Discrete logarithms: Recent progress in small characteristic

Antoine Joux
CryptoExperts
Chaire de Cryptologie de la Fondation de l'UPMC — LIP6

Journées C2, 28 mars 2014

Discrete logarithms

- Given a multiplicative group G with generator g
- Computing discrete logarithms is inversing $n \to g^n$
- Hard in general and used as a hard problem in cryptography
- Algorithmic viewpoint
  - Generic algorithms (for any G)
    - Pohlig-Hellman
    - Baby step, Giant step and Pollard's Rho
  - Specific algorithms (make use of group representation)

Classical groups for Dlog in Cryptography

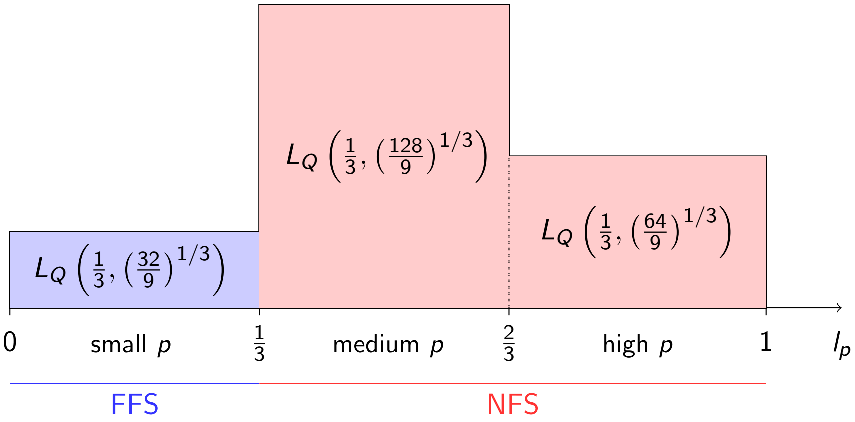
- Integers modulo p
- More general finite fields $\mathbb{F}_{p^k}$
- Elliptic curves over finite fields

Index calculus algorithms

- Relation generation phase
  - Generates many sparse equations
  - Modulo group order for discrete log
    (Modulo 2 for factoring)
- Linear algebra phase
  - Large sparse system
  - Numbers of unknowns in range up to dozens of millions
  - Number of equations potentially very large
  - Need to use large computers to solve such systems
- Individual logarithm phase

Complexity of Index calculus algorithms (before 2013)

$$L_Q(\beta, c) = \exp((c + o(1))(\log Q)^{\beta}(\log\log Q)^{1-\beta}).$$



Discrete Logarithms, simplified FFS [JL06]

- Finite field of the form $\mathbb{F}_{p^k}$
- Choose two univariate polynomials f_1 and f_2
  - with degrees d_1 and d_2 and $d_1 d_2 \geq k$.
  - Such that $x - f_1(f_2(x))$ has:
    - an irreducible factor of degree k (modulo p).
- This defines the finite field by the relations:
  - $x = f_1(y)$ and $y = f_2(x)$

Commutative diagram

$$
\begin{tikzcd}
 & \mathbb{F}_p[X, Y] \arrow[dl, "Y \mapsto f_2(X)"] \arrow[dr, "X \mapsto f_1(Y)"] & \\
\mathbb{F}_p[X] \arrow[dr, "X \mapsto x"] & & \mathbb{F}_p[Y] \arrow[dl, "Y \mapsto y"] \\
 & \mathbb{F}_{p^k} &
\end{tikzcd}
$$

Discrete Logarithms, simplified FFS [JL06]

- Optimal for $p = L_{p^k}(1/3)$
- Choose smoothness basis $x - \alpha$ and $y - \alpha$
- Consider elements:

$$\begin{aligned} xy + ay + bx + c &= x\, f_2(x) + af_2(x) + bx + c \\ &= y\, f_1(y) + ay + bf_1(y) + c \end{aligned}$$

- When both sides split $\Rightarrow$ Relation
- Heuristic cost of finding relation (sieving):

$$(d_1 + 1)!\,(d_2 + 1)!$$

- Individual log. descent negligible compared to initial phase

Nice special case – Kummer extensions

- Assume $k|p-1$, then $\mathbb{F}_{p^k}$ can be defined by $x^k - t$
- If $k = d_1 d_2 - 1$, let $y = x^{d_1}$ and $tx = y^{d_2}$ (*i.e.* $tx = x^{d_1 d_2}$)
- Reduces size of smoothness basis by k
  - Indeed:

$$\begin{aligned}(X+\alpha)^p &= X^p + \alpha = t^{(p-1)/k} X + \alpha = \mu(X + \alpha/\mu), \\ (Y+\alpha)^p &= \mu^{d_1}(Y + \alpha/\mu^{d_1}).\end{aligned}$$

    where μ is a k-th root of unity in $\mathbb{F}_p$.

- Can be generalized to $k = d_1 d_2 + 1$ using $y = x^{d_1}$ and $x = t/y^{d_2}$

Linear change of variables [J13]

- Further restrict to $y = x^{d_1}$
- Then:

$$xy + ay + bx + c \quad = \quad x^{d_1+1} + ax^{d_1} + bx + c$$

- Perform change of variable: $x = aX$, we get:

$$a^{d_1+1}(X^{d_1+1} + X^{d_1} + b \cdot a^{-d_1}(X + c/(ab)).$$

- Change of variable does not affect splitting property
- One good left-hand side $\Rightarrow$ p good left-hand sides
- Amortized cost of relation reduced to

$$\left(\frac{(d_1+1)!}{p-1} + 1\right) \cdot (d_2+1)!$$

Case of Kummer extensions

- Assume $k|p-1$, i.e. $\mathbb{F}_{p^k}$ can be defined by $x^k - t$
- If $k = d_1 d_2 - 1$, let $y = x^{d_1}$ and $tx = y^{d_2}$
  - $x^{d_1+1} + ax^{d_1} + bx + c \Rightarrow a^{d_1+1}(X^{d_1+1} + X^{d_1} + b \cdot a^{-d_1}(X + c/(ab))$.
  - $(y^{d_2+1} + by^{d_2})/t + ay + c \Rightarrow$ $b^{d_2+1}\left((Y^{d_2+1} + Y^{d_2})/t + a \cdot b^{-d_2}(Y + c/(ab)\right)$.
- In both cases $\lambda = c/(ab)$ is shared by the two sides

Kummer extensions – Reassembling two sides

- Assume that:
  - $X^{d_1+1} + X^{d_1} + \theta_X(X + \lambda)$ splits and
  - $(Y^{d_2+1} + Y^{d_2})/t + \theta_Y(Y + \lambda)$ splits.
- Find a and b such that $\theta_X = b \cdot a^{-d_1}$ and $\theta_Y = a \cdot b^{-d_2}$?
- This implies $\theta_X^{d_2}\theta_Y = a^{-d_1 d_2+1} = a^{-k}$.
  - Possible iff $\theta_X^{d_2}\theta_Y$ is a k-th power
  - Gives k (conjugate) solutions !
  - From a recover b and c
  - Roots obtained by change of variable

Impact in the medium prime case

- In theory, reduces constant in $L(1/3)$ complexity of function field sieve.

- In practice, Kummer extensions esp. good for records:
  - First 1175-bit field $\mathbb{F}_{p^{47}}$ with p close to 2^{25}
  - Then 1425-bit field $\mathbb{F}_{p^{57}}$ with p close to 2^{25}
  - Previous finite field record was 923 bits
  - Timings: about 32000 CPU-hours compared to 895000 CPU-hours

- $47 = 6 \cdot 8 - 1$
- $57 = 7 \cdot 8 + 1$

Small characteristic – Setting [J13b]

- Define finite field by a relation:

$$x^{p^\ell} = \frac{h_0(x)}{h_1(x)}, \qquad \begin{array}{c} \mathbb{F}_{p^{k\ell}} \\ \big|k \\ \mathbb{F}_{p^\ell} \end{array}$$

  gives degree $k = \deg(I(x))$ extension, where $I(x)$ is a divisor of $h_1(x)x^{p^\ell} - h_0(x)$.

- We have a systematic relation:

$$x^{p^\ell} - x \;=\; \prod_{\alpha \in \mathbb{F}_{p^\ell}} (x - \alpha).$$

Small characteristic – Basic idea [J13b]

- Use more general change of variable: $x = \frac{aX+b}{cX+d}$, we get:

$$(cX+d)\cdot(aX+b)^{p^\ell} - (aX+b)\cdot(cX+d)^{p^\ell} =$$
$$(cX+d)\cdot \prod_{\alpha\in\mathbb{F}_{p^\ell}} ((a-\alpha\, c)X + (b-\alpha\, d))$$

- Moreover, after expanding the left-hand size, we find:

$$(ca^q - ac^q)X^{q+1} + (da^q - bc^q)X^q + (cb^q - ad^q)X + (db^q - bd^q),$$

where $q = p^\ell$.
It becomes a low degree polynomial after multiplying by h_1 and replacing $h_1(X)\, X^q$.

- As a consequence, multiplicative relations are very easy to find

Small characteristic – Choice of a, b, c and d

- If a, b, c and d are in $\mathbb{F}_q$ left-hand side is:

$$(ad - bc)(X^q - X) \Rightarrow \text{Trivial relation}$$

- Take a, b, c and d in small extension field such as $\mathbb{F}_{q^2}$
- Some choices of (a, b, c, d) are equivalent. Good parametrization is:

$$PGL_2(\mathbb{F}_{q^2})/PGL_2(\mathbb{F}_q)$$

Small characteristic – Resulting Complexity [J13b]

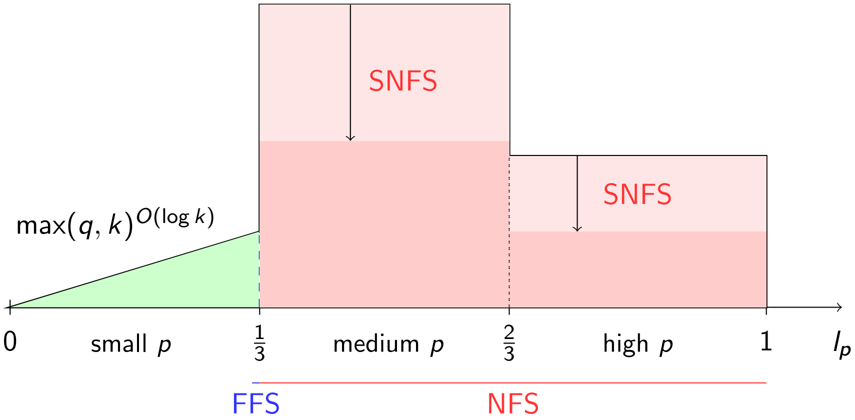
- Logarithms of smoothness basis in polynomials time
  - Because base field is very small compared to extension field
- Hard part is individual logarithms
  - Usual descent algorithm not good enough
  - Need to be completed by new descent algorithm
  - Resulting complexity is:

$$L(\cancel{1/4+}o(1)).$$

  - Practical application:
    - New records in $\mathbb{F}_{2^{1778}}$, $\mathbb{F}_{2^{4080}}$ and $\mathbb{F}_{2^{6168}}$ recently announced
    - Other records by Göloğlu, Granger, McGuire and Zumbrägel

- $1778 = 2 \cdot 7 \cdot (2^7 - 1)$, 220 CPU-hours
- $4080 = 2 \cdot 8 \cdot (2^8 - 1)$, 14100 CPU-hours
- $6168 = 3 \cdot 8 \cdot (2^8 + 1)$ 550 CPU-hours

Resulting Complexity



Descent strategies

- Continued fractions (high degrees)
- Classical descent (for high to mid degrees, need subfield)
- Bilinear descent (for mid to low degrees)
- Quasi-polynomial descent (all degrees)

Continued fractions

- Given target $Z(x)$ find matrix:

$$\begin{pmatrix} A_1(x) & A_2(x) \\ B_1(x) & B_2(x) \end{pmatrix}, \text{such that}$$

$$Z(x) \equiv \frac{A_1(x)}{B_1(x)} \equiv \frac{A_2(x)}{B_2(x)} \pmod{I(x)}.$$

- With continued fraction or half-Gcd algorithms.
- Reduce degree by 2. Many representations:

$$Z(x) \equiv \frac{c_1(x)A_1(x) + c_2(x)A_2(x)}{c_1(x)B_1(x) + c_2(x)B_2(x)} \pmod{I(x)}.$$

Classical descent

- Need two variables x and y
- If $q = p^\ell$, let:

$$\begin{aligned} y &= x^{p^{\ell_1}} \qquad \text{then} \\ y^{p^{\ell_2}} &= x^{p^\ell} = \frac{h_0(x)}{h_1(x)}. \end{aligned}$$

- Let $F(x, y)$ be a (low degree) bivariate polynomial, then:

$$F(x, x^{p^{\ell_1}})^{p^{\ell_2}} = \tilde{F}(x^{p^{\ell_2}}, h_0(x)/h_1(x)) \quad \text{in } \mathbb{F}_{q^k}.$$

- Need to force $z(x)$ as divisor of $F(x, x^{p^{\ell_1}})$ (linear algebra)
- Low arity in descent

Bilinear descent

- Search for k_1 and k_2 such that:

$$z(x)|\mathrm{Num}\left(\tilde{k_1}\left(\frac{h_0(x)}{h_1(x)}\right)k_2(x) - k_1(x)\tilde{k_2}\left(\frac{h_0(x)}{h_1(x)}\right)\right).$$

- Then $z(x)$ appears on the left in:

$$k_1(x)^{p^\ell} \cdot k_2(x) - k_1(x) \cdot k_2(x)^{p^\ell} = k_2(x) \cdot \prod_{\alpha \in \mathbb{F}_{p^\ell}} (k_1(x) - k_2(x)).$$

- Arity q in descent

How to find k_1 and k_2 ?

- Algebraic approach : divisibility condition as a bilinear system
  - In general, use Groebner bases
  - For low-degree, it degenerates into easy linear algebra
- Lattice reduction approach :
  - Further assume that k_1 and k_2 split into linear term
  - Since $z(x)$ is irreducible, it encodes a finite field
  - Take logarithms of elements :

$$\frac{x-\alpha}{h_0(x)/h_1(x)-\alpha^q}.$$

  - Find low weight sum of logarithms equal to 0
- Is there a more direct/efficient approach ?

Quasi-polynomial descent

- Make $z(x)$ appear on the right in:

$$(z(x)+\lambda_1)^{p^\ell}\cdot(z(x)+\lambda_2)-(z(x)+\lambda_1)\cdot(z(x)+\lambda_2)^{p^\ell}=$$
$$(z(x)+\lambda_2)\cdot\prod_{\alpha\in\mathbb{F}_{p^\ell}}((1+\alpha)z(x)+\lambda_1+\alpha\,\lambda_2)$$

- Need $\approx q^2$ equations.
- Simultaneous descent of all $z(x)+\lambda$
- Requires extra linear algebra step
- Arity q^2 in descent

Descent Tree

- Continued fractions, **at most one application**
- Classical descent, **many levels possible**
- Bilinear descent, **in practice 4-5 levels max.**
- Quasi-polynomial descent **in practice 2 levels max.**

Conclusion

Questions ?