The security of Groups of Unknown Order based on Jacobians of Hyperelliptic Curves

Jonathan Lee*

Microsoft Research

February 28, 2020

Abstract

Recent work using groups of unknown order to construct verifiable delay functions, polynomial commitment schemes and non interactive zero knowledge proofs have provoked fresh interest in the construction of efficient cryptographic groups of unknown order. It has been suggested that the Jacobian of hyperelliptic curves of genus 3 could be suitable for this purpose. Regrettably, efficient algorithms to compute the order of the Jacobian of a hyperelliptic curve are known. Concretely, it is unclear whether these groups are competitive with RSA groups or class groups at or above the 128 bit security level.

1 Background

There has been recent burst of interest in cryptographic groups of unknown order, stemming in part from their use in the construction of accumulators, verifiable delay functions, integer and polynomial commitments [8, 9, 11, 16, 17, 22]. Long standing examples of groups of unknown order are RSA groups $(\mathbb{Z}/N\mathbb{Z})^\times$ for $N = pq$ semiprime, and the ideal class group of the imaginary quadratic fields $\mathbb{Q}(\sqrt{-D})$ for $D > 0$. We refer the reader to [6, 7, 10, 14] for details of these groups. In both cases there are index-calculus methods [7, 20, 21] which permit the computation of the order $n \sim N, \sqrt{D}$ of these groups in time $L_n(a, b) = \exp((b + o(1)) \log^a n (\log \log n)^{1-a})$ for $a = 1/3, 1/2$ respectively and $b = O(1)$. As a result, current estimates [5, 7] suggest $\log N \sim 3072$ or $\log D \sim 1827$ for the 128 bit security level.

Recently, Dobson and Galbraith [17] suggest that Jacobians of hyperelliptic curves, particularly of genus 3 over $\mathbb{F}_q$, may be candidate groups of this form. In particular, they note that solving the DLP in these groups is not known to be solvable in less that $O(q^{4/3} \log^{O(1)} q)$ time, and conjecture that it is conservative to assume computing the group order requires $O(q \log^{O(1)} q)$ operations. Concretely, they suggest $\log q \sim 100$ might be sufficient in practice.

2 Jacobians of curves

Let k be a field with $\mathrm{char}(k) \neq 2$. Then a hyperelliptic curve C of genus g is the smooth completion of the affine curve given by $y^2 = f(x)$ with f monic, square-free, $\deg(f) = 2g + 1$. Concretely the smooth completion adds a point at infinity. Let $\overline{k}$ be the algebraic closure of k. A divisor on C is an element of the free $\mathbb{Z}$-module over C i.e. a formal sum of points $D = \sum_{P \in C} m_P[P]$ where all but finitely many of the m_P are zero.

The degree of D is the $\sum_{P \in C} m_P$, and the degree 0 divisors are a group $\mathrm{Div}^0(C)$ under addition. For any $g \in \overline{k}(C) = \overline{k}[X, Y]/(Y^2 - f(X))$, we can define a divisor $(g) = \sum_{P \in C} \mathrm{ord}_P(g)[P]$ where $\mathrm{ord}_P(g)$ is the order of zero or pole of g at P. Since $\mathrm{ord}_P(rs) = \mathrm{ord}_P(r) + \mathrm{ord}_P(s)$, the set of divisors of this form are a subgroup of $\mathrm{Div}^0(C)$, called the principal divisors $\mathcal{P}(C)$. The quotient $\mathrm{Div}^0(C)/\mathcal{P}(C)$ is a group, called the Jacobian $J(C)$. The Hasse-Weil bound gives $(q^{1/2} - 1)^{2g} \leq |J(C)| \leq (q^{1/2} + 1)^{2g}$. Concretely, one may represent any element of $J(C)$ with a divisor of form $\sum_{i \in \{1,\ldots,g\}} [P_i] - g\infty$. For any (x, y) in the affine part

*jonatlee@microsoft.com

of C, the function $g(X, Y) = X - x$ implies that $[(x, y)] + [(x, -y)] - 2[\infty] \in \mathcal{P}(C)$. So we can insist that the x-coordinates of the P_i are distinct.

In elliptic curves, $g = 1$ and there is a trival map between divisors $D_i = [P_i] - [\infty]$ and points $P_i \in C$. The classical geometric interpretation of point addition by equating the relationship $P_1 + P_2 + P_3 = 0$ with P_1, P_2, P_2 lying on a line naturally extends to the existence of a linear function vanishing on C only at P_1, P_2, P_3, i.e. that the divisor $[P_1]+[P_2]+[P_3]-3[\infty]$ is principal. For $g > 1$, this generalizes to the existence of a polynomial in $\overline{k}(C)$ with zeros given by the affine part of the sum of the divisors D_1, D_2, D_3. For the $g > 1$ case, it is convenient to use the Mumford representation, where a divisor D is represented by a pair of polynomials $r, s \in k[X]$ with $\deg(s) < \deg(r) \leq g$. If the affine part of D is $\sum_i [P_i]$, $P_i = (x_i, y_i)$, then $r = \prod_i (X - x_i)$ and $s(X_i) = y_i$. Cantor [12] gave explicit formulae for computation of group operations in this representation, since substantially optimized [15].

3 Schoof-Pila type algorithms

We give a brief presentation of the key points of the Schoof-Pila type algorithms, and refer the interested reader to [1, 23, 24] for a more in depth discussion.

Let k be a finite field $\mathbb{F}_q$. Given a hyperelliptic curve C over k, $J = J(C)$ is a variety over k, and we can naturally also consider J over $\overline{k}$. Then J/k is the restriction of $J/\overline{k}$ to the fixed points of the Frobenius endomorphism $\pi : z \to z^q$. Let $\chi \in k[X]$ be the characteristic polynomial of π; $|J(C)| = \chi(1)$. The complex roots of χ are guaranteed to have modulus $q^{1/2}$ and $\deg(\chi) \leq 2g$, so the χ are bounded by $2\binom{2g}{g} q^g \sim 2^{2g+1} q^g$. So to find $\chi(1)$ it suffices to reconstruct $\chi \mod \ell$ for most primes $\leq O(g \log q)$. π acts on the ℓ-torsion points $J[\ell] \subset J/\overline{k}$, a $2g$-dimensional vector space over k, with characteristic polynomial $\chi \mod \ell$. Since ℓ is small, one can compute a basis of $J[\ell]$ explicitly and decompose points in $J[\ell]$ in such a basis. In this basis the action is a $2g \times 2g$ matrix.

It remains to compute the action of π on the basis of $J[\ell]$, without computing dense polynomials of linear degree. To do this one uses a iterative square-and-multiply modulo ℓJ. This is done by finding some set of functions vanishing on $J[\ell]$, and reducing all intermediate results modulo these functions. For the $g = 1$ case, these are the multivariate division polynomials ψ_ℓ, with $d = \deg(\psi_\ell) = O(\ell^2)$. Following Elkies, for $\sim 1/2$ of all primes one can instead work modulo a polynomial of degree $d = O(\ell)$ which reduces the asymptotic complexity substantially. For $g > 1$, Cantor gives analogous polynomials for the Mumford form [13], and Abelard [1] proves that these polynomials have degree $d = O_g(\ell^3)$ in general, and $d = O(\ell^2)$ for $g = 3$. This allows the complexity to be effectively bounded. Abelard also shows that for curves with real multiplication, the degree is reduced to $d = O(\ell^{3/g})$.

4 Asymptotic complexities of existing results

We write $\tilde{O}(x)$ for $O(x \log^c x)$ for some constant c. We collect a range of results [1, 4, 18, 24, 27]. Note that these results count bit operations under the asymptotic that operations in $\mathbb{F}_q$ cost $\tilde{O}(\log q)$ bit operations.

Genus g	Simple	Fast Resultants [27]	With RM [3, 2]
$g = 1$ [24]	$\tilde{O}(\log^4(q))$	—	—
$g = 2$ [18]	$\tilde{O}(\log^8(q))$	$\tilde{O}(\log^{8-2/\omega}(q))$	$\tilde{O}(\log^5(q))$
$g = 3$ [1]	$\tilde{O}(\log^{14}(q))$	$\tilde{O}(\log^{14-4/\omega}(q))$	$\tilde{O}(\log^6(q))$
$g > 3$ [4]	$\tilde{O}_g(\log^{O(g)}(q))$	—	$\tilde{O}_g(\log^9(q))$

Figure 1: Asymptotic complexity of finding group orders for Jacobians of curves of genus g over $\mathbb{F}_q$.

The value ω is associated to the multiplication of $n \times n$ matrices for $n \sim (\log q)^{2(g-1)/\omega}$ matrices. Neglecting the fast resultant optimization, in general these bounds have the form $\tilde{O}_g(d^{4g} \log^{O(1)} q)$, which is to be expected as the bottleneck is reduction by $2g$ polynomials of degree d, which is naturally quadratic in d^{2g}; loops and replacement of arithmetic in k with bit operations account for the $\log^{O(1)} q$. For $g > 3$, the

natural conjecture is that the number of bit operations is $\tilde{O}_g(\log^{6g+3}(q))$. It should be noted that to prove these results, one uses resultant-based techniques [1]. In practice, Gröbner basis reduction is likely to be more efficient, but it is difficult to rigorously prove polynomial-time bounds on these algorithms.

5 Security implications of point counting

It has been recently proposed [17] that taking $g = 3$, $q \sim 2^{100}$ provides an effective set of parameters. They estimate the cost of finding discrete logs in this group with the methods of Laine and Lauter [19] to be around 2^{113} field operations; this estimate neglects the implied log factors in $\tilde{O}$ notation and so is conservative.

However, Abelard [1] provides algorithms to compute the order of this group that are polynomial in $\log q \sim 100$. Taking $\omega = 3$, the number of bit operations required to compute the order of the group is on paper only $\tilde{O}(\log^{38/3} q)$. Neglecting the $\tilde{O}$ as in [17] leads to an estimate of only $\sim 2^{84}$ bit operations to compute the order of the group. To obtain an estimate of 2^{128} bit operations for $g = 3$ would require $\log q \sim 1100$. Using point compression, this suggests the size of a group element would be ~ 3300 bits.

λ	RSA ($\log_2 N$) [5]	Class Group ($\log_2 D$) [7]	Genus 3 ($\log_2 \lvert J(C)\rvert$)
128	3072	1827	3300
192	7680	3598	110000

Figure 2: Group sizes implying an estimate λ bits of security for various candidate groups of unknown order.

However, these naive estimates neglect large constant factors. In [1], a computation with real multiplication with $q = 2^{64} - 19$, $\ell = 13$, $g = 3$ is reported, taking ~ 9 days on a 14 core 2.20GHz machine. An additional computation without use of the real multiplication for $q = 2^{64} - 19$, $\ell = 3$, $g = 3$ is reported, taking ~ 14 days on the same machine. In both cases the number of field operations is $\tilde{O}(d^{4g})$, where $d = O(\ell^{2/3})$ with real multiplication and $d = O(\ell^2)$ without. These results suggest an estimate for the number of cycles to compute $\chi \mod \ell$ for each $\log q \sim \ell < 60$ of $\sim 2^{108-112}$ cycles on one core. This would reduce the set of candidate group orders from an interval of size $\sim 2^{250}$ Hasse-Weil interval to a collection of arithmetic progressions of size $\sim 2^{180}$, which is amenable to a $O(2^{90})$ square root attack. This suggests that direct computation of the group order may be at least competitive with finding discrete logs.

It should also be noted that this naive estimate assumes no future advances. In the case of Schoof's algorithm, improvements of Elkies and Atkin allowed for the replacement of division polynomials of degree $O(\ell^2)$ by polynomials of degree $O(\ell)$ for some primes. For curves with real multiplication, this splitting can be proved generically for all ℓ and all genera, leading to the improved complexities of $\tilde{O}(\log q^6)$ in the genus 3 case [3]. Any development extending this kind of splitting to a non-trivial fraction of primes for generic curves of genus 3 would immediately reduce the complexity of order finding substantially.

6 Security of Trustless Setup

In [17], it is suggested that choosing $f \in k[X]$ a uniformly random monic, squarefree, degree $2g+1$ polynomial may be suitable as a way to pick a "nothing-up-my-sleeve" curve. Concretely one might derive the coefficients from the keystream of some stream cipher. It is natural to expect that curves of this form have $|J(C)|$ roughly uniform in their Hasse-Weil interval, and so for at least a subexponential fraction of curves the curve order will be smooth. In this case, a malicious party can generate many curves, and use generic group methods to find a curve of smooth order in time $L_{q^3}(1/2, \sqrt{2})$.

Sutherland [25] relates $|J(C)|$ to the size of related Jacobians over extension fields. This is used directly to find Jacobians of known, near-prime order. These are distinguished by the existence of a related Jacobian whose order happens to be smooth. The additional complexity of this transformation in genus 3 is $O(q^{1/4})$. Concretely, at $\log q \sim 50$, Sutherland has to test ~ 600 curves before finding the order of a curve, taking ~ 4 hours on one core, and for $\log q \sim 61$ similar results are obtained in ~ 8 days on one core. This suggests the following attack. A malicious actor runs the setup, and has some limited ability to choose a seed. They use these techniques to relate $|J(C)|$ to $|J(C')|$, and hope that $|J(C')|$ is smooth. In this case, they find $|J(C')|$

and deduce $|J(C)|$. However, $|J(C)|$ is still generic. In particular a recipient of $J(C)$ cannot simply check whether $|J(C)|$ is itself smooth, but must know which related Jacobian to check.

Broadly, this suggests that for the setup phase of hyperelliptic curves to be trustless, either the setup must be rigid or the group must be large enough to protect against these subexponential attacks. Matching $L_{q^3}(1/2, \sqrt{2})$ to the $L_D(1/2, 3/2\sqrt{2})$ attacks on the class group suggests that for non-rigid Jacobians $\log(q^3) \sim 1100$ is required for 128 bit security.

7 Conclusions

The existence of polynomial time algorithms to compute the order of Jacobians on curves of fixed genus may pose a substantial challenge to the use of these groups as groups of unknown order in a cryptographic setting. The largest computations of this form are of only a few core-days, whilst the largest public factorization and discrete log computations are ~ 1000 core-year computations. It is unclear whether practical improvements to the current algorithms will be forthcoming, as has happened with index calculus or with the Elkies-Atkin improvements to Schoof's algorithm [26, 24].

Choosing parameters to be conservative on paper at the 128 bit level causes genus 3 curves to be of comparable size to RSA or Class groups. More work is needed before the Jacobian groups of genus 3 curves can be recommended as efficient, secure groups of unknown order.

References

[1] S. Abelard. *Counting points on hyperelliptic curves in large characteristic : algorithms and complexity*. Theses, Université de Lorraine, Sept. 2018.

[2] S. Abelard. Counting points on hyperelliptic curves with explicit real multiplication in arbitrary genus. *Journal of Complexity*, 57:101440, 2020.

[3] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. Counting points on genus-3 hyperelliptic curves with explicit real multiplication. *The Open Book Series*, 2(1):1–19, 2019.

[4] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. Improved complexity bounds for counting points on hyperelliptic curves. *Found. Comput. Math.*, 19(3):591–621, June 2019.

[5] E. Barker, W. Barker, W. Burr, W. Polk, M. Smid, P. D. Gallagher, and U. S. For. Nist special publication 800-57 recommendation for key management – part 1: General, 2012.

[6] J. Benaloh and M. de Mare. One-way accumulators: A decentralized alternative to digital signatures. In T. Helleseth, editor, *Advances in Cryptology — EUROCRYPT '93*, pages 274–285, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.

[7] J.-F. Biasse, M. J. Jacobson, and A. K. Silvester. Security estimates for quadratic field based cryptosystems. In R. Steinfeld and P. Hawkes, editors, *Information Security and Privacy*, pages 233–247, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[8] D. Boneh, B. Bünz, and B. Fisch. Batching techniques for accumulators with applications to iops and stateless blockchains. In *Annual International Cryptology Conference*, pages 561–586. Springer, 2019.

[9] D. Boneh, B. Bünz, and B. Fisch. A survey of two verifiable delay functions. Cryptology ePrint Archive, Report 2018/712, 2018. `https://eprint.iacr.org/2018/712`.

[10] J. Buchmann and S. Hamdy. A survey on iq cryptography. In *In Proceedings of Public Key Cryptography and Computational Number Theory*, pages 1–15, 2001.

[11] B. Bünz, B. Fisch, and A. Szepieniec. Transparent snarks from dark compilers. Cryptology ePrint Archive, Report 2019/1229, 2019. `https://eprint.iacr.org/2019/1229`.

[12] D. G. Cantor. Computing in the jacobian of a hyperelliptic curve. *Mathematics of computation*, 48(177):95–101, 1987.

[13] D. G. Cantor. On the analogue of the division polynomials for hyperelliptic curves. *Journal fur die reine und angewandte Mathematik*, 447:91–146, 1994.

[14] H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer Publishing Company, Incorporated, 2010.

[15] C. Costello and K. Lauter. Group law computations on jacobians of hyperelliptic curves. In *International Workshop on Selected Areas in Cryptography*, pages 92–117. Springer, 2011.

[16] I. Damgård and E. Fujisaki. A statistically-hiding integer commitment scheme based on groups with hidden order. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 125–142. Springer, 2002.

[17] S. Dobson and S. D. Galbraith. Trustless groups of unknown order with hyperelliptic curves. Cryptology ePrint Archive, Report 2020/196, 2020. `https://eprint.iacr.org/2020/196`.

[18] P. Gaudry and Éric Schost. Genus 2 point counting over prime fields. *Journal of Symbolic Computation*, 47(4):368 – 400, 2012. Special Issue for Joachim von zur Gathen at 60.

[19] K. Laine and K. Lauter. Time-memory trade-offs for index calculus in genus 3. *Journal of Mathematical Cryptology*, 9(2):95–114, 2015.

[20] J. D. Lee and R. Venkatesan. Rigorous analysis of a randomised number field sieve. *Journal of Number Theory*, 187:92 – 159, 2018.

[21] A. K. Lenstra, H. W. Lenstra, M. S. Manasse, and J. M. Pollard. The number field sieve. In A. K. Lenstra and H. W. Lenstra, editors, *The development of the number field sieve*, pages 11–42, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.

[22] H. Lipmaa. On diophantine complexity and statistical zero-knowledge arguments. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 398–415. Springer, 2003.

[23] J. Pila. Frobenius maps of abelian varieties and finding roots of unity in finite fields. *Mathematics of Computation*, 55(192):745–763, 1990.

[24] R. Schoof. Counting points on elliptic curves over finite fields. *Journal de théorie des nombres de Bordeaux*, 7(1):219–254, 1995.

[25] A. Sutherland. A generic approach to searching for jacobians. *Mathematics of Computation*, 78(265):485–507, 2009.

[26] T. C.-N. D. Team. CADO-NFS, an implementation of the number field sieve algorithm.

[27] G. Villard. On computing the resultant of generic bivariate polynomials. In *Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*, ISSAC '18, page 391–398, New York, NY, USA, 2018. Association for Computing Machinery.