a set



Counting test:

$C^*(p,\Delta,\ldots)$

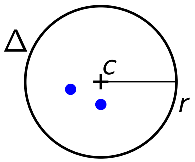
Input: $p \in \mathbb{C}[z]$ of degree d, $\Delta = D(c,r)$ a disk
Output: $\#(\Delta,p)$ or -1 (**can not decide**)

Exclusion test:

Root Counters

Complex disk: $\Delta = D(c, r) := \{z \text{ s.t. } |z - c| \leq r\}$

Polynomial: $p \in \mathbb{C}[z]$ of degree d
$\#(S, p) :=$ nb of roots of p in S counted with multiplicities, S a set



Counting test:

$C^*(p, \Delta, \ldots)$

Input: $p \in \mathbb{C}[z]$ of degree d, $\Delta = D(c, r)$ a disk
Output: $\#(\Delta, p)$ or -1 (**can not decide**)

Exclusion test:

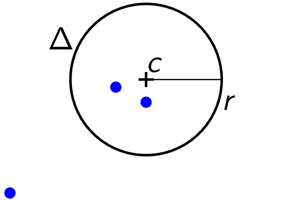
$C^0(p, \Delta, \ldots)$

Input: $p \in \mathbb{C}[z]$ of degree d, $\Delta = D(c, r)$ a disk
Output: **true** $(\#(\Delta, p) = 0)$ or **can not decide**

Root Counters

Complex disk: $\Delta = D(c, r) := \{z \text{ s.t. } |z - c| \leq r\}$

Polynomial: $p \in \mathbb{C}[z]$ of degree d
$\#(S, p) :=$ nb of roots of p in S counted with multiplicities, S a set



Counting test:

$C^*(p, \Delta, \dots)$

Input: $p \in \mathbb{C}[z]$ of degree d, $\Delta = D(c, r)$ a disk
Output: $\#(\Delta, p)$ or -1 (**can not decide**)

Exclusion test:

$C^0(p, \Delta, \dots)$

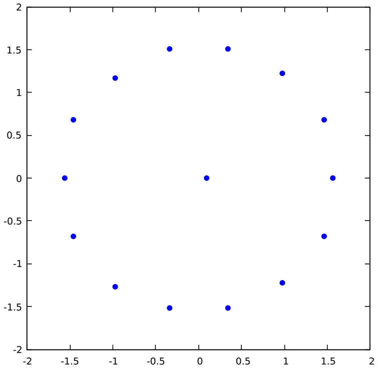
Input: $p \in \mathbb{C}[z]$ of degree d, $\Delta = D(c, r)$ a disk
Output: **true** ($\#(\Delta, p) = 0$) or **can not decide**

Core tools for root-finding algorithms based on subdivision

Root Clustering Problem

Input: a polynomial $p \in \mathbb{C}[z]$ of degree $d > 1$

Output:



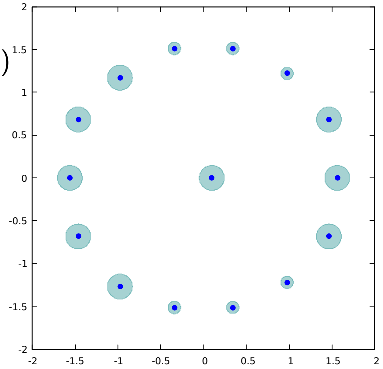
Example: Mignotte-like polynomial: $z^d - 2(2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root Clustering Problem

Input: a polynomial $p \in \mathbb{C}[z]$ of degree $d > 1$

Output: a set of pairs $\{(\Delta^1, m^1), \ldots, (\Delta^\ell, m^\ell)\}$ where

- the Δ_j's are pairwise-disjoint disks
- $\forall j$, $\#(\Delta^j, p) = m^j$, and $\#(3\Delta^j, p) = m^j$ (natural clusters)
- $Z(\mathbb{C}, p) = \bigcup_{j=1}^{\ell} Z(\Delta^j, p)$ and $\ell > 1$



Notations: $Z(S, p)$: roots of p in S

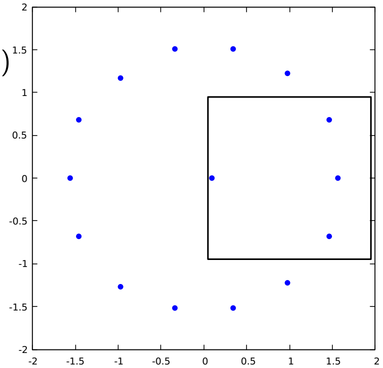
Example: Mignotte-like polynomial: $z^d - 2(2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root Clustering Problem Local Version

Input: a polynomial $p \in \mathbb{C}[z]$, a complex box B

Output: a set of pairs $\{(\Delta^1, m^1), \ldots, (\Delta^\ell, m^\ell)\}$ where

- the Δ_j's are pairwise-disjoint disks
- $\forall j$, $\#(\Delta^j, p) = m^j$,
  and $\#(3\Delta^j, p) = m^j$ (natural clusters)



Notations: $Z(S, p)$: roots of p in S

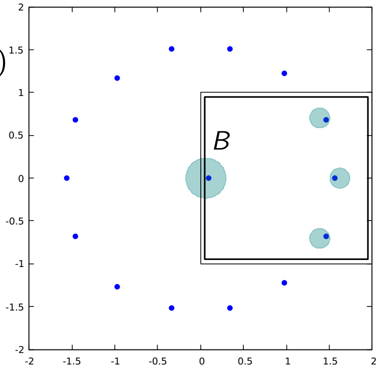
Example: Mignotte-like polynomial: $z^d - 2(2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root Clustering Problem Local Version

Input: a polynomial $p \in \mathbb{C}[z]$, a complex box B

Output: a set of pairs $\{(\Delta^1, m^1), \dots, (\Delta^\ell, m^\ell)\}$ where

- the Δ_j's are pairwise-disjoint disks
- $\forall j$, $\#(\Delta^j, p) = m^j$,
  and $\#(3\Delta^j, p) = m^j$ (natural clusters)
- $Z(B, p) \subseteq \bigcup_{j=1}^{\ell} Z(\Delta^j, p) \subseteq Z(\delta B, p)$,
  for $\delta > 1$



Notations: $Z(S, p)$: roots of p in S

Example: Mignotte-like polynomial: $z^d - 2(2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Oracle Polynomials

Let $\alpha \in \mathbb{C}$.

Oracle for α: function $\mathcal{O}_\alpha : \mathbb{N} \rightarrow \mathbb{C}$ s.t. $|\alpha - \mathcal{O}_\alpha(L)| \leq 2^{-L}$

Let $p \in \mathbb{C}[z]$.

Oracle for p: function $\mathcal{O}_p : \mathbb{N} \rightarrow \mathbb{C}[z]$ s.t. $\|p - \mathcal{O}_p(L)\|_\infty \leq 2^{-L}$

Local Root Clustering Algorithm

[BSS+16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation*, pages 71–78. ACM, 2016.

Input polynomial: p given as an oracle

Near optimal: bit complexity $\widetilde{O}(d^2(\sigma + d))$
for the benchmark problem ($p \in \mathbb{Z}[z]$, degree d, bit-size σ)

Implementation: `C` package `Ccluster`[1]
interface for julia: `Ccluster.jl`[2]
available in SINGULAR

[1] `https://github.com/rimbach/Ccluster`
[2] `https://github.com/rimbach/Ccluster.jl`

Local Root Clustering Algorithm

[BSS+16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation*, pages 71–78. ACM, 2016.

Input polynomial: p given as an oracle

Near optimal: bit complexity $\widetilde{O}(d^2(\sigma + d))$
for the benchmark problem ($p \in \mathbb{Z}[z]$, degree d, bit-size σ)

In practice: Still not the users choice for global problems (`MPsolve`)

Implementation: `C` package `Ccluster`[1]
interface for **julia**: `Ccluster.jl`[2]
available in SINGULAR

[1] `https://github.com/rimbach/Ccluster`
[2] `https://github.com/rimbach/Ccluster.jl`

Outline of [BSS+16]

Counting test:

Exclusion test:

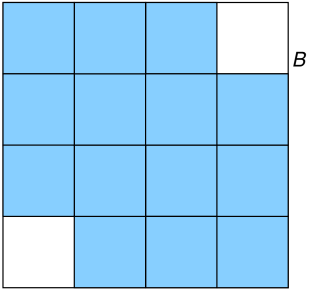
Subdivision approach:

Outline of [BSS+16]

Counting test:

Exclusion test:

Subdivision approach:

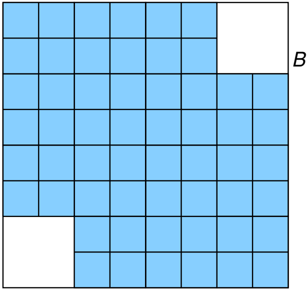


Outline of [BSS+16]

Counting test:

Exclusion test:

Subdivision approach:

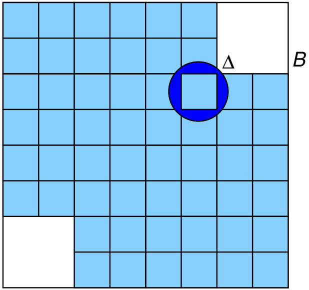


Outline of [BSS+16]

Counting test:

Exclusion test: $C^0(\Delta, p)$

Subdivision approach:

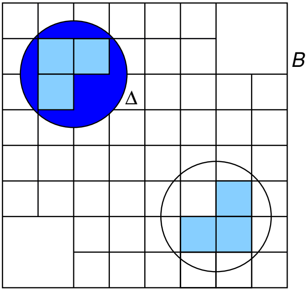


Outline of [BSS+16]

Counting test: $C^*(\Delta, p)$

Exclusion test: $C^0(\Delta, p)$

Subdivision approach:



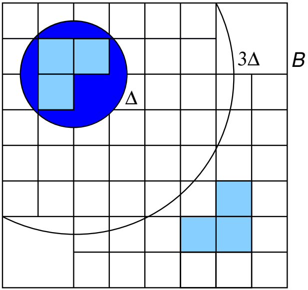
Outline of [BSS$^+$16]

Counting test: $C^*(\Delta, p)$

Exclusion test: $C^0(\Delta, p)$

Based on Pellet's Theorem: for $\Delta = D(c, r)$, requires to compute the coefficients of $p(c + rz)$

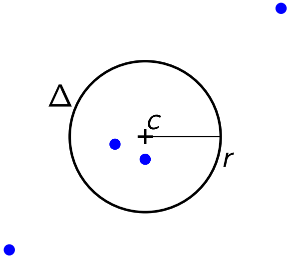
Subdivision approach:



Our Contributions

New Counting and Exclusion tests: based on power sums approximations

- require Δ to be "well isolated"
- do not require to shift p in $c + rz$
- evaluate p on a small nb. of points



Unsure Exclusion test:

- assume that Δ is well isolated
- check necessary conditions to verify the result
- experimentally reliable

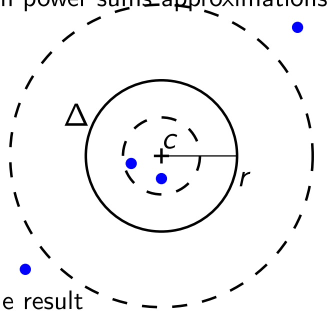
Algorithm for the Global Root Clustering Problem:

- uses our unsure exclusion test
- the output is checked as a post-procedure

Our Contributions

New Counting and Exclusion tests: based on power sums approximations

- require Δ to be "well isolated"
- do not require to shift p in $c + rz$
- evaluate p on a small nb. of points



Unsure Exclusion test:

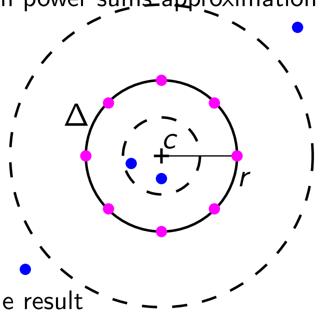
- assume that Δ is well isolated
- check necessary conditions to verify the result
- experimentally reliable

Algorithm for the Global Root Clustering Problem:

- uses our unsure exclusion test
- the output is checked as a post-procedure

Our Contributions

New Counting and Exclusion tests: based on power sums approximations



- require Δ to be "well isolated"
- do not require to shift p in $c + rz$
- evaluate p on a small nb. of points

Unsure Exclusion test:

- assume that Δ is well isolated
- check necessary conditions to verify the result
- experimentally reliable

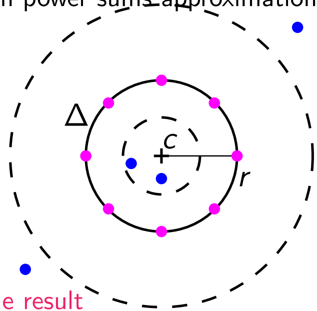
Algorithm for the Global Root Clustering Problem:

- uses our unsure exclusion test
- the output is checked as a post-procedure

Our Contributions

New Counting and Exclusion tests: based on power sums approximations

- require Δ to be "well isolated"
- do not require to shift p in $c + rz$
- evaluate p on a small nb. of points



Unsure Exclusion test:

- assume that Δ is well isolated
- check necessary conditions to verify the result
- experimentally reliable

Algorithm for the Global Root Clustering Problem:

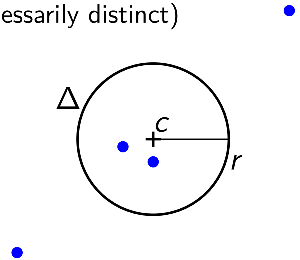
- uses our unsure exclusion test
- the output is checked as a post-procedure

Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$

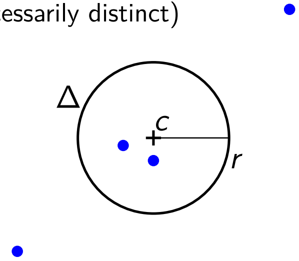
h-th power sum of p in Δ:

$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$



Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$



h-th power sum of *p* in Δ:

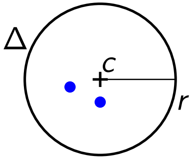
$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$

Remarks:

(i) $\#(\Delta, p) = s_0(\Delta, p)$

Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$



h-th power sum of *p* in Δ:

$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$

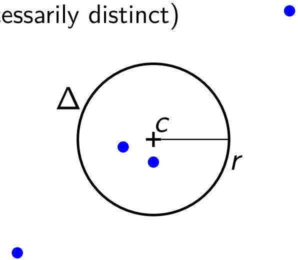
Remarks:

(i) $\#(\Delta, p) = s_0(\Delta, p)$

(ii) $\#(\Delta, p) = 0 \Rightarrow s_h(\Delta, p) = 0$ for any h

Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$



h-th power sum of *p* in Δ:

$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$

Remarks:

(i) $\#(\Delta, p) = s_0(\Delta, p)$

(ii) $\#(\Delta, p) = 0 \Rightarrow s_h(\Delta, p) = 0$ for any h

Let $p_\Delta = p(c + rz)$:

(iii) $\#(\Delta, p) = s_0(D(0, 1), p_\Delta)$

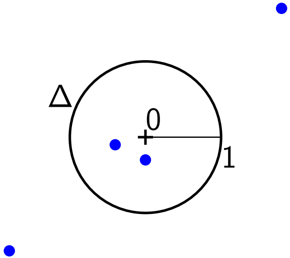
(iv) $\#(\Delta, p) = 0 \Rightarrow s_h(D(0, 1), p_\Delta) = 0$ for any h

Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.

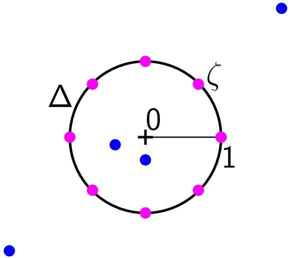


Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.

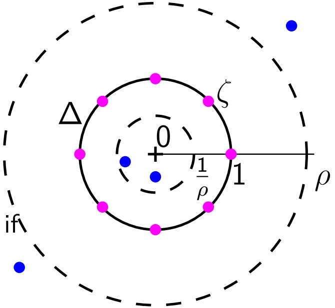


Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.



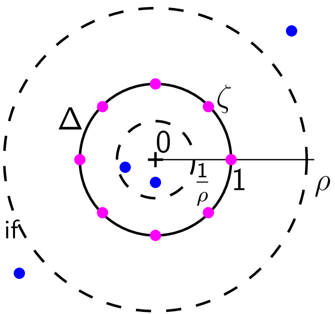
Definition: $\Delta = D(c,r)$ is ρ isolated, for $\rho > 1$, if $D(c, r\rho) \setminus D(c, \frac{r}{\rho})$ contains no root

Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q}\sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Definition: $\Delta = D(c,r)$ is ρ isolated, for $\rho > 1$, if $D(c, r\rho) \setminus D(c, \frac{r}{\rho})$ contains no root

Theorem [Sch82]: Let $\rho > 1$;
suppose $D(0,1)$ is ρ-isolated and contains d_Δ roots. Then

$$|s_h^* - s_h(D(0,1), p)| \leq \frac{d_\Delta \theta^{q+h} + (d - d_\Delta)\theta^{q-h}}{1 - \theta^q} \text{ where } \theta = \frac{1}{\rho}$$

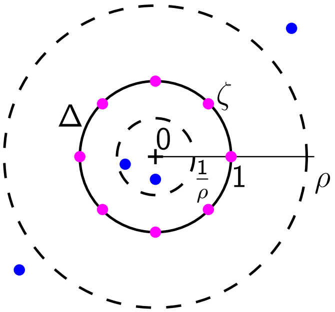
[Sch82] Arnold Schönhage.
The fundamental theorem of algebra in terms of computational complexity.
Manuscript. Univ. of Tübingen, Germany, 1982.

Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(0,1)$ is ρ-isolated. Then

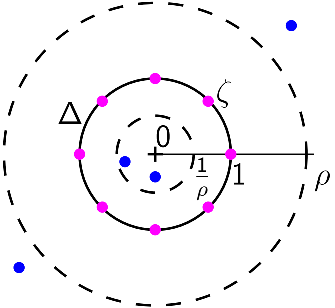
(i) $|s_0^* - s_0(D(0,1), p)| \leq \dfrac{d\theta^q}{1-\theta^q}$ where $\theta = \dfrac{1}{\rho}$

Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(0,1)$ is ρ-isolated. Then

(i) $|s_0^* - s_0(D(0,1),p)| \leq \dfrac{d\theta^q}{1-\theta^q}$ where $\theta = \dfrac{1}{\rho}$

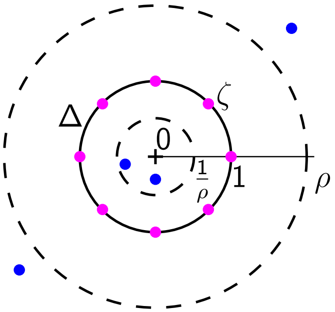
(ii) Fix $e > 0$. If $q = \lceil \log_\theta(\frac{e}{d+e}) \rceil$ then $|s_0^* - s_0(D(0,1),p)| \leq e$.

Approximation of the Power Sums in $D(0,1)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{1}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(\zeta^g)}{p(\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(0,1)$ is ρ-isolated. Then

(i) $|s_0^* - s_0(D(0,1), p)| \leq \dfrac{d\theta^q}{1-\theta^q}$ where $\theta = \dfrac{1}{\rho}$

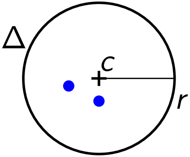
(ii) Fix $e > 0$. If $q = \lceil \log_\theta(\frac{e}{d+e}) \rceil$ then $|s_0^* - s_0(D(0,1), p)| \leq e$.

Remark: $s_0(D(0,1), p)$ is an integer, thus error $e < \frac{1}{4}$ is enough to recover it from s_0^* !

Example: when $\rho = 2$ and $d = 500$, $q = 11$ in enough!

Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$



h-th power sum of p in Δ:

$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$

Remarks:

(i) $\#(\Delta, p) = s_0(\Delta, p)$

(ii) $\#(\Delta, p) = 0 \Rightarrow s_h(\Delta, p) = 0$ for any h

Let $p_\Delta = p(c + rz)$:

(iii) $\#(\Delta, p) = s_0(D(0, 1), p_\Delta)$

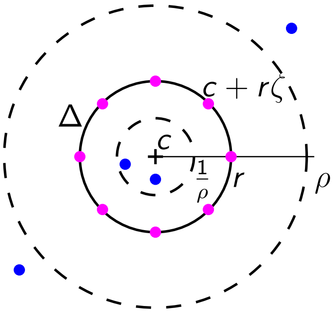
(iv) $\#(\Delta, p) = 0 \Rightarrow s_h(D(0, 1), p_\Delta) = 0$ for any h

Approximation of the Power Sums in $\Delta = D(c, r)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(c + r\zeta^g)}{p(c + r\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(c, r)$ is ρ-isolated. Then

(i) $|s_0^* - s_0(D(c, r), p)| \leq \dfrac{d\theta^q}{1 - \theta^q}$ where $\theta = \dfrac{1}{\rho}$

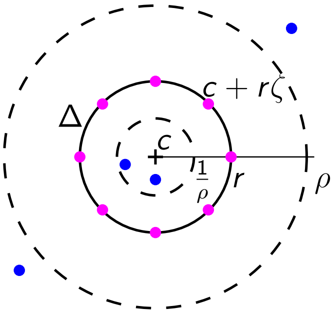
(ii) Fix $e > 0$. If $q = \lceil \log_\theta(\frac{e}{d+e}) \rceil$ then $|s_0^* - s_0(D(c, r), p)| \leq e$.

Approximation of the Power Sums in $\Delta = D(c, r)$

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(c + r\zeta^g)}{p(c + r\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(c, r)$ is ρ-isolated. Then

(i) $|s_0^* - s_0(D(c, r), p)| \leq \dfrac{d\theta^q}{1 - \theta^q}$ where $\theta = \dfrac{1}{\rho}$

(ii) Fix $e > 0$. If $q = \lceil \log_\theta(\frac{e}{d+e}) \rceil$ then $|s_0^* - s_0(D(c, r), p)| \leq e$.

Remark: Computing $s_0(D(c, r), p)$ does **not** require to compute the coefficients of $p_\Delta = p(c + rz)$!

Counting and Exclusion Tests

$P^*(p, \Delta, \rho)$ //*Output in* $\{0, 1, \ldots, d\}$

Input: $p \in \mathbb{C}[z]$ of degree d, $\rho > 1$, Δ a ρ-isolated disk
Output: $\#(\Delta, p)$

1. $e \leftarrow 1/4$, $\theta \leftarrow 1/\rho$
2. $q \leftarrow \lceil \log_\theta(\frac{e}{d+e}) \rceil$
3. compute $s_0^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^g \frac{p'(c+r\zeta^g)}{p(c+r\zeta^g)}$
4. **return** the unique integer in $D(s_0^*, e)$

Counting and Exclusion Tests

$P^*(p, \Delta, \rho)$ *//Output in* $\{0, 1, \ldots, d\}$

Input: $p \in \mathbb{C}[z]$ of degree d, $\rho > 1$, Δ a ρ-isolated disk
Output: $\#(\Delta, p)$

1. $e \leftarrow 1/4$, $\theta \leftarrow 1/\rho$
2. $q \leftarrow \lceil \log_\theta(\frac{e}{d+e}) \rceil$
3. compute $s_0^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^g \frac{p'(c+r\zeta^g)}{p(c+r\zeta^g)}$
4. **return** the unique integer in $D(s_0^*, e)$

Remarks: One can derive

1. an implementable version for oracle polynomials

Counting and Exclusion Tests

$P^*(p, \Delta, \rho)$ *//Output in* $\{0, 1, \ldots, d\}$

Input: $p \in \mathbb{C}[z]$ of degree d, $\rho > 1$, Δ a ρ-isolated disk
Output: $\#(\Delta, p)$

Remarks: One can derive

1. an implementable version for oracle polynomials
2. an exclusion test:

$P^0(p, \Delta, \rho)$ *//Output in* { **true**, **false** }

Input: $p \in \mathbb{C}[z]$ of degree d, $\rho > 1$, Δ a ρ-isolated disk
Output: **true** iff p has no root in Δ

1. return $P^*(p, \Delta, \rho) == 0$

Counting and Exclusion Tests

$P^*(p, \Delta, \rho)$ *//Output in* $\{0, 1, \dots, d\}$

Input: $p \in \mathbb{C}[z]$ of degree d, $\rho > 1$, Δ a ρ-isolated disk
Output: $\#(\Delta, p)$

Remarks: One can derive

1. an implementable version for oracle polynomials
2. an exclusion test:

Question: What if ρ is not known?

Unsure Exclusion Test

$\widetilde{P^0}(p, \Delta)$

Input: $p \in \mathbb{C}[z]$ of degree d, Δ a disk
Output: in { **true**, **can not decide** }

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
1. $e \leftarrow 1/4$, $\theta \leftarrow 1/\rho$
2. $q \leftarrow \lceil \log_\theta(\frac{e}{d+e}) \rceil$
3. compute $s_0^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^g \frac{p'(c+r\zeta^g)}{p(c+r\zeta^g)}$
4. **if** $D(s_0^*, e)$ does not contain zero
5. **return can not decide**
6. **return true**

Unsure Exclusion Test

$\widetilde{P^0}(p, \Delta)$

Input: $p \in \mathbb{C}[z]$ of degree d, Δ a disk
Output: in { **true**, **can not decide** }

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
1. $e \leftarrow 1/4$, $\theta \leftarrow 1/\rho$
2. $q \leftarrow \lceil \log_\theta(\frac{e}{d+e}) \rceil$
3. compute $s_0^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^g \frac{p'(c+r\zeta^g)}{p(c+r\zeta^g)}$
4. **if** $D(s_0^*, e)$ does not contain zero
5. **return can not decide**
6. **return true**

Remark: Even if the output of $\widetilde{P^0}(p, \Delta)$ is **true**, it may be wrong

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests			
d	n	t_0/t (%)	t_1/t_0	#F		
100 random dense polynomials per degree						
64	116302	87.2	1.0	4		
128	227842	90.5	.54	21		
191	340348	92.0	.42	26		
100 random sparse (10 monomials) polynomials per degree						
64	115850	86.2	.90	10		
128	226266	91.3	.36	11		
191	331966	92.1	.25	11		

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests			
d	n	t_0/t (%)	t_1/t_0	#F		
100 random dense polynomials per degree						
64	116302	87.2	1.0	4		
128	227842	90.5	.54	21		
191	340348	92.0	.42	26		
100 random sparse (10 monomials) polynomials per degree						
64	115850	86.2	.90	10		
128	226266	91.3	.36	11		
191	331966	92.1	.25	11		

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests			
d	n	t_0/t (%)	t_1/t_0	#F		
100 random dense polynomials per degree						
64	116302	87.2	1.0	4		
128	227842	90.5	.54	21		
191	340348	92.0	.42	26		
100 random sparse (10 monomials) polynomials per degree						
64	115850	86.2	.90	10		
128	226266	91.3	.36	11		
191	331966	92.1	.25	11		

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests			
d	n	t_0/t (%)	t_1/t_0	#F		
100 random dense polynomials per degree						
64	116302	87.2	1.0	4		
128	227842	90.5	.54	21		
191	340348	92.0	.42	26		
100 random sparse (10 monomials) polynomials per degree						
64	115850	86.2	.90	10		
128	226266	91.3	.36	11		
191	331966	92.1	.25	11		

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests

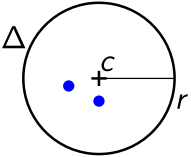
Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests			
d	n	t_0/t (%)	t_1/t_0	#F		
100 random dense polynomials per degree						
64	116302	87.2	1.0	4		
128	227842	90.5	.54	21		
191	340348	92.0	.42	26		
100 random sparse (10 monomials) polynomials per degree						
64	115850	86.2	.90	10		
128	226266	91.3	.36	11		
191	331966	92.1	.25	11		

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests

Power Sums

Let $\Delta = D(c, r)$, $p \in \mathbb{C}[z]$ of degree d
Let $\alpha_1, \ldots, \alpha_{d_\Delta}$ be the roots of p in Δ (non necessarily distinct)
Let $h \in \mathbb{Z}$



h-th power sum of p in Δ:

$$s_h(\Delta, p) = \alpha_1^h + \ldots + \alpha_{d_\Delta}^h$$

Remarks:

(i) $\#(\Delta, p) = s_0(\Delta, p)$

(ii) $\#(\Delta, p) = 0 \Rightarrow s_h(\Delta, p) = 0$ for any h

Let $p_\Delta = p(c + rz)$:

(iii) $\#(\Delta, p) = s_0(D(0, 1), p_\Delta)$

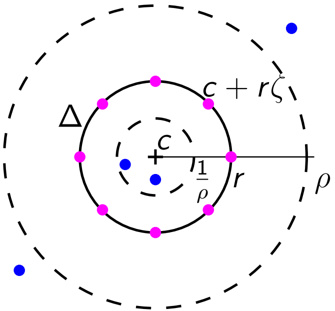
(iv) $\#(\Delta, p) = 0 \Rightarrow s_h(D(0, 1), p_\Delta) = 0$ for any h

Approximation of the Power Sums in $\Delta = D(c, r)$ (II)

Let $h \in \mathbb{Z}$, $q \in \mathbb{N}_*$ s.t. $q > h$ and define

$$s_h^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(c + r\zeta^g)}{p(c + r\zeta^g)}$$

where ζ is a primitive q-th root of unity.



Corollary: Let $\rho > 1$;
suppose $D(c, r)$ is ρ-isolated. Then

(i) $|s_h^* - s_h(D(c, r), p)| \leq \frac{d\theta^{q-h}}{1 - \theta^q}$ where $\theta = \frac{1}{\rho}$

(ii) Fix $e > 0$. If $q = \lceil \log_\theta(\frac{e}{d+e}) \rceil + h$ then $|s_h^* - s_h(D(c, r), p)| \leq e$.

Unsure Exclusion Test

$\widetilde{P^0}(p, \Delta, k)$

Input: $p \in \mathbb{C}[z]$ of degree d, Δ a disk, k an integer ≥ 0
Output: in { **true**, **can not decide** }

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
1. $e \leftarrow 1/4$, $\theta \leftarrow 1/\rho$
2. $q \leftarrow \lceil \log_\theta(\frac{e}{d+e}) \rceil + k$
3. evaluate p and p' at $c + r\zeta^g$ for $g = 0, \ldots, q-1$
4. **for** $h = 0, \ldots, k$ **do**
5. compute $s_h^* = \frac{r}{q} \sum_{g=0}^{q-1} \zeta^{g(h+1)} \frac{p'(c+r\zeta^g)}{p(c+r\zeta^g)}$
6. **if** $0 \notin D(s_h^*, e)$ **then**
7. **return can not decide**
8. **return true**

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests, $k=0$		$\widetilde{P}^0$-tests, $k=1$		$\widetilde{P}^0$-tests, $k=2$	
d	n	t_0/t (%)	t_1/t_0	#F	t'_1/t_0	#F'	t''_1/t_0	#F"
100 random dense polynomials per degree								
64	116302	87.2	1.0	4	1.0	0	1.1	0
128	227842	90.5	.54	21	.57	0	.59	0
191	340348	92.0	.42	26	.43	1	.45	0
100 random sparse (10 monomials) polynomials per degree								
64	115850	86.2	.90	10	.95	0	.98	0
128	226266	91.3	.36	11	.37	0	.40	0
191	331966	92.1	.25	11	.26	2	.28	0

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests with $k=0$

t'_1: time spent in $\widetilde{P}^0$-tests with $k=1$
#F': nb of wrong res. in $\widetilde{P}^0$-tests with $k=1$
t''_1: time spent in $\widetilde{P}^0$-tests with $k=2$
#F": nb of wrong res. in $\widetilde{P}^0$-tests with $k=2$

Unsure Exclusion Test: Experiments

		C^0-tests	$\widetilde{P}^0$-tests, $k=0$		$\widetilde{P}^0$-tests, $k=1$		$\widetilde{P}^0$-tests, $k=2$	
d	n	t_0/t (%)	t_1/t_0	#F	t'_1/t_0	#F'	t''_1/t_0	#F"
100 random dense polynomials per degree								
64	116302	87.2	1.0	4	1.0	0	1.1	0
128	227842	90.5	.54	21	.57	0	.59	0
191	340348	92.0	.42	26	.43	1	.45	0
100 random sparse (10 monomials) polynomials per degree								
64	115850	86.2	.90	10	.95	0	.98	0
128	226266	91.3	.36	11	.37	0	.40	0
191	331966	92.1	.25	11	.26	2	.28	0

Legend: d: degree
n: total number of exclusion tests
t: sequential time of `Ccluster`
t_0: time spent in C^0-tests
t_1: time spent in $\widetilde{P}^0$-tests
#F: nb of wrong res. in $\widetilde{P}^0$-tests with $k=0$

t'_1: time spent in $\widetilde{P}^0$-tests with $k=1$
#F': nb of wrong res. in $\widetilde{P}^0$-tests with $k=1$
t''_1: time spent in $\widetilde{P}^0$-tests with $k=2$
#F": nb of wrong res. in $\widetilde{P}^0$-tests with $k=2$

Subdivision Algorithm with Unsure Exclusion Test

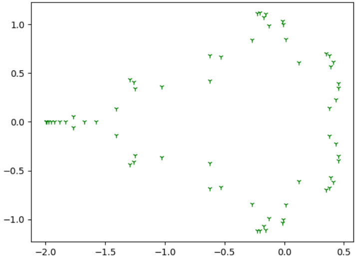
- for the (global) Root Clustering Problem
- uses $\widetilde{P^0}$-test with $k = 2$
- always terminates, but may fail: in this case, reports failure
- implemented in `C` within `Ccluster`: `CclusterF`

Subdivision Algorithm with Unsure Exclusion Test

- for the (global) Root Clustering Problem
- uses $\widetilde{P^0}$-test with $k = 2$
- always terminates, but may fail: in this case, reports failure
- implemented in `C` within `Ccluster`: `CclusterF`
- faster for sparse and procedural polynomial

Procedure: $\texttt{Mandelbrot}_k(z)$
Input: $k \in \mathbb{N}^*, z \in \mathbb{C}$
Output: $\alpha \in \mathbb{C}$

1. **if** $k = 1$ **then**
2. **return** z
3. **else**
4. **return** $z\texttt{Mandelbrot}_{k-1}(z)^2 + 1$



$k = 6$ (deg $= 63$)

Subdivision Algorithm with Unsure Exclusion Test

Results:

	Ccluster	CclusterF		
d	t	#Fails	t'	t'/t (%)
100 random dense polynomials per degree				
64	31.5	0	41.2	130
128	222	0	149	67.3
191	665	0	340	51.1
100 random sparse (10 monomials) polynomials per degree				
64	27.9	0	31.7	113
128	216	0	100	46.3
191	638	0	209	32.7
Mandelbrot polynomials				
127	3.46	0	0.56	16.1
255	18.4	0	1.79	9.70
511	118	0	7.61	6.42

Legend: t, t': seq. times in s. on an
Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz machine with Linux

Subdivision Algorithm with Unsure Exclusion Test

Results:

	Ccluster	CclusterF		
d	t	#Fails	t'	t'/t (%)
100 random dense polynomials per degree				
64	31.5	0	41.2	130
128	222	0	149	67.3
191	665	0	340	51.1
100 random sparse (10 monomials) polynomials per degree				
64	27.9	0	31.7	113
128	216	0	100	46.3
191	638	0	209	32.7
Mandelbrot polynomials				
127	3.46	0	0.56	16.1
255	18.4	0	1.79	9.70
511	118	0	7.61	6.42

Legend: t, t': seq. times in s. on an Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz machine with Linux

Subdivision Algorithm with Unsure Exclusion Test

Results:

	Ccluster	CclusterF		
d	t	#Fails	t'	t'/t (%)
100 random dense polynomials per degree				
64	31.5	0	41.2	130
128	222	0	149	67.3
191	665	0	340	51.1
100 random sparse (10 monomials) polynomials per degree				
64	27.9	0	31.7	113
128	216	0	100	46.3
191	638	0	209	32.7
Mandelbrot polynomials				
127	3.46	0	0.56	16.1
255	18.4	0	1.79	9.70
511	118	0	7.61	6.42

Legend: t, t': seq. times in s. on an
Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz machine with Linux

Remarks and Future Works

- Ongoing work!
- Probabilistic and deterministic support of our $\widetilde{P^0}$-test

Remarks and Future Works

- Ongoing work!
- Probabilistic and deterministic support of our $\widetilde{P^0}$-test

Thank you!