Discrete Logarithm Computation in Hyperelliptic Function Fields

Michael J. Jacobson, Jr.

jacobs@cpsc.ucalgary.ca

UNIVERSITY OF CALGARY

UNCG Summer School in Computational Number Theory 2016: Function Fields

The Discrete Logarithm Problem

Definition

Let G be a finite cyclic group with generator g. Given $h \in G$, the *discrete logarithm problem* (DLP) is to find x mod $|G|$ with $h = g^x$.

Examples:

- For $G = \mathbb{Z}/n\mathbb{Z}$, the DLP is easy (modular inversion)
- For $G = \mathbb{F}_q^*$, number field sieve (q prime) and function field sieve ($q = 2^n$) solve the DLP in subexponential time.

What about $G = Cl^0(F)$?

Is the DLP Hard in $Cl^0(F)$?

For function fields of genus 1 or 2 best-known attacks are generic (except for special cases).

- Thus, as hard as possible — $\sqrt{|Cl^0(F)|} \approx q^{g/2}$ operations.
- Consequence: can use small finite field (eg. elliptic curve $F = \mathbb{F}_q$ with $q \approx 2^{256}$ gives the same security as $\mathbb{F}_p^*$ with $p \approx 2^{3072}$)

Two basic approaches to solving the DLP:

1. solve it in the given group (via generic or specific algorithm),
2. find an explicit isomorphism between the given group and a group with easier DLP (like $\mathbb{Z}/n\mathbb{Z}$).

Generic Methods

Obvious upper bound for group of order N is $O(N)$.

Generic algorithms (like Pollard-rho) yield a slightly better but still exponential run-time.

Pollard-rho: expected run-time $O(\sqrt{N})$ based on birthday paradox. Works for any group.

Also **Pollard-λ (kangaroo)** variant, which makes use of upper and lower bounds on the discrete logarithm x.

Pollard-rho

Given $P, Q \in G$, assume $Q = xP$.

Construct a *random walk* in the group $f : G \to G$:

- Compute $P_0 = a_0 P + b_0 Q$ for $a_0, b_0 \in \mathbb{Z}$.
- Define a partition of G into sets S_j (should be 20 sets) and $M_j = a_j P + b_j Q$ for fixed $a_j, b_j \in \mathbb{Z}$.
- Set $f(R) = R + M_j$ if $R \in S_j$.

For $i \geq 1$, define $P_i = f(P_{i-1})$.

- Note that if $P_{i-1} = u_{i-1}P + v_{i-1}Q \in S_j$ then $P_i = (u_{i-1} + a_j)P + (v_{i-1} + b_j)Q$.
- Can maintain the (u_i, v_i) modulo $|G|$.

Pollard-rho, cont.

Compute and store the P_i and (u_i, v_i) until $P_i = P_j$ for some $i \neq j$. Then:

$$\begin{aligned} u_i P + v_i Q &= u_j P + v_j Q \\ (u_i - u_j)P &= (v_j - v_i)Q = (v_j - v_i)xQ \ . \end{aligned}$$

This implies

$$u_i - u_j \equiv x(v_j - v_i) \pmod{|G|}$$

and if $\gcd(v_j - v_i, |G|) = 1$ we have

$$x \equiv (u_i - u_j)(v_j - v_i)^{-1} \pmod{|G|} \ .$$

Improvements

Low memory variant: only store (P_i, P_{2i}), compute until $P_i = P_{2i}$. Only required to store 2 points on the curve.

Parallelization: m processors yields speed-up of m

Automorphisms: if G has an efficiently computable automorphism of order ℓ, then we can speed up by a factor of $\sqrt{\ell}$. Idea:

- perform random walk on equivalence classes with respect to the automorphism
- effectively reduces size of the group by a factor of ℓ — DLP requires $O(\sqrt{|G|/\ell})$ operations.

If G has such an automorphism, it must be chosen larger to compensate for this attack.

Pohlig-Hellman

Assume that $|G| = \prod_{i=1}^{m} p_i^{e_i}$, p_i distinct primes.

Idea:

- solve the DLP modulo each $p_i^{e_i}$, use CRT to compute x.
- run-time bounded by $O((\log|G|)\sqrt{p_{max}})$ group operations where p_{max} is the largest prime dividing $|G|$.

Point is that $|G|$ should be prime or almost prime to resist this attack.

Pohlig-Hellman: Idea

Let $Q = xP$ and observe that

$$x \equiv z_0 + z_1 p_i + z_2 p_i^2 + \cdots + z_{e_i - 1} p_i^{e_i - 1} \pmod{p_i^{e_i}} \ .$$

Compute z_j given $z_0, \ldots, z_{j-1}$ by solving DLP in a subgroup of order p_i.

To compute z_0 :

- Solve the DLP for $P_0 = (|G|/p_i)P$ and $Q_0 = (|G|/p_i)Q$.
- The order of P_0 and Q_0 in $\langle P \rangle$ is p_i, so

$$Q_0 = z_0 P_0$$

  and we can compute z_0 in $O(\sqrt{p_i})$ operations.

Computing z_1 given z_0

Compute $P_1 = \frac{|G|}{p_i^2}(Q - z_0 P)$ and solve $Q_1 = z_1 P_0$ (order p_i subgroup).

Works because

$$
\begin{aligned}
Q_1 &= \frac{|G|}{p_i^2}\,(Q - z_0 P) \\
&= \frac{|G|}{p_i^2}(x - z_0)P \\
&= (x - z_0)\left(\frac{|G|}{p_i^2}P\right) \\
&= (z_0 + z_1 p_i - z_0)\left(\frac{|G|}{p_i^2}P\right) \\
&= z_1\left(\frac{|G|}{p_i}P\right) \\
&= z_1 P_0 \quad .
\end{aligned}
$$

Computing z_j given $z_0, \ldots, z_{j-1}$

Compute z_j by solving $Q_j = z_j P_0$ (again in a group of order p_i) where

$$\begin{aligned} Q_j &= \frac{|G|}{p_i^{j+1}} \left(Q - z_0 P - z_1 p_i P - z_2 p_i^2 P - \cdots - z_{j-1} p_i^{j-1} P \right) \\ &= \left(x - z_0 - z_1 p_i - z_2 p_i^2 - \cdots - z_{j-1} p_i^{j-1} \right) \frac{|G|}{p_i^{j+1}} P \\ &= (z_j p_i^j) \frac{|G|}{p_i^{j+1}} P \\ &= z_j P_0 \quad . \end{aligned}$$

In total:

- must solve $\sum_{i=1}^{m} e_i \leq \log_2 |G|$ instances of DLPs
- complexity of each is bounded by $O(\sqrt{p_{max}})$ where p_{max} is the largest prime dividing $|G|$.

Notation: Subexponential Function

Notation: $L_x[\alpha, \beta] = O(\exp(\beta(\log x)^{\alpha}(\log\log x)^{1-\alpha}))$.

$$L_x[0, \beta] = O(\exp(\beta \log\log x)) = O(\log^{\beta} x) \rightarrow \text{polynomial time}$$
$$L_x[1, \beta] = O(x^{\beta}) \rightarrow \text{exponential}$$

$0 < \alpha < 1 \rightarrow$ subexponential

Example (factoring N):

- self-initializing quadatic sieve: $L_N[1/2, 1]$ bit operations
- number field sieve: $L_N[1/3, 64/9]$ bit operations

Index Calculus

Define a *factor base* $FB = \{p \in G \mid p \text{ has some distinguishing property}\}$.

- Want FB to generate all of G
- Want a significant portion of G to be efficiently expressed as linear combinations of elements in FB ("smooth" with respect to FB).

Idea:

- Apply Pollard-rho random walk, yielding $P_i = u_i P + v_i Q \in G$.
- Find $m = |FB| + c$ *smooth* $P_j = \sum_{i=1}^{|FB|} e_i p_i$, and record $\vec{v}_j = (e_1, \ldots, e_{|FB|})$.
- Solve $M\vec{z}^T = \vec{0}^T$ where $M = [\vec{v}_1^T \mid \ldots \mid \vec{v}_M^T]$
- Implies $\sum_{j=1}^{m} z_j P_j = 0$: can solve for x after substituting $P_j = u_j P + v_j Q$

Index Calculus: Running Time

Can be faster than generic methods provided that:

1. can find a suitable factor base (high smoothness probability),
2. easy way to represent group elements over the factor base.

Examples:

- Enge/Gaudry (2002): high-genus hyperelliptic curves with $g \gg \log q$): running time $L_N[1/2, \beta]$
- Gaudry/Thomé/Thériault/Diem (2007): $\tilde{O}(q^{2-2/g})$ (faster than Pollard-rho for $g \geq 3$ as $q \leftarrow \infty$)

Doesn't seem to work for genus 1 and 2

Weil and Tate-Lichtenbaum Pairings

If q has order k modulo $|G|$, then the DLP in $Cl^0(\mathbb{F}_q)$ reduces to the DLP in $\mathbb{F}_{q^k}^*$

- Menezes/Okamoto/Vanstone (1991, genus 1), Frey/Rück (1994, genus g hyperelliptic):
- complexity $L_{q^k}[1/3, \beta]$, better than generic if k is small

Eg. Tate-Lichtenbaum pairing: let $|G| = n \mid q - 1$ and E be an elliptic curve over $\mathbb{F}_q$ such that E has a point of order n. Then

$$\tau_n : E(\mathbb{F}_q)[n] \times E(\mathbb{F}_q)/nE(\mathbb{F}_q) \to \mu_n \subseteq \mathbb{F}_{q^k}$$

is a non-degenerate Galois-invariant bilinear pairing.

- Compute DLP x by computing $\tau_n(P, P)$ and $\tau_n(P, Q) = \tau_n(P, P)^x$, and solving DLP in $\mathbb{F}_{q^k}$.

Weil Descent

Suppose E is a non-supersingular curve defined over a binary field $\mathbb{F}_{2^m}$ with $m = dl$.

Frey (1998), Gaudry/Hess/Smart (2002): map the ECDLP to the DLP in a Jacobian variety of a curve of larger genus (usually d) defined over $\mathbb{F}_{2^l}$.

- In some cases, can use subexponential discrete logarithm algorithms to solve DLP.
- J./Menezes/Stein (2001): E.g. for $q = 2^{155} = 2^{5\times 31}$, can solve elliptic curve DLP by reducing to DLP on genus 31 hyperelliptic curve over $\mathbb{F}_{2^5}$.

Other Attacks

Anomalous Curves: If $F = \mathbb{F}_{p^n}$ and $|G| = p$, then $G \cong \mathbb{Z}_p^+$

- DLP easily solved given an efficiently computable isomorphism
- Araki, Satoh, Semaev (1997): polynomial time for genus 1
- Rück (1999): polynomial time for genus g

Summation Polynomials:

- Semaev (2004): express $P = P_1 + \ldots P_k$ algebraically, solve multivariate system of equations to find decompositions of points P
- Ongoing work, but does not (yet?) seem to be efficient in practice

Low-degree $C_{a,b}$ curves

- Enge/Gaudry/Thomé (2011) : $L_{q^g}[1/3, \beta]$ (heuristic) if $n \approx g^\alpha$ and $d \approx g^{1-\alpha}$ for $\alpha \in [1/3, 2/3]$

Summary

DLP believed hard for groups that are:

- large (Pollard-rho),
- prime-order (Pohlig-Hellman)
- $Cl^0(F)$ of genus 1 or 2 hyperelliptic function fields (index-calculus)

Avoid:

- Anomolous curves: defined over $\mathbb{F}_{p^n}$ and $|G| = p$)
- MOV/Frey-Rück: small embedding degree ($q^k \equiv 1 \pmod{|G|}$) for small k), including supersingular curves
- Weil descent: $q = p^m$, m composite
- genus > 2