On OM Algorithm and Cluster Pictures II:
handling low precision

Adrien Poteaux[1] Tristan Vaccon[2] Martin Weimann [3]

1. Université Lille, Cristal, Lille, France
2. Université de Limoges, CNRS, XLIM, Limoges, France
3. Université de Caen-Normandie, LMNO, Caen, France

ISSAC 2026, July 17th

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:     A = Qp(5)
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Case of

$$f = X^2 + O(p^{20}),$$

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
 In:   A = Qp(5)
 In:   B = PolynomialRing(A,'X')
 In:   X = B.gen()
 In:   f = X^2+X+1
 In:   f
Out:   (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Case of

$$f = X^2 + O(p^{20}),$$

X^2: 0 is a double root.

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Case of

$$f = X^2 + O(p^{20}),$$

X^2: 0 is a double root.

$X^2 - p^{20}$: two simple roots, $X^2 - p^{20} = (X - p^{10})(X + p^{10})$.

Where are the roots of approximate polynomials?

Assume one would like to study the 5-adic properties of $f = X^2 + X + 1$.

```
In:   A = Qp(5)
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2+X+1
In:   f
Out:  (1 + O(5^20))*X^2 + (1 + O(5^20))*X + 1 + O(5^20)
```

One here will naturally works with the **approximate** polynomial $f + O(5^{20})$.

Where are the roots of approximate polynomials?

Case of

$$f = X^2 + O(p^{20}),$$

X^2: 0 is a double root.

$X^2 - p^{20}$: two simple roots, $X^2 - p^{20} = (X - p^{10})(X + p^{10})$.

Is there a good answer to **handle and clusterize all the roots** of the polynomials of the form $f + O(p^N)$?

Table of contents

Table of contents

Table of contents

Setting

Discrete-valuation fields

Let K be a field with:

- discrete valuation $v_K : K \to \mathbb{R} \cup \{+\infty\}$ making it complete,
- integer ring O_K,
- uniformizer π
- residue field κ.

Setting

Discrete-valuation fields

Let K be a field with:

- discrete valuation $v_K : K \to \mathbb{R} \cup \{+\infty\}$ making it complete,
- integer ring O_K,
- uniformizer π
- residue field κ.

Example

K	valuation	R	π	κ
$\mathbb{Q}_p$	p-adic	$\mathbb{Z}_p$	p	$\mathbb{F}_p$
$\mathbb{F}_q((t))$	t-adic	$\mathbb{F}_q[[t]]$	t	$\mathbb{F}_q$
$\mathbb{Q}((t))$	t-adic	$\mathbb{Q}[[t]]$	t	$\mathbb{Q}$

and their finite extensions.

Setting

Discrete-valuation fields

Let K be a field with:

- discrete valuation $v_K : K \to \mathbb{R} \cup \{+\infty\}$ making it complete,
- integer ring O_K,
- uniformizer π
- residue field κ.

Example

K	valuation	R	π	κ
$\mathbb{Q}_p$	p-adic	$\mathbb{Z}_p$	p	$\mathbb{F}_p$
$\mathbb{F}_q((t))$	t-adic	$\mathbb{F}_q[[t]]$	t	$\mathbb{F}_q$
$\mathbb{Q}((t))$	t-adic	$\mathbb{Q}[[t]]$	t	$\mathbb{Q}$

and their finite extensions.

Topology somehow in-between discrete and continuous mathematics.

Non-archimedean balls

Definition (Ball/disks)

$$B_K(\alpha;\lambda) := \{x \in K : \ v_K(x-\alpha) \geq \lambda\}.$$

Non-archimedean balls

Definition (Ball/disks)

$$B_K(\alpha; \lambda) := \{x \in K : \; v_K(x - \alpha) \geq \lambda\}.$$

Definition (Discoid)

Let $\varphi \in K[x]$ be a monic polynomial and let $\lambda \in \mathbb{Q}$. Then

$$D_K(\varphi; \lambda) := \{x \in K : \; v_K(\varphi(x)) \geq \lambda\}$$

is the **discoid** with center φ and radius λ over K.

Non-archimedean balls

Definition (Ball/disks)

$$B_K(\alpha;\lambda) := \{x \in K :\ v_K(x-\alpha) \geq \lambda\}\,.$$

Definition (Discoid)

Let $\varphi \in K[x]$ be a monic polynomial and let $\lambda \in \mathbb{Q}$. Then

$$D_K(\varphi;\lambda) := \{x \in K :\ v_K(\varphi(x)) \geq \lambda\}$$

is the **discoid** with center φ and radius λ over K.

Definition (Gauss valuation)

Let $f = \sum_{i=0}^{d} a_i X^i \in K[X]$, we define the Gauss valuation of f as:

$$\mu_0(f) := \min_{0\leq i\leq d} v_K(a_i).$$

Remark

If $g = f + O(p^N)$ then

$$\mu_0(f-g) \geq N.$$

Table of contents

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\right). \end{array}$$

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d} (x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d} (x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\Big(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\Big). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d} (x - \alpha_k)$ and $f^* = \prod_{k=1}^{d} (x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x - \alpha_k)$ and $f^* = \prod_{k=1}^{d}(x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Our strategy is then the following:

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x - \alpha_k)$ and $f^* = \prod_{k=1}^{d}(x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Our strategy is then the following:

1. `Relative configuration of the roots` of f:

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x - \alpha_k)$ and $f^* = \prod_{k=1}^{d}(x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Our strategy is then the following:

1. `Relative configuration of the roots` of f: OM algorithm.

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x-\alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R}\cup\{\infty\} & \longrightarrow & \mathbb{R}\cup\{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha-\alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x-\alpha_k)$ and $f^* = \prod_{k=1}^{d}(x-\alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f-f^*))).$$

Our strategy is then the following:

1. `Relative configuration of the roots` of f: OM algorithm.
2. Compute the error functions at the roots and their inverses.

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x - \alpha_k)$ *and* $f^* = \prod_{k=1}^{d}(x - \alpha_k^*)$. *Their roots (in* $\overline{K}$*) can be ordered such that for all* k*,*

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Our strategy is then the following:

1. `Relative configuration of the roots` of f: OM algorithm.
2. Compute the error functions at the roots and their inverses.
3. Compute the relative configuration of Brink's balls:

David Brink: New light on Hensel's lemma, 2006

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x-\alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R}\cup\{\infty\} & \longrightarrow & \mathbb{R}\cup\{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha-\alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d}(x-\alpha_k)$ *and* $f^* = \prod_{k=1}^{d}(x-\alpha_k^*)$. *Their roots (in* $\overline{K}$*) can be ordered such that for all* k*,*

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f-f^*))).$$

Our strategy is then the following:

1. `Relative configuration of the roots` of f: OM algorithm.
2. Compute the error functions at the roots and their inverses.
3. Compute the relative configuration of Brink's balls: `app. Berkovich skeleton` .

Berkovich skeleton of the roots

Proposition

1. *If two disks in K have a non-trivial intersection, then one* ***contains*** *the other.*

Berkovich skeleton of the roots

Proposition

1. *If two disks in K have a non-trivial intersection, then one **contains** the other.*
2. *Any set of disks has a forest / tree-like structure for inclusion.*

Berkovich skeleton of the roots

Proposition

1. *If two disks in K have a non-trivial intersection, then one* **contains** *the other.*
2. *Any set of disks has a forest / tree-like structure for inclusion.*

Definition

The **Berkovich skeleton of the roots** of $f = \prod_{i=1}^{d}(X - \alpha_i)$ is the tree (for the inclusion relation) of the disks

$$B_{\overline{K}}(\alpha_i, v_{\overline{K}}(\alpha_i - \alpha_j))\,.$$

Berkovich skeleton of the roots

Proposition

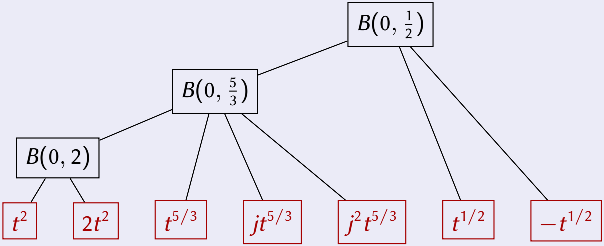
1. *If two disks in K have a non-trivial intersection, then one* **contains** *the other.*
2. *Any set of disks has a forest / tree-like structure for inclusion.*

Definition

The **Berkovich skeleton of the roots** of $f = \prod_{i=1}^{d}(X - \alpha_i)$ is the tree (for the inclusion relation) of the disks

$$B_{\overline{K}}(\alpha_i, v_{\overline{K}}(\alpha_i - \alpha_j)) .$$

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$



Two words on cluster pictures

What is a cluster picture?

Can be understood as a top-down representation of the Berkovich skeleton.

Two words on cluster pictures

What is a cluster picture?

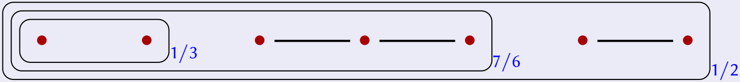
Can be understood as a top-down representation of the Berkovich skeleton. Galois action on the balls (or discoids) is also represented.

Two words on cluster pictures

What is a cluster picture?

Can be understood as a top-down representation of the Berkovich skeleton. Galois action on the balls (or discoids) is also represented.

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$

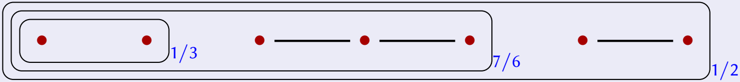


Two words on cluster pictures

What is a cluster picture?

Can be understood as a top-down representation of the Berkovich skeleton. Galois action on the balls (or discoids) is also represented.

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$



Applications of cluster pictures?

Semistable model, lattice of regular forms, regulator, etc... for curves in arithmetic geometry (see e.g. **Kunzweiler2021**; **Best2022**).

Table of contents

OM - Berkovich correspondance

polynomial

$\downarrow$ *OM algorithms*

valuative tree — augmenting valuations using key polynomials φ and value λ

$\updownarrow$ *Ruth 2015, correspondance by the sup-norm*

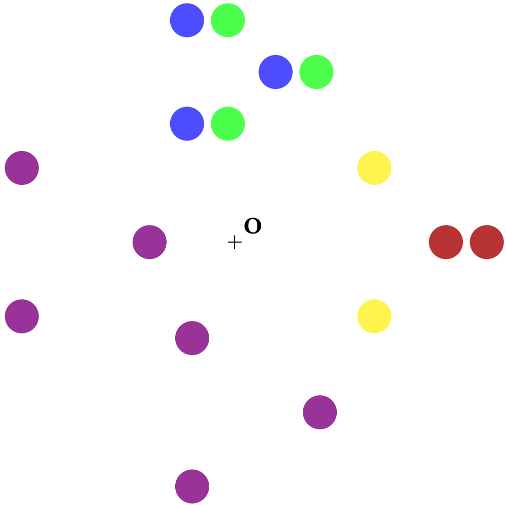
tree of diskoids — Corresponding discoid $D_K(\varphi; \lambda)$

$$\{x \in K : \; v_K(\varphi(x)) \geq \lambda\} = \cup_{i=1}^{d} B_{\overline{K}}(\alpha_i, \gamma)$$

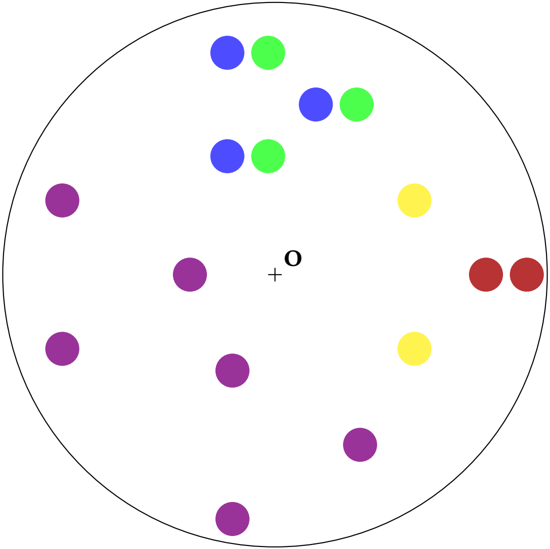
$\downarrow$ *Kunzweiler 2021 + PVW 2025, only if there is no wild ramification*

Berkovich skeleton

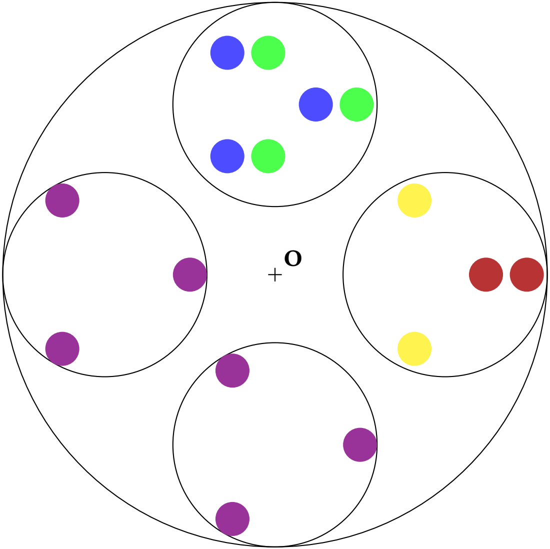
The discoid point of view on the OM algorithm



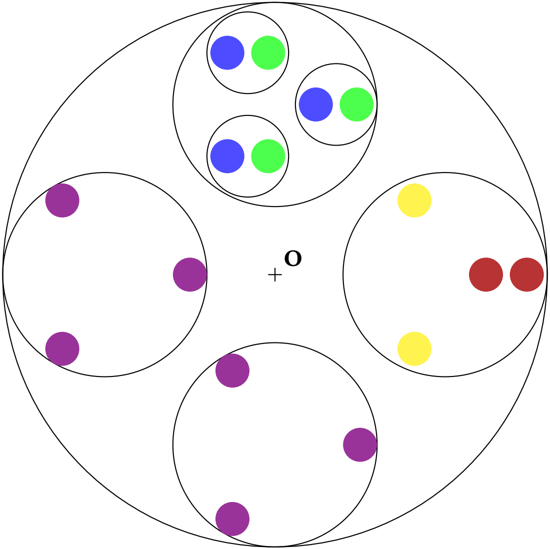
The discoid point of view on the OM algorithm



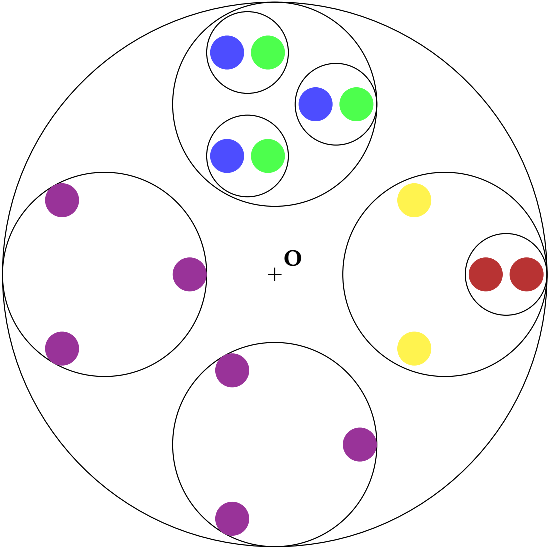
The discoid point of view on the OM algorithm



The discoid point of view on the OM algorithm



The discoid point of view on the OM algorithm



New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

Theorem

Let $f \in A[x]$ be monic of degree d and let $\sigma \in \mathbb{N}$. We have a deterministic algorithm that compute an ***irreducible factorization*** *of $f + O(\pi^\sigma)$:*

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

Theorem

*Let $f \in A[x]$ be monic of degree d and let $\sigma \in \mathbb{N}$. We have a deterministic algorithm that compute an **irreducible factorization** of $f + O(\pi^\sigma)$:*

$$f = f_1^{m_1} \cdots f_s^{m_s} + O(\pi^\sigma),$$

with the truncations $f_i + O(\pi^\sigma)$ distinct and irreducible.

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

Theorem

*Let $f \in A[x]$ be monic of degree d and let $\sigma \in \mathbb{N}$. We have a deterministic algorithm that compute an **irreducible factorization** of $f + O(\pi^\sigma)$:*

$$f = f_1^{m_1} \cdots f_s^{m_s} + O(\pi^\sigma),$$

with the truncations $f_i + O(\pi^\sigma)$ distinct and irreducible. Complexity is in

$$\boxed{\tilde{O}(d^2\sigma + d\sigma^2)}$$

operations in κ. [1]

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

Theorem

*Let $f \in A[x]$ be monic of degree d and let $\sigma \in \mathbb{N}$. We have a deterministic algorithm that compute an **irreducible factorization** of $f + O(\pi^\sigma)$:*

$$f = f_1^{m_1} \cdots f_s^{m_s} + O(\pi^\sigma),$$

with the truncations $f_i + O(\pi^\sigma)$ distinct and irreducible. Complexity is in

$$\boxed{\tilde{O}(d^2\sigma + d\sigma^2)}$$

operations in κ. [1]

Remark

If $f \in k[[t]][x]$ with $\mathrm{char}(k) = 0$ or $> \deg(f)$ then **vdHL25** can do it in $\tilde{O}(d\sigma)$.

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

New version of the OM algorithm

We have adapted the OM algorithm to better handle small precision.

Theorem

Let $f \in A[x]$ be monic of degree d and let $\sigma \in \mathbb{N}$. We have a deterministic algorithm that compute an ***irreducible factorization*** *of $f + O(\pi^\sigma)$:*

$$f = f_1^{m_1} \cdots f_s^{m_s} + O(\pi^\sigma),$$

with the truncations $f_i + O(\pi^\sigma)$ distinct and irreducible. Complexity is in

$$\boxed{\tilde{O}(d^2\sigma + d\sigma^2)}$$

operations in κ. [1]

Remark

If $f \in k[[t]][x]$ with $\mathrm{char}(k) = 0$ or $> \deg(f)$ then **vdHL25** can do it in $\tilde{O}(d\sigma)$.

Remark

The Berkovich skeleton of the roots can still be recovered from the valuative tree of this OM algorithm.

---

[1] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

Table of contents

Definition (Error function at a root)

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d}(x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\Big(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\Big). \end{array}$$

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d} (x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{cccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\right). \end{array}$$

Theorem (Brink's continuity of roots)

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d} (x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\ \alpha - \alpha_l\ )\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d} (x - \alpha_k)$ and $f^* = \prod_{k=1}^{d} (x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Definition (Error function at a root)

Let $f = \prod_{k=1}^{d} (x - \alpha_k) \in K[x]$. For any root α of f we associate the error function

$$\begin{array}{rccc} \Phi_\alpha : & \mathbb{R} \cup \{\infty\} & \longrightarrow & \mathbb{R} \cup \{\infty\} \\ & u & \mapsto & \displaystyle\sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right). \end{array}$$

Theorem (Brink's continuity of roots)

Let $f = \prod_{k=1}^{d} (x - \alpha_k)$ and $f^* = \prod_{k=1}^{d} (x - \alpha_k^*)$. Their roots (in $\overline{K}$) can be ordered such that for all k,

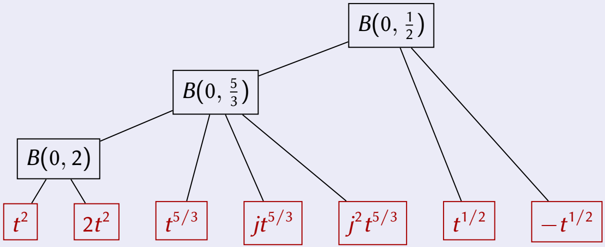
$$\alpha_k^* \in B(\alpha_k, \Phi_{\alpha_k}^{-1}(\mu_0(f - f^*))).$$

Example: from the Berkovich skelon to the distance between roots

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$

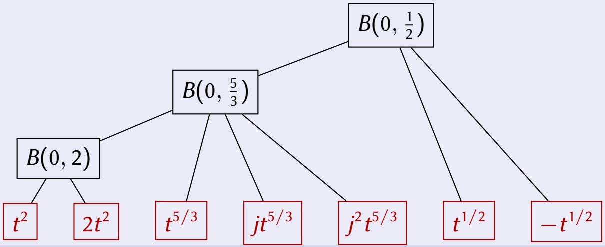
Example: from the Berkovich skelon to the distance between roots

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$



Example: from the Berkovich skelon to the distance between roots

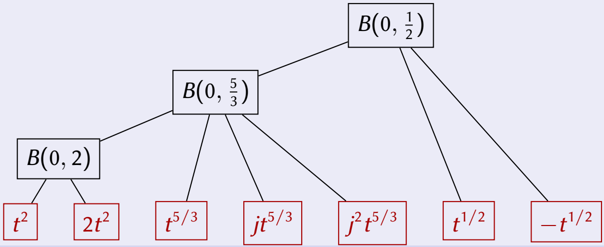
Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$



Matrix of the valuations of the differences of roots

Example: from the Berkovich skelon to the distance between roots

Case of $f_5 := (x - t^2)(x - 2t^2)(x^3 - t^5)(x^2 - t) \in \mathbb{Q}((t))[x]$



Matrix of the valuations of the differences of roots

$$MV_{f_5} = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & \frac{5}{3} & \frac{5}{3} & \frac{5}{3} & \frac{5}{3} \\ \frac{1}{2} & \frac{1}{2} & \frac{5}{3} & 0 & \frac{5}{3} & \frac{5}{3} & \frac{5}{3} \\ \frac{1}{2} & \frac{1}{2} & \frac{5}{3} & \frac{5}{3} & 0 & \frac{5}{3} & \frac{5}{3} \\ \frac{1}{2} & \frac{1}{2} & \frac{5}{3} & \frac{5}{3} & \frac{5}{3} & 0 & 2 \\ \frac{1}{2} & \frac{1}{2} & \frac{5}{3} & \frac{5}{3} & \frac{5}{3} & 2 & 0 \end{pmatrix}$$

Example: distance between roots to Brink's balls

Case of the root $t^{\frac{1}{2}}$ of f_5.

Example: distance between roots to Brink's balls

Case of the root $t^{\frac{1}{2}}$ of f_5.

Corresponding row of MV_{f_5}:

Example: distance between roots to Brink's balls

Case of the root $t^{\frac{1}{2}}$ of f_5.

Corresponding row of MV_{f_5}:

$$\frac{1}{2} \quad \frac{1}{2} \quad \frac{5}{3} \quad \frac{5}{3} \quad \frac{5}{3} \quad 0 \quad 2 \quad \cdot$$

Example: distance between roots to Brink's balls

Case of the root $t^{\frac{1}{2}}$ of f_5.

Corresponding row of MV_{f_5}:

$$\frac{1}{2} \quad \frac{1}{2} \quad \frac{5}{3} \quad \frac{5}{3} \quad \frac{5}{3} \quad 0 \quad 2 \ \cdot$$

Definition of Φ_α:

$$\Phi_\alpha = \sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right).$$

Example: distance between roots to Brink's balls

Case of the root $t^{\frac{1}{2}}$ of f_5.

Corresponding row of MV_{f_5}:

$$\frac{1}{2} \quad \frac{1}{2} \quad \frac{5}{3} \quad \frac{5}{3} \quad \frac{5}{3} \quad 0 \quad 2 \;\cdot$$

Definition of Φ_α:

$$\Phi_\alpha = \sum_{l=1}^{d} \min\left(u, v_{\overline{K}}(\alpha - \alpha_l)\right).$$

Thus

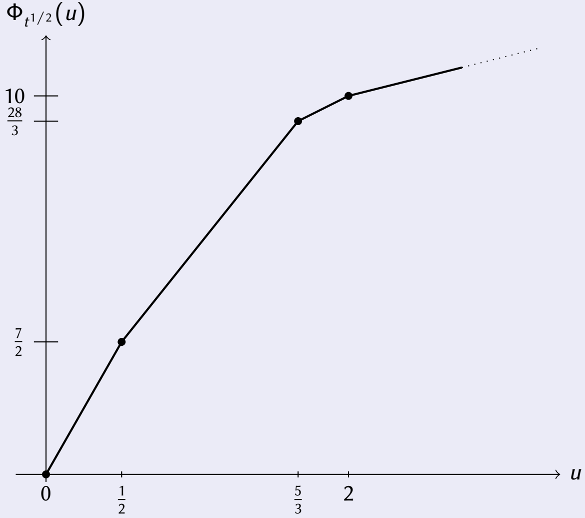
$$\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u.$$

Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$

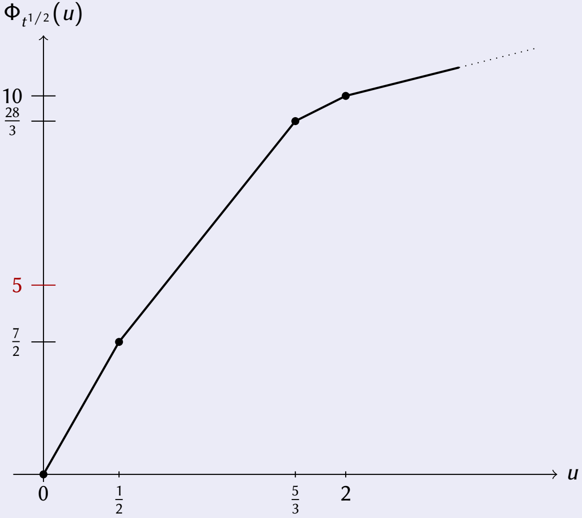
Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



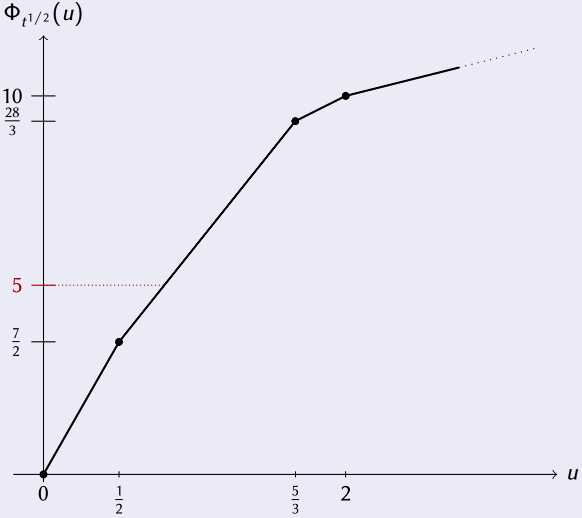
Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



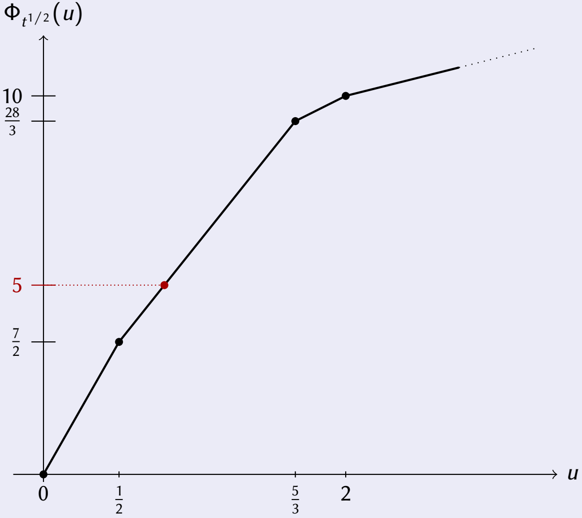
Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



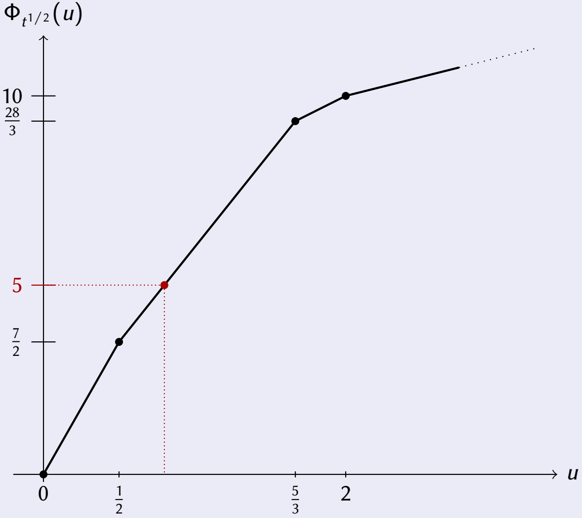
Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



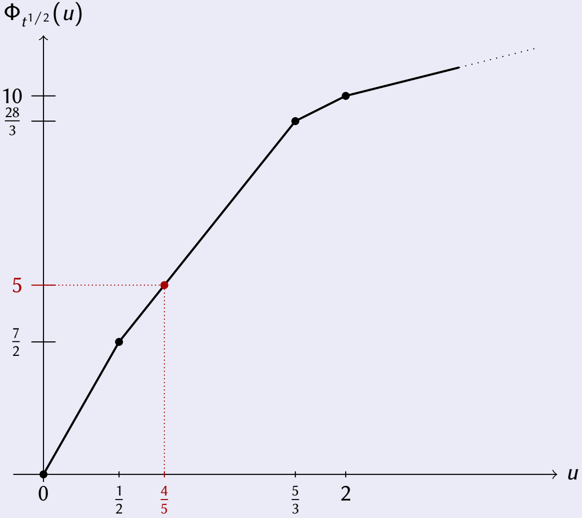
Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$



Inverting Φ_α

Case of $\Phi_{t^{\frac{1}{2}}} = 2\min\left(u, \frac{1}{2}\right) + 3\min\left(u, \frac{5}{3}\right) + \min(u, 2) + u$

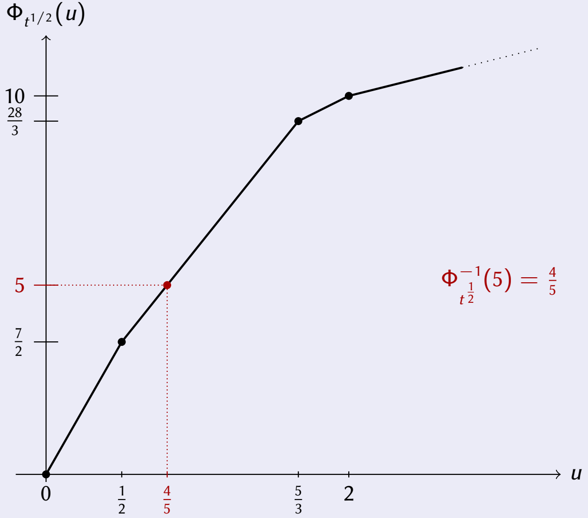


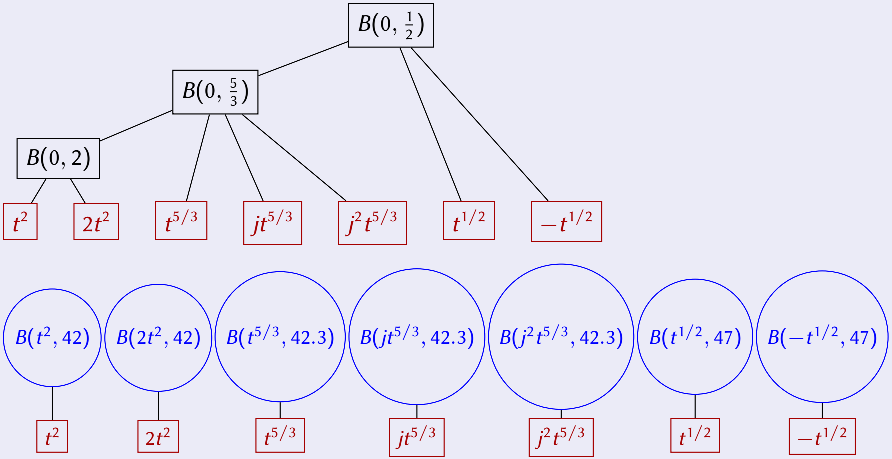
Table of contents

Reorganization as a tree

Case of f_5 and precision 50: presenting all balls

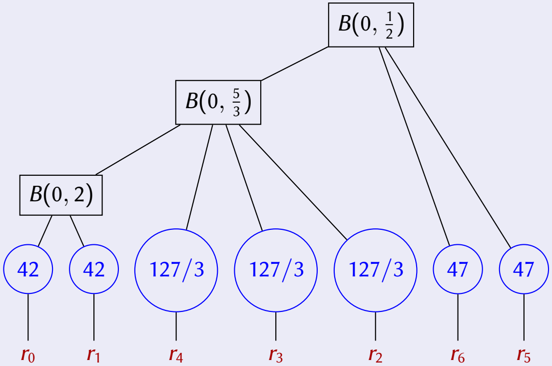
Reorganization as a tree

Case of f_5 and precision 50: presenting all balls



After computation: final form

Approximate Berkovich skeleton of $f_5 + O(\pi^{50})$ $(127/3 = 42.333\ldots)$



Final theorem

Proposition

Let $f \in K[x]$ be monic of degree d. Let $\sigma \in \mathbb{N}$.

[2] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

Final theorem

Proposition

Let $f \in K[x]$ be monic of degree d. Let $\sigma \in \mathbb{N}$.
Then, we have a deterministic algorithm to compute an ***approximate Berkovich skeleton*** *of $f + O(\pi^{\sigma})$ in complexity $\tilde{O}(d^2\sigma + d\sigma^2)$ operations in κ* [2]*,* ***unless wild ramification occurs***

[2] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

Final theorem

Proposition

Let $f \in K[x]$ be monic of degree d. Let $\sigma \in \mathbb{N}$.
Then, we have a deterministic algorithm to compute an ***approximate Berkovich skeleton*** *of $f + O(\pi^\sigma)$ in complexity $\tilde{O}(d^2\sigma + d\sigma^2)$ operations in κ* [2], ***unless wild ramification occurs***

Remark

- We can handle non-simple roots.

[2] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

Final theorem

Proposition

Let $f \in K[x]$ be monic of degree d. Let $\sigma \in \mathbb{N}$.
Then, we have a deterministic algorithm to compute an ***approximate Berkovich skeleton*** *of $f + O(\pi^{\sigma})$ in complexity $\tilde{O}(d^2\sigma + d\sigma^2)$ operations in κ* [2]*,* ***unless wild ramification occurs***

Remark

- We can handle non-simple roots.
- It is also valid for **approximate cluster pictures**.

[2] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

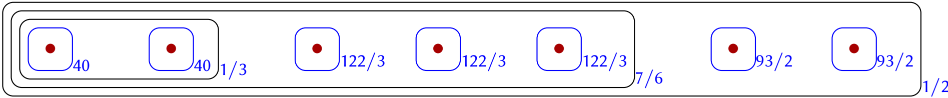
Final theorem

Proposition

Let $f \in K[x]$ be monic of degree d. Let $\sigma \in \mathbb{N}$.
Then, we have a deterministic algorithm to compute an ***approximate Berkovich skeleton*** *of $f + O(\pi^{\sigma})$ in complexity $\tilde{O}(d^2\sigma + d\sigma^2)$ operations in κ*[2]*,* ***unless wild ramification occurs***

Remark

- We can handle non-simple roots.
- It is also valid for **approximate cluster pictures**.



[2] Plus some univariate factorizations over κ of degree sum at most d and whose cost fits in the aimed bound if κ is finite.

Going back to the first example: $x^2 + O(t^{20})$

```
In:   A = FunctionField(QQ,'t')
```

Going back to the first example: $x^2 + O(t^{20})$

```
In:    A = FunctionField(QQ,'t')
In:    t = A.gen()
```

Going back to the first example: $x^2 + O(t^{20})$

```
In:  A = FunctionField(QQ,'t')
In:  t = A.gen()
In:  B = PolynomialRing(A,'X')
In:  X = B.gen()
```

Going back to the first example: $x^2 + O(t^{20})$

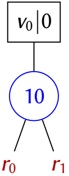
```
In:    A = FunctionField(QQ,'t')
In:    t = A.gen()
In:    B = PolynomialRing(A,'X')
In:    X = B.gen()
In:    f = X^2
```

Going back to the first example: $x^2 + O(t^{20})$

```
In:   A = FunctionField(QQ,'t')
In:   t = A.gen()
In:   B = PolynomialRing(A,'X')
In:   X = B.gen()
In:   f = X^2
In:   Forest_detailed_approximate_berkovich_skeleton(f, 20, 20)
```

Going back to the first example: $x^2 + O(t^{20})$

```
In:    A = FunctionField(QQ,'t')
In:    t = A.gen()
In:    B = PolynomialRing(A,'X')
In:    X = B.gen()
In:    f = X^2
In:    Forest_detailed_approximate_berkovich_skeleton(f, 20, 20)
```



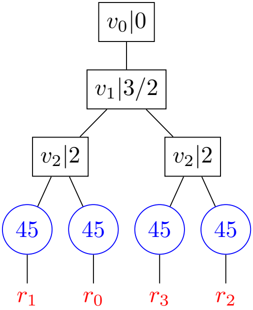
In black: Balls of given log-radii defined by diskoids defined by the following valuations:

- v_0: Gauss valuation induced by (t)-adic valuation

In blue: Balls coming from Brink's inequality.

In red: Roots.

Case of $f_2 = (x^2 - t^3)^2 - xt^6 - t^7 \in k[[t]][x]$ at precision 50



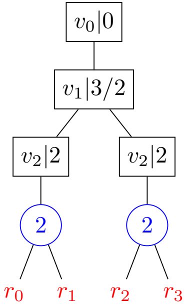
In black: Balls of given log-radii defined by diskoids defined by the following valuations:

- v_0: Gauss valuation induced by (t)-adic valuation
- v_1: Augmentation of v_0 by $v(\Phi_1 = x) = 3/2$
- v_2: Augmentation of v_1 by $v(\Phi_2 = \Phi_1^2 - t^3) = 7/2$

In blue: Balls coming from Brink's inequality.

In red: Roots.

Case of $f_2 = (x^2 - t^3)^2 - xt^6 - t^7 \in k[[t]][x]$ at precision 7



In black: Balls of given log-radii defined by diskoids defined by the following valuations:

v_0: Gauss valuation induced by (t)-adic valuation

v_1: Augmentation of v_0 by $v(\Phi_1 = x) = 3/2$

v_2: Augmentation of v_1 by $v(\Phi_2 = \Phi_1^2 - t^3) = 7/2$

In blue: Balls coming from Brink's inequality.

In red: Roots.

Conclusion

Summary

- Toy implementation of the computation of (approximate) cluster picture, Berkovich skeleton of the roots, valuative tree: `https://gist.github.com/TristanVaccon`

Conclusion

Summary

- Toy implementation of the computation of (approximate) cluster picture, Berkovich skeleton of the roots, valuative tree: `https://gist.github.com/TristanVaccon`
- New OM algorithm that can handle multiplicity and wildness.

Conclusion

Summary

- Toy implementation of the computation of (approximate) cluster picture, Berkovich skeleton of the roots, valuative tree: `https://gist.github.com/TristanVaccon`
- New OM algorithm that can handle multiplicity and wildness.
- Complexity estimate of those computations.

Conclusion

Summary

- Toy implementation of the computation of (approximate) cluster picture, Berkovich skeleton of the roots, valuative tree: `https://gist.github.com/TristanVaccon`
- New OM algorithm that can handle multiplicity and wildness.
- Complexity estimate of those computations.

Future works

- More efficient implementation and integration to SageMath.

Conclusion

Summary

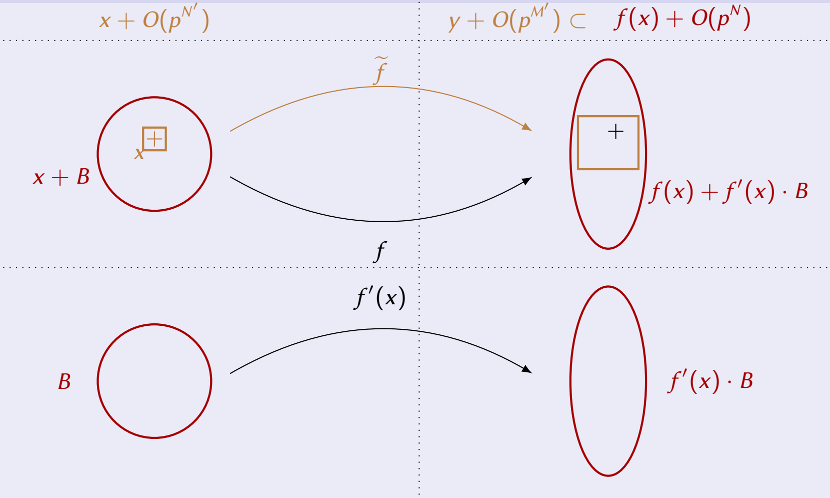
- Toy implementation of the computation of (approximate) cluster picture, Berkovich skeleton of the roots, valuative tree: `https://gist.github.com/TristanVaccon`
- New OM algorithm that can handle multiplicity and wildness.
- Complexity estimate of those computations.

Future works

- More efficient implementation and integration to SageMath.
- Berkovich skeleton and cluster pictures in the case of wild ramification?

Thank you for your attention

Thank you



References

Cluster pictures, discoids, applications

- **Best2022**, *A user's guide to the local arithmetic of hyperelliptic curves*, Best, Alex J *et al.*, 2022.
- **Kunzweiler2021**, *Models of curves and integral differential forms*, Kunzweiler, Sabrina, PhD Thesis 2021.
- **Ruth2015**, *Models of curves and valuations*, Rüth, Julian, PhD Thesis 2015.

OM algorithm and related topics

- **HenselianFacto2024**: *Polynomial factorization over henselian fields*, Alberich-Carramiñana, Maria *et al.*, Journal of FOCM 2024.
- **vdHL2025**: *Plane curve germs and contact factorization* van der Hoeven, Joris and Lecerf, Grégoire in: AAECC 36.1 (2025),
- **PW2022**, *Local polynomial factorisation: improving the Montes algorithm*, Poteaux, Adrien and Weimann, Martin, ISSAC 2022.

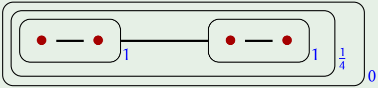
Correspondance between the two

- **PVW2025**, *On OM Algorithms and Cluster Pictures*, Poteaux, Adrien, **V., T.**, and Weimann, Martin, ISSAC 2025.

Bonus: wild (counter-)example

Example (**Kunzweiler2021**)

Let $\varphi_2 := x^4 - 2 \in \mathbb{Q}_2[x]$. Because $\mathrm{val}_2(2\sqrt[4]{2}) = \frac{5}{4}$, its cluster picture is:



However, φ_2 is irreducible and one can not get a longer chain of valuation than $[v_0, v_1(x) = \frac{1}{4}, v_2(x^4 - 2) = +\infty]$.
In other words, the two clusters with two roots and relative depth 1 can not be seen on the chain of valuations.

Valuation on $K[x]$

Definition

Let $V(K[x])$ denote the set of discrete valuations on $K[x]$ which extend v_K and satisfy $v(x) \geq 0$.

Valuation on $K[x]$

Definition

Let $V(K[x])$ denote the set of discrete valuations on $K[x]$ which extend v_K and satisfy $v(x) \geq 0$.

Remark

The Gauss valuation μ_0 is in $V(K[x])$.

Valuation on $K[x]$

Definition

Let $V(K[x])$ denote the set of discrete valuations on $K[x]$ which extend v_K and satisfy $v(x) \geq 0$.

Remark

The Gauss valuation μ_0 is in $V(K[x])$.

Let $v \in V(K[x])$.

Definition

- The **valuation ring** of v is $O_v := \{f \in K[x] : \ v(f) \geq 0\}$.
- Its prime ideal is $O_v^+ := \{f \in K[x] : \ v(f) > 0\}$.
- its **residue ring** $\mathfrak{R}_v := O_v/O_v^+$ (always isomorphic to some $\kappa_v[\zeta]$ with κ_v a finite extension of κ)

Augmented valuation, inductive valuation

Definition

Let $v \in V(K[x])$, φ a **(key)** polynomial over v and $\lambda > v(\varphi)$.
We define the **augmented valuation**

$$v' = [v, v'(\varphi) = \lambda] \in V(K[x])$$

as follows:

$$\boxed{v'(f) := \min_i \left(v(f_i) + i\lambda\right)}$$

where the unique φ-expansion of f is $f = \sum_i f_i \varphi^i$.

Augmented valuation, inductive valuation

Definition

Let $v \in V(K[x])$, φ a **(key)** polynomial over v and $\lambda > v(\varphi)$.
We define the **augmented valuation**

$$v' = [v, v'(\varphi) = \lambda] \in V(K[x])$$

as follows:

$$\boxed{v'(f) := \min_i \left(v(f_i) + i\lambda\right)}$$

where the unique φ-expansion of f is $f = \sum_i f_i \varphi^i$.

Definition

Starting from v_0, an **inductive valuation** is defined recursively as:

$$v_n = [v_0, v_1(\varphi_1) = \lambda_1, \ldots, v_n(\varphi_n) = \lambda_n]$$

via $v_k := [v_{k-1}, v_k(\varphi_k) = \lambda_k]$ for $k = 1, \cdots, n$.

Bonus: Dissection by slope of Newton polygon: Classical case

Definition

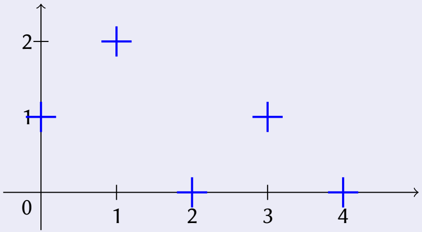
If $P = a_0 + a_1 X + \cdots + a_n X^n$, $NP(P)$ is the lower convex hull of the set

$$\{(i, \mathrm{val}(a_i)) \text{ for i in } [\![0, n]\!]\} .$$

Then P splits with one factor by slope of $NP(P)$ and these factors are relatively prime.

Example

$P = 3 + 9X + 7X^2 + 3X^3 + 2X^4$, over $\mathbb{Q}_3$



P splits into two factors.

Bonus: Dissection by slope of Newton polygon: Classical case

Definition

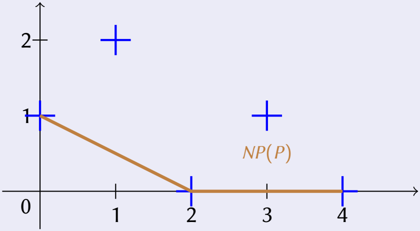
If $P = a_0 + a_1 X + \cdots + a_n X^n$, $NP(P)$ is the lower convex hull of the set

$$\{(i, \mathrm{val}(a_i)) \text{ for i in } [\![0, n]\!]\}\,.$$

Then P splits with one factor by slope of $NP(P)$ and these factors are relatively prime.

Example

$P = 3 + 9X + 7X^2 + 3X^3 + 2X^4$, over $\mathbb{Q}_3$



P splits into two factors.

Bonus: Dissection by slopes of Newton polygon: inductive valuation

Definition

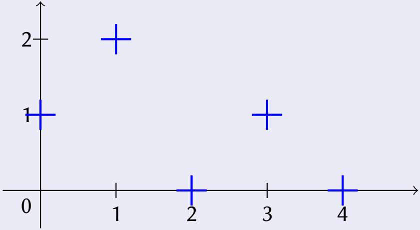
If μ is a valuation, $\varphi \in K[x]$ and $P = a_0 + a_1 X + \cdots + a_n X^n = b_0 + b_1 \varphi + \cdots + a_k \varphi^k$, $NP_{\mu,\varphi}(P)$ is the lower convex hull of the set

$$\{(i, \mu(b_i)) \text{ for i in } [\![0, n]\!]\}.$$

Then in the OM algorithm, applied on P, μ has one descendent by slope of NP_μ.

Example

$P = t + 5t^2\varphi + \varphi^2 + (7t + t^2)\varphi^3 + \varphi^4$, in $\mathbb{Q}((t))[x]$



Here v_0 is the Gaussian valuation.

$$\varphi = x^3 - t.$$

Bonus: Dissection by slopes of Newton polygon: inductive valuation

Definition

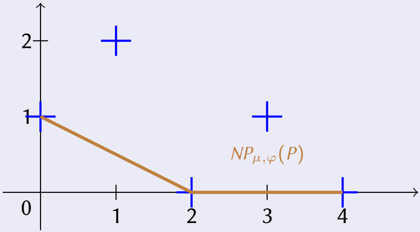
If μ is a valuation, $\varphi \in K[x]$ and $P = a_0 + a_1X + \cdots + a_nX^n = b_0 + b_1\varphi + \cdots + a_k\varphi^k$, $NP_{\mu,\varphi}(P)$ is the lower convex hull of the set

$$\{(i, \mu(b_i)) \text{ for i in } [\![0, n]\!]\}\,.$$

Then in the OM algorithm, applied on P, μ has one descendent by slope of NP_μ.

Example

$P = t + 5t^2\varphi + \varphi^2 + (7t + t^2)\varphi^3 + \varphi^4$, in $\mathbb{Q}((t))[x]$



Here v_0 is the Gaussian valuation.

$$\varphi = x^3 - t.$$

Bonus: Dissection by slopes of Newton polygon: inductive valuation

Definition

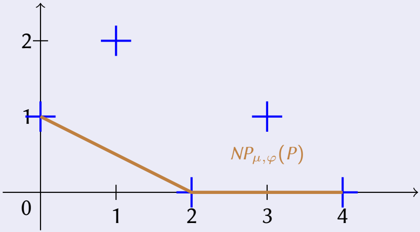
If μ is a valuation, $\varphi \in K[x]$ and $P = a_0 + a_1 X + \cdots + a_n X^n = b_0 + b_1\varphi + \cdots + a_k\varphi^k$, $NP_{\mu,\varphi}(P)$ is the lower convex hull of the set

$$\{(i, \mu(b_i)) \text{ for i in } [\![0, n]\!]\}\,.$$

Then in the OM algorithm, applied on P, μ has one descendent by slope of NP_μ.

Example

$P = t + 5t^2\varphi + \varphi^2 + (7t + t^2)\varphi^3 + \varphi^4$, in $\mathbb{Q}((t))[x]$



Here v_0 is the Gaussian valuation.

$$\varphi = x^3 - t.$$

Then v_0 splits into two valuations.

Bonus: Residual factorization

Theorem

If $f \in O_K[x]$ *is monic with*

$$f \bmod \pi = g \times h \in \kappa[x]$$

with g, h *relatively primes, then* f *splits in* $O_K[x]$ *with factors obtained from* g *and* h *by Hensel-Newton.*

Bonus: Residual factorization

Theorem

If $f \in O_K[x]$ is monic with

$$f \bmod \pi = g \times h \in \kappa[x]$$

with g, h relatively primes, then f splits in $O_K[x]$ with factors obtained from g and h by Hensel-Newton.

Theorem

If $f \in O_K[x]$ is monic with reduction

$$\overline{f} = g \times h \in \mathfrak{R}_v$$

*with g, h relatively primes, then the augmentations of v should follow **two branches**: one for g and one for h.*

Bonus: Refinement of the valuation

Not splitting

If there is no dissection by residual polynomial nor by Newton polygon, then

$$\overline{f} = g^n \in \mathfrak{R}_v,$$

for some irreducible $g \in \mathfrak{R}_v$,

Bonus: Refinement of the valuation

Not splitting

If there is no dissection by residual polynomial nor by Newton polygon, then

$$\overline{f} = g^n \in \mathfrak{R}_v,$$

for some irreducible $g \in \mathfrak{R}_v$,

Refinement or irreducibility

- If $n = 1$: f is **irreducible**.

Bonus: Refinement of the valuation

Not splitting

If there is no dissection by residual polynomial nor by Newton polygon, then

$$\overline{f} = g^n \in \mathfrak{R}_v,$$

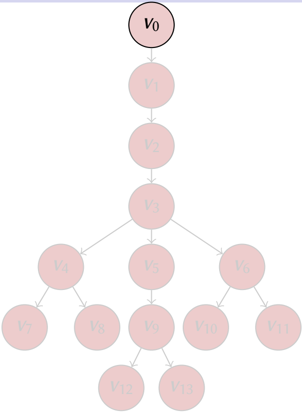
for some irreducible $g \in \mathfrak{R}_v$,

Refinement or irreducibility

- If $n = 1$: f is **irreducible**.
- Else, $n > 1$ and one **refines** v using an n-th root approximation of f.

Bonus: OM algorithm in practice: construction of the valuative tree

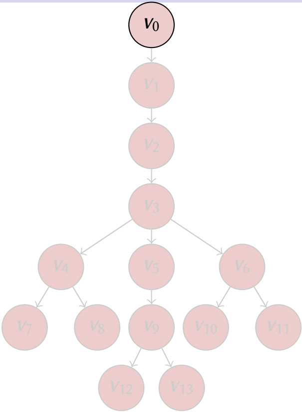
- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.



Inside the computation

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

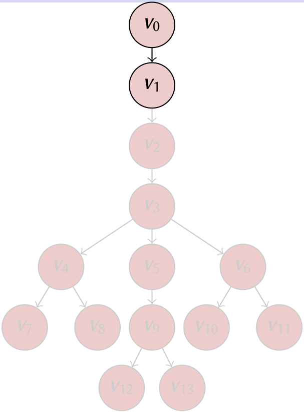


Inside the computation

- Residual polynomial of f_3 in $\mathfrak{R}_{v_0}$: x^{96}.
- $NP_{v_0,x}(f_3)$ has one slope: $-\frac{1}{2}$.
- Extension using $v_1(x) = \frac{1}{2}$.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

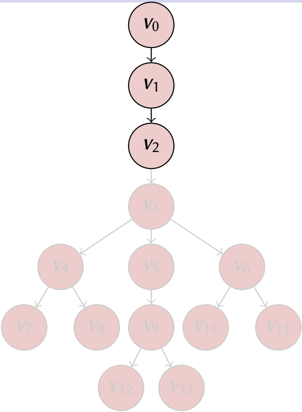


Inside the computation

- Residual polynomial of f_3 in $\mathfrak{R}_{v_1}$: $(x-1)^{48}$.
- $NP_{v_1, x^2 - t}(f_3)$ has one slope: $-\frac{4}{3}$.
- Extension with $v_2(x^2 - t) = \frac{4}{3}$, with $x^2 - t$ being a 48-th approximate root of f_3.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

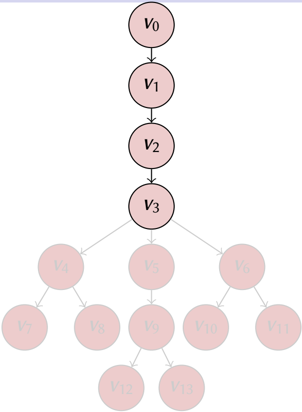


Inside the computation

- Residual polynomial of f_3 in $\mathfrak{R}_{v_2}$: $(x+2)^{16}$.
- $NP_{v_2, x^6 - 3tx^4 + 3t^2x^2 + 2t^4 - t^3}(f_3)$ has one slope: $-\frac{13}{3}$.
- Extension with $v_3(x^6 - 3tx^4 + 3t^2x^2 + 2t^4 - t^3) = \frac{13}{3}$.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

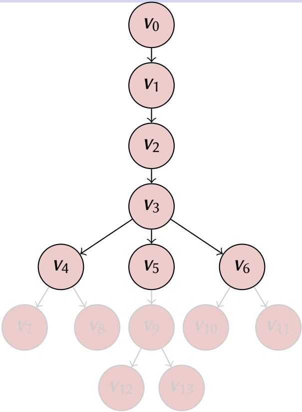


Inside the computation

- Residual polynomial of f_3 in $\mathfrak{R}_{v_3}$: $64(x+1)^2 x^{10} (x^2 - x + 1)^2$.
- Many extensions: v_4, v_5, v_6.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

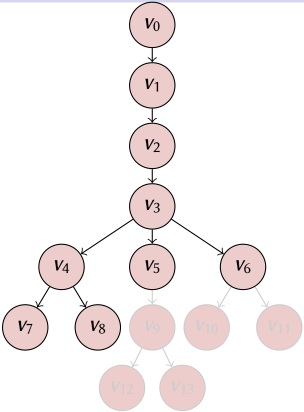


Inside the computation

More splitting and refinements.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

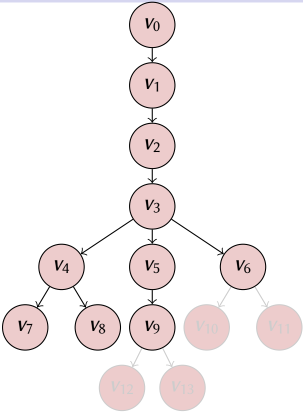


Inside the computation

More splitting and refinements.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

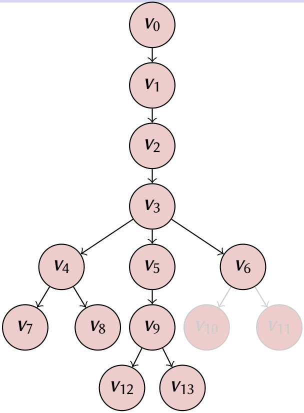


Inside the computation

More splitting and refinements.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.

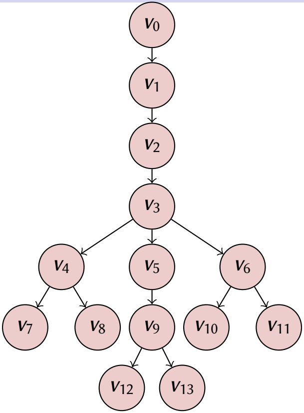


Inside the computation

More splitting and refinements.

Bonus: OM algorithm in practice: construction of the valuative tree

- $\varphi_1 = x^2 - t$,
- $\varphi_2 = \varphi_1^3 + 2\,t^4$,
- $\varphi_3 = (\varphi_2^5 + 4\,t^{27})\,(\varphi_2^3 - 2\,t^{13}) + t^{51}$,
- $f_3 = \varphi_3^2 - t^{82} + t^{83}$.
- f_3 has degree 96.
- f_3 has 6 irreducible factors



Inside the computation

More splitting and refinements.