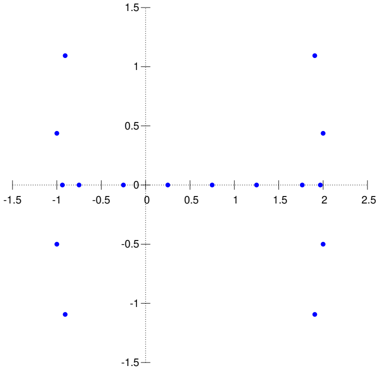
Complex Root Clustering

Joint works with

V. Pan[1], M. Pouget[2], C. Yap[3]

[1] Lehman College, City University of New York, USA
[2] INRIA Nancy - Grand Est, France
[3] Courant Institute of Mathematical Sciences, New York University, USA

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d,

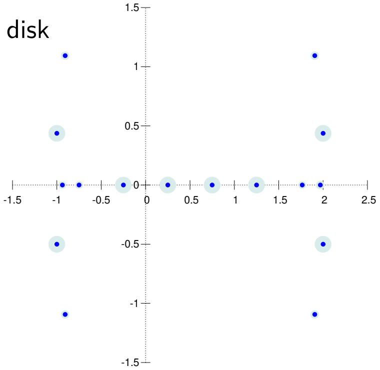


Example: Bernoulli pol. of deg 16: $\sum B_i z^i$, where B_i are Bernoulli numb.

Root approximation problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$

Output: d disks $\Delta^1, \ldots, \Delta^d$ of radii $\leq \varepsilon$
each containing a root of f
each complex root contained in a disk

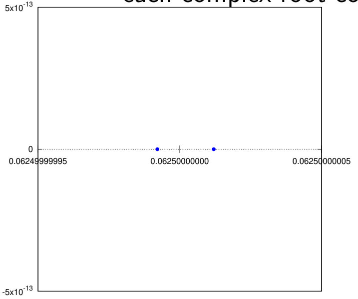


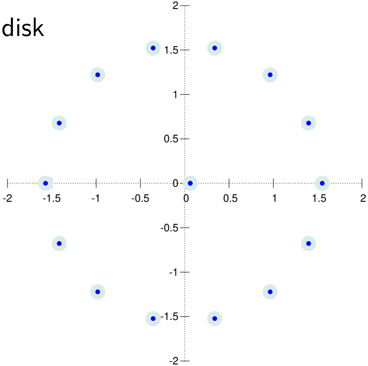
Example: Bernoulli pol. of deg 16: $\sum B_i z^i$, where B_i are Bernoulli numb.

Root approximation problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$

Output: d disks $\Delta^1, \dots, \Delta^d$ of radii $\leq \varepsilon$
each containing a root of f
each complex root contained in a disk



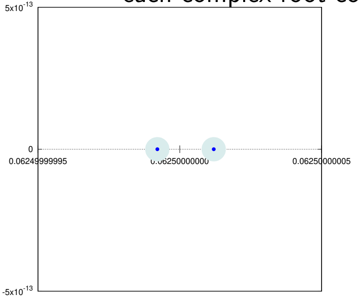


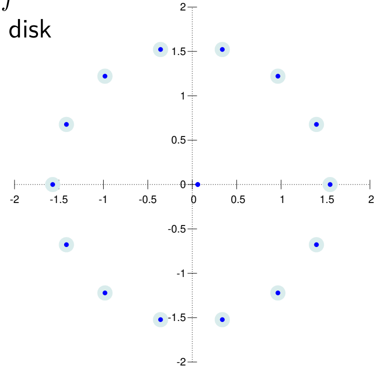
Example: Mignotte-like polynomial: $z^d - (2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root isolation problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, (a size $\varepsilon > 0$)

Output: ℓ pairwise disjoint disks $\Delta^1, \ldots, \Delta^\ell$ (of radii $\leq \varepsilon$)
each containing a unique root of f
each complex root contained in a disk



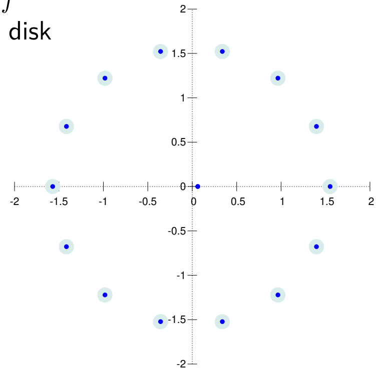


Example: Mignotte-like polynomial: $z^d - (2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root isolation problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, (a size $\varepsilon > 0$)

Output: ℓ pairwise disjoint disks $\Delta^1, \ldots, \Delta^\ell$ (of radii $\leq \varepsilon$)
each containing a unique root of f
each complex root contained in a disk



Problem: deciding Zero
Are two roots equal or not?

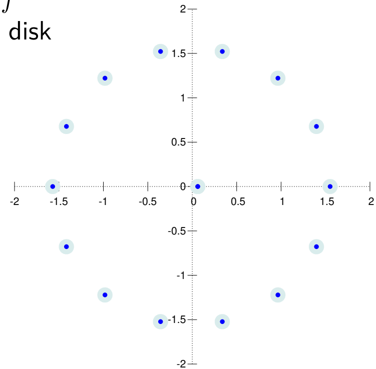
Example: Mignotte-like polynomial: $z^d - (2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root isolation problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, (a size $\varepsilon > 0$
and either a lower bound for the sep. of the roots, or $f \in \mathbb{Z}[z]$

Output: ℓ pairwise disjoint disks $\Delta^1, \ldots, \Delta^\ell$ (of radii $\leq \varepsilon$)
each containing a unique root of f
each complex root contained in a disk

Problem: deciding Zero
Are two roots equal or not?

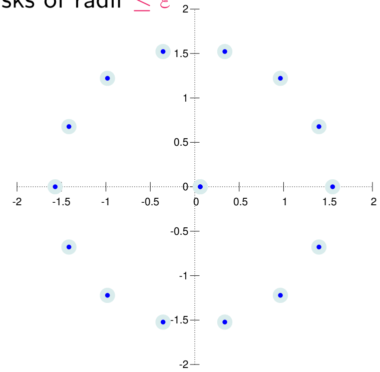


Example: Mignotte-like polynomial: $z^d - (2^\sigma z - 1)^2$, where $d = 16, \sigma = 4$

Root clustering problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$

Output: ℓ pairs $(\Delta^1, m^1), \ldots, (\Delta^\ell, m^\ell)$ where:
- the Δ^j's are pairwise disjoint disks of radii $\leq \varepsilon$
- $m^j = \#(\Delta^j, f)$
- $Z(\mathbb{C}, f) \subseteq \bigcup \Delta^j$

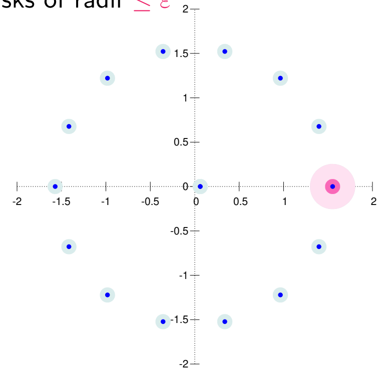


Notations: $Z(\mathcal{S}, f) :=$ roots of f in set $\mathcal{S}$
$\#(\mathcal{S}, f) :=$ nb of roots of f in set $\mathcal{S}$, count. with mult.

Root clustering problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$

Output: ℓ pairs $(\Delta^1, m^1), \dots, (\Delta^\ell, m^\ell)$ where:
- the Δ^j's are pairwise disjoint disks of radii $\leq \varepsilon$
- $m^j = \#(\Delta^j, f) = \#(3\Delta^j, f)$
- $Z(\mathbb{C}, f) \subseteq \bigcup \Delta^j$



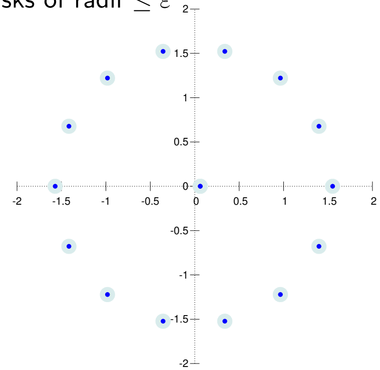
Notations: $Z(\mathcal{S}, f)$:= roots of f in set $\mathcal{S}$
$\#(\mathcal{S}, f)$:= nb of roots of f in set $\mathcal{S}$, count. with mult.

Global root clustering problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$

Output: ℓ pairs $(\Delta^1, m^1), \ldots, (\Delta^\ell, m^\ell)$ where:

- the Δ^j's are pairwise disjoint disks of radii $\leq \varepsilon$
- $m^j = \#(\Delta^j, f) = \#(3\Delta^j, f)$
- $Z(\mathbb{C}, f) \subseteq \bigcup \Delta^j$



Notations: $Z(\mathcal{S}, f) :=$ roots of f in set $\mathcal{S}$

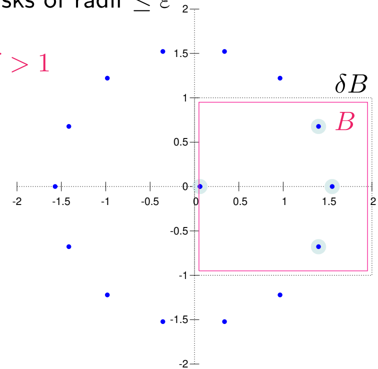
$\#(\mathcal{S}, f) :=$ nb of roots of f in set $\mathcal{S}$, count. with mult.

Local root clustering problem

Input: a polynomial $f \in \mathbb{C}[z]$ of degree d, a size $\varepsilon > 0$
a Region of Interest (RoI) B (a box)

Output: ℓ pairs $(\Delta^1, m^1), \dots, (\Delta^\ell, m^\ell)$ where:
- the Δ^j's are pairwise disjoint disks of radii $\leq \varepsilon$
- $m^j = \#(\Delta^j, f) = \#(3\Delta^j, f)$
- $Z(B, f) \subseteq \bigcup \Delta^j \subseteq Z(\delta B, f)$, $\delta > 1$

Problem: deciding Zero
Is a root on ∂B?



Notations: $Z(\mathcal{S}, f) :=$ roots of f in set $\mathcal{S}$
$\#(\mathcal{S}, f) :=$ nb of roots of f in set $\mathcal{S}$, count. with mult.

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
- by an oracle, when $f \in \mathbb{C}[z]$:

Definition: Oracle for f: function $\mathcal{O}_f : \mathbb{Z} \rightarrow \mathbb{C}[z]$ s.t. $\|\mathcal{O}_f(L) - f\|_\infty < 2^{-L}$

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
  $\rightarrow$ root isolation is meaningful
- by an oracle, when $f \in \mathbb{C}[z]$:
  $\rightarrow$ if lower bound for root's separation: isolation is meaningful

Definition: Oracle for f: function $\mathcal{O}_f : \mathbb{Z} \rightarrow \mathbb{C}[z]$ s.t. $\|\mathcal{O}_f(L) - f\|_\infty < 2^{-L}$

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
  $\rightarrow$ root isolation is meaningful
- by an oracle, when $f \in \mathbb{C}[z]$:
  $\rightarrow$ if lower bound for root's separation: isolation is meaningful
  $\rightarrow$ otherwise only root clustering is meaningful

Definition: Oracle for f: function $\mathcal{O}_f : \mathbb{Z} \rightarrow \mathbb{C}[z]$ s.t. $\|\mathcal{O}_f(L) - f\|_\infty < 2^{-L}$

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
  $\rightarrow$ root isolation is meaningful
- by an oracle, when $f \in \mathbb{C}[z]$:
  $\rightarrow$ if lower bound for root's separation: isolation is meaningful
  $\rightarrow$ otherwise only root clustering is meaningful

Pros of Local root finding:

- the RoI B may contain a few roots of f

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
  $\rightarrow$ root isolation is meaningful
- by an oracle, when $f \in \mathbb{C}[z]$:
  $\rightarrow$ if lower bound for root's separation: isolation is meaningful
  $\rightarrow$ otherwise only root clustering is meaningful

Pros of Local root finding:

- the RoI B may contain a few roots of f
- $|\alpha_1(f)|$ increases with $\frac{\|f\|_\infty}{|\mathrm{lcf}(f)|}$

Notations: $\alpha_1(f)$: a root of f with greatest norm
$\mathrm{lcf}(f)$: the leading coefficient of f

Root finding problems . . .

f can be represented:

- exactly, when $f \in \mathbb{Z}[z]$ (or $\mathbb{Q}[z]$):
  $\rightarrow$ root isolation is meaningful
- by an oracle, when $f \in \mathbb{C}[z]$:
  $\rightarrow$ if lower bound for root's separation: isolation is meaningful
  $\rightarrow$ otherwise only root clustering is meaningful

Pros of Local root finding:

- the RoI B may contain a few roots of f
- $|\alpha_1(f)|$ increases with $\dfrac{\|f\|_\infty}{|\mathrm{lcf}(f)|}$

Problem: deciding Zero
When f is oracle. Is $\mathrm{lcf}(f) = 0$? (Is $\|\alpha_1(f)\| = +\infty$?)
f has nominal degree d. What is it's true degree?

Notations: $\alpha_1(f)$: a root of f with greatest norm
$\mathrm{lcf}(f)$: the leading coefficient of f

Root finding problems . . . and algorithms

- Root approximation:
  Schönage-Pan[Pan02], global, no implementation

[Pan02] Victor Y Pan.
Univariate polynomials: nearly optimal algorithms for numerical factorization and root-finding.
Journal of Symbolic Computation, 33(5):701–733, 2002.

Root finding problems . . . and algorithms

- Root approximation:
  Schönage-Pan[Pan02], global, no implementation
  Ehrlich iterations, global, implemented[BR14] `MPSolve`

[BR14] Dario A Bini and Leonardo Robol.
Solving secular and polynomial equations: A multiprecision algorithm.
Journal of Computational and Applied Mathematics, 272:276–292, 2014.

Root finding problems . . . and algorithms

- Root approximation:
  Schönage-Pan[Pan02], global, no implementation
  Ehrlich iterations, global, implemented[BR14] `MPSolve`
- Root isolation, real case
  Sagraloff et al.[SM16], local, implemented[KRS16]

[KRS16] Alexander Kobel, Fabrice Rouillier, and Michael Sagraloff.
Computing real roots of real polynomials ... and now for real!
ISSAC '16, pages 303–310, New York, NY, USA, 2016. ACM.

[SM16] Michael Sagraloff and Kurt Mehlhorn.
Computing real roots of real polynomials.
J. of Symb. Comp., 73:46–86, 2016.

Root finding problems . . . and algorithms

- Root approximation:
  Schönage-Pan[Pan02], global, no implementation
  Ehrlich iterations, global, implemented[BR14] `MPSolve`
- Root isolation, real case
  Sagraloff et al.[SM16], local, implemented[KRS16]
- Root clustering:
  Becker et al.[BSS$^+$16]: local, implemented[IPY18]

[BSS$^+$16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *ISSAC 16*, pages 71–78. ACM, 2016.

[IPY18] Rémi Imbach, Victor Y. Pan, and Chee Yap.
Implementation of a near-optimal complex root clustering algorithm.
In *Mathematical Software – ICMS 2018*, pages 235–244, Cham, 2018.

Root finding problems . . . and algorithms

Bit complexity:

- Root approximation:
  Schönage-Pan[Pan02], global, no implementation $\widetilde{O}(d^2\sigma)$
  Ehrlich iterations, global, implemented[BR14] `MPSolve` ?
- Root isolation, real case
  Sagraloff et al.[SM16], local, implemented[KRS16]
  $\widetilde{O}(d^2(d+\sigma))$
- Root clustering:
  Becker et al.[BSS+16]: local, implemented[IPY18]
  $\widetilde{O}(d^2(d+\sigma))$

Benchmark problem: Isolate all roots of $f \in \mathbb{Z}[z]$ squarefree
d: degree, σ: $\log \|f\|_\infty$

Notation: $\widetilde{O}$: O without logarithmic factor

Multivariate root finding

$$\left\{\begin{array}{lcl} p_1(z_1, z_2, \ldots, z_n) & = & 0 \\ p_2(z_1, z_2, \ldots, z_n) & = & 0 \\ \ldots & & \\ p_n(z_1, z_2, \ldots, z_n) & = & 0 \end{array}\right.$$

Multivariate root finding, bivariate case

$$\left\{\begin{array}{lcl} p_1(z_1,z_2) & = & 0 \\ p_2(z_1,z_2) & = & 0 \end{array}\right., p_i \in \mathbb{Z}[z_1,z_2]$$

Multivariate root finding, bivariate case

$$\left\{ \begin{array}{rcl} p_1(z_1,z_2) & = & 0 \\ p_2(z_1,z_2) & = & 0 \end{array} \right. , p_i \in \mathbb{Z}[z_1,z_2]$$

$$\Big\downarrow \text{ symbolic step}$$

$$\left\{ \ldots, \quad \left\{ \begin{array}{rcl} f_1(z_1) & = & 0 \\ f_2(z_1,z_2) & = & 0 \end{array} \right. , \deg_{z_i}(f_i) \geq 1, \quad \ldots \right\}$$

Multivariate root finding, bivariate case

$$\begin{cases} p_1(z_1, z_2) &= 0 \\ p_2(z_1, z_2) &= 0 \end{cases}, p_i \in \mathbb{Z}[z_1, z_2]$$

$\downarrow$ symbolic step

$$\left\{ \ldots, \begin{cases} f_1(z_1) &= 0 \\ f_2(z_1, z_2) &= 0 \end{cases}, \deg_{z_i}(f_i) \geq 1, \quad \ldots \right\}$$

$\downarrow$ numeric step: univariate root finding

1. find roots of $f_1 \in \mathbb{Z}[z_1]$
2. find roots of $f_2(\alpha, z_2) \in \mathbb{C}[z_2]$ for α root of f_1;
   $f_2(\alpha, z_2)$ known as an oracle

Multivariate root finding, bivariate case

$$\left\{\begin{array}{lcl} p_1(z_1,z_2) & = & 0 \\ p_2(z_1,z_2) & = & 0 \end{array}\right., p_i \in \mathbb{Z}[z_1,z_2]$$

$\downarrow$ symbolic step

$$\left\{ \dots, \quad \left\{\begin{array}{lcl} f_1(z_1) & = & 0 \\ f_2(z_1,z_2) & = & 0 \end{array}\right., \deg_{z_i}(f_i) \geq 1, \quad \dots \right\}$$

$\downarrow$ numeric step: univariate root finding

1. find roots of $f_1 \in \mathbb{Z}[z_1]$
2. find roots of $f_2(\alpha, z_2) \in \mathbb{C}[z_2]$ for α root of f_1;
   $f_2(\alpha, z_2)$ known as an oracle

Problem: deciding Zero
Is $\text{lcf}(f_2(\alpha, z_2)) = 0$?
$f_2(\alpha, z_2)$ has nominal degree $\deg_{z_2}(f_2)$. What is it's true degree?

Multivariate root finding, bivariate case

$$\left\{\begin{array}{lcl} p_1(z_1,z_2) & = & 0 \\ p_2(z_1,z_2) & = & 0 \end{array}\right., p_i \in \mathbb{Z}[z_1,z_2]$$

$$\Big\downarrow \text{ symbolic step}$$

$$\left\{ \ldots, \quad \left\{\begin{array}{lcl} f_1(z_1) & = & 0 \\ f_2(z_1,z_2) & = & 0 \end{array}\right., \deg_{z_i}(f_i) \geq 1, \quad \ldots \right\}$$

$$\Big\downarrow \text{ numeric step: univariate root finding}$$

1. find roots of $f_1 \in \mathbb{Z}[z_1]$
2. find roots of $f_2(\alpha, z_2) \in \mathbb{C}[z_2]$ for α root of f_1;
   $f_2(\alpha, z_2)$ known as an oracle

Usual assumption: $f_1 = f_2 = 0$ is regular (ensured by symbolic step)

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

ToC

0 - Univariate case:

[BSS+16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *ISSAC 16*, pages 71–78. ACM, 2016.

ToC

0 - Univariate case:

1 - Multivariate triangular case

[IPY20] Rémi Imbach, Marc Pouget, and Chee Yap.
Clustering complex zeros of triangular systems of polynomials.
Mathematics in Computer Science, pages 1–22, 2020.

$$\begin{cases} f_1(z_1) & = & 0 \\ f_2(z_1, z_2) & = & 0 \end{cases}, \deg_{z_i}(f_i) \geq 1,$$

ToC

0 - Univariate case:

1 - Multivariate triangular case

[IPY20] Rémi Imbach, Marc Pouget, and Chee Yap.
Clustering complex zeros of triangular systems of polynomials.
Mathematics in Computer Science, pages 1–22, 2020.

$$\begin{cases} f_1(z_1) & = & 0 \\ f_2(z_1, z_2) & = & 0 \end{cases}, \deg_{z_i}(f_i) \geq 1,$$

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

ToC

0 - Univariate case:

1 - Multivariate triangular case

[IPY20] Rémi Imbach, Marc Pouget, and Chee Yap.
Clustering complex zeros of triangular systems of polynomials.
Mathematics in Computer Science, pages 1–22, 2020.

$$\begin{cases} f_1(z_1) & = & 0 \\ f_2(z_1, z_2) & = & 0 \end{cases}, \deg_{z_i}(f_i) \geq 1,$$

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

Weaker assumption: $f_1 = f_2 = 0$ is weakly regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

ToC

0 - Univariate case:

1 - Multivariate triangular case

2 - Back to univariate case

[IP20] Rémi Imbach and Victor Y. Pan.
New progress in univariate polynomial root finding.
ISSAC '20, page 249–256, New York, NY, USA, 2020. ACM.

Soft Exact computation

Numerical methods are fast
but not robust to Zero problems

Zero problems can be tackled with exact computation
in only a few cases

Soft exact computation (for instance local root clustering)
is a middle path?
avoids Zero problem

Oracle numbers and polynomials

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

Oracle for α: function $\mathcal{O}_\alpha : \mathbb{Z} \rightarrow \square\mathbb{C}$
s.t. $\alpha \in \mathcal{O}_\alpha(L)$ and $w(\mathcal{O}_\alpha(L)) \leq 2^{-L}$

Notations: $\square\mathbb{C}$: set of complex interval

Oracle numbers and polynomials

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

Oracle for α: function $\mathcal{O}_\alpha : \mathbb{Z} \rightarrow \square\mathbb{C}$
s.t. $\alpha \in \mathcal{O}_\alpha(L)$ and $w(\mathcal{O}_\alpha(L)) \leq 2^{-L}$

Let $f \in \mathbb{C}[z]$

Oracle for f: function $\mathcal{O}_f : \mathbb{Z} \rightarrow \square\mathbb{C}[z]$
s.t. $f \in \mathcal{O}_f(L)$ and $w(\mathcal{O}_f(L)) \leq 2^{-L}$
$\simeq$ oracles for the coeffs of f

Notations: $\square\mathbb{C}$: set of complex interval
$\square\mathbb{C}[z]$: polynomials with coefficients in $\square\mathbb{C}$

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \dots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:

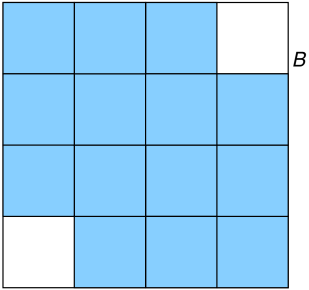
Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \ldots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:



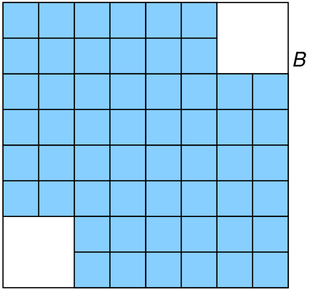
Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \ldots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:



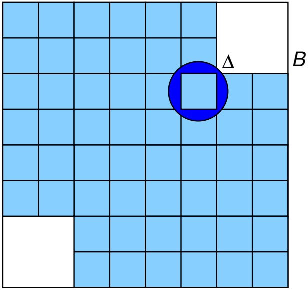
Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS$^+$16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \dots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:



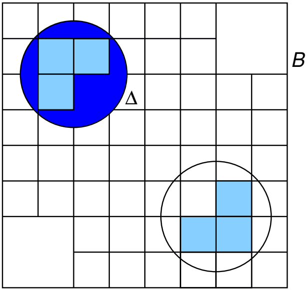
Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \ldots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:



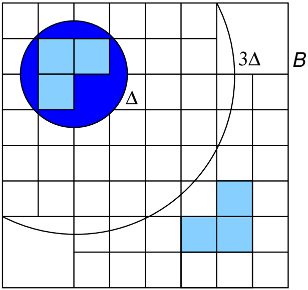
Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \dots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \Rightarrow \#(\Delta, f) = 0$

Subdivision approach:



Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS$^+$16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \ldots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0 \;\Leftrightarrow P^*(\Delta, \mathcal{O}_f) = 0$

Subdivision approach:

Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

Outline of [BSS+16]

Root counter: $P^* : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0, \dots, d\}$
$P^*(\Delta, \mathcal{O}_f) \geq 0 \Rightarrow \#(\Delta, f) = m$

Exclusion test: $P^0 : (\Delta, \mathcal{O}_f) \mapsto m \in \{-1, 0\}$
$P^0(\Delta, \mathcal{O}_f) = 0$

Subdivision approach:

Bounding the depth of the subdivision tree:
no root in $2B \Rightarrow P^*(\Delta(B), \mathcal{O}_f)$ returns 0

Notations: $\#(S, f)$: sum of multiplicities of roots of f in S

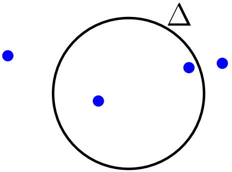
The Pellet's test

Pellet's Theorem: Let Δ be a disc centered in c and radius r.
Let $f \in \mathbb{C}[z]$ and $f_\Delta = f(c + rz)$.

If $\exists\ 0 \leq m \leq d$ s.t.

$$|(f_\Delta)_m| > \sum_{i \neq k} |(f_\Delta)_i| \qquad (1)$$

then f has exactly m roots in Δ.



Notations: $(f)_m$: coeff. of the monomial of degree m of f

The Pellet's test

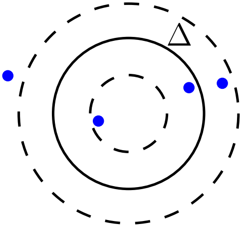
Pellet's Theorem: Let Δ be a disc centered in c and radius r.
Let $f \in \mathbb{C}[z]$ and $f_\Delta = f(c + rz)$.

If $\exists\ 0 \leq m \leq d$ s.t.

$$|(f_\Delta)_m| > \sum_{i \neq k} |(f_\Delta)_i| \qquad (1)$$

then f has exactly m roots in Δ.

If f has no root in this annulus $\rightarrow$
$\exists m$ s.t. eq. (1) holds.



Notations: $(f)_m$: coeff. of the monomial of degree m of f

The Pellet's test

Pellet's Theorem: Let Δ be a disc centered in c and radius r.
Let $f \in \mathbb{C}[z]$ and $f_\Delta = f(c + rz)$.

If $\exists\ 0 \leq m \leq d$ s.t.

$$|(f_\Delta)_m| > \sum_{i \neq k} |(f_\Delta)_i| \quad (1)$$

then f has exactly m roots in Δ.

PelletTest(Δ, f) *//Output in $\{-1, 0, 1, \ldots, d\}$*

1. compute f_Δ
2. **for** m **from** 0 **to** d **do**
3. **if** $|(f_\Delta)_m| > \sum_{i \neq k} |(f_\Delta)_i|$
4. **return** m *//m roots (with mult.) in Δ*
5. **return** -1 *//roots near the boundary of Δ*

The soft Pellet's test: for interval polynomials

Pellet's Theorem: Let Δ be a complex disc centered in c and radius r.
Let $\square f \in \square\mathbb{C}[z]$ and $\square f_\Delta = \square f(c + rz)$.

If $\exists\ 0 \leq m \leq d$ s.t.

$$|(\square f_\Delta)_m| > \sum_{i \neq k} |(\square f_\Delta)_i|$$

then any $f \in \square f$ has exactly m roots in Δ.

The soft Pellet's test: for interval polynomials

Pellet's Theorem: Let Δ be a complex disc centered in c and radius r.
Let $\square f \in \square\mathbb{C}[z]$ and $\square f_\Delta = \square f(c + rz)$.

If $\exists\ 0 \leq m \leq d$ s.t.

$$|(\square f_\Delta)_m| > \sum_{i \neq k} |(\square f_\Delta)_i|$$

then any $f \in \square f$ has exactly m roots in Δ.

SoftPelletTest($\Delta, \square f$) *//Output in* $\{-2, -1, 0, 1, \dots, d\}$

1. compute $\square f_\Delta$
2. **for** m **from** 0 **to** d **do**
3. **if** $|(\square f_\Delta)_m| > \sum_{i \neq k} |(\square f_\Delta)_i|$
4. **return** m *//any $f \in \square f$ has m roots in Δ*
5. **if** $|(\square f_\Delta)_m|$ and $\sum_{i \neq k} |(\square f_\Delta)_i|$ overlap
6. **return** -2 *//not enough precision on $\square f$*
7. **return** -1 *//roots near the boundary of Δ*

The soft Pellet's test: for oracle polynomials

Loop on precision:

SoftPelletTest($\Delta, \square f$) *//Output in* $\{-2, -1, 0, 1, \ldots, d\}$

1. compute $\square f_\Delta$
2. **for** m **from** 0 **to** d **do**
3. **if** $|(\square f_\Delta)_m| > \sum_{i \neq k} |(\square f_\Delta)_i|$
4. **return** m *//any* $f \in \square f$ *has* m *roots in* Δ
5. **if** $|(\square f_\Delta)_m|$ and $\sum_{i \neq k} |(\square f_\Delta)_i|$ overlap
6. **return** -2 *//not enough precision on* $\square f$
7. **return** -1 *//roots near the boundary of* Δ

The soft Pellet's test: for oracle polynomials

Loop on precision:

$P^*(\Delta, \mathcal{O}_f)$ *//Output in* $\{-1, 0, 1, \dots, d\}$

1. $L \leftarrow 53$, $\square f \leftarrow \mathcal{O}_f(L)$, $m \leftarrow$SoftPelletTest$(\Delta, \square f)$
2. **while** $m = -2$ **do**
3. $L \leftarrow 2L$, $\square f \leftarrow \mathcal{O}_f(L)$, $m \leftarrow$SoftPelletTest$(\Delta, \square f)$
4. **return** m

SoftPelletTest$(\Delta, \square f)$ *//Output in* $\{-2, -1, 0, 1, \dots, d\}$

1. compute $\square f_\Delta$
2. **for** m **from** 0 **to** d **do**
3. **if** $|(\square f_\Delta)_m| > \sum_{i \neq k} |(\square f_\Delta)_i|$
4. **return** m *//any* $f \in \square f$ *has* m *roots in* Δ
5. **if** $|(\square f_\Delta)_m|$ and $\sum_{i \neq k} |(\square f_\Delta)_i|$ overlap
6. **return** -2 *//not enough precision on* $\square f$
7. **return** -1 *//roots near the boundary of* Δ

Univariate root clustering algorithms

ClusterOracle: solves the Local Clustering Problem (LCP) in 1D ([BSS+16])

P^* embedded in a subdivision framework
accepts oracle polynomials in input

[BSS+16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *ISSAC 16*, pages 71–78. ACM, 2016.

Univariate root clustering algorithms

ClusterOracle: solves the Local Clustering Problem (LCP) in 1D ([BSS+16])

P^* embedded in a subdivision framework
accepts oracle polynomials in input

ClusterInterval: solves the LCP in 1D
Input: interval polynomial
Output: a flag in {**success**,**fail**}, a list of natural clusters
SoftPelletTest embedded in a subdivision framework
returns **fail** when SoftPelletTest returns -2

[BSS+16] Ruben Becker, Michael Sagraloff, Vikram Sharma, Juan Xu, and Chee Yap.
Complexity analysis of root clustering for a complex polynomial.
In *ISSAC 16*, pages 71–78. ACM, 2016.

ToC

0 - Univariate case:

1 - Multivariate triangular case

[IPY20] Rémi Imbach, Marc Pouget, and Chee Yap.
Clustering complex zeros of triangular systems of polynomials.
Mathematics in Computer Science, pages 1–22, 2020.

$$\begin{cases} f_1(z_1) & = & 0 \\ f_2(z_1, z_2) & = & 0 \end{cases}, \deg_{z_i}(f_i) \geq 1, \quad f_i \in \mathbb{Q}[z_1, z_2]$$

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

Local solution Clustering Problem (LCP)

Input: a polynomial map $\boldsymbol{f} : \mathbb{C}^2 \to \mathbb{C}^2$,
a polybox $\boldsymbol{B} \subset \mathbb{C}^2$, the Region of Interest (RoI),
$\varepsilon > 0$

Output:

Notations: $\boldsymbol{f} = (f_1, f_2)$,
$\boldsymbol{B} = (B_1, B_2)$ where the B_i's are square complex boxes

Local solution Clustering Problem (LCP)

Input: a polynomial map $\boldsymbol{f} : \mathbb{C}^2 \to \mathbb{C}^2$,
a polybox $\boldsymbol{B} \subset \mathbb{C}^2$, the Region of Interest (RoI),
$\varepsilon > 0$

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

- the $\boldsymbol{\Delta}^j$s are pairwise disjoint polydiscs of radius $r(\boldsymbol{\Delta}^j) \leq \varepsilon$,

Notations: $\boldsymbol{f} = (f_1, f_2)$,
$\boldsymbol{B} = (B_1, B_2)$ where the B_i's are square complex boxes
$\boldsymbol{\Delta}^j = (\Delta^j_1, \Delta^j_2)$ where the Δ^j_i's are complex discs
$r(\boldsymbol{\Delta}^j) = \max_i r(\Delta^j_i)$

Local solution Clustering Problem (LCP)

Input: a polynomial map $\boldsymbol{f} : \mathbb{C}^2 \rightarrow \mathbb{C}^2$,
a polybox $\boldsymbol{B} \subset \mathbb{C}^2$, the Region of Interest (RoI),
$\varepsilon > 0$

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

- the $\boldsymbol{\Delta}^j$s are pairwise disjoint polydiscs of radius $r(\boldsymbol{\Delta}^j) \leq \varepsilon$,
- $m^j = \#(\boldsymbol{\Delta}^j, \boldsymbol{f}) = \#(3\boldsymbol{\Delta}^j, \boldsymbol{f})$ for all $1 \leq j \leq \ell$, and

Notations: $\boldsymbol{f} = (f_1, f_2)$,
$\boldsymbol{B} = (B_1, B_2)$ where the B_i's are square complex boxes
$\boldsymbol{\Delta}^j = (\Delta_1^j, \Delta_2^j)$ where the Δ_i^j's are complex discs
$r(\boldsymbol{\Delta}^j) = \max_i r(\Delta_i^j)$
$\#(S, \boldsymbol{f})$: nb. of sols (with mult.) of $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ in S

Local solution Clustering Problem (LCP)

Input: a polynomial map $\boldsymbol{f} : \mathbb{C}^2 \rightarrow \mathbb{C}^2$,
a polybox $\boldsymbol{B} \subset \mathbb{C}^2$, the Region of Interest (RoI),
$\varepsilon > 0$

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

- the $\boldsymbol{\Delta}^j$s are pairwise disjoint polydiscs of radius $r(\boldsymbol{\Delta}^j) \leq \varepsilon$,
- $m^j = \#(\boldsymbol{\Delta}^j, \boldsymbol{f}) = \#(3\boldsymbol{\Delta}^j, \boldsymbol{f})$ for all $1 \leq j \leq \ell$, and
- $Z(\boldsymbol{B}, \boldsymbol{f}) \subseteq \bigcup_{j=1}^{\ell} Z(\boldsymbol{\Delta}^j, \boldsymbol{f}) \subseteq Z(\delta \boldsymbol{B}, \boldsymbol{f})$ for $\delta > 1$

Notations: $\boldsymbol{f} = (f_1, f_2)$,
$\boldsymbol{B} = (B_1, B_2)$ where the B_i's are square complex boxes
$\boldsymbol{\Delta}^j = (\Delta_1^j, \Delta_2^j)$ where the Δ_i^j's are complex discs
$r(\boldsymbol{\Delta}^j) = \max_i r(\Delta_i^j)$
$\#(S, \boldsymbol{f})$: nb. of sols (with mult.) of $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ in S
$Z(S, \boldsymbol{f})$: sols of $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ in S

Local solution Clustering Problem (LCP)

Input: a polynomial map $\boldsymbol{f} : \mathbb{C}^2 \rightarrow \mathbb{C}^2$,
a polybox $\boldsymbol{B} \subset \mathbb{C}^2$, the Region of Interest (RoI),
$\varepsilon > 0$

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

- the $\boldsymbol{\Delta}^j$s are pairwise disjoint polydiscs of radius $r(\boldsymbol{\Delta}^j) \leq \varepsilon$,
- $m^j = \#(\boldsymbol{\Delta}^j, \boldsymbol{f}) = \#(3\boldsymbol{\Delta}^j, \boldsymbol{f})$ for all $1 \leq j \leq \ell$, and
- $Z(\boldsymbol{B}, \boldsymbol{f}) \subseteq \bigcup_{j=1}^{\ell} Z(\boldsymbol{\Delta}^j, \boldsymbol{f}) \subseteq Z(\delta \boldsymbol{B}, \boldsymbol{f})$ for $\delta > 1$

Definition: a pair $(\boldsymbol{\Delta}, m)$ is called natural cluster (relative to $\boldsymbol{f}$)
when it satisfies:

$$m = \#(\boldsymbol{\Delta}, \boldsymbol{f}) = \#(3\boldsymbol{\Delta}, \boldsymbol{f}) \geq 1$$

if $r(\boldsymbol{\Delta}) \leq \epsilon$, it is a natural ϵ-cluster

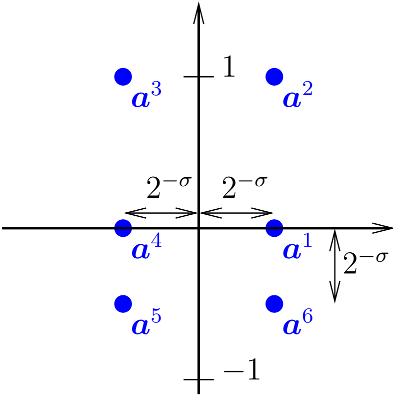
Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})\ (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^{\sigma} z_1^2)\ (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$\boldsymbol{a}^1 = (2^{-\sigma}, 0)$
$\boldsymbol{a}^2 = (2^{-\sigma}, 1)$
$\boldsymbol{a}^3 = (-2^{-\sigma}, 1)$
$\boldsymbol{a}^4 = (-2^{-\sigma}, 0)$
$\boldsymbol{a}^5 = (-2^{-\sigma}, -2^{-\sigma})$
$\boldsymbol{a}^6 = (2^{-\sigma}, -2^{-\sigma})$



Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})\ (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^{\sigma} z_1^2)\ (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$\boldsymbol{a}^1 = (2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^1, \boldsymbol{f}) = 1$

$\boldsymbol{a}^2 = (2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^2, \boldsymbol{f}) = 1$

$\boldsymbol{a}^3 = (-2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^3, \boldsymbol{f}) = 1$

$\boldsymbol{a}^4 = (-2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^4, \boldsymbol{f}) = 1$

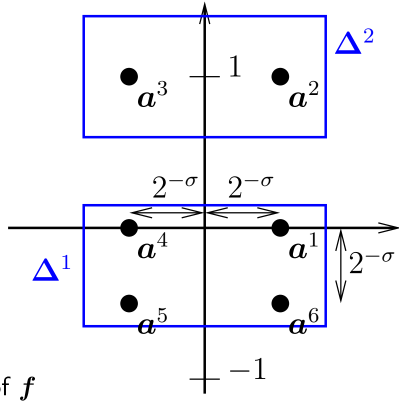
$\boldsymbol{a}^5 = (-2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^5, \boldsymbol{f}) = 1$

$\boldsymbol{a}^6 = (2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^6, \boldsymbol{f}) = 1$

Natural clusters:

$(\boldsymbol{\Delta}^1, 4)$

$(\boldsymbol{\Delta}^2, 2)$



Notations: $m(\boldsymbol{a}, \boldsymbol{f})$: multiplicity of $\boldsymbol{a}$ as a sol. of $\boldsymbol{f}$

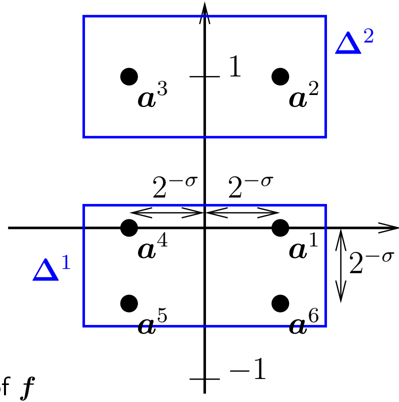
Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})^2 (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^\sigma z_1^2)^2 (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$\boldsymbol{a}^1 = (2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^1, \boldsymbol{f}) = 2$
$\boldsymbol{a}^2 = (2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^2, \boldsymbol{f}) = 2$
$\boldsymbol{a}^3 = (-2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^3, \boldsymbol{f}) = 1$
$\boldsymbol{a}^4 = (-2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^4, \boldsymbol{f}) = 1$
$\boldsymbol{a}^5 = (-2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^5, \boldsymbol{f}) = 2$
$\boldsymbol{a}^6 = (2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^6, \boldsymbol{f}) = 4$



Natural clusters:
$(\boldsymbol{\Delta}^1, 9)$
$(\boldsymbol{\Delta}^2, 3)$

Notations: $m(\boldsymbol{a}, \boldsymbol{f})$: multiplicity of $\boldsymbol{a}$ as a sol. of $\boldsymbol{f}$

Number of solutions in a polydisc

Let $\boldsymbol{\Delta} = (\Delta_1, \Delta_2)$ and $\boldsymbol{m} = (m_1, m_2)$.

Proposition 1: Suppose

(i) f_1 has m_1 roots in Δ_1 with multiplicity

(ii) $\forall \alpha_1 \in Z(\Delta_1, f_1)$, $f_2(\alpha_1)$ has m_2 roots in Δ_2 with multiplicity

Then $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Notation: $f_2(\alpha_1) \in \mathbb{C}[z_2]$: partial specialization of $f_2 \in \mathbb{Q}[z_1, z_2]$ in $\alpha_1 \in \mathbb{C}$

Number of solutions in a polydisc

Let $\mathbf{\Delta} = (\Delta_1, \Delta_2)$ and $\boldsymbol{m} = (m_1, m_2)$.

Proposition 1: Suppose

(i) f_1 has m_1 roots in Δ_1 with multiplicity

(ii) $\forall \alpha_1 \in Z(\Delta_1, f_1)$, $f_2(\alpha_1)$ has m_2 roots in Δ_2 with multiplicity

Then $\boldsymbol{f}(\boldsymbol{z}) = \mathbf{0}$ has $m_2 \times m_1$ solutions in $\mathbf{\Delta}$ with multiplicity.

Proof: direct consequence of

Theorem [ZFX11]: Let $\boldsymbol{\alpha} \in Z(\mathbb{C}^2, \boldsymbol{f})$, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$. Then

$$m(\boldsymbol{\alpha}, \boldsymbol{f}) = m(\alpha_2, f_2(\alpha_1)) \times m(\alpha_1, f_1)$$

[ZFX11] Zhihai Zhang, Tian Fang, and Bican Xia.
Real solution isolation with multiplicity of zero-dimensional triangular systems.
Science China Information Sciences, 54(1):60–69, 2011.

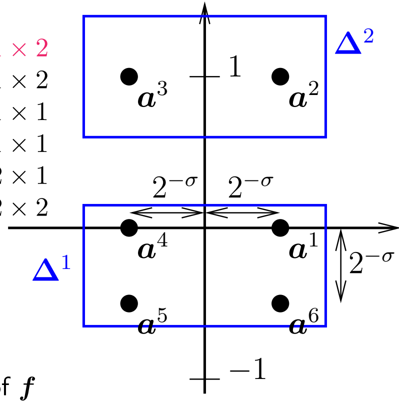
Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})^2 (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^{\sigma} z_1^2)^2 (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$\boldsymbol{a}^1 = (2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^1, \boldsymbol{f}) = 2 =: 1 \times 2$
$\boldsymbol{a}^2 = (2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^2, \boldsymbol{f}) = 2 =: 1 \times 2$
$\boldsymbol{a}^3 = (-2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^3, \boldsymbol{f}) = 1 =: 1 \times 1$
$\boldsymbol{a}^4 = (-2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^4, \boldsymbol{f}) = 1 =: 1 \times 1$
$\boldsymbol{a}^5 = (-2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^5, \boldsymbol{f}) = 2 =: 2 \times 1$
$\boldsymbol{a}^6 = (2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^6, \boldsymbol{f}) = 4 =: 2 \times 2$



Natural clusters:
$(\boldsymbol{\Delta}^1, 9)$
$(\boldsymbol{\Delta}^2, 3)$

Notations: $m(\boldsymbol{a}, \boldsymbol{f})$: multiplicity of $\boldsymbol{a}$ as a sol. of $\boldsymbol{f}$

Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})^2 (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^{\sigma} z_1^2)^2 (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$\boldsymbol{a}^1 = (2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^1, \boldsymbol{f}) = 2 = 1 \times 2$

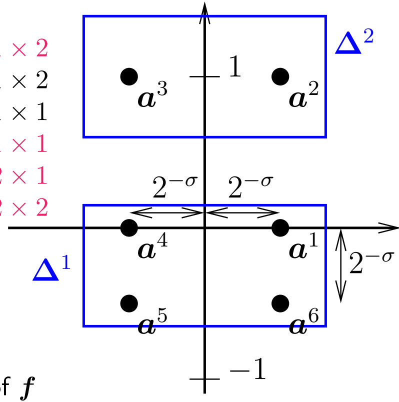
$\boldsymbol{a}^2 = (2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^2, \boldsymbol{f}) = 2 = 1 \times 2$

$\boldsymbol{a}^3 = (-2^{-\sigma}, 1) \leftarrow m(\boldsymbol{a}^3, \boldsymbol{f}) = 1 = 1 \times 1$

$\boldsymbol{a}^4 = (-2^{-\sigma}, 0) \leftarrow m(\boldsymbol{a}^4, \boldsymbol{f}) = 1 = 1 \times 1$

$\boldsymbol{a}^5 = (-2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^5, \boldsymbol{f}) = 2 = 2 \times 1$

$\boldsymbol{a}^6 = (2^{-\sigma}, -2^{-\sigma}) \leftarrow m(\boldsymbol{a}^6, \boldsymbol{f}) = 4 = 2 \times 2$



Natural clusters:

$(\boldsymbol{\Delta}^1, 9) \leftarrow 9 = 3 \times 3$

$(\boldsymbol{\Delta}^2, 3) \leftarrow 3 = 1 \times 3$

Notations: $m(\boldsymbol{a}, \boldsymbol{f})$: multiplicity of $\boldsymbol{a}$ as a sol. of $\boldsymbol{f}$

Number of solutions in a polydisc

Let $\boldsymbol{\Delta} = (\Delta_1, \Delta_2)$ and $\boldsymbol{m} = (m_1, m_2)$.

Proposition 1: Suppose

(i) f_1 has m_1 roots in Δ_1 with multiplicity

(ii) $\forall \alpha_1 \in Z(\Delta_1, f_1)$, $f_2(\alpha_1)$ has m_2 roots in Δ_2 with multiplicity

Then $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

Number of solutions in a polydisc

Let $\boldsymbol{\Delta} = (\Delta_1, \Delta_2)$ and $\boldsymbol{m} = (m_1, m_2)$.

Proposition 1: Suppose

(i) f_1 has m_1 roots in Δ_1 with multiplicity

(ii) $\forall \alpha_1 \in Z(\Delta_1, f_1)$, $f_2(\alpha_1)$ has m_2 roots in Δ_2 with multiplicity

Then $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

Corollary 2: If $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower,
$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Number of solutions in a polydisc

Let $\boldsymbol{\Delta} = (\Delta_1, \Delta_2)$ and $\boldsymbol{m} = (m_1, m_2)$.

Proposition 1: Suppose

(i) f_1 has m_1 roots in Δ_1 with multiplicity

(ii) $\forall \alpha_1 \in Z(\Delta_1, f_1)$, $f_2(\alpha_1)$ has m_2 roots in Δ_2 with multiplicity

Then $\boldsymbol{f}(\boldsymbol{z}) = \mathbf{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural ε-tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural ε-cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural ε-cluster relative to $f_2(\alpha_1)$

Corollary 2: If $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower,
$\boldsymbol{f}(\boldsymbol{z}) = \mathbf{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

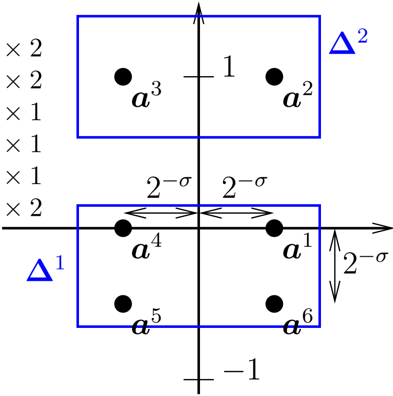
Example

System: Let $\sigma \geq 3$ and $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ be:

$$\begin{cases} (z_1 - 2^{-\sigma})^2 (z_1 + 2^{-\sigma}) & = & 0 \\ (z_2 + 2^{\sigma} z_1^2)^2 (z_2 - 1) z_2 & = & 0 \end{cases}$$

Solutions: $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has 6 solutions, all real:

$$\begin{array}{llll} \boldsymbol{a}^1 = (2^{-\sigma} & , & 0) & \leftarrow m(\boldsymbol{a}^1, \boldsymbol{f}) = 2 =: 1 \times 2 \\ \boldsymbol{a}^2 = (2^{-\sigma} & , & 1) & \leftarrow m(\boldsymbol{a}^2, \boldsymbol{f}) = 2 =: 1 \times 2 \\ \boldsymbol{a}^3 = (-2^{-\sigma} & , & 1) & \leftarrow m(\boldsymbol{a}^3, \boldsymbol{f}) = 1 =: 1 \times 1 \\ \boldsymbol{a}^4 = (-2^{-\sigma} & , & 0) & \leftarrow m(\boldsymbol{a}^4, \boldsymbol{f}) = 1 =: 1 \times 1 \\ \boldsymbol{a}^5 = (-2^{-\sigma} & , & -2^{-\sigma}) & \leftarrow m(\boldsymbol{a}^5, \boldsymbol{f}) = 2 =: 2 \times 1 \\ \boldsymbol{a}^6 = (2^{-\sigma} & , & -2^{-\sigma}) & \leftarrow m(\boldsymbol{a}^6, \boldsymbol{f}) = 4 =: 2 \times 2 \end{array}$$



Natural clusters:

$(\boldsymbol{\Delta}^1, 9) \leftarrow 9 = 3 \times 3$

$(\boldsymbol{\Delta}^2, 3) \leftarrow 3 = 1 \times 3$

Natural towers:

$(\boldsymbol{\Delta}^1, (3, 3))$

$(\boldsymbol{\Delta}^2, (1, 3))$

Pellet's test and natural towers

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Pellet's test and natural towers

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Proposition 3: Suppose

(i) SoftPelletTest(Δ_1, f_1) returns $m_1 \geq 1$

(ii) SoftPelletTest$(\Delta_2, f_2(\square\Delta_1))$ returns $m_2 \geq 1$

Then $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower relative to $\boldsymbol{f}$.

Notation: $f_2(\square\alpha_1) \in \square\mathbb{C}[z_2]$: partial specialization of $f_2 \in \mathbb{Q}[z_1, z_2]$ in $\square\alpha_1 \in \mathbb{C}$

Pellet's test and natural towers

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

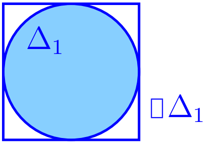
- (i) (Δ_1, m_1) is a natural cluster relative to f_1
- (ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Proposition 3: Suppose

- (i) SoftPelletTest(Δ_1, f_1) returns $m_1 \geq 1$
- (ii) SoftPelletTest$(\Delta_2, f_2(\Box\Delta_1))$ returns $m_2 \geq 1$

Then $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower relative to $\boldsymbol{f}$.



Notation: $f_2(\Box\alpha_1) \in \Box\mathbb{C}[z_2]$: partial specialization of $f_2 \in \mathbb{Q}[z_1, z_2]$ in $\Box\alpha_1 \in \mathbb{C}$

Pellet's test and natural towers

Definition: A pair $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower (relative to $\boldsymbol{f}$) if

(i) (Δ_1, m_1) is a natural cluster relative to f_1

(ii) $\forall \alpha_1 \in \Delta_1$, (Δ_2, m_2) is a natural cluster relative to $f_2(\alpha_1)$

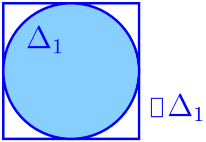
$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has $m_2 \times m_1$ solutions in $\boldsymbol{\Delta}$ with multiplicity.

Proposition 3: Suppose

(i) SoftPelletTest(Δ_1, f_1) returns $m_1 \geq 1$

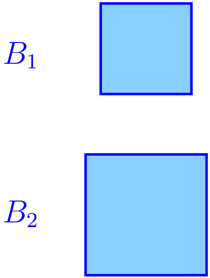
(ii) SoftPelletTest$(\Delta_2, f_2(\square\Delta_1))$ returns $m_2 \geq 1$

Then $(\boldsymbol{\Delta}, \boldsymbol{m})$ is a natural tower relative to $\boldsymbol{f}$.



Notation: $f_2(\square\alpha_1) \in \square\mathbb{C}[z_2]$: partial specialization of $f_2 \in \mathbb{Q}[z_1, z_2]$ in $\square\alpha_1 \in \mathbb{C}$

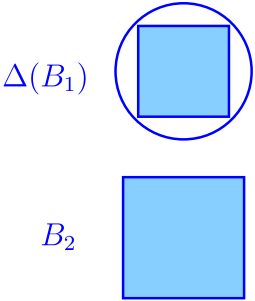
Main data structure



A **tower** is a triple $\mathcal{T} = \langle \ell, \boldsymbol{B}, \boldsymbol{L} \rangle$ where

- ℓ is an integer in $\{0, 1, 2\}$ called **level**
- $\boldsymbol{B} = (B_1, B_2)$ is a polybox called **domain**
- $\boldsymbol{L} = (L_1, L_2)$ is a vector in $(\mathbb{Z})^2$ called **precision**

Main data structure



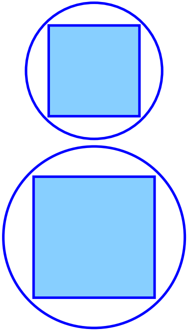
A **tower** is a triple $\mathcal{T} = \langle \ell, \boldsymbol{B}, \boldsymbol{L} \rangle$ where

- ℓ is an integer in $\{0, 1, 2\}$ called **level**
- $\boldsymbol{B} = (B_1, B_2)$ is a polybox called **domain**
- $\boldsymbol{L} = (L_1, L_2)$ is a vector in $(\mathbb{Z})^2$ called **precision**

We will garantee that if $\ell = 1$, $\exists m_1$ so that:

(i) $(\Delta(B_1), m_1)$ is a natural 2^{-L_1} cluster

Main data structure



A **tower** is a triple $\mathcal{T} = \langle \ell, \boldsymbol{B}, \boldsymbol{L} \rangle$ where

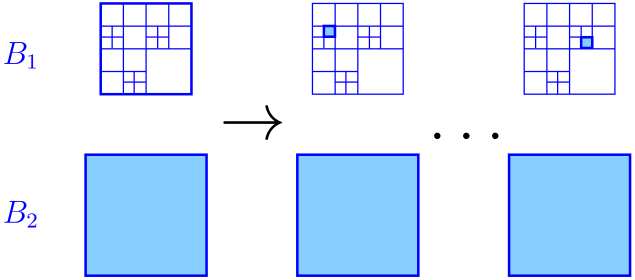
- ℓ is an integer in $\{0, 1, 2\}$ called **level**
- $\boldsymbol{B} = (B_1, B_2)$ is a polybox called **domain**
- $\boldsymbol{L} = (L_1, L_2)$ is a vector in $(\mathbb{Z})^2$ called **precision**

We will garantee that if $\ell = 2$, $\exists \boldsymbol{m}$ so that:

(i) $(\Delta(B_1), m_1)$ is a natural 2^{-L_1} cluster

(ii) $(\Delta(\boldsymbol{B}), \boldsymbol{m})$ is a natural 2^{-L_2} tower (relative to $\boldsymbol{f}$)

Lift of a tower from level 0 to level 1

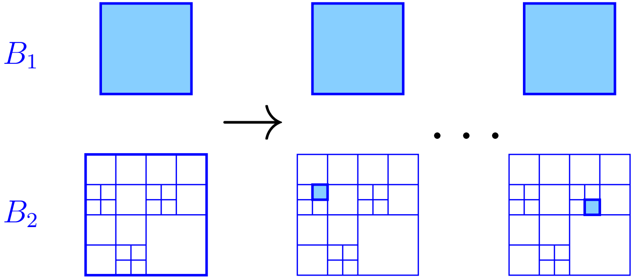


Cluster1($\boldsymbol{f}, \mathcal{T}$)

Input: $\boldsymbol{f} = (f_1, f_2)$, $\mathcal{T} = \langle \ell, \boldsymbol{B}, \boldsymbol{L} \rangle$ a tower
Output: a list of towers at level 1

1. call ClusterOracle for f_1, B_1, 2^{-L_1}

Lift of a tower from level 1 to level 2



Cluster2($\boldsymbol{f}, \mathcal{T}$)

Input: $\boldsymbol{f} = (f_1, f_2)$, $\mathcal{T} = \langle \ell, \boldsymbol{B}, \boldsymbol{L} \rangle$ a tower at level 1
Output: a flag in {**success**, **fail**} and a list of towers at level 2

1. call ClusterInterval for $f_2(\square\Delta(B_1))$, B_2, 2^{-L_2}
fail if SoftPelletTest returns -2 (*i.e.* not enough prec. on $\square\Delta(B_1)$)

Main algorithm

ClusterTri($\boldsymbol{f}, \boldsymbol{B}, L$)

Input: a triangular system $\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$, a polybox $\boldsymbol{B}$, $L > 0$
Output: a set of natural 2^{-L}-towers solving the LCP

1. . . . //*interleave Cluster1 and Cluster2*

Our implementation

`Ccluster`: library in `C` based on

- FLINT[1]: arithmetic for the geometric algorithm
- Arb[2]: arbitrary precision floating arithmetic with error bounds

Available at `https://github.com/rimbach/Ccluster`

`Ccluster.jl`: package for julia[3] based on $\mathbb{N}e^{m}\mathcal{O}$[4]

- interface for `Ccluster`
- `Tcluster`: implementation of ClusterTri

Available at `https://github.com/rimbach/Ccluster.jl`

[1] https://github.com/wbhart/flint2
[2] http://arblib.org/
[3] https://julialang.org/
[4] http://nemocas.org/

Benchmark: systems

Type of a triangular system:

$\boldsymbol{f}(\boldsymbol{z}) = \mathbf{0}$ has type $(d_1, \ldots, d_n)$ if f_i has degree d_i in z_i, $\forall 1 \leq i \leq n$

Table: for each type, average on 5 random dense systems

seq. times on a `Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz`

type								
Systems with only simple solutions								
(9,9,9)								
(6,6,6,6)								
(9,9,9,9)								
(6,6,6,6,6)								
(9,9,9,9,9)								
(2,2,2,2,2,2,2,2,2,2)								
Systems with multiple solutions								
(9,9)								
(6,6,6)								
(9,9,9)								
(6,6,6,6)								

Benchmark: local vs global comparison

Type of a triangular system:
$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has type $(d_1, \ldots, d_n)$ if f_i has degree d_i in z_i, $\forall 1 \leq i \leq n$

Table: for each type, average on 5 random dense systems
seq. times on a `Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz`

	`Tcluster` **local**		`Tcluster` **global**					
type	(`#Clus`, `#Sols`)	t (s)	(`#Clus`, `#Sols`)	t (s)				
Systems with only simple solutions								
(9,9,9)	(149 : 149)	0.24	(729 : 729)	1.21				
(6,6,6,6)	(63.4 : 63.4)	0.10	(1296 : 1296)	1.73				
(9,9,9,9)	(559 : 559)	1.06	(6561 : 6561)	12.9				
(6,6,6,6,6)	(155 : 155)	0.37	(7776 : 7776)	11.1				
(9,9,9,9,9)	(1739 : 1739)	4.83	(59049 : 59049)	113				
(2,2,2,2,2,2,2,2,2,2)	(0 : 0)	0.13	(1024 : 1024)	2.42				
Systems with multiple solutions								
(9,9)	(23.8: 13.6)	0.03	(81 : 45)	0.15				
(6,6,6)	(35.2: 8.80)	0.05	(216 : 54)	0.24				
(9,9,9)	(113 : 37.6)	0.22	(729 : 225)	1.06				
(6,6,6,6)	(81.6: 10.2)	0.21	(1296: 162)	1.28				

`Tcluster` **local** : $\boldsymbol{B} = ([-1,1] + \imath[-1,1])^n$, $\varepsilon = 2^{-53}$
`Tcluster` **global**: $\boldsymbol{B}$ chosen with upper bound for roots

Benchmark: extern comparison

Type of a triangular system:
$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has type $(d_1, \ldots, d_n)$ if f_i has degree d_i in z_i, $\forall 1 \leq i \leq n$

Table: for each type, average on 5 random dense systems
seq. times on a `Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz`

	`Tcluster` **local**		`Tcluster` **global**		`HomCont.jl`			
type	(`#Clus`, `#Sols`)	t (s)	(`#Clus`, `#Sols`)	t (s)	`#Sols`	t (s)		
Systems with only simple solutions								
(9,9,9)	(149 : 149)	0.24	(729 : 729)	1.21	729	4.21		
(6,6,6,6)	(63.4 : 63.4)	0.10	(1296 : 1296)	1.73	1296	4.70		
(9,9,9,9)	(559 : 559)	1.06	(6561 : 6561)	12.9	6561	14.0		
(6,6,6,6,6)	(155 : 155)	0.37	(7776 : 7776)	11.1	7776	11.5		
(9,9,9,9,9)	(1739 : 1739)	4.83	(59049 : 59049)	113	59049	116		
(2,2,2,2,2,2,2,2,2,2)	(0 : 0)	0.13	(1024 : 1024)	2.42	1024	4.84		
Systems with multiple solutions								
(9,9)	(23.8: 13.6)	0.03	(81 : 45)	0.15	33.6	3.27		
(6,6,6)	(35.2: 8.80)	0.05	(216 : 54)	0.24	53.2	2.75		
(9,9,9)	(113 : 37.6)	0.22	(729 : 225)	1.06	159	28.4		
(6,6,6,6)	(81.6: 10.2)	0.21	(1296: 162)	1.28	134	8.06		

`Tcluster` **local** : $\boldsymbol{B} = ([-1, 1] + \imath[-1, 1])^n$, $\varepsilon = 2^{-53}$
`Tcluster` **global**: $\boldsymbol{B}$ chosen with upper bound for roots
`HomCont.jl`: `HomotopyContinuation.jl`

Benchmark:

Type of a triangular system:
$\boldsymbol{f}(\boldsymbol{z}) = \mathbf{0}$ has type $(d_1, \ldots, d_n)$ if f_i has degree d_i in z_i, $\forall 1 \leq i \leq n$

Table: for each type, average on 5 random dense systems
seq. times on a `Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz`

	`Tcluster` **local**		`Tcluster` **global**		`HomCont.jl`			
type	(`#Clus`, `#Sols`)	t (s)	(`#Clus`, `#Sols`)	t (s)	`#Sols`	t (s)		
Systems with only simple solutions								
(9,9,9)	(149 : 149)	0.24	(729 : 729)	1.21	729	4.21		
(6,6,6,6)	(63.4 : 63.4)	0.10	(1296 : 1296)	1.73	1296	4.70		
(9,9,9,9)	(559 : 559)	1.06	(6561 : 6561)	12.9	6561	14.0		
(6,6,6,6,6)	(155 : 155)	0.37	(7776 : 7776)	11.1	7776	11.5		
(9,9,9,9,9)	(1739 : 1739)	4.83	(59049 : 59049)	113	59049	116		
(2,2,2,2,2,2,2,2,2,2)	(0 : 0)	0.13	(1024 : 1024)	2.42	1024	4.84		
Systems with multiple solutions								
(9,9)	(23.8: 13.6)	0.03	(81 : 45)	0.15	33.6	3.27		
(6,6,6)	(35.2: 8.80)	0.05	(216 : 54)	0.24	53.2	2.75		
(9,9,9)	(113 : 37.6)	0.22	(729 : 225)	1.06	159	28.4		
(6,6,6,6)	(81.6: 10.2)	0.21	(1296: 162)	1.28	134	8.06		

`Tcluster` **local** : $\boldsymbol{B} = ([-1,1] + \imath[-1,1])^n$, $\varepsilon = 2^{-53}$
`Tcluster` **global**: $\boldsymbol{B}$ chosen with upper bound for roots
`HomCont.jl`: `HomotopyContinuation.jl`

Benchmark:

Type of a triangular system:
$\boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{0}$ has type $(d_1, \ldots, d_n)$ if f_i has degree d_i in z_i, $\forall 1 \leq i \leq n$

Table: for each type, average on 5 random dense systems
seq. times on a `Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz`

	Tcluster **local**		Tcluster **global**		HomCont.jl		triang_solve	
type	(#Clus, #Sols)	t (s)	(#Clus, #Sols)	t (s)	#Sols	t (s)	#Sols	t (s)
Systems with only simple solutions								
(9,9,9)	(149 : 149)	0.24	(729 : 729)	1.21	729	4.21	729	0.37
(6,6,6,6)	(63.4 : 63.4)	0.10	(1296 : 1296)	1.73	1296	4.70	1296	0.93
(9,9,9,9)	(559 : 559)	1.06	(6561 : 6561)	12.9	6561	14.0	6561	8.57
(6,6,6,6,6)	(155 : 155)	0.37	(7776 : 7776)	11.1	7776	11.5	7776	19.1
(9,9,9,9,9)	(1739 : 1739)	4.83	(59049 : 59049)	113	59049	116	59049	702
(2,2,2,2,2,2,2,2,2,2)	(0 : 0)	0.13	(1024 : 1024)	2.42	1024	4.84	1024	3.9
Systems with multiple solutions								
(9,9)	(23.8: 13.6)	0.03	(81 : 45)	0.15	33.6	3.27	45	0.03
(6,6,6)	(35.2: 8.80)	0.05	(216 : 54)	0.24	53.2	2.75	54	0.05
(9,9,9)	(113 : 37.6)	0.22	(729 : 225)	1.06	159	28.4	225	0.23
(6,6,6,6)	(81.6: 10.2)	0.21	(1296: 162)	1.28	134	8.06	162	0.15

`Tcluster` **local** : $\boldsymbol{B} = ([-1,1] + \imath[-1,1])^n$, $\varepsilon = 2^{-53}$
`Tcluster` **global**: $\boldsymbol{B}$ chosen with upper bound for roots
`HomCont.jl`: `HomotopyContinuation.jl`
`triang_solve`: Singular solver for triangular systems

Ongoing and future works

Bit complexity of ClusterTri($\boldsymbol{f}, \boldsymbol{B}, L$):

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

i.e. any root α_1 of f_1 can be extended to $\deg_{z_2}(f_2)$ solutions of $\boldsymbol{f}$

Ongoing and future works

Bit complexity of ClusterTri($\boldsymbol{f}, \boldsymbol{B}, L$):

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

i.e. any root α_1 of f_1 can be extended to $\deg_{z_2}(f_2)$ solutions of $\boldsymbol{f}$

Weaker assumption: $f_1 = f_2 = 0$ is weakly regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

i.e. any root α_1 of f_1 can be extended to at least 1 solution of $\boldsymbol{f}$

Ongoing and future works

Bit complexity of ClusterTri($\boldsymbol{f}, \boldsymbol{B}, L$):

Usual assumption: $f_1 = f_2 = 0$ is regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2) = \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

i.e. any root α_1 of f_1 can be extended to $\deg_{z_2}(f_2)$ solutions of $\boldsymbol{f}$

Weaker assumption: $f_1 = f_2 = 0$ is weakly regular

$$\forall \alpha \in Z(\mathbb{C}, f_1), \quad \deg_{z_2}(f_2(\alpha, z_2)) \geq 1$$

i.e. any root α_1 of f_1 can be extended to at least 1 solution of $\boldsymbol{f}$

$\rightarrow$ compute clusters of roots of $f \in \mathbb{C}[z_2]$
with nominal degree $d = \deg_{z_2}(f_2)$
with true degree $d' \leq d$ (unknown)

$\rightarrow$ but the bit complexity of [BSS+16] is stated for $1/2 \leq \mathrm{lcf}(f) < 1$

Ongoing and future works

Bit complexity of ClusterTri$(\boldsymbol{f}, \boldsymbol{B}, L)$:

Application to planar curve topology via CAD:

$$\left\{ \begin{array}{rcl} p(z_1, z_2) & = & 0 \end{array} \right.$$

requires to solve

$$f_1(z_1) = p(z_1, z_2) = 0 \text{ which is weakly regular}$$

where $f_1 = \mathrm{Res}_{z_2}(p, \frac{\partial p}{\partial z_2})$

ToC

0 - Univariate case:

1 - Multivariate triangular case

2 - Back to univariate case

Cauchy's theorem: if no root of f on $\partial\Delta$,

$$\#(\Delta, f) = \frac{1}{2\pi\imath} \int_{\partial\Delta} \frac{f'(z)}{f(z)} dz$$

[IP20] Rémi Imbach and Victor Y. Pan.
New progress in univariate polynomial root finding.
ISSAC '20, page 249–256, New York, NY, USA, 2020. ACM.

ToC

0 - Univariate case:

1 - Multivariate triangular case

2 - Back to univariate case

Cauchy's theorem: if no root of f on $\partial\Delta$,

$$\#(\Delta, f) = \frac{1}{2\pi\imath}\int_{\partial\Delta}\frac{f'(z)}{f(z)}dz$$

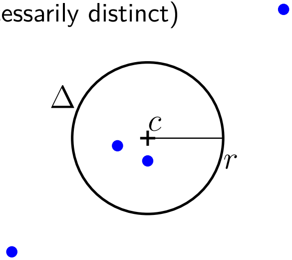
$\rightarrow$ close approximation by sampling f'/f on $O(\log d)$ points on $\partial\Delta$

$\rightarrow$ Cauchy root counter and exclusion test

[IP20] Rémi Imbach and Victor Y. Pan.
New progress in univariate polynomial root finding.
ISSAC '20, page 249–256, New York, NY, USA, 2020. ACM.

Power Sums

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



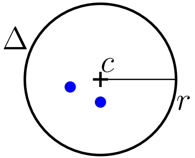
h-th power sum of f in Δ:

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

Notation: $D(c,r)$: disk centered in c with radius r

Power Sums

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



h-th power sum of f in Δ:

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

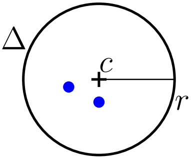
Remarks:

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

Notation: $D(c, r)$: disk centered in c with radius r

Power Sums

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



h-th power sum of f in Δ:

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

Remarks:

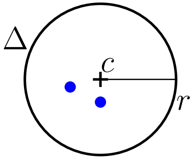
(i) $\#(\Delta, f) = s_0(\Delta, f)$

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

Notation: $D(c, r)$: disk centered in c with radius r

Power Sums

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



h-th power sum of f in Δ:

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

Remarks:

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

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

Let $f_\Delta = f(c + rz)$:

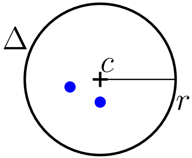
(iii) $\#(\Delta, f) = s_0(D(0,1), f_\Delta)$

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

Notation: $D(c, r)$: disk centered in c with radius r ; $D(0,1)$: unit disk

Power Sums

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



h-th power sum of f in Δ:

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

Remarks:

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

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

Let $f_\Delta = f(c + rz)$:

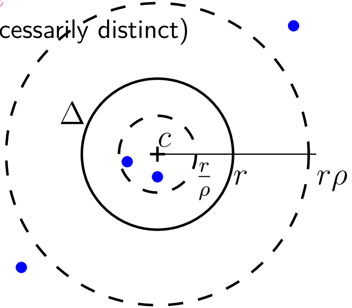
(iii) $\#(\Delta, f) = s_0(D(0,1), f_\Delta)$

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

Notation: $D(c, r)$: disk centered in c with radius r ; $D(0, 1)$: unit disk

Power Sums

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



h-th power sum of f in Δ:

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

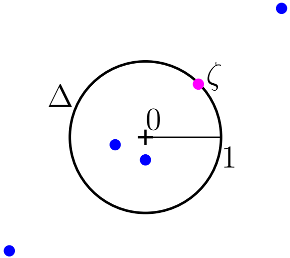
Definition: $\Delta = D(c, r)$ is ρ isolated, for $\rho > 1$, if
$D(c, r\rho) \setminus D(c, \frac{r}{\rho})$
ρ: isolation ratio of Δ contains no root

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

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

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

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

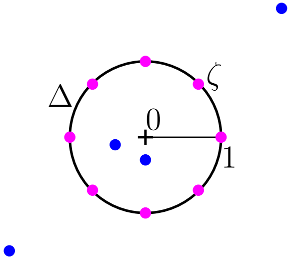


Approximation of the Power Sums of f in $D(0,1)$

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

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

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

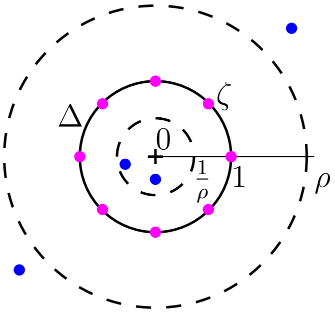


Approximation of the Power Sums of f in $D(0,1)$

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

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

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



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

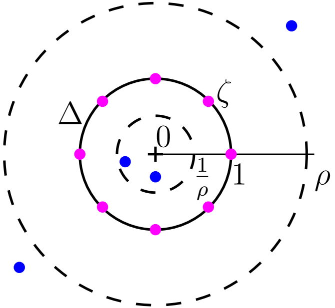
(i) $|s_h^* - s_h(D(0,1), f)| \leq \frac{d\theta^{q-h}}{1-\theta^q}$ where $\theta = \frac{1}{\rho}$

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

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

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

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



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

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

(ii) If $q = \lceil \log_\rho(4d+1) \rceil + h$ then $|s_h^* - s_h(D(0,1), f)| \leq 1/4$.

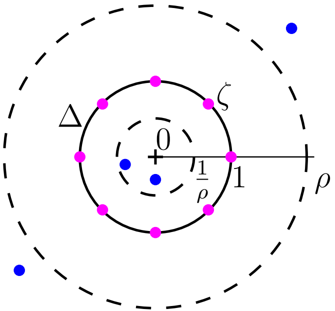
[Sch82] Arnold Schönhage.
The fundamental theorem of algebra in terms of computational complexity.
Manuscript. Univ. of Tübingen, Germany, 1982.

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

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

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

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



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

(i) $|s_h^* - s_h(D(0,1),f)| \leq \frac{d\theta^{q-h}}{1-\theta^q}$ where $\theta = \frac{1}{\rho}$

(ii) If $q = \lceil \log_\rho(4d+1) \rceil + h$ then $|s_h^* - s_h(D(0,1),f)| \leq 1/4$.

Remark: $s_0(D(0,1),f)$ is an integer, thus error $1/4$ is enough

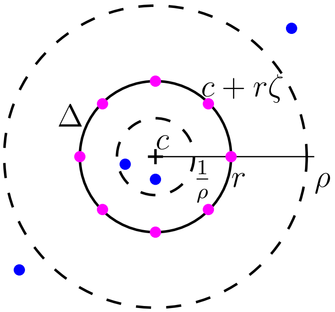
Example: when $\rho = 2$ and $d = 500$, $q = 11$ allows to recover $s_0(D(0,1),f)$

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

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

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

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

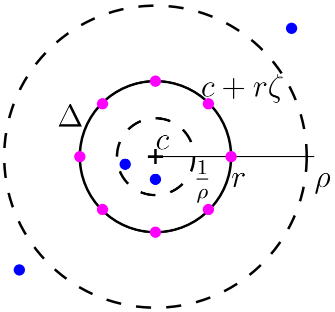


Approximation of the Power Sums of f_Δ in $D(0,1)$

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

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

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



Theorem [Sch82]: Let $\rho > 1$;
suppose $\Delta = D(c, r)$ is ρ-isolated. Then

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

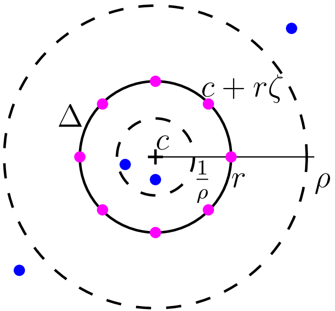
(ii) If $q = \lceil \log_\rho(4d+1) \rceil + h$ then $|s_h^* - s_h(D(0,1), f_\Delta)| \leq 1/4$.

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

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

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

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



Theorem [Sch82]: Let $\rho > 1$;
suppose $\Delta = D(c,r)$ is ρ-isolated. Then

(i) $|s_h^* - s_h(D(0,1), f_\Delta)| \leq \dfrac{d\theta^{q-h}}{1-\theta^q}$ where $\theta = \dfrac{1}{\rho}$

(ii) If $q = \lceil \log_\rho(4d+1) \rceil + h$ then $|s_h^* - s_h(D(0,1), f_\Delta)| \leq 1/4$.

Remark: Approximating $s_h(D(0,1), f_\Delta)$ does **not** require to compute the coefficients of $f_\Delta = f(c + rz)$

Root counter and exclusion test

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

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

1. compute s_0^* s.t. $|s_0^* - s_0(D(0,1), f_\Delta)| \leq \frac{1}{4}$
2. **return** the unique integer in $D(s_0^*, \frac{1}{4})$

Root counter and exclusion test

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

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

1. compute s_0^* s.t. $|s_0^* - s_0(D(0,1), f_\Delta)| \leq \frac{1}{4}$
2. **return** the unique integer in $D(s_0^*, \frac{1}{4})$

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

Input: ...
Output: **true** iff f has no root in Δ

1. **return** $C^*(f, \Delta, \rho) == 0$

Root counter and exclusion test

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

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

1. compute s_0^* s.t. $|s_0^* - s_0(D(0,1), f_\Delta)| \leq \frac{1}{4}$
2. **return** the unique integer in $D(s_0^*, \frac{1}{4})$

Remarks:

1. if $\rho \simeq 2$, costs an $O(\log d)$ evaluations of f and f'
   $\rightarrow$ can be very fast when f can be evaluated fast

Root counter and exclusion test

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

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

1. compute s_0^* s.t. $|s_0^* - s_0(D(0,1), f_\Delta)| \leq \frac{1}{4}$
2. **return** the unique integer in $D(s_0^*, \frac{1}{4})$

Remarks:

1. if $\rho \simeq 2$, costs an $O(\log d)$ evaluations of f and f'
   $\rightarrow$ can be very fast when f can be evaluated fast
2. do not require to know the coeffs of f
   $\rightarrow$ a (fast) procedure for evaluation of f is sufficient

Root counter and exclusion test

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

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

1. compute s_0^* s.t. $|s_0^* - s_0(D(0,1), f_\Delta)| \leq \frac{1}{4}$
2. **return** the unique integer in $D(s_0^*, \frac{1}{4})$

Remarks:

1. if $\rho \simeq 2$, costs an $O(\log d)$ evaluations of f and f'
   $\rightarrow$ can be very fast when f can be evaluated fast
2. do not require to know the coeffs of f
   $\rightarrow$ a (fast) procedure for evaluation of f is sufficient
3. when exclusion test is applied to a box of a subdivision tree, the isolation ratio is unknown

Question: What if ρ is unknown?

Unsure Exclusion Test

$\widetilde{C}^0(f, \Delta, k)$

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

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
2. **for** $h = 0, \dots, k$ **do**
3. compute s_h^* s.t. $|s_h^* - s_h(D(0,1), f_\Delta)| \leq \frac{1}{4}$
4. **if** $D(s_h^*, \frac{1}{4})$ does not contain zero
5. **return can not decide**
6. **return true**

Unsure Exclusion Test

$\widetilde{C}^0(f, \Delta, k)$

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

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
2. **for** $h = 0, \ldots, k$ **do**
3. compute s_h^* s.t. $|s_h^* - s_h(D(0,1), f_\Delta)| \leq \frac{1}{4}$
4. **if** $D(s_h^*, \frac{1}{4})$ does not contain zero
5. **return can not decide**
6. **return true**

Why $\rho = 4/3$? to have:
If $2B$ contains no root, then $\widetilde{C}^0(f, \Delta(B), k)$ returns **true**

Unsure Exclusion Test

$\widetilde{C}^0(f, \Delta, k)$

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

0. Let $\rho = \frac{4}{3}$, and assume Δ is ρ-isolated
2. **for** $h = 0, \dots, k$ **do**
3. compute s_h^* s.t. $|s_h^* - s_h(D(0,1), f_\Delta)| \leq \frac{1}{4}$
4. **if** $D(s_h^*, \frac{1}{4})$ does not contain zero
5. **return can not decide**
6. **return true**

Why $\rho = 4/3$? to have:
If $2B$ contains no root, then $\widetilde{C}^0(f, \Delta(B), k)$ returns **true**

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

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$			
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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$			
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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$			
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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k = 0$			
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: number of exclusion tests in [BSS$^+$16]
t: sequential time of [BSS$^+$16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$			
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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$		$\widetilde{C^0}$-tests, $k=1$		$\widetilde{C^0}$-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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests with $k=0$
t_1': time spent in $\widetilde{C^0}$-tests with $k=1$
#F': nb of wrong res. in $\widetilde{C^0}$-tests with $k=1$
t_1'': time spent in $\widetilde{C^0}$-tests with $k=2$
#F": nb of wrong res. in $\widetilde{C^0}$-tests with $k=2$

Unsure Exclusion Test: Experiments

		P^0-tests	$\widetilde{C^0}$-tests, $k=0$		$\widetilde{C^0}$-tests, $k=1$		$\widetilde{C^0}$-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: number of exclusion tests in [BSS+16]
t: sequential time of [BSS+16]
t_0: time spent in P^0-tests
t_1: time spent in $\widetilde{C^0}$-tests
#F: nb of wrong res. in $\widetilde{C^0}$-tests with $k=0$
t'_1: time spent in $\widetilde{C^0}$-tests with $k=1$
#F': nb of wrong res. in $\widetilde{C^0}$-tests with $k=1$
t''_1: time spent in $\widetilde{C^0}$-tests with $k=2$
#F": nb of wrong res. in $\widetilde{C^0}$-tests with $k=2$

Subdivision Algorithm with Unsure Exclusion Test

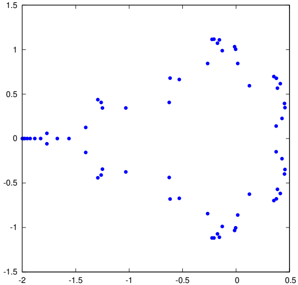
- for the (global) Root Clustering Problem
- uses $\widetilde{C^0}$-test with $k = 2$
- always terminates, but may fail: in this case, reports failure
- implemented in `C` within `Ccluster`: `CclusterF`

Subdivision Algorithm with Unsure Exclusion Test

- for the (global) Root Clustering Problem
- uses $\widetilde{C^0}$-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: $\texttt{Mandelbrot}_\ell(z)$
Input: $\ell \in \mathbb{N}^*, z \in \mathbb{C}$
Output: $\alpha \in \mathbb{C}$

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



$\ell = 6 \ (d = 63)$

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

Ongoing and future works

Deterministic support for the Cauchy root counter with unknown ρ
$\rightarrow$ now requires $O(d)$ evaluations ... can we afford this in practice?

Cauchy subdivision root finder
$\rightarrow$ without coefficients of f
$\rightarrow$ only black box for evaluation of f and f'

Soft Exact computation

Numerical methods are fast

but not robust to Zero problems

Zero problems can be tackled with exact computation

in only a few cases

Soft exact computation (for instance local root clustering)

is a middle path?

avoids Zero problem