Computing L-polynomials of Non-Hyperelliptic Genus 4 and 5 curves

Raminder Singh Ruprai, Steven D. Galbraith
Information Security Group, Royal Holloway University of London, Egham, Surrey, UK

Background

Let C be a non-singular, projective, non-hyperelliptic curve over a finite field $\mathbb{F}_q$ of genus g. Let $\tilde{C}$ be an affine, plane model of degree d of the curve C. When dealing with curves of genus 4 and 5, $\tilde{C}$ can have singular points. This affects parts of the algorithm as we shall see. We are interested in using the cardinality of the group of $\mathbb{F}_q$-rational points on the Jacobian variety of C, $J_C(\mathbb{F}_q)$, to help find the L-polynomial of the given curve.

For a given positive integer k, let N_k be the number of $\mathbb{F}_{q^k}$-rational points on C. The Zeta function of C is then the formal power series

$$Z(t) = \exp\left(\sum_{k=1}^{\infty} \frac{N_k t^k}{k}\right) = \frac{L(t)}{(1-t)(1-qt)} \qquad (1)$$

where the L-polynomial $L(t) = \sum_{i=0}^{2g} a_i t^i$. The well-known theorem of Weil [5] provides us with many of the facts that we require.

Theorem 1 (Weil)**.** *Let C be a genus g curve defined over $\mathbb{F}_q$. For $k \geq 1$ we let $J_C(\mathbb{F}_{q^k})$ denote the group of $\mathbb{F}_{q^k}$-rational points on the Jacobian variety of C.*

1. *The L-polynomial has integer coefficients satisfying $a_0 = 1$ and $a_{2g-i} = q^{g-i} a_i$, for $0 \leq i < g$.*
2. *$L(t) = \prod_{i=1}^{2g}(1-\alpha_i t)$, with $|\alpha_i| = \sqrt{q}$.*
3. *$|a_i| \leq \binom{2g}{i} q^{i/2}$.*
4. *$N_k = q^k + 1 - \sum_{i=1}^{2g} \alpha_i^k$.*
5. *$\#J_C(\mathbb{F}_{q^k}) = \prod_{j=1}^{k} L(\omega_k^j) = \prod_{i=1}^{2g}(1-\alpha_i^k)$, where ω_k is a principal k^{th} root of unity.*

In particular $\#J_C(\mathbb{F}_q) = L(1)$ so by applying part (2) of Theorem 1 we obtain the Weil interval

$$(\sqrt{q}-1)^{2g} \leq \#J_C(\mathbb{F}_q) \leq (\sqrt{q}+1)^{2g}. \qquad (2)$$

A proof of Theorem 1 can be found in chapters 8 and 10 of [3].

Currently there are no fast algorithms which can compute the L-polynomial of a general non-hyperelliptic curve over $\mathbb{F}_q$ where q is a large prime (as opposed to a large power of a small prime). The first basic method would be to use a *Baby-Step Giant-Step* (BSGS) approach to compute $\#J_C(\mathbb{F}_q), \#J_C(\mathbb{F}_{q^2}), \ldots, \#J_C(\mathbb{F}_{q^g})$ and then by simple algebra solve for the coefficients of the L-polynomial. This would have running time of $\tilde{O}(q^{g^2/2})$. Another basic method would be to just count the $\mathbb{F}_{q^k}$-rational points on C (i.e. N_k) for $1 \leq k \leq g$ and from point (4) of Theorem 1 we can compute the coefficients of the L-polynomial. This has running time $\tilde{O}(q^g)$ which is faster than the first method for genus $g \geq 2$. Elkies in [2] describes a method which has a similar second stage to the algorithm presented here. In genus 4 the L-polynomial of a curve is recovered in time $\tilde{O}(q^{3/2})$ and for genus 5 in time $\tilde{O}(q^2)$. We improve on this for curves of low degree.

Main Theorem

Theorem 2. *For a genus 4 or 5 non-hyperelliptic curve there exists an algorithm to compute all the coefficients of the L-polynomial of the curve in time $\tilde{O}(q^2)$ in the worst case but in time $\tilde{O}(q^{4/3})$ when the plane model is of degree 5.*

Stage 1 is the dominant part of the algorithm and determines the best and worst case running time.

Algorithm: Stage 1

The first stage of the algorithm is to compute $\#J_C(\mathbb{F}_q) = L(1)$. We use a variant of Diem's discrete logarithm algorithm in [1] to do this.

Input: An affine, plane model $\tilde{C}$ of degree d of a curve C. Denote $\tilde{C}_{ns}$ to be the non-singular part of $\tilde{C}$. Also define a fixed point $P_0 \in \tilde{C}_{ns}(\mathbb{F}_q)$ (used to represent the elements in $J_C(\mathbb{F}_q)$). Let $r < 1$ be a positive rational number defined in [1].
Output: A positive integer equal to $\#J_C(\mathbb{F}_q)$ and therefore $L(1)$.

1. Calculate a random divisor D which represents an element of $J_C(\mathbb{F}_q)$ where D splits completely over points in $\tilde{C}_{ns}(\mathbb{F}_q)$.
2. Fix a 'factor-base' $\mathcal{F} \subset \tilde{C}_{ns}(\mathbb{F}_q)$ such that $\#\mathcal{F} = \lceil q^r \rceil$ and $\mathcal{F}$ contains the points in D. (If no such set exists, output "failure" and terminate.)
3. From step 1 store the relation as a row in a matrix M. Each column of M represents the points in $\mathcal{F}$ so each row will represent the multiplicities of the corresponding points.
4. Go through all pairs of points in $\mathcal{F}$, $\mathcal{F}_i$ and $\mathcal{F}_j$ where $1 \leq i, j \leq \#\mathcal{F}$ and form a line through each pair of points. Compute the intersection of the line and $\tilde{C}$, as described in [1]. If all the intersection points are in $\tilde{C}_{ns}(\mathbb{F}_q)$ add a row to M as described in the previous step.
5. Calculate a number of random vectors $v_k \in \ker(M^t)$ for $0 < k < \dim(\ker(M^t))$
6. Calculate the greatest common divisor of $\{v_{1_1}, v_{2_1}, \ldots, v_{k_1}\}$. This is a multiple of the order of D. From this the exact order of D can be obtained and using the Weil interval, (2), we have a set of possibilities for L(1) which can be checked by simple trial and error.

N.B. When implementing step 6 of the algorithm more often than not the greatest common divisor of $\{v_{1_1}, v_{2_1}, \ldots, v_{k_1}\}$ is equal to $L(1)$ rather than being a multiple of L(1).
The overall running time is $\tilde{O}(q^2)$ for a curve of any degree but the running time is $\tilde{O}(q^{4/3})$ when we have a degree 5, genus 4 or 5 curve. We can calculate N_1 the number of $\mathbb{F}_q$-rational points on C using the naive method in time $\tilde{O}(q)$. (As there may be singular points on $\tilde{C}$ and points at infinity extra care will need to be taken when calculating N_1.) From N_1 we have the first non-trivial coefficient of the L-polynomial

$$a_1 = q + 1 - N_1.$$

Algorithm: Stage 2

We now know $L(1)$ and a_1. Taking the genus 4 case first, from Theorem 1 we can rewrite $L(-1)$ in two ways

$$L(-1) = 2(1+q^4) - L(1) + a_2(1+q^2) + a_4, \qquad (3)$$
$$L(-1) = L(1) - 2a_1(1+q^3) - 2a_3(1+q). \qquad (4)$$

Using a similar technique to Sutherland in [4] we let D_2 be a random divisor representing an element of $J_C(\mathbb{F}_{q^2})$ and let D_{2L} be the $L(1)^{\text{th}}$ power of D_2 (i.e. $D_{2L} = L(1) * D_2$). We can find the coefficient a_3 using a BSGS strategy based on the condition that $L(-1) * D_{2L} = 0$ in time $\tilde{O}(q^{3/4})$. This still leaves coefficients a_2 and a_4. However as we now know $L(-1)$, using equation (3) we can rewrite a_4 in terms of a_2. Working in $J_C(\mathbb{F}_{q^3})$ which has order $\#J_C(\mathbb{F}_{q^3}) = L(1) \cdot L(\omega_3) \cdot L(\omega_3^2)$ we can use BSGS again to find a_2 and therefore a_4 in time $\tilde{O}(q^{1/2})$. Both of these running times are less than the running time in Stage 1 so Stage 1 dominates and Theorem 2 is proved for genus 4.

For genus 5 we apply a very similar method but finding $L(-1)$ requires more work. In this case we have

$$a_2 = \frac{L(1) - (1+q^5) - a_1(1+q^4) - a_3(1+q^2) - a_4(1+q) - a_5}{1+q^3}, \qquad (5)$$
$$L(-1) = 2(1+q^5) - L(1) + 2a_2(1+q^3) + 2a_4(1+q), \qquad (6)$$
$$L(-1) = L(1) - 2a_1(1+q^4) - 2a_3(1+q^2) - 2a_5. \qquad (7)$$

Using equation (5) and ideas of Sutherland [4], we get a restricted bound for a_2. We can put this into equation (6) which gives

$$L(-1) = K + 2x(1+q^3) + 2a_4(1+q), \qquad (8)$$

where K is a known constant, $0 \leq x \leq 240q^{1/2}$ *(approximately)* and the bounds on a_4 are given in part (3) of theorem 1. We can then use BSGS to find the correct value of $L(-1)$ and therefore the values of a_2 and a_4 in time $\tilde{O}(q^{5/4})$. Again we have to do more work to find a_3 and a_5 by doing BSGS in $J_C(\mathbb{F}_{q^3})$ which takes time $\tilde{O}(q^{3/4})$. Both of these times are dominated by Stage 1 and therefore Theorem 2 is proved for genus 5.

References

[1] Claus Diem. Index calculus in class groups of plane curves of small degree. In *Proceedings of Algorithm Number Theory Symposium - ANTS VII*, volume 4076 of *Springer-Verlag LNCS*, pages 543–557. Springer-Verlag, 2006.

[2] Noam D. Elkies. Elliptic and modular curves over finite fields and related computational issues. In D.A. Buell and J.T. Teitelbaum, editors, *Computational Perspectives on Number Theory: Proceedings of a Conference in Honor of A.O.L. Atkin*, volume 7 of *Studies in Advanced Mathematics*, pages 21–76. American Mathematical Society, International Press, 1998.

[3] Dino Lorenzini. *An invitation to arithmetic geometry*, volume 9 of *Graduate Studies in Mathematics*. 1996.

[4] Andrew V. Sutherland. A generic approach to searching for jacobians. *to appear in Mathematics of Computation*, 2007.

[5] André Weil. Numbers of solutions of equations in finite fields. *Bulletin of the American Mathematical Society*, 55:497–508, 1949.