Discrete Logarithms

Carl Pomerance, Dartmouth College

Suppose that G is a group and $g \in G$ has finite order m. Then for each $t \in \langle g \rangle$ the integers n with $g^n = t$ form a residue class mod m. Denote it by

$$\log_g t.$$

The discrete logarithm problem is the computational task of finding a representative of this residue class; that is, finding an integer n with $g^n = t$.

Finding a discrete logarithm can be *very* easy. For example, say $G = \mathbb{Z}/m\mathbb{Z}$ and $g = 1$. More specifically, say $m = 100$ and $t = 17$. Then $\log_g t = 17$ (or more precisely 17 mod 100).

Lets make it harder: take g as some other generator of $\mathbb{Z}/m\mathbb{Z}$. But then computing $\log_g t$ is really solving the congruence

$$ng \equiv t \bmod m$$

for n, which we've known how to do easily essentially since Euclid.

The cyclic group of order m:

What does this title mean, especially the key word "The"?

Take $G_1 = \mathbb{Z}/100\mathbb{Z}$ and $G_2 = (\mathbb{Z}/101\mathbb{Z})^\times$. Both are cyclic groups of order 100. Both are generated by 3. And 17 is in both groups.

So, there are two versions of computing $\log_3 17$, one in G_1 and one in G_2.

In G_1, we are solving $3n \equiv 17 \bmod 100$. The inverse of 3 is 67, so $n \equiv 17 \cdot 67 \equiv 39 \bmod 100$.

In G_2, we are solving $3^n \equiv 17 \bmod 101$. And this seems much harder.

The moral: when someone talks about *the* cyclic group of a given order, they are not concerned with computational issues.

Well, how can we solve $3^n \equiv 17 \bmod 101$?

Clearly, one way is trial and error, where we compute each power of 3 mod 101 till we find our target 17. The complexity of doing this in a cyclic group of order m is $O(m)$ (and this upper bound stands as a lower bound as well for most target elements t).

Note that the group order is 100, which is $2^2 \cdot 5^2$. Can we reduce it to smaller problems?

In solving $3^n \equiv 17 \bmod 101$, we might ask the gentler question: is n even?

That is, is 17 a square? That is, what is $\left(\frac{17}{101}\right)$?

By the reciprocity law (for Jacobi symbols),

$$\left(\frac{17}{101}\right) = \left(\frac{101}{17}\right) = \left(\frac{2}{17}\right) = 1,$$

so yes, $\log_3 17$ is even.

Even without Jacobi symbols, we could have answered this by computing $17^{50} \bmod 101$. It is 1 if and only if 17 is a square if and only if $\log_3 17$ is even. (Recall: powering is easy via repeated squaring.)

Can we also easily see whether $\log_3 17$ is 0 or 2 mod 4? Yes, compute 17^{25} mod 101. If it is 1, then $\log_3 17$ is 0 mod 4 and if it is -1, then 2 mod 4.

If its supposed to be easy, lets try it: In binary, 25 is 11001. So we consider the sequence 1, 11, 110, 1100, 11001 as follows:

$$17^1 \equiv 17, \quad 17^2 \equiv 87 \equiv -14, \quad 17^3 \equiv -36$$
$$17^6 \equiv -17, \quad 17^{12} \equiv -14, \quad 17^{24} \equiv -6$$
$$17^{25} \equiv -1$$

Thus, $\log_3 17$ is 2 mod 4.

And what about mod 5 and mod 25?

From the prior calculations, if we were observant, we noticed that $17^5 \equiv -1 \bmod 101$. Thus, $17^{20} \equiv 1 \bmod 101$, so that $\log_3 17$ is 0 mod 5.

So, $\log_3 17$ is one of 5 possibilities: 10, 30, 50, 70, 90. Now $3^5 \equiv 41 \bmod 101$, so $3^{10} \equiv -36 \bmod 101$.

Thus, 10 is out. We have $3^{20} \equiv -17 \bmod 101$, so we see that the answer is 70, since $3^{50} \equiv -1 \bmod 101$ (true for any cyclic generator in an even order group).

There are two thoughts/questions suggested by these calculations:

- Are there strategies of reducing a bigger discrete log problem to a smaller one?

- Are there special strategies for the family of groups $(\mathbb{Z}/p\mathbb{Z})^\times$, where p is prime?

If g has order m, $t \in \langle g \rangle$, and $d \mid m$, then write

$$\log_g t = n = n_1 d + n_2, \quad 0 \leq n_2 < d.$$

If we can find n_1, n_2, we can find n. Note that

$$t = g^n = g^{n_1 d + n_2},$$

so that

$$t^{m/d} = g^{n_1 m + n_2 m/d} = g^{n_2 m/d} = (g^{m/d})^{n_2}.$$

Thus, n_2 is the solution of a dl problem in the group $\langle g^{m/d} \rangle$ of order d. And if we solve it, then

$$(g^d)^{n_1} = t g^{-n_2},$$

so n_1 is a solution of a dl problem in the group $\langle g^d \rangle$ of order m/d.

This kind of reduction is attributed to Pohlig and Hellman and because of it, cryptographers prefer groups of large prime order, or of an order divisible by a large prime.

Cryptographers?

The Diffie–Hellman key-exchange protocol:

Say we have a cyclic group generated by g, which everyone knows. Alice has a secret integer a and "publishes" g^a. Similarly, Bob has a secret integer b and publishes g^b.

Alice and Bob want to set up a secure session with a secret key that only they know, yet they want to set this up over a pubic line. Here's how they do it: Alice takes Bob's group element g^b and raises it to her secret exponent a, getting $(g^b)^a = g^{ab}$. Bob arrives at the same group element via a different method, namely $(g^a)^b = g^{ab}$.

Eve (an eavesdropper) knows something's afoot and knows g^a and g^b, but apparently cannot easily compute g^{ab} without finding either a or b, that is without solving the dl problem.

The second question: Can we exploit any special structure in $(\mathbb{Z}/p\mathbb{Z})^\times$ to compute dl's there? Yes, we can.

Use the following facts about this group: It is a homomorphic image of semigroup $\mathbb{Z}$ under times. A factorization of an element of $\mathbb{Z}$ coprime to p then maps to a "relation" among group elements.

For example, in $(\mathbb{Z}/101\mathbb{Z})^\times$, we have

$$5^3 \equiv 125 \equiv 24 \equiv 2^3 \cdot 3 \bmod 101, \quad 2^7 \equiv 128 \equiv 27 \equiv 3^3 \bmod 101.$$

Thus,

$$3\log_3 5 \equiv 3\log_3 2 + 1 \bmod 100, \quad 7\log_3 2 \equiv 3 \bmod 100,$$

from which it may be deduced that

$$\log_3 2 \equiv 43 \cdot 3 \equiv 29 \bmod 100, \quad \log_3 5 \equiv 96 \bmod 100.$$

For example, just using $\log_3 2 \equiv 29 \bmod 100$ and using $17 \cdot 6 \equiv 1 \bmod 101$, we have

$$\log_3 17 + \log_3 2 + 1 \equiv 0 \bmod 100,$$

so

$$\log_3 17 \equiv 70 \bmod 100.$$

This kind of thing can be formalized into the "index calculus" algorithm:

- Choose random numbers r, each time compute g^r mod p, and save any that happen to factor into small primes.

- After enough of these have been saved, we can use linear algebra over the ring $\mathbb{Z}/(p-1)\mathbb{Z}$ to solve for the dl's of the small primes.

- Assuming this is accomplished, again choose random numbers r until one is found where $g^r t$ factors into small primes.

If

$$g^r t \equiv p_1^{a_1} \dots p_k^{a_k} \bmod p,$$

then using the pre-computed numbers $\log_g p_i$, we get

$$\log_g t \equiv -r + a_1 \log_g p_1 + \dots + a_k \log_g p_k \mod (p-1).$$

This kind of idea can be copied for any group which is a homomorphic image of a multiplicative structure where we have factorization into "small" elements. (The set of small elements used is called the "factor base".)

So, for example, the index calculus method can be used in many cases for finding dl's in $\mathbb{F}_q^\times$. Eg, say $q = p^a$, with p prime and a large. We can view $\mathbb{F}_q$ as $\mathbb{F}_p[x]/(f(x))$ where f is irreducible of degree a. And $\mathbb{F}_p[x]$ is a Euclidean domain.

If a is small, we can view $\mathbb{F}_q$ as $\mathcal{O}_K/(p)$ where K is an algebraic number field of degree a over $\mathbb{Q}$ in which p is inert. Even though $\mathcal{O}_K$ may not be a Euclidean domain, and perhaps not even a PID, we do have unique factorization of ideals and we do have a sense of size afforded by the norm. Problems remain, but in many cases the index calculus method is useful.

And there are very important improved versions that employ ideas from the number field sieve for factoring integers.

Thus, cryptographers tend to shy away from the groups $\mathbb{F}_q^\times$.

What generic algorithms might exist other than listing all of the powers of g?

Well, there's "baby steps, giant steps" (known in the CS world as "meet in the middle"):

- Have g of order m and $t \in \langle g \rangle$. Find $k = \lceil \sqrt{m} \rceil$ and g^{-1}.
- Compute the baby steps $tg^0, tg^{-1}, \ldots, tg^{-(k-1)}$ and the giant steps $g^0, g^k, \ldots, g^{(k-1)k}$.
- Sort both lists and find a coincidence between them, say $tg^{-i} = g^{jk}$. Then $t = g^{i+jk}$ and $\log_g t = i + jk$.

Why must there be a coincidence between the two lists?

Well, since $t \in \langle g \rangle$, there is some $n \in [0, m-1]$ with $g^n = t$. Write n in base k, so that since $k^2 > m-1$, we have $n = i + jk$ for some integers $i, j \in [0, k-1]$. And thus, $tg^{-i} = g^{jk}$.

The algorithm presupposes labels for group elements that allows them to be sorted. Sorting can be done in time not much larger than the size of the set to be sorted, and after this, finding the match between the two parts takes only $O(k) = O(\sqrt{m}\,)$ comparisons.

In all, baby steps, giant steps takes $O(\sqrt{m}\log m)$ group operations. It is essentially a universal algorithm, so cryptographers can't avoid it.

A downside of baby steps, giant steps is that it is not so easy to distribute the work to many computers. Another algorithm due to Pollard can be distributed and is what's used in practice to benchmark cryptosystems. It's interesting that Pollard's method is heuristic while baby steps, giant steps is rigorous. Of course, if an answer is found, it is easily checked, so the heuristic part deals with whether the algorithm will terminate within the supposed time bound (which is also about $\sqrt{m}$).

So, can we find a family of convenient groups for which the only dl algorithms take exponential time?

It's hard to prove that it is so, but many people feel that elliptic curve groups over finite fields fit this bill.

What are they? If F is a field (of characteristic not 2 nor 3) and a, b are elements with $4a^3 - 27b^2 \neq 0$, then the solutions to the equation $y^2z = x^3 + axz^2 + bz^3$ in F^3 (viewed projectively) can be endowed with a natural commutative group operation. The identity is the projective point $(0 : 1 : 0)$ and all other points have $z \neq 0$, so may be viewed as solutions to $y^2 = x^3 + ax + b$ in F^2.

The group operation will be illustrated on the board ... .

The group is denoted $E(F) = E_{a,b}(F)$. We can do it as well in characteristics 2 and 3, but the formulas work out a little differently.

What can we say about the order of the group $E(F)$? If $F = \mathbb{Q}$, then it is possible for the group to be finite (but then always of order at most 12) and also possible for it to be infinite. If $F = \mathbb{C}$, then there is a natural way to make the group isomorphic to $\mathbb{C}/\Lambda$ where Λ is the $\mathbb{Z}$ module generated by a basis of $\mathbb{C}$ over $\mathbb{R}$.

But what about the situation with $E(\mathbb{F}_q)$? Here we have the theorem of Hasse–Weil: $\#E(\mathbb{F}_q)$ is within $2\sqrt{q}$ of $q+1$, that is, $\#E(\mathbb{F}_q) \in [(\sqrt{q}-1)^2, (\sqrt{q}+1)^2]$.

Further, by a theorem of Deuring, each number in the interval is the order of some elliptic curve over $\mathbb{F}_q$.

Thus, "cryptographically interesting" elliptic curves over $\mathbb{F}_q$ are those with $\#E(\mathbb{F}_q)$ a prime number in the interval $[(\sqrt{q}-1)^2, (\sqrt{q}+1)^2]$, or with the order nearly prime, say twice a prime.

An aside: We believe that for each prime power q there are about $4\sqrt{q}/\log q$ primes in the above interval, but we don't have a proof that there is even one prime. For cryptography, it doesn't matter, since if you find one you find one, and it does not matter that analytic number theorists are not smart enough to prove that it must have existed.

Just like $\mathbb{F}_q$, elliptic curves have lots of structure. Can any of this be exploited to help with the dl problem?

The Weil pairing "attack" (also known as MOV):

A. Menezes, T. Okamoto, and S. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, IEEE Trans. Inform. Theory **39** (1993), 1639–1646.

Weil proved that for each natural number n there is a map e_n from $E[n] \times E[n]$ (ordered pairs of n-torsion points on $E(\overline{\mathbb{F}_q})$) to $\overline{\mathbb{F}_q}$ that has various nice properties (alternating, bilinear, etc.). In addition, Miller made it algorithmic, so it can be computed at a given pair of points in the domain quickly. This is all quite interesting in connection with the dl problem since if $E[n] \subset E(\mathbb{F}_{q^k})$ for some k, then the range of e_n is in $\mathbb{F}_{q^k}$, and via e_n one can reduce a dl problem in a cyclic subgroup of $E(\mathbb{F}_q)$ of order n to a dl problem in $\mathbb{F}_{q^k}^{\times}$. So, if k is small, we're in business.

It is shown that the only cases with $k \leq 6$ occur for a few families of curves, with

$$\#E(\mathbb{F}_q) = q + 1 - t,$$

and t, which is always at most $2\sqrt{q}$ in absolute value, satisfies $t^2 = jq$ for $j = 0, 1, 2, 3, 4$. In the cases where $t \neq 0$, the group is far from cyclic, it is of the form $C_k \times C_k$. So, these would not have been used for cryptography in any event. In the cases when $t = 0$, the group is either C_{q+1} or $C_{(q+1)/2} \times C_2$. So, these so-called supersingular cases are off the table for cryptographic purposes.

In addition, any curve where the multiplicative order of q modulo $q + 1 - t$ is small is also vulnerable.

There are more complicated attacks based on "Tate pairing" and on "Tate–Lichtenbaum" pairing.

G. Frey and H.-G. Rück, A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves, Math. Comp. **62** (1994), 865–874.

G. Frey, M. Müller, and H.-G. Rück, The Tate pairing and discrete logarithm applied to elliptic curve crypotosystems, IEEE Trans. Inform. Theory **45** (1999), 1717–1719.

G. Frey, Applications of arithmetical geometry to cryptographic constructions, Finite fields and applications (Augsburg, 1999), 128–161, Springer, Berlin, 2001.

Weil descent:

This idea is based on two concepts:

Weil showed that there is an explicit homomorphism from $E(\mathbb{F}_q)$ to the Jacobian variety of a certain hyperelliptic curve of genus g.

Adleman, DeMarrais, and Huang showed that there is an index-calculus attack on the dl problem for Jacobian varieties of hyperelliptic curves of genus greater than 1.

Putting this together, Gaudry, Hess, and Smart showed that certain elliptic curves (with characteristic 2) are vulnerable.

P. Gaudry, F. Hess, and N. Smart, Constructive and destructive facets of Weil descent on elliptic curves, J. Cryptology **15** (2002), 19–46.

F. Hess, Generalising the GHS attack on the elliptic curve discrete logarithm problem, LMS J. Comput. Math. **7** (2004), 167–192.

So, are elliptic curve cryptosystems sunk?

For the Weil descent attacks to be successful, the genus of the curve found should not be too large. If $\mathbb{F}_q$ is a finite field of characteristic 2 and we have an elliptic curve $E(\mathbb{F}_{q^n})$, then the Weil descent involves a curve over $\mathbb{F}_q$ of genus g where g depends on the given elliptic curve. For the attack to be useful, one needs at least $g \geq n$, but g not too large, say $O(n^2 \log q / \log(n \log q))$.

In a new paper still being written
(K. Karabina, A. Menezes, C. Pomerance, and I. E. Shparlinski, On the asymptotic effectiveness of Weil descent attacks)
we study the question statistically and ask what happens for a typical elliptic curve. We show that in fact the genus g grows exponentially with n and also obtain somewhat larger exponential upper bounds. Something like
$2^{(1/2+o(1))n} \le g \le 2^{(2/3+o(1))n}$ almost always.

Whew! We saved the system.

By the way, one of the ingredients in the argument is to study the factorization of $x^n - 1$ in $\mathbb{F}_2[x]$.