Regev's attack on hyperelliptic cryptosystems

Razvan Barbulescu[1,2] and Gaetan Bisson[3]

[1] Institut de mathématiques de Bordeaux (Université de Bordeaux, CNRS, Bordeaux INP),
Talence, France
[2] Inria Bordeaux, Talence, France
[3] Laboratoire GAATI, University of French Polynesia

Abstract. Hyperelliptic curve cryptography (HECC) is a candidate to standardization which is a competitive alternative to elliptic curve cryptography (ECC). We extend Regev's algorithm to this setting. For genus-two curves relevant to cryptography, this yields a quantum attack up to nine times faster than the state-of-the-art. This implies that HECC is slightly weaker than ECC. In a more theoretical direction we show that Regev's algorithm obtains its full speedup with respect to Shor's when the genus is high, a setting which is already known to be inadequate for cryptography.

Keywords: HECC · quantum algorithms · DLP

1 Introduction

Hyperelliptic curve cryptography (see [Kob89]) is a competitive alternative to elliptic curve cryptography. The two receive equal attention in research works (see e.g. [CFA+05]). At a given security level, both require the same signature size and have implementations of similar cost (see e.g. [BS]). The fastest known classical attacks on those cryptosystems have the same cost up to a very narrow difference (see e.g. [BCM14b]) and the same number of quantum gates is required to implement Shor's attack (see e.g. [CGHZ23]). NIST standardized elliptic curve cryptography in 1999 (see [NIST99]) and bought patents on hyperelliptic curve cryptography to stimulate research. Note also that pairing-based cryptography provides advanced primitives which can be based on hyperelliptic curves; see e.g. [KT08, FS11, Dry12, AFK24].

Hyperelliptic curves have a geometric invariant called the genus, which is a positive integer g; elliptic curves form the particular case where $g = 1$. The Jacobian variety of a hyperelliptic curve defined over a field is an abelian group. Pollard's rho algorithm runs in $\tilde{O}(Q^{1/2})$ time where Q is the group order; it is the fastest known classical algorithm for solving the discrete logarithm problem (DLP) in Jacobian varieties of genus-one and -two hyperelliptic curves. Such curves are thus suitable candidates for cryptographic use. When the genus is three or more, the index calculus algorithm has a better complexity which remains exponential for small genera (see [Gau00]) and becomes subexponential for large genera (see [ADH94]).

Turning to quantum attacks, Shor's algorithm has variants for integer factorization and DLP in various groups. No practical implementation of Shor's algorithm at more than 20 bits of security has been publicly announced although this could conceivably be achieved in the coming years. In this spirit, it is important to identify cases where Shor's algorithm can be implemented with as few quantum resources as possible at a given security level. Indeed, as record attacks are made on specific problems, they help us extrapolate the

This work was supported in part by *Agence Nationale de la Recherche* under grants ANR-20-CE40-0013 and ANR-22-PNCQ-0002.

E-mail: razvan.barbulescu@u-bordeaux.fr (Razvan Barbulescu)

Date of this document: 2024-12-12.

cost of applying Shor's algorithm to other instances. For example, it is much harder to properly estimate the cost of integer factoring using Shor's algorithm than to extrapolate the cost of solving DLP on elliptic curves from that of integer factoring. Therefore we seek to estimate the precise cost of quantum attacks on all hyperelliptic curves, including those of genus three and more, although they are not directly used in cryptography. Indeed, a polynomial difference between two instances of the discrete logarithm problem cannot constitute a security guarantee.

In 2023, Regev [Reg23] proposed a quantum algorithm with a better complexity than Shor's. It was initially designed for integer factoring but was later extended to computing DLP's in the multiplicative group of finite fields [EG24] and in elliptic curves [BBP24]. Pilatte [Pil24] proved the correctness of all variants except for elliptic curves, where it relies on a series of conjectures from number theory.

In this paper, we extend Regev's algorithm to Jacobian varieties of hyperelliptic curves of arbitrary genus g; we show that it offers a significant speedup to attack the discrete logarithm problem compared to Shor's algorithm. This speedup grows with the genus and is already significant for $g = 2$.

Roadmap. In Section 2, we extend Regev's algorithm to Jacobian varieties of hyperelliptic curves of arbitrary genus g. In Section 3, we prove, under a heuristic assumption, that Regev's algorithm provides a minimum speedup of $\min(g, \sqrt{n})$ with respect to Shor's, where n denotes the bit size of the group order. In Section 4, we treat the specific case where $g = 2$ and we show that the actual speedup is greater than the minimum value of two: for specific curves relevant to cryptography, we obtain a speedup of nine. This is possible by extending the strategy of [BBP24]. Finally, in Section 5, we discuss how the heuristic assumption may be eliminated for large genera.

2 Extending Regev's algorithm

2.1 Hyperelliptic curves

A hyperelliptic curve H is an algebraic curve given by an equation of the form $y^2 + h(x)y = f(x)$ for two polynomials h and f such that $\deg h < \deg f$. Here, we briefly state results needed in this article and refer the reader to [CFA+05] for most proofs. Such a curve H has a unique non-affine point which we denote by ∞. The genus of H is the integer g such that $\deg f = 2g + 1$ or $2g + 2$. In particular, an elliptic curve is a hyperelliptic curve of genus $g = 1$. If h and f have coefficients in a field K of which L is an extension, then $H(L)$ denotes the group of L-rational points on H. By the Hasse–Weil theorem, when L is the finite field $\mathbb{F}_q$ of cardinality q, we have

$$|\#H(\mathbb{F}_q) - (q + 1)| \leq 2g\sqrt{q}.$$

The Jacobian variety $\mathrm{Jac}_L(H)$ is an abelian group which we simply write as $\mathrm{Jac}(H)$ in the case when $L = K$. The Mordell–Weil theorem states that, when L is a number field, we have $\mathrm{Jac}_L(H) \simeq T \times \mathbb{Z}^r$ for some finite group T and some integer r called the rank. When $L = K = \mathbb{F}_q$, an extension of the Hasse–Weil theorem above (see [CFA+05, th. 14.15]) yields

$$\# \mathrm{Jac}(H) \in \left[(\sqrt{q} - 1)^{2g}, (\sqrt{q} + 1)^{2g}\right].$$

The divisor representation of a point on $\mathrm{Jac}_L(H)$ is an expression of the form $D = x_1P_1 + \cdots + x_kP_k$ where $x_1, \ldots, x_n$ are integers and $P_1, \ldots, P_k$ are points of $H(L)$. A divisor representation is reduced if it is of the form $D = P_1 + \cdots + P_r - r(\infty)$ with $r \leq g$ where no point P_i is the opposite of another. Any point of $\mathrm{Jac}_L(H)$ admits a unique reduced divisor representation.

The Mumford representation of a point of $\mathrm{Jac}_L(H)$ is a couple $(u, v) \in L[x]$ such that $v^2 + vh - f$ is a multiple of u. A Mumford representation is reduced if $\deg v \leq \deg u \leq g$. Given the reduced divisor representation of a point of the Jacobian, one can compute the reduced Mumford representation via the CRT algorithm (see [CFA$^+$05, Sec 14.3.2]) whose complexity is $O(g(\log g)^2)$ (see [vzGG03, Sec 10.2]).

Cantor's algorithm [Can87] computes the addition of two points in reduced Mumford representation; it proceeds in two steps: the composition, which produces a non-reduced representation of the sum, and the reduction, which outputs an equivalent reduced Mumford representation. Both steps use $O(g \log g)$ operations in $\mathbb{F}_q$ assuming $q \geq g$ as is always the case in cryptography.

By [HvdH21], the binary cost of computing multiplications in $\mathbb{F}_q$ is $O(q \log q)$. Since the binary size n of the order of the Jacobian variety of a hyperelliptic curve H satisfies $n = \log_2 |\mathrm{Jac}(H)| \sim g \log_2 q$ by the Hasse–Weil bound, we deduce that computing its addition law takes $O(n(\log n)^3)$ binary operations.

2.2 Regev's algorithm in a nutshell

Consider an abelian group G using additive notation. Given two group elements a and b, the discrete logarithm problem consists in finding an integer x such that $a = [x]b$. To achieve this, Shor's algorithm computes, in quantum superposition, $[z]b$ for all n-bit integers z, where n is the bit size of the order of G. Regev's algorithm evaluates, also in superposition, expressions of the form $[z_1]b_1 + \cdots + [z_d]b_d$ where d is a parameter to be optimized, the coefficients $z_1, \ldots, z_d$ are n/d-bit integers, $b_d = b$, $b_{d-1} = a$ and the b_i's are carefully-selected elements of G for $1 \leq i \leq d-2$.

The improved complexity relies on two new ideas. Firstly, there exists a linear combination which sums to 0 and whose coefficients are n/d-bit integers. The idea is natural because the lattice

$$L = \left\{ (z_1, \ldots, z_d) \in \mathbb{Z}^d : \sum_{i=1}^{d} [z_i] b_i = 0 \right\} \tag{1}$$

can be shown to have volume $|G|$. The commonly-used Gaussian heuristic states that a randomly selected sublattice of $\mathbb{Z}^d$ admits a basis formed only of vectors of length less than $\exp(O(d)) \operatorname{Vol}(L)^{1/d}$. Secondly, the elements $b_1, \ldots, b_d$ can be chosen small, namely such that for any $\varepsilon \in \{0,1\}^d$, the sum $\sum_{i=1}^{d} [\varepsilon_i] b_i$ can be computed in $\tilde{O}(n)$ operations, that is, the cost of a single addition in G.

2.3 An extension of Regev's algorithm to Jacobian varieties of hyperelliptic curves

As presented above for general groups, Regev's algorithm extends readily to Jacobian varieties of hyperelliptic curves. What remains is to show that the elements $b_1, \ldots, b_d$ can be chosen in such a way that the sums $[z_1]b_1 + \cdots + [z_d]b_d$ can be efficiently computed and yield a solution to the DLP.

The elements $b_1, \ldots, b_{d-2}$ in Regev's algorithm have to be small, in other words, they have to admit representations of just a few bits. When the genus $g = g(n)$ grows to infinity with n we take, for $i \leq d-2$, $b_i = (P_i) - (\infty)$ for some points $P_i \in H(\mathbb{F}_q)$. When g is constant, in particular when $g = 1$ or $g = 2$, we let $b_i = P_{i,1} + \cdots + P_{i,g} - g(\infty)$ where all $P_{i,j}$ are points whose x and y coordinates admit lifts to $\mathbb{Q}$ of small height.

We now evaluate which speedup this yields compared to Shor's algorithm as $g \to \infty$ and in the specific case $g = 2$.

3 Speedup for large genera

The DLP variant of Regev's algorithm was first restricted [EG24] to computing discrete logarithms in $G = \mathbb{F}_p^\times$. A careful scrutiny of the proof shows that it applies equally to any group where the lattice of Equation (1) admits a basis of relatively short vectors and where a set of elements can be added much faster than general elements of G. These conditions are to be compared to the definition of Regev-friendly groups in [BBP24] and are made precise in the following result.

Theorem 1 (Theorem 1 in [EG24], reformulated)**.** *Let K be a constant. Let G be a commutative group whose order is an n-bit integer and where the group law can be computed in $\tilde{O}(n)$ binary operations. Let $d = d(n) = O(\sqrt{n})$. Assume that the lattice L defined in Equation (1) admits a basis whose vectors have length less than $\exp(Kn/d)$. If G has a subset S such that for any $s_1, \ldots, s_d \in S$ and any $\varepsilon \in \{0,1\}^d$, the sum $\sum_{i=1}^d [\varepsilon_i] s_i$ can be computed in $\tilde{O}(n)$ binary operations, then there exists an explicit probabilistic algorithm to compute any discrete logarithm in G by repeating $d+4$ times a quantum procedure of $\tilde{O}\left((d + \frac{n}{d})n\right)$ quantum gates, with a success rate of $1 - o(1)$.*

In the following, G is the Jacobian variety of a hyperelliptic curve H of genus g defined over a finite field $\mathbb{F}_q$. Its elements are computed in Mumford representation. The condition on the lattice involved in the algorithm are unconditionally proven only in the case when $G = \mathbb{F}_p^\times$ (see [Pil24]) although there are heuristic arguments that it holds for many types of groups.

Heuristic 1. *There exists a constant K such that the following holds. Let H be a hyperelliptic curve over $\mathbb{F}_q$ of genus g such that $|\operatorname{Jac}(H)|$ is an n-bit integer. Let $d = \min(g, \sqrt{n})$ and $b_1, \ldots, b_d$ be elements of $\operatorname{Jac}(H)$ of the form $(P) - (\infty)$ with $P \in H(\mathbb{F}_q)$ drawn uniformly at random. Then, almost surely, $b_1, \ldots, b_d$ span $\operatorname{Jac}(H)$ and the lattice L in Equation 1 has a basis where each basis vector has norm at most $T = \exp(Kn/d)$.*

We are now ready to prove the main result of this section.

Theorem 2 (under Heuristic 1)**.** *Let H be a hyperelliptic curve of genus g defined over $\mathbb{F}_q$, such that the cardinality of its Jacobian variety, $|\operatorname{Jac}(H)|$, is a n-bit integer. Then there exists an explicit probabilistic quantum algorithm which succeeds with probability $1 - o(1)$ and has complexity $\tilde{O}((d + \frac{n}{d})n)$ where $d = \min(g, \sqrt{n})$.*

Proof. The set $S = \{(P) - (\infty) : P \in H(\mathbb{F}_q)\}$ will play the role of the set of small elements of $\operatorname{Jac}_{\mathbb{F}_q}(H)$. We seek to apply Theorem 1 with $d = \sqrt{g}$. This requires three ingredients:

1. The addition law in the Jacobian variety is efficiently computable; indeed, Cantor's algorithm [CFA+05] has complexity $\tilde{O}(g)M(q) = \tilde{O}(n)$.
2. Consider now the size of S. Since $|\operatorname{Jac}(H)| \sim q^g$ is an n-bit integer, we have $n \sim g \log_2 q$. Since $|H(\mathbb{F}_q)| \sim q$, we have $|S| \sim q$. Then, it follows that $\log_2 S \sim \frac{n}{g} \geq n^{1/2} \geq d$.
3. Finally, we evaluate sums of the form $\sum_{i=1}^d [\varepsilon_i] s_i$ as follows. To compute the Mumford representation of the divisor $L_\varepsilon = \sum_{i=1}^d [\varepsilon_i](P_i) - [\sum_{i=1}^d \varepsilon_i](\infty)$, we set $u = \prod_{i=1}^d (x - x_i)$ and compute $v(x)$ such that $v(x_i) = y_i$. This can be done in time $O(M(g) \log g) M(\mathbb{F}_q)$ (see [vzGG03, Sec 10.2] and further $O(n (\log n)^2) = \tilde{O}(n)$.

Since the conditions of Theorem 1 are satisfied, the conclusion follows. □

This demonstrates that the full speedup which Regev's algorithm provides over Shor's is $\sqrt{n}$, where n is the bit size of the group order, assuming $g \geq \sqrt{n}$. Furthermore, as g and n both go to infinity, so does this speedup.

4 Speedup for $g = 2$

We now turn to the case $g = 2$ which is most relevant from a cryptographic viewpoint. The previous section shows that Regev's algorithm achieves a speedup of $\min(g, \sqrt{n}) = 2$ over Shor's. Here, we describe a strategy which yields an improved speedup of 8 or 9 on concrete examples. The key idea is to search for "twist" curves admitting suitable base elements $b_1, \ldots, b_{d-2}$.

4.1 A general strategy to find small elements on Jacobian varieties

We recall the strategy of [BBP24, Sec 3.3] for elliptic curves and extend it to hyperelliptic curves. Consider the equation defining H: $y^2 = f(x)$. The Mordell–Weil theorem, which states that $\mathrm{Jac}_{\mathbb{Q}}(H) \simeq T \times \mathbb{Z}^r$ for a finite group T and a positive integer r, is constructive and allows us to extract a free family from any set of elements of $\mathrm{Jac}_{\mathbb{Q}}(H)$ (see e.g. [Sch95]). A reduction to $\mathrm{Jac}_{\mathbb{F}_q}(H)$ of a basis of the free part of $\mathrm{Jac}_{\mathbb{Q}}(H)$ may then be used as the elements $b_1, \ldots, b_{d-2}$ in Regev's algorithm; their height are small and their independence over $\mathbb{Q}$ makes it almost certain that Heuristic 1 holds.

In [BBP24, Sec 3.3], the following is noted for $g = 1$ but extends readily to greater genera. If the curve H_δ of equation $\delta y^2 = f(x)$ has a Jacobian variety of large Mordell–Weil rank and if δ is a square in $\mathbb{F}_q$ then the map $(x, y) \mapsto (x, \sqrt{\delta} y)$ is an birational equivalence from H to H_δ which induces an isomorphism between the associated Jacobian varieties. Hence, the discrete logarithm problem of $\mathrm{Jac}_{\mathbb{F}_q}(H)$ may be reduced to that of $\mathrm{Jac}_{\mathbb{F}_q}(H_\delta)$; since the latter has large rank over $\mathbb{Q}$, a free set of small-height generators for $\mathrm{Jac}_{\mathbb{Q}}(H_\delta)$ can then be used in Regev's algorithm. Concretely, if $r = \operatorname{rank} \mathrm{Jac}_{\mathbb{Q}}(H_\delta)$, we can run Regev's algorithm with the parameter $d = r + 2$.

We now show that, in fact, any value of δ can be used.

4.2 Exploiting birational equivalence defined over extension fields

For arbitrary values of δ, the birational equivalence $H \to H_\delta$ may be defined over an extension of the base field $\mathbb{F}_q$. We show that this nevertheless induces a reduction of the discrete logarithm from $\mathrm{Jac}_{\mathbb{F}_q}(H)$ to $\mathrm{Jac}_{\mathbb{F}_q}(H_\delta)$. The reduction proceeds as follows.

Denote by $P = [\alpha]Q$ the discrete logarithm problem to be solved on $\mathrm{Jac}_{\mathbb{F}_q}(H)$ where $\alpha \in \mathbb{Z}$ is the unknown. Let κ be such that the birational equivalence $\varphi : H \to H_\delta$ is defined over $\mathbb{F}_{q^\kappa}$. Through φ, the DLP becomes $\varphi(P) = [\alpha]\varphi(Q)$ on $J = \mathrm{Jac}_{\mathbb{F}_{q^\kappa}}(H_\delta)$. Now, let π be the Frobenius endomorphism of $\mathbb{F}_q$ acting on J; since it commutes with scalar multiplication, we have $\pi^k(\varphi(P)) = [\alpha]\pi^k(\varphi(Q))$ and thus we deduce

$$\sum_{k=1}^{\kappa} \pi^k \varphi(P) = [\alpha] \sum_{k=1}^{\kappa} \pi^k \varphi(Q)$$

where both sums are stabilized by π and are thus points in $\mathrm{Jac}_{\mathbb{F}_q}(H_\delta)$. This yields a discrete logarithm problem in $\mathrm{Jac}_{\mathbb{F}_q}(H_\delta)$ which cannot be trivial (e.g. of the form $0 = [\alpha]0$) since, in cryptographic applications, Q being a generator of a large subgroup of $\mathrm{Jac}_{\mathbb{F}_q}(H)$, it cannot be in the trace zero subgroup.

The original discrete logarithm problem in $\mathrm{Jac}_{\mathbb{F}_q}(H)$ is thus reduced to another in $\mathrm{Jac}_{\mathbb{F}_q}(H_\delta)$ where we can exploit the large rank of $\mathrm{Jac}_{\mathbb{Q}}(H_\delta)$.

4.3 Cryptographic examples

Example 1. Take the Buhler–Koblitz curve "4GLV127-BK" which has been considered for cryptographic use [BK98, BCM14a]. It is defined as $C : y^2 = x^5 + 17$ over $\mathbb{F}_q$ where $q = 2^{64}(2^{63} - 27443) + 1$; its Jacobian variety $\mathrm{Jac}(C)$ has a 254-bit prime group order.

Using Magma [BCP97] we searched for "twist curves" of the form $C_\delta : y^2 = x^5 + 17\delta$ whose Jacobian variety has large rank over $\mathbb{Q}$. We proceeded as follows. First, for each integer $\delta \in \{1, \dots, 128 \cdot 10^4\}$, we enumerated all rational points on $J_\delta = \mathrm{Jac}_{\mathbb{Q}}(\mathbb{C}_\delta)$ of height less than 10^3 and extracted a reduced basis; this quickly gave us a lower bound on $\mathrm{rank}(J_\delta)$. Second, when this lower bound was two or greater, we determined the full Mordell–Weil group of J_δ and its rank under the GRH; a timeout was reached for three values of δ (2450, 131625, and 273904) which were consequently excluded. This entire computation ran on 128 cores for a total time of 448 960 seconds or about one hour of elapsed time. The distribution of $\mathrm{rank}(J_\delta)$ is as follows.

$\mathrm{rank}(J_\delta)$	$\#\{\delta\}$
$0^\star$	1 265 371
$1^\star$	14 282
2	191
3	89
4	50
5	9
6	3
7	2
≥ 8	0

The asterisks on the first two columns denote that they count Jacobians J_δ of which the rank could not be proven to be two or greater by enumerating rational points of height less than 10^3. This serves as a useful heuristic to quickly filter out "unpromising" twists and thus speed up computations; nevertheless, these Jacobians could very well be (and some indeed are) of greater rank.

For $\delta = 190304$, we find that the variety $\mathrm{Jac}_{\mathbb{Q}}(C_\delta)$ has rank seven and admits the following basis in projective Mumford coordinates:

$$\begin{aligned}
(x + 18, -1160, 1),\\
(x - 26, -3888, 1),\\
(x^2 - \frac{7}{9}x - 338, -\frac{881}{27}x - \frac{5242}{3}, 2),\\
(x^2 + 936, \frac{338}{3}x + 3888, 2),\\
(x^2 - 32x + 416, 55x - 1392, 2),\\
(x^2 - \frac{2409}{64}x + \frac{21333}{32}, -\frac{79605}{512}x + \frac{1063409}{256}, 2),\\
(x^2 - \frac{5345}{256}x + \frac{35373}{128}, -\frac{111569}{4096}x + \frac{4153949}{2048}, 2).
\end{aligned}$$

Note that, while C, C_δ, and their Jacobian varieties are defined over $\mathbb{F}_q$, the birational map

$$\varphi : \begin{cases} C \longrightarrow C_\delta \\ (x, y) \longmapsto (\delta^{1/5}x, \delta^{1/2}y) \end{cases}$$

is only defined over $\mathbb{F}_{q^5}$ since δ is a square modulo q. We can thus apply the result of the Section 4.2 for $\kappa = 5$. As a consequence, we can attack the discrete logarithm problem in $\mathrm{Jac}_{\mathbb{F}_q}(C)$ nine times faster than Shor's algorithm.

Example 2. Hyperelliptic curves of the form $y^2 = x^5 + \delta x$ have been proposed for cryptography, especially for pairing-based applications [KT08, Dry12]. Section 4.2 implies that, over finite fields where δ does not vanish, the discrete logarithm problems of all such curves are equivalent; indeed, we have the birational equivalence:

$$\begin{array}{ccc} y^2 = x^5 + \delta x & \longrightarrow & y^2 = x^5 + x \\ (x, y) & \longmapsto & \left(\delta^{1/4} x, \delta^{5/8} y\right) \end{array}$$

Therefore, to attack such curves using Regev's algorithm, it suffices to find one which has large rank over the rationals. A search performed similarly to the above example exhibited the curve

$$y^2 = x^5 + 3083871x$$

of which the Jacobian variety has rank six over $\mathbb{Q}$. Consequently, we can attack the discrete logarithm problem in all such curves eight times faster than Shor's algorithm.

Note, however, that our strategy does not apply to hyperelliptic curves for which no twist of large rank and with small generators can be identified; for example, this is the case of the Gaudry–Schost curve [GS04, Appendix A].

5 Conclusion and open questions

The twist strategy outlined above allows us to obtain almost the full speedup of Regev's algorithm with respect to Shor's: for the cryptographic examples given above, we can run Regev's algorithm using the parameter $d = 9$ whereas the optimal value would be $\sqrt{256} = 16$. Furthermore, the speedup is greater than the one obtained for elliptic curves in [BBP24]. This corroborates with the fact that the speedup is $\min(g, \sqrt{n})$ when g is large.

It is an open question to prove Heuristic 1 because Pillate's proof [Pil24] uses classical results about the rational primes which have no proven analogue in the case of the Jacobian varieties of hyperelliptic curves. In particular one needs to estimate partial sums associated to the Dirichlet L-function of the character or the class group of a function field. An even more exploratory question would be to estimate what ranks can be obtained by twisting a given hyperelliptic curve.

References

[ADH94] Leonard M Adleman, Jonathan DeMarrais, and Ming-Deh Huang. A subexponential algorithm for discrete logarithms over the rational subgroup of the Jacobians of large genus hyperelliptic curves over finite fields. In *Algorithmic Number Theory – ANTS-I*, pages 28–40. Springer, 1994.

[AFK24] Mónica P Arenas, Georgios Fotiadis, and Elisavet Konstantinou. Special tnfs-secure pairings on ordinary genus 2 hyperelliptic curves. In *International Conference on Cryptology in Africa*, pages 285–310. Springer, 2024.

[BBP24] Razvan Barbulescu, Mugurel Barcau, and Vicentiu Pasol. A comprehensive analysis of regev's quantum algorithm. *Cryptology ePrint Archive*, 2024.

[BCM14a] Joppe Bos, Craig Costello, and Andrea Miele. Elliptic and hyperelliptic curves: a practical security analysis. In *Public-Key Cryptography — PKC 2014*, volume 8383 of *Lecture Notes in Computer Science*, pages 203–220, 2014. doi:10.1007/978-3-642-54631-0_12.

[BCM14b] Joppe W Bos, Craig Costello, and Andrea Miele. Elliptic and hyperelliptic curves: A practical security analysis. In *Public-Key Cryptography–PKC 2014: 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings 17*, pages 203–220. Springer, 2014.

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system: the user language. *Journal of Symbolic Computation*, 24(3–4):235–265, 1997. doi:10.1006/jsco.1996.0125.

[BK98] Joe Buhler and Neal Koblitz. Lattice basis reduction, Jacobi sums and hyperelliptic cryptosystems. *Bulletin of the Australian Mathematical Society*, 58:147–154, 1998. doi:10.1017/S000497270003207X.

[BS] Peter Birkner and Peter Schwabe. Documentation of the Hector library. Available online at https://cryptosith.org/crypto/data/hector_documentation.pdf.

[Can87] David G Cantor. Computing in the Jacobian of a hyperelliptic curve. *Mathematics of computation*, 48(177):95–101, 1987.

[CFA$^+$05] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. CRC press, 2005.

[CGHZ23] Chao Chen, Peidong Guan, Yan Huang, and Fangguo Zhang. Quantum circuits for hyperelliptic curve discrete logarithms over the Mersenne prime fields. *Quantum Information Processing*, 22(7):274, 2023.

[Dry12] Robert Dryło. Constructing pairing-friendly genus 2 curves with split jacobian. In *Progress in Cryptology – INDOCRYPT 2012*, volume 7668 of *Lecture Notes in Computer Science*, pages 431–453, 2012. doi:10.1007/978-3-642-34931-7_25.

[EG24] Martin Ekerå and Joel Gärtner. Extending Regev's factoring algorithm to compute discrete logarithms. In *International Conference on Post-Quantum Cryptography–PQC 2024*, pages 211–242. Springer, 2024.

[FS11] David Mandell Freeman and Takakazu Satoh. Constructing pairing-friendly hyperelliptic curves using weil restriction. *Journal of Number Theory*, 131(5):959–983, 2011.

[Gau00] Pierrick Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In *Advances in cryptology–EUROCRYPT 2000*, pages 19–34. Springer, 2000.

[GS04] Pierrick Gaudry and Éric Schost. Construction of secure random curves of genus 2 over prime fields. In *Advances in Cryptology-EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004. Proceedings 23*, pages 239–256. Springer, 2004.

[HvdH21] David Harvey and Joris van der Hoeven. Integer multiplication in time $o(n \log n)$. *Annals of Mathematics*, 193(2):563–617, 2021.

[Kob89] Neal Koblitz. Hyperelliptic cryptosystems. *Journal of cryptology*, 1:139–150, 1989.

[KT08] Mitsuru Kawazoe and Tetsuya Takahashi. Pairing-friendly hyperelliptic curves with ordinary jacobians of type $y^2 = x^5 + ax$. In *Pairing-Based Cryptography – Pairing 2008*, volume 5209 of *Lecture Notes in Computer Science*, pages 164–177. Springer, 2008. doi:10.1007/978-3-540-85538-5_12.

[NIST99] The National Institute of Standards and Technology. Digital signature standard. Available online at https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf, 1999.

[Pil24] Cédric Pilatte. Unconditional correctness of recent quantum algorithms for factoring and computing discrete logarithms. *arXiv preprint arXiv:2404.16450*, 2024.

[Reg23] Oded Regev. An efficient quantum factoring algorithm. *arXiv preprint arXiv:2308.06572*, 2023.

[Sch95] Edward F Schaefer. 2-descent on the Jacobians of hyperelliptic curves. *Journal of number theory*, 51(2):219–232, 1995.

[vzGG03] Joachim von zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge university press, 2003.