

# Discrete logarithm in finite fields of small characteristic

Édouard Rousseau  
*Université de Versailles*

June 1, 2017

# CONTENTS

## INTRODUCTION

- The discrete logarithm problem

- Terminology

- Historical background

## INDEX CALCULUS

- Overview

- An example

## QUASI-POLYNOMIAL ALGORITHMS

- Barbulescu, Gaudry, Joux and Thomé algorithm

- Granger, Kleinjung and Zumbrägel algorithm

# CONTEXT

- ▶  $G = \langle g \rangle$  cyclic group generated by  $g$
- ▶  $N = \text{Card } G$ .

We have a *bijection*

$$\begin{array}{ccc} \exp_g : \mathbb{Z}/N\mathbb{Z} & \rightarrow & G \\ \bar{n} & \mapsto & g^n \end{array}$$

- ▶  $\log_g := \exp_g^{-1}$
- ▶ Compute  $g^n$  from  $n$  : **easy**
- ▶ Compute  $n$  from  $g^n$  : **hard** (discrete logarithm problem)

# DEFINITIONS

Notation :

$$L_N(\alpha) = \exp((c + o(1))(\log N)^\alpha (\log \log N)^{1-\alpha}).$$

Two families of algorithms :

- ▶ The *generic* algorithms (complexity :  $O(\sqrt{N})$ )
- ▶ The *index calculus* algorithms, using group structure

Terminology :

- ▶ *small characteristic* :  $\mathbb{F}_q$  with  $q = p^k$  and  $p \ll q$
- ▶ *quasi-polynomial* complexity :  $\log q^{O(\log \log q)}$

# HISTORICAL BACKGROUND

- ▶ First appearance in [DH76]
- ▶ First sub-exponential algorithm [A79] :  $L(1/2)$
- ▶ Between 1984 and 2006 : algorithms in  $L(1/3)$

And more recently, in finite fields of small characteristic :

- ▶ New algorithm with  $L(1/4)$  complexity [Joux13]
- ▶ *Quasi-polynomial* algorithm [BGJT14]
- ▶ Second quasi-polynomial algorithm [GKZ14]

# OVERVIEW

*Goal* : find  $\log_g(h)$

0. first choose  $F \subset G$  with  $\langle F \rangle = G$
1. find multiplicative relations between elements in  $F$
2. solve the associated linear system
3. express  $h$  as a product of elements in  $F$

The steps 1 and 3 depends on the representation of the finite field, and give different complexities.

## AN EXAMPLE

*Context :*

- ▶  $G = \mathbb{F}_p^\times$  for a prime  $p$  and  $N = |G|$
- ▶  $F = \{f \mid f \leq B, f \text{ prime}\}$  for a chosen integer  $B$

*Step 1 : relations generation*

- ▶ randomly choose  $e \in \mathbb{Z}/N\mathbb{Z}$
- ▶ test if  $g^e$  is  $B$ -smooth
- ▶ if it is the case, it yields a relation in  $G$  :

$$g^e = \prod_{f \in F} f^{e_f}, \text{ où } e_f \in \mathbb{N}$$

that can be written

$$e = \sum_{f \in F} e_f \log_g(f).$$

## AN EXAMPLE

*Step 2 : linear algebra : solve the linear system*

*Step 3 : express  $h$  in function of the elements in  $F$  :*

- ▶ randomly choose  $e \in \mathbb{Z}/N\mathbb{Z}$
- ▶ test if  $hg^e$  is  $B$ -smooth
- ▶ if it is the case, it yields a relation :

$$\log_g(h) = \sum_{f \in F} e_f \log_g(f) - e$$

Depends on  $B$  :

- ▶  $B$  large : easier to find relations
- ▶  $B$  large : need more relations to solve the system

Complexity :  $L(1/2)$



# BARBULESCU, GAUDRY, JOUX AND THOMÉ

## ALGORITHM

*Context :*

- ▶  $\mathbb{K} = \mathbb{F}_{q^{2k}} = \mathbb{F}_{q^2}[X]/(I_X)$  with  $I_X$  irreducible polynomial of degree  $k$  dividing  $h_1 X^q - h_0$  and  $\deg h_i \leq 2$ .
  - ▶ **heuristic** : existence of  $h_i$
- ▶  $F = \{\text{degree one polynomials}\}$

*Idea:* **descent** process:

$$P \rightsquigarrow O(q^2 k) P_j$$

with  $\deg P_j \leq \lceil \frac{1}{2} \deg P \rceil$

- ▶ Complexity :  $(q^2 k)^{O(\log k)}$ .

# THE DESCENT

Based on the equation :

$$X^q Y - X Y^q = Y \prod_{a \in \mathbb{F}_q} (X - aY) = \prod_{\alpha \in \mathbb{P}^1(\mathbb{F}_q)} (X - \alpha Y) \quad (1)$$

- Substitute  $X$  by  $aP + b$  and  $Y$  by  $cP + d$  in (1), with  $a, b, c, d \in \mathbb{F}_{q^2}$ , use  $X^q = \frac{h_0}{h_1} \bmod I_X$  :

$$\frac{1}{h_1^D} \mathcal{L}_{a,b,c,d} = \lambda \prod_{\alpha \in \mathbb{P}^1(\mathbb{F}_q)} (P - \mu_\alpha)$$

where  $\lambda, \mu_\alpha \in \mathbb{F}_{q^2}$  and  $\deg \mathcal{L}_{a,b,c,d} = D \leq 3 \deg P$

# THE DESCENT

- ▶ keep only the equations  $(E_{a,b,c,d})$  where  $\mathcal{L}_{a,b,c,d}$  is  $\left\lceil \frac{\deg P}{2} \right\rceil$ -smooth
- ▶ combine these equations to keep only  $P$  in the right hand side.
- ▶ left hand side : irreducible polynomials of degree  $\leq \left\lceil \frac{\deg P}{2} \right\rceil$ .
- ▶ **heuristics :**
  - ▶ The existence of the combination
  - ▶ The smoothness of the polynomials  $\mathcal{L}_{a,b,c,d}$

# GRANGER, KLEIJUNG AND ZUMBRÄGEL ALGORITHM

*Context:*

- ▶  $\mathbb{K} = \mathbb{F}_{q^k} = \mathbb{F}_q[X]/(I_X)$  with  $I_X$  irreducible polynomial of degree  $k$  dividing  $h_1 X^q - h_0$ .

*Ideas:*

- ▶ **descent** process
- ▶ “on the fly” elimination

# “ON THE FLY” ELIMINATION

*Input:*

- ▶  $Q \in \mathbb{F}_{q^m}[X]$  and  $\deg Q = 2$

*Output:*

- ▶  $Q \rightsquigarrow O(q) Q_i$  with  $\deg Q_i = 1$  and  $Q_i \in \mathbb{F}_{q^m}[X]$

*Ideas:*

- ▶  $P = X^{q+1} + aX^q + bX + c$  splits completely in  $\mathbb{F}_{q^m}[X]$  for  $\approx q^{m-3}$  triples  $(a, b, c)$ .

# “ON THE FLY” ELIMINATION

$$P = \frac{1}{h_1}((X + a)h_0 + (bX + c)h_1) \mod I_X$$

- ▶ if  $Q|(X + a)h_0 + (bX + c)h_1$  (degree 3), we have

$$h_1P = QL \mod I_X$$

where  $L$  is of degree 1 and  $P$  splits completely.

# THE DESCENT

$Q \in \mathbb{F}_q[X]$  irreducible of degree  $2d$ , we have

$$Q = \prod_{i=0}^{d-1} Q_i = \prod_{i=0}^{d-1} Q_0^{[i]}$$

with

- ▶  $Q_i$  irreducible of degree 2 in  $\mathbb{F}_{q^d}[X]$
- ▶  $Q_i = Q_0^{[i]}$  conjugate of  $Q_0$

# THE DESCENT

- ▶ “on the fly” elimination:  $Q_0^{[i]} \rightsquigarrow O(q) P_j^{[i]}$
- ▶  $Q \rightsquigarrow O(q) N_j$  with  $N_j = \prod_{i=0}^{d-1} P_j^{[i]}$

$N_j = R_j^{e_j}$  with  $R_j \in \mathbb{F}_q[X]$  irreducible of degree  $f_j$  and  $e_j f_j = d$

- ▶  $Q \rightsquigarrow O(q) R_j$  with  $\deg R_j \mid d$
- ▶ complexity:  $q^{O(\log q)}$ .