Arithmetic algorithms for cryptology — 9 February 2015, Paris

The quasi-polynomial algorithm

Razvan Barbulescu
CNRS and IMJ-PRG

Outline of the talk

- Finite fields of small characteristic
- Classical algorithms for DLP in small characteristic
- The quasi-polynomial algorithm

Finite fields

Definition

Given a prime p and an irreducible polynomial $\varphi \in \mathbb{F}_p$, the field defined by φ is the set $\mathbb{F}_p[x]/\langle\varphi\rangle$, endowed by the operations

- addition: add elements as polynomials;
- multiplication: multiply elements as polynomials, then reduce modulo φ;
- inversion: extended Euclid algorithm.

The prime p is the characteristic of the field of modulus φ.

Example

$\varphi = x^2 + x + 1 \in \mathbb{F}_2[x]$ is irreducible because it has no roots, so it defines a field of 4 elements: $0, 1, x, x+1$. In order to compute the inverse of an element, say $a = x$, we use EEA for a and $b = \varphi$:

$$1 = 1 \cdot (x^2 + x + 1) + (x + 1) \cdot x$$

. The gcd is always 1 because φ is irreducible. Here $x^{-1} = x + 1$.

Easy isomorphism

Properties

- If φ_1 and φ_2 are two irreducible polynomials in $\mathbb{F}_p[x]$ of same degree, then
$$\mathbb{F}_p[x]/\langle\varphi_1\rangle \simeq \mathbb{F}_p[x]/\langle\varphi_2\rangle$$
as fields. The isomorphism is computed in polynomial time and corresponds to a change of coordinates.
- For all p and n, there are $(1+o(1))p^n/n$ irreducible polynomials over $\mathbb{F}_p$ of degree n.

$\boxed{\mathbb{F}_{p^n} \text{ or } \mathrm{GF}(p^n) \text{ denote “any field of } p^n \text{ elements”}}$

Example

Polynomials $\varphi_1 = x^3 + x + 1$ and $\varphi_2 = x^3 + x^2 + 1$ are irreducible modulo 2 because they have degree ≤ 3 and no roots. We compute a, b, c so that
$$\varphi_1(a + bx + cx^2) \equiv 0 \mod \varphi_2.$$
Then, we map any element $P(x)$ in the field of modulus φ_1 to the field of modulus φ_2 as follows
$$P(x) \mapsto P(a + bx + cx^2).$$
Here $P(x) = x^2 + x$, and for example $x^2 + x + 1 \mapsto (x^2 + x)^2 + (x^2 + x) + 1 = x^2$.

DLP in finite fields

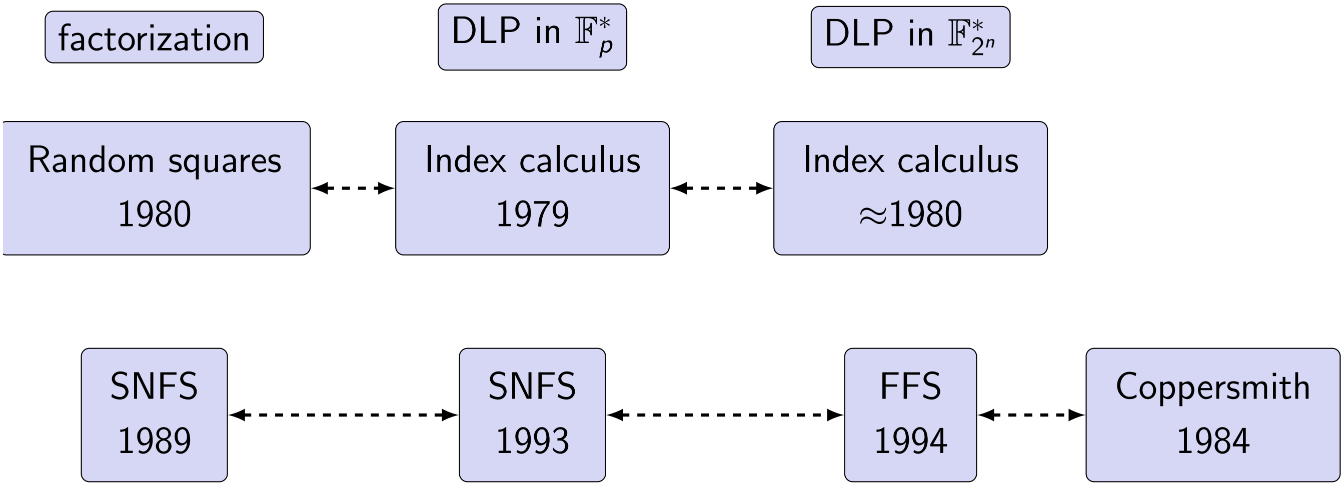
Multiplicative group

- the multiplicative group of $\mathbb{F}_{p^n}^*$ is cyclic
- its cardinality is $p^n - 1$, which can be prime, e.g. $2^{607} - 1$ is prime.
- A proportion of $\varphi(p^n - 1)/(p^n - 1)$ elements are generators, so easy to find.
- For all $a \in (\mathbb{F}_{p^n})^*$, $a^{p^n-1} = 1$.

Advantages

- by selecting a sparse modulus, e.g. $x^n + x + 1$ when irreducible, multiplication becomes faster;
- the complexity to multiply polynomials is slightly better for polynomials than for numbers;
- fast arithmetic is implemented by the C libraries: NTL and gf2x;
- Intel processors offer instructions to multiply polynomials over $\mathbb{F}_2$;
- if dedicated hardware is produced(FPGA), it is easier to implement multiplication in $\mathbb{F}_{2^n}$ and $\mathbb{F}_{3^n}$ than in $\mathbb{F}_p$.

History



Chronology

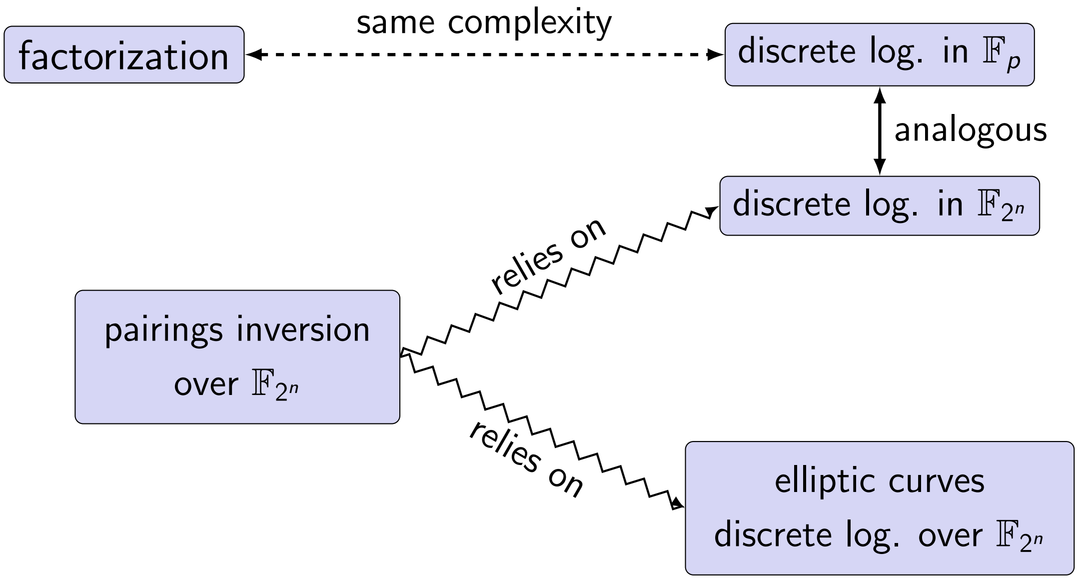
- In 1984, the algorithm of Coppersmith was the first of complexity $L(1/3)$.
- In 1989, the Special number field sieve(SNFS) had the same complexity.
- In 1993 and 1994 SNFS was transfered to DLP in $\mathbb{F}_p$ and, by analogy, to $\mathbb{F}_{2^n}$.
- In 1999, it was explained that the algorithm of Coppersmith was a particular case of FFS(same complexity).

Revival thanks to pairings

Utilization of small characteristic fields

- Since 1984, small characteristic seemed much weaker than large characteristic and factorization, so it was abandoned.
- In 2000 Antoine Joux proposed to use pairings in cryptography, large or small characteristic.
- Pairings in characteristic 2 and 3 are the fastest and lead to many works of implementation.
- In 2013 Joux, Boneh and Franklin received the Gödel prize for their works on pairings.
- The NIST and some private companies were studying pairings for standardization and commercial applications.

Relations of small characteristic DLP to other problems



F_Q is the field of Q elements, Q prime power.

Outline of the talk

- Finite fields of small characteristic
- Classical algorithms for DLP in small characteristic
- The quasi-polynomial algorithm

Smoothness

Definition

A polynomial in $\mathbb{F}_q[t]$ is m-smooth if it factors into polynomials of degree less than or equal to m.

Theorem

The probability that a degree-n polynomial is m-smooth is $1/u^{u(1+o(1))}$ where $u = \frac{n}{m}$.

Cases:

- $n = D$, $m = D/6$ gives a constant probability;
- $n = D$, $m = 1$ gives a probability $1/D! \approx 1/D^D$.

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$
for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$t^5 \equiv 2(t+1)(t^3 + t^2 + 2t + 1) \mod \varphi$$

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$
for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$\begin{aligned} t^5 &\equiv 2(t+1)(t^3+t^2+2t+1) && \bmod \varphi \\ t^6 &\equiv 2(t^2+1)(t^2+t+2) && \bmod \varphi \end{aligned}$$

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$
for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$\begin{aligned} t^5 &\equiv 2(t+1)(t^3+t^2+2t+1) && \bmod \varphi \\ t^6 &\equiv 2(t^2+1)(t^2+t+2) && \bmod \varphi \\ t^7 &\equiv 2(t+2)(t+1)(t+1) && \bmod \varphi \end{aligned}$$

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$ for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$\begin{aligned} t^5 &\equiv 2(t+1)(t^3+t^2+2t+1) && \bmod \varphi \\ t^6 &\equiv 2(t^2+1)(t^2+t+2) && \bmod \varphi \\ t^7 &\equiv 2(t+2)(t+1)(t+1) && \bmod \varphi \end{aligned}$$

The last relation gives:

$$7 \log_g t \equiv \log_g 2 + 1 \log_g(t+2) + 2 \log_g(t+1) \mod 11$$

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$
for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$\begin{aligned} t^5 &\equiv 2(t+1)(t^3+t^2+2t+1) && \mod \varphi \\ t^6 &\equiv 2(t^2+1)(t^2+t+2) && \mod \varphi \\ t^7 &\equiv 2(t+2)(t+1)(t+1) && \mod \varphi \end{aligned}$$

The last relation gives:

$$7 \log_g t \equiv \cancel{\log_g 2} + 1 \log_g(t+2) + 2 \log_g(t+1) \mod 11$$

Proposition

If $a \in \mathbb{F}_q^*$ and ℓ is a factor of $q^k - 1$ coprime to $(q-1)$, then $\log a \equiv 0 \mod \ell$.

Obtaining relations

The finite field $\mathbb{F}_{q^k}$ is represented as $\mathbb{F}_q[t]/\varphi$
for an irreducible polynomial $\varphi \in \mathbb{F}_q[t]$ of degree k.

Example

Take $q = 3$, $k = 5$, $\varphi = t^5 + t^4 + 2t^3 + 1$, $g = t \in \mathbb{F}_{3^5}$ and $\ell = 11 \mid 3^5 - 1$. We have

$$\begin{aligned} t^5 &\equiv 2(t+1)(t^3+t^2+2t+1) && \mod \varphi \\ t^6 &\equiv 2(t^2+1)(t^2+t+2) && \mod \varphi \end{aligned}$$

$$t^8 \equiv \dots$$

The last relation gives:

$$\begin{aligned} 7\log_g t &\equiv 1\log_g(t+2) + 2\log_g(t+1) && \mod 11 \\ 8\log_g(t+1) &\equiv 1\log_g(t+2) && \mod 11 \\ 9\log_g(t+2) &\equiv 2\log_g t && \mod 11 \end{aligned}$$

We find $\log_g(t+1) \equiv 158 \mod 11$ and $\log_g(t+2) \equiv 54 \mod 11$.

Descent

Example (cont'd)

Let us compute $\log_g P$ for an arbitrary polynomial, say $P = t^4 + t + 2$.
We have

$$\begin{aligned} P^2 &\equiv t^4 + t^3 + 2t^2 + 2t + 2 && \bmod \varphi \\ P^3 &\equiv 2(t+1)(t+2)(t^2+1) && \bmod \varphi \\ P^4 &\equiv (t+1)(t+2)t^2 && \bmod \varphi. \end{aligned}$$

Descent

Example (cont'd)

Let us compute $\log_g P$ for an arbitrary polynomial, say $P = t^4 + t + 2$.
We have

$$\begin{aligned} P^2 &\equiv t^4 + t^3 + 2t^2 + 2t + 2 && \mod \varphi \\ P^3 &\equiv 2(t+1)(t+2)(t^2+1) && \mod \varphi \\ P^4 &\equiv (t+1)(t+2)t^2 && \mod \varphi. \end{aligned}$$

By taking discrete logarithms we obtain

$$4 \log_g P = 1 \log_g(t+1) + 1 \log_g(t+2) + 2 \log_g t.$$

So $\log_g P = 114$.

Discrete logarithms of constants

Here ℓ is a prime factor of the group order $q^k - 1$, larger than $q - 1$.

Elements of $\mathbb{F}_q$

Elements of $\mathbb{F}_q \subset \mathbb{F}_{q^k}$ are represented in $\mathbb{F}_q[t]/\langle\varphi\rangle$ by constants a. They satisfy $a^{q-1} = 1$, so we have

$$\log_g(a^{q-1}) \equiv \log_g(1) \equiv 0 \mod \ell.$$

Hence,

$$(q-1)\log_g a \equiv 0 \mod \ell.$$

Since ℓ is prime and larger than $q - 1$,

$$\log_g a \equiv 0 \mod \ell.$$

Outline of the talk

- Finite fields of small characteristic
- Classical algorithms for DLP in small characteristic
- The quasi-polynomial algorithm

Main result

Theorem (based on heuristic assumptions)

Let K be any finite field $\mathbb{F}_{q^k}$. A discrete logarithm in K can be computed in heuristic time

$$\max(q,k)^{O(\log k)}.$$

Cases:

- $K = \mathbb{F}_{2^n}$, with prime n. Complexity is $n^{O(\log n)}$. Much better than $L_{2^n}(1/4 + o(1)) \approx 2^{\sqrt[4]{n}}$ (previous state-of-art: Joux 2013).
- $K = \mathbb{F}_{q^k}$, with $q = k^{O(1)}$. Complexity is $\log Q^{O(\log\log Q)}$, where $Q = \#K$. Again, this is $L_Q(o(1))$.
- $K = \mathbb{F}_{q^k}$, with $q \approx L_{q^k}(\alpha)$. Complexity is $L_{q^k}(\alpha + o(1))$, i.e. better than Joux-Lercier or FFS for $\alpha < 1/3$.

A well-chosen model for $\mathbb{F}_{q^{2k}}$

Simple case first

We suppose first $k \approx q$ and $k \leq q+2$.

Choosing φ (same as for Joux' algorithm)

Try random $h_0, h_1 \in \mathbb{F}_{q^2}[t]$ with $\deg h_0, \deg h_1 \leq 2$ until $T(t) := h_1(t)t^q - h_0(t)$ has an irreducible factor φ of degree k.

Heuristic

The existence of h_0 and h_1 is heuristic, but found in practice in $O(k)$ trials.

Properties of φ

- $h_1(t)t^q \equiv h_0(t) \mod \varphi$;
- $P(t^q) \equiv P\left(\frac{h_0}{h_1}\right) \mod \varphi$;
- $P^q \equiv \tilde{P}(t^q) \equiv \tilde{P}\left(\frac{h_0}{h_1}\right) \mod \varphi$,
  where the tilde denotes the conjugation in $\mathbb{F}_{q^2}$.

A famous identity

Recall the identity

$$x^q - x = \prod_{\alpha \in \mathbb{F}_q} (x - \alpha).$$

We further have $x^q y - x y^q = \prod_{(\alpha:\beta) \in \mathbb{P}^1(\mathbb{F}_q)} (\beta x - \alpha y)$.

A famous identity

Recall the identity

$$x^q - x = \prod_{\alpha \in \mathbb{F}_q} (x - \alpha).$$

We further have $x^q y - x y^q = \prod_{(\alpha:\beta) \in \mathbb{P}^1(\mathbb{F}_q)} (\beta x - \alpha y)$.

A machine to make relations

- $x = t$ and $y = 1$: $h_0/h_1 - t \equiv t^q - t \equiv \prod_{\alpha \in \mathbb{F}_q} (t - \alpha)$.
  If the numerator of the left hand side is smooth, we obtain relations among linear polynomials.
- $x = t + a$, $a \in \mathbb{F}_q$, and $y = 1$: same relation.
- $x = t + a$, $a \in \mathbb{F}_{q^2}$, and $y = 1$: new relations. Joux' algorithm uses this idea.
- Let P be the polynomial whose logarithm is requested.

A famous identity

Recall the identity

$$x^q - x = \prod_{\alpha \in \mathbb{F}_q} (x - \alpha).$$

We further have $x^q y - x y^q = \prod_{(\alpha:\beta) \in \mathbb{P}^1(\mathbb{F}_q)} (\beta x - \alpha y)$.

A machine to make relations

- $x = t$ and $y = 1$: $h_0/h_1 - t \equiv t^q - t \equiv \prod_{\alpha \in \mathbb{F}_q} (t - \alpha)$.
  If the numerator of the left hand side is smooth, we obtain relations among linear polynomials.
- $x = t + a$, $a \in \mathbb{F}_q$, and $y = 1$: same relation.
- $x = t + a$, $a \in \mathbb{F}_{q^2}$, and $y = 1$: new relations. Joux' algorithm uses this idea.
- Let P be the polynomial whose logarithm is requested.
  $x = aP + b$ and $y = cP + d$, $a, b, c, d \in \mathbb{F}_{q^2}$: let us show that the left side is congruent to a small degree polynomial, whereas the right hand side is smooth in some new sense.

The right hand side is “smooth”

$$
\begin{aligned}
(aP+b)^q(cP+d)-(aP+b)(cP+d)^q &= \prod_{(\alpha,\beta)\in\mathbb{P}^1(\mathbb{F}_q)} \beta(aP+b)-\alpha(cP+d) \\
&= \prod_{(\alpha,\beta)\in\mathbb{P}^1(\mathbb{F}_q)} (-c\alpha+a\beta)P-(d\alpha-b\beta) \\
&= \lambda \prod_{(\alpha,\beta)\in\mathbb{P}^1(\mathbb{F}_q)} \left(P-\frac{d\alpha-b\beta}{a\beta-c\alpha}\right),
\end{aligned}
$$

Here $q+1$ out of the q^2+1 elements of $\{1\}\bigcup\{P+\gamma:\gamma\in\mathbb{F}_{q^2}\}$ occur.

The left hand side is small

For $m \in \mathrm{GL}_2(\mathbb{F}_{q^2})$, let $\mathcal{L}_m$ be the residue

$$\mathcal{L}_m := h_1^{\deg P}\left((aP+b)^q(cP+d) - (aP+b)(cP+d)^q\right) \mod \varphi(t).$$

The left hand side is small

For $m \in \mathrm{GL}_2(\mathbb{F}_{q^2})$, let $\mathcal{L}_m$ be the residue

$$\mathcal{L}_m := h_1^{\deg P}\left((aP+b)^q(cP+d) - (aP+b)(cP+d)^q\right) \mod \varphi(t).$$

We have $\deg \mathcal{L}_m \leq 3 \deg P$. Indeed, we have

$$\begin{aligned} \mathcal{L}_m &= h_1^{\deg P}(\tilde{a}\tilde{P}(t^q) + \tilde{b})(cP+d) - (aP(t)+b)(\tilde{c}\tilde{P}(t^q) + \tilde{d}) \\ &= h_1^{\deg P}\left(\tilde{a}\tilde{P}\left(\frac{h_0}{h_1}\right) + \tilde{b}\right)(cP+d) - (aP+b)\left(\tilde{c}\tilde{P}\left(\frac{h_0}{h_1}\right) + \tilde{d}\right). \end{aligned}$$

For a constant proportion of matrices m, $\mathcal{L}_m$ is $(\deg P)/2$-smooth.

Procedure to "break" a polynomial P

Each matrix m in the quotient set $\mathcal{P}_q := \mathrm{PGL}_2(\mathbb{F}_{q^2})/\mathrm{PGL}_2(\mathbb{F}_q)$ such that $\mathcal{L}_m$ is $(\deg P)/2$-smooth leads to a different equation

$$\prod_i P_{i,m}^{e_{i,m}} = \lambda \prod_{\gamma \in \mathbb{P}^1(\mathbb{F}_{q^2})} (P+\gamma)^{v_m(\gamma)},$$

where

- $\deg P_i \leq (\deg P)/2$;
- $v_m(\gamma)$ are integer exponents;
- λ is a constant in $\mathbb{F}_{q^2}$.

By taking discrete logarithm we find

$$\sum_i e_{i,m} \log P_{i,m} \equiv \sum_\gamma v_m(\gamma) \log(P+\gamma) \mod \ell.$$

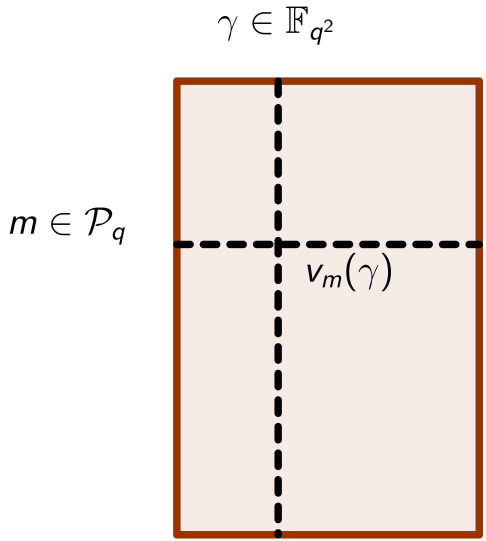
Heuristic

We have enough equations and we can combine them to obtain

$$\sum_{i,m} e'_{i,m} \log P_{i,m} \equiv \log P \mod \ell.$$

Linear algebra step for P

Since $\#\mathrm{PGL}_2(\mathbb{F}_{q^i}) = q^{3i} - q^i$, $\#\mathcal{P}_q = q^3 + q$. A constant fraction give linear equations among logarithms, so the matrix below has more rows than columns.



The heuristic states that we can combine the rows to obtain row

$$(1, 0, \ldots, 0).$$

Building block of the quasi-polynomial algorithm

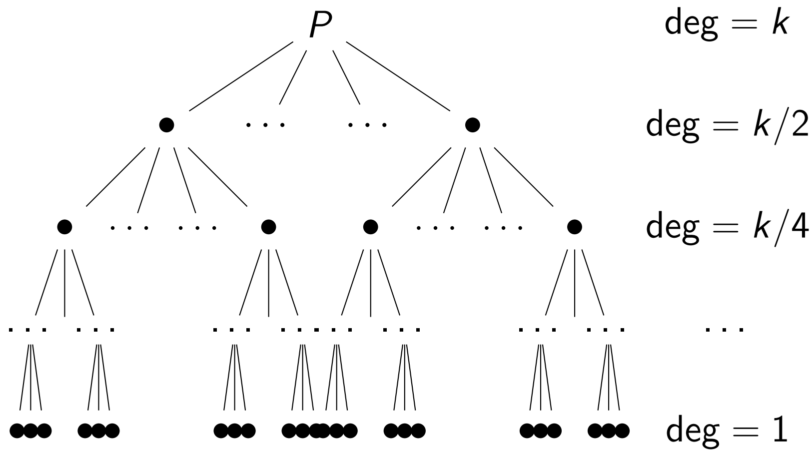
We have just proved:

Proposition (Under heuristic assumptions)

There exists an algorithm whose complexity is polynomial in q and k and which can be used for the following two tasks.

1. Given an element of $\mathbb{F}_{q^{2k}}$ represented by a polynomial $P \in \mathbb{F}_{q^2}[t]$ with $2 \leq \deg P \leq k-1$, the algorithm returns an expression of $\log P$ as a linear combination of at most $O(kq^2)$ logarithms $\log P_i$ with $\deg P_i \leq \lceil \frac{1}{2} \deg P \rceil$ and of $\log h_1$.
2. The algorithm returns the logarithm of h_1 and the logarithms of all the elements of $\mathbb{F}_{q^{2k}}$ of the form $t + a$, for a in $\mathbb{F}_{q^2}$.

Complexity



Tree characteristics

- depth$=\log k$ because we half the degree at each level;
- arity$=O(q^2 k)$ because the sons are polynomials in the LHS of the q^2 equations used;
- number of nodes$=q^{O(\log k)}$ because $k \leq q+2$.

Conclusion

- DLP in small characteristic finite fields was introduced because it has faster arithmetic;
- it was abandoned in 1984 because of the algorithm of Coppersmith
- it was revived in 2000 by Joux
- it is <u>asymptotically</u> weak because of the quasi-polynomial algorithm.
- Small characteristic pairings are broken for the sizes proposed for cryptography.