A few more index calculus algorithms for the Elliptic Curve Discrete Logarithm Problem

Gary McGuire[*] Daniela Mueller[†]

School of Mathematics and Statistics, University College Dublin, Ireland

Abstract

The introduction of summation polynomials by Semaev has opened up new avenues of investigation in index calculus type algorithms for the ECDLP, and several recent papers have explored their use. Most of these papers use Gröbner basis computations at some point. We propose an algorithm to solve the ECDLP using summation polynomials that does not involve Gröbner basis computations. Our algorithm makes use of a technique for fast evaluation of the summation polynomials.

Let E be an elliptic curve over a finite field $\mathbb{F}_q$, where q is a prime power. In practice, q is often a prime number or a large power of 2. Let P and Q be points on E. The Elliptic Curve Discrete Logarithm Problem (ECDLP) is finding an integer l (if it exists) such that $Q = lP$. The integer l is called the discrete logarithm of Q to base P. The ECDLP is a hard problem that underlies many cryptographic schemes and is thus an area of active research. The introduction of summation polynomials by [Sem04] has led to algorithms that resemble the index calculus algorithm of the DLP over finite fields.

Definition: [Sem04] Let E be an elliptic curve over a field K. For $n \geq 2$, we define the summation polynomial $S_n = S_n(X_1, X_2, \ldots, X_n)$ of E by the following property. Let $x_1, x_2, \ldots, x_n \in \overline{K}$, then $S_n(x_1, x_2, \ldots, x_n) = 0$ if and only if there exist $y_1, y_2, \ldots, y_n \in \overline{K}$ such that $(x_i, y_i) \in E(\overline{K})$ and $(x_1, y_1) + (x_2, y_2) + \ldots + (x_n, y_n) = \mathcal{O}$, where $\mathcal{O}$ is the identity element of E.

Most papers have focused on elliptic curves over an extension field $\mathbb{F}_{q^n}$, and use subfields in the algorithm, see e.g. [Gau09] or [FHJ⁺14]. The case of elliptic curves over prime order fields seems to be much harder to tackle. Our algorithm is aimed at prime order fields, although it is valid for any finite field. A recent article [APS17] has shown how to simplify the index calculus algorithm using summation polynomials to avoid the linear algebra step and reduce the number of Gröbner basis computations. Unlike previous algorithms, they choose a random factor base. Our algorithm is based on theirs but does not involve a Gröbner basis computation. This leads to a significant speedup over other prime field algorithms. It uses a method for fast evaluation of summation polynomials, which we have developed for this project.

We propose the following algorithm to solve the ECDLP.

Algorithm:

Input: elliptic curve E over $\mathbb{F}_p$, points P and Q on E, integers m, s, summation polynomial S_m

Output: $\log_P(Q)$

1. Compute random integers $a_1, ..., a_s,\ b_1, ..., b_s$. The factor base $\mathcal{F}$ consists of all points $\{a_1 P + b_1 Q, ..., a_s P + b_s Q\}$. The corresponding set containing only the x-coordinates of the factor base points is denoted $V = \{x | (x, y) \in \mathcal{F}\}$.
2. Choose $\{x_1, \ldots, x_m\}$ a multiset of size m with each $x_i \in V$ and check if $S_m(x_1, \ldots, x_m) = 0$. If not, repeat with another multiset. If S_m is non-zero for all multisets of size m, go back to step 1.
3. If $S_m(x_1, \ldots, x_m) = 0$ for some $\{x_1, \ldots, x_m\}$, then there exist y_i such that $(x_1, y_1) + \ldots + (x_m, y_m) = \mathcal{O}$ where either (x_i, y_i) or $-(x_i, y_i)$ are in $\mathcal{F}$. Substituting each $\pm(x_i, y_i)$ with the corresponding

[*] Research supported by Science Foundation Ireland Grant 13/IA/1914.

[†] Research supported by a Postgraduate Government of Ireland Scholarship from the Irish Research Council.

$\pm(a_iP + b_iQ)$, we get a relation of the form $\sum_{i=1}^{m} \pm a_iP + \sum_{i=1}^{m} \pm b_iQ = \mathcal{O}$ and can solve for the discrete logarithm of Q, provided $\sum_{i=1}^{m} \pm b_i$ is invertible modulo the order of E.

Steps 1 and 3 agree with the algorithm proposed in [APS17]. In step 2 we make use of a fast evaluation technique for summation polynomials.

In the following, we outline our complexity analysis for this algorithm. Full details can be found in our article [MM17].

Lemma: The probability of obtaining a relation of length m with each point coming from a different partition of the factor base of size $\frac{s}{m}$ is $\frac{2^{m-1}s^m}{p \cdot m^m}$.

We would like the probability of finding a relation in the factor base to be close to 1, i.e. we want $\frac{2^{m-1}s^m}{m^m} \approx p$, so we should choose the factor base size s accordingly.

Lemma: The complexity of computing a factor base of size s is $O(s \log^3 p)$.

Proposition: Evaluating S_m at a point $(x_1, \ldots, x_m)$ can be done in $O(\log^2 p)$ steps for $m \ll p$.

Sketch of proof: It follows directly from [Sem04] that S_3 can be evaluated with at most 8 multiplications and 11 additions. For larger m, we make use of the fact that $Res_X(f(a,X), g(a,X)) = Res_X(f,g)(a)$ whenever the leading coefficients are non-zero. We can write $S_4(x_1,x_2,x_3,x_4) = a_2(b_0(a_2b_0 - 2a_0b_2 - a_1b_1) + a_0b_1^2) + b_2(a_1(a_1b_0 - a_0b_1) + a_0^2b_2)$, where $S_3(x_1,x_2,X) = a_2X^2 + a_1X + a_0$ and $S_3(x_3,x_4,X) = b_2X^2 + b_1X + b_0$, and thus can evaluate it with at most 21 multiplications and 24 additions. For $m \geq 5$, $S_m(x_1,\ldots,x_m) = Res_X(S_3(x_1,x_2,X), S_{m-1}(x_3,\ldots,x_m,X))$. We replace the Sylvester matrix in the resultant computation by six smaller matrices, and evaluate S_3 and S_{m-1} recursively. We arrive at a complexity of $(17m-47)O(\log^2 p) + (17m-44)O(\log p) + 6(O(2^{3(m-3)}) + O(2^{3(m-4)}) + \ldots + O(2^{3\cdot 2}))$.

Theorem: The complexity of our algorithm is $O(p \log^2 p)$ for $m \ll s$, $m \ll p$ and $\frac{s^{m-1}}{m!} \geq \log p$.

The described evaluation technique is significantly faster than first computing S_m and then evaluating it, and allows us to evaluate S_9 and S_{10}, although nobody has actually computed them yet as far as we know. We have run extensive experiments in Magma V2.21-6 [BCP97], which agree with our complexity analysis and show that our algorithm is faster than other prime field algorithms.

References

[APS17] Alessandro Amadori, Federico Pintore, and Massimiliano Sala. On the discrete logarithm problem for prime-field elliptic curves. Cryptology ePrint Archive, Report 2017/609, 2017. https://eprint.iacr.org/2017/609.

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

[FHJ$^+$14] Jean-Charles Faugère, Louise Huot, Antoine Joux, Guénaël Renault, and Vanessa Vitse. Symmetrized summation polynomials: Using small order torsion points to speed up elliptic curve index calculus. *Advances in Cryptology – EUROCRYPT 2014 Lecture Notes in Computer Science*, 8441:40–57, 2014.

[Gau09] Pierrick Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem. *Journal of Symbolic Computation*, 44(12):1690–1702, 2009.

[MM17] Gary McGuire and Daniela Mueller. A few more index calculus algorithms for the elliptic curve discrete logarithm problem. Cryptology ePrint Archive, Report 2017/1262, 2017. http://eprint.iacr.org/2017/1262.

[Sem04] Igor Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. Cryptology ePrint Archive, Report 2004/031, 2004. http://eprint.iacr.org/2004/031.