Hyperelliptic Cryptosystems – Efficiency and Subexponential Attacks

Andreas Enge

HYPERELLIPTIC CRYPTOSYSTEMS

Efficiency and Subexponential Attacks

Andreas Enge

HYPERELLIPTIC CRYPTOSYSTEMS

Efficiency and Subexponential Attacks

Dissertation

von

Andreas Enge

eingereicht am
Institut für Mathematik der
Mathematisch-Naturwissenschaftlichen Fakultät
der Universität Augsburg im
August 2000

Erster Berichterstatter: Prof. Dr. Dieter Jungnickel
Zweiter Berichterstatter: Prof. Dr. Gerhard Frey
Tag der mündlichen Prüfung: 8. Dezember 2000

Typeset using LaTeX
Printed in Germany by Libri Books on Demand
ISBN 3-8311-1868-X

Meiner Familie
in Liebe gewidmet

Contents

Preface **1**

1 Public key cryptography and discrete logarithms **5**

1.1 Public key cryptography in groups 5

1.2 Discrete logarithms 7

1.2.1 Brute force algorithm 7

1.2.2 Pohlig–Hellman attack 8

1.2.3 Shanks's baby step giant step algorithm 9

1.2.4 Pollard's ρ- and λ-algorithms 9

1.2.5 Cryptographic consequences 13

1.3 Subexponentiality 13

2 Algebraic curves and function fields **17**

2.1 Algebraic curves 18

2.1.1 Affine plane curves 18

2.1.2 Projective plane curves 18

2.1.3 Curves over perfect fields 19

2.2 Function fields 20

2.2.1 Definition of function fields 20

2.2.2 Discrete valuations 20

2.2.3 Field of constants 22

2.2.4 Residue class field 23

2.2.5 Decomposition law 23

2.2.6 Number fields . . . 24
2.3 Divisors and Jacobians . . . 24
2.3.1 Divisors . . . 24
2.3.2 Riemann–Roch Theorem . . . 25
2.3.3 Finiteness of the Jacobian . . . 26
2.4 Ideal classes and regulator . . . 27
2.4.1 Ideal class groups . . . 27
2.4.2 Regulator . . . 28
2.4.3 Ideal class number versus regulator . . . 29
2.5 Weil's theorem . . . 30
2.6 Cyclic extensions . . . 31
2.6.1 Kummer extensions . . . 32
2.6.2 Artin–Schreier extensions . . . 33

3 Hyperelliptic curves **37**
3.1 Normal forms . . . 38
3.1.1 Definitions . . . 38
3.1.2 Characteristic different from 2 . . . 39
3.1.3 Characteristic 2 . . . 41
3.2 Ideal (class) groups . . . 45
3.2.1 Imaginary and real quadratic representations . . . 46
3.2.2 Decomposition of prime ideals . . . 46
3.2.3 Principal divisors . . . 49
3.2.4 Semireduced divisors . . . 50
3.2.5 Reduced divisors . . . 55
3.3 Arithmetic . . . 57
3.3.1 The extended Euclidean algorithm . . . 57
3.3.2 Composition . . . 58
3.3.3 Reduction . . . 59

4 Efficiency of hyperelliptic cryptosystems 65
4.1 Cryptographic setting . . . 65
4.2 Probability distribution . . . 66
4.3 Average complexity of the Euclidean algorithm . . . 67
4.4 Some more probabilities . . . 73
4.5 Average number of field operations . . . 75
4.6 Average bit complexity . . . 79

5 Smoothness 83
5.1 Arithmetical semigroups and formations . . . 84
5.2 Prime divisor theorem . . . 88
5.3 The subexponential function . . . 91
5.4 Smoothness in arithmetical semigroups . . . 92
5.5 Smoothness in class groups . . . 95

6 Subexponential algorithms for groups with unknown structure 103
6.1 Parameters . . . 104
6.1.1 Generating property . . . 104
6.1.2 Maximal exponent . . . 107
6.1.3 Two-parametric problems . . . 107
6.2 Algorithm . . . 108
6.2.1 Finding the group structure . . . 109
6.2.2 Computing discrete logarithms . . . 110
6.3 Analysis . . . 110
6.3.1 Finding a relation . . . 110
6.3.2 Linear algebra . . . 112
6.3.3 Success probability . . . 113
6.3.4 Running time . . . 114
6.3.5 Subexponentiality . . . 115
6.4 Previous algorithms . . . 117
6.5 Group structure . . . 119
6.6 Implementation . . . 119

7 Subexponential algorithms for groups with known order 123

7.1 Algorithm 124

7.2 Linear algebra 125

7.3 Analysis 127

7.3.1 Success probability 127

7.3.2 Running time 128

7.3.3 Subexponentiality 130

7.4 Cyclic subgroups 133

7.4.1 Perturbing with elements of the complement 133

7.4.2 Using a basis for G 134

7.5 Implementation 135

Bibliography 137

Index 147

Preface

During the past decades, cryptography has left the realm of the military and secret services and is on its way to becoming a tool routinely used by the general public. There are two reasons for these changes.

On the one hand, the increasing dependence of our economy on electronic means of communication and the growing acceptance of the Internet for carrying out commercial transactions and exchanging private information creates new demands for the protection of electronic data. The recent hacker attacks on Internet based companies and the comprehensive spying network "Echelon" installed by the anglosaxon countries over Europe demonstrate the threat the new information technologies constitute for our economic prosperity and for the civil rights. In fact, it is easily possible to scan and even modify electronic messages on a large scale basis.

The only effective countermeasure is to routinely encrypt all electronic transactions. When using conventional, *symmetric* cryptosystems, the two communicating parties have to agree on a common secret key beforehand, so that this type of cryptography is perfectly suited for the needs of hierarchically structured organisations with sufficient means of distributing secret keys. The general demand for data protection, however, cannot be met by symmetric cryptography, since a network like the Internet is an only loosely organised community of up to hundreds of millions of participants, who moreover can easily fake their identities and thus cannot trust one another too much. On the other hand, *public key* or *asymmetric* cryptography, discovered in the late seventies, offers a solution to the problems occurring precisely in heterogenous networks.

Public key cryptography emerged from an amazing synthesis of engineering, computer science and branches of mathematics which were so far counted among the "purest" ones, namely number theory and algebraic geometry. The first such systems were based on the perceived difficulty of factoring large numbers and computing discrete logarithms in large finite fields. In complexity theory, a problem is usually considered "hard" if only exponential algorithms are known to solve it. It turned out, however, that the two problems mentioned above can be solved in subexponential time, so that key sizes of about 1000 bits are needed to produce secure systems. An attractive alternative is

provided by cryptosystems based on the difficulty of computing discrete logarithms in algebraic groups, the state of the art being elliptic curve cryptosystems. Since no general subexponential attacks are known for elliptic curve cryptosystems, they require key sizes of only about 160 bits. A natural extension of this concept is to consider Jacobians of more general curves over finite fields, for instance of hyperelliptic curves. Surprisingly enough, it was soon conjectured that hyperelliptic cryptosystems of higher genus are less secure than elliptic ones since they are probably prone to subexponential attacks. Another possible generalisation, cryptosystems based on class groups of number fields, turned out to be vulnerable by subexponential algorithms as well.

In this book, we treat two main topics. First, we examine hyperelliptic cryptosystems, providing a detailed analysis of the arithmetic in hyperelliptic Jacobians. Second, we study subexponential algorithms for computing discrete logarithms in finite abelian groups, which compromise the security of the corresponding public key cryptosystems. In fact, the existence of subexponential algorithms for apparently very different groups leads to the intuition that these groups are linked by some common structural properties. We develop a general framework for subexponential algorithms, which indeed covers all examples treated in the literature, and hopefully serves to devise algorithms for further groups in the future. The two topics come together in the description of a discrete logarithm algorithm in hyperelliptic Jacobians, for which we present the first proofs of subexponential running times.

In the first chapter, we give a short introduction to the concepts of public key cryptography to motivate the study of discrete logarithm algorithms and to demonstrate the power of subexponential attacks.

The second and third chapters deal with algebraic curves and their Jacobians and especially with hyperelliptic curves. In the cryptographic community, there is a certain awe of algebraic curve cryptosystems because of the depth of the underlying mathematical concepts. In the spirit of [Eng99], which has been used successfully to teach undergraduate courses on elliptic curve cryptography, we provide an introduction which is as elementary and self-contained as possible. To this purpose, the presentation is based on the analogy between number fields and function fields of algebraic curves, and thus ultimately on the theory of Dedekind rings. While due to space restrictions the chapter on general curves has to rely heavily on references to the literature, we develop the theory of hyperelliptic curves in great detail, providing proofs for all assertions. Hyperelliptic curves over fields of even characteristic, which have received less attention in the literature, but which turn out to be especially attractive for implementations, are treated with equal care. We provide a complete algorithmic characterisation of these curves and their real and imaginary quadratic models and show how to realise their arithmetic, hereby generalising several algorithms which have previously been formulated for odd characteristic only.

The existence of several algorithms for realising the arithmetic and the fact that the security of hyperelliptic cryptosystems is determined by two parameters lead to the question which algorithms and which curves to choose to achieve the most efficient arithmetic for

a given security level, a question which is addressed in Chapter 4. The basis of the analysis is the exact average complexity of a fundamental algorithm, namely the Euclidean algorithm on polynomials over finite fields.

The general framework for subexponential discrete logarithm algorithms is the topic of Chapter 5. We derive the structure a group must have so that the algorithms based on collecting relations may be formulated for the group, and verify that the classical examples and the new case of hyperelliptic curves fit into this context. The proofs of subexponentiality require that a certain smoothness assumption be fulfilled. We show that this is the case for hyperelliptic curves of large genus, hereby providing the first proof of such a theorem for class groups which does not rely on any unproven assumption. The generic model of Section 5.1 has been found together with Pierrick Gaudry and the results of Section 5.5 are joint work with Andreas Stein.

The last two chapters present two algorithms for computing discrete logarithms in the general context; for both of them, we rigorously prove a subexponential complexity. For hyperelliptic curves, the subexponentiality can only be obtained if their genus is "large", and the dependence of the running time on the genus can be quantified explicitly. The first algorithm applies without any further knowledge on the group; as a byproduct, it usually solves the fundamental task of computing the elementary divisors of the group and its canonic representation as a product of cyclic subgroups.

The second algorithm, which has been developed in collaboration with Pierrick Gaudry, is much faster. It applies to cyclic groups of known order, which are precisely the groups used in cryptography, and has basically the same complexity for all examples fitting into the general framework. For finite fields, we recover the running time of the fastest algorithms described in the literature; for class groups of imaginary quadratic number fields, we obtain a considerably faster algorithm than the previously known best one. Furthermore, the analysis shows that discrete logarithms can be computed in Jacobians of hyperelliptic curves of large genus as fast as in finite fields if only algorithms with a proven complexity are taken into account. Thus, hyperelliptic cryptosystems are even less secure than suspected so far. Moreover, the new algorithm can be implemented with only minor modifications, contradicting the observation made until now that subexponential algorithms either have a rigorously proven complexity, but are not implementable efficiently; or can be easily implemented, but have an only conjectured subexponential complexity.

Acknowledgements

I am grateful to Dieter Jungnickel, who marvellously supervised my research during the past years; I could always count on his help and advice when I needed it. I thank him and Reinhard Schertz for supporting me in finding the necessary funding for my work. In particular, my research was supported by a grant of the University of Augsburg

It was a special pleasure to spend several months at the Centre for Applied Cryptographic Research of the University of Waterloo, where I discovered most of the topics treated in this book and found many of the results. I thank Scott Vanstone and Alfred Menezes for the invitations and the funding of my stays and for fruitful discussions, which directed me in my research.

Special thanks are due to Pierrick Gaudry and Andreas Stein; working with them was a rewarding and an instructive experience.

I am grateful to my family and my friends, whose love and friendship gave me the force for this endeavour.

Chapter 1

Public key cryptography and discrete logarithms

The importance of public key cryptography for encrypting electronic transactions over a network like the Internet has already been pointed out in the preface. Maybe even more important for electronic commerce are digital signatures, which prevent a message from being altered during the transmission and allow to uniquely identify its sender. By their very nature, digital signatures cannot be realised in conventional symmetric cryptosystems, but only by asymmetric cryptography.

This introductory chapter gives a short overview of public key cryptography based on the discrete logarithm problem in cyclic groups and of possible attacks. "Conventional" such cryptosystems utilise the multiplicative group of finite fields, but the existence of subexponential attacks in this setting has led to the invention of cryptosystems based on algebraic groups. While the material of this chapter is fairly standard, it has been included to make the book self-contained and to provide motivation for the following chapters.

1.1 Public key cryptography in groups

To convey the flavour of public key cryptography, we describe one encryption and one signature scheme. Further examples can be found in [MOV97], Chapters 8 and 11.

Assume that an additively written group G and an element α of G of known finite order N are given. The first public key protocols described in the literature used the multiplicative group of finite prime fields; the generalisation to arbitrary groups, however, is straightforward.

Suppose that two persons, Kevin and Laura, wish to interact securely. Kevin chooses a random integer $k \in \{0, \ldots, N-1\}$ as his *secret key*, computes $k\alpha$ and makes this group element known as his *public key*. Similarly, Laura chooses a secret key $l \in \{0, \ldots, N-1\}$ and publishes $l\alpha$.

When Kevin wants to encrypt a message so that only Laura can decipher it, he chooses a further random number $r \in \{0, \ldots, N-1\}$ and sends $r\alpha$ to Laura. Now both Laura and Kevin share $rl\alpha$ as their common secret: Laura computes $rl\alpha = l(r\alpha)$ from the knowledge of $r\alpha$ and her secret key l, Kevin computes $rl\alpha = r(l\alpha)$ from the knowledge of r and Laura's public key $l\alpha$. So far, the procedure is known as Diffie–Hellman key exchange, with a slight modification due to ElGamal ([DH76, ElG85]). To send a message to Laura, Kevin may encode it as an element $m \in G$ and send $m + rl\alpha$; Laura subtracts $rl\alpha$ and recovers the original message. As at most N distinct messages may be encoded by elements of G, the original message usually has to be broken up into blocks, each of which is encrypted using a different random number r. This procedure is not very efficient; moreover, depending on the group, it is often not clear how to encode messages. (The last difficulty arises, for instance, in groups based on algebraic curves.) Thus, it is common to use a hybrid scheme by extracting a bit sequence from $rl\alpha$ and using it as the key for a conventional, symmetric encryption algorithm.

To sign a message, Kevin has to somehow prove his knowledge of his private key k (without actually revealing the key itself) to identify himself as the sender; and he has to link this information to the message to prevent its alteration. A possible implementation of this general idea is given by ElGamal's signature scheme ([ElG85]). To cope with encoding difficulties, we transfer the problem into the integers. So we assume that the message to be signed is an integer $m \in \{0, \ldots, N-1\}$ (which may, for instance, be obtained from the original message by applying a hash function). Moreover, we assume the existence of a bijection $g : G \to \{0, \ldots, N-1\}$. (Such a bijection is not always easy to obtain, especially for groups associated with algebraic curves. In practice, it is sufficient to use a low degree map, for which only few elements of G map to the same value.) Kevin then chooses a random integer $r \in \{0, \ldots, N-1\}$ which is coprime to N, computes $\beta = r\alpha$ and solves the congruence

$$m \equiv kg(\beta) + rs \pmod{N}.$$

Since r and N are coprime, there is a unique solution $s \in \{0, \ldots, N-1\}$. The pair (β, s) constitutes the signature for m and is sent to Laura along with m.

Laura may check the validity of the signature as follows. She computes $m\alpha$ and $(kg(\beta) + rs)\alpha = g(\beta)(k\alpha) + s\beta$ from the knowledge of β and s and Kevin's public key $k\alpha$. If the two values agree, she accepts the signature as valid.

1.2 Discrete logarithms

Clearly, the security of the encryption and signature schemes of the previous section depends on the difficulty of the *discrete logarithm problem* in G. Given $\beta = r\alpha$, we call r the discrete logarithm of β to the base α, denoted by $r = \log_\alpha \beta$; it is determined uniquely modulo the order N of α. (The notion of "logarithm" is derived from multiplicatively written groups; however, we prefer additive notation, which is more common for algebraic groups.) Someone who is able to compute Kevin's private key k from his public key $k\alpha$ by solving a discrete logarithm problem is henceforth able to decipher all encrypted messages sent to Kevin and to forge his signature.

Unfortunately, there might be simpler attacks. During a Diffie–Hellman key exchange as described in Section 1.1, an eavesdropper overhears $r\alpha$. He already knows $l\alpha$, and to crack the system he must compute $rl\alpha$. This problem is known as the Diffie–Hellman problem. As no algorithm is known to solve it without computing discrete logarithms, it is generally believed that the Diffie–Hellman problem and the discrete logarithm problem are equivalent. Polynomial time equivalence has indeed been shown for some classes of groups in [MW99].

Since all computations actually take place in the subgroup generated by α, we may assume that G is cyclic of cardinality N.

Clearly, the difficulty of the discrete logarithm problem depends on the group G. If G were infinite, for instance, then a bisection technique would probably yield an efficient algorithm: As there are only finitely many bit strings of a given length, the length of a binary representation of $l\alpha$ would somehow have to correlate with the size of l. This approach is no more available for finite groups, which makes the discrete logarithm problem more difficult in general.

Still, the problem may be easy for particular representations of the abstract cyclic group with N elements. For $G = \mathbb{Z}/N\mathbb{Z}$, it is possible to compute α^{-1} by the extended Euclidean algorithm and to derive $l = (l\alpha)\alpha^{-1}$ in polynomial time. During the remainder of this section, we give a brief survey of algorithms for solving the discrete logarithm problem in arbitrary finite cyclic groups fitting into the following model: Group elements are encoded by unique bit strings, which allows to compare elements obtained in different ways and to sort a subset of G. The group operations (addition, inversion) are executed by an oracle, which on input of an element returns its inverse or of two elements returns their sum.

To fix the notation, let $\beta = l\alpha$ be the element for which the discrete logarithm l is sought.

1.2.1 Brute force algorithm

The simplest algorithm for computing discrete logarithms is the brute force attack. One may check each possible solution $l \in \{0, \ldots, N-1\}$ if it is the correct discrete logarithm

by computing $l\alpha$ and comparing with β. Similar approaches are often used to crack conventional, symmetric cryptosystems which do not offer enough structure for more efficient algorithms. It is easily possible to distribute the range of elements to be tested over a computer network. Since there is almost no overhead in communication involved, the speed-up achievable by this type of parallelisation is roughly proportional to the number of participating machines.

Of course, the exact complexity of the algorithm depends on the concrete problem; in any case, the expected number of group operations is in $\Omega(N)$. Instances with a key length of 56 bits, i.e. 2^{56} possibilities to test, are solved routinely, sometimes in only a few hours by specialised hardware ([McN99]). Hence, cryptosystems with the once popular key length of 64 bits are also considered insecure; a lower bound generally deemed secure for the next few years is 80 bits, the next generation of symmetric cryptosystems will have key lengths of at least 128 bits ([NIS97]). A first consequence for discrete logarithm cryptosystems is that the group order should not be less than 2^{80}.

1.2.2 Pohlig–Hellman attack

A simple application of the Chinese Remainder Theorem allows to break the discrete logarithm problem in G into a number of such problems in its Sylow subgroups. Moreover, a lifting argument shows that it is in fact sufficient to compute logarithms in the subgroups of G of prime order. The following algorithm has been described in [PH78].

Write the factorisation of N as

$$N = \prod_{i=1}^{r} p_i^{\nu_i}.$$

The Chinese Remainder Theorem allows to reconstruct l if it is known modulo all the $p_i^{\nu_i}$. To compute $l \bmod p^\nu$, an inductive lifting argument can be used. Assume that $l \bmod p^e$ is known (in the beginning, $e = 0$), and write $l = (l_2 p + l_1)p^e + l_0$ with $l_0 \in \{0, \ldots, p^e - 1\}$ known and $l_1 \in \{0, \ldots, p-1\}$ and l_2 unknown. The correct value of l_1, i.e. of $l \bmod p^{e+1}$, is computed as follows. Let $\gamma = \frac{N}{p}$ be a generator of the subgroup of order p of G. Then

$$\frac{N}{p^{e+1}}(\beta - l_0\alpha) = \frac{N}{p^{e+1}}(l_2 p + l_1)p^e\alpha = (l_2 p + l_1)\gamma = l_1\gamma,$$

and

$$l_1 = \log_\gamma \left(\frac{N}{p^{e+1}}(\beta - l_0\alpha) \right).$$

Consequently, the security of a discrete logarithm cryptosystem does not depend so much on N, but rather on its largest prime factor. To resist the combination of a Pohlig–Hellman attack and a brute force attack, the largest prime factor of N must be at least as large as 2^{80}. Since the arithmetic efficiency of the group law and the amount of data

to be transmitted depend on N, one is interested in keeping N as small as possible. Thus by the discussion above, N should be prime or "almost prime" in the sense that it is the product of a large prime and some rather small cofactor.

1.2.3 Shanks's baby step giant step algorithm

Besides the brute force algorithm of complexity in $\Theta(N)$ group operations, there are several algorithms of complexity $\Theta\left(\sqrt{N}\right)$ which compute discrete logarithms in arbitrary groups. One of them is due to Shanks [Sha71]; it is deterministic, but its space requirement is also in $\Theta\left(\sqrt{N}\right)$ and it is not parallelisable in an efficient way. The basic idea is that looking up an element in an ordered set takes only logarithmic time, so that it is possible to test a large number of candidates for the discrete logarithm quasi simultaneously.

Let $L = \left\lceil \sqrt{N} \right\rceil$. Compute the list of *baby steps* $(i\alpha, i)$ for $i = \{0, \ldots, L\}$ and sort it by the first component. Compute $L\alpha$ and the *giant steps* $\beta - jL\alpha$ for $j \in \{0, \ldots, L\}$ and look up the computed elements in the baby step list. If one of them is found and equals $i\alpha$, then $l = jL + i$. The algorithm is guaranteed to succeed since any $l \in \{0, \ldots, N-1\}$ has such a representation.

Strictly speaking, the algorithm requires $\Theta(N)$ additions and $\Theta(N \log N)$ comparisons of group elements; since comparisons are usually much faster than additions, the effort for executing them is commonly omitted in the literature and Shanks's algorithm is referred to as a "square root algorithm".

1.2.4 Pollard's ρ- and λ-algorithms

In [Pol78], Pollard suggests two probabilistic algorithms for computing discrete logarithms which require practically no storage space. They exploit the concept of a *random walk*. Starting with a random group element $x_0 = a_0\alpha + b_0\beta$, the ρ algorithm constructs recursively sequences $(x_i)_{i\geq 0}$, $(a_i)_{i\geq 0}$ and $(b_i)_{i\geq 0}$ by

$$x_{i+1} = \begin{cases} \alpha x_i \\ \beta x_i \\ 2x_i \end{cases} \qquad a_{i+1} = \begin{cases} a_i + 1 \\ a_i \\ 2a_i \end{cases} \qquad b_{i+1} = \begin{cases} b_i \\ b_i + 1 \\ 2b_i \end{cases} \tag{1.1}$$

Notice that the invariant $x_i = a_i\alpha + b_i\beta$ holds if x_{i+1}, a_{i+1} and b_{i+1} are always chosen from the same line. As soon as a *collision* $x_i = x_j$ occurs, the relation

$$l(b_j - b_i)\alpha = (b_j - b_i)\beta = (a_i - a_j)\alpha$$

implies

$$l(b_j - b_i) \equiv a_i - a_j \pmod{N}.$$

If N is a large prime, then this equation determines l with an overwhelming probability.

It remains to explain how the line in (1.1) is chosen. It is desirable that x_{i+1} depends uniquely on x_i, since otherwise the sequence $(x_i)_{i\geq 0}$ is simply a random sequence of elements of G, which would have to be stored completely to detect a collision. Pollard illustrates his algorithms by a kangaroo jumping from one location x_i to the next one x_{i+1}, and the length of its leap is uniquely determined by the state of the ground in the place x_i. In practice, to preserve a maximal randomness, one partitions G "randomly" into three sets of roughly equal size $G = T_1 \dot\cup T_2 \dot\cup T_3$ and selects x_{i+1} from line j if $x_i \in T_j$; the partition is usually obtained by applying a hash function to x_i.

Since G is finite, the sequence $(x_i)_{i\geq 0}$ becomes ultimately periodic, i.e. the kangaroo crosses its own path, which can be pictured by the ρ shaped Figure 1.1.

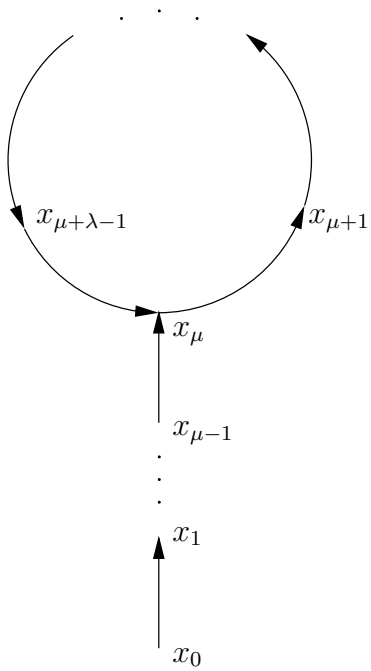

Figure 1.1: Pollard's ρ

Pollard suggests a cycle finding strategy due to Floyd (see [Knu81], Exercise 3.1.6), which consists of computing $y_i = (x_i, a_i, b_i, x_{2i}, a_{2i}, b_{2i})$ until $x_i = x_{2i}$. This corresponds to observing two kangaroos, an old and a young one, which both start in x_0. But whenever the old kangaroo makes one jump, the young one jumps twice. The scheme requires to store only the positions of the two kangaroos, i.e. one tuple y_i, at a time at the expense

of having to apply the iterating function thrice to move from y_i to y_{i+1}. If μ denotes the length of the preperiod and λ the length of the period as sketched in Figure 1.1, then a collision $x_i = x_{2i}$ occurs as soon as $i \geq \mu$ and $\lambda | i$.

Assuming that the iterating function $x_i \mapsto x_{i+1}$ is a random element of the N^N maps from G to G, it is possible to show that the expected value of i for which the first collision is detected is

$$\frac{\pi^2}{12}\sqrt{\frac{\pi}{2}N} \approx 1.03\sqrt{N}.$$

Extensive experiments showed that Pollard's iterating function requires on average a constant percentage more steps; however, differently chosen functions yield the predicted complexity ([Tes01].

Pollard's second algorithm is particularly useful if the discrete logarithm is known to lie in some range $[l_1, l_2]$ of length $L = l_2 - l_1 \ll N$; it then needs an expected number of $O(\sqrt{L})$ group operations. Let $S = \{s_1, \ldots, s_k\}$ be a small set of some rather small positive integers and $G = T_1 \dot\cup \ldots \dot\cup T_k$ a (random) partition of G into k sets of roughly equal size; in one example, Pollard uses $S = \{1, 4, 16, 64\}$. We consider the path of a tame kangaroo, which starts its jumps at some known place $x_0 = a_0\alpha$, for instance, $x_0 = l_1\alpha$, and construct the sequences $(x_i)_{i\geq 0}$ and $(a_i)_{i\geq 0}$ by

$$x_{i+1} = x_i + s_j\alpha_j \text{ and } a_{i+1} = a_i + s_j \text{ for } x_i \in T_j.$$

After some number $m+1$ of steps, the tame kangaroo passes the limit l_2, i.e., $a_{m+1} > l_2$, and we dig a hole in the place x_m to put up a kangaroo trap. A second, wild kangaroo starts in the location $x'_0 = \beta = l\alpha$ with unknown index l; we let $a'_0 = 0$ and compute sequences $(x'_i)_{i\geq 0}$ and $(a'_i)_{i\geq 0}$ by the same rules as for the tame kangaroo. If in some place, the wild kangaroo steps in the footprints of the tame one, i.e., $x_i = x'_j$ for some $i \leq m$, then it will follow the exact same path and end up in the trap, i.e., $x'_{m'} = x_m$ for some m'. Then the paths of the two kangaroos can be pictured by the λ shaped Figure 1.2, and the discrete logarithm can be computed via

$$l + a'_{m'} \equiv a_m \pmod N.$$

If on the contrary the wild kangaroo passes the index l_2, which is the case for $a'_{m'} > l_2$ for some m', then it has avoided the trap, and the algorithm must be restarted from the beginning.

It is observed in [OW99] that a combination of the ρ- and the λ-method is easily parallelisable. Consider a troop of wild kangaroos (or fleas, if you do not live in Australia and believe in the German proverb saying that nothing is more difficult to keep under control than a bag of fleas). Each of them starts at a random location $x_0 = a_0\alpha + b_0\beta$ and jumps around according to the rules of the ρ-method or some variation of it. Beforehand, a set $D \subseteq G$ of *distinguished points* is fixed, which must have the only property that they are easily recognisable, for instance because their binary representation begins by a chain of

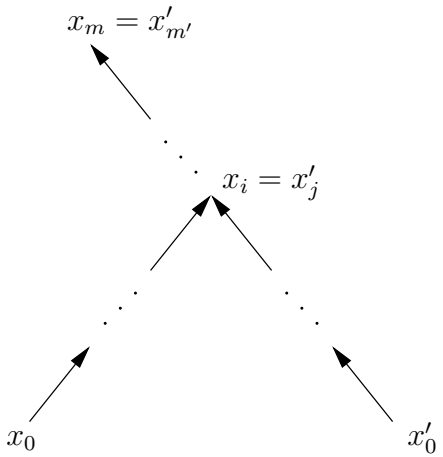

Figure 1.2: Pollard's λ

zeroes. In the distinguished points, biologists have set up observation equipment. As soon as a kangaroo reaches such a point, i.e. $x_i \in D$, it is spotted and a_i and b_i are recorded. If a second wild kangaroo reaches the same point, i.e. $x'_j = a'_j\alpha + b'_j\beta = x_i$, then

$$l(b'_j - b_i) \equiv a_i - a'_j \pmod{N},$$

and l is likely to be found. Each participant in the quest for l can keep their own troop of kangaroos, but the set of distinguished points reached has to be reported to a central server, which searches its internal database of sent in points for a collision. Observe that from the moment where two kangaroos have reached the same point, they march in unison, and the collision is detected as soon as they arrive at the next distinguished point. So the maximal time for which a collision may be pending is the maximal distance between two distinguished points, which can be kept small by choosing a large proportion of such points. On the other hand, the administrative overhead, i.e. the network traffic and the effort for handling the database at the central server, grows with the number of distinguished points. A reasonable choice of this parameter results in an algorithm the running time of which decreases roughly linearly with the number of participating computers.

The approach has been used successfully to compute discrete logarithms on an elliptic curve with approximately 2^{108} elements in an effort distributed over the Internet ([INR00, Har01]).

1.2.5 Cryptographic consequences

The conclusion to draw from the algorithms described above is that the order of a group must have a large prime factor, say of size about 2^{160}, so that the discrete logarithm problem in the group has a chance of being sufficiently hard for a user to trust a cryptosystem built on top of it. Notice that this condition is only necessary as is nicely illustrated by the easy discrete logarithm problem in $\mathbb{Z}/N\mathbb{Z}$ mentioned in the introduction to Section 1.2. In an abstract setting, it is also sufficient: Shoup showed that if the group operations are performed by an oracle so that no additional structural properties can be exploited, the algorithms above are optimal; precisely, any discrete logarithm algorithm has a complexity in $\Omega\left(\sqrt{p}\right)$ group operations, where p is the largest prime factor of N ([Sho97], see also [Nec94]). However, all concrete groups carry additional structure, so that this result is mainly of theoretical interest. Indeed, there is no proven meaningful lower bound for the complexity of any class of discrete logarithm problems.

The following section shows that a more efficient, *subexponential* algorithm is available for the multiplicative groups of finite fields, and Chapters 5 to 7 are devoted to derive a similar algorithm for further groups.

1.3 Subexponentiality

If G is the multiplicative group of a finite prime field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$, particularly efficient algorithms are available for computing discrete logarithms. They use additional features of this special representation, namely the unique decomposition of numbers into primes and the additive structure of $\mathbb{Z}/p\mathbb{Z}$. Their running time is *subexponential*, i.e. bounded above by

$$L_N(\alpha, c) := e^{c(\log N)^{\alpha}(\log\log N)^{1-\alpha}},$$

the *subexponential function* with respect to the input size $\log N$ of the problem and parameters $\alpha \in (0; 1)$ and $c > 0$. For $\alpha = 0$, the subexponential function is in fact the polynomial $(\log N)^c$ in the input size; for $\alpha = 1$, it is the exponential function $(e^c)^{\log N}$. So the smaller α, the more "tamely" the function behaves. Usually, subexponential algorithms with a proven running time have $\alpha = \frac{1}{2}$, so we abridge $L_N(1/2, c)$ by $L_N(c)$. (For finite fields, there are further algorithms with a conjectured running time of $L_N(1/3, c)$, the *number* and the *function field sieve* .) See Figure 1.3 for the graphs in logarithmic scale of some subexponential and polynomial functions.

As a large portion of this book is devoted to developing a general framework for subexponential discrete logarithm algorithms, we only sketch a simple version for finite fields in this place. It will provide us with the intuition for the requirements in the general setting, which we ultimately apply to Jacobian groups of algebraic curves.

Suppose that α is a generator of the multiplicative group $(\mathbb{Z}/p\mathbb{Z})^{\times}$, the elements of which are represented by their smallest positive residues modulo p, i.e. by integers in

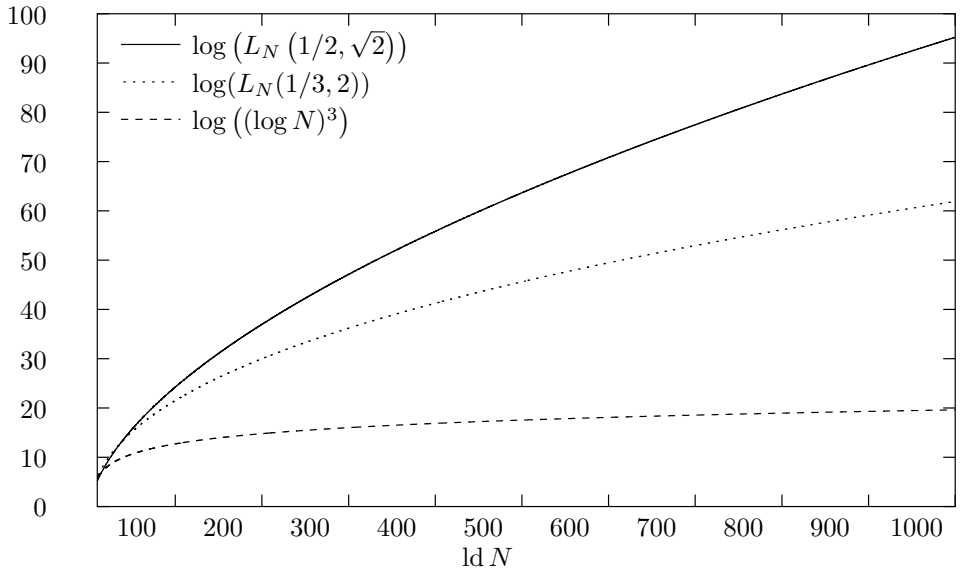

Figure 1.3: Subexponential and polynomial functions

$\{1, \ldots, p-1\}$. Let $\beta = \alpha^l$, where l is the sought discrete logarithm. We fix a *smoothness bound* $S \in \mathbb{N}$ and construct the *factor base* $\mathcal{P}_S = \{p_1, \ldots, p_n\}$ consisting of all prime numbers p_i with $\operatorname{ld} p_i \leq S$.

In a first step, we compute the discrete logarithms of the elements of the factor base as follows. We construct "linear combinations" (in multiplicative notation) $\prod_{i=1}^{n} p_i^{a_i}$ and reduce modulo p. (This is where the additive structure of $\mathbb{Z}/p\mathbb{Z}$ comes in, as reduction modulo p amounts to repetitive subtraction of p.) If the resulting number is S*–smooth*, i.e. can be expressed as another linear combination $\prod_{i=1}^{n} p_i^{b_i}$, then we have obtained a *relation* of the form $\prod_{i=1}^{n} p_i^{c_i} = 1$, where $c_i = b_i - a_i$ and very probably not all of the c_i equal zero. It follows that $\sum_{i=1}^{n} c_i \log_\alpha p_i = 0$. If enough such relations have been gathered, some linear algebra reveals the $\log_\alpha p_i$.

In a second step, β is expressed in terms of the factor base elements by a very similar procedure. Compute $\beta \prod_{i=1}^{n} p_i^{a_i}$ for random a_i and reduce modulo p until the resulting number is S–smooth and can be expressed as $\prod_{i=1}^{n} p_i^{b_i}$. Then $\beta = \prod_{i=1}^{n} p_i^{c_i}$ with $c_i = b_i - a_i$, so that $\log_\alpha \beta = \sum_{i=1}^{n} c_i \log_\alpha p_i$, and all the quantities in this sum are known.

Obviously, S has to be chosen according to two restrictions. First, it must not be too large. Precisely, it must be small enough so that $\mathcal{P}_S$ is of subexponential size and thus can be constructed in subexponential time. If the a_i are sufficiently random and uniformly distributed over a suitable range, then the linear combinations of factor base elements

represent random group elements, approximately according to a uniform distribution. So secondly, S has to be large enough so that one out of a subexponential number of group elements is S–smooth, and a relation can be obtained in subexponential time.

In finite prime fields, S can be chosen satisfying these two requirements, and an overall subexponential running time is obtained. (Precise formulations of algorithms and running time analyses follow in Chapters 6 and 7, which also deal with more general finite fields.)

Due to the applicability of subexponential algorithms, finite fields must be even larger than discussed in the previous section to provide secure cryptosystems. Currently, fields with $N \approx 2^{1000}$ are estimated as secure as "general" groups with $N \approx 2^{130}$ (cf. [LV00]), where by "general" we understand that only the algorithms of Section 1.2 apply.

The only general groups in this sense known and manageable at present are derived from algebraic curves, namely Jacobians of elliptic and hyperelliptic curves over finite fields. Until recently, they were believed to be as secure as general groups. For elliptic curves, this assumption has not been refuted (yet?) except for some easily avoided special cases ([MOV93, FR94, SA98, Sem98, Sma99]). Certain instance (of large genus) of hyperelliptic curves, however, were soon suspected to be attackable by subexponential algorithms ([ADH94]). The main part of this book is devoted to developing a general framework for subexponential discrete logarithm algorithms, see Chapters 6 and 7. Besides showing that all groups for which subexponential attacks are known can be treated in this context, we verify that also large genus hyperelliptic Jacobians are covered. By the analysis of Chapter 5, we present the first proof of subexponentiality for a discrete logarithm algorithm in hyperelliptic Jacobians over finite fields.

Chapter 2

Algebraic curves and function fields

We have seen in the previous chapter that public key cryptosystems may be constructed on top of any finite abelian group, provided that the discrete logarithm problem in the group is hard. Our aim in this chapter is to show that such groups may be derived from algebraic curves over finite fields, especially from so-called *elliptic* and *hyperelliptic curves*.

There are basically two different, but intertwining approaches to algebraic curves, the geometric and the field theoretic one. At first sight, the geometric point of view seems more intuitive, as it adopts concepts known from differential geometry over the real and complex numbers and generalises them to curves over arbitrary fields. Basically, all occurring functions are required to be rational instead of differentiable. In cryptography, elliptic curves are usually studied geometrically and their group law is given by rational functions.

The field theoretic approach was initiated by Dedekind and Weber in [DW82], who examined the rational functions on a curve rather than the curve itself. Its main attraction is the fact that function fields are close analogues of number fields, both being quotient fields of Dedekind rings. Thus, many number theoretic results carry over directly. Interestingly enough, the major open problem in number theory, the validity of the Riemann hypothesis, was proved in the function field setting by Weil (see Section 2.5). For a unified treatment see, for instance, [Eic63, Art67]. While the group associated with an elliptic curve is the curve itself, this is no more the case for hyperelliptic curves; instead, the group is a higher dimensional variety. It turns out, however, that it is described more easily as a structure derived from the function field of the curve than as a geometric object. Thus, our presentation will be more field theoretic.

For a thorough introduction to algebraic curves, see [Ful69], in which the geometric point of view is emphasised, or [Che51, Eic63, Sti93], in which the function field approach is adopted.

2.1 Algebraic curves

2.1.1 Affine plane curves

Let K be a field. During this and the next section we assume that K is algebraically closed, a restriction which will be relaxed later. An *affine plane curve* over K is an irreducible polynomial $C \in K[X,Y]\backslash K[X]$. A *point* $P = (x,y) \in K \times K$ lies on C if $C(P) = C(x,y) = 0$. Since no confusion is likely to occur, we refer to either the defining polynomial or the set of points lying on the curve as "the curve C". The irreducibility of C ensures that it is not the union of several curves, like $Y^2 - X^2$ is the union of the two lines $Y - X$ and $Y + X$.

To a curve C we associate its *coordinate ring* $K[C] = K[X,Y]/(C)$. The elements of $K[C]$ may be interpreted as *polynomial functions* from C to K; for algebraically closed K, it can be deduced from the irreducibility of C that $K[C]$ is exactly the ring of polynomial functions from C to K.

The field of fractions $K(C)$ of $K[C]$ is called the *function field* of C. It consists of the *rational functions* from C to K. As $K(C)$ is the smallest field containing $K[X]$ (and thus $K(X)$) and $K[X][Y]/(C)$, it is actually the finite algebraic extension $K(X)[Y]/(C)$ of $K(X)$.

A point on C is called *singular* if the partial derivatives $\frac{\partial C}{\partial X}$ and $\frac{\partial C}{\partial Y}$ vanish simultaneously in the point, *non-singular* otherwise. Curves without singular points are called *non-singular* or *smooth*; such curves are of special interest since the points on them correspond to discrete valuations of their function field, which ultimately allows to trade the geometric for the field theoretic point of view, see Section 2.2.

2.1.2 Projective plane curves

It turns out that the function field of a smooth curve has more valuations than the curve has points. To obtain a perfect correspondence, we have to add further points to the curve by switching to its projective closure.

The *projective plane* $\mathbb{P}^2(K)$ is the set of lines $K(x,y,z) = \{\lambda(x,y,z) : \lambda \in K\}$ through the origin with $(x,y,z) \in K^3\backslash\{(0,0,0)\}$ in three dimensional space. We denote the *projective point* $K(x,y,z)$ by $(x : y : z)$. Two sorts of points can be distinguished: If $P = (x : y : z)$ with $z \neq 0$, then $P = (\frac{x}{z} : \frac{y}{z} : 1)$; it is called *finite* and corresponds to the point $(\frac{x}{z}, \frac{y}{z})$ of the *affine plane* $\mathbb{A}^2(K) = K \times K$. If $z = 0$, then P is called an *infinite*

point. Thus, the projective plane can be seen as the disjoint union of an affine plane and points at infinity (which, in turn, form a projective line).

A *projective plane curve* over K is defined as an irreducible homogeneous polynomial $C^* \in K[X, Y, Z]$ which involves all three variables. A point $P = (x : y : z)$ lies on C^* if $C^*(P) = C^*(x, y, z) = 0$; the homogeneity of C^* ensures that C^* vanishes either in all representatives of P or in none of them.

If C^* is a projective curve, then its *dehomogenisation* $C = C^*(X, Y, 1)$ with respect to Z is an affine curve (unless $C \in K[X]$, which is a degenerate case we may exclude). Conversely, if C is an affine curve, then $C^* = Z^{\deg C} C\left(\frac{X}{Z}, \frac{Y}{Z}\right)$ is a projective curve, the *projective closure* of C. Points (x, y) on C correspond to finite points $(x : y : 1)$ on C^*. It is possible to associate a coordinate ring and a function field to C^* in a natural way, and it turns out that they are isomorphic to the corresponding structures of C. (For further details, see [Ful69], Chapter 4, or [Eng99], Chapter 2.)

A point on the projective closure C^* of an affine curve C is *singular* if the partial derivatives of C^* with respect to X, Y and Z vanish simultaneously in the point. For finite points, this definition is equivalent with singularity on C. If C^* does not contain any singular point, then it is called *non-singular* or *smooth*. The points on a smooth projective curves are indeed in a one-to-one correspondence with the discrete valuations of its function field. We briefly note that for a given curve C there is a smooth projective curve X such that $K(C) \simeq K(X)$. However, X will not be plane in general, but a one-dimensional subvariety in a higher dimensional space (see [Ful69], Chapter 7). Curves with isomorphic function fields are called *birationally equivalent*, cf. [Ful69], Section 2.6.

From now on, when we write down an affine equation C, it will often be silently understood that it stands for its projective closure or even its smooth projective model. As we mainly take the field theoretic approach, this will not pose any problems. But before introducing function fields in more detail, we have to cover the case of ground fields K which are not algebraically closed.

2.1.3 Curves over perfect fields

We assume from now on that K is a *perfect* field, i.e. that any finite algebraic extension of K is separable. Fields of characteristic zero are always perfect. If the characteristic of K is a prime p, the field is perfect if and only if for any $x \in K$ there is a p-th root of x in K. For instance, finite fields $\mathbb{F}_q = \mathbb{F}_{p^m}$ are perfect since $x^{q/p} = x^{p^{m-1}}$ is a p-th root of x. For a perfect field K, the algebraic closure $\overline{K}$ is Galois over K.

Let C be a plane affine curve defined over K, that is, an irreducible polynomial in $K[X, Y] \backslash K[X]$. (We restrict the presentation to affine curves, but of course exactly the same argumentation holds for projective curves.) Then C can be regarded as a curve over any extension field of K unless it becomes reduced. So we require now and in the remainder of this text that the curve be *absolutely irreducible*, i.e. irreducible in $\overline{K}[X, Y]$.

The *function field* $K(C)$ associated with C/K is defined as above as the field of fractions of its *coordinate ring*$K[C] = K[X,Y]/(C)$, i.e. as $K(X)[Y]/(C)$.

If $\sigma \in \mathrm{Gal}_{\overline{K}/K}$ is an element of the Galois group of $\overline{K}/K$, then it acts in the obvious way on points $P = (x,y)$ of $C/\overline{K}$ by $P^\sigma = (x^\sigma, y^\sigma)$. Hereby, points on C are transformed into each other since $C^\sigma = C$ and $C(P^\sigma) = C^\sigma(P^\sigma) = (C(P))^\sigma$. The points fixed by all elements of the Galois group are exactly the *K-rational points* of C, i.e. those with coordinates in K. In general, any point P on $C/\overline{K}$ is in fact defined over a finite Galois extension K_0/K, so that the orbit $\mathrm{Gal}_{\overline{K}/K}(P) = \mathrm{Gal}_{K_0/K}(P)$ is finite. Such an orbit is usually referred to as a *closed point*, which is justified by the fact that on a smooth projective curve the orbits are in a one-to-one correspondence with the discrete valuations of its function field and thus play the role of ordinary points in the algebraically closed setting. The *degree* of a closed point is its cardinality.

If $K = \mathbb{F}_q$ is finite, then $\mathrm{Gal}_{\overline{K}/K}$ is topologically generated by the *Frobenius automorphism* $\sigma : \overline{K} \to \overline{K}$, $x \mapsto x^q$; any finite extension K_0/K is Galois, and its Galois group is the cyclic group generated by $\sigma|_{K_0}$.

2.2 Function fields

In the previous sections, function fields occurred as the fields of rational functions on algebraic curves (whence they got their name). In this section, we provide an independent definition and show how the new concept is related to the geometric point of view.

2.2.1 Definition of function fields

Let K be a perfect field. A *function field* F/K (in one variable) is a finite algebraic extension of the *rational function field* $K(X)$. Note that $K(X)$ is not perfect any more, so that $F/K(X)$ need not be separable. However, it is always possible to choose an element $X \in F$ such that $F/K(X)$ is a simple separable extension $K(X)[Y]/(C)$ (see [Sti93], Proposition III.9.2). Here, $C \in K(X)[Y]$ is an irreducible polynomial, and after multiplying with the least common denominator of its coefficients we may assume that $C \in K[X,Y]$. This shows that F is indeed the function field of the plane curve C in the sense of the previous sections. However, C is only defined up to birational equivalence.

2.2.2 Discrete valuations

To relate the points on C with the structure of F, we need the essentially equivalent notions of *discrete valuations* and *discrete valuation rings*.

A *discrete valuation* of a function field F/K is a map $v : F \to \mathbb{Z} \cup \{\infty\}$ satisfying

1) $v(z) = 0$ for $z \in K^\times$, $v(x) = \infty \Leftrightarrow x = 0$

2) $v(xy) = v(x) + v(y)$

3) $v(x + y) \geq \min\{v(x), v(y)\}$ (*triangle inequality*)

for all x, $y \in F$.

It can be shown that a discrete valuation satisfies

$$v(x + y) = \min\{v(x), v(y)\}$$

for x, $y \in F$ with $v(x) \neq v(y)$ (*strict triangle inequality*, see [Sti93], Lemma I.1.10). Property 2) implies that $v(F^\times)$ is an ideal of $\mathbb{Z}$, i.e. of the form $m\mathbb{Z}$ with $m \in \mathbb{Z}$. Excluding the trivial case $m = 0$ we may replace v by $\frac{1}{m}v$ and assume that $m = 1$.

A *discrete valuation ring* of a function field F/K is a local ring $\mathcal{O}$ which is not a field, which contains K, whose field of fractions is F and whose maximal ideal $\mathfrak{P}$ is principal. Clearly, discrete valuation rings and their maximal ideals are in a one-to-one correspondence: Each discrete valuation ring $\mathcal{O}$ has exactly one maximal ideal $\mathfrak{P}$; conversely, $\mathcal{O}$ is the ring of multipliers of $\mathfrak{P}$, i.e., $\mathcal{O} = \{z \in F : z\mathfrak{P} \subseteq \mathfrak{P}\}$. On the other hand, a discrete valuation ring $\mathcal{O}$ with maximal ideal $\mathfrak{P}$ defines a discrete valuation by $v_\mathfrak{P}(x) = \max\{\nu : x \in \mathfrak{P}^\nu\}$ and $v_\mathfrak{P}(\frac{x}{y}) = v(x) - v(y)$ for x, $y \in \mathcal{O}$. Conversely, to a discrete valuation v corresponds the discrete valuation ring $\mathcal{O} = \{z \in F : v(z) \geq 0\}$ with maximal ideal $\mathfrak{P} = \{z \in F : v(z) > 0\}$. A generator of $\mathfrak{P}$ is an element $t \in F$ with $v(t) = 1$; it is called a *uniformising* or *local parameter* for v.

The above considerations show that the notions of discrete valuations, discrete valuation rings and maximal ideals of discrete valuation rings are basically interchangeable. Indeed, the term "place" is used in the literature to designate one of the above according to the author's likings (cf. [Sch31, Eic63, Sti93]). From the next section on, we will often use the additional term "*prime divisor*".

If K is algebraically closed and C a projective curve defined over K, then each non-singular point P on C defines a valuation of its function field F. Indeed, it can be verified that the set $\{z \in F : z(P) = 0\}$ of functions with *zeroP* is the maximal ideal of a valuation ring, so that the above mentioned correspondence gives rise to a valuation v_P of F satisfying

$$v_P(z) > 0 \text{ for } z \in F \text{ with } z(P) = 0$$

and

$$v_P(z) < 0 \text{ for } z \in F \text{ with } \frac{1}{z}(P) = 0,$$

i.e. for a function z which is *not defined* at P or of which P is a *pole* ([Ful69], Section 3.2). The valuation can be made explicit by determining a local parameter t for P and writing a function z as $z = t^\nu \frac{z_1}{z_2}$ with z_1, $z_2 \in F$ of which P is neither a zero nor a pole and

$\nu \in \mathbb{Z}$. Then $v_P(z) = \nu$. Conversely, if C is smooth, then each valuation of F arises in this way ([Ful69], Section 7.1). If K is not algebraically closed, then the valuations of F are in a one-to-one correspondence with the closed points of C (see Section 2.1.3).

Examples.

1) Let K be algebraically closed, and consider the rational function field $K(X)$. It is the function field of the *projective line* Y. As the Y–coordinate of any point on the line is zero, it may be omitted, and the points on the line are given by $(x : 1)$ with $x \in K$ and $\infty = (1 : 0)$. The finite point $(x : 1)$ corresponds to the valuation v_x with local parameter $X - x$: For $r \in K(X)$, write $r = (X - x)^\nu \frac{f}{g}$ with $\nu \in \mathbb{Z}$, f, $g \in K[X]$ such that $f(x)$, $g(x) \neq 0$. Then $v_x(r) = \nu$. The infinite point corresponds to the degree valuation v_∞ with local parameter $\frac{1}{x}$, such that $v_\infty(r) = -\deg r = \deg g - \deg f$ for $r = \frac{f}{g}$, f, $g \in K[X]$. (In later sections, we will often identify the infinite valuation v_∞ with the infinite point ∞ to simplify the notation.)

2) Now let $K = \mathbb{F}_q$ be finite. The K–rational points on the projective line are again given by $(x : 1)$ with $x \in K$ and ∞. Further closed points are sets $\{(x_0 : 1), \ldots, (x_{k-1} : 1)\}$ with $x_i = x_0^{q^i}$, and x_0 lies in $\mathbb{F}_{q^k}$, but in no subfield of $\mathbb{F}_{q^k}$. Then $p = \prod_{i=0}^{k-1}(X - x_i)$ is the minimal polynomial of x_0 over K and induces the valuation v_p with local parameter p, i.e. $v_p(r) = \nu$ if $r = p^\nu \frac{f}{g}$ with $\nu \in \mathbb{Z}$ and f, $g \in K[X]$ such that $p \nmid f, g$.

□

2.2.3 Field of constants

Let v be a valuation of F/K. By definition, v vanishes on the constants in K. Furthermore, v vanishes on the algebraic closure $\overline{K} \cap F$ of K in F: Let $z \in F$ be algebraic over K, and assume $v(z) < 0$. (If $v(z) > 0$, consider $\frac{1}{z}$ instead, since $v(\frac{1}{z}) = -v(z)$.) There is a monic equation $\sum_{i=0}^{n} x_i z^i = 0$ with $x_i \in K$ and $x_n = 1$ satisfied by z. As $v(x_i z^i) = iv(z) < jv(z) = v(x_j z^j)$ for $i > j$ and x_i, $x_j \neq 0$, the strict triangle inequality implies $v\left(\sum_{i=0}^{n} x_i z^i\right) = nv(z)$, a contradiction to $v(0) = \infty$.

So from a valuation theoretic point of view, $\overline{K} \cap F$ cannot be distinguished from K, and $\overline{K} \cap F$ is called the (exact) *constant field* of K. (It can be shown that for $z \in F \backslash \overline{K}$ there is a valuation v with $v(z) \neq 0$, which justifies the additional "exact".) The constant field is a finite and, since K is assumed to be perfect, a separable extension of the field of definition K.

If $F = K(C)$ is the function field of a plane curve C, then K is the field of constants of F if and only if C is absolutely irreducible (see [Sti93], Proposition III.6.6). In the remainder of this chapter, we assume this case.

2.2.4 Residue class field

Let $\mathcal{O}$ be a discrete valuation ring of F/K with maximal ideal $\mathfrak{P}$. The image $F_{\mathfrak{P}} = \mathcal{O}/\mathfrak{P}$ of the reduction map $\mathcal{O} \to F_{\mathfrak{P}}$, $x \mapsto x + \mathfrak{P}$, is called the *residue class field* of $\mathfrak{P}$. By the characterisation of the constant field of F we have that $K \subseteq F_{\mathfrak{P}}$, and in fact $F_{\mathfrak{P}}$ is a finite extension of K ([Sti93], Proposition I.1.14); its degree is denoted by $\deg \mathfrak{P}$ and called the degree of $\mathfrak{P}$. If K is algebraically closed, then $\deg \mathfrak{P} = 1$; otherwise, if C is a smooth model for F, the degree of $\mathfrak{P}$ is the same as the degree of the closed point corresponding to $\mathfrak{P}$ defined in Section 2.1.3 ([Sti93], Theorem III.6.3 (c)).

2.2.5 Decomposition law

The valuations of the rational function field $K(X)$ are well known from the example in Section 2.2.2. Any function field is a finite extension of $K(X)$, so it is of interest to examine the behaviour of valuations in finite extensions of function fields. The following results are not specific to function fields; they hold in any Dedekind ring. For proofs, see [Sti93], Section III.1.

Let F'/K be a finite extension of F/K with the same constant field K, and let $v_{\mathfrak{p}}$ be a valuation of F with discrete valuation ring $\mathcal{O}_{\mathfrak{p}}$ and maximal ideal $\mathfrak{p}$. Then there are finitely many discrete valuation rings of F' containing $\mathcal{O}_{\mathfrak{p}}$; their maximal ideals $\mathfrak{P}_1, \ldots, \mathfrak{P}_r$ satisfy $\mathfrak{P}_i \cap F = \mathfrak{p}$. The $\mathfrak{P}_i$ are called the *extensions of* $\mathfrak{p}$ or the *prime ideals above* $\mathfrak{p}$. If t is a local parameter for $v_{\mathfrak{p}}$, then $v_{\mathfrak{P}_i}(t)$ is a positive integer e_i, called the *ramification index* of $\mathfrak{P}_i/\mathfrak{p}$. Letting $\mathcal{O}_{\mathfrak{P}} = \bigcap_{i=1}^{r} \mathcal{O}_{\mathfrak{P}_i}$, the integral closure of $\mathcal{O}_{\mathfrak{p}}$ in F' ([Sti93], Corollary III.3.5), the ideals $\mathfrak{P}_i \cap \mathcal{O}_{\mathfrak{P}}$ satisfy $\mathfrak{p}\mathcal{O}_{\mathfrak{P}} = \prod_{i=1}^{r} (\mathfrak{P}_i \cap \mathcal{O}_{\mathfrak{P}})^{e_i}$, whence it is common to write $\mathfrak{P}_i | \mathfrak{p}$. The residue class field $F_{\mathfrak{P}_i}$ is a finite extension of $F_{\mathfrak{p}}$; the degree $f_i = [F_{\mathfrak{P}_i} : F_{\mathfrak{p}}]$ is called the *inertia degree* of $\mathfrak{P}_i/\mathfrak{p}$.

Extensions $\mathfrak{P}_i/\mathfrak{p}$ are called *ramified* if $e_i > 1$, *unramified* otherwise. The prime divisor $\mathfrak{p}$ is called ramified if some of its extensions is. Similarly, $\mathfrak{P}_i$ is called *inert* if $f_i > 1$, and $\mathfrak{p}$ is called inert if some of its extensions is. A prime divisor $\mathfrak{p}$ which is neither ramified nor inert is called *completely splitting*.

The most important fact of this section is the *decomposition law*

$$\sum_{i=1}^{r} e_i f_i = [F' : F].$$

If $F' : F$ is Galois, then its Galois group acts transitively on the prime divisors above $\mathfrak{p}$. Consequently, $e_i = e$ and $f_i = f$ are independent of i, and the decomposition law has the simpler form

$$ref = [F' : F].$$

2.2.6 Number fields

Number fields are of interest in our context as they provide further examples of groups to which the algorithms of Chapters 6 and 7 may be applied. In fact, certain number fields have been suggested for use in cryptography ([BW88]).

A *number field* is a finite algebraic extension F of the rational numbers $\mathbb{Q}$. Number and function fields are both fields of fractions of Dedekind rings; so it suffices to provide a dictionary between structures in number and function fields to see that the theory of function fields developed so far carries over to number fields.

The ring of integers $\mathbb{Z}$ corresponds to the ring of polynomials $K[X]$, its field of fractions $\mathbb{Q}$ to the rational function field $K(X)$. The integers carry a valuation theory analogous to $K[X]$. Each prime number p defines a discrete valuation $v_\mathfrak{p}$ via $v_\mathfrak{p}(r) = \nu$ if $r = p^\nu \frac{f}{g} \in \mathbb{Q}$ with $\nu \in \mathbb{Z}$, f, $g \in \mathbb{Z}$ and $p \nmid fg$. Let $\mathcal{O}_\mathfrak{p} = \left\{ \frac{f}{g} : p \nmid g \right\}$, then the residue class field of $v_\mathfrak{p}$ is $\mathcal{O}_\mathfrak{p}/\mathfrak{p} \simeq \mathbb{Z}/p\mathbb{Z}$. Denoting by $\mathcal{O}_\mathfrak{P}$ the integral closure of $\mathcal{O}_\mathfrak{p}$ in F, we have $\mathfrak{p}\mathcal{O}_\mathfrak{P} = \prod_{i=1}^{r} \mathfrak{P}_i^{e_i}$ with ideals $\mathfrak{P}_i$ of $\mathcal{O}_\mathfrak{P}$, and f_i is defined by $f_i = [\mathcal{O}_\mathfrak{P}/\mathfrak{P}_i : \mathbb{Z}/p\mathbb{Z}]$. Then the decomposition law $[F : Q] = \sum_{i=1}^{r} e_i f_i$ holds.

In number fields, it is common to replace this local theory by a global one: Let $\mathcal{O}$ be the integral closure of $\mathbb{Z} = \bigcap_{v_\mathfrak{p} \text{ valuation of } \mathbb{Q}} \mathcal{O}_\mathfrak{p}$ in F; then $\mathfrak{p}\mathcal{O} = \prod_{i=1}^{r} (\mathfrak{P}_i \cap \mathcal{O})^{e_i}$, and $\mathcal{O}_\mathfrak{P}/\mathfrak{P}_i \simeq \mathcal{O}/(\mathfrak{P}_i \cap \mathcal{O})$. We examine the corresponding situation for function fields, in which $K[X] = \bigcap_{v_\mathfrak{p} \neq v_\infty \text{ valuation of } K(X)} \mathcal{O}_\mathfrak{p}$, more closely in Section 2.4, where we also briefly discuss the analogue of the infinite valuations in the number field case.

2.3 Divisors and Jacobians

The classical group associated with a number field is its *class group*, a finite abelian group defined as the quotient of the fractional ideals of the number field by the principal ideals. In this section, we define the class group of a function field and show that it is a finite abelian group if the constant field is finite.

2.3.1 Divisors

Let F/K be a function field. Its *group of divisors* is the free abelian group over its discrete valuations,

$$\mathrm{Div}(F/K) = \left\{ \sum m_\mathfrak{P} \mathfrak{P} : \text{almost all } m_\mathfrak{P} \text{ equal zero} \right\},$$

where the sum is taken over all maximal ideals $\mathfrak{P}$ of discrete valuation rings of F. For function fields it is common to use additive notation; however, the divisor group is the

exact analogue of the multiplicatively written group of fractional ideals of a number field. The *degree* of a divisor $D = \sum m_{\mathfrak{P}}\mathfrak{P}$ is

$$\deg D = \sum m_{\mathfrak{P}} \deg \mathfrak{P}.$$

As the maximal ideals $\mathfrak{P}$ are the irreducible components of a divisor, we call them *prime divisors*.

To an element $z \in F$, we may associate its *principal divisor*

$$\operatorname{div}(z) = \sum v_{\mathfrak{P}}(z)\mathfrak{P},$$

its *zero divisor*

$$\operatorname{div}_0(z) = \sum_{\mathfrak{P}: v_{\mathfrak{P}}(z)>0} v_{\mathfrak{P}}(z)\mathfrak{P}$$

and its *pole divisor*

$$\operatorname{div}_\infty(z) = \sum_{\mathfrak{P}: v_{\mathfrak{P}}(z)<0} -v_{\mathfrak{P}}(z)\mathfrak{P}.$$

It turns out that $\operatorname{div}_0(z)$ and $\operatorname{div}_\infty(z)$ are indeed divisors, i.e. that any function has only finitely many zeroes and poles. Moreover, counting multiplicities appropriately, a function has as many zeroes as poles; precisely,

$$\deg(\operatorname{div}_0(z)) = \deg(\operatorname{div}_\infty(z)) = [F : K(z)]$$

([Sti93], Theorem I.4.11). Thus, the group of principal divisors $\operatorname{Prin}(F/K)$ is a subgroup of the group $\operatorname{Div}^0(F/K)$ of divisors of degree zero. The quotient group

$$J(F/K) = \operatorname{Div}^0(F/K)/\operatorname{Prin}(F/K)$$

is called the *divisor class group* or *Jacobian* of F/K. We write $D \sim D'$ if D and D' are two degree zero divisors of the same class. (Sometimes, the divisor class group is defined as $\operatorname{Div}(F/K)/\operatorname{Prin}(F/K)$. This does not constitute a major difference since $\operatorname{Div}(F/K)/\operatorname{Prin}(F/K) \simeq J(F/Q) \times \mathbb{Z}$.)

2.3.2 Riemann–Roch Theorem

For a divisor $D \in \operatorname{Div}(F/K)$ let $\mathcal{L}(D)$ denote the subset of F consisting of the functions z with $\operatorname{div}(z) \geq -D$, where the comparison of divisors is componentwise. From the definition of a discrete valuation follows that $\mathcal{L}(D)$ is a vector space over K; moreover, its dimension $l(D)$ is finite ([Ful69], Section 8.2; [Sti93], Proposition I.4.9). The Riemann–Roch Theorem relates $l(D)$ and $\deg D$. In fact, for our purposes Riemann's weaker theorem is sufficient; it states that there is a constant g such that

$$l(D) \geq \deg D + 1 - g$$

for all $D \in \mathrm{Div}(F/K)$. The smallest such g is a non-negative integer, called the *genus* of F/K ([Ful69], Section 8.3; [Sti93], Theorem I.4.17). For instance, the rational function field $K(X)/K$ has genus zero ([Ful69], Section 8.3; [Sti93], Example I.4.18).

For the sake of completeness, we also quote the Riemann–Roch Theorem, which provides the exact difference between $l(D)$ and $\deg D + 1 - g$. The main assertion of the theorem is that there exists a divisor W of degree $2g - 2$ (called a *canonical divisor*) such that

$$l(D) = \deg D + 1 - g + l(W - D)$$

for all $D \in \mathrm{Div}(F/K)$ ([Ful69], Section 8.6; [Sti93], Theorem I.5.15). The proof of the Riemann–Roch Theorem is considerably more technical than that of Riemann's theorem, requiring the notion of a differential of a function field.

2.3.3 Finiteness of the Jacobian

Assume that $K = \mathbb{F}_q$ is a finite field; we wish to verify that $J = J(F/K)$ is a finite group. Let g be the genus of F/K. As a matter of fact, J is a g–dimensional algebraic variety, which implies its finiteness. But the variety structure of J is too complicated for large g to be of use in computations since J is embedded in some very high dimensional space. (However, in the case of *elliptic curves*, i.e. for $g = 1$, the Jacobian is isomorphic to the curve itself, and it is efficient to compute the group law by algebraic formulae.)

Instead, our reasoning follows the definition of the Jacobian as a quotient group of the degree zero part of the divisor group. Let D_1 be a fixed divisor of degree 1, which exists by a theorem of Schmidt's ([Sch31], § 8; [Sti93], Corollary V.1.11). (Anyway, we will restrict ourselves to curves with a prime divisor of degree 1 in later sections.) We proceed by showing that any divisor class contains a divisor of the special form $D' - gD_1$ with an *effective* or *positive* divisor D', i.e. a divisor $D' \geq 0$. Let D be a degree zero divisor. By Riemann's theorem, $l(D + gD_1) \geq 1$. Choose $z \in \mathcal{L}(D + gD_1)$ and let $D' = D + gD_1 + \mathrm{div}(z)$, so that $D \sim D' - gD_1$. By definition of $\mathcal{L}$, we have $\mathrm{div}(z) \geq -D - gD_1$ so that $D' \geq 0$. Necessarily, the degree of D' is g, so that D' is composed of prime divisors of degree at most g. Since such prime divisors extend valuations of $K(X)$ induced by irreducible polynomials of degree at most g and each such valuation has only finitely many extensions (see Section 2.2.5), there are only finitely many possibilities for D', and $J(F/K)$ is finite. The *divisor class number* of F is denoted by $h = |J(F/K)|$. Section 2.5 deals with a closer estimate for h than the one which could be obtained by the reasoning above.

2.4 Ideal classes and regulator

2.4.1 Ideal class groups

It has been mentioned above that divisor class groups of function fields F/K are closely related to ideal class groups of number fields. In fact, the analogy falls short, as divisors in function fields do not correspond to ideals in some ring: A prime divisor $\mathfrak{P}$ is the maximal ideal of some discrete valuation ring, and different prime divisors are ideals of different rings. To be able to develop an ideal theory, we have to designate a ring in F. Remember that the corresponding ring in the number field case is the integral closure of $\mathbb{Z}$ in the number field. Assume that $F/K = K(C)$ is the function field of some affine plane curve C. Then the analogue of $\mathbb{Z}$ is $K[X]$, and we may consider the ring $\mathcal{O}$ as the integral closure of $K[X]$ in $K(C)$. Since $K[X]$ is the intersection of all valuation rings $\mathcal{O}_\mathfrak{p}$ of $K(X)$ corresponding to finite valuations (i.e. valuations different from ∞), the ring $\mathcal{O}$ is the intersection of all valuation rings $\mathcal{O}_\mathfrak{P}$ of F/K extending some finite valuation of $K(X)$ ([Sti93], Theorem III.2.6). Its field of fractions is F.

The finite valuations of F/K (i.e. the valuations not extending ∞) are in a one-to-one correspondence with the prime ideals of $\mathcal{O}$. Precisely, the maximal ideal $\mathfrak{P}$ of a finite valuation of F/K corresponds to the prime ideal $\mathfrak{P} \cap \mathcal{O}$ of $\mathcal{O}$. Recall the notion of a *Dedekind ring*, which is an integral domain in which all (fractional) ideals can be written as a unique product of prime ideals (with possibly negative multiplicities). For instance, the principal ideal domain $K[X]$ is a Dedekind ring. Thus, $\mathcal{O}$ is the integral closure of a Dedekind ring in a finite algebraic extension, namely $K(C)$, whence it inherits the Dedekind property. Consequently, a fractional ideal $\mathfrak{a}$ of $\mathcal{O}$ admits a unique decomposition into a product of prime ideals $\mathfrak{a} = \prod_{\mathfrak{P}\nmid\infty} (\mathfrak{P} \cap \mathcal{O})^{\nu_\mathfrak{P}}$, and the group of fractional ideals of $\mathcal{O}$ is in fact the free, multiplicatively written abelian group over the prime ideals $\mathfrak{P} \cap \mathcal{O}$. We associate to the ideal $\mathfrak{a}$ the divisor $\operatorname{div}(\mathfrak{a}) = \sum_{\mathfrak{P}\nmid\infty} \nu_\mathfrak{P}\mathfrak{P}$. (The distinction between an ideal and its divisor may seem awkward at first, but it has several advantages. We need the multiplicative notation in later sections when dealing with concrete representations of ideals, for which addition is already defined differently. However, by reverting completely to multiplicative notation, we would lose the analogy to the divisor group; moreover, as the ideals $\mathfrak{P}$ lie in different domains, we would always have to take care to write "$\mathfrak{P} \cap \mathcal{O}$", which in turn would constitute an unpleasant notational overhead.)

A fractional ideal of $\mathcal{O}$ is *integral* if and only if its divisor is positive; so a divisor in $\operatorname{Div}(F/K)$ belongs to an integral ideal of $\mathcal{O}$ if and only if its is positive and does not contain an infinite prime divisor.

Let $\mathcal{I}(\mathcal{O})$ be the group of divisors of fractional ideals of $\mathcal{O}$, i.e. the free abelian group over the finite valuations of F/K. To an element $z \in F$ can be associated the divisor of its *principal ideal* $\sum_{\mathfrak{P}\nmid\infty} v_\mathfrak{P}(z)(\mathfrak{P} \cap \mathcal{O})$. If $\operatorname{Prin}(\mathcal{O})$ denotes the subgroup of $\mathcal{I}(\mathcal{O})$ formed by the divisors of principal ideals, then the *ideal class group* of $\mathcal{O}$ is the abelian group $\mathfrak{H}(\mathcal{O}) = \mathcal{I}(\mathcal{O})/\operatorname{Prin}(\mathcal{O})$.

Thus by examining $\mathcal{O}$, we lose exactly the information on the infinite valuations of F/K. So $\mathcal{O}$ describes the valuation theory of the affine part of the curve C. Note that the distinction between "finite" and "infinite" valuations of F/K is not intrinsic to the function field; it arises only when we fix a specific affine model C for F. For instance, dehomogenising the projective closure of C with respect to a different projective line than Z (which corresponds to regarding F as a finite algebraic extension of $K(X')$ for $X' \in F$ different from X) results in a different ring $\mathcal{O}$.

2.4.2 Regulator

Assume again that F/K is the function field of some affine curve C and that $\mathcal{O}$ is the integral closure of $K[X]$ in F. Let $\infty_1, \ldots, \infty_r$ be the distinct extensions of ∞ in F, so that the pole divisor of X is $\operatorname{div}_\infty(X) = \sum_{i=1}^r \infty_i^{e_i}$, and denote by $\operatorname{Div}_\infty(\mathcal{O})$ the set of divisors composed of only infinite prime divisors, i.e., $\operatorname{Div}_\infty(\mathcal{O}) = \mathbb{Z}\infty_1 + \cdots + \mathbb{Z}\infty_r$. Let $\operatorname{Div}^0_\infty(\mathcal{O})$ be the degree zero part of $\operatorname{Div}_\infty(\mathcal{O})$, and consider the projection map

$$\pi_{\mathcal{O}} : \operatorname{Div}^0(F/K) \to \mathcal{I}(\mathcal{O}), \quad \sum_{\mathfrak{P}} m_{\mathfrak{P}}\mathfrak{P} \mapsto \sum_{\mathfrak{P}\nmid\infty} m_{\mathfrak{P}}\mathfrak{P}.$$

Suppose that $\operatorname{Div}_\infty(\mathcal{O})$ contains a divisor D_1 of degree 1, which is the case if and only if the greatest common divisor of the inertia degrees of the ∞_i is 1. (In practice, we will assume that there is some infinite prime divisor which is not inert, so this is no serious restriction.) Since the preimage of $\sum_{\mathfrak{P}\nmid\infty} m_{\mathfrak{P}}\mathfrak{P}$ under $\pi_{\mathcal{O}}$ contains $\sum_{\mathfrak{P}\nmid\infty} m_{\mathfrak{P}}\mathfrak{P} - \left(\sum_{\mathfrak{P}\nmid\infty} m_{\mathfrak{P}} \deg \mathfrak{P}\right) D_1$, the map $\pi_{\mathcal{O}}$ is surjective and induces the short exact sequence

$$0 \to \operatorname{Div}^0_\infty(\mathcal{O}) \to \operatorname{Div}^0(F/K) \xrightarrow{\pi_{\mathcal{O}}} \mathcal{I}(\mathcal{O}) \to 0.$$

Restricting to principal divisors and ideals, we obtain the exact sequence

$$0 \to \operatorname{Div}^0_\infty(\mathcal{O}) \cap \operatorname{Prin}(F/K) \to \operatorname{Prin}(F/K) \xrightarrow{\pi_{\mathcal{O}}} \operatorname{Prin}(\mathcal{O}) \to 0,$$

from which we deduce the exact sequence

$$0 \to \operatorname{Div}^0_\infty(\mathcal{O})/\operatorname{Div}^0_\infty(\mathcal{O}) \cap \operatorname{Prin}(F/K) \to J(F/K) \to \mathfrak{H}(\mathcal{O}) \to 0$$

by taking quotients. Thus, the divisor class group $J = J(F/K)$ is isomorphic to the product of the ideal class group $\mathfrak{H}(\mathcal{O})$ and $\operatorname{Div}^0_\infty(\mathcal{O})/\operatorname{Div}^0_\infty(\mathcal{O}) \cap \operatorname{Prin}(F/K)$. In particular, for K a finite field, the finiteness of J implies that the *ideal class number* $|\mathfrak{H}(\mathcal{O})|$ and the *regulator* $R(\mathcal{O}) = |\operatorname{Div}^0_\infty(\mathcal{O})/\operatorname{Div}^0_\infty(\mathcal{O}) \cap \operatorname{Prin}(F/K)|$ of $\mathcal{O}$ are finite.

The regulator has a natural interpretation in terms of the units $\mathcal{O}^\times$ of $\mathcal{O}$. Note that

$$\mathcal{O}^\times = \{\varepsilon \in F : v_{\mathfrak{P}}(\varepsilon) = 0 \text{ for all } \mathfrak{P} \nmid \infty\}.$$

Thus,

$$\mathrm{Div}^0_\infty(\mathcal{O}) \cap \mathrm{Prin}(F/K) = \mathrm{div}(\mathcal{O}^\times) = \{\mathrm{div}(\varepsilon) : \varepsilon \in \mathcal{O}^\times\}.$$

Assume now that at least one infinite prime divisor is not inert, say $\deg \infty_r = 1$. Then it is easy to show that the group $\mathrm{Div}^0_\infty(\mathcal{O})$ is a free $\mathbb{Z}$–module of rank $r-1$, since $\mathbb{Z}^{r-1} \to \mathrm{Div}^0_\infty(\mathcal{O})$, $(m_1, \ldots, m_{r-1}) \mapsto \sum_{i=1}^{r-1} m_i \infty_i - \left(\sum_{i=1}^{r-1} m_i \deg \infty_i\right) \infty_r$ is an isomorphism. $\mathrm{Div}^0_\infty \cap \mathrm{Prin}(F/K)$ is a submodule of Div^0_∞ of finite index. By the Elementary Divisor Theorem, there are a basis $(D_1, \ldots, D_{r-1})$ of $\mathrm{Div}^0_\infty(\mathcal{O})$ and unique positive integers $d_1 | \ldots | d_{r-1}$ such that $(d_1 D_1, \ldots, d_{r-1} D_{r-1})$ is a basis of $\mathrm{Div}^0_\infty(\mathcal{O}) \cap \mathrm{Prin}(F/K)$. Choose ε_i such that $\mathrm{div}(\varepsilon_i) = d_i D_i$; then $R(\mathcal{O}) = d_1 \cdots d_{r-1}$ and $\{\varepsilon_1, \ldots, \varepsilon_{r-1}\}$ is a set of *fundamental units* of $\mathcal{O}$, i.e., $\mathcal{O}^\times = K^\times \times \langle \varepsilon_1 \rangle \times \cdots \times \langle \varepsilon_{r-1} \rangle$.

We note that the situation is similar to that in number fields F, in which the embeddings $\sigma : F \to \mathbb{C}$ provide the analogue of the infinite valuations. (More precisely, $|\sigma(\cdot)|$ defines an absolute value on F which extends the usual absolute value $|\cdot|$ on $\mathbb{Q}$, and the function $\log |\sigma(\cdot)| : F \to \mathbb{R} \cup \{\infty\}$ replaces the infinite valuation. The situation is complicated by the fact that $|\sigma(\cdot)|$ is archimedian, whence it does not correspond to a discrete valuation.)

2.4.3 Ideal class number versus regulator

We have shown in Section 2.4.2 that the Jacobian J of a function field F in which at least one infinite prime divisor is not inert over a finite field K can be seen as the product of two groups, the ideal class group, which describes the affine part of J, and the group $\mathrm{Div}^0_\infty(\mathcal{O}) / \mathrm{Div}^0_\infty(\mathcal{O}) \cap \mathrm{Prin}(F/K)$, which describes the part of J "at infinity" and which is of size R.

As seen in Chapter 1, the group J is only suited for use in cryptography if the class number $h = |J|$ has a large prime factor. Since it is simpler to compute in $\mathfrak{H}(\mathcal{O})$ than in J, one usually substitutes $\mathfrak{H}(\mathcal{O})$ for J. If the regulator R is small, then the large prime factor of h will also occur in $|\mathfrak{H}(\mathcal{O})|$ since

$$h = |\mathfrak{H}(\mathcal{O})| R.$$

Thus, one does not lose any security when switching to $\mathfrak{H}(\mathcal{O})$ in this case. An especially favourable situation arises when ∞ is completely ramified in F, so that $r = 1$ and $R = 1$ in the notation of Section 2.4.2. Then $\mathfrak{H}(\mathcal{O})$ is in fact isomorphic to J.

On the other hand, when R is large, the ideal class number $|\mathfrak{H}(\mathcal{O})|$ will usually be small, so $\mathfrak{H}(\mathcal{O})$ does not allow to build secure cryptosystems. From a geometric point of view, one might argue that one has simply chosen an inappropriate affine model of the function field. Indeed, it is often possible to select the line at infinity differently, so that the new ideal class group "captures" the large prime factor of h. However, it may be necessary to additionally extend the field of constants, i.e. to switch to FK'/K' with a finite algebraic extension K'/K, to ensure that an infinite prime divisor is not inert (cf. the discussions in Sections 3.1.1 and 3.2.1).

2.5 Weil's theorem

One of the conclusions of Chapter 1 was that a group must be sufficiently large to provide secure cryptosystems. So it is of interest to derive bounds which estimate the order of magnitude of a Jacobian of given genus g over a finite field $K = \mathbb{F}_q$ with q elements. Most results of this section are given without proof, see [Sti93], Chapter V, for further details.

In a first step, the numbers of divisors of different degrees are coded into one object, the *zeta function* of F/K. If A_k denotes the number of positive divisors of F/K of degree k, then the zeta function of F/K is given by

$$Z(t) = \sum_{k=0}^{\infty} A_k t^k.$$

(It may be regarded as a formal power series in characteristic zero or a power series over $\mathbb{C}$ since it converges for $|t| < \frac{1}{q}$.)

Example. Let $F = K(X)$ be rational. Note that then $h = 1$: $K[X]$ is a principal ideal domain, so that its ideal class number is 1. Furthermore, there is exactly one infinite prime divisor, namely ∞, which is of degree 1, so that $R = 1$. As h is the product of the ideal class number and the regulator (see Section 2.4.3), the claim follows.

To estimate A_k, let D be a positive divisor of degree k in which ∞ occurs with multiplicity $l \in \{0, \ldots, k\}$. Then $D - k\infty$ is of degree zero and thus a principal divisor $\mathrm{div}(z)$ for some $z \in F^\times$. Since the finite part of $\mathrm{div}(z)$ is positive and of degree $k - l$, the function z is in fact a polynomial of degree $k - l$. The only elements of F with divisor zero are the constants, so z is defined up to multiplication by elements of $K^\times$ and may be assumed to be monic. So there are q^{k-l} possibilities for D with the above properties. Summing up for $l = 0, \ldots, k$, we have shown that $A_k = \frac{q^{k+1}-1}{q-1}$. Thus,

$$\begin{aligned} Z(t) &= \sum_{k=0}^{\infty} \frac{q^{k+1}-1}{q-1} t^k = \frac{1}{q-1}\left(q\sum_{k=0}^{\infty}(qt)^k - \sum_{k=0}^{\infty} t^k\right) \\ &= \frac{1}{q-1}\left(\frac{q}{1-qt} - \frac{1}{1-t}\right) = \frac{1}{(1-t)(1-qt)}. \end{aligned}$$

□

In general, one counts the number of positive divisors in each class separately. The calculations are similar, since by a result of Schmidt's ([Sch31], § 8; [Sti93], Corollary V.1.11) a function field over a finite field contains a prime divisor of degree 1, which may play the role of ∞. It turns out that $Z(t)$ is a rational function in $\mathbb{Q}(t)$ with denominator $(1 - t)(1 - qt)$; its numerator is called the *L–polynomial* of F/K

and denoted by $L(t)$. The L–polynomial is of degree $2g$, and its value at the place 1 equals h. Furthermore, the number A_1 of prime divisors of degree 1, i.e. of points on the smooth projective model, can be derived from the coefficient of the linear term of $L(t)$: From $L(t) = (1-t)(1-qt)\sum_{k=0}^{\infty} A_k t^k$ we deduce that this coefficient is $A_1 - (q+1)A_0 = A_1 - (q+1)$.

Since $L(0) = Z(0) = A_0 = 1$, the L–polynomial can be written as $L(t) = \prod_{i=1}^{2g}(1-\alpha_i t)$, where the α_i are its *reciprocal roots*. Weil was able to prove the analogue of the Riemann hypothesis in this case, which states that $|\alpha_i| = \sqrt{q}$ ([Wei48]). A simpler proof is due to Bombieri ([Bom74]), and the case of genus 1 was solved by Hasse in [Has33, Has34]; this is why the theorem is also known as "Hasse–Weil Theorem". From the properties of the L–polynomial mentioned above we immediately deduce bounds on the class number h and on the number A_1 of points on a smooth projective curve of genus g over $\mathbb{F}_q$:

$$h = L(1) = \prod_{i=1}^{2g} |1-\alpha_i|,$$

so that

$$(\sqrt{q}-1)^{2g} \leq h \leq (\sqrt{q}+1)^{2g}.$$

$$\begin{aligned} |A_1 - (q+1)| &= |\text{coefficient of the linear term of } L(t)| \\ &= \left| \sum_{i=1}^{2g} \alpha_i \right| \leq \sum_{i=1}^{2g} |\alpha_i| = 2g\sqrt{q} \end{aligned}$$

For the cryptographic applications it is sufficient to keep in mind that the size of the Jacobian of a curve of genus g over the field with q elements is about q^g. If a desired level of security, i.e. an approximate size of q^g, is fixed, one may select curves of higher genus and at the same time lower the field size, or vice versa. The question which combination of parameters yields the fastest arithmetic for hyperelliptic curves will be addressed in Chapter 4.

2.6 Cyclic extensions

The only concrete example of a function field presented so far, the rational function field, has a trivial Jacobian group, and is clearly not of interest in the cryptographic context. Any other function field is a finite extension of the rational function field. In this section, we examine more closely the simplest extensions, namely the cyclic ones. A *cyclic extension* of a field is defined as a Galois extension with cyclic Galois group. If its degree is prime and the ground field contains sufficiently many roots of unity, it is necessarily of Kummer or Artin–Schreier type. Cyclic extensions of a function field

over a perfect field were studied by Hasse in [Has35], who generalised the quadratic case examined by Artin in [Art24a, Art24b]. The theory developed by Hasse is valid for any relative cyclic extension of function fields; however, we restrict the presentation to "cyclic coverings of the projective line", i.e. cyclic extensions of the rational function field, which are of primary interest to us.

Let K be a perfect field of characteristic q a prime or zero, and n the desired degree of the extension.

2.6.1 Kummer extensions

Assume that q is zero or does not divide n, and that K contains the n-th roots of unity. (If K is the finite field $\mathbb{F}_r$, this means that $n|r-1$.) Any cyclic extension of $K(X)$ of degree n is the function field of a curve of the form

$$C = Y^n - u$$

with $u \in K(X)$, and conversely, any such curve generates a cyclic extension of $K(X)$. (Strictly speaking, C is not a curve in the sense of Section 2.1, as u need not be a polynomial. However, the field theoretic approach remains unchanged, and so does the geometry, unless we try to speak of points whose X–coordinates are poles of u. In any case, it is possible to multiply such a polynomial $C \in K(X)[Y]$ by the least common denominator of its coefficients to obtain a curve in the strict sense. This may, however, introduce singularities. We treat the necessary transformations for hyperelliptic curves in Sections 3.1.2 and 3.1.3.) The irreducibility of C translates into the condition that $u \neq u_0^d$ for any divisor d of n, $d > 1$ and $u_0 \in K(X)$. Furthermore, C is assumed to be absolutely irreducible, which means that $u \neq a u_0^d$ for any $d|n$, $d > 1$, $u_0 \in K(X)$ and $a \in K$.

In what follows, let y by the image of Y in $K(C)$. The Galois group of $K(C)/K(X)$ is realised by the substitutions $y \mapsto y\zeta^\nu$ for a fixed primitive n-th root of unity $\zeta \in K$ and $\nu \in \{0, \ldots, n-1\}$.

Any other generating element of $K(C)/K(X)$ with a minimal polynomial of the form above is given by $y^a u_0$ with $a \in \mathbb{Z}$, $\gcd(a, n) = 1$ and $u_0 \in K(X)^\times$. Its minimal polynomial is

$$Y^n - u^a u_0^n.$$

Applying such substitutions with $a = 1$ it is possible to eliminate the denominator of u and any occurrence of an n-th power of an irreducible polynomial in the numerator. Thus we may assume that $u = \prod_{i=1}^r p_i^{\nu_i}$ with irreducible polynomials p_i and $\nu_i \in \{1, \ldots, n-1\}$. Let $\mathfrak{p}_i$ denote the prime divisor of $K(X)/K$ with local parameter p_i. Then as a divisor of $K(X)/K$,

$$\operatorname{div}(u) = \sum_{i=1}^r \nu_i \mathfrak{p}_i - \left(\sum_{i=1}^r \nu_i \deg p_i \right) \infty.$$

Our aim is now to describe the decomposition of prime divisors of $K(X)$ in $K(C)$. Proofs for the case $n = 2$ will be given in Section 3.2.2. Denote by $\mathfrak{p} \neq \infty$ a prime divisor of $K(X)$ with local parameter p, by ν the power to which it occurs in $\operatorname{div}(u)$, by e and f its ramification index and inertia degree, and by $K(X)_\mathfrak{p} = K[X]/(p)$ its residue class field. By the results of Section 2.2.5 on Galois extensions, e and f are well-defined divisors of n, and the number of extensions of $\mathfrak{p}$ is $\frac{n}{ef}$. Let $d = \gcd(\nu, n)$ with $d = n$ for $\nu = 0$. Then $e = \frac{n}{d}$, i.e., $\mathfrak{p}$ is ramified if and only if it occurs in $\operatorname{div}(u)$. Its inertia degree is the smallest exponent $f|d$ such that the polynomial $Y^n - u^{ef} \pmod p$ has a root in $K(X)_\mathfrak{p}$.

The decomposition of ∞ is obtained in the same way. Replacing the generating element y of $K(C)/K(X)$ by $y' = \frac{X^i}{y}$ for a suitable value of $i \geq 0$ and u by $u' = \frac{X^{ni}}{u}$, we obtain $v_\infty(u') = -\deg u' \in \{0, \ldots, n-1\}$. Then $e = \frac{n}{\gcd(n, v_\infty(u'))}$, and ∞ is ramified if and only if $v_\infty(u') \neq 0$, i.e., $n \nmid \sum_{i=1}^r \nu_i \deg p_i$. The inertia degree of ∞ is the smallest exponent f such that the polynomial $Y^n - (u')^{ef} \pmod \infty$ has a root in K_∞, which means that there is an element $z \in K(X)$ of non-positive degree such that $\deg(z^n - (u')^{ef}) < 0$.

The genus of $K(C)$ in the case of an absolutely irreducible curve C is

$$\frac{n}{2} \sum_{\mathfrak{p} \text{ ramified}} \left(1 - \frac{1}{e}\right) \deg \mathfrak{p} - (n-1).$$

2.6.2 Artin–Schreier extensions

If the characteristic q is a prime dividing n, then any polynomial of the form $Y^n - u$ is inseparable and thus not Galois. The general form of a cyclic extension of degree q was discovered by Artin and Schreier. Extensions of degree a power of q can be constructed as towers of Artin–Schreier extensions ([AS27] covers the cases of degree q and q^2). Finally, for general n, a cyclic extension is composed of an Artin–Schreier tower of degree q^ν with $q^\nu \| n$ and a Kummer extension of degree $\frac{n}{q^\nu}$. We restrict ourselves to the description of a simple Artin–Schreier extension.

Any cyclic extension of $K(X)$ of degree q is the function field of a curve of the form

$$C = Y^q - Y - u$$

with $u \in K(X)$, and conversely any such curve generates a cyclic extension of $K(X)$. (This "curve" is not a curve in the strict sense; cf. the discussion in Section 2.6.1.) The irreducibility of C translates into the condition that $u \neq u_0^q - u_0$ for any $u_0 \in K(X)$. Furthermore, C is assumed to be absolutely irreducible, which means that $u \neq a + (u_0^q - u_0)$ for any $u_0 \in K(X)$ and $a \in K$.

Denote again by y the image of Y in $K(C)$. The Galois group of $K(C)/K(X)$ is realised by the substitutions $y \mapsto y + \nu$ for $\nu \in \{0, \ldots, q-1\}$.

Any other generating element of $K(C)/K(X)$ with a minimal polynomial of the form above is given by $ay + u_0$ with $a \in \mathbb{Z}$, $\gcd(a, q) = 1$ and $u_0 \in K$. Its minimal polynomial is

$$Y^q - Y - (au + (u_0^q - u_0)).$$

(Notice the analogy to Kummer extensions: The multiplicative structures occurring for Kummer extensions correspond to additive structures now.) Let the pole divisor of u as an element of $K(X)/K$ be

$$\mathrm{div}_\infty(u) = \sum_{i=1}^{r} \mu_i \mathfrak{p}_i.$$

(It is possible that one of the $\mathfrak{p}_i$ equals ∞.) Then after applying substitutions as mentioned above, i.e. switching to a different generating element of $K(C)/K(X)$, one may assume that the pole divisor of u is

$$\mathrm{div}_\infty(u) = \sum_{i=1}^{r} \nu_i \mathfrak{p}_i$$

with $\nu_i = \mu_i$ if $q \nmid \mu_i$; $0 \le \nu_i < \mu_i$ and $q \nmid \nu_i$ otherwise.

To see this, write down the partial fraction decomposition

$$u = \sum_{i=1}^{r} u_i + r_0$$

with $r_0 \in K$,

$$\begin{aligned} u_i &= \sum_{j=1}^{\nu_i} \frac{r_{ij}}{p_i^j},\ r_{ij} \in K[X] \text{ of degree less than } \deg p_i, \\ &\quad \text{if } \mathfrak{p}_i \text{ is finite with local parameter } p_i; \\ u_i &= \sum_{j=1}^{\nu_i} r_{ij} X^j,\ r_{ij} \in K, \text{ if } \mathfrak{p}_i = \infty. \end{aligned}$$

Assume that $0 < \nu_i = q\nu_0$ for some i with finite $\mathfrak{p}_i$. The field K is perfect, and so is the residue class field $K(X)_{\mathfrak{p}_i}$, so that there is a polynomial u_0 of degree less than $\deg p_i$ with $r_{i,\nu_i} = u_0^q$ in $K(X)_{\mathfrak{p}_i}$, i.e., $p_i | r_{i,\nu_i} - u_0^q$. We may then replace u by $u - \left(\frac{u_0^q}{p_i^{\nu_i}} - \frac{u_0}{p_i^{\nu_0}}\right)$, which reduces ν_i by at least 1 and does not affect u_j for $j \ne i$. Similarly, if $0 < \nu_i = q\nu_0$ for $\mathfrak{p}_i = \infty$, then we may choose $u_0 \in K$ such that $r_{i,\nu_i} = u_0^q$ and replace u by $u - (u_0^q X^{\nu_i} - u_0 X^{\nu_0})$. The process is repeated until u has the desired properties.

Thus, let the pole divisor of u be

$$\sum_{i=1}^{r} \nu_i \mathfrak{p}_i \text{ with } \nu_i > 0 \text{ and } q \nmid \nu_i.$$

Any prime divisor $\mathfrak{p}$ not occurring in this sum is unramified in $K(C)/K(X)$; its inertia degree is 1 or q (i.e., it is completely splitting or has a unique inert extension) depending on whether $Y^q - Y - u \pmod{\mathfrak{p}}$ has a root in the residue class field $K(X)_{\mathfrak{p}}$ of $\mathfrak{p}$ or not. If $\mathfrak{p}$ occurs in the pole divisor of u, then it is totally ramified, i.e., its ramification index is $e = q$.

If the curve is absolutely irreducible (i.e. the modified u is not constant), its genus is

$$\frac{q-1}{2} \sum_{i=1}^{r} (\nu_i + 1) \deg \mathfrak{p}_i - (q-1).$$

Chapter 3

Hyperelliptic curves

The first type of algebraic groups suggested in the literature for use in cryptography were those formed by elliptic curves over finite fields ([Mil86, Kob87]). They are implementable in an efficient way and constitute the state of the art in public key cryptography. Their attraction basically results from the fact that the discrete logarithm problem on elliptic curves is resistant against subexponential attacks as described in Section 1.3. Noticing that an elliptic curve group is precisely the Jacobian of a hyperelliptic curve of genus 1, it is natural to investigate Jacobians of higher genus hyperelliptic curves as a possible replacement ([Kob89]). To allow an effective implementation, the abstract concepts of the previous chapter have to be filled with algorithmic life. In this spirit, we develop the arithmetical theory of hyperelliptic curves in this chapter.

There are different definitions of hyperelliptic curves in the literature. In algebraic geometry, it is common to define them by abstract properties of their function fields; on the other hand, the cryptographic community usually assumes a more operational point of view and defines them by concrete curve equations. Starting from the abstract definition, we investigate the resulting models of hyperelliptic curves in Section 3.1. To the best of my knowledge, this is the first attempt to provide an exhaustive classification of hyperelliptic curves and their imaginary or real quadratic models in even characteristic. If an imaginary quadratic model is chosen, then the Jacobian of a hyperelliptic curve can be represented by its ideal class group. In Section 3.2 we develop a unique representation for the ideal classes by pairs of univariate polynomials with certain properties. Finally in Section 3.3 we show how to realise the arithmetic of a hyperelliptic Jacobian in terms of these representations. Among others, we generalise algorithms which were previously described in odd or zero characteristic only to arbitrary curves. An average case analysis and a comparison of the different algorithms is the topic of Chapter 4.

3.1 Normal forms

3.1.1 Definitions

In algebraic geometry, a hyperelliptic function field is commonly defined as a quadratic extension of positive genus of a function field of genus zero ([Che51, Poo96]; there is an equivalent, but more complicated definition via differentials). The condition that the genus be positive excludes, among others, the trivial case of a quadratic constant field extension. Some authors go even further and require that the genus be at least 2. Since the theory to be developed in this chapter applies without changes to curves of genus 1, i.e. *elliptic curves*, this is merely a question of terminology. We shall include the case of genus 1 and treat elliptic as special hyperelliptic curves.

Stichtenoth gives basically the same definition, but requires that the ground field be rational ([Sti93], Definition VI.2.1). Since a function field of genus zero is rational if and only if it has a divisor of degree 1 ([Sti93], Proposition I.6.3), Chevalley's and Stichtenoth's definitions agree for fields with a degree 1 divisor, for instance, for K algebraically closed or finite (see [Sch31], § 8, or [Sti93], Corollary V.1.11). Being chiefly interested in curves over finite fields, we may use Stichtenoth's definition without losing generality.

Definition 3.1 *A* hyperelliptic function field *is a quadratic extension of positive genus of a rational function field.*

An extension $F/K(X)$ of degree 2 is automatically cyclic, so that we may apply the results on Kummer and Artin–Schreier extensions of Section 2.6. For a prime divisor $\mathfrak{p}$ of $K(X)$, let r denote the number of its extensions in F, f its inertia degree and e its ramification index. By the decomposition law of Section 2.2.5, we have $efr = 2$. Consequently, $\mathfrak{p}$ may be either (totally) ramified for $e = 2$, (completely) splitting for $r = 2$ or inert for $f = 2$.

If all prime divisors of degree 1 of $K(X)$ are inert, then the corresponding curve has no rational point, i.e. no point with coefficients in K. In the following we ignore this degenerate case, arguing that it results from viewing the function field over a "wrong" constant field: The curve has points defined over a quadratic extension K_1/K, so that switching to the function field FK_1/K_1 resolves the degeneracy.

Then for the sake of simplicity, we may assume that the infinite prime divisor ∞ of $K(X)$ is not inert. Otherwise we may choose a non-inert prime divisor $\mathfrak{p}$ of degree 1 with local parameter p. The pole divisor of $\frac{1}{p}$ has the form $\mathfrak{P}_1 + \mathfrak{P}_2$ (with $\mathfrak{P}_1 = \mathfrak{P}_2$ for $\mathfrak{p}$ ramified). Thus, $[F : K(\frac{1}{p})] = \deg(\mathrm{div}_\infty(\frac{1}{p})) = 2$ (see Section 2.3.1), and F is a hyperelliptic extension of $K(\frac{1}{p})$ in which the infinite prime divisor is not inert. On the geometric side, this different choice of the infinite prime divisor corresponds to a different affine model of the same projective curve (cf. the discussion of Section 2.4.2).

Definition 3.2 *A* hyperelliptic curve *is a non-singular affine curve $C \in K[X,Y]$ whose function field $K(C)$ is hyperelliptic over $K(X)$ and in which the infinite prime divisor of $K(X)$ is ramified or splitting. In the first case, the curve is called* imaginary quadratic*, in the second case,* real quadratic*.*

As explained in Section 2.4.2, the unit rank of $\mathcal{O} = K[X,Y]/(C)$ is zero for an imaginary quadratic and 1 for a real quadratic hyperelliptic curve C in analogy to the case of imaginary and real quadratic number fields, which explains the terminology.

To use hyperelliptic curves in cryptography, it is important to make the arithmetic of their Jacobians effective. So in the cryptographic community, it is common to leave the abstract point of view and to define hyperelliptic curves explicitly by polynomials of certain types.

In odd characteristic, they were first examined by Artin, and it is well-known that a hyperelliptic curve of genus g admits an affine model of the form

$$Y^2 - u$$

with $u \in K[X]$ monic of degree $2g+1$ resp. $2g+2$; in the first case, the curve is imaginary quadratic, in the second case, real quadratic ([Art24a, Art24b]; see also [Can87, SSW96]).

Hyperelliptic curves in even characteristic are less well understood. To allow a unified treatment of any characteristic, Koblitz examines (imaginary quadratic) curves of genus g of the form

$$Y^2 + vY - u$$

with $v \in K[X]$ of degree at most g and $u \in K[X]$ monic of degree $2g+1$ ([Kob89]). An excellent elementary introduction to curves of this type, which does not require any number theoretic knowledge, is given in [MWZ98]. Zuccherato develops a theory of real quadratic curves in even characteristic, but does not provide a complete characterisation of real and imaginary quadratic curves in terms of the curve equations ([Zuc97a], Chapter 4).

Applying the theory of Kummer and Artin–Schreier extensions, we take the abstract Definition 3.1 as a starting point to derive explicit equations for hyperelliptic curves in the following two sections. We show that depending on the characteristic of the ground field and the ramification behaviour of the infinite prime divisor, a hyperelliptic function field can be represented by hyperelliptic curves in certain normal forms. Given a hyperelliptic curve, we show how to transform it into one of the normal forms and thus provide an algorithmic characterisation of imaginary and real quadratic curves.

3.1.2 Characteristic different from 2

Let K have characteristic zero or an odd prime and F/K be a hyperelliptic function field. Then by the results of Section 2.6.1, F is the function field of a curve of the form $C = Y^2 - u$ with a square-free polynomial $u \in K[X]$. Let y be the image of Y in F.

Assume first that the degree of u is odd. If the leading coefficient $a \in K$ of u is not 1, we may rewrite u as a polynomial u' in $X' = \frac{X}{a}$ with leading coefficient $a^{\deg u+1}$ and switch to the generating element $y' = \frac{y}{a^{(\deg u+1)/2}}$ of F/K with minimal polynomial $Y^2 - \frac{u'}{a^{\deg u+1}}$ over $K(X')$. So u can be assumed to be monic. The infinite prime divisor is ramified, and the genus of F is

$$g = \sum_{p|u,\ p \text{ irreducible}} \frac{1}{2} \deg p + \frac{1}{2} \deg \infty - (2-1) = \frac{1}{2}(\deg u - 1),$$

so u is of degree $2g+1$.

Now let $\deg u$ be even. According to Section 2.6.1, the infinite prime divisor is not ramified. It is splitting if and only if there is an element $z \in K(X)$ of non-positive degree such that $\deg\left(z^2 - \frac{X^{\deg u}}{u}\right) < 0$. This is only the case if the leading coefficient of u is a square in K; then z may be chosen as a square root of this coefficient. Replacing y by $\frac{y}{z}$, the polynomial u is replaced by a monic polynomial. The genus of F is then

$$g = \sum_{p|u,\ p \text{ irreducible}} \frac{1}{2} \deg p - (2-1) = \frac{1}{2}(\deg u - 2),$$

so u is of degree $2g+2$.

Theorem 3.3 *In odd or zero characteristic, any curve of the form*

$$C = Y^2 - u$$

with $u \in K[X]$ monic and square-free of degree at least 3 is hyperelliptic. If $\deg u = 2g+1$ is odd, then C is imaginary quadratic of genus g; if $\deg u = 2g+2$ is even, then C is real quadratic of genus g. Conversely, any hyperelliptic function field over K allows an affine model of this type.

Proof: It remains to show that a curve of the given form has no affine singularity. Assume that $P = (x, y) \in \overline{K} \times \overline{K}$ is a singular point on C, i.e. a point for which C, $\frac{\partial C}{\partial X}$ and $\frac{\partial C}{\partial Y}$ vanish simultaneously. Then $\frac{\partial C}{\partial Y}(x, y) = 2y$ implies $y = 0$, so that $C(x, y) = -u(x)$ and x is a zero of u. On the other hand, $\frac{\partial C}{\partial X} = -\frac{\partial u}{\partial X}$ implies $\frac{\partial u}{\partial X}(x) = 0$, so that x is a multiple root of u, which contradicts that u is square-free. □

Theorem 3.4 *Let K be finite of odd characteristic. Given a non-singular affine curve $C \in K[X, Y]$ which is monic and quadratic in Y, it can be decided in deterministic polynomial time whether the curve is hyperelliptic, and if so, whether it is imaginary or real quadratic.*

Proof: Let

$$C = Y^2 + vY - u$$

with u, $v \in K[X]$ and let y be the image of Y in $K(C)$. Then switching to the generating element $y' = y + \frac{v}{2}$ of $K(C)/K(X)$ amounts to completing the square in C, and the minimal polynomial of y' over $K(X)$ is given by

$$C' = Y^2 - u' \text{ with } u' = u + \frac{v^2}{4}.$$

Let $u' = u''\overline{u}^2$ with u'' square-free. Then $y'' = \frac{y'}{\overline{u}}$ generates $K(C)/K(X)$, and its minimal polynomial is

$$C'' = Y^2 - u''$$

(cf. Section 2.6.1). If $\deg u'' \geq 3$ is odd, then C is hyperelliptic and imaginary quadratic. If $\deg u'' \geq 4$ is even and the leading coefficient of u'' is a square in K, then C is hyperelliptic and real quadratic. Otherwise, C is not hyperelliptic.

The finiteness of K is only needed to ensure that the polynomial arithmetic and the test for being a square can be carried out in polynomial time. □

3.1.3 Characteristic 2

Let K have even characteristic and F/K be a hyperelliptic function field. Then F is generated over $K(X)$ by an element y with minimal polynomial in the Artin–Schreier form $Y^2 + Y + w$ of Section 2.6.2 with $w \in K(X)$ such that all irreducible polynomials in the denominator of w occur to an odd power and w is either of odd positive or of non-positive degree.

To turn the minimal polynomial into a curve equation, we have to get rid of the denominator of w. Notice that if $w = \frac{r}{s}$ with r, $s \in K[X]$, we might consider the generating element ys with minimal polynomial $Y^2 + sY + sr$. However, this may introduce unnecessary singularities because any multiple root x of s leads to the singular point $(x, 0)$. To avoid these singularities, we split off the square part of the denominator of w and write $w = \frac{r}{s^2 t}$ with s, $t \in K[X]$ monic, $r \in K[X]$, $\gcd(r, st) = 1$ and t square-free. The generating element yst of $F/K(X)$ has the minimal polynomial

$$C = Y^2 + stY + rt = Y^2 + vY + u.$$

Since the prime divisors in the denominator of w occur to an odd power, any irreducible polynomial p dividing s also divides t, i.e., t is the square-free part of st. Hence, any irreducible polynomial dividing v divides u to the first power.

Concerning the ramification of the infinite prime divisor, consider first the case that w is of odd positive degree, i.e., $\deg r > 2\deg s + \deg t$ and $\deg r + \deg t$ is odd. Then ∞ is ramified, and the genus of F/K is

$$\begin{aligned} g &= \frac{1}{2}\left(\sum_{p|s^2t,\ p \text{ irreducible}} (v_p(s^2t)+1)\deg p + (\deg w + 1)\deg\infty\right) - 1 \\ &= \frac{1}{2}(\deg(s^2t) + \deg t + \deg r - \deg(s^2t) - 1) = \frac{1}{2}(\deg(rt-1)) \\ &= \frac{1}{2}(\deg u - 1). \end{aligned}$$

Thus, $\deg u = 2g+1$ and $\deg v = \frac{1}{2}\deg(s^2t^2) \le \lfloor \frac{1}{2}\deg(rt)\rfloor = \lfloor \frac{1}{2}\deg u\rfloor = g$.
Now, let $\deg w \le 0$. Then ∞ is unramified, and the genus of F/K is

$$g = \frac{1}{2}\sum_{p|s^2t,\ p \text{ irreducible}} (v_p(s^2t)+1)\deg p - 1 = \deg(st) - 1 = \deg v - 1.$$

Thus, $\deg v = g+1$ and $\deg u = \deg(rt) \le \deg(s^2t^2) = 2g+2$ since $\deg w \le 0$. More precisely, ∞ is splitting if and only if there is an element $z \in K(X)$ with $z = 0$ or $\deg z \le 0$ such that $z^2 + z + w = 0$ or $\deg(z^2+z+w) < 0$. If $\deg w < 0$, i.e., $\deg u < 2g+2$, then $z = 0$ is such an element; if $\deg w = 0$, i.e., $\deg u = 2g+2$, such an element exists if and only if the leading coefficient of r, which equals the leading coefficient of u, is $z^2 + z$ for some $z \in K$.

Theorem 3.5 *Let* $\operatorname{char} K = 2$. *Consider curves of the form*

$$C = Y^2 + vY + u$$

with $u, v \in K[X]$, *v monic and any irreducible polynomial dividing v is a simple divisor of u. If $g \ge 1$,* $\deg u = 2g+1$ *and* $\deg v \le \lfloor \frac{1}{2}\deg u\rfloor = g$, *then C is an imaginary quadratic hyperelliptic curve. If $g \ge 1$,* $\deg v = g+1$ *and* $\deg u < 2\deg v = 2g+2$, *or* $\deg u = 2g+2$ *and the leading coefficient of u equals z^2+z for some element $z \in K$, then C is a real quadratic hyperelliptic curve. Conversely, any hyperelliptic function field over K has an affine model of this type.*

Proof: Again the only point which remains to be verified is the non-singularity of C. Assume that $P = (x,y) \in \overline{K} \times \overline{K}$ is a singular point on C. Since $\frac{\partial C}{\partial Y} = v$, x is a root of v. Let p be the irreducible polynomial of x over K, so that p divides v and p is a simple divisor of u. From the curve equation we deduce that $0 = y^2 + v(x)y + u(x) = y^2$, so $y = 0$. The vanishing of $\frac{\partial C}{\partial X} = \frac{\partial v}{\partial X}Y + \frac{\partial u}{\partial X}$ in (x,y) implies that $\frac{\partial u}{\partial X}(x) = 0$, so x is a multiple root of u, a contradiction. □

If in the situation of Theorem 3.5 $\deg u = 2g + 1$, a is the leading coefficient of u and y is the image of Y in $K(C)$, then we may rewrite u and v as polynomials u' and v' in $X' = \frac{X}{a}$ as in Section 3.1.2. The leading coefficient of u' is then $a^{\deg u+1}$, and the minimal polynomial of $y' = \frac{y}{a^{(\deg u+1)/2}}$ over $K(X')$ is of the form

$$C'' = Y^2 + \frac{v'}{a^{(\deg u+1)/2}} - \frac{u'}{a^{\deg u+1}} = Y^2 + v'' - u''$$

with u'' monic. This shows that any imaginary quadratic curve can be transformed into the model chosen by Koblitz in [Kob89]. However, the condition that v is monic is lost during the transformation.

The rather strong property that any hyperelliptic function field in even characteristic has an affine model in which each irreducible factor of v is a simple divisor of u as well as Hasse's article [Has35] have passed unperceived by the cryptographic community. It is an interesting open question whether this property can be used to speed up the arithmetic of the Jacobian.

Theorem 3.6 *Let K be finite of even characteristic. Given a non-singular affine curve $C \in K[X, Y]$ which is monic and quadratic in Y, it can be decided in probabilistic polynomial time whether the curve is hyperelliptic, and if so, whether it is imaginary or real quadratic.*

Proof: Let $C = Y^2 + vY + u$ with u, $v \in K[X]$ and let y be the image of Y in $K(C)$. Switching to the generating element $y' = \frac{y}{v}$ of $K(C)/K(X)$ brings C into the Artin–Schreier form

$$C' = Y^2 + Y + u' \text{ with } u' = \frac{u}{v^2}.$$

Section 2.6.2 shows how to transform the curve equation into the form $C'' = Y^2 + Y + u''$ such that the irreducible polynomials in the denominator of u'' occur to an odd multiplicity, and the degree of u'' is either positive and odd or non-positive. This step is of probabilistic polynomial complexity; besides computing the partial fraction decomposition of u' it involves taking square roots in certain residue class fields which are extensions of K of degree bounded by $\deg v$. Notice that the root of an element $x \in \mathbb{F}_{2^m}$ can be computed in polynomial time as $x^{2^{m-1}}$.

The discussion of the beginning of this section shows how to obtain one of the normal forms of Theorem 3.5. To answer the question, one might then have to solve a general quadratic equation in K, which can be done in deterministic polynomial time by solving a system of linear equations over $\mathbb{F}_2$ (see also the discussion below). □

The previous result complements the discussion of [Zuc97a]. Zuccherato examines only hyperelliptic curves $C = Y^2 + vY + u$ with u *and* v monic and gives the necessary condition for such a curve to be real quadratic that $\deg v \geq 1$ ([Zuc97a] and [Zuc97b]

contain the typographical error $\deg v > 1$). He states that C is real quadratic if and only if it has a root $y \in K((\frac{1}{X}))$, the set of Laurent series in $\frac{1}{X}$ with coefficients in K, which is the completion of $K(X)$ with respect to the infinite valuation. This condition can also be checked algorithmically by solving a recurrent sequence of quadratic equations over K.

Example. In [Zuc97a], p. 32, Zuccherato claims that

$$C = Y^2 + (X+1)Y + (X^6 + X^2 + X + 1)$$

is a real quadratic hyperelliptic curve over $\mathbb{F}_2$. Letting $y = \sum_{i=-\infty}^{N} c_i X^i$ be a hypothetical root of C in $\mathbb{F}_2((\frac{1}{X}))$, one finds $N = 3$ and the quadratic equations

$$\begin{aligned}
c_3^2 + 1 &= 0 \\
c_2^2 + c_3 &= 0 \\
c_2 + c_3 &= 0 \\
c_1^2 + c_1 + c_2 + 1 &= 0 \\
c_0 + c_1 + 1 &= 0 \\
c_0^2 + c_{-1} + c_0 + 1 &= 0 \\
c_{-2i} + c_{-2i+1} &= 0 \text{ for } i \geq 1 \\
c_{-i}^2 + c_{-2i-1} + c_{-2i} &= 0 \text{ for } i \geq 1.
\end{aligned}$$

The first equations have two distinct solutions for $(c_3, c_2, c_1, c_0, c_{-1})$, namely $(1,1,0,1,1)$ and $(1,1,1,0,1)$; the following equations are in fact linear in the variables with smallest index they contain and can thus be solved recursively (in fact, over $\mathbb{F}_2$ any quadratic equation is linear since $c_i^2 = c_i$). □

In general when searching a root $y = \sum_{i=-\infty}^{N} c_i X^i$ of $C = Y^2 + vY + u$ in $K((\frac{1}{X}))$, one obtains $N = \max\left(\deg v, \left\lfloor \frac{1}{2} \deg u \right\rfloor\right)$ and first has to solve a system of quadratic equations in the variables $c_N, \ldots, c_{-\deg v}$. If this is not possible, then C is not real quadratic (it may be imaginary quadratic or not hyperelliptic in the sense of Definition 3.2 at all). Otherwise, for any $i < -\deg v$ there is a unique equation among the remaining ones in which all occurring variables c_j satisfy $j \geq i$ and c_i occurs linearly; thus, the set of infinitely many equations has a solution which can be found recursively.

Notice that the system of quadratic equations in $c_N, \ldots, c_{-\deg v}$ is of a special type since it does not involve mixed terms $c_i c_j$, so that it can be solved in polynomial time. Rewrite $c_i = \sum_{j=1}^{t} c_{ij} e_j$ for a fixed basis $\{e_1, \ldots, e_t\}$ of $K = \mathbb{F}_{2^t}$ over $\mathbb{F}_2$ with new variables c_{ij} taking values in $\mathbb{F}_2$. Then each quadratic equation of the type

$$c_r^2 + \sum_i a_i c_i + b = 0$$

over $\mathbb{F}_{2^t}$ with $a_i = \sum_{j=1}^{t} a_{ij} e_j \in \mathbb{F}_{2^t}$ and $b = \sum_{j=1}^{t} b_j e_j \in \mathbb{F}_{2^t}$ can be written as

$$\sum_{j=1}^{t} c_{rj}^2 e_j^2 + \sum_i \sum_{j,\,k=1}^{t} a_{ij} c_{ik} e_j e_k + \sum_{j=1}^{t} b_j e_j = 0.$$

Taking into account that $c_{rj}^2 = c_{rj}$ since $c_{rj} \in \mathbb{F}_2$, expressing $e_j e_k$ as a linear combination of the basis elements $e_1, \ldots, e_t$ and equating coefficients, we obtain a set of t linear equations over $\mathbb{F}_2$. Likewise, any linear equation over $\mathbb{F}_{2^t}$ translates into t linear equations over $\mathbb{F}_2$. Thus, we obtain a linear system over $\mathbb{F}_2$ with t times as many variables and equations as before, which can be solved in polynomial time.

The previous discussion shows that Zuccherato's algorithm for recognising real quadratic curves is of polynomial time. However, it does not distinguish between imaginary quadratic curves and curves for which the infinite prime divisor is inert. Even worse in the cryptographic context is that it does not allow to determine the genus of the curve, which by the results of Section 2.5 is an important security parameter for any cryptosystem based on its Jacobian.

Example. We apply the technique of the proof of Theorem 3.6 to the same curve $C = Y^2 + vY + u$ over $\mathbb{F}_2$ with $v = X + 1$ and $u = X^6 + X^2 + X + 1$ as above. Then

$$u' = \frac{u}{v^2} = \frac{X^5 + X^4 + X^3 + X^2 + 1}{X + 1} = X^4 + X^2 + \frac{1}{X + 1}.$$

By the discussion of Section 2.6.2 we may replace u' by

$$u'' = u' + (X^2 + X)^2 = \frac{1}{X + 1} = \frac{r}{s^2 t}$$

with $r = s = 1$ and $t = X + 1$. Then the normal form of C is given by

$$C'' = Y^2 + (X + 1)Y + (X + 1).$$

A trivial extension of Theorem 3.5 shows that C'' is of genus zero, so that $K(C)$ is not hyperelliptic, but in fact rational. Indeed, if y is the image of Y in $K(C'')$, then $K(C) = K(C'') = K(X, y) = K(y)$ since

$$X = \frac{y^2}{y + 1} - 1 \in K(y).$$

□

3.2 Ideal (class) groups

During this section, we assume that the hyperelliptic function field is represented by a hyperelliptic curve in the normal form $C = Y^2 + vY - u$ with u, $v \in K[X]$ satisfying the

properties derived in Sections 3.1.2 and 3.1.3, that y is the image of Y in $K(C)$ and that $\mathcal{O}$ is the integral closure of $K[X]$ in $K(C)$. Also recall the projection map of Section 2.4.2:

$$\pi_{\mathcal{O}} : \mathrm{Div}^0(K(C)/K) \to \mathcal{I}(\mathcal{O}), \quad \sum m_{\mathfrak{P}}\mathfrak{P} \mapsto \sum_{\mathfrak{P}\nmid\infty} m_{\mathfrak{P}}\mathfrak{P}.$$

3.2.1 Imaginary and real quadratic representations

As emphasised several times already, the arithmetic of a Jacobian is particularly simple if it can be realised by the ideal class group of an affine curve. This is indeed the case for imaginary quadratic hyperelliptic curves.

Proposition 3.7 *If C is an imaginary quadratic hyperelliptic curve, then its Jacobian $J(K(C)/K)$ is isomorphic to the ideal class group $\mathfrak{H}(O)$.*

Proof: See Section 2.4.3. □

It is sometimes possible to convert a real into an imaginary quadratic representation by a procedure similar to that described in Section 3.1.1, which was used to make the infinite prime divisor ramified or splitting. Assume that there is a finite ramified prime divisor $\mathfrak{p}$ of $K(X)$ of degree 1 with local parameter $p = X - x$, corresponding to a point (x, y) on C. Then the pole divisor of $\frac{1}{p}$ is $\mathrm{div}_\infty(\frac{1}{p}) = \mathrm{div}_0(p) = 2\mathfrak{p}$, so that $[K(C) : K(\frac{1}{p})] = 2$. So $K(C)$ is a hyperelliptic function field over $K(\frac{1}{p})$, and in this representation, the infinite prime divisor is ramified.

If the smallest degree of a ramified prime divisor is $d > 1$, then the local parameter of such a prime divisor splits completely over an extension of K of degree d and corresponds to d points on the curve defined over this extension, whence it is possible to obtain the ramification of the infinite prime divisor after a constant field extension of degree d. Let C be real quadratic of genus g. From the discussion of Sections 2.6.1 and 2.6.2 follows that the local parameters of ramified prime divisors are either irreducible factors of a polynomial u of degree $2g+2$ in odd or zero characteristic, or of a polynomial v of degree $g+1$ in even characteristic. Thus, $d|2g+2$ resp. $d|g+1$.

3.2.2 Decomposition of prime ideals

In Sections 3.1.2 and 3.1.3 we have already seen how to determine the splitting behaviour of a rational prime divisor, and the discussion of Section 2.4.1 shows that the prime ideals of $K[X]$ have the same splitting behaviour in $\mathcal{O}$. For later algorithms, we need concrete generators of the prime ideals of $\mathcal{O}$ above a given prime ideal of $K[X]$. The problem is solved by Kummer's theorem, which is applicable if certain integral bases are known.

Proposition 3.8 *Let $\mathfrak{p}$ be a finite rational prime divisor with local parameter $p \in K[X]$, $\mathcal{O}_\mathfrak{p}$ its valuation ring and $\overline{\mathcal{O}}_\mathfrak{p}$ the integral closure of $\mathcal{O}_\mathfrak{p}$ in $K(C)$. Then $\{1, y\}$ is a* local integral power basis *for $K(C)$ at $\mathfrak{p}$, i.e., $\overline{\mathcal{O}}_\mathfrak{p} = \mathcal{O}_\mathfrak{p} + y\mathcal{O}_\mathfrak{p}$. Furthermore, $\mathcal{O} = K[C] = K[X, Y]/(C)$.*

For the proof of the proposition, it is useful to consider the trace and the norm function of the Galois extension $K(C)/K(X)$.

Definition 3.9 *The* hyperelliptic involution *or* conjugation $\overline{\cdot}$ *is the unique non-trivial automorphism of $K(C)/K(X)$; it assigns to y the second root $\overline{y} = -y - v$ of C and satisfies $\overline{\overline{z}} = z$ for $z \in K(C)$. The* trace *and* norm *functions of $K(C)/K(X)$ are defined by*

$$\mathrm{Tr}_{K(C)/K(X)}(z) = z + \overline{z} \in K(X) \text{ and } \mathrm{N}_{K(C)/K(X)}(z) = z\overline{z} \in K(X)$$

for $z \in K(C)$.

Proof of Proposition 3.8: Clearly, $\{1, y\}$ is a basis of $K(C)/K(X)$ contained in $\overline{\mathcal{O}}_\mathfrak{p}$. Conversely, let $ay + b$ with a, $b \in K(X)$ be an element of $\overline{\mathcal{O}}_\mathfrak{p}$. We have to show that a, $b \in \mathcal{O}_\mathfrak{p}$. Since $ay + b$ is integral over $\mathcal{O}_\mathfrak{p}$, its minimal polynomial

$$Y^2 - \mathrm{Tr}_{K(C)/K(X)}(ay + b)Y + \mathrm{N}_{K(C)/K(X)}(ay + b)$$

has coefficients in $\mathcal{O}_\mathfrak{p}$. Hereby,

$$\mathrm{Tr}_{K(C)/K(X)}(ay + b) = -av + 2b$$

and

$$\mathrm{N}_{K(C)/K(X)}(ay + b) = -a^2u - abv + b^2.$$

Assume first that the characteristic of K is odd or zero. Then $v = 0$ and u is square-free, and $-av + 2b \in \mathcal{O}_\mathfrak{p}$ implies $b \in \mathcal{O}_\mathfrak{p}$. This in turn induces $a^2u \in \mathcal{O}_\mathfrak{p}$, i.e. $0 \le v_\mathfrak{p}(a^2u) = 2v_\mathfrak{p}(a) + v_\mathfrak{p}(u) \le 2v_\mathfrak{p}(a) + 1$. As $v_\mathfrak{p}$ takes only integral values, $v_\mathfrak{p}(a) \ge 0$ and $a \in \mathcal{O}_\mathfrak{p}$.

Let now $\mathrm{char}\, K = 2$, so that $av \in \mathcal{O}_\mathfrak{p}$. If $v_\mathfrak{p}(v) = 0$, then $a \in \mathcal{O}_\mathfrak{p}$ and $b^2 + avb \in \mathcal{O}_\mathfrak{p}$. If $v_\mathfrak{p}(b) < 0$, then $v_\mathfrak{p}(b^2) < v_\mathfrak{p}(b) \le v_\mathfrak{p}(avb)$, so that $v_\mathfrak{p}(b^2 + avb) = 2v_\mathfrak{p}(b) < 0$ by the strict triangle inequality, a contradiction. If $v_\mathfrak{p}(v) > 0$, then $v_\mathfrak{p}(u) = 1$ by the properties of the normal form of C (see Section 3.1.3). Then

$$\begin{aligned} v_\mathfrak{p}(a^2u) &= 2v_\mathfrak{p}(a) + 1; \\ v_\mathfrak{p}(b^2) &= 2v_\mathfrak{p}(b); \\ v_\mathfrak{p}(abv) &\ge v_\mathfrak{p}(a) + v_\mathfrak{p}(b) + 1. \end{aligned}$$

If $v_\mathfrak{p}(a) \ge v_\mathfrak{p}(b)$, then the unique minimum of these three values is $2v_\mathfrak{p}(b)$. Since $\mathrm{N}_{K(C)/K(X)}(ay + b) \in \mathcal{O}_\mathfrak{p}$, the strict triangle inequality implies $v_\mathfrak{p}(b) \ge 0$, and then

$v_\mathfrak{p}(a) \geq v_\mathfrak{p}(b) \geq 0$. A similar reasoning applies in the remaining case, so that indeed a, $b \in \mathcal{O}_\mathfrak{p}$.

Thus, $\overline{\mathcal{O}}_\mathfrak{p} = \mathcal{O}_\mathfrak{p} + y\mathcal{O}_\mathfrak{p}$. From $\overline{\mathcal{O}}_\mathfrak{p} = \bigcap_{\mathfrak{P}|\mathfrak{p}} \mathcal{O}_\mathfrak{p}$ and $\mathcal{O} = \bigcap_{\mathfrak{P}\nmid\infty} \mathcal{O}_\mathfrak{P}$ we deduce

$$\begin{aligned} \mathcal{O} &= \bigcap_{\mathfrak{p}\neq\infty} \overline{\mathcal{O}}_\mathfrak{p} = \left(\bigcap_{\mathfrak{p}\neq\infty} \mathcal{O}_\mathfrak{p}\right) + y\left(\bigcap_{\mathfrak{p}\neq\infty} \mathcal{O}_\mathfrak{p}\right) \\ &= K[X] + yK[X] = K[X,y] = K[X,Y]/(C). \end{aligned}$$

□

The knowledge of the local integral power basis $\{1, y\}$ at $\mathfrak{p}$ allows to apply Kummer's theorem ([Sti93], Theorem III.3.7) and to deduce the prime ideals above (p) from the decomposition of the minimal polynomial of y modulo p as follows.

Proposition 3.10 (Kummer's theorem for hyperelliptic curves) *Let $\mathfrak{p}$ be a discrete valuation of $K(X)$ with local parameter p, and consider the roots of*

$$Y^2 + vY - u \pmod{p}$$

in the residue class field $K(X)_\mathfrak{p} = K[X]/(p)$.

- *If there is no root, then $\mathfrak{p}$ is inert, and the only extension $\mathfrak{P}$ of $\mathfrak{p}$ satisfies $p\mathcal{O} = \mathfrak{P} \cap \mathcal{O}$.*
- *If there is a double root $b + (p)$, then $\mathfrak{p}$ is ramified, and the unique extension $\mathfrak{P}$ of $\mathfrak{p}$ satisfies $p\mathcal{O} = (\mathfrak{P} \cap \mathcal{O})^2$ with $\mathfrak{P} \cap \mathcal{O} = (p, y - b) = p\mathcal{O} + (y - b)\mathcal{O}$. This case occurs if and only if p divides the discriminant $v^2 + 4u$ of the curve.*
- *If there are two distinct roots $b + (p)$ and $b' + (p) = -b - v + (p)$, then $\mathfrak{p}$ is splitting, and the extensions $\mathfrak{P}_1$ and $\mathfrak{P}_2$ of $\mathfrak{p}$ satisfy $p\mathcal{O} = (\mathfrak{P}_1 \cap \mathcal{O})(\mathfrak{P}_2 \cap \mathcal{O})$ with $\mathfrak{P}_1 \cap \mathcal{O} = (p, y - b)$ and $\mathfrak{P}_2 \cap \mathcal{O} = (p, y - b')$.*

In particular, any prime ideal of $\mathcal{O}$ is generated by at most two elements.

The hyperelliptic involution can be extended in a natural way to ideals and divisors.

Definition and proposition 3.11 *Let $\mathfrak{p}$ be a finite prime divisor of $K(X)$ and $\mathfrak{P}$ an extension of $\mathfrak{p}$ in $K(C)$. If $\mathfrak{p}$ is inert or ramified, let its conjugate $\overline{\mathfrak{P}}$ be $\mathfrak{P}$ again. If $\mathfrak{p}$ is splitting, let $\overline{\mathfrak{P}}$ be the other extension of $\mathfrak{p}$. For a prime ideal $\mathfrak{P} \cap \mathcal{O}$ of $\mathcal{O}$ its conjugate is $\overline{\mathfrak{P}} \cap \mathcal{O}$. This conjugation is extended additively to $\mathcal{I}(\mathcal{O})$ and multiplicatively to the group of fractional ideals of $\mathcal{O}$.*

Then $\overline{\overline{\mathfrak{P}}} = \mathfrak{P}$, and $v_{\overline{\mathfrak{P}}}(z) = v_\mathfrak{P}(\overline{z})$ for any $z \in K(C)$.

Proof: That $\overline{\overline{\mathfrak{P}}} = \mathfrak{P}$ is obvious. By Kummer's theorem, any prime ideal is generated by two elements; moreover, if $\mathfrak{P} \cap \mathcal{O} = (r, s)$, then $\overline{\mathfrak{P}} \cap \mathcal{O} = (\overline{r}, \overline{s})$. This proves the assertion of the proposition for $z \in \mathcal{O}$, and the general result follows since $K(C)$ is the field of fractions of $\mathcal{O}$. (In fact, it also holds for the infinite prime divisors, but we only need it for finite ones.) □

3.2.3 Principal divisors

It will turn out in Section 3.2.4 that any ideal of $\mathcal{O}$ is generated by two elements $a \in K[X]$ and $c(y - b)$ with b, $c \in K[X]$. If an ideal is principal, then its divisor can be derived from the divisor of its generator via the equation

$$\operatorname{div}(z\mathcal{O}) = \pi_{\mathcal{O}}(\operatorname{div} z) \text{ for any } z \in K(C).$$

Furthermore, the divisor of a finally generated ideal can be deduced from the divisors of its generators.

Definition and proposition 3.12 *The* greatest common divisor *of two elements of* $\operatorname{Div}(K(C)/K)$ *is defined by*

$$\gcd\left(\sum m_{\mathfrak{P}}\mathfrak{P}, \sum n_{\mathfrak{P}}\mathfrak{P}\right) = \sum \min(m_{\mathfrak{P}}, n_{\mathfrak{P}})\mathfrak{P}.$$

For r, $s \in \mathcal{O}$,

$$\operatorname{div}(r\mathcal{O} + s\mathcal{O}) = \gcd(\pi_{\mathcal{O}}(\operatorname{div} r), \pi_{\mathcal{O}}(\operatorname{div} s)) = \pi_{\mathcal{O}}(\gcd(\operatorname{div} r, \operatorname{div} s)).$$

This definition and proposition can be extended to the case of several divisors resp. generators by induction.

Proof: The assertion follows immediately from the decomposability of ideals of $\mathcal{O}$ into a unique product of prime ideals and the observation that

$$\prod(\mathfrak{P} \cap \mathcal{O})^{m_{\mathfrak{P}}} = \{r \in \mathcal{O} : v_{\mathfrak{P}}(r) \geq m_{\mathfrak{P}} \text{ for all } \mathfrak{P}\},$$

see Section 2.2.2. □

So it is sufficient to determine the principal divisors of elements of the special form above in order to compute the divisor of an ideal and thus its decomposition into prime ideals.

Proposition 3.13 *Let* a, $b \in K[X]$. *For a prime divisor* $\mathfrak{p}$ *of* $K(X)$ *let* $\mathfrak{P}$ *denote an extension in* $K(C)$.

- *If* $a = \prod p^{\nu_p}$ *with* $p \in K[X]$ *irreducible,* $\nu_p \geq 0$ *and* $\mathfrak{p}$ *the prime divisor with local parameter* p*, then*

$$\operatorname{div}(a\mathcal{O}) = \sum_{\mathfrak{p} \text{ inert}} \nu_p \mathfrak{P} + \sum_{\mathfrak{p} \text{ ramified}} 2\nu_p \mathfrak{P} + \sum_{\mathfrak{p} \text{ splitting}} (\nu_p \mathfrak{P} + \nu_{\mathfrak{p}} \overline{\mathfrak{P}}).$$

- *If* $\mathrm{N}_{K(C)/K(X)}(y-b) = b^2 + bv - u = \prod p^{\nu_p}$ *with* $p \in K[X]$ *irreducible,* $\nu_p \geq 0$ *and* $\mathfrak{p}$ *the prime divisor with local parameter* p*, then* $\nu_p > 0$ *implies that* $\mathfrak{p}$ *is not inert and that* $b + (p)$ *is a root of* $Y^2 + vY - u \pmod p$*. If* $\mathfrak{p}$ *is ramified, then* $\nu_p \in \{0, 1\}$*. Let* $\mathfrak{P}$ *be such that* $\mathfrak{P} \cap \mathcal{O} = (p, y - b)$*. Then*

$$\operatorname{div}((y-b)\mathcal{O}) = \sum \nu_p \mathfrak{P}.$$

Proof: The first assertion is trivial. For the second assertion, note that $b^2 + bv - u = (y-b)(\overline{y} - b)$, so that

$$\begin{aligned} \operatorname{div}((b^2 + bv - u)\mathcal{O}) &= \operatorname{div}((y-b)\mathcal{O}) + \operatorname{div}((\overline{y-b})\mathcal{O}) \\ &= \operatorname{div}((y-b)\mathcal{O}) + \overline{\operatorname{div}((y-b)\mathcal{O})} \end{aligned}$$

by Proposition 3.11. Let $\mathfrak{p}$ be a finite prime divisor with local parameter p. The previous equation shows that if $\nu_p = 0$, then no extension of $\mathfrak{p}$ occurs in $\operatorname{div}((y-b)\mathcal{O})$. By Kummer's theorem 3.10 this is, for instance, the case if p is inert, since otherwise $b + (p)$ would be a root of $Y^2 + vY - u \pmod p$. If $\mathfrak{p}$ is splitting and $\nu_p \geq 1$, then Kummer's theorem shows that $\mathfrak{P}$ with $\mathfrak{P} \cap \mathcal{O} = (p, y - b)$ is an extension of $\mathfrak{p}$, and the second extension is $\overline{\mathfrak{P}}$. Thus, $v_{\mathfrak{P}}(y-b) \geq 1$. If $v_{\overline{\mathfrak{P}}}(y-b) = v_{\mathfrak{P}}(\overline{y-b}) = v_{\mathfrak{P}}(-y-b-v) \geq 1$, then the triangle inequality would imply $v_{\mathfrak{P}}(-v) = v_{\mathfrak{P}}((y-b) + (-y-b-v)) \geq 1$, so that $p|v$ and $\mathfrak{p}$ would be ramified, a contradiction. So $v_{\overline{\mathfrak{P}}}(y-b) = 0$. The equation above implies $v_{\mathfrak{P}}(y-b) + v_{\overline{\mathfrak{P}}}(y-b) = \nu_p$, so that $v_{\mathfrak{P}}(y-b) = \nu_p$. If $\mathfrak{p}$ is ramified and $\nu_p \geq 1$, then p is a divisor of v and a simple divisor of u (remember that C was supposed to be in one of the normal forms of Sections 3.1.2 and 3.1.3). Thus, $p|b$ and the strict triangle inequality yields $\nu_p = 1$. The extension of $\mathfrak{p}$ is $\mathfrak{P}$ with $\mathfrak{P} \cap \mathcal{O} = (p, y - b)$, and the equation above yields $v_{\mathfrak{P}}(y-b) + v_{\overline{\mathfrak{P}}}(y-b) = 2v_{\mathfrak{P}}(y-b) = 2\nu_p$. □

3.2.4 Semireduced divisors

As a first step towards finding unique representatives for the ideal classes, we examine how to canonically represent ideals.

Theorem 3.14 *Any integral ideal of* $\mathcal{O}$ *admits a unique representation as* $(d)(a, y - b)$ *with* a*,* b*,* $d \in K[X]$*,* d*,* a *monic,* $\deg b < \deg a$ *and*

$$a | \mathrm{N}_{K(C)/K(X)}(y-b) = b^2 + bv - u.$$

From Proposition 3.10, the theorem holds for the prime ideals of $\mathcal{O}$. Since any integral ideal is the (uniquely determined) product of finitely many prime ideals, the existence is shown by verifying that the product of two ideals of the given form is in the same form again. We give a constructive proof, which will provide the basis for the arithmetic in hyperelliptic Jacobians. For this proof the concept of the norm of an ideal plays an important role.

Proposition and definition 3.15 *Let $\mathfrak{a}$ be an integral ideal of $\mathcal{O}$ and $\overline{\mathfrak{a}}$ its conjugate. Then $\mathfrak{a}\overline{\mathfrak{a}}$ is a principal ideal which is generated by an element $a \in K[X]$. The ideal*

$$\mathrm{N}(\mathfrak{a}) = \mathfrak{a}\overline{\mathfrak{a}} \cap K[X] = aK[X]$$

is called the norm *of $\mathfrak{a}$.*

Proof: If $\mathfrak{a} = \mathfrak{P} \cap \mathcal{O}$ is a prime ideal, where $\mathfrak{P}$ is a prime divisor of $K(C)$ extending a prime divisor $\mathfrak{p}$ of $K(X)$ with local parameter p, then the assertion follows from Proposition 3.10: If $\mathfrak{P}$ is inert, then $\mathrm{N}(\mathfrak{a}) = p^2K[X]$, otherwise $\mathrm{N}(\mathfrak{a}) = pK[X]$. Since any integral ideal is a product of prime ideals, this already proves the more general result. □

Lemma 3.16 *If $\mathfrak{a} = (a, y - b)$ with $a,\ b \in K[X]$ and $a|b^2 + bv - u$, then $\mathrm{N}(\mathfrak{a}) = a$; if $\mathfrak{b} = (d)$ with $d \in K[X]$, then $\mathrm{N}(\mathfrak{b}) = d^2$.*

Proof: Let $c = \frac{b^2+bv-u}{a}$. We claim that $\gcd(a, 2b + v, c) = 1$. Otherwise, let $p \in K[X]$ be a common irreducible divisor. If the characteristic of K is different from 2, then $v = 0$ and $p|b$, so that $p^2|b^2 - ac = u$, contradicting the assumption that C is in the normal form of Section 3.1.2. If the characteristic of K is 2, then $p|v$ implies that it is a simple divisor of u since C is in the normal form of Section 3.1.3. Thus, $p|ac + bv + u = b^2$, so $p|b$ and $p^2|ac + b^2 + bv = u$, a contradiction. Now,

$$\begin{aligned}
\mathfrak{a}\overline{\mathfrak{a}} &= (a, y - b)(a, \overline{y - b}) \\
&= (a^2, a(y - b), a(\overline{y} - b), ac) \\
&= (a)(a, (y - b) + (\overline{y} - b), y - b, c) \\
&= (a)(a, -2b - v, c, y - b) \\
&= (a)(\gcd(a, -2b - v, c), y - b) \\
&= (a)
\end{aligned}$$

The assertion on $\mathfrak{b}$ follows trivially from Kummer's theorem 3.10. □

Proof of Theorem 3.14: Let $\mathfrak{a}_1 = (a_1, y - b_1)$ and $\mathfrak{a}_2 = (a_2, y - b_2)$ with $a_i,\ b_i \in K[X]$ and $a_i|b_i^2 + b_iv - u$. With the remarks above, it is sufficient to show that $\mathfrak{a} = \mathfrak{a}_1\mathfrak{a}_2 = (d)(a, y - b)$

with $a|b^2+bv-u$ to prove the existence of the desired representation for any integral ideal; clearly, it is then possible to multiply a and d by suitable constants in $K^\times \subseteq \mathcal{O}^\times$ to obtain that they are monic, and to add a suitable multiple of a to b to obtain $\deg b < \deg a$.

Let $c_i = \frac{b_i^2+b_iv-u}{a_i}$. Then

$$\begin{aligned}\mathfrak{a} &= (a_1, y-b_1)(a_2, y-b_2)\\ &= (a_1a_2, a_1y-a_1b_2, a_2y-a_2b_1, (b_1+b_2+v)y-(b_1b_2+u))\end{aligned} \tag{3.1}$$

since

$$(y-b_1)(y-b_2) = y^2-(b_1+b_2)y+b_1b_2 = -vy+u-(b_1+b_2)y+b_1b_2.$$

Write

$$d = \gcd(a_1, a_2, b_1+b_2+v) = u_1a_1+u_2a_2+u_3(b_1+b_2+v)$$

with u_1, u_2, $u_3 \in K[X]$; such a representation can be effectively computed by the extended Euclidean algorithm, see Algorithm 3.24. Let

$$\begin{aligned}b &= \frac{u_1a_1b_2+u_2a_2b_1+u_3(b_1b_2+u)}{d}\\ &= \frac{u_1a_1b_2+(d-u_1a_1-u_3(b_1+b_2+v))b_1+u_3(b_1b_2+u)}{d}\\ &= b_1+\frac{a_1}{d}(u_1(b_2-b_1)-u_3c_1)\\ &= b_2+\frac{a_2}{d}(u_2(b_1-b_2)-u_3c_2)\\ &\in K[X].\end{aligned}$$

Then

$$d(y-b) = u_1a_1(y-b_1)+u_2a_2(y-b_2)+u_3((b_1+b_2+v)y-(b_1b_2+u)) \in \mathfrak{a}.$$

Subtracting suitable multiples of $d(y-b)$ from the generators of $\mathfrak{a}$ in (3.1) which involve y yields

$$\begin{aligned}\mathfrak{a} &= (a_1a_2, a_1(b-b_2), a_2(b-b_1), (b_1+b_2+v)b-(b_1b_2+u), d(y-b))\\ &= \Big(\frac{a_1a_2}{d}d, \frac{a_1a_2}{d}(u_2(b_1-b_2)-u_3c_2), \frac{a_1a_2}{d}(u_1(b_2-b_1)-u_3c_1),\\ &\quad \frac{a_1a_2}{d}(u_1c_2+u_2c_1), d(y-b)\Big)\end{aligned}$$

from the formulae for b above and

$$\begin{aligned}&(b_1+b_2+v)b\\ &\quad= \frac{(u_1a_1b_1+u_2a_2b_2)(b_1+b_2+v)+(d-u_1a_1-u_2a_2)(b_1b_2+u)}{d}\\ &\quad= \frac{a_1a_2}{d}(u_1c_2+u_2c_2)+(b_1b_2+u).\end{aligned}$$

With the help of a symbolic algebra programme and using the relations $b_i^2 = a_i c_i - b_i v + u$, it is not difficult to compute that

$$\begin{aligned} \mathrm{N}_{K(C)/K(X)}(y-b) &= b^2 + bv - u \\ &= \frac{a_1 a_2}{d}\big((u_1u_2(b_1+b_2) + u_3(u_1c_2+u_2c_1))v \\ &\quad + u_3^2 c_1c_2 + (u_1a_1 + 2u_3b_1)u_1c_2 + (u_2a_2 + 2u_3b_2)u_2c_1 \\ &\quad + 2u_1u_2(b_1b_2 - u)\big) \\ &\in (d^{-1})\mathfrak{a} \end{aligned}$$

is divisible by $\frac{a_1a_2}{d^2}$.

Letting

$$t = \gcd\Big(d, u_2(b_1-b_2) - u_3c_2, u_1(b_2-b_1) - u_3c_1, u_1c_2 + u_2c_1, \frac{(b^2+bv-u)d^2}{a_1a_2}\Big),$$

we obtain

$$\mathfrak{a} = (d)\left(\frac{a_1a_2}{d^2}t, y-b\right)$$

with $\frac{a_1a_2}{d^2} | b^2 + bv - u$. By definition, the ideal norm is multiplicative; thus, Lemma 3.16 implies

$$a_1a_2tK[X] = d^2\frac{a_1a_2}{d^2} = \mathrm{N}(\mathfrak{a}) = \mathrm{N}(\mathfrak{a}_1)\,\mathrm{N}(\mathfrak{a}_2) = a_1a_2K[X],$$

and $t = 1$. Letting $a = \frac{a_1a_2}{d}$ provides the desired representation of $\mathfrak{a}$.

Concerning the uniqueness, observe that $\mathcal{O} = K[X] + yK[X]$ by Proposition 3.8. So

$$\begin{aligned} \mathfrak{a} &= (d)(a, y-b) = d(a\mathcal{O} + (y-b)\mathcal{O}) \\ &= daK[X] + dayK[X] + d(y-b)K[X] + d(-(b+v)y + u)K[X]. \end{aligned}$$

Thus if $ry + s \in \mathfrak{a}$ with $r,\ s \in K[X]$, then $d|r$. Since $dy - db \in \mathfrak{a}$, we deduce that $d = \gcd\{r : ry+s \in \mathfrak{a}\}$ is uniquely defined by $\mathfrak{a}$. Also, a is unique since $\mathrm{N}(\mathfrak{a}) = d^2aK[X]$.

The uniqueness of b is a simple consequence of the following result, which describes the decomposition of $\mathrm{div}(a\mathcal{O} + (y-b)\mathcal{O})$ into prime divisors. □

Corollary 3.17 *Let $a,\ b \in K[X]$ such that $a | N_{K(C)/K(X)}(y-b) = b^2 + bv - u$. If $a = \prod p^{\nu_p}$ with $p \in K[X]$ irreducible, $\nu_p \geq 0$ and $\mathfrak{p}$ the prime divisor with local parameter p, then $\nu_p > 0$ implies that $\mathfrak{p}$ is not inert and that $b+(p)$ is a root of $Y^2+vY-u \pmod p$. If $\mathfrak{p}$ is ramified, then $\nu_p \in \{0,1\}$. Let $\mathfrak{P}$ be the prime divisor of $K(C)$ with $\mathfrak{P} \cap \mathcal{O} = (p, y-b)$. Then $\mathrm{div}((a, y-b)\mathcal{O}) = \sum \nu_p\mathfrak{P}$. Furthermore, $\deg(\mathrm{div}((a, y-b)\mathcal{O})) = \deg a$.*

Proof: The assertion follows immediately from Propositions 3.13 and 3.12. □

Proof of the uniqueness in Theorem 3.14: The uniqueness of a and d has already been shown. The previous corollary shows that b is uniquely defined modulo the product of all prime divisors of a. Now the condition $a|b^2 + bv - u$ determines b modulo a. □

Recall from Section 2.4.1 that a divisor in $\mathrm{Div}(K(C)/K)$ is the divisor of an integral ideal if and only if it is positive and does not contain any infinite prime divisor. If $(d)(a, y-b)$ is an integral ideal of $\mathcal{O}$, then one may split off the principal part (d) and obtain a "simpler" ideal whose divisor is in the same ideal class. The following proposition characterises the divisors of ideals of the form $(a, y - b)$.

Proposition and definition 3.18 *Let $\mathfrak{a}$ be an integral ideal of $\mathcal{O}$ and $D = \sum \nu_{\mathfrak{P}}\mathfrak{P}$ its divisor. The following assertions are equivalent:*

1) *$\mathfrak{a}$ is not divisible by an integral ideal of $K[X]$, i.e., $(p^{-1})\mathfrak{a}$ is not integral for any (irreducible) polynomial $p \in K[X]$.*

2) *$D - \pi_{\mathcal{O}}(\mathrm{div}(p)) = D - \mathrm{div}_0(p) \not\geq 0$ for any (irreducible) polynomial $p \in K[X]$.*

3) *If $\mathfrak{P}$ is inert, then $\nu_{\mathfrak{P}} = 0$; if $\mathfrak{P}$ is ramified, then $\nu_{\mathfrak{P}} \in \{0; 1\}$; if $\mathfrak{P}$ is splitting, then $\nu_{\mathfrak{P}} = 0$ or $\nu_{\overline{\mathfrak{P}}} = 0$.*

4) *$\mathfrak{a} = (a, y - b)$ for a, $b \in K[X]$, a monic, $\deg b < \deg a$ and $a|b^2 + bv - u$.*

If these assertions are satisfied, then $D = \mathrm{div}((a, y - b)\mathcal{O})$ is called semireduced *and denoted by $\mathrm{div}(a, b)$ for the sake of simplicity.*

Proof: The equivalence of 1) and 2) is clear by the definition of the divisor of an ideal; the equivalence of 2) and 3) follows from the decomposition law 3.10. By Theorem 3.14, 1) implies 4), and by Proposition 3.13, 4) implies 3). □

Proposition 3.19 *Any ideal class contains a (not necessarily unique) semireduced divisor.*

Proof: We first show that any ideal class contains the divisor of an integral ideal. If $\mathfrak{a}$ is a fractional ideal, the prime divisor $\mathfrak{P}$ occurs with negative multiplicity in $\mathrm{div}(\mathfrak{a})$ and $\mathfrak{P}$ extends the prime divisor of $K(X)$ with local parameter p, then it suffices to multiply $\mathfrak{a}$ by a suitable power of the principal ideal (p) to ensure that $\mathfrak{P}$ occurs with non-negative multiplicity.

Such an integral ideal can be represented as $(d)(a, y - b)$ by Theorem 3.14, and $\mathrm{div}(a, b)$ is a semireduced divisor in the same ideal class as $\mathrm{div}(\mathfrak{a})$. □

Proposition 3.20 *If $D = \operatorname{div}(a, b)$ is a semireduced divisor, then $\overline{D} = \operatorname{div}(a, -b - v \bmod a)$ is semireduced and lies in the opposite ideal class. Precisely, $D + \overline{D}$ is the divisor of an ideal (d) with $d \in K[X]$.*

Proof: That $\overline{D}$ is semireduced is clear from the third of the equivalent properties of Proposition 3.18. The divisor D corresponds to the ideal $(a, y - b)$, so $\overline{D}$ corresponds to

$$\begin{aligned}\overline{(a, y-b)} &= (\overline{a}, \overline{y-b}) = (a, -y - v - b) \\ &= (a, y - (-b - v)) = (a, y - (-b - v \bmod a)),\end{aligned}$$

whence $\overline{D} = \operatorname{div}(a, -b - v \bmod a)$. For a divisor $\mathfrak{p}$ of $K(X)$, let p denote its local parameter. Write $D = \sum_{\mathfrak{p} \text{ not inert}} \nu_{\mathfrak{p}} \mathfrak{P}_{\mathfrak{p}}$, where $\mathfrak{P}_{\mathfrak{p}}$ is a suitable extension of $\mathfrak{p}$. Then

$$D + \overline{D} = \sum_{\mathfrak{p} \text{ not inert}} \nu_{\mathfrak{p}}(\mathfrak{P}_{\mathfrak{p}} + \overline{\mathfrak{P}}_{\mathfrak{p}}) = \sum_{\mathfrak{p} \text{ not inert}} \nu_{\mathfrak{p}} \operatorname{div}(p\mathcal{O}) = \operatorname{div}\left(\prod_{\mathfrak{p} \text{ not inert}} p^{\nu_p} \mathcal{O} \right).$$

□

3.2.5 Reduced divisors

While any ideal class contains a semireduced divisor, there may be several of them in a given class. An additional size restriction resolves this ambiguity at least in the imaginary quadratic case.

Definition and proposition 3.21 *Let g be the genus of $K(C)/K$. A semireduced divisor is called* reduced *if its degree is at most g. Any ideal class contains a reduced representative.*

Proof: Let ∞_1 be an infinite prime divisor of $K(C)$, which by Definition 3.2 is ramified or splitting. For $\operatorname{div}(\mathfrak{a}) \in \mathcal{I}(\mathcal{O})$, let $D = \operatorname{div}(\mathfrak{a}) - \deg(\operatorname{div}(\mathfrak{a}))\infty_1 \in \operatorname{Div}^0(K(C)/K)$, such that $\pi_{\mathcal{O}}(D) = \operatorname{div}(\mathfrak{a})$. From the discussion of Section 2.3.3 we know that there is a positive divisor D' of degree g such that $D' - g\infty_1$ and D are in the same divisor class. It follows from the exact sequence in Section 2.4.2 that $\pi_{\mathcal{O}}(D')$ and $\pi_{\mathcal{O}}(D)$ are in the same ideal class; furthermore, $\deg(\pi_{\mathcal{O}}(D')) \le \deg D' = g$. (It is possible that $\deg(\pi_{\mathcal{O}}(D')) < g$, precisely if D' contains an infinite prime divisor.) If $\pi_{\mathcal{O}}(D')$ is not semireduced, then by Theorem 3.14 it is possible to remove the zero divisor of a polynomial in $K[X]$ to turn it into a semireduced divisor, while further reducing its degree. □

For imaginary quadratic curves, the reduced representative is in fact unique. Riemann's theorem of Section 2.3.2 provides a lower bound on the dimension of the vector spaces $\mathcal{L}(D)$, which was used in the previous proof to derive the existence of a reduced divisor in each class. To show the uniqueness, we consequently need an upper bound on the dimensions.

Proposition 3.22 *If g is the genus of a function field F/K and D a divisor with $0 \leq \deg D \leq 2g$, then*

$$l(D) \leq \left\lfloor \frac{\deg D}{2} \right\rfloor + 1.$$

Proof: The case $\deg D \leq 2g-2$ is known as Clifford's theorem, see [Ful69], Section 8.6, or [Sti93], Theorem I.6.11. For $\deg D = 2g-1$ or $\deg D = 2g-2$, the Riemann–Roch Theorem yields equality, since a canonical divisor W satisfies $\deg W = 2g-2$, so $\deg(W-D) < 0$ and $l(W-D) = 0$. □

Theorem 3.23 *If C is imaginary quadratic, then any ideal class contains a unique reduced divisor.*

Proof: Let D_1 and D_2 be reduced divisors in the same ideal class, i.e., $D_1 - D_2$ is the divisor of some principal ideal. By Proposition 3.20, there is a polynomial $d \in K[X]$ such that $D_1 - D_2 = D_1 + \overline{D_2} - \operatorname{div}(d\mathcal{O})$, and $D_1 + \overline{D_2}$ is the divisor of a principal fractional ideal $b\mathcal{O}$ for some $b \in K(C)$. Since ∞_1 is the only extension of ∞, this implies $\operatorname{div} b = D_1 + \overline{D_2} - (\deg D_1 + \deg \overline{D_2})\infty_1 \geq -2g\infty_1$, where g is the genus of the curve. Thus, $b \in \mathcal{L}(2g\infty_1)$. By Proposition 3.22, $l(2g\infty_1) = g+1$; from $\operatorname{div}_\infty(X) = 2\infty_1$ we deduce that the linearly independent elements $1, X, \ldots, X^g$ are contained in $\mathcal{L}(2g\infty_1)$, whence they form a basis and $b \in K[X]$. Thus,

$$D_1 = D_2 - \operatorname{div}(d\mathcal{O}) + \operatorname{div}(b\mathcal{O}) = D_2 - \operatorname{div}(d'\mathcal{O}) + \operatorname{div}(b'\mathcal{O})$$

for $b' = \frac{b}{\gcd(b,d)}$ and $d' = \frac{d}{\gcd(b,d)}$. Since b' and d' are coprime, this shows that $D_2 - \operatorname{div}(d\mathcal{O})$ is positive, and the fact that D_2 is semireduced implies $\operatorname{div}(d'\mathcal{O}) = 0$. Similarly, $\operatorname{div}(b'\mathcal{O}) = 0$, and $D_1 = D_2$. □

Remark. For the proof it is crucial that the infinite prime divisor is totally ramified, so that a principal ideal determines a unique principal divisor of degree zero. Furthermore, it is crucial that the extension degree $n = [K(C) : K(X)]$ equals 2; otherwise, $\deg_\infty(X) = n\infty_1$, and the powers of X contained in $\mathcal{L}(2g\infty_1)$ are $1, X, \ldots, X^{\lfloor 2g/n \rfloor}$, which are less than $g+1$.

Remark. The divisors $\operatorname{div}(a,b)$ and $\operatorname{div}(\lambda a, b)$ are the same for any $\lambda \in K^\times$. We henceforth drop the assumption that a is monic; when adding reduced divisors, this saves an inversion and up to g multiplications in K and makes the arithmetic more efficient. As a drawback, this normalisation step has to be carried out now to check two reduced divisors for equality. Since a comparison occurs less frequently in cryptography than an addition, it is nevertheless worthwhile to work with these non-unique representatives. Also, b is only defined up to adding multiples of a. However, keeping the degree of b as low as possible makes the arithmetic more efficient.

3.3 Arithmetic

In this section we describe algorithms to realise the group law in the Jacobian of an imaginary quadratic hyperelliptic curve, assuming the notation of Section 3.2. Furthermore, let g denote the genus of $K(C)/K$. The Jacobian is isomorphic to the ideal class group, and we have seen in Section 3.2.5 that each ideal class is represented by a unique reduced divisor. This situation is completely analogous to the case of the class group of an imaginary quadratic number field, which was studied comprehensively by Gauß in his "Disquisitiones arithmeticae" [Gau01], cast in terms of "binary quadratic forms". As in these number fields, the algorithms for adding two ideal classes proceed in two steps. First, the reduced representatives D_1 and D_2 are *composed* to yield a semireduced divisor in the ideal class of $D_1 + D_2$. This step basically consists of multiplying the corresponding ideals and splitting off the obvious principal factor, as already seen in the proof of Theorem 3.14. Second, this semireduced divisor is *reduced*, which step is less obvious. Observe that the proof of Theorem 3.23, which shows that any ideal class is uniquely represented by a reduced divisor, is based on the Riemann–Roch Theorem and is not constructive. So the reduction algorithms can be interpreted as effective versions of the Riemann–Roch Theorem for hyperelliptic curves. We describe the analogue of the classic reduction algorithm due to Gauß and generalise two further algorithms, which go back to Cantor and Lagrange and which have previously been described for odd or zero characteristic only.

The complete algorithm, consisting of the composition and a reduction step, has been described in [Can87] and is therefore known as "Cantor's algorithm".

3.3.1 The extended Euclidean algorithm

Several of the algorithms needed for realising the arithmetic of hyperelliptic Jacobians rely on the knowledge of greatest common divisors of polynomials, which can be computed using the Euclidean algorithm. To fix the notation for the remainder of this chapter, we recall the algorithm on two polynomials.

Algorithm 3.24 (Extended Euclidean algorithm) *Let* r_0, $r_1 \in K[X]$ *with* $\deg r_1 \leq \deg r_0$.

1) Set $u_0 = 1$, $u_1 = 0$, $v_0 = 0$ *and* $v_1 = 1$.

2) Compute

$$r_{k-1} = q_k r_k + r_{k+1} \text{ with } \deg r_{k+1} < \deg r_k$$

for $k = 1, \ldots, t$ *and*

$$u_{k+1} = u_{k-1} - q_k u_k \text{ and } v_{k+1} = v_{k-1} - q_k v_k$$

for $k = 1, \ldots, t-1$, *where* t *is the smallest index such that* $r_{t+1} = 0$.

3) Set $\overline{u} = u_t$ and $\overline{v} = v_t$. Then

$$\gcd(r_0, r_1) = r_t = \overline{u} r_0 + \overline{v} r_1.$$

The correctness of the algorithm follows easily from the observations that $\gcd(r_{k-1}, r_k) = \gcd(r_k, r_{k+1})$ for $k = 1, \ldots, t$ and that $r_k = u_k r_0 + v_k r_1$ during the loop in Step 2). We note the following relations on the degrees of the involved polynomials:

$$\begin{aligned} \deg q_k &= \deg r_{k-1} - \deg r_k \quad \text{for } k \geq 1, && (3.2)\\ \deg u_k &= \deg r_1 - \deg r_{k-1} \quad \text{for } k \geq 2, && (3.3)\\ \deg v_k &= \deg r_0 - \deg r_{k-1} \quad \text{for } k \geq 2. && (3.4) \end{aligned}$$

Thus, $\deg \overline{u} < \deg r_1$ and $\deg \overline{v} < \deg r_0$.

Some text books describe a variant of the algorithm, in which $v_0, \ldots, v_{t-1}$ are not computed and $\overline{v}$ is derived in the end from the other polynomials as $\overline{v} = v_t = \frac{r_t - r_0 u_t}{r_1}$. However, this variant requires more operations in K.

If the greatest common divisor of more than two polynomials $f_1, \ldots, f_n$ is sought, then the Euclidean algorithm can be applied recursively. Suppose that

$$f = \gcd(f_1, \ldots, f_{n-1}) = u_1 f_1 + \cdots + u_{n-1} f_{n-1}$$

is already known. Another application of the Euclidean algorithm yields

$$\gcd(f, f_n) = \overline{u} f + \overline{v} f_n,$$

so that

$$\gcd(f_1, \ldots, f_n) = \gcd(f, f_n) = (\overline{u} u_1) f_1 + \cdots + (\overline{u} u_{n-1}) f_{n-1} + \overline{v} f_n.$$

Sometimes, only one of the multiplier $\overline{u}$ resp. $\overline{v}$ is needed, so that the converse sequence of the v_k resp. the u_k need not be computed. We refer to the corresponding algorithm as the *half-extended* Euclidean algorithm.

3.3.2 Composition

Composition of general binary quadratic forms has been described first by Gauß in [Gau01]; the special case of forms of the same discriminant, which in our setting correspond to reduced divisors of the same curve, is treated in Article 242.

Let $D_1 = \mathrm{div}(a_1, b_1)$ and $D_2 = \mathrm{div}(a_2, b_2)$ be reduced. We have already seen in the proof of Theorem 3.14 how to multiply the ideals corresponding to D_1 and D_2; the result is an ideal $(d)(a, y - b)$ corresponding to $D_1 + D_2$. Then clearly $\mathrm{div}(a, b)$ is a divisor in the

same ideal class as $D_1 + D_2$. (This shows that the algorithm remains valid even if D_1 and D_2 are not reduced, but only semireduced, which is, however, of no practical interest.) During the proof of Theorem 3.14, several formulae for b were given; in practice, one should use a formula which requires as few polynomial multiplications as possible. To collect the results on hyperelliptic arithmetic in one place, we summarise the algorithm again.

Algorithm 3.25 (Composition) *Let* $\operatorname{div}(a_1, b_1)$ *and* $\operatorname{div}(a_2, b_2)$ *be two (semi-)reduced divisors. The following steps determine a semireduced divisor* $\operatorname{div}(a, b)$ *in the same ideal class as* $\operatorname{div}(a_1, b_1) + \operatorname{div}(a_2, b_2)$.

Compute the greatest common divisor

$$d = \gcd(a_1, a_2, b_1 + b_2 + v) = u_1 a_1 + u_2 a_2 + u_3(b_1 + b_2 + v)$$

together with a representation by the extended Euclidean Algorithm 3.24. Let

$$a = \frac{a_1 a_2}{d^2}$$

and

$$b = b_1 + \frac{u_1 a_1 (b_2 - b_1) - u_3(b_1^2 + b_1 v - u)}{d} \bmod a.$$

3.3.3 Reduction

The divisor computed by the composition algorithm will usually not be reduced; in the worst case, it has degree $2g$, namely for $\deg a_1 = \deg a_2 = g$ and $\gcd(a_1, a_2, b_1 + b_2 + v) = 1$. In fact, this is also the typical case, see Section 4.5. In the reduction phase, the obtained divisor is replaced by a divisor in the same ideal class of degree at most g.

The most classic reduction algorithm is also due to Gauß for binary quadratic forms ([Gau01], Article 171). The case of hyperelliptic curves over finite fields of odd characteristic was treated by Artin ([Art24a], § 10).

Algorithm 3.26 (Gauß reduction) *Let* $D = \operatorname{div}(a_0, b_0)$ *be a semireduced divisor. Repeat the following steps for* $k = 1, \ldots, t$, *where* $t \geq 0$ *is the smallest index such that* $\deg a_t \leq g$*:*

$$\begin{aligned} a_k &= \frac{b_{k-1}^2 + b_{k-1} v - u}{a_{k-1}} \\ b_k &= -b_{k-1} - v \bmod a_k. \end{aligned}$$

Then the process comes to an end since $\deg a_k \leq \deg a_{k-1} - 2$ *for* $\deg a_{k-1} \geq g + 2$ *and* $\deg a_k \leq g$ *for* $\deg a_{k-1} = g + 1$. *The divisor* $\operatorname{div}(a_t, b_t)$ *is the reduced representative of* $\operatorname{div}(a_0, b_0)$.

Proof: For $k \geq 1$, let $a = a_{k-1}$, $b = b_{k-1}$ and $c = \frac{b^2+bv-u}{a}$, so that

$$ac = \mathrm{N}_{K(C)/K(X)}(y-b) = (y-b)(\overline{y}-b).$$

Thus, $\mathrm{div}(ac, b)$ and $\mathrm{div}(c, b)$ are semireduced, and Corollary 3.17 implies that

$$\begin{aligned} \mathrm{div}((y-b)\mathcal{O}) &= \gcd(\mathrm{div}((b^2+bv-u)\mathcal{O}), \mathrm{div}((y-b)\mathcal{O})) \\ &= \gcd(\mathrm{div}(ac\mathcal{O}), \mathrm{div}((y-b)\mathcal{O})) \\ &= \mathrm{div}(ac, b) \\ &= \mathrm{div}(a, b) + \mathrm{div}(c, b). \end{aligned}$$

Since $(y-b)\mathcal{O}$ is a principal ideal, $\mathrm{div}(c, b)$ is a divisor in the opposite ideal class of $\mathrm{div}(a, b)$, and Proposition 3.20 implies that $\overline{\mathrm{div}(c, b)} = \mathrm{div}(c, -b-v) = \mathrm{div}(a_k, b_k)$ is in the same ideal class as $\mathrm{div}(a, b)$.

The assertion on the degree follows from the observation that

$$\begin{aligned} \deg(b^2+bv-u) &\leq \max\{2\deg b, \deg b + g, 2g+1\} \\ &\leq \max\{2\deg a - 2, 2g+1\}. \end{aligned}$$

□

Gauß reduction amounts to replacing $\mathrm{div}(a, b)$ by the equivalent divisor

$$\overline{\mathrm{div}((y-b)\mathcal{O}) - \mathrm{div}(a, b)},$$

which usually has a degree smaller by 2 (see Section 4.5). Cantor observed that it is possible to use an appropriate divisor $\mathrm{div}((dy-c)\mathcal{O})$ in the place of $\mathrm{div}((y-b)\mathcal{O})$, so that usually only one reduction step is needed ([Can87]). We describe his concept generalised to arbitrary characteristic.

Intuitively, c and d should be chosen such that $\frac{c}{d} \equiv b \pmod{a}$, so that $\mathrm{div}((dy-c)\mathcal{O}) \equiv \mathrm{div}(d\mathcal{O}) + \mathrm{div}((y-b)\mathcal{O}) \pmod{\mathrm{div}(a\mathcal{O})}$. Since $\mathrm{div}(a, b) = \gcd(\mathrm{div}(a\mathcal{O}), \mathrm{div}((y-b)\mathcal{O}))$, taking the greatest common divisor with $\mathrm{div}(a\mathcal{O})$ will eliminate $\mathrm{div}(d\mathcal{O})$, unless a and d are not coprime.

Algorithm 3.27 (Cantor reduction) *Let* $\mathrm{div}(a, b)$ *be a semireduced divisor. Suppose that there are polynomials* c, d *and* λ *such that* $c = \lambda a + db$, $\deg c \leq \frac{\deg a + g}{2}$, $\deg d \leq \frac{\deg a - g - 1}{2}$ *and* $\gcd(\lambda, d) = 1$. *Let* $s = \gcd(c, d) = \gcd(a, d)$, $a' = \frac{a}{s}$, $c' = \frac{c}{s}$ *and* $d' = \frac{d}{s}$. *Let furthermore*

$$\overline{a} = \frac{{c'}^2 + c'd'v - {d'}^2 u}{a'},$$

$\overline{d}$ *such that* $d'\overline{d} \equiv 1 \pmod{\overline{a}}$ *and*

$$\overline{b} = -\overline{d}c' - v \bmod \overline{a}.$$

Then $\mathrm{div}\left(\overline{a}, \overline{b}\right) + \mathrm{div}(s, b \bmod s)$, *which can be computed using the composition algorithm of Section 3.3.2, is the reduced representative in the ideal class of* $\mathrm{div}(a, b)$.

Proof: Notice first that $\gcd(\lambda, d) = 1$ implies $gcd(d', \overline{a}) = 1$, so that $\overline{d}$ with the desired property exists. We closely follow the argumentation in [Can87].

The equation $a = sa'$ implies $\operatorname{div}(a, b) = \operatorname{div}(s, b \bmod s) + \operatorname{div}(a', b)$, and

$$\begin{aligned}
\operatorname{div}(a', b) &= \gcd(\operatorname{div}(a'\mathcal{O}), \operatorname{div}((y-b)\mathcal{O})) \\
&= \gcd(\operatorname{div}(a'\mathcal{O}), \operatorname{div}((d'y - d'b)\mathcal{O})) \\
&\quad \text{since } a' \text{ and } d' \text{ are coprime} \\
&= \gcd(\operatorname{div}(a'\mathcal{O}), \operatorname{div}((d'y - c')\mathcal{O})) \\
&\quad \text{since } c' \equiv d'b \pmod{a'} \\
&= D.
\end{aligned}$$

By Proposition 3.20, D is in the same ideal class as $\overline{\operatorname{div}((d'y - c')\mathcal{O}) - D}$, and mimicking the argumentation in the correctness proof of Gauß reduction it is not difficult to show that the latter divisor equals

$$\begin{aligned}
&\gcd(\operatorname{div}(\overline{a}\mathcal{O}), \operatorname{div}((d'(-y - v) - c')\mathcal{O})) \\
&\quad = \gcd\left(\operatorname{div}(\overline{a}\mathcal{O}), \operatorname{div}\left(\left(\overline{d}d'(-y - v) - \overline{d}c'\right)\mathcal{O}\right)\right) \text{ since } \overline{a} \text{ and } \overline{d} \text{ are coprime} \\
&\quad = \gcd\left(\operatorname{div}(\overline{a}\mathcal{O}), \operatorname{div}\left(\left(-y - v - \overline{d}c'\right)\mathcal{O}\right)\right) \text{ because } \overline{d}d' \equiv 1 \pmod{\overline{a}} \\
&\quad = \operatorname{div}\left(\overline{a}, -\overline{d}c' - v\right).
\end{aligned}$$

This shows that $\operatorname{div}(\overline{a}, \overline{b}) + \operatorname{div}(s, b \bmod s)$ lies in the same ideal class as $\operatorname{div}(a, b)$. To verify that it is reduced, notice that by Algorithm 3.25 its degree is

$$\begin{aligned}
&\deg \overline{a} + \deg s \\
&\quad \leq \max\{2 \deg c', \deg c' + \deg d' + g, 2 \deg d' + 2g + 1\} - \deg a' + \deg s \\
&\quad \leq \max\{\deg a + g, \deg a + g - \frac{1}{2}, \deg a + g\} - \deg a \\
&\quad = g.
\end{aligned}$$

□

The computation of $\overline{d} = (d')^{-1} \pmod{\overline{a}}$ requires to determine the greatest common divisor of d' and $\overline{a}$ by the extended Euclidean Algorithm 3.24. In practice, a different arrangement allows to compute s and $\overline{d}$ simultaneously by only one application of the Euclidean algorithm. In the sequel, when speaking of "Cantor reduction", we shall mean this more efficient version.

Algorithm 3.28 (Improved Cantor reduction) *In the situation of Algorithm 3.28, let*

$$\tilde{a} = \frac{c^2 + cdv - u}{a}.$$

Determine $s = \gcd(\tilde{a}, d) = r\tilde{a} + \tilde{d}d$. *If* $\deg s \geq 1$, *let* $\overline{a} = \frac{\tilde{a}}{s}$, $c' = \frac{c}{s}$ *and* $\overline{d} = \tilde{d}$; *otherwise, let* $\overline{a} = \tilde{a}$, $c' = c$ *and* $\overline{d} = s^{-1}\tilde{d}$. *Let*

$$\overline{b} = -\overline{d}c' - v \bmod \tilde{a},$$

so that $\operatorname{div}(a, b)$ *is equivalent to the reduced divisor*

$$\operatorname{div}\left(\overline{a}, \overline{b}\right) + \operatorname{div}(s, b \bmod s).$$

Unfortunately, c and d with the imposed degree bounds need not exist. Observe that c is a multiple of the greatest common divisor of a and b, and if $\deg(\gcd(a, b)) > \frac{\deg a + g}{2}$, then no c can keep the intended bound. In this case, it would seem optimal to choose $c = \gcd(a, b)$ and to compute more than one reduction step. But even then the reduction process may fail; in fact, it may even increase the degree of the divisor, as illustrated by the following, fairly general example.

Example. Suppose that $u = \prod_{i=1}^{g-1}(X - x_i)u_1$ with distinct $x_i \in K$ for $i = 1, \ldots, g-1$, so that $P_i = (x_i, 0)$ lies on C for $i = 1, \ldots g-1$. Let $a_1 = \prod_{i=1}^{g-1}(X - x_i)$ and $Q = (x_Q, y_Q)$ and $R = (x_R, y_R)$ be two further points on C with coordinates in K, $x_1, \ldots, x_{g-1}$, x_Q and x_R all distinct from each other, $y_Q \neq 0$, $y_R \neq 0$ and $\frac{y_Q}{a_1(x_Q)} \neq \frac{y_R}{a_1(x_R)}$. Let $a = a_1(X - x_Q)(X - x_R)$ and $b = a_1 l$, where l is the unique linear polynomial such that $l(x_Q) = \frac{y_Q}{a_1(x_Q)}$ and $l(x_R) = \frac{y_R}{a_1(x_R)}$. Then

$$\sum_{i=1}^{g-1} P_i + Q + R = \operatorname{div}(a, b)$$

is semireduced of degree $g + 1$. The choice for c with lowest possible degree is

$$c = \gcd(a, b) = a_1 = \lambda a + db$$

with uniquely determined $\lambda \in K^\times$ and $d \in K[X]$ linear. If $s = \gcd(a_1, d) = 1$, then

$$\overline{a} = \frac{a_1 + dv - d^2 u_1}{(X - x_Q)(X - x_R)},$$

where $\deg(d^2 u_1) = g + 4$, $\deg(a_1 + dv) \leq g + 1$ and hence $\deg(\operatorname{div}(\overline{a}, \overline{b})) = \deg \overline{a} = g + 2 > \deg a$. Otherwise, the degree of $\operatorname{div}(\overline{a}, \overline{b})$ is smaller by $\deg s = 1$, but this is compensated by the addition of $\operatorname{div}(s, b \bmod s)$. □

In case of failure, which is extremely unlikely as shown by the analysis of Section 4.5, one must choose another reduction algorithm. To check whether polynomials c and d with the desired degrees do exist and to compute them in this case, one can use the extended Euclidean algorithm 3.24 applied to a and b. With the notation of Section 3.3.1, if there

is no index $t' \leq t$ such that $\deg r_{t'} \leq \frac{\deg a+g}{2}$, then no suitable polynomials c and d exist. Otherwise, let t' be the smallest such index. Then $c = r_{t'}$, $\lambda = u_{t'}$ and $d = v_{t'}$ are suitable choices: Equation (3.4) implies that

$$\deg d = \deg a - \deg r_{t'-1} \leq \deg a - \left(\frac{\deg a + g}{2} + \frac{1}{2}\right) = \frac{\deg a - g - 1}{2}.$$

Furthermore, during the execution of the Euclidean algorithm, u_k and v_k are always coprime, so that the additional constraint $\gcd(\lambda, d) = 1$ is also met. (Even if this were not the case, dividing c, λ and d by $\gcd(\lambda, d)$ would establish this condition while lowering the degrees of c and d even further.)

The most costly steps in the Gauß reduction procedure are the computations of the a_k, each involving one multiplication and one division of rather high degree polynomials. In the original reduction algorithm each step is independent of the previous one. However, as soon as one reduction step has been carried out, the formula for a_k can be rewritten using information from the previous step. The following algorithm has been suggested for hyperelliptic curves over a field of odd characteristic by Paulus and Stein in [PS98], who attribute the explicit formulae in the setting of a quadratic number field to Tenner. The main idea can be traced back to Lagrange ([Gra73], Théoreme II and Corollaire 1). Again we provide a generalised version for arbitrary characteristic.

Algorithm 3.29 (Lagrange reduction) *Let $D = \operatorname{div}(a_0, b_0)$ be a semireduced divisor. Repeat the following steps for $k = 1, \ldots, t$, where $t \geq 0$ is the smallest index such that $\deg a_t \leq g$:*

$$\begin{aligned} a_1 &= \frac{b_0^2 + b_0 v - u}{a_0} \\ -b_0 - v &= q_1 a_1 + b_1 \quad \textit{with } \deg b_1 < \deg a_1 \end{aligned}$$

For $k \geq 2$:

$$\begin{aligned} a_k &= a_{k-2} + q_{k-1}(b_{k-2} - b_{k-1}) \\ -b_{k-1} - v &= q_k a_k + b_k \quad \textit{with } \deg b_k < \deg a_k. \end{aligned}$$

Then the process comes to an end, and the divisor $\operatorname{div}(a_t, b_t)$ is the reduced representative of $\operatorname{div}(a_0, b_0)$.

Proof: We show that the sequence of divisors generated by the algorithm is exactly the same as in the Gauß reduction process. To this purpose it is sufficient to show that

$$a_k = \frac{b_{k-1}^2 + b_{k-1} v - u}{a_{k-1}},$$

which implies that also b_k is the same polynomial as in the usual Gauß reduction. We proceed by induction on k, the assertion being obviously true for $k = 1$. Hence let $k \geq 2$.

$$\begin{aligned} a_k a_{k-1} &= a_{k-2}a_{k-1} + q_{k-1}a_{k-1}(b_{k-2} - b_{k-1}) \\ &= b_{k-2}^2 + b_{k-2}v - u - (b_{k-2} + b_{k-1} + v)(b_{k-2} - b_{k-1}) \\ &\quad \text{by the induction hypothesis and the construction of } q_{k-1} \\ &= b_{k-1}^2 + b_{k-1}v - u, \end{aligned}$$

which proves the assertion. □

Notice that except for the first step the computation of a_k needs only one polynomial multiplication and no polynomial division, and that q_k is a by-product of the computation of b_k.

Chapter 4

Efficiency of hyperelliptic cryptosystems

Hyperelliptic curves over finite fields are characterised by two parameters, their genus g and the size q of their constant field. When choosing a hyperelliptic curve for use in cryptography, a natural criterion — besides adequate security — is the efficiency of the underlying arithmetic. While the running time of algorithms is usually measured by providing upper bounds for the worst case, this approach does not allow to compare different algorithms as long as the bounds are not tight. Therefore, in this chapter we derive explicitly the average number of operations needed for carrying out the arithmetic in hyperelliptic Jacobians of varying genus and size of the constant field.

Since the extended Euclidean algorithm on polynomials is a building block of several algorithms of Section 3.3, we provide an average case analysis of the Euclidean algorithm on the way.

4.1 Cryptographic setting

Our aim is to compare hyperelliptic cryptosystems in which the genus g and the size q of the constant field vary, but which offer the same level of security. Hereby, we assume that the Jacobian is realised by the ideal class group of an imaginary quadratic hyperelliptic curve as in Section 3.3. We have seen in Section 1.2 that the main parameter determining the security of a public key cryptosystem based on discrete logarithms is the cardinality of the underlying group; for Jacobians of curves over finite fields, Weil's theorem of Section 2.5 implies that their cardinality is about q^g. Thus, we fix a desired level of security $L = e^l$ and consider curves for which $q^g \approx L$ or equivalently $q \log q \approx l$.

(According to the discussion in Section 1.2, $L \approx 10^{50}$ or $l \approx 115$ is generally considered adequate.) A given security level can be reached for smaller values of g at the expense of larger values of q, and vice versa. For larger genus, the algorithms require to handle polynomials of larger degree, which in turn results in more operations in the constant field; on the other hand, any single field operation in a larger constant field becomes more demanding. So it is a priori unclear which is the optimal choice of parameters.

During the analysis we neglect the additional constraint that the class number has a large prime factor so that the cryptosystem is not vulnerable by the Pohlig–Hellman attack; its fulfillment depends on the concrete curve and not only on g and q, so that a generic analysis seems impossible. An important consequence of the subexponential algorithms in Chapters 6 and 7 is that g cannot be too large in relation to q to preserve the security of the system. We may thus assume that g is sufficiently small so that it is not worthwhile to realise the polynomial arithmetic by fast Fourier transform techniques.

4.2 Probability distribution

To obtain average case results, we need the probability distributions of the objects under consideration, in our case of the polynomials a and b defining elements $\operatorname{div}(a, b)$ of the Jacobians. As the analysis will reveal, a uniform distribution over the polynomials of some fixed degree is most desirable, since then further intermediate results of the Euclidean algorithm are also uniformly distributed. For a fixed curve, most of the pairs of polynomials (a, b) do not define a divisor, so clearly a uniform distribution is only achievable if we also let the curve vary. Even then we do not obtain an exact uniform distribution, but we provide a heuristic argument that the deviation is sufficiently small.

Heuristic 4.1 *Let the hyperelliptic curve C of fixed genus g be chosen randomly according to a uniform distribution on the defining pairs of suitable polynomials (v, u). If $\operatorname{div}(a, b)$ is a uniformly selected element of $J(H)$, then we may assume that a varies uniformly over all polynomials of degree g and b over all polynomials of degree $g - 1$. Likewise, if $\operatorname{div}(a_1, b_1)$ and $\operatorname{div}(a_2, b_2)$ are two uniformly selected elements of $J(H)$, then we may assume that a_1 and a_2 vary uniformly and independently over all polynomials of degree g and b_1 and b_2 over all polynomials of degree $g - 1$.*

Justification: We examine only the case of two divisors because all arguments hold and slightly simplify if only one divisor is considered. We suppose that hyperelliptic curves are given by the normal forms of Theorems 3.3 and 3.5 and make two further simplifying assumptions, which account for the heuristic not being a theorem: First, in odd characteristic we assume that any two polynomials $v = 0$ and u monic of degree $2g+1$ define a hyperelliptic curve, not taking into account that a negligible proportion of these curves is singular. Hence there are q^{2g+1} curves altogether. In even characteristic,

we assume that any two polynomials v monic of degree at most g and u of degree $2g+1$ such that the square-free part t of v divides u define a hyperelliptic curve, not taking into account that some of these curves are singular and that others violate the condition of Theorem 3.5 that any irreducible factor of v is a *simple* factor of u. So for any given v, there are $(q-1)q^{2g+1-\deg t}$ possible values of u. Second, we assume that all Jacobians have cardinality q^g (cf. Section 2.5), so that $(q-1)q^g$ pairs of polynomials (a,b) define a Jacobian element because we do not require a to be monic, cf. the remark in Section 3.2.5.

To avoid unnecessary repetitions, we present only the even characteristic case, which is clearly more intricate.

Let (a_1, b_1, a_2, b_2) be a random quadruple of polynomials such that $\deg b_1 < \deg a_1 \leq g$ and $\deg b_2 < \deg a_2 \leq g$. Let for the moment v be fixed with square-free part t. Since two random polynomials are coprime with probability $1 - 1/q$ (see Lemma 4.6) and we take g to be small and thus q to be large, we may assume that

$$\gcd(a_1, a_2) = \gcd(a_1, t) = \gcd(a_2, t) = 1,$$

i.e., that $\operatorname{lcm}(a_1, a_2, t) = a_1 a_2 t$. Then there are $(q-1)q^{2g+1-\deg t-\deg a_1-\deg a_2}$ curves admitting $\operatorname{div}(a_1, b_1)$ and $\operatorname{div}(a_2, b_2)$ simultaneously as elements of their Jacobians. Namely these curves correspond to the solutions u of degree $2g+1$ of the equations $u \equiv b_1^2 + b_1 v \pmod{a_1}$, $u \equiv b_2^2 + b_2 v \pmod{a_2}$ and $u \equiv 0 \pmod{t}$, which by the Chinese Remainder Theorem translate into a single equation modulo $a_1 a_2 t$. Hence the probability of selecting a curve with given v containing both of the divisors on their Jacobians is $q^{-\deg a_1 - \deg a_2}$, which is in fact independent of v. Once such a curve is fixed, the conditional probability of choosing (a_1, b_1, a_2, b_2) is given by $\left(\frac{1}{(q-1)q^g}\right)^2$. Thus, the quadruple occurs with a total probability of $q^{-2g-\deg a_1-\deg a_2}(q-1)^{-2}$.

Then all the quadruples of polynomials with $\deg a_1 = \deg a_2 = g$ have the same probability of $q^{-4g}(q-1)^{-2}$. Hence, the event $\deg a_1 = \deg a_2 = g$ and $\deg b_1 = \deg b_2 = g-1$ occurs with probability

$$\frac{(q-1)^4 q^{4g-2}}{(q-1)^2 q^{4g}} = \left(1 - \frac{1}{q}\right)^2,$$

which is very close to 1. □

4.3 Average complexity of the Euclidean algorithm

The composition step of Section 3.3.2 and the Cantor reduction of Section 3.3.3 rely on the extended Euclidean algorithm 3.24, or to be more precise on its half-extended version, since one of the multipliers is not needed. Consequently we have to analyse the average complexity of the Euclidean algorithm for polynomials over finite fields. We

hereby concentrate on the case of two polynomials, since Lemma 4.6 shows that these are usually coprime and hence no further computations involving a third polynomial are needed.

While there is an abundant literature on the Euclidean algorithm on integers (see [Knu81], Section 4.5.3 and the references therein), I am aware of only a few articles dealing with its average complexity when applied to polynomials: Arnold and John Knopfmacher analyse the average number of polynomial divisions in [KK88], while Ma and von zur Gathen present the average number of field operations for different variants of the algorithm over $\mathbb{F}_2[X]$ in [MG90]. In [Ma87], Chapter 3, Ma gives a more comprehensive account of the Euclidean algorithm over any finite field, including the average complexity. However, neither of them treats the computation of the multipliers. Ma uses a rather intricate counting argument, which appears to be difficult to generalise to the extended algorithm. Instead we prefer to explicitly compute the probability distributions of the random variables involved, hereby presenting alternative proofs for some of Ma's results.

Throughout this section we assume that we are given two polynomials r_0 and r_1 over a finite field $\mathbb{F}_q$ of fixed degrees $d_0 \geq d_1 \geq 1$, respectively, and that these input data are uniformly distributed over all polynomials of degrees d_0 and d_1. A run of the extended Euclidean algorithm 3.24 creates sequences $(r_0, \ldots, r_t)$, $(u_0, \ldots, u_t)$ and $(v_0, \ldots, v_t)$ of elements of $\mathbb{F}_q[X]$, where t and the r_k, u_k and v_k are random variables. Denote the random variable $\deg r_k$ by d_k with the convention $d_k = -\infty$ for $k > t$. By a slight abuse of notation, we let d_t denote the degree of the last non-zero remainder and d_{t-1} the degree of the next to last one.

We are interested in the average number of arithmetic operations in $\mathbb{F}_q$ needed for carrying out the algorithm; since additions and subtractions are essentially for free, we restrict our attention to the number of multiplications and inversions in $\mathbb{F}_q$. The following theorem states the main result of this section:

Theorem 4.2 *Let r_0 and r_1 be uniformly distributed over the polynomials over $\mathbb{F}_q$ of degrees $d_0 \geq d_1 \geq 1$, respectively. Denote by $E_m(d_0, d_1)$ and $E_i(d_0, d_1)$ the expected number of multiplications and inversions in $\mathbb{F}_q$ performed by the non-extended Euclidean algorithm, and by $E^{\overline{u}}(d_0, d_1)$ and $E^{\overline{v}}(d_0, d_1)$ the expected number of additional multiplications needed to compute the multipliers $\overline{u}$ and $\overline{v}$, respectively. Then all these quantities are polynomials in $1/q$; more precisely,*

$$\begin{aligned} & E_m(d_0, d_1)(q-1)^2 q^{d_1} \\ &= \; (d_0 d_1 + d_0 + d_1 - 1)\, q^{d_1+2} - \left(2d_0 d_1 + \tfrac{1}{2} d_1^2 + 2d_0 + \tfrac{7}{2} d_1 - 3\right) q^{d_1+1} \\ &\quad + \left(d_0 d_1 + d_1^2 + d_0 + 3d_1 - 2\right) q^{d_1} - \left(\tfrac{1}{2} d_1^2 + \tfrac{1}{2} d_1 - 1\right) q^{d_1-1} \\ &\quad + q^2 - 2q \end{aligned}$$

$$E_i(d_0, d_1) = d_1 - \frac{d_1 - 1}{q}$$

$$
\begin{aligned}
&E^{\overline{u}}(d_0,d_1)(q-1)^2q^{d_1}\\
&= \left(d_1^2-d_1-2\right)q^{d_1+2}-\left(\tfrac{5}{2}d_1^2+\tfrac{5}{2}d_1-10\right)q^{d_1+1}+\left(2d_1^2+5d_1-2\right)q^{d_1}\\
&\quad -\left(\tfrac{1}{2}d_1^2+\tfrac{3}{2}d_1\right)q^{d_1-1}+\left(\tfrac{1}{2}d_1^2+\tfrac{3}{2}d_1\right)q^3-\left(2d_1^2+5d_1-2\right)q^2\\
&\quad +\left(\tfrac{5}{2}d_1^2+\tfrac{5}{2}d_1-10\right)q-\left(d_1^2-d_1-2\right)\\
&E^{\overline{v}}(d_0,d_1)(q-1)^2q^{d_1}\\
&= (2d_0-d_1)(d_1-1)q^{d_1+2}-\left(5d_0d_1-\tfrac{5}{2}d_1^2-2d_0+\tfrac{9}{2}d_1-7\right)q^{d_1+1}\\
&\quad +\left(4d_0d_1-2d_1^2+5d_1-1\right)q^{d_1}-\left(d_0d_1-\tfrac{1}{2}d_1^2+\tfrac{3}{2}d_1\right)q^{d_1-1}\\
&\quad +\left(d_0d_1-\tfrac{3}{2}d_1^2+2d_0-\tfrac{7}{2}d_1\right)q^2-(2d_0d_1-3d_1^2+2d_0-2d_1+7)q\\
&\quad +\left(d_0d_1-\tfrac{3}{2}d_1^2+\tfrac{3}{2}d_1+1\right)
\end{aligned}
$$

To prove the theorem, we certainly need to know the arithmetic complexity of the basic polynomial operations. Let f_1 and f_2 be polynomials of degree $n_1 \geq n_2$, respectively. Assuming conventional polynomial arithmetic as explained in Section 4.1, the number of field multiplications needed to compute $f_1 f_2$ depends on the number of non-zero coefficients of f_1 and f_2. However, observing that the number of non-zero non-leading coefficients of a random polynomial of degree n over $\mathbb{F}_q$ is Binomial$(n, 1-1/q)$–distributed, we suppose that all polynomials involved have no zero coefficients, which biases the results only negligibly since q is large. Then multiplying f_1 and f_2 takes $(n_1+1)(n_2+1)$ field multiplications, and a polynomial division with remainder of f_1 by f_2 takes one inversion for the leading coefficient of f_2 and $(n_2+1)(n_1-n_2+1)$ multiplications. (This differs from Ma's assumption of needing $n_2(n_1-n_2+1)$ multiplications and n_1-n_2+1 divisions ([Ma87], Fact 3.3). Taking into account that divisions are more costly in a finite field than multiplications and observing that all divisions are actually by the leading coefficient of f_2, it is more efficient to invert this coefficient once and for all and to replace the divisions by multiplications.)

Noticing that the final division by r_t can be omitted if $d_t = 0$, we see that the non-extended Euclidean algorithm requires

$$\sum_{k=1,\ldots,t\,:\,d_k\neq 0}(d_k+1)(d_{k-1}-d_k+1)$$

multiplications and $|\{k=1,\ldots,t\,:\,d_k\neq 0\}|$ inversions. The computation of the u_k needs additional

$$\sum_{k=4}^{t}(d_{k-2}-d_{k-1}+1)(d_1-d_{k-2}+1)$$

and the computation of the v_k

$$\sum_{k=3}^{t}(d_{k-2}-d_{k-1}+1)(d_0-d_{k-2}+1)$$

multiplications. Notice that $u_2 = 1$, $u_3 = -q_2$ and $v_2 = -q_1$ are given for free, and that the degrees of the q_k, u_k and v_k can be expressed in terms of the d_k by (3.2) to (3.4).

Then the expected numbers of field operations are given by

$$\begin{aligned} E_m(d_0,d_1) &= (d_1+1)(d_0-d_1+1) + \sum_{\mu=1}^{\infty} P(d_2=\mu)(\mu+1)(d_1-\mu+1) \\ &\quad + \sum_{k=3}^{\infty}\sum_{\mu=0}^{\infty}\sum_{\nu=1}^{\infty} P(d_{k-1}=\mu \wedge d_k=\nu)(\nu+1)(\mu-\nu+1) \qquad (4.1) \\ E_i(d_0,d_1) &= \sum_{\tau=1}^{\infty} \tau P(t=\tau) - P(\deg\gcd(d_0,d_1))=0) \qquad (4.2) \\ E^{\overline{u}}(d_0,d_1) &= \sum_{k=4}^{\infty}\sum_{\mu=0}^{\infty}\sum_{\nu=0}^{\infty} P(d_{k-2}=\mu \wedge d_{k-1}=\nu \wedge d_k \neq -\infty) \qquad (4.3) \\ &\quad (\mu-\nu+1)(d_1-\mu+1) \\ E^{\overline{v}}(d_0,d_1) &= \sum_{k=3}^{\infty}\sum_{\mu=0}^{\infty}\sum_{\nu=0}^{\infty} P(d_{k-2}=\mu \wedge d_{k-1}=\nu \wedge d_k \neq -\infty) \qquad (4.4) \\ &\quad (\mu-\nu+1)(d_0-\mu+1) \end{aligned}$$

The condition "$d_k \neq \infty$" in the expressions for $E^{\overline{u}}$ and $E^{\overline{v}}$ assures that the summation stops with $k = t$ instead of $k = t + 1$.

Now, to prove Theorem 4.2, it is sufficient to determine the various probabilities occurring in the above formulae. The key observation is the following lemma:

Lemma 4.3 *Let $f_1 \in \mathbb{F}_q[X]$ be uniformly distributed over the polynomials of degree α_1, $f_2 \in \mathbb{F}_q[X]$ be fixed of degree $\alpha_2 \leq \alpha_1$ and r the remainder and s the quotient of f_1 divided by f_2. Then r is uniformly distributed over the polynomials of degree less than α_2 and s is uniformly distributed over the polynomials of degree $\alpha_1 - \alpha_2$. Moreover, the distributions of r and s are independent. In the context of the Euclidean algorithm it follows for $\mu > \nu \geq 0$ that*

$$\begin{aligned} P(d_k=\nu | d_{k-1}=\mu) &= \frac{\#\text{ polynomials of degree } \nu}{\#\text{ polynomials of degree less than } \mu} \\ &= (q-1)q^{\nu-\mu}, \\ P(d_k=-\infty | d_{k-1}=\mu) &= q^{-\mu}. \end{aligned}$$

Proof: Note that for fixed f_2 the map $f_1 \mapsto r$ is an epimorphism of the additive group of polynomials of degree α_1 onto the additive group of polynomials of degree less than α_2. Since any preimage under a group epimorphism has the same cardinality, this proves the

assertion on r. More precisely, the pair of remainder and quotient polynomials (r, s) with $\deg r < \alpha_2$ and $\deg s = \alpha_1 - \alpha_2$ is obtained from exactly the polynomial $f_1 = sf_2 + r$, so that it occurs with probability $\frac{1}{(q-1)q^{\alpha_1}}$ independently of r and s. This shows the assertion on s and also proves the independence of the two probability distributions. □

These conditional probabilities can be used to determine the total probability of d_k admitting specific values:

Lemma 4.4

$$P(d_k = -\infty) = \begin{cases} \frac{1}{q^{d_1}} \sum_{j=0}^{k-2} \binom{d_1}{j} (q-1)^j & \text{for } 2 \le k \le d_1 + 1 \\ 1 & \text{for } k \ge d_1 + 2 \end{cases}$$

$$P(d_k = \nu) = \begin{cases} \frac{(q-1)^{k-1}}{q^{d_1}} q^\nu \binom{d_1 - \nu - 1}{k-2} & \text{for } \begin{array}{l} 2 \le k \le d_1 + 1, \\ 0 \le \nu \le d_1 - k + 1 \end{array} \\ 0 & \text{otherwise} \end{cases}$$

Proof: Observe first that $d_k \le d_{k-1} - 1$ implies $d_k \le d_1 - k + 1$ for $k \ge 2$; this proves the trivial parts of the assertion. We now restrict ourselves to the interesting cases and proceed by induction for $2 \le k \le d_1$, noting that the assertions for $k = 2$ follow directly from Lemma 4.3.

$$\begin{aligned} & P(d_{k+1} = -\infty) \\ &= \sum_{\mu=0}^{d_1-k+1} P(d_k = \mu) P(d_{k+1} = -\infty | d_k = \mu) + P(d_k = -\infty) \\ &= \sum_{\mu=0}^{d_1-k+1} \frac{(q-1)^{k-1}}{q^{d_1}} q^\mu \tbinom{d_1-\mu-1}{k-2} \frac{1}{q^\mu} + \frac{1}{q^{d_1}} \sum_{j=0}^{k-2} \tbinom{d_1}{j} (q-1)^j \\ &\quad \text{by the induction hypothesis and Lemma 4.3} \\ &= \frac{(q-1)^{k-1}}{q^{d_1}} \underbrace{\sum_{\mu=k-2}^{d_1-1} \binom{\mu}{k-2}}_{= \binom{d_1}{k-1}} + \frac{1}{q_{d_1}} \sum_{j=0}^{k-2} \tbinom{d_1}{j} (q-1)^j \end{aligned}$$

Let now $0 \leq \nu \leq d_1 - k$.

$$\begin{aligned} P(d_{k+1} = \nu) &= \sum_{\mu=\nu+1}^{d1-k+1} P(d_k = \mu) P(d_{k+1} = \nu | d_k = \mu) \\ &= \sum_{\mu=\nu+1}^{d1-k+1} \frac{(q-1)^{k-1}}{q^{d_1}} q^{\mu} \binom{d_1 - \mu - 1}{k-2} \frac{(q-1)q^{\nu}}{q^{\mu}} \\ &= \frac{(q-1)^k}{q^{d_1}} q^{\nu} \sum_{\mu=k-2}^{d_1-\nu-2} \binom{\mu}{k-2} \\ &= \frac{(q-1)^k}{q^{d_1}} q^{\nu} \binom{d_1 - \nu - 1}{k-1} \end{aligned}$$

□

The probability distribution of t has already been determined by Ma ([Ma87], Lemma 3.11); we give a simple proof in our setting:

Lemma 4.5

$$P(t = k) = \begin{cases} \binom{d_1}{k-1} \frac{(q-1)^{k-1}}{q^{d_1}} & \text{for } 1 \leq k \leq d_1 + 1 \\ 0 & \text{otherwise} \end{cases}$$

Proof: From the remark at the beginning of the proof of Lemma 4.4 it is clear that $t \leq d_1 + 1$. Hence, let $1 \leq k \leq d_1 + 1$. Then

$$\begin{aligned} P(t = k) &= P(d_{k-1} \neq -\infty) P(d_k = -\infty | d_{k-1} \neq -\infty) \\ &= \sum_{\mu=0}^{\infty} P(d_{k-1} = \mu) P(d_k = -\infty | d_{k-1} = \mu), \end{aligned}$$

and substituting the results of Lemmata 4.3 and 4.4 into this formula yields the desired expression. □

Proof of of Theorem 4.2: Lemmata 4.3 and 4.4 allow us to compute the joint probability distribution of sequences of remainder degrees; for instance,

$$\begin{aligned} &P(d_{k-2} = \mu \wedge d_{k-1} = \nu \wedge d_k \neq \infty) \\ &= P(d_{k-2} = \mu) P(d_{k-1} = \nu | d_{k-2} = \mu)(1 - P(d_k = -\infty | d_{k-1} = \nu)). \end{aligned}$$

Substituting the results of the lemmata into (4.1) to (4.4) yields the desired formulae after some tedious calculations with a symbolic algebra programme. Note then that the numerator of E_m has a double zero in 1, so it is divisible by $(q-1)^2$. Comparing degrees

of the numerator and denominator shows that E_m is indeed a polynomial in $1/q$. Similar reasonings apply to $E^{\overline{u}}$ and $E^{\overline{v}}$, and E_i is trivially a polynomial in $1/q$. □

Lemma 4.3 shows that with probability $1 - \frac{1}{q}$, we have $d_k = d_{k-1} - 1$ for $1 \leq d_{k-1} \leq d_1$. Hence the leading coefficients of E_m, E_i, $E^{\overline{u}}$ and $E^{\overline{v}}$ could alternatively be determined by examining this typical case.

While it is well known that the asymptotic worst-case complexity of the (extended) Euclidean algorithm is quadratic for the number of multiplications and linear for the number of inversions, Theorem 4.2 shows that this still holds in the average case and for small degree polynomials.

4.4 Some more probabilities

For the composition step of Section 3.3.2 we have to carry out further computations with the greatest common divisor and the multipliers, so that the probability distributions of their respective degrees are of interest. Recall from Section 3.3.1 that d_t denotes the degree of the greatest common divisor and that the multiplier degrees are related to d_{t-1} by (3.3) and (3.4).

Lemma 4.6

$$P(d_{t-1} = \mu \wedge d_t = \nu) = \begin{cases} \frac{1}{q^{d_1}} & \text{for } \mu = d_0 \text{ and } \nu = d_1 \\ \frac{q-1}{q^{d_1}} & \text{for } \mu = d_1 \text{ and } 0 \leq \nu < d_1 \\ \frac{(q-1)^2}{q^{\mu+1}} & \text{for } 1 \leq \mu < d_1 \text{ and } 0 \leq \nu < \mu \\ 0 & \text{otherwise} \end{cases}$$

$$P(d_{t-1} = \mu) = \begin{cases} \frac{1}{q^{d_1}} & \text{for } \mu = d_0 \\ d_1 \frac{q-1}{q^{d_1}} & \text{for } \mu = d_1 \\ \mu \frac{(q-1)^2}{q^{\mu+1}} & \text{for } 1 \leq \mu < d_1 \\ 0 & \text{otherwise} \end{cases}$$

$$P(d_t = \nu) = \begin{cases} \frac{1}{q^{d_1}} & \text{for } \nu = d_1 \\ \frac{q-1}{q^{\nu+1}} & \text{for } 0 \leq \nu < d_1 \\ 0 & \text{otherwise} \end{cases}$$

Proof: Observing that

$$\begin{aligned}
&P(d_{t-1} = \mu \wedge d_t = \nu) \\
&= \sum_{k=1}^{\infty} P(d_{k-1} = \mu) P(d_k = \nu | d_{k-1} = \mu) P(d_{k+1} = -\infty | d_k = \nu), \\
&P(d_{t-1} = \mu) = \sum_{\nu=0}^{\infty} P(d_{t-1} = \mu \wedge d_t = \nu) \quad \text{and} \\
&P(d_t = \nu) = \sum_{\mu=0}^{\infty} P(d_{t-1} = \mu \wedge d_t = \nu),
\end{aligned}$$

this is an easy consequence of Lemmata 4.3 and 4.4. □

The result for d_t has already been proved by Ma and von zur Gathen ([MG90], Proposition 2.5). Letting $\nu = 0$, the lemma shows that two random polynomials are coprime with probability $1 - 1/q$.

A further result is needed to analyse Cantor's reduction algorithm of Section 3.3.3; the reduction involves the computation of $\gcd(r_k, v_k)$, and these two polynomials are not independently distributed, so that the previous lemma cannot be applied. However, it is possible to give an estimate of the probability that r_k and v_k are coprime.

Lemma 4.7 *Let $k \geq 2$ and $d_{k-1} > 0$. Then*

$$P(r_k \text{ and } v_k \text{ coprime}) \geq 1 - \frac{d_0 - 1}{q}.$$

Proof: Observe first that v_k and v_{k-1} are coprime for $k \geq 1$. The assertion trivially follows for $k = 1$ from $v_1 = 1$ and can be proved by induction for $k \geq 2$, since the formula for v_{k+1} in Algorithm 3.24 implies that $\gcd(v_{k+1}, v_k) = \gcd(v_k, v_{k-1})$.

Let π be an irreducible divisor of r_k. Then π also divides

$$v_k = v_{k-2} - q_{k-1} v_{k-1}$$

if and only if π is a common divisor of v_{k-1} and v_{k-2}, which is impossible by the observation above, or if π does not divide v_{k-1} and $q_{k-1} \equiv v_{k-1}^{-1} v_{k-2} \pmod{\pi}$. By Lemma 4.3, q_{k-1} is uniformly distributed over a space of non-constant polynomials of fixed degree, and this distribution is independent of r_k and thus of π. So the probability that q_{k-1} satisfies the congruence is at most $\frac{1}{q}$. Since r_k has no more than $d_k \leq d_0 - 1$ distinct irreducible factors, the assertion follows immediately. □

4.5 Average number of field operations

In this section we determine the average number of field multiplications and inversions performed during the different algorithms of Sections 3.3.2 and 3.3.3. We give correct asymptotic formulae for $q \to \infty$, i.e., we do not take into account events that have probability in $\frac{1}{q}O(1)$.

We have to distinguish several cases. In odd characteristic we assume $v = 0$, while in even characteristic we have $v \neq 0$ and the exact complexities depend on $\deg v$; a complete description would comprise the cases $v = 1$, $\deg v = g$, $\deg v = g - 1$ and $1 \leq \deg v \leq g - 2$. In order to keep the presentation concise we concentrate on two cases especially attractive for implementation, namely the curves with $v = 1$ and the curves with $\deg v = 1$, i.e. $v = X$.

Concerning the polynomial arithmetic, it should be noted that squaring a polynomial is more efficient than multiplying two distinct polynomials. In characteristic 2, squaring a polynomial of degree n takes $n + 1$ squarings in the underlying field since

$$\left(\sum_{i=0}^{n} a_i X^i\right)^2 = \sum_{i=0}^{n} a_i^2 X^{2i}.$$

Assuming that the field arithmetic is implemented using normal bases, these squarings are essentially for free, hence we do not count them (see [Jun93], Section 3.3). In odd characteristic, we can write

$$\left(\sum_{i=0}^{n} a_i X^i\right)^2 = \sum_{i=0}^{n} a_i^2 X^{2i} + \sum_{i=0}^{n} \sum_{j=0}^{i-1} (a_i a_j + a_i a_j) X^{i+j}$$

and need $(n + 1) + \frac{1}{2}n(n + 1) = \frac{1}{2}(n + 1)(n + 2)$ field multiplications.

All complexities given are quadratic or cubic polynomials in g with fixed leading coefficient, but the other coefficients may vary according to the arrangement of the computations. Exchanging two multiplications, for instance, may already have an impact. Since all proofs are very alike, we present only one proof in detail and proceed to simply outline the order of the computations in the subsequent theorems, leaving the details to the reader. While much care has been taken to find the optimal arrangements of computations, it is possible that sometimes better ones exist. We denote the characteristic of $K = \mathbb{F}_q$ by p.

According to Section 4.2, we assume that the composition algorithm is applied to uniformly distributed polynomials a_1, a_2, b_1 and b_2 of degrees g, g, $g - 1$ and $g - 1$, respectively. As for the explicit formulae known for elliptic curves, we have to distinguish the cases that a divisor is doubled or that two distinct divisors are added.

Theorem 4.8 *On average, two distinct reduced divisors can be composed with*

- $8g^2 + 5g - 2 + \frac{1}{q}O(g^2)$ *multiplications and* $g + 2 + \frac{1}{q}O(g)$ *inversions for* p *odd;*
- $7g^2 + 7g - 1 + \frac{1}{q}O(g^2)$ *multiplications and* $g + 1 + \frac{1}{q}O(g)$ *inversions for* $p = 2$, $v \in \{1, X\}$.

On average, a reduced divisor can be composed with itself with

- $7g^2 + 7g - 1 + \frac{1}{q}O(g^2)$ *multiplications and* $g + 1 + \frac{1}{q}O(g)$ *inversions for* p *odd;*
- $4g + 2$ *multiplications and one inversion for* $p = 2$, $v = 1$*;*
- $4g^2 + 5g + 2$ *multiplications and two inversions for* $p = 2$, $v = X$.

Proof: Assume the notation of Algorithm 3.25. We first analyse the composition step for distinct divisors and assume that $p \neq 2$. The average number of multiplications during the computation of $\gcd(a_1, a_2)$ and u_1 is in $2g^2 + g - 3 + \frac{1}{q}O(g^2)$, the average number of inversions is in $g + \frac{1}{q}O(g)$ by Theorem 4.2. The greatest common divisor has, by Lemma 4.6, degree zero with probability $1 - \frac{1}{q}$. We can thus assume this case, since otherwise we apply the Euclidean algorithm a second time and all further computations are covered by the "$\frac{1}{q}O(g^2)$" and "$\frac{1}{q}O(g)$" terms. So we suppose $u_3 = 0$. Moreover, Lemma 4.6 shows that $\deg u_1 = g - 1$ is most likely. Compute d^{-1} and then $a = ((d^{-1})^2 a_1)a_2$ with $1+(g+1)+(g+1)^2$ multiplications and $b_1+((d^{-1}u_1)(b_2-b_1))a_1$ with $g+g^2+(2g-1)(g+1)$ multiplications. Since the result is most likely of degree $3g - 2$ and a of degree $2g$, the reduction modulo a needs additional $(2g + 1)(g - 1)$ multiplications and one inversion, which proves the desired result.

The same reasoning applies to the case $p = 2$, but a different approach is preferable. Compute $d = \gcd(a_2, b_1 + b_2 + v)$, d^{-1} and u_3 with expected $2g^2 - 4 + \frac{1}{q}O(g^2)$ multiplications and $g + \frac{1}{q}O(g)$ inversions, and $a = ((d^{-1})^2 a_1)a_2$ with $(g + 1) + (g + 1)^2$ multiplications. Reduce u modulo a to obtain $\tilde{u}$ by $2(2g + 1)$ multiplications and one inversion. Compute $\tilde{b} = b_1^2 + b_1 v$ for free since squaring and multiplying by $v \in \{1, X\}$ are free, then $b_1 + (d^{-1}u_3)(\tilde{b} + \tilde{u})$ with $g + 2g^2$ multiplications, noticing that the degree of u_3 is most likely $g - 1$ by Lemma 4.6 and the degree of $\tilde{b} + \tilde{u}$ most likely $2g - 1$ by Lemma 4.3. The final reduction modulo a needs again $(2g + 1)(g - 1)$ multiplications, but this time no inversion since the inverted leading coefficient of a is still known from the reduction of u. This approach needs fewer multiplications as soon as $g \geq 3$. For $g = 2$, it saves one inversion at the expense of two multiplications, which is usually faster (see the references in Section 4.6).

The same approach of computing $\gcd(a_1, 2b_1 + v)$ must be taken for doubling a point, which results in the other complexities given. Notice that a should be computed as

$(d^{-1}a_1)^2$. For $p = 2$ and $v = X$, the values of d and u_3 are trivial to determine; the given complexity is then valid for the most probable case $\deg d = 0$, the case $d = X$ needs even fewer operations. If $p = 2$ and $v = 1$, then we even have $d = u_3 = 1$, which implies a further simplification. □

In order to apply the results of Section 4.3 and in accordance with Section 4.2 we assume that the composition algorithm has yielded a random semireduced divisor $\mathrm{div}(a, b)$ of degree $2g$, so that a and b are approximately uniformly distributed over all polynomials of degree $2g$ and $2g - 1$, respectively.

Theorem 4.9 *On average, Gauß reduction can be executed with*

- $\frac{7}{4}g^3 + \frac{33}{8}g^2 + \frac{1}{4}g + \frac{1}{q}O(g^4)$ *multiplications and* $\frac{1}{2}g + 1 + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *even;*
- $\frac{7}{4}g^3 + \frac{39}{8}g^2 + \frac{9}{4}g + \frac{17}{8} + \frac{1}{q}O(g^4)$ *multiplications and* $\frac{1}{2}g + \frac{3}{2} + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *odd;*
- $\frac{7}{6}g^3 + 3g^2 - \frac{1}{6}g + \frac{1}{q}O(g^4)$ *multiplications and* $\frac{1}{2}g + 1 + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$*,* $v \in \{1, X\}$*,* g *even;*
- $\frac{7}{6}g^3 + \frac{7}{2}g^2 + \frac{4}{3}g + 2 + \frac{1}{q}O(g^4)$ *multiplications and* $\frac{1}{2}g + \frac{3}{2} + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$*,* $v \in \{1, X\}$*,* g *odd.*

Proof: Assuming the notation of Algorithm 3.26, Lemma 4.3 shows that with probability in $1 - \frac{1}{q}O(g)$ we have $t = \lceil \frac{g}{2} \rceil$, $\deg a_k = 2(g - k)$ and $\deg b_k = 2(g - k) - 1$ for $k < t$ and $\deg a_t = g$. Now some simple computations reveal the results above. □

If $g = 2$, then in the Cantor reduction step we have $c = b$ and $d = 1$, so that it equals Gauß reduction. Also, Lagrange reduction differs from Gauß reduction only if $g \geq 3$, so that henceforth we assume $g \geq 3$.

Theorem 4.10 *If* $g \geq 3$*, Cantor reduction is successful with probability at least* $1 - \frac{1}{q^{g+1}}$*. On average, it can then be executed with*

- $11g^2 + 3g - 8 + \frac{1}{q}O(g^3)$ *multiplications and* $g + 1 + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *even;*
- $11g^2 + 7g - 4 + \frac{1}{q}O(g^3)$ *multiplications and* $g + 2 + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *odd;*
- $\frac{21}{2}g^2 + g - 9 + \frac{1}{q}O(g^3)$ *multiplications and* $g + 1 + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$*,* $v \in \{1, X\}$*,* g *even;*

- $\frac{21}{2}g^2 + 6g - \frac{9}{2} + \frac{1}{q}O(g^3)$ *multiplications and* $g + 2 + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$, $v \in \{1, X\}$, g *odd.*

Proof: Assume the notation of Algorithm 3.28. Cantor reduction is successful at least when $\deg(\gcd(a, b)) \leq g$, which by Lemma 4.6 happens with probability $1 - \frac{1}{q^{g+1}}$. In this case, applying the extended Euclidean algorithm 3.24 to a and b without computing the u_k and stopping as soon as the desired degree is reached yields c and d and requires

$$\sum_{k=2}^{t'}(d_{k-1} + 1)(d_{k-2} - d_{k-1} + 1) + \sum_{k=3}^{t'}(d_{k-2} - d_{k-1} + 1)(d_0 - d_{k-2} + 1)$$

multiplications and $t' - 1$ inversions, where by Lemma 4.3 the probability that all $d_k = 2g - k$ and $t' = \lceil \frac{g}{2} \rceil$ is in $1 - \frac{1}{q}O(g)$. Compute $\tilde{a}$ and apply the half-extended Euclidean algorithm to $\tilde{a}$, which is usually of degree g, and d, which is usually of degree $\lceil \frac{g}{2} \rceil - 1$, to compute the greatest common divisor s and $\tilde{d}$. By Lemma 4.7 we most probably have $\deg s = 0$, so that $\overline{a} = \tilde{a}$ and it is sufficient to compute $\overline{b} = (-v - \overline{d}(c \bmod \overline{a}) \bmod \overline{a}$ to obtain the reduced divisor $\mathrm{div}(\overline{a}, \overline{b})$. □

Theorem 4.11 *If* $g \geq 3$, *then on average Lagrange reduction can be executed with*

- $9g^2 - 5 + \frac{1}{q}O(g^3)$ *multiplications and* $\frac{1}{2}g + 1 + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *even;*
- $9g^2 + g - 2 + \frac{1}{q}O(g^3)$ *multiplications and* $\frac{1}{2}g + \frac{3}{2} + \frac{1}{q}O(g^2)$ *inversions for* p *odd,* g *odd;*
- $7g^2 - g - 5 + \frac{1}{q}O(g^3)$ *multiplications and* $\frac{1}{2}g + 1 + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$, $v \in \{1, X\}$, g *even;*
- $7g^2 - 2 + \frac{1}{q}O(g^3)$ *multiplications and* $\frac{1}{2}g + \frac{3}{2} + \frac{1}{q}O(g^2)$ *inversions for* $p = 2$, $v \in \{1, X\}$, g *odd.*

Proof: The analysis is straightforward since by Lemma 4.3 we have $t = \lceil \frac{g}{2} \rceil$, $\deg a_k = 2(g - k)$, $\deg b_k = 2(g - k) - 1$ and $\deg q_k = 1$ for $k \leq \lfloor \frac{g}{2} \rfloor$ with total probability in $1 - \frac{1}{q}O(g)$. If g is odd, then $\deg a_t = g$ and $\deg q_t = 0$ with overwhelming probability. □

To estimate the number of field operations required for a full addition step we have to choose between the three reduction procedures. Comparison of the numbers in Theorems 4.9 to 4.11 yields the following result.

Theorem 4.12 *For large* q, *Lagrange reduction is the fastest reduction procedure.*

We can now use the results of Theorems 4.8, 4.9 and 4.11 to determine the number of field operations needed in the different cases.

Theorem 4.13 *On average, two distinct divisors can be added with the following numbers of field operations:*

	multiplications	*inversions*
$p \neq 2$, g *even*	$17g^2 + 5g - 7 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + 3 + \frac{1}{q}O(g^2)$
$p \neq 2$, g *odd*	$17g^2 + 6g - 4 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + \frac{7}{2} + \frac{1}{q}O(g^2)$
$p = 2$, g *even*	$14g^2 + 6g - 6 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + 2 + \frac{1}{q}O(g^2)$
$p = 2$, g *odd*	$14g^2 + 7g - 3 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + \frac{5}{2} + \frac{1}{q}O(g^2)$

On average, a divisor can be doubled with the following numbers of field operations:

	multiplications	*inversions*
$p \neq 2$, g *even*	$16g^2 + 7g - 6 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + 2 + \frac{1}{q}O(g^2)$
$p \neq 2$, g *odd*	$16g^2 + 8g - 3 + \frac{1}{q}O(g^3)$	$\frac{3}{2}g + \frac{5}{2} + \frac{1}{q}O(g^2)$
$p = 2$, $v = 1$, g *even*	$7g^2 + 3g - 3 + \frac{1}{q}O(g^3)$	$\frac{1}{2}g + 2 + \frac{1}{q}O(g^2)$
$p = 2$, $v = 1$, g *odd*	$7g^2 + 4g + \frac{1}{q}O(g^3)$	$\frac{1}{2}g + \frac{5}{2} + \frac{1}{q}O(g^2)$
$p = 2$, $v = X$, g *even*	$11g^2 + 4g - 3 + \frac{1}{q}O(g^3)$	$\frac{1}{2}g + 3 + \frac{1}{q}O(g^2)$
$p = 2$, $v = X$, g *odd*	$11g^2 + 5g + \frac{1}{q}O(g^3)$	$\frac{1}{2}g + \frac{7}{2} + \frac{1}{q}O(g^2)$

Conclusion 4.14 *For large q, the arithmetic in hyperelliptic Jacobians of genus at least 2 is faster in even characteristic with $v \in \{1, X\}$ than in odd characteristic. The number of field multiplications and inversions strictly increases with the genus.*

4.6 Average bit complexity

While Theorem 4.13 shows that the number of field operations increases strictly as the genus grows, the cryptographic setting described in Section 4.1 implies that at the same time the field size decreases because $\log q = \frac{l}{g}$ for some constant security parameter l. Hence the field operations may become more efficient, and the bit complexity of adding divisors depends very much on the implementation of the field arithmetic. We examine three basic situations in which the complexity of field operations is constant or grows with $\log q$ or $\log^2 q$.

Throughout this section we assume that l is "sufficiently large" and g "not too large" compared to l, so that the terms "$\frac{1}{q}O(g^4)$" etc. in the complexities can be neglected.

As justified in Section 4.1, this condition is fulfilled in practice. To assure mathematical rigour, we may assume that $g \leq l^\varepsilon$ for some constant $\varepsilon < 1$. Then

$$\frac{1}{q}O(g^4) = \frac{g^4}{e^{l/g}}O(1) \subseteq \frac{l^{4\varepsilon}}{(e^{1-\varepsilon})^l}O(1) \subseteq o(1) \text{ for } l \to \infty,$$

and the following discussion yields asymptotic results for $l \to \infty$.

If the finite field is so small that its elements can be represented in single registers and the field operations can be performed with single precision, then they take constant time, and the assertion of Theorem 4.13 directly carries over to the bit complexity.

In custom-designed hardware for finite field arithmetic the complexity of the field operations usually grows with $\log q$. Thus, assume that a multiplication needs time $m \log q$ and an inversion $i \log q$. As we get technical problems when comparing the arithmetics for genus g and $g+1$, one being even and one odd, and a too fine-grained analysis is not even meaningful in our rather coarse model, we restrict our comparison to either odd or even genus, i.e. remain within one row of the tables in Theorem 4.13. Then, an operation in the Jacobian can be executed in an average time of

$$(\alpha g^2 + \beta g - \gamma)m \log q + (\delta g + \eta)i \log q = l\left(\left(\alpha g + \beta - \frac{\gamma}{g}\right)m + \left(\delta + \frac{\eta}{g}\right)i\right)$$

for some non-negative constants α, β, γ, δ, η. The derivative of this expression with respect to g is positive whenever

$$\frac{i}{m} < \frac{4\alpha + \gamma}{\eta} \leq \frac{\alpha g^2 + \gamma}{\eta}.$$

Plugging in the actual values of Theorem 4.13 shows that this is the case for $i/m \leq 11$, which is fulfilled in practice (cf. [Ber68], Chapters 2.3 and 2.4). Hence in this situation the arithmetic in hyperelliptic Jacobians becomes slower as the genus grows.

In a generic software implementation, the field operations usually take time proportional to $\log^2 q$, say a multiplication needs $m \log^2 q$ and an inversion $i \log^2 q$. Then the operations in the Jacobians take time

$$l^2 m\left(\alpha + \frac{\beta g - \gamma}{g^2}\right)$$

for the field multiplications and

$$l^2 i \frac{\delta g + \eta}{g^2}$$

for the inversions. Thus the effort for field inversions is strictly decreasing. Also, with the constants set forth in Theorem 4.13, the effort for multiplications is strictly decreasing except for the step between $g = 2$ and $g = 4$ in odd characteristic. This conclusion, however, has to be approached with care, as it is based on the non-leading coefficients

β and γ, which are bound to change already when two multiplications are interchanged. Anyway, the relative speed-up obtainable from the multiplications is very limited. Considering, for instance, the doubling step for $p = 2$, $v = 1$ and g even, we observe that the total effort for multiplications varies between $7.75ml^2$ (for $g = 2$) and $7ml^2$ (for $g \to \infty$). The effect is even less pronounced in the other cases with a higher value of α.

How advantageous it is with respect to the inversions to increase the genus depends on the ratio i/m. The larger this ratio is, the more one can profit from a higher genus. In any case, the hyperbolic shape of the complexity curve results in fast dropping marginal profits.

This discussion shows that given a desired security level, i.e. an appropriate group size, hyperelliptic cryptosystems should preferably be implemented in characteristic 2. If the complexity of the field operations is constant or grows with $\log q$, then the smallest possible genus fitting the system requirements should be chosen. If the complexity of the field operations grows with $\log^2 q$, a higher genus, to be determined by computational experiments, might be recommendable. For instance, the smallest genus such that the field elements fit into one machine word could be a good choice; from then on, the efficiency would drop again.

In any case, due to optimised formulae available for adding points on elliptic curves, hyperelliptic arithmetic based on the generic algorithms of Section 3.3 is less efficient than elliptic curve arithmetic. It might be worthwhile to investigate explicit formulae for the addition law in low-genus hyperelliptic Jacobians. Spallek, for instance, reports such an implementation of genus 2 curves which is slower than elliptic curve arithmetic by a factor of only 2.5 for $q^g \approx 10^{40}$ ([Spa94], p. 31). Further experimental results can be found in [Kri97].

Chapter 5

Smoothness

Our goal in this and the following chapters is to develop a general theory for subexponential discrete logarithm algorithms and to apply this theory to class groups, especially of hyperelliptic curves. The general approach is motivated by the fact that subexponential algorithms for computing discrete logarithms are scattered in the literature and described over and over again in different settings. The algorithms to be developed in Chapters 6 and 7 provide a framework which is as independent of the concrete group as possible. It covers all examples known from the literature and is successfully applied to hyperelliptic Jacobians, for which we provide the first algorithm with a fully proved subexponential running time. It is hoped that this framework allows a concise presentation of subexponential algorithms yet to be discovered in further groups, in the sense that it alleviates the burden of proving those parts of the algorithms which remain unchanged and that only the details relating to the particular group have to be filled in.

Taking the example of a finite prime field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ of Section 1.3 as a starting point, we note that the algorithm relies on the fact that first each element of $\mathbb{F}_p$, when represented by a natural integer, is composed of building blocks, i.e, it is the product of prime elements, and second, it is so in a non-unique way: From a relation of the form

$$\prod p_i^{\mu_i} \equiv \prod p_i^{\nu_i} \pmod{p}$$

we deduce a relation

$$\sum (\mu_i - \nu_i) \log p_i = 0$$

of discrete logarithms. In Section 5.1 we develop a general model for this particular behaviour. This model allows to *formulate* the algorithms of Chapters 6 and 7 in a terminology independent of the concrete group. However, the algorithms are probabilistic, and their success probability — and thus ultimately their running time — depend on

an additional *smoothness assumption*, as also mentioned in Section 1.3. Basically, the assumption is that a sufficiently large proportion of the group elements is *smooth*, i.e. composed of small building blocks. Whether the smoothness assumption indeed holds depends on the concrete group, and its verification is precisely the part which must be filled in for each group separately. So with the general framework of Chapters 6 and 7, the proof of subexponentiality of the discrete logarithm algorithm for a particular group boils down to verifying the validity of a smoothness result in the group.

The subsequent sections of this chapter deal with the smoothness assumption in the case of hyperelliptic curves. We obtain the first such result for function fields.

5.1 Arithmetical semigroups and formations

Decomposability of elements into building blocks can be modelled by a free abelian monoid $\mathbb{M}$, written additively, over a countable set $\mathcal{P}$, whose elements are called *primes*. To lay the grounds for the smoothness concept, we need a notion of *size* which is compatible with the structure of $\mathbb{M}$. Thus, let

$$\deg : \mathbb{M} \to \mathbb{R}^+$$

be a homomorphism of monoids, which to each element of $\mathbb{M}$ associates its size. As $\mathbb{M}$ is free over the set of primes, any such homomorphism is given by assigning a non-negative size to each prime and extending additively. In all practical examples, the size function has a natural interpretation; in particular, the size $\deg(m)$ of an element $m \in \mathbb{M}$ is usually closely related to its bit size in a computer representation in the sense that the latter is in $\Theta(\deg(m))$. The setting above was introduced by Knopfmacher in [Kno75]; he calls $\mathbb{M}$ an *additive arithmetical semigroup*.

So far, the decomposition of an element into a sum of primes is unique, but as mentioned in the introduction to this chapter, we need some ambiguity to deduce the logarithms of the prime elements. Assume that there is an equivalence relation $\sim$ on $\mathbb{M}$ which is compatible with its composition law such that $\mathbb{M}/\sim$ is a finite abelian group. Then $\mathbb{M}/\sim$ is called an *arithmetical formation* in [Kno75], and we assume that the group G in which discrete logarithms are sought has this structure.

Knopfmacher was mainly interested in generalisations of the prime number theorem and asymptotic number theoretic results to this more general situation, i.e. in counting problems. Since we need to work with and compute in concrete arithmetical formations, we require that they meet further restrictions, which basically ensure that the arithmetic is of polynomial complexity.

Let N be the order of G. In general, one would expect that the elements of G are represented by $O(\log N)$ bits and measure algorithmic complexities as functions of N. It is, however, possible to construct groups whose elements are naturally represented

by bit strings of length in $O(\log N')$ for some value N' considerably larger than N. For instance, Jacobian groups of hyperelliptic curves of genus g over $\mathbb{F}_2$ are given by $\Theta(\log N')$ bits for $N' = 2^g$, whereas the only previously known lower bound on N in this case is the Hasse–Weil bound $(\sqrt{2}-1)^g \leq 1$. While this situation results in a waste of bandwidth for cryptographic applications and is thus unlikely to occur, for preserving as much generality as possible we henceforth denote by $\log N'$ the input size of the problem and measure all complexities by functions in $\log N'$. It turns out that factors polynomial in $\log N'$ do not affect the subexponential running time. Hence to simplify the analysis we follow [GG99] and for some positive function f of N' denote by $O^\sim(f)$ the class of functions which are in $O(f)$ up to a factor bounded by some power of $\log N'$. Formally,

$$O^\sim(f) = \bigcup_{k=0}^{\infty} O(f(N') \log^k N').$$

To ensure that on the other hand N' is not much smaller than N, we require $N \in O^\sim(N')$.

To be able to write down group elements, we demand that each element of G has a canonic representative in $\mathbb{M}$ of bit size in $O^\sim(1)$. Thus, the elements of G inherit the decomposability into a sum of primes from their canonic representatives, and we interpret $\deg(g)$ as the size of the canonic representative of $g \in G$. (At first sight, this reintroduces the unique decomposability of an element of G into prime elements. Observe, however, that if $m \equiv m_1 + m_2 \pmod{\sim}$, then the sum of the decompositions of m_1 and m_2 is a further decomposition of the element m of G.) Furthermore, the arithmetic in G, i.e. addition, negation and test for equality, must be performed by manipulating the canonic representatives in time polynomial in $\log N'$. In addition, we require that $\deg(p) \geq 1$ for $p \in \mathcal{P}$ and that $\deg(g) \in O^\sim(1)$ for any $g \in G$, so that the number of primes in the decomposition of a group element, counting multiplicities, is also bounded above by $O^\sim(1)$.

For a *smoothness bound* $S \in \mathbb{N}$ denote by $\mathcal{P}_S$ the set of primes of size at most S and by n_S the cardinality of $\mathcal{P}_S$. An element of G is called *S–smooth* if its decomposition involves only primes of $\mathcal{P}_S$. As the size of the elements of G is in $O^\sim(1)$, a distinction into smooth and non-smooth elements arises only for $S \in O^\sim(1)$, which we henceforth assume. For technical reasons we need $\log n_S \in O^\sim(1)$. From an algorithmic point of view, we require that $\mathcal{P}_S$ can be constructed in $O^\sim(n_S^2)$. (If n'_S denotes the number of elements of the monoid $\mathbb{M}$ of size at most S, then we often have $O^\sim(n_S) = O^\sim(n'_S)$. The quadratic complexity of the construction of $\mathcal{P}_S$ can then be achieved by enumerating all elements of $\mathbb{M}$ of size not exceeding S and trial division by the smaller elements. In imaginary quadratic number and function fields, $\mathcal{P}_S$ is constructed in a different way, see the examples below.) Furthermore, we suppose that the elements of G can be tested for S–smoothness and, if possible, be decomposed into a sum of primes from $\mathcal{P}_S$ in $O^\sim(n_S)$, which usually amounts to trial division by the elements of $\mathcal{P}_S$. In all cases considered below, the smoothness test and the decomposition are even in $O^\sim(\sqrt{n_S})$ or $O^\sim(1)$, which results in better running times.

The most efficient smoothness test available for integers to date, which is subexponential in $\log n'_S$, is non-deterministic and not completely reliable in the sense that it may not recognise a smooth element. Thus, we extend our model as follows: The smoothness test rejects all non-smooth elements; it recognises a smooth element up to a certain error probability, which may depend on the element tested, but does not exceed 1/2.

It should be noted that we could work with a more general, but less intuitive definition of decomposability into a sum of primes without involving the quotient of a free abelian monoid and of smoothness without involving the notion of size. In fact, the set of primes could be assumed to be either the finite set $\mathcal{P} = \{1, \ldots, k\}$ or $\mathcal{P} = \mathbb{N}$, and the decomposition into a sum of primes could be seen as some map from G to the free abelian monoid over $\mathcal{P}$, i.e. to the sequences $(\nu_i)_{i\in\mathcal{P}}$ with $\nu_i \in \mathbb{N}_0$ and almost all ν_i equal zero. Then, for a finite set $\mathcal{P}_S \subseteq \mathcal{P}$, an element would be called $\mathcal{P}_S$–smooth if $\nu_i = 0$ for $i \notin \mathcal{P}_S$.

However, all examples considered in the literature are covered by the more intuitive concept, and I feel that stripping off more meaning than necessary does rather obscure the matter.

Examples.

1) Finite prime fields $G = \mathbb{F}_p^\times$

   G can be represented as $(\mathbb{N}, \cdot)/\sim$, where $m_1 \sim m_2$ if and only if $p|m_1 - m_2$, and $\mathcal{P}$ is the set of natural prime numbers. The size of an element is given by its logarithm to the base 2, $\deg(m) = \operatorname{ld} m$, and $N' = N = p - 1$.

2) Finite fields $G = \mathbb{F}_{2^k}^\times$ of characteristic 2

   G can be represented as $(\mathbb{F}_2[X]\backslash\{0\}, \cdot)/\sim$, where $f_1 \sim f_2$ if and only if $f|f_1 - f_2$ for some fixed irreducible polynomial f of degree k in $\mathbb{F}_2[X]$, and $\mathcal{P}$ is the set of irreducible polynomials over $\mathbb{F}_2$. The size of an element is given by its usual degree and $N' = N = 2^k - 1$.

3) Finite fields $G = \mathbb{F}_{p^k}^\times$, p prime

   G can be represented by the polynomials of degree less than k over $\mathbb{F}_p$. Denote by $\mathbb{F}_p[X]'$ the set of monic polynomials over $\mathbb{F}_p$. Noticing that any polynomial is the unique product of its leading coefficient and a monic polynomial, G can be represented as $(\mathbb{N}, \cdot) \times (\mathbb{F}_p[X]', \cdot)/\sim$, where $(m_1, f_1) \sim (m_2, f_2)$ if and only if $p|m_1 - m_2$ and $f|f_1 - f_2$ for some fixed irreducible polynomial f of degree k over $\mathbb{F}_p$. The set of primes $\mathcal{P}$ is given by the union of the set of natural primes and the set of monic irreducible polynomials over $\mathbb{F}_p$, each embedded into the cartesian product. The size of an element is $\deg(m_1, f_1) = \operatorname{ld} m_1 + \deg f_1$ and $N' = N = p^k - 1$. Notice that these definitions are compatible with Examples 1) and 2).

4) Jacobians of imaginary quadratic hyperelliptic curves

The Jacobian G of an imaginary quadratic hyperelliptic curve C is isomorphic to $\mathbb{M}/\sim$, where $\mathbb{M}$ is the free abelian monoid over the set $\mathcal{P}$ of finite prime divisors of $K(C)$ and $\sim$ is the equivalence relation induced by the divisors of principal ideals of $\mathcal{O} = K[X,Y]/(C)$, see Proposition 3.7. The size of a divisor of $\mathbb{M}$ is given by its degree.

Since inert prime ideals belong to principal ideals, we may in fact remove them from $\mathcal{P}$. Thus, $\mathcal{P}_S$ consists of all prime divisors of $K(C)$ extending finite ramified and splitting prime divisors of $K(X)$ whose local parameters are of degree at most S. It can be constructed by enumerating all irreducible polynomials of $K[X]$ of degree at most S in time $O^\sim(q^{2S})$ and determining their respective extensions by Kummer's theorem 3.10 in polynomial time. Notice that $O^\sim(q^S) = O^\sim(n_S)$ by Theorem 5.2.

The elements of G are represented by reduced divisors $\mathrm{div}(a,b)$, which satisfy $\deg(\mathrm{div}(a,b)) = \deg a \leq g$, see Section 3.2.5. So the input size of the problem is $O(\log N')$ for $N' = q^g$, for which the assumption $N \in O^\sim(N')$ holds indeed:

Proposition 5.1 *In a hyperelliptic curve of genus g over $\mathbb{F}_q$, the ideal class number N satisfies*

$$N \leq (2g+1)q^g.$$

Proof: For q a prime, the result is due to Artin ([Art24b], §24, Formula (8)). His arguments are easily extended to the general case replacing the Artin character by the general quadratic character. □

A reduced divisor $\mathrm{div}(a,b)$ is S–smooth if and only if all irreducible factors of a are of degree at most S; its decomposition into prime divisors is obtained from the factorisation of a as explained in Corollary 3.17. So the smoothness test and the decomposition are reduced to the analogous problems over the univariate polynomials. For $K = \mathbb{F}_q$ a finite field, these can be solved by probabilistic algorithms in expected time polynomial in $\log q$ and S.

5) Ideal class groups of imaginary quadratic number fields

Let $K = \mathbb{Q}(\sqrt{D})$ be an imaginary quadratic number field of discriminant $D < 0$ and $\mathcal{O}$ its ring of integers. The class group G of K is defined as $\mathbb{M}/\sim$, where $\mathbb{M}$ is the set of integral ideals of $\mathcal{O}$ (a free abelian monoid over the set $\mathcal{P}$ of prime ideals), and $\sim$ is induced by the principal ideals.

Let $y = \frac{D+\sqrt{D}}{2}$ with minimal polynomial $Y^2 - DY + \frac{D^2-D}{4}$. Then y generates an integral power basis, i.e., $\mathcal{O} = \mathbb{Z} + \mathbb{Z}y$ (cf. Proposition 3.8). Since the unit rank of $\mathcal{O}$ is zero, it is possible to develop a theory exactly analogous to those of imaginary quadratic hyperelliptic curves (cf. Section 2.4.2). In particular, any ideal class has a unique reduced representative $(a, y-b)$ with $a,\ b \in \mathbb{Z}$, $\frac{-a+D}{2} < b \leq \frac{a+D}{2}$,

$a|b^2 - Db + \frac{D^2-D}{4}$, $a^2 \leq b^2 - Db + \frac{D^2-D}{4}$ and $b > \frac{D}{2}$ if $a^2 = b^2 - Db + \frac{D^2-D}{4}$. (This definition of reducedness may look unfamiliar. It is more common to consider ideals of the form $\left(a, \frac{\sqrt{D}-b'}{2}\right)$ with $b' \equiv D \pmod 2$ and $a|(b')^2 - D$, which can be summarised as $4a|(b')^2 - D$. Letting $c = \frac{(b')^2-D}{4a}$, an ideal is called reduced if $-a < b' \leq a \leq c$ and $b > 0$ if $a = c$. From $\frac{\sqrt{D}-b'}{2} = y - b$ for $b = \frac{b'+D}{2}$ it is easily seen that the two definitions of reducedness coincide). These conditions imply $a < \sqrt{\frac{D}{3}}$, which is the analogue of $\deg a \leq g$ for reduced hyperelliptic divisors.

The arithmetic of the ideal class group can be realised by polynomial time algorithms as in Section 3.3. The norm of an integral ideal of $\mathcal{O}$ is a certain principal ideal of $\mathbb{Z}$ (cf. Proposition 3.15); to simplify the notation, it is common to define the norm of an ideal as the unique positive generator of this ideal of $\mathbb{Z}$, so that the norm of $(d)(a, y - b)$ with $a|b^2 - Db + \frac{D^2-D}{4}$ is $d^2|a|$. Then we can define the size of an ideal as the dual logarithm of its norm. Thus $\mathcal{P}_S$ consists of all prime ideals extending splitting or ramified rational primes not larger than 2^S and of all prime ideals extending inert rational primes not larger than $\sqrt{2^S}$. Again, we let $N' = N$. Then these rational primes can be enumerated in $O^\sim\left(2^{2S}\right)$, which equals $O^\sim(n_S^2)$ under the generalised Riemann hypothesis. The set $\mathcal{P}_S$ can then be constructed with the help of Kummer's theorem 3.10 for number fields: For a rational prime p, find the roots of $Y^2 - DY + \frac{D^2-D}{4} \pmod p$. If there is no root, then p is inert and $p\mathcal{O}$ is a principal prime ideal. If there is the double root $b + (p)$, then p is ramified and $(p, y - b)$ is the unique extension of p. If there are two roots $b + (p)$ and $b' + (p) = D - b + (p)$, then p is splitting and $(p, y - b)$ and $(p, y - b')$ are the extensions of p.

The decomposition of a reduced ideal $\mathfrak{a} = (a, y - b)$ into prime ideals is completely analogous to Corollary 3.17. Let p be a prime dividing a with multiplicity $\nu \geq 0$. If $\nu = 0$, then no prime ideal extending p occurs in the decomposition of $\mathfrak{a}$. Otherwise, p is not inert, and $(p, y - b)$ is an extension of p which occurs with multiplicity ν in the decomposition of $\mathfrak{a}$. If p is splitting, then the conjugate extension $(p, \overline{y} - b) = (p, y - (D - b))$ does not occur in $\mathfrak{a}$, if p is ramified, then in fact $\nu = 1$. So as for hyperelliptic Jacobians, inert prime ideals need not be included into $\mathcal{P}$ after all.

□

5.2 Prime divisor theorem

In the remainder of this chapter, we derive a smoothness result for semireduced divisors of an (imaginary or real quadratic) hyperelliptic curve C over a finite field $\mathbb{F}_q$ and survey

corresponding results in already known cases. The question how many semireduced divisors of given degree are smooth is basically combinatorial: Given a set of components (finite prime divisors) of small size, how many objects (divisors) of given size can be composed from them with respect to certain additional constraints (semireducedness)? To answer the question, a crucial point is clearly the knowledge of the number of components. In the context of hyperelliptic Jacobians, we are interested in the number of finite splitting or ramified prime divisors of given degrees, which by the discussion of Sections 2.1.3 and 2.2.2 are closely related to the number of points on C with coordinates in extension fields of $\mathbb{F}_q$.

For $k \in \mathbb{N}_0$, let $\pi_+(k)$, $\pi_0(k)$ and $\pi_-(k)$ denote the number of irreducible polynomials of degree k which are local parameters of splitting, ramified resp. inert prime divisors of $K(X)$, $\pi(k) = \pi_+(k) + \pi_0(k) + \pi_-(k)$ the number of irreducible polynomials of degree k and $\Pi_+(k) = \sum_{i=1}^{k} \pi_+(k)$. Let $\mathfrak{P}$ be a finite prime divisor of $K(C)$ of degree k which extends a prime divisor $\mathfrak{p}$ of $K(X)$ with local parameter p. By definition of the degree (see Sections 2.1.3 and 2.2.4), $\mathfrak{P}$ corresponds to k points on C with coordinates in the algebraic closure of $\mathbb{F}_q$. If $\mathfrak{P}$ is ramified or splitting, then $\mathfrak{p}$ and thus p are of degree k. Let $x_0, \ldots, x_{k-1}$ with $x_i = x_0^{q^i}$ be the distinct roots of p in $\mathbb{F}_{q^k}$ and let $\mathfrak{P} \cap \mathcal{O} = (p, y - b)$. Then the points corresponding to $\mathfrak{P}$ are the $(x_i, b(x_i))$ with coordinates in $\mathbb{F}_{q^k}$, but no subfield of $\mathbb{F}_{q^k}$. If $\mathfrak{P}$ is inert, then k is even and $\mathfrak{p}$ and thus p are of degree $k/2$. Let $x_0, \ldots, x_{k/2-1}$ with $x_i = x_0^{q^i}$ be the roots of p in $\mathbb{F}_{q^{k/2}}$. Since $\mathfrak{p}$ is inert, by Kummer's theorem 3.10 the polynomial $Y^2 + vY - u \pmod p$ does not have a root in $K[X]$, so $Y^2 + v(x_i)Y - u(x_i)$ does not have a root in $\mathbb{F}_{q^{k/2}}$, but two distinct roots in $\mathbb{F}_{q^k}$. Thus, $\mathfrak{P}$ corresponds to k points with X–coordinates in $\mathbb{F}_{q^{k/2}}$ and Y–coordinates in $\mathbb{F}_{q^k}$, but in no subfield. To simplify the notation, let $\pi_-(k) = 0$ for k half-integral, but not integral. Taking into account that each splitting prime divisor of $K(X)$ extends to two prime divisors of $K(C)$ and each ramified prime divisor of $K(X)$ to only one, the preceding discussion shows that the number of finite points on C with coordinates in an extension field $\mathbb{F}_{q^k}$, but no smaller extension field, is given by

$$k(2\pi_+(k) + \pi_0(k) + \pi_-(k/2)).$$

In addition, there are η infinite points defined over $\mathbb{F}_q$ on the smooth projective model of C, corresponding to the infinite valuations of $K(C)$, with $\eta = 1$ for an imaginary and $\eta = 2$ for a real quadratic curve. (The curve itself has only one infinite point. However, it is not smooth for $g \geq 2$, so that we cannot apply Weil's theorem directly.) Hence the total number of points N_k on the smooth projective model of C with coordinates in $\mathbb{F}_{q^k}$ is given by

$$N_k = \sum_{i|k} i(2\pi_+(i) + \pi_0(i) + \pi_-(i/2)) + \eta. \tag{5.1}$$

On the other hand, Weil's theorem provides a close approximation of this number; by

the results of Section 2.5,

$$q^k - 2gq^{k/2} + 1 \leq N_k \leq q^g + 2gq^{k/2} + 1.$$

We obtain the following result, which is the analogue of the prime number theorem in $\mathbb{Z}$.

Theorem 5.2 (Prime Divisor Theorem) *The number of monic irreducible polynomials of degree at most k which are local parameters of splitting prime divisors of $K(X)$ is given by*

$$\Pi_+(k) \geq \frac{1}{2k}\left(q^k - 2(g+1)(q^{k/2}+1)\right).$$

If $0 < \varepsilon \leq \frac{1}{4}$ and $k \geq \frac{1}{\varepsilon}\log_q(2g+6+\sqrt{2})$, then furthermore

$$\frac{1}{2k}\left(q^k - q^{k(\frac{1}{2}+\varepsilon)}\right) \leq \pi_+(k) \leq \frac{1}{2k}\left(q^k + q^{k(\frac{1}{2}+\varepsilon)}\right).$$

Proof: Weil's theorem and (5.1) imply

$$q^k - 2gq^{k/2} - \sum_{i=1}^{\infty} i\pi_0(i) - \sum_{i|k} i\pi_-\left(\frac{i}{2}\right) - 1 \leq \sum_{i|k} 2i\pi_+(i) \leq q^k + 2gq^{k/2}.$$

As $\sum_{i=1}^{\infty} i\pi_0(i)$ is the summed up degree of all ramified prime divisors and a prime divisor is ramified if and only if it divides the discriminant v^2+4u of C (cf. Proposition 3.10), which by Theorems 3.3 and 3.5 has degree $2g+1$ for an imaginary and $2g+2$ for a real quadratic curve, we have $\sum_{i=1}^{\infty} i\pi_0(i) \leq 2(g+1)$. If k is odd, then $\sum_{i|k} i\pi_-\left(\frac{i}{2}\right)$ is zero, otherwise it is

$$\sum_{i|\frac{k}{2}} 2i\pi_-(i) \leq 2\sum_{i|\frac{k}{2}} i\pi(i) = 2q^{k/2}.$$

This shows that

$$q^k - 2(g+1)(q^{k/2}+1) \leq \sum_{i|k} 2i\pi_+(i) \leq q^k + 2gq^{k/2}. \tag{5.2}$$

Taking into account that

$$\Pi_+(k) = \sum_{i=1}^{k} \pi_+(i) \geq \frac{1}{2k}\sum_{i|k} 2i\pi_+(i),$$

the first assertion is proved.

Letting $f(k) = \sum_{i|k} 2i\pi_+(i)$, Möbius inversion implies

$$2k\pi_+(k) = \sum_{i|k} \mu(k/i)f(i),$$

where the Möbius function μ takes values in $\{0, \pm 1\}$ and $\mu(1) = 1$. Hence for $k \geq \frac{1}{\varepsilon} \log_q(2g + 6 + \sqrt{2})$ we have

$$\begin{aligned}
2k\pi_+(k) &\geq f(k) - \sum_{i=1}^{\lfloor k/2 \rfloor} f(i) \\
&\geq q^k - (2g+2)q^{k/2} - (2g+2) - \sum_{i=1}^{\lfloor k/2 \rfloor} (q^i + 2gq^{i/2}) \text{ by (5.2)} \\
&\geq q^k - (2g+2)q^{k/2} - \frac{q}{q-1}(q^{k/2} - 1) - 2 \\
&\quad -2g\frac{\sqrt{q}}{\sqrt{q}-1}(q^{k/4} - 1) - 2g \\
&\geq q^k - (2g+4)q^{k/2} - 2g(2+\sqrt{2})q^{k/4} \text{ since } q \geq 2 \\
&\geq q^k - (2g+4)q^{k/2} - (2+\sqrt{2})q^{k/2} \\
&\quad \text{since } 2g \leq 2g + 6 + \sqrt{2} \leq q^{k\varepsilon} \leq q^{k/4} \\
&\geq q^k - q^{k(\frac{1}{2}+\varepsilon)}
\end{aligned}$$

The upper bound for $\pi_+(k)$ is derived in a similar way. □

5.3 The subexponential function

The concept of subexponentiality has already been introduced in Section 1.3. Since the smoothness theorems of the following sections involve the subexponential function, it is helpful to have a closer look at this function first.

Recall that the subexponential function with respect to parameters $\alpha \in (0; 1)$, $c > 0$ and the input size $\log N'$ is defined by

$$L_{N'}(\alpha, c) = e^{c(\log N')^\alpha (\log\log N')^{1-\alpha}}.$$

In all situations we will encounter, we have $\alpha = 1/2$, so that we write $L_{N'}(c)$ for $L_{N'}(1/2, c)$. The definition of L implies the following simple relations:

$$\begin{aligned}
L_{N'}(c_1)L_{N'}(c_2) &= L_{N'}(c_1 + c_2), \\
L_{N'}(c)^k &= L_{N'}(kc_1), \\
L_{N'}(c_1) + L_{N'}(c_2) &\in \Theta(L_{N'}(\max(c_1, c_2))).
\end{aligned}$$

Denote by $o(1)$ the set of real valued functions tending to zero as $N' \to \infty$. Then functions in $O^\sim(1)$ or $O(L_{N'}(\alpha, c))$ for $\alpha < 1/2$ are contained in $L_{N'}(o(1))$, and $L_{N'^k}(c) \in L_{N'}(\sqrt{k}c + o(1))$.

During the algorithms of the following chapters it will often be the case that operations of polynomial complexity, e.g. group operations, are repeated a subexponential number of times; say, an operation of complexity in $O(\log^k N')$ is repeated $L(\rho)$ times for some $k \in \mathbb{N}$ and $\rho > 0$. Then, the discussion above implies that all these operations together are performed in a time of $L_{N'}(\rho + o(1))$. It is common practice to add the term "$o(1)$", which allows to abstract from any polynomial time overhead.

We mention a few simple rules for computing with $o(1)$; since $1 + o(1)$ is the set of functions tending to 1 for $N' \to \infty$, we have

$$\begin{aligned} c(1 + o(1)) &= c + o(1) \text{ for any } c \in \mathbb{R}, \\ \alpha(N') + o(1) &\subseteq \alpha(N')(1 + o(1)) \text{ if } \alpha(N') \text{ is bounded away from zero}, \\ \log(1 + o(1)) &= o(1). \end{aligned}$$

The key step of the discrete logarithm algorithms of Chapters 6 and 7 is the test of a random group element for smoothness. The desired smoothness results are of the form: "One out of a subexponential number of group elements is smooth", which implies that the test has to be repeated an expected subexponential number of times until a smooth element is found. Evidently, the ratio of smooth elements increases with the smoothness bound S, so that this part of the algorithm becomes faster for larger S. On the other hand, the construction of $\mathcal{P}_S$ (and a linear algebra step to be explained later) must also be carried out in subexponential time. So $\mathcal{P}_S$ may only be of subexponential size, and S must not be too large. Only if these two contradicting requirements are met, the complete algorithm will be of subexponential complexity.

5.4 Smoothness in arithmetical semigroups

Sometimes, the smoothness assumption in an arithmetical formation reduces to the same assumption in the underlying arithmetical semigroup. This is the case if the elements of the formation are represented by exactly those elements of the semigroup not exceeding a certain size, without taking further restrictions into account. For instance, $\mathbb{F}_p^\times$ is represented by exactly those rational integers whose size is less than $\operatorname{ld} p$ and $\mathbb{F}_{2^m}^\times$ by exactly those binary polynomials whose degree is less than m. Class groups of imaginary quadratic number fields or hyperelliptic curves, however, do not fall into this category since only the (semi-)reduced fraction of the small elements of the semigroup represents the formation.

Smoothness results in arithmetical semigroups are obtained more easily. Since the small prime elements can be composed arbitrarily to form smooth objects, it is basically sufficient to know the number of prime elements of any given size. Then the theory of generating functions and analytic means (see [Ten90, Ten95]) can be employed to derive the smoothness assertions.

For arithmetical semigroups in which the size function is discrete as in the case of polynomials, Knopfmacher introduces Axiom A$^{\#}$ in [Kno79], Chapter 1. It states that the number of objects of size k is in

$$cq^k + O(q^{\alpha k})) \text{ for some } q > 1, c > 0 \text{ and } 0 \leq \alpha < 1.$$

(This allows to deduce an abstract analogue of the Prime Divisor Theorem 5.2 on the number of prime elements of given size when this size tends to infinity, see [IMW91]). In this context, Manstavičius has obtained very general results in [Man92b, Man92a]. The special cases of univariate polynomials over finite fields ([Car87, AD93, BP98, PGF98]) and divisors in algebraic function fields ([Heß99], Chapter 4) have received considerable attention in the literature.

The situation is slightly different if the size is not discrete, for instance in $(\mathbb{N}, \cdot)$, which was the first example considered in the literature. However, an analytic approach yields similar results in this case ([Hen85, Hil86, HT86]).

In this section, we collect some smoothness results available for arithmetical semigroups and of interest in the cryptographic context. Let N' be as above, N the cardinality of the group and N_S the number of S–smooth elements in the group, where S is chosen such that the cardinality of $\mathcal{P}_S$ is subexponential in $\log N'$. Unless stated otherwise, all asymptotic results, involving the term "$o(1)$", are to be understood for $N' \to \infty$.

Consider first the case of finite prime fields $G = \mathbb{F}_p^\times$, in which $N' = N = p - 1$. Clearly, $n_S = |\mathcal{P}_S| \leq 2^S$; more precisely, $n_S \sim \frac{2^S}{S \log 2}$ asymptotically for $S \to \infty$ by the ordinary prime number theorem. Thus, letting $S = \lceil \mathrm{ld}(L_N(\rho)) \rceil$ for some positive constant ρ we obtain $n_S \in L_N(\rho + o(1))$. The number N_S is the number of integers between 1 and N without prime factors larger than 2^S, commonly denoted by $\psi(N, 2^S)$. A theorem of Pomerance's provides the desired smoothness result ([Pom87], Lemma 3.1):

Theorem 5.3 *If $G = \mathbb{F}_p^\times$ and $S = \lceil \mathrm{ld}(L_N(\rho)) \rceil$ for some $\rho > 0$, then*

$$\frac{N}{N_S} = \frac{N}{\psi(N, 2^S)} \in L_N\left(\frac{1}{2\rho} + o(1)\right).$$

To treat more general finite fields $\mathbb{F}_{p^k}^\times$ with $N' = N = p^k - 1$, let $N_p(i, S)$ denote the number of S–smooth monic polynomials over $\mathbb{F}_p$ of degree i, i.e. the number of polynomials each irreducible factor of which is of degree at most S. If $p = 2$, then $N_S = \sum_{i=0}^{k-1} N_2(i, S)$, and the biggest contribution to this sum comes from $N_2(k-1, S)$, so that it is sufficient to consider only this term. In general, $N_S = \psi(p-1, 2^S) \sum_{i=0}^{k-1} N_p(i, S)$, which is larger than, but close to $\psi(p-1, 2^S) N_p(k-1, S)$. Theorem 2.1 of [BP98] estimates $N_p(k-1, S)$ asymptotically:

Theorem 5.4 *Let q be a prime power such that $q^S \geq i \log^2 i$ and $u = \frac{i}{S}$. Then*

$$N_q(i,S) \in \frac{q^i}{u^{u(1+o(1))}}$$

for S, $u \to \infty$.

According to Example 3) of Section 5.1, $\mathcal{P}_S$ contains the monic irreducible polynomials over $\mathbb{F}_p$ of degree up to S and — if $p \geq 3$ — further integers. The number of polynomials in $\mathcal{P}_S$ is within a logarithmic factor of p^S (since the number of all monic polynomials of degree up to S is $p^S - 1$ and the number of irreducible polynomials of degree S is in $\frac{p^S}{S} + O(p^{S/2})$ by the "Prime Polynomial Theorem" 3.25 in [LN97]), so we let $S = \lceil \log_p(L_N(\rho)) \rceil$ for some $\rho > 0$. (Rounding up S may increase the size of $\mathcal{P}_S$ by a factor of p, so that it may become larger than subexponential. Conditions preventing this undesired behaviour are discussed in Section 7.3.3.)

Theorem 5.5 *If $G = \mathbb{F}_{p^k}^\times$ and $S = \lceil \log_p(L_N(\rho)) \rceil$ for some $\rho > 0$, then*

$$\frac{N}{N_S} < \frac{p}{\psi(p-1, 2^S)} \frac{p^{k-1}}{N_p(k-1,S)},$$

and

$$\frac{p^{k-1}}{N_p(k-1,S)} \in L_N\left(\frac{1}{2\rho} + o(1)\right)$$

for $\frac{k}{\log p} \to \infty$.

Proof: The first assertion has already been explained above. Notice that Theorem 5.4 applies, since $p^S \geq L_N(\rho)$ is at least subexponential in $\log(p^k - 1)$, so that it eventually becomes larger than any polynomial in $k - 1$. Thus, $\frac{p^{k-1}}{N_p(k-1,S)} \in u^{u(1+o(1))}$ for $u = \frac{k-1}{S} = \frac{k}{S}\frac{k-1}{k} \in \frac{k}{S}(1 + o(1))$. (Throughout the remainder of this proof, $o(1)$ denotes the set of functions tending to zero for $\frac{k}{\log p} \to \infty$.) Notice that $N \in p^k(1 + o(1))$, so that $\log N \log\log N \in k \log p \log(k \log p)(1 + o(1))$ by the rules for computing with $o(1)$ of Section 5.3. So

$$\begin{aligned} S &\geq \log_p(L_N(\rho)) \\ &\in \log_p\left(e^{\rho(1+o(1))\sqrt{k \log p \log(k \log p)}}\right) \\ &= \rho\sqrt{\frac{k \log(k \log p)}{\log p}}(1 + o(1)). \end{aligned}$$

On the other hand,

$$\begin{aligned} S &< \log_p(L_N(\rho)) + 1 \\ &\in \rho\sqrt{\frac{k\log(k\log p)}{\log p}}\left(1 + o(1) + \frac{1}{\sqrt{\frac{k\log(k\log p)}{\log p}}}\right), \end{aligned}$$

and the condition $\frac{k}{\log p} \to \infty$ implies that

$$S \in \sqrt{\frac{k\log(k\log p)}{\log p}}(1 + o(1)).$$

(Less technically spoken, the condition implies that the rounding of S has no effect asymptotically.) Hence,

$$\begin{aligned} u &\in \frac{k}{S}(1 + o(1)) = \frac{1}{\rho}\sqrt{\frac{k\log(k\log p)}{\log p}}(1 + o(1)), \\ u\log u &\in \frac{1}{\rho}\sqrt{\frac{k\log(k\log p)}{\log p}}(1 + o(1)) \\ &\quad \left(\frac{1}{2}(\log(k\log p) - \log\log(k\log p)) - \log\rho + o(1)\right) \\ &= \frac{1}{2\rho}\sqrt{k\log p\log(k\log p)}(1 + o(1)) \\ &= \frac{1}{2\rho}\sqrt{\log N\log\log N}(1 + o(1)) \quad \text{and} \\ u^{u(1+o(1))} &= e^{u\log u(1+o(1))} = L_N\left(\frac{1}{2\rho} + o(1)\right) \end{aligned}$$

□

5.5 Smoothness in class groups

For (semi-)reduced ideals in quadratic number or function fields the situation is more complicated than in arithmetical semigroups because the components cannot be joined freely any more. In semireduced divisors of hyperelliptic Jacobians, for instance, a ramified prime divisor must not appear more than once, and the splitting prime divisors come in pairs and at most one prime divisor of each pair is allowed to occur. I am aware of only one article dealing with smoothness in class groups. Seysen has examined the density of smooth reduced ideals in imaginary quadratic number fields and obtained the following result.

Theorem 5.6 *Let K be an imaginary quadratic number field of discriminant D. Denote by N the ideal class number of K, let $S = \lceil \log L_N(\rho) \rceil$ for some constant $\rho > 0$, and let N_S be the number of S–smooth reduced ideals. If the generalised Riemann hypothesis holds, then there is a function $\beta(D)$ in $o(1)$ for $D \to \infty$ such that*

$$\frac{N}{N_S} \leq L\left(\frac{1}{2\rho} + \beta(D)\right).$$

Proof: Due to a theorem of Siegel's, $\log|D| \in (2 + o(1)) \log N$ ([Sie36]), so that $L_N(\rho + o(1)) = L_{|D|}\left(\frac{1}{\sqrt{2}}\rho + o(1)\right)$. Then Proposition 4.4 of [Sey87] shows that under the generalised Riemann hypothesis, $\frac{N}{N_S}$ is bounded above by some function in

$$L_{|D|}\left(\frac{\sqrt{2}}{4\rho} + o(1)\right) = L_N\left(\frac{1}{2\rho} + o(1)\right).$$

□

Our aim for the remainder of this section is to determine the number $N(n, S)$ of S–smooth semireduced divisors of $K(C)$ of degree n, where C is an imaginary or real quadratic hyperelliptic curve. Ultimately, we are interested in $N(g, S)$, which counts the biggest portion of all S–smooth reduced divisors. We hereby restrict our attention to divisors composed of only splitting prime divisors; since the number of ramified prime divisors is bounded by $2g+2$, it is negligible asymptotically. To the best of my knowledge, our results are the first ones concerning reduced divisors in function fields.

The following proofs are purely combinatorial and do not rely on the theory of generating functions. The employed techniques might also be used to develop more elementary proofs for the classical cases, e.g. for finite fields.

The first result is inspired by Theorem 2.2 of [BP98]; it is weaker than the subsequent main theorem, but can be used to simplify its proof.

Theorem 5.7 *Let* $\max\left\{8\log_q\left(2g + 6 + \sqrt{2}\right), 2\log_q\left(\left(6 + \frac{10}{3}\sqrt{2}\right)n\right)\right\} + 2 \leq S$ *and* $u = \frac{n}{S}$. *Then*

$$N(n, S) \geq \frac{q^n}{2n^{\lceil u \rceil}}.$$

Proof: Assume first that $S \leq n$. Since the Prime Divisor Theorem 5.2 shows that the number of splitting prime divisors grows exponentially with their degree, we restrict ourselves to counting a set of special semireduced divisors containing only prime divisors of rather large degree, hoping to cover the biggest part of all semireduced divisors. To ensure a large degree for all its prime divisors, a divisor should have as few of them as possible, and for an S–smooth divisor of degree n this means $\lceil u \rceil$ prime divisors. We

distribute the degrees of these prime divisors as evenly as possible. Thus, let $S_0 = \left\lfloor \frac{n}{\lceil u \rceil} \right\rfloor$, $S_1 = S_0 + 1$, $r_1 = n - \lceil u \rceil S_0$ and $r_0 = \lceil u \rceil - r_1$, and let $\tilde{N}(n, S)$ be the number of semireduced divisors containing r_0 distinct splitting prime divisors of degree S_0 and r_1 distinct splitting prime divisors of degree S_1. As $S_0 r_0 + S_1 r_1 = n$, these divisors are of degree n, and as $S_0 \leq \frac{n}{\lceil u \rceil} \leq S$, they are S–smooth unless $S_0 = S$ and $S_1 = S + 1$. In this case, however, $\lceil u \rceil$ divides n and $r_1 = 0$, so that they are S–smooth nevertheless. Thus, $N(n, S) \geq \tilde{N}(n, S)$. To estimate the latter number, notice that there are $\binom{\pi_+(S_i)}{r_i}$ possibilities for choosing r_i splitting prime polynomials of degree S_i and that each prime polynomial leaves the choice of one out of two prime divisors. So the following relations hold:

$$\begin{aligned}
\tilde{N}(n, S) &= 2^{r_0} \binom{\pi_+(S_0)}{r_0} 2^{r_1} \binom{\pi_+(S_1)}{r_1} \\
&\geq 2^{r_0 + r_1} \frac{(\pi_+(S_0) - (r_0 - 1))^{r_0}}{r_0!} \frac{(\pi_+(S_1) - (r_1 - 1))^{r_1}}{r_1!} \\
&\geq \frac{\sqrt{2}^{r_0 + r_1 - 2}}{r_0^{r_0} r_1^{r_1}} 2^{r_0 + r_1} \frac{\left(q^{S_0} - q^{\frac{3}{4} S_0} - 2S_0(r_0 - 1)\right)^{r_0}}{(2S_0)^{r_0}} \\
&\quad \frac{\left(q^{S_1} - q^{\frac{3}{4} S_1} - 2S_1(r_1 - 1)\right)^{r_1}}{(2S_1)^{r_1}} \\
&\quad \text{by } r! \leq \frac{r^r}{\sqrt{2}^{r-1}} \text{ for } r \geq 0 \text{ and by Theorem 5.2 with } \varepsilon = \frac{1}{4} \\
&\geq \frac{\sqrt{2}^{r_0 + r_1 - 2}}{n^{\lceil u \rceil}} \left(q^{S_0} - q^{\frac{3}{4} S_0} - 2n\right)^{r_0} \left(q^{S_1} - q^{\frac{3}{4} S_1} - 2n\right)^{r_1}
\end{aligned}$$

Theorem 5.2 is applicable because $S_1 > S_0 > \frac{n}{\frac{n}{S} + 1} - 1 \geq \frac{1}{2} S - 1$. Notice now that $S_0 \geq 4 \log_q(2g + 6 + \sqrt{2}) \geq 4 \log_q(8 + \sqrt{2})$ implies $q^{\frac{1}{4} S_0} \geq 8 + \sqrt{2}$ and $q^{\frac{3}{4} S_0} \leq \frac{q^{S_0}}{8 + \sqrt{2}}$. Similarly, $q^{\frac{3}{4} S_1} \leq \frac{q^{S_1}}{8 + \sqrt{2}}$. Furthermore, letting $c = 1 - \frac{1}{8 + \sqrt{2}} - \frac{1}{\sqrt{2}}$, we deduce that $2n \leq c q^{S_0} \leq c q^{S_1}$ as soon as S satisfies the second lower bound. Hence,

$$\tilde{N}(n, S) \geq \frac{\sqrt{2}^{r_0 + r_1}}{2n^{\lceil u \rceil}} \frac{q^{S_0 r_0}}{\sqrt{2}^{r_0}} \frac{q^{S_1 r_1}}{\sqrt{2}^{r_1}} = \frac{q^n}{2n^{\lceil u \rceil}}.$$

Finally, if $S > n$, then

$$N(n, S) = N(n, n) \geq \frac{q^n}{2n} = \frac{q^n}{2n^{\lceil u \rceil}}.$$

□

Theorem 5.7 is not yet sufficient to prove the subexponentiality result needed for the discrete logarithm algorithms. In fact, we need a bound for $N(n, S)$ of about $\frac{q^n}{u^u}$, so that we have to improve the bound of the theorem above by a factor of about S^u.

When S is of the order of $\log n$, the desired result can be derived easily from Theorem 5.7.

Corollary 5.8 *Suppose that under the conditions of Theorem 5.7, we have furthermore $S \le k \log n$ for some constant $k > 0$. Then*

$$N(n,S) \ge \frac{q^n}{u^{u\left(\left(1+\frac{1}{u}\right)\left(1+\frac{\log(k\log n)}{\log u}\right)+\frac{\log 2}{u\log u}\right)}} \in \frac{q^n}{u^{u(1+o(1))}} \text{ for } u \to \infty.$$

Proof: In this special case, the denominator of the formula in Theorem 5.7 satisfies

$$2n^{\lceil u\rceil} = 2S^{\lceil u\rceil}u^{\lceil u\rceil} \le 2(k\log n)^{u+1}u^{u+1} = u^{u\left(\left(1+\frac{1}{u}\right)\left(1+\frac{\log(k\log n)}{\log u}\right)+\frac{\log 2}{u\log u}\right)}.$$

The asymptotic result follows because $n \to \infty$ as $u \to \infty$ and $\frac{\log\log n}{\log u} \le \frac{\log\log n}{\log n - \log(k\log n)} \to 0$ $(n \to \infty)$ □

For larger S, we need to follow a different approach, since n and u differ considerably. Still, we have to assume that S is not too large compared to n; precisely, we require $S \le n^{1-\varepsilon}$ for some $\varepsilon \in (0;1)$. As hyperelliptic function fields are the function field analogue of quadratic number fields, it can be expected that results and techniques concerning smooth ideals in quadratic number fields carry over to our problem. Indeed, this is the case. The following theorem and its proof are inspired by Theorem 5.2 in [Sey87]. We can use Theorem 5.7 above to simplify the proof.

Theorem 5.9 *If there is a constant $\varepsilon \in (0;1)$ such that S, n and $u = \frac{n}{S}$ satisfy*

$$\max\left\{16\log_q(2g+6+\sqrt{2})+4, 4\log_q\left(\left(6+\frac{10}{3}\sqrt{2}\right)n\right)+4, \log n\right\}$$
$$\le S \le n^{1-\varepsilon},\ n \ge 29 \text{ and } \frac{4}{\varepsilon}u\log u \ge 1,$$

then

$$N(n,S) \ge \frac{q^n}{u^{u\left(1+\frac{\log\log u+2+\log\frac{4}{\varepsilon}}{\log u}+\frac{3}{\varepsilon u}\right)}} \in \frac{q^n}{u^{u(1+o(1))}} \text{ for } u \to \infty.$$

Proof: For Theorem 5.7, we counted all divisors with $\lceil u\rceil$ prime divisors all of which had degree S_0 or S_0+1. To verify that the number of smooth divisors is in fact larger, we must allow more flexibility in the size of the components. Thus, we consider divisors containing $\lfloor u\rfloor$ prime divisors whose degrees vary within a certain factor of S. To reach the total degree n, we pad by prime divisors of smaller degree.

Precisely, let $S-1 \ge w := \left\lfloor\left(1-\frac{1}{\log n}\right)S\right\rfloor \ge \left(1-\frac{1}{\log n}-\frac{1}{S}\right)S \ge \frac{S}{2}$ for $S \ge 5$ and $n \ge 29$. Let $\mathcal{S}$ be a set of prime divisors containing exactly one divisor above each

splitting prime polynomial p with $w+1 \leq \deg p \leq S$. We consider divisors of the form $D = D_1 + D_2$, where D_1 contains exactly $\lfloor u \rfloor$ (not necessarily distinct) prime divisors from $\mathcal{S}$ and D_2 is semireduced and w–smooth of degree $n - \deg D_1$. From the construction of $\mathcal{S}$ it follows that D_1 is semireduced and S–smooth and that D_1 and D_2 share no prime divisor. Furthermore, $\deg D = n$, so that $N(n,S)$ is bounded below by the number of such divisors D. Let $\mathcal{I}$ be the set of possible divisors D_1. Then the above discussion implies

$$N(n,S) \geq \sum_{D_1 \in \mathcal{I}} N(n - \deg D_1, w).$$

From $w \geq \frac{S}{2}$ and the restrictions imposed on S we see that Theorem 5.7 applies to the situation, so that

$$N(n - \deg D_1, w) \geq \frac{q^{n-\deg D_1}}{2(n-\deg D_1)^{\lceil \frac{n-\deg D_1}{w} \rceil}}.$$

The logarithm of the denominator is bounded above by

$$\begin{aligned}
\left(\frac{n-\deg D_1}{w} + 2\right)\log n &\leq \left(\frac{n}{w} - (u-1) + 2\right)\log n \\
&\leq \left(\left(\frac{1}{1 - \frac{1}{\log n} - \frac{1}{S}} - 1\right)u + 3\right)\log n \\
&\leq \frac{2}{\log n - 2}u + 3\log n \text{ since } S \geq \log n \\
&\leq 2u + \frac{3}{\varepsilon}\log u \text{ since } \log n \geq 3 \text{ and } n^{\varepsilon} \leq u
\end{aligned}$$

Hence,

$$N(n,S) \geq \frac{q^n}{u^{u\left(\frac{2}{\log u} + \frac{3}{\varepsilon u}\right)}} \sum_{D_1 \in \mathcal{I}} q^{-\deg D_1}.$$

The last sum can be computed using the Prime Divisor Theorem. Let $\mathcal{S} = \{\mathfrak{P}_1, \ldots, \mathfrak{P}_l\}$.

$$\begin{aligned}
\sum_{D_1 \in \mathcal{J}} q^{-\deg D_1} &= \sum_{a_i \geq 0,\, a_1 + \cdots + a_l = \lfloor u \rfloor} q^{-a_1 \deg \mathfrak{P}_1 - \cdots - a_k \deg \mathfrak{P}_l} \\
&\geq \frac{\left(\sum_{i=1}^{l} q^{-\deg \mathfrak{P}_i}\right)^{\lfloor u \rfloor}}{\lfloor u \rfloor!} \\
&\quad \text{since by multiplying out } \left(\sum_{i=1}^{k} q^{-\deg \mathfrak{P}_i}\right)^{\lfloor u \rfloor} \text{ each term of} \\
&\quad \text{the previous sum is obtained at most } \lfloor u \rfloor! \text{ times} \\
&\geq u^{-\lfloor u \rfloor} \left(\sum_{j=w+1}^{S} \pi_+(j) q^{-j}\right)^{\lfloor u \rfloor} \\
&\geq u^{-\lfloor u \rfloor} \left(\sum_{j=w+1}^{S} \frac{1}{2j} \left(1 - \frac{1}{q^{j\left(\frac{1}{2} - \frac{1}{8}\right)}}\right)\right)^{\lfloor u \rfloor} \\
&\quad \text{by Theorem 5.2 with } \varepsilon = \frac{1}{8} \\
&\geq u^{-\lfloor u \rfloor} \left(\frac{1}{4} \sum_{j=w+1}^{S} \frac{1}{j}\right)^{\lfloor u \rfloor} \quad \text{since } q^{\frac{3}{8}(w+1)} \geq 2^{\frac{3}{2}} > 2 \\
&\geq u^{-\lfloor u \rfloor} \left(\frac{S-w}{4S}\right)^{\lfloor u \rfloor} \\
&\geq (4u \log n)^{-\lfloor u \rfloor} \\
&\geq \left(\frac{4}{\varepsilon} u \log u\right)^{-\lfloor u \rfloor} \geq \left(u^{1 + \frac{\log\log u + \log \frac{4}{\varepsilon}}{\log u}}\right)^{-u}
\end{aligned}$$

This achieves the proof of the theorem. □

The smoothness result needed for the algorithms of Chapters 6 and 7 is that one out of a subexponential number of reduced divisors is smooth, so we are interested in the case $n = g$. The subexponential function is written with respect to $N' = q^g$. Since $\mathcal{P}_S$ is of size $O(q^S)$, which must be subexponential, we let

$$S = \left\lceil \log_q L_{q^g}(\rho) \right\rceil = \left\lceil \rho \sqrt{\frac{g \log(g \log q)}{\log q}} \right\rceil$$

with a constant $\rho > 0$. (In fact, rounding up the value for S may make $\mathcal{P}_S$ of exponential size. Conditions preventing this situation are discussed in Section 6.1.3.)

Our aim is to use Corollary 5.8 and Theorem 5.9 to obtain asymptotic results for $g \to \infty$. Notice that either the conditions of the corollary or of the theorem are fulfilled for any $\varepsilon \in (0; \frac{1}{2})$ and g large enough. Since $u \to \infty$ as $g \to \infty$, we have

$$N(g, S) \geq \frac{q^g}{u^{u(1+\alpha(g))}}$$

with $\alpha(g) \to 0$ for $g \to \infty$. In our special situation,

$$u = \frac{g}{S} \leq \frac{1}{\rho} \sqrt{\frac{g \log q}{\log(g \log q)}} \leq \frac{1}{\rho} \sqrt{g \log q}$$

and hence

$$\log u \leq \frac{1}{2} \log(g \log q) - \log \rho,$$

and the logarithm of the denominator of $N(g, S)$ is given by

$$\begin{aligned} (1 + \alpha(g)) u \log u &\leq \frac{1}{2\rho}(1 + \alpha(g)) \left(1 - \frac{2 \log \rho}{\log(g \log q)}\right) \sqrt{(g \log q) \log(g \log q)} \\ &\in \left(\frac{1}{2\rho} + o(1)\right) \sqrt{(g \log q) \log(g \log q)}. \end{aligned}$$

This proves the following result:

Theorem 5.10 *Let $S = \lceil \log_q L(\rho) \rceil$ for a constant $\rho > 0$. Then there is a function $\beta(g)$ in $o(1)$ for $g \to \infty$ such that*

$$N(g, S) \geq L_{q^g}\left(-\frac{1}{2\rho} - \beta(g)\right) q^g.$$

Chapter 6

Subexponential algorithms for groups with unknown structure

By definition, a discrete logarithm problem is always solved in a finite cyclic group with known generator. Two basic situations can be distinguished. In this chapter, we develop an algorithm for the case that the cardinality of the group is unknown, in the next chapter we describe a faster algorithm exploiting the knowledge of the group order. In the cryptographic context, the first situation is unlikely to occur, since the group order is needed to realise signature schemes and to check resistance against the Pohlig–Hellman attack. Nevertheless, all published algorithms for class groups with a (sometimes only conjectured) subexponential running time solve the discrete logarithm problem in the first setting. This is justified by observing that the algorithms solve a more fundamental problem: A byproduct of the computations is the order of the generating element, i.e. the cardinality of the group.

More generally yet, the discrete logarithm problem is usually embedded into a larger group; for instance, Jacobians of hyperelliptic curves are not cyclic in general, so that the base element of the logarithm cannot generate the full group and the discrete logarithm problem has to be solved in a cyclic subgroup. The smoothness results of the previous chapter, however, are only proved for the full group. In practice, this is no serious restriction. As there is no apparent relation between smoothness and membership in a subgroup, one usually assumes that the distribution of smooth elements is the same in the group and its subgroups and arrives at so-called "heuristic algorithms", the running times of which are only conjectured to be subexponential. It is an interesting open question whether smoothness results can be transferred from the full group to (certain) subgroups; compare, for instance, the result of class field theory stating that the density of prime ideals is the same in each class of certain class groups. The discussion in [Kno75], Chapter 9, might be a good starting point for further research.

Meanwhile, being interested only in algorithms the running times of which we can prove, we instead have to work with the complete group. Then in fact our discrete logarithm algorithm attempts to solve the fundamental problem of computing the group structure, i.e. the decomposition of the group into a product of cyclic groups $G = \langle \alpha_1 \rangle \cdots \langle \alpha_r \rangle$ with α_i of known order e_i. If the orders e_i meet the further restrictions $e_i \geq 2$ and $e_i | e_{i+1}$, then by the Elementary Divisor Theorem they are unique, and the algorithm has determined important invariants of the group. However, the algorithm may fail in computing the group structure; indeed, if no further information is available, it determines a series of possible group structures until the candidate structure is compatible with the discrete logarithm problem. After describing and analysing the algorithm, we briefly discuss how the correct group structure can be derived from additional information in Section 6.5.

The first discrete logarithm algorithm for hyperelliptic Jacobians with a conjectured subexponential running time, which motivated this research, is due to Adleman, DeMarrais and Huang ([ADH94]). It follows a slightly different approach than the one described here. In Section 6.4, we show how to formulate the algorithm in our general framework and discuss why it is unlikely that a subexponential running time can be proved with our means.

Since the multiplicative groups of finite fields are cyclic and of known order, the only examples presented in this chapter are class groups of imaginary quadratic number or function fields.

To fix the notation for the remainder of this chapter, let $G = \mathbb{M}/\sim$ be an additively written arithmetical formation of unknown cardinality N and g_1 and g_2 two elements of G such that $g_2 = l g_1$ for $l \in \mathbb{N}_0$. The discrete logarithm of g_2 to the base g_1 is defined modulo the order of g_1, so $\log_{g_1}(g_2) = l \bmod \operatorname{ord}(g_1)$. Since this order is unknown, we seek some integer l' such that $g_2 = l' g_1$. As explained in Section 5.1, we measure the algorithmic complexity with respect to the parameter $N' = N$ for imaginary quadratic number fields and $N' = q^g$ for Jacobians of hyperelliptic curves of genus g over the field $\mathbb{F}_q$.

6.1 Parameters

6.1.1 Generating property

Having to work with the full group and not only with the subgroup generated by g_1, we must impose a further restriction on the *factor base*, i.e. the set $\mathcal{P}_S$. (The term "factor base" stems from the multiplicative setting in a finite field. Since our main focus are class groups in function fields, we prefer additive notation, but keep the naming conventions.) Recall from Section 5.1 that $\mathcal{P}_S$ consists of the n_S prime elements of $\mathbb{M}$ of size at most S, that $\deg p \geq 1$ for $p \in \mathcal{P}_S$ and that the size of each element of G is in $O^{\sim}(1)$. Let

$$d = \max\{\deg g : g \in G\},$$

so that the number of primes in a decomposition of any element of G, counting multiplicities, is bounded by $d \in O^{\sim}(1)$.

All group elements considered during the algorithm are derived as linear combinations of g_1, g_2 and the elements of $\mathcal{P}_S$. To be able to determine the group structure of G, we thus require that $\mathcal{P}_S$ generates G.

Examples.

1) Ideal class groups of imaginary quadratic number fields

   Corollary 6.2 of [Sch82] shows that under the generalised Riemann hypothesis, the ideal class group of an imaginary quadratic number field of discriminant D is generated by the splitting prime ideals of norm in $O(\log^2 |D|)$, i.e. of size in $O(\log \log |D|)$, with effectively computable constants. Since $\log |D| \in (2 + o(1)) \log N$ ([Sie36]) and $n_S \in O(2^S)$, this implies that a factor base of polynomial size would in fact suffice to obtain the generating property. Allowing a factor base of subexponential size, we may thus assume that it generates the class group. The proof of the result in [Sch82] is based on the observation that under the generalised Riemann hypothesis, any non-principal character of the group of ideals admits a non-trivial value for a prime ideal of small norm (cf. [LMO79], Corollary 1.3; [Bac90], Theorem 4). We renounce at providing details; an analogous reasoning is developed for function fields below.

2) Jacobians and ideal class groups of curves over finite fields

   In [MST99] it is shown that the ideal class group of a real quadratic hyperelliptic curve over a finite field of odd characteristic is generated by prime divisors of small degree. The proof is based on the generalised Riemann hypothesis for function fields, i.e. on Weil's theorem of Section 2.5, and carries over to Jacobians and ideal class groups of arbitrary curves over any finite field.

   Recall the theory of group characters. A *character* of a finite abelian group is a homomorphism from the group into the multiplicative group of the complex roots of unity; the trivial character, assigning 1 to each element, is called *principal*. The group of characters of G is denoted by $\hat{G}$. The subgroups of G are in a one-to-one correspondence with the subgroups of $\hat{G}$ via the map which to a subgroup U of G associates the subgroup of characters which become principal when restricted to U ([Hal76], Theorem 13.2.3). We cite from [MST99], Corollary 1, slightly reformulated:

   Theorem 6.1 *Let C be a curve of genus g over $K = \mathbb{F}_q$ and χ a character of finite order of $\mathrm{Div}(K(C)/K)$ which is not principal when restricted to $\mathrm{Div}^0(K(C)/K)$. Then there is a prime divisor $\mathfrak{P}$ of $K(C)$ of degree at most $\left\lceil \frac{2\log(4g-2)}{\log q} \right\rceil$ such that $\chi(\mathfrak{P}) \neq 1$.*

Theorem 6.2 *In the situation of Theorem 6.1, let D_1 be a divisor of degree 1, which exists since K is finite. Then $J(K(C)/K)$ is generated by the degree zero divisors of the form $\mathfrak{P} - (\deg \mathfrak{P})D_1$, where $\mathfrak{P}$ varies over the prime divisors of degree at most $\lceil 2\log_q(4g-2)\rceil$. Letting $\mathcal{O}$ denote the integral closure of $K(X)$ in $K(C)$ and assuming as always that there is a divisor of degree 1 composed of only infinite prime divisors, the ideal class group $\mathfrak{H}(\mathcal{O})$ is generated by the finite prime divisors of degree at most $\lceil 2\log_q(4g-2)\rceil$.*

Proof: Let U be the subgroup of $J(K(C)/K)$ generated by the degree zero divisors of the theorem. To show that $U = J(K(C)/K)$ we have to verify that any non-principal character χ of $J(K(C)/K)$ remains non-principal when restricted to U. The map

$$\begin{aligned} \pi \quad : \quad & \mathrm{Div}(K(C)/K) \to J(K(C)/K), \\ & D \mapsto D - (\deg D)D_1 + \mathrm{Prin}(K(C)/K), \end{aligned}$$

is an epimorphism, so that χ induces a character χ' of $\mathrm{Div}(K(C)/K)$ of finite order via

$$\chi'(D) = \chi(\pi(D)).$$

In fact, the restriction of π to $\mathrm{Div}^0(K(C)/K)$ is already surjective, and the non-principality of χ implies that of the restriction of χ' to $\mathrm{Div}^0(K(C)/K)$. Thus, by Theorem 6.1, there is a prime divisor $\mathfrak{P}$ of degree at most $\lceil 2\log_q(4g-2)\rceil$ such that $1 \neq \chi'(\mathfrak{P}) = \chi(\mathfrak{P} - (\deg \mathfrak{P})D_1 + \mathrm{Prin}(K(C)/K)$, and χ is not principal on U.

If furthermore D_1 consists of only infinite prime divisors, i.e., $D_1 \in \mathrm{Div}_\infty(\mathcal{O})$ in the notation of Section 2.4.2, then the map

$$\begin{aligned} \psi \quad : \quad & J(K(C)/K) \to \mathfrak{H}(\mathcal{O}), \\ & \sum_{\mathfrak{P}\nmid\infty} \nu_{\mathfrak{P}}\mathfrak{P} + \sum_{\mathfrak{P}\mid\infty} \nu_{\mathfrak{P}}\mathfrak{P} + \mathrm{Prin}(K(C)/K) \mapsto \sum_{\mathfrak{P}\nmid\infty} \nu_{\mathfrak{P}}\mathfrak{P} + \mathrm{Prin}(\mathcal{O}), \end{aligned}$$

is an epimorphism, and $\mathfrak{H}(\mathcal{O})$ is generated by

$$\begin{aligned} & \big\{\psi\left(\mathfrak{P} - (\deg \mathfrak{P})D_1 + \mathrm{Prin}(K(C)/K)\right) : \mathfrak{P} \text{ prime divisor}, \\ & \quad \deg \mathfrak{P} \leq \lceil 2\log_q(4g-2)\rceil \big\} \\ & = \quad \big\{\mathfrak{P} + \mathrm{Prin}(\mathcal{O}) : \mathfrak{P} \text{ finite prime divisor}, \deg \mathfrak{P} \leq \lceil 2\log_q(4g-2)\rceil\big\} \end{aligned}$$

□

Notice that in the hyperelliptic case, any inert prime divisor is principal (cf. Kummer's Theorem 3.10), so that the ideal class group is in fact generated by the finite splitting and ramified prime divisors matching the degree bound.

□

6.1.2 Maximal exponent

A second important parameter concerning the generation of G is the maximal "exponent" E such that each element of the group can be expressed as a linear combination of primes in $\mathcal{P}_S$ with coefficients in the range $\{0, \ldots, E-1\}$. If N were known, then one could clearly choose $E = N$. Unfortunately, to date there is no polynomial time algorithm computing the size of a hyperelliptic Jacobian; notice that Pila's deterministic algorithm [Pil90], often referred to as "polynomial", is so only for fixed genus. The same is true for the algorithm described by Huang and Ierardi in [HI98]. Instead, one can work with an upper bound $\overline{N}$ on N, which, to keep the running time bounds during the computation of a linear combination, should satisfy $\log \overline{N} \in O^{\sim}(1)$.

For imaginary quadratic number fields of discriminant D, an approximation $\overline{N}$ of the class number h such that $h \leq \overline{N} < 2h$ can be computed in time polynomial in $\log |D|$, assuming the validity of the generalised Riemann hypothesis. The algorithm is based on the class number formula

$$h = \frac{\sqrt{|D|}}{\pi} \prod_{p \text{ prime}} \frac{1}{1 - \frac{\chi(p)}{p}}$$

for $D < -4$ with the quadratic character $\chi(p) = \left(\frac{D}{p}\right)$. It can be shown that under the generalised Riemann hypothesis the desired approximation is obtained by truncating the infinite product after a polynomial number of factors ([McC89], p. 468; [Sch82], Theorem 6.3).

A similar approach should be possible for hyperelliptic Jacobians (cf. Section 6.5). However, the simpler bound $N \leq \overline{N} = (2g+1)q^g$ of Proposition 5.1 is available without further computation.

To simplify the analysis of the algorithm, we let

$$E = 5\overline{N} + d,$$

where $\overline{N} \geq \max\{N, N'\}$ and $\log \overline{N} \in O^{\sim}(1)$.

6.1.3 Two-parametric problems

In a hyperelliptic Jacobian of genus g over a finite field $\mathbb{F}_q$ the input size $\log N'$ with $N' = q^g$ depends on the two parameters g and q. Ideally, a discrete logarithm algorithm would have a subexponential running time for $N' \to \infty$, regardless of how the growth of N' is distributed on q and g; in particular, the case $g = 1$ and $q \to \infty$ would provide a subexponential algorithm for elliptic curves. Unfortunately, the subexponentiality can only be proved under further restrictions. The Prime Divisor Theorem 5.2 implies that the size n_S of $\mathcal{P}_S$ is in $\Omega\left(\frac{1}{S} q^S\right) \cap O\left(q^S\right)$; since $S \in O^{\sim}(1)$, q^S is the order of magnitude of n_S up to negligible factors. Clearly, a necessary condition for subexponentiality is that

n_S and thus q^S are subexponential in $g \log q$. The problem is that we have to round up the value assigned to S, so that for $S = \lceil \log_q L_N(\rho) \rceil$ the value q^S can be (almost) as big as $q^{\log_q L(\rho)+1} = qL(\rho)$. Hence we must assume that q is subexponential in $g \log q$, and since q is exponential in $\log q$, this can only happen for rather large g. (The necessity of this condition can also be derived from the observation that $\mathcal{P}_S$ contains at least the prime divisors of degree 1.) The following theorem shows that a sufficient condition for the subexponentiality of n_S is the existence of a constant $\vartheta > 0$ such that all problem instances under consideration satisfy $g \geq \vartheta \log q$.

Theorem 6.3 *If $g \geq \vartheta \log q$ and $S = \lceil \log_q L_{q^g}(\rho) \rceil$, then $q \leq L_{q^g}\left(\frac{1}{\sqrt{\vartheta}}\right)$, and*

$$n_S \leq 2L_{q^g}\left(\rho + \frac{1}{\sqrt{\vartheta}}\right) \in L_{q^g}\left(\rho + \frac{1}{\sqrt{\vartheta}} + o(1)\right).$$

Proof: We compute

$$q = e^{\log q} = e^{\frac{1}{\sqrt{\vartheta}}\sqrt{\vartheta(\log q)^2}} \leq e^{\frac{1}{\sqrt{\vartheta}}\sqrt{g \log q}} \leq L_{q^g}\left(\frac{1}{\sqrt{\vartheta}}\right).$$

Since each prime divisor of size at most S corresponds to a monic polynomial of degree at most S and conversely each such polynomial corresponds to at most two prime divisors, we have

$$n_S \leq 2q^S \leq 2qL_{q^g}(\rho) \leq 2L_{q^g}\left(\frac{1}{\sqrt{\vartheta}}\right) L_{q^g}(\rho) = 2L_{q^g}\left(\rho + \frac{1}{\sqrt{\vartheta}}\right).$$

□

A similar problem occurs for general finite fields $\mathbb{F}_{p^k}$, and it can be solved in a similar manner, cf. Section 7.3.3.

6.2 Algorithm

The algorithm proceeds in two stages. First, a possible group structure is determined, second, individual discrete logarithm problems are tried to be solved. If the second step is not successful, then there has been a mistake in the first one, and the algorithm must be started all over again. However, we shall show in Section 6.3.3 that the probability of failure is negligibly low. Notice that the algorithm should not be implemented as described, since the description is optimised to simplify the proof of the running time, not implementational efficiency. See Section 6.6 for a few thoughts on practicability.

6.2.1 Finding the group structure

Assume that the factor base $\mathcal{P}_S = \{\mathfrak{P}_1, \ldots, \mathfrak{P}_n\}$ with $n = n_S$ generates the group and let the exponent E be as in Section 6.1.2. Then the group homomorphism

$$\mathbb{Z}^n \to G,\ (e_1, \ldots, e_n) \mapsto e_1\mathfrak{P}_1 + \ldots e_n\mathfrak{P}_n,$$

is surjective, and if Γ is its kernel, then

$$\mathbb{Z}^n/\Gamma \simeq G.$$

As $|G| = N$ is finite, Γ is a full lattice of determinant N, the elements of which are called *relations*. During the first stage of the algorithm, we try to determine a basis for Γ. Starting with the empty matrix A, by a randomised procedure described below we alternately create a new relation and add it as a new column to A. In the case where N is known it is then easy to determine whether the columns of A generate Γ. Since we do not wish to make this assumption, we have to generate a rather large number of relations, pretend that they generate Γ and try to solve the discrete logarithm problem. If we do not succeed, we did not determine the correct group structure and start again with the first phase. (Beware that even if we solve the discrete logarithm problem, our computed group structure need not be correct.) In the case that the columns of A do generate Γ, the structure of G is closely related to a special transform of A; recall the following definitions (see [Coh93], Section 2.4):

Definition and proposition 6.4 *Let $A = (a_{ij})$ be an integral $n \times m$-matrix of rank n.*

1) *A is in* column echelon form *if its first $m - n$ columns are zero and its last n columns form an upper triangular matrix.*

2) *A is in* Hermite normal form *if it is in column echelon form and $a_{i,i+m-n} > 0$ for $i = 1, \ldots, n$ and $0 \le a_{i,j+m-n} < a_{i,i+m-n}$ for $i = 1, \ldots, n$, $j = i+1, \ldots, n$. There is a unique matrix $B \in \mathbb{Z}^{n\times m}$ in Hermite normal form such that $B = AT$ for a unimodular matrix $T \in \mathbb{Z}^{m\times m}$. Hence the columns of B and A span the same lattice, and the* essential part *of B, i.e. its non-zero columns, forms a canonical basis for this lattice.*

3) *Suppose that $B \in \mathbb{Z}^{n\times n}$ is the essential part of the Hermite normal form of A. Then there is a unimodular matrix $S \in \mathbb{Z}^{n\times n}$ such that $\Delta = SB$ is a diagonal matrix with diagonal entries $d_1|\cdots|d_n$; Δ is called the* Smith normal form *or* elementary divisor form *of A, and $d_1, \ldots, d_n$ are the uniquely determined* elementary divisors *of A.*

To obtain the relations, we follow ideas first presented by McCurley in [McC89]. Basically we compute random linear combinations of prime elements and try to express their canonical representatives as another linear combination of prime elements.

The following algorithm succeeds with a high probability in finding the group structure:

Algorithm 6.5

1) Let A be the empty matrix. Choose a smoothness bound S and construct the factor base $\mathcal{P}_S = \{\mathfrak{P}_1, \ldots, \mathfrak{P}_n\}$.

2) Find $20n$ relations as follows: Repeatedly select a random vector $\mathbf{e} = (e_1, \ldots, e_n) \in \{0, \ldots E-1\}^n$ and compute the canonic representative of $e_1\mathfrak{P}_1 + \cdots + e_n\mathfrak{P}_n$ until it "factors" over $\mathcal{P}_S$ as $r_1\mathfrak{P}_1 + \cdots + r_n\mathfrak{P}_n$. Then $(r_1 - e_1, \ldots r_n - e_n)^T \in \Gamma$; add this column to A.

3) Compute the rank of A. If A does not have full rank, then go to Step 2).

4) Otherwise construct $40n \operatorname{ld} E$ new relations by the procedure described under Step 2).

5) Compute the elementary divisor form of A.

6.2.2 Computing discrete logarithms

To relate g_1 and g_2 with the primes in $\mathcal{P}_S$, we have to find S–smooth elements $\tilde{g}_1 \sim g_1$ and $\tilde{g}_2 \sim g_2$ in $\mathbb{M}$. To do so, we again choose random vectors $\mathbf{e} \in \{0, \ldots E-1\}^n$ until $g_1 + \sum_{i=1}^n e_i\mathfrak{P}_i$ decomposes as $\sum_{i=1}^n r_i\mathfrak{P}_i$, and let $\tilde{g}_1 = \sum_{i=1}^n (r_i - e_i)\mathfrak{P}_i$; an analogous procedure yields $\tilde{g}_2$.

Assume that the algorithm of Section 6.2.1 has yielded a basis for Γ and that $B \in \mathbb{Z}^{n\times n}$ is the essential part of its Hermite normal form and $\Delta = SB$ its elementary divisor form with diagonal entries $d_1 | \cdots | d_n$. Denote by $\mathbf{c}^{(j)}$ the coefficient vector of $\tilde{g}_j$ with respect to $\mathfrak{P}_1, \ldots, \mathfrak{P}_n$. Then $g_2 - lg_1 = 0$ in G implies $\tilde{g}_2 - l\tilde{g}_1 \sim 0$ in $\mathbb{M}$. Since the columns of B generate Γ, we know that $\mathbf{c}^{(2)} - l\mathbf{c}^{(1)} \in \operatorname{Im} B$, or equivalently $S\mathbf{c}^{(2)} - lS\mathbf{c}^{(1)} \in \operatorname{Im} \Delta$. Letting $S\mathbf{c}^{(j)} = \mathbf{a}^{(j)} = \left(a_1^{(j)}, \ldots, a_n^{(j)}\right)^T$, this is equivalent to

$$a_i^{(2)} \equiv l a_i^{(1)} \pmod{d_i} \quad \text{for } i = 1, \ldots, n,$$

from which l can be determined modulo d_n, which is the exponent of the group.

If the algorithm of Section 6.2.1 did not succeed in finding a basis of Γ, but only of a sublattice, then the above congruences may or may not have a solution. In the first case the solution is the correct discrete logarithm, otherwise we declare failure and start the whole group structure determination again.

6.3 Analysis

6.3.1 Finding a relation

The crucial part of Algorithm 6.5 is the creation of relations in Steps 2) and 4). If the randomly generated linear combinations of primes in $\mathcal{P}_S$ were uniformly distributed over

the elements of G, then the probability of finding a relation would be $\frac{N_S}{N}$. In this section we show that the derivation from the uniform distribution is sufficiently small, using techniques inspired by those in [Buc90] and [Ste96]. In a first step we determine how many exponent vectors $\mathbf{e}$ yield a fixed relation $\mathbf{c}$.

Lemma 6.6 *Let $\mathbf{c} \in \Gamma$. Then the number of vectors $\mathbf{e} \in \{0, \ldots, E-1\}^n$ which yield the relation $\mathbf{c}$ equals the number of S-smooth elements $\sum_{i=1}^{r} r_i \mathfrak{P}_i$ such that $\mathbf{r} - \mathbf{c} \in \{0, \ldots E-1\}^n$.*

Proof: It follows from the description of the relation generating process in Step 2) of Algorithm 6.5 that $\mathbf{e}$ creates the relation $\mathbf{c}$ if and only if $\mathbf{e} + \mathbf{c}$ is the coefficient vector of an S–smooth element. □

Since the coefficients in the decomposition of a group element are contained in the set $\{0, \ldots, d\}$, the lemma allows to make a more precise assertion for relations contained in the cubes $W^- = \{d+1-E, \ldots, 0\}^n$ and $W^+ = \{1-E, \ldots, d\}^n$.

Corollary 6.7

1) *Let $\mathbf{c} \in \Gamma \cap W^-$. Then there are exactly N_S choices for $\mathbf{e} \in \{0, \ldots, E-1\}^n$ which yield the relation $\mathbf{c}$.*

2) *Let $\mathbf{c} \in \Gamma \cap W^+$. Then there are at most N_S exponent vectors $\mathbf{e} \in \{0, \ldots, E-1\}^n$ which yield the relation $\mathbf{c}$.*

3) *Let $\mathbf{c} \in \Gamma \backslash W^+$. Then there is no exponent vector $\mathbf{e} \in \{0, \ldots, E-1\}^n$ which yields the relation $\mathbf{c}$.*

The corollary implies that a uniform choice of $\mathbf{e} \in \{0, \ldots, E-1\}^n$ yields a relation with probability between

$$\frac{|\Gamma \cap W^-| N_S}{E^n} \quad \text{and} \quad \frac{|\Gamma \cap W^+| N_S}{E^n},$$

and we have to estimate the cardinalities of intersections between a lattice and a cube. This can be done using a theorem due to Lenstra ([Len88], Lemma 4.1), which we cite in a slightly different phrasing:

Theorem 6.8 *Let $\Gamma' \subseteq \mathbb{Z}^n$ be a full lattice of determinant D and W' an axes parallel cube with integral vertices and side length $w-1$. Then*

$$\frac{1}{D}\left(1 - \frac{D-1}{w}\right) w^n \leq |\Gamma' \cap W'| \leq \frac{1}{D}\left(1 + \frac{D-1}{w}\right) w^n.$$

Applying the theorem to our situation we find that

$$\begin{aligned}
\frac{|\Gamma \cap W^-|}{E^n} &\geq \frac{\frac{1}{N}\left(1 - \frac{N-1}{E-d}\right)(E-d)^n}{E^n} \\
&= \frac{1}{N}\left(1 - \frac{N-1}{E-d}\right)\left(1 - \frac{1}{E/d}\right)^{\frac{E}{d}\frac{nd}{E}} \\
&\geq \frac{1}{N}\left(1 - \frac{1}{5}\right)\left(1 - \frac{1}{E/d}\right)^{\frac{E}{d}\frac{1}{40}} \\
&\quad \text{for } E \geq 5N + d \text{ and } E \geq 40nd\text{; since } n \text{ is subexponential} \\
&\quad \text{in } \log N',\ d \in O^{\sim}(1) \text{ and } E \geq N', \text{ the second condition} \\
&\quad \text{is fulfilled asymptotically} \\
&\geq \frac{4}{5\sqrt[40]{e}N} \quad \text{where } e \text{ is Euler's constant;}
\end{aligned}$$

and similarly

$$\frac{|\Gamma \cap W^+|}{E^n} \leq \frac{6\sqrt[40]{e}}{5N}.$$

Thus we have shown the following result:

Theorem 6.9 *If $E \geq \max\{5N + d, 40nd\}$ and $\mathbf{e}$ is chosen uniformly in the range $\{0, \ldots, E-1\}^n$, then the probability of finding a relation lies between*

$$\frac{4}{5\sqrt[40]{e}}\frac{N_S}{N} \quad \text{and} \quad \frac{6\sqrt[40]{e}}{5}\frac{N_S}{N}.$$

6.3.2 Linear algebra

Since we are dealing with matrices of subexponential size, we must make sure that all matrix operations take time polynomial in the sizes of the matrices and their entries. Moreover, the exact exponents of the polynomial time bounds for the matrix operations have a direct impact on the constant of the subexponential time bound for the algorithm. Hence a judicious arrangement of the computations is necessary. We need the following results.

Theorem 6.10 *Let $A = (a_{ij}) \in \mathbb{Z}^{n \times m}$ with $m \geq n$, $|A| = \max\{|a_{ij}| : i = 1, \ldots, n,\ j = 1, \ldots, m\}$ and $f(k) = k \log^2 k \log\log k$.*

1) *The rank of A can be determined in time $O(mn^2 f(n \log(n|A|)))$.*

2) *If A has full rank n, then its Hermite normal form can be computed in time $O(mn^2 f(n \log(n|A|)))$.*

3) If $B \in \mathbb{Z}^{n\times n}$ is the essential part of the Hermite normal form of A and N its determinant, then its elementary divisor form Δ and the unimodular matrix S such that $\Delta = SB$ can be computed in time $O(n^3 \log N f(\log N))$.

Proof: See [HM91], Proposition 2.3, Corollary 2.2 and Theorem 2.6. Notice that $|B| \leq N$. By the remark in the last paragraph of [HM91], the multiplier matrices are obtained within the same time bound when the original matrix is square and of full rank, which is the case in 3). □

6.3.3 Success probability

In this section we estimate the probability that one run of Steps 1) to 5) in Algorithm 6.5 determines the correct group structure, in which case the discrete logarithm computation is bound to succeed.

While according to Section 6.3.1 there is a positive probability of finding relations, there is no guarantee that a new relation is not already contained in the lattice generated so far. Hence there is a small chance that the $20n$ relations created in Step 2) of Algorithm 6.5 do not generate a full lattice.

Formally, assume that during the algorithm we have generated a sublattice Γ_1 of Γ of dimension less than n, and let $\mathbf{c}$ be a further relation as determined in Step 2). We call $\mathbf{c}$ *useful* if it increases the dimension of Γ_1, and it is our aim to determine the probability that a newly created relation is useful.

Denote by $\Gamma_2 \subsetneq \Gamma$ a full sublattice which contains $\mathbb{Q}\Gamma_1 \cap \Gamma$. Then all relations outside Γ_2 are useful, and the probability of finding a relation within Γ_2 is by Corollary 6.7 bounded above by

$$\begin{aligned}\frac{|\Gamma_2 \cap W^+|}{E^n} N_S &\leq \frac{1}{kN}\left(1 + \frac{kN-1}{E+d}\right)\left(\frac{E+d}{E}\right)^n N_S \\ &\quad \text{by Theorem 6.8, where } k \geq 2 \text{ is the index of } \Gamma_2 \text{ in } \Gamma \\ &\leq \frac{7\sqrt[40]{e}}{10}\frac{N_S}{N}\end{aligned}$$

by arguments analogous to those used in the proof of Theorem 6.9.

Hence the conditional probability that a newly found relation is useless, which is the probability of finding a useless relation divided by the probability of finding any relation, is bounded above by $\frac{7\sqrt[40]{e}N_S}{10N} / \frac{4N_S}{5\sqrt[40]{e}N} = \frac{7\sqrt[20]{e}}{8} < \frac{18}{19}$ according to Theorem 6.9, and the probability that a newly found relation is useful is at least $1/19$.

Theorem 6.11 *The probability of success for one run of the algorithm is asymptotically 1 for $N' \to \infty$.*

Proof: We first compute the probability that the matrix A obtained after Step 2) has full rank, which is equivalent to saying that n of the $20n$ relations computed are useful. Let X denote a Binomial$(20n, 1/19)$-distributed random variable. By the discussion above, the probability that A has full rank is at least

$$\begin{aligned} P(X \geq n) &\geq 1 - P\left(\left|X - \frac{20}{19}n\right| \geq \frac{1}{19}n\right) \\ &\geq 1 - \frac{\mathrm{Var}(X)}{\frac{1}{361}n^2} \end{aligned}$$

by Tschebyscheff's inequality (see any statistics textbook). Since $\mathrm{Var}(X) = \frac{360}{361}n$, the matrix A has full rank with probability at least $1 - \frac{360}{n}$, which tends to 1 for $N' \to \infty$.

A similar reasoning applies to Step 4). Now let the full lattice $\Gamma_2 \subsetneq \Gamma$ be already generated, and call a relation useful if it decreases the index of Γ_2 in Γ. Then the same reasoning as above shows that a new relation is useful with probability at least $1/19$. We now have to estimate the number of useful relations needed to find a generating system of Γ. Let Γ_1 be the lattice obtained in Step 2). Then the number of useful relations needed for Γ is bounded above by

$$\mathrm{ld}[\Gamma : \Gamma_1] = \mathrm{ld}(\det \Gamma_1) - \mathrm{ld}(\det \Gamma) \leq \mathrm{ld}(\det \Gamma_1).$$

From the description of the relation collecting phase in Algorithm 6.5 we know that all relations constructed lie in the cube $\{1 - E, \ldots, d\}^n$. Now Hadamard's upper bound shows that $|\det \Gamma_1| \leq (\sqrt{n}E)^n \leq E^{2n}$ at least asymptotically since n is subexponential and E exponential in $\log N'$. Hence,

$$\mathrm{ld}(\det \Gamma_1) \leq 2n \,\mathrm{ld}\, E.$$

Simulating the creation of relations again by a binomially distributed variable shows that the probability of obtaining a generating system is asymptotically 1 for $N' \to \infty$. □

6.3.4 Running time

Recall that the complexity of one group operation is on $O^{\sim}(1)$ by the assumptions set forth in Section 5.1. Moreover, multiplying a group element by an integer in $\{0, \ldots, E-1\}$ by a double and add scheme takes time in $O^{\sim}(\log E) = O^{\sim}(1)$.

Let t_s and t_d be upper bounds on the expected time needed for a smoothness test and the decomposition of a smooth group element into a sum of primes. By the assumptions of Section 5.1, we have $t_s,\ t_d \in O^{\sim}(d + n) = O^{\sim}(n)$, for instance by realising the decomposition through trial division by the elements of $\mathcal{P}_S$.

We determine the expected time needed for one run of the algorithm, assuming that each step is executed only once and no jump back to Step 2) is required. By assumption, the factor base $\mathcal{P}_S$ can be constructed in time in $O^\sim(n^2)$. The time needed for computing a linear combination of the n prime elements and testing it for smoothness is in $O^\sim(n+t_s) = O^\sim(n)$, which also covers the possible decomposition in the case of smoothness, and by the assumptions of Section 5.1, at least one out of $2 \in O(1)$ smooth elements is recognised as such. From Theorem 6.9 we know that the expected number of trials for obtaining a smooth element is in $O\left(\frac{N}{N_S}\right)$. Hence the expected time needed for carrying out Step 2) is in $O^\sim\left(n^2 \frac{N}{N_S}\right)$. The same bound holds for Step 4) since $\log E \in O^\sim(1)$.

Observing that the entries in A lie between 0 and $d \in O^\sim(1)$ and that the number of columns of A is in $O^\sim(n)$, the discussion of Section 6.3.2 shows that the rank of A and its elementary divisor form are computed in $O^\sim(n^4)$. So the total time for one execution of the steps of Algorithm 6.5 is in $O^\sim\left(n^2 + n^4 + n^2 \frac{N}{N_S}\right)$.

Smoothing the elements g_1 and g_2 as described in Section 6.2.2 is performed in expected time $O^\sim\left(n \frac{N}{N_S}\right)$; this follows from arguments analogous to those of Section 6.3.1. With the notation of Section 6.2.2, computing $\mathbf{a}^{(1)}$ and $\mathbf{a}^{(2)}$ and solving the system of equations modulo the d_i takes time in $O^\sim(n^2)$ since the transformation matrix S is known from the computation of the elementary divisor form of A. Hence, these steps are dominated by the running time of Algorithm 6.5.

Since by the analysis of Section 6.3.3 the complete algorithm has to be repeated only an expected $O(1)$ number of times, the overall expected complexity is

$$O^\sim\left(n^2 + n^4 + n^2 \frac{N}{N_S}\right) \subseteq O^\sim\left(n^4 + n^2 \frac{N}{N_S}\right).$$

6.3.5 Subexponentiality

Assume that the bound S can be chosen such that

$$n \in O(L_{N'}(\rho + o(1)))$$

and

$$\frac{N}{N_S} \in O(L_{N'}(\sigma + o(1)))$$

for some constants $\rho, \sigma > 0$. Then the expected running time of the algorithm is in

$$O\left(L_{N'}\left(\max(4\rho, 2\rho + \sigma) + o(1)\right)\right).$$

Examples.

1) Ideal class groups of imaginary quadratic number fields of discriminant D

Theorem 5.6 shows that with $S = \lceil \log L_N(\rho) \rceil$ and assuming the validity of the generalised Riemann hypothesis, we have $\sigma = \frac{1}{2\rho}$. Then the running time of the algorithm is in

$$O\left(L_N\left(\max\left\{4\rho, 2\rho + \frac{1}{2\rho}\right\} + o(1)\right)\right)$$

for any $\rho > 0$.

To find an optimal value for the parameter ρ, we have to minimise the function $\rho \mapsto \max\left\{4\rho, 2\rho + \frac{1}{2\rho}\right\}$, which is unimodal and thus has a unique global minimum. The strictly convex function $\rho \mapsto 2\rho + \frac{1}{2\rho}$ admits its unique global minimum at

$$\overline{\rho} = \frac{1}{2}.$$

Both functions agree for

$$\rho^* = \frac{1}{2}.$$

Hence f admits its unique minimum at $\min\{\overline{\rho}, \rho^*\} = \frac{1}{2}$. This proves the following result.

Theorem 6.12 *Assuming the generalised Riemann hypothesis, the algorithm of Section 6.2 computes discrete logarithms in the ideal class groups of imaginary quadratic number fields of discriminant D in expected time in*

$$O\left(L_N\left(2 + o(1)\right)\right) = O\left(L_{|D|}\left(\sqrt{2} + o(1)\right)\right).$$

2) Jacobians of imaginary quadratic hyperelliptic curves over finite fields

As explained in Section 6.1.3, we consider only problem instances with $g \geq \vartheta \log q$ for some positive constant ϑ. With $S = \lceil \log_q(L_{q^g}(\rho)) \rceil$, Theorem 6.3 shows that $n \in L_{q^g}\left(\rho + \frac{1}{\sqrt{\vartheta}} + o(1)\right)$ and Theorem 5.10 shows that $\frac{N}{N_S} \in O^{\sim}\left(\frac{N'}{N_S}\right) \subseteq O\left(L_{q^g}\left(\frac{1}{2\rho} + o(1)\right)\right)$. Thus, the running time of the algorithm is in

$$O\left(L_{q^g}\left(\max\left\{4\left(\rho + \frac{1}{\sqrt{\vartheta}}\right), 2\left(\rho + \frac{1}{\sqrt{\vartheta}}\right) + \frac{1}{2\rho}\right\} + o(1)\right)\right)$$

for any $\rho > 0$.

Repeating the optimisation step for ρ as above, we find that $\overline{\rho} = \frac{1}{2}$ and $\rho^*(\vartheta) = \frac{1}{2}\left(\sqrt{1 + \frac{1}{\vartheta}} - \sqrt{\frac{1}{\vartheta}}\right)$ depending on ϑ. Since $\rho^*(\vartheta) < \overline{\rho}$ for all positive values of ϑ, the optimal value for ρ is $\rho^*(\vartheta)$. This proves the following result.

Theorem 6.13 *The algorithm of Section 6.2 computes discrete logarithms in the Jacobians of imaginary quadratic hyperelliptic curves of genus g over finite fields $\mathbb{F}_q$ satisfying $g \geq \vartheta \log q$ for some positive constant ϑ in expected time in*

$$O\left(L_{q^g}\left(2\left(\sqrt{1+\frac{1}{\vartheta}}+\sqrt{\frac{1}{\vartheta}}\right)+o(1)\right)\right).$$

If $g/\log q$ tends to infinity for the instances under consideration, e.g. if q is constant and $g \to \infty$, then the complexity is

$$O\left(L_{q^g}\left(2+o(1)\right)\right).$$

□

6.4 Previous algorithms

The algorithm above was first described by Hafner and McCurley in the special setting of imaginary quadratic number fields ([McC89, HM89]).

Buchmann generalised Hafner and McCurley's algorithm to determine the class group structure of an arbitrary number field in [Buc90]. Since for number fields of degree larger than 2 no smoothness result has been verified, the subexponential running time of his algorithm remains a conjecture.

Adleman, DeMarrais and Huang were the first to consider hyperelliptic Jacobians and to devise a discrete logarithm algorithm with a conjectured subexponential running time ([ADH94]). The basic difference to our algorithm is the way in which the relations are created: Adleman, DeMarrais and Huang randomly choose polynomial functions $ay + b \in \mathbb{F}_q[C]$ with a, $b \in \mathbb{F}_q[X]$ until the divisor of the corresponding principal ideal is smooth, whence it forms a relation. In our general model this corresponds to directly choosing elements of the submonoid $\mathbb{M}_0 = \{m \in \mathbb{M} : m \sim 0\}$ of $\mathbb{M}$ and decomposing them over the factor base. This generalisation, however, is not completely satisfying, since $\mathbb{M}_0$ is not accessible in an abstract way, so that the selection of elements of $\mathbb{M}_0$ has to be described for each type of discrete logarithm problem separately.

Maybe, this is also the source of problems for proving a subexponential running time of the algorithm in [ADH94]. On one hand, we can work with a nearly uniform distribution over a finite set, namely the elements of G, in our algorithm, which allows to derive assertions on the probability of finding a relation. On the other hand, the coefficients of our relations may become as large as about $E \approx N'$, so that we have the potential of reaching all relations corresponding to elements of $\mathbb{M}_0$ of degree up to about nN', cf. Corollary 6.7. In the algorithm of [ADH94], one needs to fix an at most subexponential

bound for the degrees of a and b, so that only elements of $\mathbb{M}_0$ of subexponential degree can be constructed. It is even unclear whether this smaller part of $\mathbb{M}_0$ suffices to generate the complete relation lattice Γ.

In its present formulation, the algorithm of Section 6.2 does not allow to compute discrete logarithms in Jacobians of real quadratic hyperelliptic curves over finite fields; in fact, there is no known way to realise the arithmetic of the Jacobian in this case. Instead, it has been suggested to work with a different structure, the so-called *infrastructure*. Although the infrastructure is not a group, it is possible to define an arithmetic and a discrete logarithm problem and to base cryptosystems on it ([SSW96, MVZ98]). In [MST99], the authors describe an algorithm for solving the infrastructure logarithm problem over a field of odd characteristic which follows the same principles as ours. Assuming the validity of Theorem 5.10, they are able to prove a subexponential running time. Considering the case $\vartheta = 1$, they unfortunately do not take into account that the smoothness bound S has to be rounded up. Their assumption that $q^S \in L_{q^g}(\rho + o(1))$ and consequently their running time analysis are, however, correct for $g/\log q \to \infty$. Since their analysis is based on a higher complexity of the elementary divisor form computation, their running time is worse than that of our algorithm. Correcting the analysis of [MST99] and using Theorem 6.10, the following result can be proved:

Theorem 6.14 *There is a subexponential algorithm computing discrete logarithms in the infrastructures of real quadratic hyperelliptic curves of genus g over finite fields $\mathbb{F}_q$ of odd characteristic satisfying $g \geq \vartheta \log q$ for some positive constant ϑ. Its expected running time is in*

$$O\left(L_{q^g}\left(2\left(\sqrt{1+\frac{1}{\vartheta}}+\sqrt{\frac{1}{\vartheta}}\right)+o(1)\right)\right).$$

If $g/\log q$ tends to infinity for the instances under consideration, e.g. if q is constant and $g \to \infty$, then the complexity of the algorithm is

$$O\left(L_{q^g}\left(2+o(1)\right)\right).$$

Using the theory of real quadratic hyperelliptic curves over fields of characteristic 2 developed in [MVZ98] it should be possible to extend the theorem to curves over arbitrary finite fields.

We noted in Section 3.2.1 that the existence of a ramified prime divisor of degree 1 allows to transform a real into an imaginary quadratic hyperelliptic curve with the same function field. After a constant field extension of degree at most $2g+2$, such a divisor always exists. Paulus and Rück observed that under this transformation the infrastructure logarithm problem on the real quadratic curve and the logarithm problem in the Jacobian of the imaginary quadratic curve are equivalent. (In [PR99], the authors consider only the case of odd characteristic, but the result should carry over to general curves.)

6.5 Group structure

If g_2 lies in the subgroup generated by g_1, then our algorithm is bound to ultimately find its discrete logarithm. Otherwise, i.e. if no discrete logarithm exists, which is never the case in the cryptographic context, it will run forever. This problem can be solved by taking into account that one run of the algorithm may fail for two reasons: It fails if Algorithm 6.5 does not determine the correct group structure or if no discrete logarithm exists. So it is of interest to decide when the first case occurs.

Notice that in the notation of Section 6.2, the determinant $d_1 \cdots d_n$ of the elementary divisor form of A is a multiple of N; it equals N if and only if the columns of A generate the relation lattice Γ, which means that the group structure problem for G is solved correctly. Thus, it is sufficient to know a bound $\overline{N}$ on the group order such that $N \leq \overline{N} < 2N$. Then $d_1 \cdots d_n$ is the correct group order N if and only if it does not exceed $\overline{N}$. It has been mentioned in Section 6.1.2 that the problem of computing such a bound $\overline{N}$ in polynomial time has been solved for imaginary quadratic number fields. The method is generalised to computing an analogous bound on the cardinality of real quadratic Jacobians over finite fields of odd characteristic in [Ste96]. A similar approach should be possible to derive a bound for Jacobians of imaginary quadratic hyperelliptic curves over any finite field.

6.6 Implementation

The algorithm is clearly not suited for a direct implementation in its present formulation. We present a few practical improvements for hyperelliptic Jacobians, which belong to two distinct categories.

Some practical enhancements do not alter the rigorous proof of the running time, but we did not care to include them into the description because they result in only a polynomial speed-up, which would have vanished in the $o(1)$ term. A first simple observation is that if $\mathfrak{P}$ is a splitting prime divisor, then $\overline{\mathfrak{P}} \sim -\mathfrak{P}$, so that it is sufficient to include only one out of each pair of splitting prime divisors into the factor base, if at the same time negative coefficients for the decomposition into primes are allowed. So the size of the factor base can be halved approximately. From a more abstract point of view, we have used the hyperelliptic involution σ, which is the unique non-trivial automorphism of $K(C)/K(X)$, and the known relation that $\sigma(\mathfrak{P}) = \nu_{\mathfrak{P}}\mathfrak{P}$ with some known constant $\nu_{\mathfrak{P}} \in \mathbb{Z}$ for any prime divisor $\mathfrak{P}$ to keep only one prime divisor in each orbit of σ in the factor base. This approach can sometimes be generalised to more general automorphisms of $K(C)$, which need not fix $K(X)$; for instance, if $K = \mathbb{F}_{p^m}$, the Frobenius automorphism of raising to the p-th power might be employed. The method is effective if integers $\nu_{\mathfrak{P}}$ do exist and can be determined easily. For details, cf. [Gau00].

In practice, computing $20n + 40n \log E$ relations seems exaggerated; one would expect that in the beginning each additional relation will increase the rank of A by 1 and that only slightly more than n relations will suffice to generate Γ. However, the obtainable speed-up would again be only polynomial, so from a theoretical point of view there is no reason to refine the argumentation. In practice, it is recommendable to create some more than n relations and — instead of restarting the complete algorithm in case it fails — possibly add further relations and use incremental algorithms to update the elementary divisor form.

The second type of improvements provides a speed-up also in theory, i.e. a better constant in the subexponential complexity. Unfortunately, it introduces less randomness so that the better running time cannot be proved rigorously. Recall the complexity of

$$O^{\sim}\left(n^4 + n^2 \frac{N}{N_S}\right)$$

proved in Section 6.3.4, the first term of which stems from the complexity of the elementary divisor form computation and the second one from the time needed to find the relations. Since the parameter ρ is chosen ultimately so as to balance the contribution of the two terms, an improvement in any of them would result in a better overall complexity.

The elementary divisor form is essential for the group structure determination; we develop an algorithm for which the linear algebra step is faster in Chapter 7, provided that the group structure is already known. Concerning the relation finding step, the computation of a random linear combination of n prime divisors has a complexity in $O^{\sim}(n)$. This can be dropped to $O^{\sim}(1)$ if it is replaced by a random walk in which in each step a random multiple of only one random prime divisor is added to the previous combination. The heuristic complexity is then lowered to

$$O^{\sim}\left(n^4 + n \frac{N}{N_S}\right)$$

provided that the smoothness test and the decomposition are dominated by the other steps; however, the constructed group elements are not distributed independently and the rigorous analysis does not apply any more. Repeating the optimisation of ρ as in Section 6.3.5, the heuristic running time bound becomes

$$O\left(L_{q^g}\left(\frac{2\sqrt{6}}{3}\left(\sqrt{1 + \frac{3}{2\vartheta}} + \sqrt{\frac{3}{2\vartheta}}\right) + o(1)\right)\right)$$

for hyperelliptic Jacobians with $g \geq \vartheta \log q$ and

$$O\left(L_N\left(\frac{2\sqrt{6}}{3} + o(1)\right)\right) = O\left(L_{|D|}\left(\frac{2}{\sqrt{3}} + o(1)\right)\right)$$

for imaginary quadratic number fields of discriminant D. In any case, this is still worse than the fully proved complexity of the algorithm of the next chapter. An alternative would be to find the relations by sieving techniques as common for finite fields.

It should also be noted that the relation collection phase can be parallelised to an arbitrary extent with a gain in speed proportional to the number of participating machines. Communicational overhead is only produced when a new relation is found, which has to be sent to a central processor to be included into the matrix. So the practical bottleneck is the linear algebra step, and a parallel algorithm for computing the elementary divisor form would push the applicability of the algorithm much further.

Hafner and McCurley's algorithm of [HM89] for imaginary quadratic number fields has been implemented by Düllmann ([Dül91]), who solved the discrete logarithm problem for discriminants of about 40 decimal digits. More recently, Jacobson presented an implementation running successfully on instances with an 80 digit discriminant in [Jac00]. The speed-up is mostly due to the use of sieving techniques. For hyperelliptic Jacobians, I know of only one implementation, which also involves sieving for finding the relations ([FP99]).

Chapter 7

Subexponential algorithms for groups with known order

The running time of the algorithm of Chapter 6 on class groups is substantially worse than that of even fully proved algorithms for finite fields. As mentioned in Section 6.6, this is due to the fact that the linear algebra step for finding the group structure is simply too costly. Now, the group structure is usually known in the cryptographic setting, in which the discrete logarithm problem has to be solved in a cyclic group of known order. In this chapter, we develop a fast subexponential algorithm for computing discrete logarithms in the case that the group order is known. The algorithm was conceived by Gaudry for hyperelliptic curves ([Gau00]), but it generalises nicely to our abstract model. Like the algorithm of Chapter 6 it consists of two phases, a relation collection and a linear algebra step. Its main gain in efficiency stems from the linear algebra step, where sparse matrix techniques can be employed. To this purpose, we extend some results on sparse linear algebra, see Section 7.2. Additionally, the relations are created faster.

It turns out that all examples for which smoothness results are available have basically the same complexity of

$$O(L_N(\sqrt{2} + o(1)).$$

(If there is an additional parameter ϑ, the bound is valid for $\vartheta \to \infty$.) This shows that class groups (with a large value of ϑ) are as vulnerable to subexponential attacks as finite fields if no algorithms with an unproven running time are taken into account.

It is common practice in the literature to distinguish between impractical algorithms with a proven and practical algorithms with only a conjectured subexponential complexity; the distinction is particularly true for finite fields, in which all proven algorithms have a running time in $L_N(1/2, c)$, while the number field and function field sieve algorithms have a conjectured running time in $L_N(1/3, c)$. Our algorithm shows that this distinction

is no more valid for class groups. While being the fastest one suggested so far, our algorithm can nevertheless be implemented with only minor changes.

For the time being, assume that G is an arithmetical formation (see Section 5.1) of known cardinality N. Let again be N' the parameter describing the input size of the problem, i.e., $N' = N$ unless G is the Jacobian of a hyperelliptic curve, in which case $N' = q^g$. Later on, we consider the situation in which discrete logarithms are sought in a cyclic subgroup of an arithmetical formation. This poses the problem that smoothness results are usually only available for the full group and not for its subgroups. The same problem occurred in Chapter 6, but there it was less important since we had to work with the complete group anyway to determine its structure. A few possible solutions are discussed in Section 7.4. While they do not cover all possible groups, we show that the running time analysis remains valid at least for all instances of cryptographic interest.

It may seem tempting to factor N and to use a Pohlig–Hellman type approach. However, this reintroduces the problem of working in subgroups by the back door. While solving the discrete logarithm problem directly in the cyclic group G, we nevertheless factor N during the algorithm to simplify the linear algebra step.

7.1 Algorithm

Let g_1 be a generator and g_2 a further element of G. The following algorithm determines the unique integer $l \in \mathbb{Z}_N = \{0, \ldots, N-1\}$ such that $g_2 = lg_1$.

Algorithm 7.1

1) *Choose a smoothness bound S and construct the factor base $\mathcal{P}_S = \{\mathfrak{P}_1, \ldots, \mathfrak{P}_n\}$ with $n = n_S$. Set $k = \lceil \mathrm{ld}(n \,\mathrm{ld}\, N) \rceil + 1$.*

2) *Construct a matrix $A = (a_{ij}) \in \mathbb{Z}_N^{n \times (2kn)}$ as follows: For $j = 1, \ldots, kn$, select randomly and uniformly α_j, $\beta_j \in \mathbb{Z}_N$ until $\alpha_j g_1 + \beta_j g_2$ is S–smooth, and write*

$$\alpha_j g_1 + \beta_j g_2 = \sum_{i=1}^{n} a_{ij} \mathfrak{P}_i.$$

For $j = kn+1, \ldots, 2kn$, write $j = (k+r)n+m$ with $0 \le r \le k-1$, $1 \le m \le n$, and select randomly and uniformly α_j, $\beta_j \in \mathbb{Z}_N$ until $\alpha_j g_1 + \beta_j g_2 - \mathfrak{P}_m$ is S–smooth; then write

$$\alpha_j g_1 + \beta_j g_2 = \mathfrak{P}_m + \sum_{i=1}^{n} b_{ij} \mathfrak{P}_i = \sum_{i=1}^{n} a_{ij} \mathfrak{P}_i.$$

3) By the randomised procedure described in Section 7.2, try to find a non-zero vector $\gamma = (\gamma_1, \ldots, \gamma_{2kn}) \in \mathrm{Ker}(A)$ *in the kernel of* A*. If the procedure fails, go back to 2).*

4) If $\sum_{j=1}^{2kn} \beta_j \gamma_j$ *is invertible in* $\mathbb{Z}_N$*, then output*

$$l = -\left(\sum_{j=1}^{2kn} \beta_j \gamma_j\right)^{-1} \left(\sum_{j=1}^{2kn} \alpha_j \gamma_j\right);$$

otherwise go back to 2).

If the algorithm halts in Step 4), then it outputs the correct discrete logarithm of g_2 to the base g_1. The fact that $\gamma \in \mathrm{Ker}(A)$ means that

$$0 = \sum_{j=1}^{2kn} a_{ij}\gamma_j \quad \forall i = 1, \ldots, n;$$

multiplying these equations by $\mathfrak{P}_i$ and summing them up yields

$$0 = \sum_{j=1}^{2kn} \left(\sum_{i=1}^{n} a_{ij}\mathfrak{P}_i\right) \gamma_j = \left(\sum_{j=1}^{2kn} \alpha_j \gamma_j\right) g_1 + \left(\sum_{j=1}^{2kn} \beta_j \gamma_j\right) g_2.$$

As g_1 and g_2 are both of N–torsion, multiplying by the inverse of $\sum_{j=1}^{2kn} \beta_j \gamma_j$ in $\mathbb{Z}_N$, if it exists, shows the correctness of the result.

7.2 Linear algebra

Since $\mathrm{rank}\, A \leq n$, it is possible to find a non-zero vector $\gamma \in \mathrm{Ker}(A)$. How this is done, however, needs further explication. On one hand, it is desirable to exploit the sparse structure of the matrix, which has only $O^{\sim}(1)$ entries per column, and the corresponding algorithms are prone to failure with a certain probability. On the other hand, algorithms for sparse linear algebra are usually described for fields in the literature. Thus, a complication is introduced by the fact that N need not be prime and $\mathbb{Z}_N$ may not be a field.

To exploit the matrix sparseness, one may use a randomised Lanczos algorithm; we rely on the following trivial corollary of Theorem 6.2 in [EK97].

Theorem 7.2 *Let* $\mathbb{F}_q$ *be the finite field with* q *elements and let* $A \in \mathbb{F}_q^{n \times d}$ *be a matrix of rank* r *with* ω *non-zero entries and* $b \in \mathbb{F}_q^n$*. There is a probabilistic algorithm which either returns a vector* $x \in \mathbb{F}_q^d$ *such that* $Ax = b$ *or reports failure. The algorithm requires* $O(r(\omega + d))$ *operations in* $\mathbb{F}_q$ *and has a failure probability of at most* $\frac{11d^2 - d}{2(q-1)}$*.*

Moreover, the solution vector returned by the algorithm can be made to vary uniformly over all possible solutions by randomising the right hand side in the following way (in fact, this randomisation is already part of the algorithm in [EK97]): Choose $y \in \mathbb{F}_q^d$ according to a uniform distribution, solve $A\overline{x} = b + Ay$ and let $x = \overline{x} - y$. If y varied over a fixed class of $\mathbb{F}_q^d / \operatorname{Ker} A$, then $\overline{x}$ would not depend on y, and x would be distributed uniformly over the solution space $\overline{x} + \operatorname{Ker} A$ of the equation. Hence, the same assertion holds when y does not belong to a fixed class.

When q is small compared to d, it is not possible to apply the theorem directly. Instead, one may switch to a field extension. While this idea does not seem to be new — it was used, for instance, in the implementation of [LO91], see also [KS91] — I did not find it detailed in the literature and thus expand on the topic. In particular, it is possible to maintain the uniform distribution over the solution vectors.

In the situation of Theorem 7.2, let p be the characteristic of $\mathbb{F}_q$, $\nu = \min\{l : q^l > 11d^2,\ p \nmid l\}$ and $q' = q^\nu$. Then $q^{\nu-2} \leq 11d^2$, so that $q' \in O(d^2q^2)$. We would like to solve a matrix equation over $\mathbb{F}_{q'}$ and project the solution onto a solution $x \in \mathbb{F}_q^d$ of $Ax = b$. For projection, one may use the trace function $\operatorname{Tr} : \mathbb{F}_{q'} \to \mathbb{F}_q$, which is a homomorphism of $\mathbb{F}_q$-vector spaces and acts on $\mathbb{F}_q$ as multiplication by ν. It can be extended to a map $\operatorname{Tr} : F_{q'}^d \to \mathbb{F}_q^d$ by componentwise application. Let $b' = \nu' b \in \mathbb{F}_q^d$ with $\nu\nu' \equiv 1 \pmod{p}$, so that $\nu b' = b$. The value ν' exists because $\gcd(\nu, p) = 1$ and can be computed by the extended Euclidean algorithm in time $O(\log \nu \log p)$, which as well as the multiplication of b by ν' is negligible compared to the following linear algebra step. Solve $Ax' = b'$ by the algorithm in [EK97]. The success probability for this step is at least $1 - \frac{11d^2 - d}{2(q'-1)} \geq \frac{1}{2}$. Let $x = \operatorname{Tr}(x')$. Then from the linearity of the trace we deduce that $b = \nu b' = \operatorname{Tr}(b') = \operatorname{Tr}(Ax') = Ax$.

Moreover, any solution $x \in \mathbb{F}_q^d$ of $Ax = b$ can be obtained in this way, and all of them have the same probability of occurring. Namely, for a given solution x, the set of solutions to $Ax' = b'$ over $\mathbb{F}_{q'}^d$ which map to x under the trace function is given by $\nu' x + (\operatorname{Ker} A \cap \operatorname{Ker} \operatorname{Tr})$, whose cardinality $(q')^{\dim(\operatorname{Ker} A \cap \operatorname{Ker} \operatorname{Tr})}$ is independent of x. Thus, we have shown the following result.

Theorem 7.3 *Let $A \in \mathbb{F}_q^{n \times d}$ be a matrix of rank r with ω non-zero entries and $b \in \mathbb{F}_q^n$. There is a probabilistic algorithm which either returns a vector $x \in \mathbb{F}_q^d$ such that $Ax = b$ or reports failure. The running time of the algorithm is in $O(r(\omega + d)\log^2(dq))$, and its failure probability is at most $\frac{1}{2}$. Moreover, the resulting vector is uniformly distributed over all possible solutions.*

This solves the linear algebra step if N is prime. Otherwise, one factors N, computes γ modulo p^ν for all $p^\nu \| N$ and combines the results by the Chinese Remainder Theorem. The computations modulo p^ν may be broken up into ν iterations modulo p via a lifting procedure: Suppose that a non-zero solution $\gamma_1 \in \{0, \ldots, p^e - 1\}^{2kn}$ is known to the

equation $Ax \equiv 0 \pmod{p^e}$, for instance $A\gamma_1 = p^e\delta$ with $\delta \in \mathbb{Z}^{2kn}$. Assume that there is a solution γ_2 of $Ax \equiv \delta \pmod{p}$. Then $p^e\gamma_2 - \gamma_1$ is a non-zero solution of $Ax \equiv 0 \pmod{p^{e+1}}$. If all computations modulo a prime return a random vector according to a uniform distribution over all possible solutions, then the combined result varies uniformly over the kernel of A.

Considering the elementary divisor form of the matrix A, however, it is easily seen that the lifting procedure may fail if (and only if) $\operatorname{rank}_{\mathbb{Q}} A \neq \operatorname{rank}_{\mathbb{Z}_p} A$ because then the matrix equation $Ax \equiv \delta \pmod{p}$ need not have a solution. This is the reason why, following [Pom87], we create the matrix A in a special way, generating many more than the $n+1$ columns one would expect to need in practice and forcing the basis elements $\mathfrak{P}_m$ into the relations. Indeed, it is proved in Lemma 4.1 and the subsequent remark of [Pom87] that with high probability the matrix has full rank over $\mathbb{Z}_p$. We recall this lemma in our notation.

Lemma 7.4 *Let V be a vector space over a field $\mathbb{F}$ with $\dim V = n < \infty$. Let $\mathcal{S}$ be a finite set of vectors in V and $b_1, \ldots, b_n$ a basis for V. Let $k \in \mathbb{N}$. We make $2kn$ independent choices of elements from $\mathcal{S}$ with an arbitrary probability distribution over $\mathcal{S}$, labelling the chosen vectors $v_1, \ldots, v_{kn}, w_1, \ldots, w_{kn}$, and we denote by V' the subspace of V spanned by $v_1, \ldots, v_{kn}$, and the vectors $b_j + w_{(j-1)k+i}$ for $j = 1, \ldots, n$ and $i = 1, \ldots, k$. Then with probability at least $1 - \frac{n}{2^{k-1}}$ we have $V = V'$.*

In our case, the vector space V is the space of column vectors of size n with coefficients in $\mathbb{Z}_p$, the basis is the canonical basis, and the set $\mathcal{S}$ is the set of all column vectors representing a smooth element of G. We see that the vectors generating V' correspond precisely to the vectors forming the matrix A. Hence the probability that the lifting is possible on $\mathbb{Z}_p$ is at least $1 - \frac{n}{2^{k-1}}$. There are at most $\frac{\operatorname{ld} N}{2}$ distinct primes p whose squares divide N, thus the probability that the lifting is possible for all of them is at least $1 - \frac{n \operatorname{ld} N}{2^k} \geq \frac{1}{2}$ for our choice of k.

In this case, repeating $\operatorname{ld}(2 \operatorname{ld} N)$ times the algorithm of Theorem 7.3, we obtain a solution of one problem modulo a prime with probability at least $1 - \frac{1}{2^{\operatorname{ld}(2 \operatorname{ld} N)}} = 1 - \frac{1}{2 \operatorname{ld} N}$. As at most $\operatorname{ld} N$ single problems have to be solved, we get a solution modulo N with probability at least $\frac{1}{2}$. Altogether, Step 3) is thus successful with a probability of at least $\frac{1}{4}$, in which case the output vector is uniformly distributed over the kernel.

7.3 Analysis

7.3.1 Success probability

To estimate the probability that the algorithm succeeds during one run of Steps 2) to 4), we assume that Step 2) has been accomplished successfully, the study of this step being postponed to the running time analysis below.

As shown in Section 7.2, Step 3) succeeds with probability at least $\frac{1}{4}$. The algorithm may also fail if $\sum_{j=1}^{2kn} \beta_j\gamma_j$ is not invertible in $\mathbb{Z}_N$ in Step 4). However, this happens with a sufficiently low probability. For given $j \leq kn$ and any β_j, as g_1 is a generator of G and α_j is uniformly distributed, the element $\alpha_j g_1 + \beta_j g_2$ is uniformly distributed over all group elements. The same holds for $j > kn$ and $\alpha_j g_1 + \beta_j g_2 - \mathfrak{P}_m$. Consequently, the matrix A and the vector β are independent random variables, so that γ and β are also independent. Let p be a prime divisor of N. Since γ is uniformly distributed over all vectors of the kernel, the probability that $\gamma \not\equiv 0 \pmod{p}$ is at least $1 - \frac{1}{p}$. Then the orthogonal space of $\gamma \bmod p$ in $\mathbb{Z}^{2kn}$ has dimension $2kn - 1$, and the conditional probability that $\beta \bmod p$ is not orthogonal to $\gamma \bmod p$ is at least $1 - \frac{1}{p}$. Hence $\sum_{j=1}^{2kn} \beta_j\gamma_j$ is invertible in $\mathbb{Z}_N$ with probability at least $\prod_{p|N} \left(1 - \frac{1}{p}\right)^2 = \left(\frac{\varphi(N)}{N}\right)^2$. From (3.41) in [RS62] we have $\frac{\varphi(N)}{N} \in \Omega(1/\log\log N)$.

Thus, the total success probability for one run of Steps 2) to 4) is in

$$\Omega\left(\frac{1}{(\log\log N)^2}\right).$$

7.3.2 Running time

Denote again by N_S the number of S–smooth group elements. Let t_s, t_d and t_f be upper bounds on the expected time needed for a smoothness test, the decomposition of a smooth group element into a sum of primes and the factorisation of N, respectively.

With the assumptions set forth in Section 5.1, $\mathcal{P}_S$ can be constructed in time $O^\sim\left(n^2\right)$. The time needed for computing one linear combination of g_1 and g_2 and testing for smoothness is in $O^\sim(t_s)$; this has to be repeated an expected $\frac{N}{N_S}$ times until a smooth element is obtained. This smooth element is recognised with a probability of at least $1/2$, so that no more than two repetitions of the previous procedure are needed on average until a column of the matrix can be filled. So the total time used in Step 2) is in

$$O^\sim\left(n\left(\frac{N}{N_S}t_s + t_d\right)\right) \subseteq O^\sim\left(n\frac{N}{N_S}t_s + n^2\right)$$

since $2kn$, $t_d \in O^\sim(n)$.

As explained in Section 7.2, Step 3) requires $\mathrm{ld}(2\,\mathrm{ld}\,N)\,\mathrm{ld}\,N \in O^\sim(1)$ executions of the algorithm behind Theorem 7.3. The number of entries in each column of A is in $O^\sim(1)$, so that Step 3) needs time in $O^\sim(t_f + n^2)$.

Finally, Step 4) can be performed in $O^\sim(1)$.

Since by the analysis of Section 7.3.1 only $O((\log\log N)^2) \subseteq O^\sim(1)$ repetitions of Steps 2) to 4) are needed on average, the total running time of the algorithm is in

$$O^\sim\left(t_f + n^2 + n\frac{N}{N_S}t_s\right).$$

Examples.

1) $G = \mathbb{F}_p^\times$, p prime

With deterministic algorithms due to Pollard and Strassen [Pol74, Str76] we have $t_s \in O^\sim(\sqrt{n})$. A more efficient probabilistic method has been proved using hyperelliptic curves. The test of [LPP93] recognises (and decomposes) a smooth number with probability at least $1/2$ in time $t_s \in O^\sim(L_n(2/3, c)$, where c is some positive constant. Thus, the total running time is in

$$O^\sim\left(t_f + n^2 + nL_n(2/3, c)\frac{N}{N_S}\right).$$

2) $G = \mathbb{F}_{2^k}^\times$

Now $t_s \in O^\sim(1)$ since a smoothness test can be performed in deterministic polynomial time by computing the distinct degree factorisation of the polynomial representing the group element. Precisely, let $f \in \mathbb{F}_2[X]'$ be the element to be tested, and $g = \frac{f}{\gcd(f,f')}$ its square-free part. Then f is S–smooth if and only if g is. Since $X^{2^i} - X$ is the product of all irreducible polynomials of degree dividing i in $\mathbb{F}_2[X]'$, the latter is the case if and only if

$$g = \operatorname{lcm}(\{\gcd(g, X^{2^i} - X) \,:\, i = 1, \ldots, S\}).$$

Computing $X^{2^i} - X \bmod g$ by successive squaring and reduction modulo g, this can be tested in time polynomial in $S \in O^\sim(1)$ and $\deg f \in O^\sim(1)$. Thus, the total running time of the algorithm is in

$$O^\sim\left(t_f + n^2 + n\frac{N}{N_S}\right).$$

3) $G = \mathbb{F}_{p^k}$, p prime

An element $(m, f) \in \mathbb{N} \times \mathbb{F}_p[X]'$ is S–smooth if and only if m and f are S–smooth. The smoothness of m can be tested in time in $O^\sim(L_p(2/3, c))$ as mentioned in Example 1). The smoothness of f can again be checked by distinct degree factorisation in time $O^\sim(1)$. Thus, the total running time of the algorithm is in

$$O^\sim\left(t_f + n^2 + nL_p(2/3, c)\frac{N}{N_S}\right).$$

4) Ideal class groups of imaginary quadratic number fields

As the smoothness test and the decomposition into primes are reduced to the case of natural integers, the analysis of Example 1) shows that the running time is in

$$O^\sim\left(t_f + n^2 + nL_n(2/3, c)\frac{N}{N_S}\right).$$

5) Jacobians of imaginary quadratic hyperelliptic curves over finite fields

Now the smoothness test and the decomposition are reduced to the case of monic polynomials, and the analysis of Example 2) carries over and shows that the running time is in

$$O^{\sim}\left(t_f + n^2 + n\frac{N}{N_S}\right).$$

□

7.3.3 Subexponentiality

Assume again that the bound S can be chosen such that

$$n \in O(L_{N'}(\rho + o(1)))$$

and

$$\frac{N}{N_S} \in O(L_{N'}(\sigma + o(1)))$$

for some constants ρ, $\sigma > 0$.

Notice that N can be factored in expected time in $O(L_N(1 + o(1)))$ by the algorithm presented in [LP92] and that $L_N(1) \in O\left(L_{N'}(1 + o(1))\right)$ since $N \in O^{\sim}(N')$. Introducing an exponent τ such that $t_s \in O^{\sim}(n^{\tau})$, we obtain that the running time of the algorithm is in

$$O\left(L_{N'}\left(\max(1, 2\rho, (1+\tau)\rho + \sigma) + o(1)\right)\right).$$

In fact, the constants for all examples presented below are worse than 1 anyway, so that the need for factoring N has no influence on our running time bounds.

Examples.

1) $G = \mathbb{F}_p^{\times}$, p prime

For $S = \lceil \log(L_N(\rho)) \rceil$ we have $\sigma = \frac{1}{2\rho}$ by Theorem 5.3. Moreover from $n \in O(L_N(\rho))$ we deduce $L_n(2/3, c) \in L_N(o(1))$, cf. Section 5.3. The running time of the algorithm is thus in

$$O\left(L_N\left(\max\left(2\rho, \rho + \frac{1}{2\rho}, 1\right) + o(1)\right)\right)$$

for any $\rho > 0$; the optimal choice $\rho = 1/\sqrt{2}$ yields a running time in

$$O(L_N(\sqrt{2} + o(1))).$$

This is precisely the complexity of the fastest previously known algorithm described in [Pom87].

2) $G = \mathbb{F}_{2^k}^\times$

For $S = \lceil \log(L_N(\rho)) \rceil$ we have $\sigma = \frac{1}{2\rho}$ by Theorem 5.5. Thus, the running time of the algorithm is in

$$O\left(L_N\left(\max\left(2\rho, \rho + \frac{1}{2\rho}, 1\right) + o(1)\right)\right)$$

for any $\rho > 0$. The optimal choice $\rho = 1/\sqrt{2}$ again yields a running time in

$$O(L_N(\sqrt{2} + o(1))),$$

which corresponds to the fastest previously known algorithms described in [Pom87] and [BP98].

3) $G = \mathbb{F}_{p^k}$, p prime

Notice first that in the polynomial representation we have chosen, it is impossible to obtain a subexponential running time for fixed $k \geq 2$ and $p \to \infty$. If we let $S = 0$, then only the constants have a chance of being smooth, and $\frac{N}{N_S} \geq \frac{p^k - 1}{p - 1} \geq p^{k-1}$ is exponential in N. If $S \geq 1$, then all p monic linear polynomials are contained in the factor base, which is thus of exponential size. Hence, we must restrict our attention to instances in which p is sufficiently small compared to k, which is a situation similar to that of Section 6.1.3.

With the notation of Theorem 5.5, we have the estimate

$$\frac{N}{N_S} \in O\left(pL_N\left(\frac{1}{2\rho} + o(1)\right)\right)$$

for $S = \lceil \log_p(L_N(\rho)) \rceil$, which introduces the additional factor p to the subexponential function. Moreover, since we have to round up S, it need not be true any more that $n \in O(L_N(\rho))$. In fact,

$$\begin{aligned}
n &\leq \sum_{i=0}^{S} \left|\{f \in \mathbb{F}_p'[X] : \deg f = i\}\right| \\
&\qquad \left|\{m \in \{1, \ldots, p-1\} : \operatorname{ld} m \leq S - i\}\right| \\
&= \sum_{i=0}^{S} p^i \min\left\{p - 1, 2^{S-i}\right\} \\
&\leq \sum_{i=0}^{S-1} p^{i+1} + p^S \\
&\in O\left(p^S\right) \\
&\subseteq O(pL_N(\rho)).
\end{aligned}$$

For a first special result we consider the case $p \in O^{\sim}(1)$, which implies $n \in O^{\sim}(L_N(\rho))$ and $L_p(2/3, c) \in L_N(o(1))$. So the running time analysis of Example 2) carries over without modification.

More generally, we must ensure that p is subexponential in $\log N \in k \log p(1+o(1))$. Following the ideas of Section 6.3.5, we consider the case $k \geq \vartheta \log p$ for some positive constant ϑ, in which $p \leq L_N\left(\frac{1}{\sqrt{\vartheta}}\right)$. Then $n \in O\left(L_N\left(\rho + \frac{1}{\sqrt{\vartheta}}\right)\right)$ and $\frac{N}{N_S} \in O\left(L_N\left(\frac{1}{2\rho} + \frac{1}{\sqrt{\vartheta}} + o(1)\right)\right)$ for the same value of S as above, $L_p(2/3, c) \in L_N(o(1))$, and the total running time is in

$$O\left(L_N\left(\max\left\{2\rho + \frac{2}{\sqrt{\vartheta}}, \rho + \frac{1}{2\rho} + \frac{2}{\sqrt{\vartheta}}, 1\right\} + o(1)\right)\right).$$

The optimal choice for ρ is $\frac{\sqrt{2}}{2}$, which yields a running time of

$$O\left(L_N\left(\sqrt{2} + \frac{2}{\sqrt{\vartheta}} + o(1)\right)\right).$$

Asymptotically for $\vartheta \to \infty$ (e.g., for p fixed), we recover the running time of Example 2).

In [AD93], Adleman and DeMarrais describe an algorithm with conjectured subexponential running time for $p > k$. They represent the field as the ring of integers of a number field modulo a prime ideal. See also [Sem95]. It is an interesting open question whether these algorithms can be made rigorous.

4) Ideal class groups of imaginary quadratic number fields

Letting $S = \lceil \log L_N(\rho) \rceil$, we have $\sigma = \frac{1}{2\rho}$ under the generalised Riemann hypothesis by Theorem 5.6. Repeating the analysis of Example 1) proves the following result.

Theorem 7.5 *Assuming the generalised Riemann hypothesis, Algorithm 7.1 computes discrete logarithms in the ideal class groups of imaginary quadratic number fields of discriminant D in expected time in*

$$O\left(L_N\left(\sqrt{2} + o(1)\right)\right) = O\left(L_{|D|}\left(1 + o(1)\right)\right).$$

5) Jacobians of imaginary quadratic hyperelliptic curves over finite fields

We assume that $g \geq \vartheta \log q$ as in Example 2) of Section 6.3.5. Letting $S = \lceil \log_q L_{q^g}(\rho) \rceil$, a similar analysis shows that the complexity of the algorithm is in

$$O\left(L_{q^g}\left(\max\left\{2\rho + \frac{2}{\sqrt{\vartheta}}, \rho + \frac{1}{2\rho} + \frac{1}{\sqrt{\vartheta}}, 1\right\} + o(1)\right)\right),$$

which is minimised for

$$\begin{aligned}\rho &= \min\{\overline{\rho}, \rho^*(\vartheta)\} = \min\left\{\frac{\sqrt{2}}{2}, \sqrt{\frac{1}{2}+\frac{1}{4\vartheta}} - \frac{1}{2\sqrt{\vartheta}}\right\} \\ &= \sqrt{\frac{1}{2}+\frac{1}{4\vartheta}} - \sqrt{\frac{1}{4\vartheta}}.\end{aligned}$$

This proves the following result.

Theorem 7.6 *Algorithm 7.1 computes discrete logarithms in Jacobians of imaginary quadratic hyperelliptic curves of genus g over finite fields $\mathbb{F}_q$ satisfying $g \geq \vartheta \log q$ for some positive constant ϑ in expected time in*

$$O\left(L_{q^g}\left(\sqrt{2}\left(\sqrt{1+\frac{1}{2\vartheta}}+\sqrt{\frac{1}{2\vartheta}}\right)+o(1)\right)\right).$$

If $g/\log q$ tends to infinity for the instances under consideration, e.g. if q is constant and $g \to \infty$, then the complexity is

$$O\left(L_{q^g}\left(\sqrt{2}+o(1)\right)\right).$$

□

7.4 Cyclic subgroups

In this section, we discuss a few approaches how Algorithm 7.1 can be used to compute discrete logarithms in cyclic subgroups. Let G be an arithmetical formation and $H = \langle g_1 \rangle$ a cyclic subgroup of G of known order N. For another element $g_2 \in H$, we wish to determine $\log_{g_1} g_2$.

7.4.1 Perturbing with elements of the complement

The simplest situation arises when $\gcd\left(|H|, \frac{|G|}{|H|}\right) = 1$; then H admits a complement H' in G, i.e., $G = H \times H'$. Assume that it is possible to select independently elements h_j of H' according to a uniform distribution in time polynomial in $\log N$. This is for instance the case if we can select random elements in G in polynomial time, because multiplying a uniformly distributed element of G by $\frac{|G|}{|H|}$ yields a uniformly distributed element of H'. Another favourable situation is the case where we know a basis of H'. (In this context,

we understand by a basis of H' a set $\{b_1, \ldots, b_r\}$ such that H' is equal to the direct sum $\langle b_1 \rangle \times \cdots \times \langle b_r \rangle$. The cardinality r of a basis is not an invariant of H', but it is bounded above by $\operatorname{ld} |H'|$.)

Then $\alpha_j g_1 + \beta_j g_2 + h_j$ is distributed uniformly and independently of β over G, so that the algorithm may be carried out with these group elements instead of $\alpha_j g_1 + \beta_j g_2$. If it is successful, then

$$\left(\sum_{j=1}^{2kn} \alpha_j \gamma_j \right) g_1 + \left(\sum_{j=1}^{2kn} \beta_j \gamma_j \right) g_2 = - \sum_{j=1}^{2kn} \gamma_j h_j \in H \cap H' = \{0\},$$

so that

$$\log_{g_1} g_2 = \left(\sum_{j=1}^{2kn} \beta_j \gamma_j \right)^{-1} \left(\sum_{j=1}^{2kn} \alpha_j \gamma_j \right)$$

as before. Also, the running time analysis remains unchanged.

While this situation seems to be very special, it is typical for cryptographic applications in which H is supposed to have large prime order and the cofactor $\frac{|G|}{|H|}$ is small, so that $|H|$ and $\frac{|G|}{|H|}$ are automatically coprime. Moreover, for $\frac{|G|}{|H|}$ polynomial in $\log N$, the structure and, in particular, a basis of $H' \simeq G/H$ can be determined in polynomial time (see, for instance, [Coh93]), and the assumptions of this subsection are satisfied.

7.4.2 Using a basis for G

Assume that a basis $\{b_1, \ldots, b_r\}$ of G along with the orders $e_1, \ldots, e_r$ of its elements are known. Then the discrete logarithm problem can be solved in two steps. Instead of directly writing g_2 as a multiple of g_1 we first express g_1 as a linear combination of the basis elements and then proceed in the same way for g_2. The discrete logarithm can be computed by a few operations modulo the e_i.

In order to write g_1 in terms of the b_i, a slight variation of the algorithm allows to use the smoothness properties. For given $j \leq kn$, pick random elements α_{ij} and β_j until $\sum \alpha_{ij} b_i + \beta_j g_1$ is S–smooth and write this element as $\sum a_{ij} p_i$. Similarly, for $j > kn$ pick random elements until $\sum \alpha_{ij} b_i + \beta_j g_1 - \mathfrak{P}_m$ is S–smooth. Here again, the elements of G which are tested for smoothness are distributed uniformly and independently of β, so that the same analysis as in Section 7.3.1 can be carried out. Hence with high probability a non-zero vector of the kernel is obtained and g_1 is expressed as a linear combination $g_1 = \sum \gamma_i b_i$. The same process yields $g_2 = \sum \delta_i b_i$. Now try to solve the system of modular equations $\delta_i \equiv l \gamma_i \pmod{e_i}$. If this is possible, then l is the correct discrete logarithm of g_2 with respect to g_1. Otherwise, g_2 does not lie in the cyclic subgroup generated by g_1. In this case, which does not occur in the cryptographic setting, the

original algorithm would run forever without giving proof of the non-existence of the discrete logarithm. Thus, the ability to detect this case is an additional advantage of the modified algorithm.

7.5 Implementation

Basically, the same remarks as in Section 6.6 apply to the algorithm of this chapter. In practice, it is sufficient to collect some more than $n+1$ relations, and one can profit from automorphisms to reduce the size of the factor base.

The computation of random linear combinations of g_1 and g_2 can also be replaced by a random walk. After precomputing a few linear combinations, one may randomly select one of them in each step and add it to the previous element. In practice, this results in a polynomial speed-up.

A first version of the algorithm has been implemented by Gaudry. The largest example he was able to attack was the Jacobian of a curve of genus 6 over $\mathbb{F}_{2^{23}}$ of size about 2^{138}; it had an automorphism of order 23 ([Gau00]). This size is so close to values deemed secure for elliptic curve cryptosystems that hyperelliptic curves of genus 6 or more should definitely be avoided. Since further implementational progress can be expected, even curves of smaller genus might be insecure. A different analysis carried out in [Gau00] for fixed genus and the field size tending to infinity shows that Algorithm 7.1 is asymptotically faster than the square root attacks of Sections 1.2.3 and 1.2.4 as soon as the genus exceeds 4. In the light of the results of Chapter 4, this means that elliptic curve cryptosystems are likely to remain the state of the art and to not be replaced by hyperelliptic cryptosystems.

Bibliography

[AD93] Leonard M. Adleman and Jonathan DeMarrais. A subexponential algorithm for discrete logarithms over all finite fields. *Mathematics of Computation*, 61(203):1–15, 1993.

[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 *[AH94]*, pages 28–40, 1994.

[AH94] Leonard M. Adleman and Ming-Deh Huang, editors. *Algorithmic Number Theory*, volume 877 of *Lecture Notes in Computer Science*, Berlin, 1994. Springer-Verlag.

[Art24a] E. Artin. Quadratische Körper im Gebiete der höheren Kongruenzen I. *Mathematische Zeitschrift*, 19:153–206, 1924.

[Art24b] E. Artin. Quadratische Körper im Gebiete der höheren Kongruenzen II. *Mathematische Zeitschrift*, 19:207–246, 1924.

[Art67] Emil Artin. *Algebraic Numbers and Algebraic Functions.* Notes on Mathematics and its Applications. Gordon and Breach Science Publishers, New York, 1967.

[AS27] Emil Artin and Otto Schreier. Eine Kennzeichnung der reell abgeschlossenen Körper. *Abhandlungen aus dem mathematischen Seminar der hamburgischen Universität*, 5:225–231, 1927.

[Bac90] Eric Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990.

[Ber68] Elwyn R. Berlekamp. *Algebraic Coding Theory.* McGraw-Hill Series in Systems Science. McGraw-Hill, New York, 1968.

[Bom74] Enrico Bombieri. Counting points on curves over finite fields. In *[DE74]*, pages 234–241, 1974.

[BP98] Renet Lovorn Bender and Carl Pomerance. Rigorous discrete logarithm computations in finite fields via smooth polynomials. In *[BT98]*, pages 221–232, 1998.

[BT98] 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*. American Mathematical Society, 1998.

[Buc90] Johannes Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. In *[Gol90]*, pages 27–41, 1990.

[Buh98] J. P. Buhler, editor. *Algorithmic Number Theory — ANTS-III*, volume 1423 of *Lecture Notes in Computer Science*, Berlin, 1998. Springer-Verlag.

[BW88] Johannes Buchmann and H. C. Williams. A key-exchange system based on imaginary quadratic fields. *Journal of Cryptology*, 1:107–118, 1988.

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

[Car87] Mireille Car. Théorèmes de densité dans $\mathbb{F}_q[x]$. *Acta Arithmetica*, 68:145–165, 1987.

[Che51] Claude Chevalley. *Introduction to the Theory of Algebraic Functions of one Variable*. American Mathematical Society, 1951.

[Coh93] Henri Cohen. *A Course in Computational Algebraic Number Theory*. Graduate Texts in Mathematics. Springer-Verlag, New York, 1993.

[Coh96] Henri Cohen, editor. *Algorithmic Number Theory — ANTS-II*, volume 1122 of *Lecture Notes in Computer Science*, Berlin, 1996. Springer-Verlag.

[DE74] A. Dold and B. Eckmann, editors. *Séminaire Bourbaki vol. 1972/73 Exposés 418-435*, volume 383 of *Lecture Notes in Mathematics*, Berlin, 1974. Springer-Verlag.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–655, November 1976.

[Dül91] Stephan Düllmann. *Ein Algorithmus zur Bestimmung der Klassengruppe positiv definiter binärer quadratischer Formen*. PhD thesis, Universität des Saarlandes, Saarbrücken, 1991.

[DW82] R. Dedekind and H. Weber. Theorie der algebraischen Functionen einer Veränderlichen. *Journal für die reine und angewandte Mathematik*, 92:181–290, 1882.

[Eic63] Martin Eichler. *Einführung in die Theorie der algebraischen Zahlen und Funktionen*, volume 27 of *Mathematische Reihen*. Birkhäuser-Verlag, Basel, 1963.

[EK97] Wayne Eberly and Erich Kaltofen. On randomized Lanczos algorithms. In *[Küc97]*, pages 176–183, 1997.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472, July 1985.

[Eng99] Andreas Enge. *Elliptic Curves and Their Applications to Cryptography — An Introduction*. Kluwer Academic Publishers, 1999.

[FP99] Ralf Flassenberg and Sachar Paulus. Sieving in function fields. *Experimental Mathematics*, 8(4):339–349, 1999.

[FR94] Gerhard Frey and Hans-Georg Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Mathematics of Computation*, 62(206):865–874, April 1994.

[Ful69] William Fulton. *Algebraic Curves*. Mathematics Lecture Note Series. The Benjamin/Cummings Publishing Company, Reading (Massachusetts), 1969.

[Fum97] Walter Fumy, editor. *Advances in Cryptology — EUROCRYPT '97*, volume 1233 of *Lecture Notes in Computer Science*, Berlin, 1997. Springer-Verlag.

[Gau01] Carl Friedrich Gauß. *Disquisitiones Arithmeticae*. Gerh. Fleischer Jun., Leipzig, 1801.

[Gau00] Pierrick Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In *[Pre00]*, pages 19–34, 2000.

[GG99] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999.

[Gol90] Catherine Goldstein, editor. *Séminaire de Théorie des Nombres, Paris 1988–1989*, Progress in Mathematics, Boston, 1990. Birkhäuser.

[Gra73] De la Grange. Recherches d'arithmétique. *Nouveaux Mémoires de l'Académie Royale des Sciences et Belles-Lettres*, pages 265–312, 1773.

[Hal76] Marshall Hall Jr. *The Theory of Groups*. Chelsea Publishing Company, New York, 2nd edition, 1976.

[Har01] Robert Harley. Advanced algorithms for algebra and number theory. PhD thesis in preparation, title susceptible to change, 2001.

[Has33] Helmut Hasse. Beweis des Analogons der Riemannschen Vermutung für die Artinschen und F. K. Schmidtschen Kongruenzzetafunktionen in gewissen elliptischen Fällen. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Phyikalische Klasse*, pages 253–262, 1933.

[Has34] Helmut Hasse. Abstrakte Begründung der komplexen Multiplikation und Riemannsche Vermutung in Funktionenkörpern. *Abhandlungen aus dem mathematischen Seminar der hamburgischen Universität*, 10:325–348, 1934.

[Has35] Helmut Hasse. Theorie der relativ-zyklischen algebraischen Funktionenkörper, insbesondere bei endlichem Konstantenkörper. *Journal für die reine und angewandte Mathematik*, 172:37–54, 1935.

[Hen85] Douglas Hensley. The number of positive integers $\leq x$ and free of prime factors $> y$. *Journal of Number Theory*, 21:286–298, 1985.

[Heß99] Florian Heß. *Zur Divisorenklassengruppenberechnung in globalen Funktionenkörpern*. PhD thesis, Technische Universität Berlin, 1999.

[HI98] Ming-Deh Huang and Doug Ierardi. Counting points on curves over finite fields. *Journal of Symbolic Computation*, 25:1–21, 1998.

[Hil86] Adolf Hildebrand. On the number of positive integers $\leq x$ and free of prime factors $> y$. *Journal of Number Theory*, 22:289–307, 1986.

[HM89] James L. Hafner and Kevin S. McCurley. A rigorous subexponential algorithm for computation of class groups. *Journal of the American Mathematical Society*, 2(4):837–850, 1989.

[HM91] James L. Hafner and Kevin S. McCurley. Asymptotically fast triangularization of matrices over rings. *SIAM Journal on Computing*, 20(6):1068–1083, 1991.

[HT86] Adolf Hildebrand and Gérald Tenenbaum. On integers free of large prime factors. *Transactions of the American Mathematical Society*, 296(1):265–290, 1986.

[IMW91] Karl-Heinz Indlekofer, Eugenius Manstavicius, and Richard Warlimont. On a certain class of infinite products with an application to arithmetical semigroups. *Archiv der Mathematik*, 56:446–453, 1991.

[INR00] Biggest public-key crypto crack ever — INRIA leads worldwide internet-distributed calculation. INRIA Press Release, April 2000. Available at http://www.inria.fr/presse/pre67.en.html.

[IZ00] Hideki Imai and Yuliang Zheng, editors. *Public Key Cryptography — 3rd International Workshop on Practice and Theory in Public Key Cryptosystems PKC 2000*, volume 1751 of *Lecture Notes in Computer Science*, Berlin, 2000. Springer-Verlag.

[Jac00] Michael J. Jacobson Jr. Computing discrete logarithms in quadratic orders. *Journal of Cryptology*, 13:473–492, 2000.

[JNNW87] David S. Johnson, Takao Nishizeki, Akihiro Nozaki, and Herbert S. Wolf, editors. *Discrete Algorithms and Complexity, Proceedings of the Japan–US Joint Seminar, June 4–6, 1986, Kyoto, Japan*, volume 15 of *Perspectives in Computing*, Orlando, 1987. Academic Press.

[Jun93] Dieter Jungnickel. *Finite Fields — Structure and Arithmetics*. BI Wissenschaftsverlag, Mannheim, 1993.

[KK88] Arnold Knopfmacher and John Knopfmacher. The exact length of the Euclidian algorithm. *Mathematika*, 35:297–304, 1988.

[Kno75] John Knopfmacher. *Abstract Analytic Number Theory*, volume 12 of *North-Holland Mathematical Library*. North-Holland Publishing Company, Amsterdam, 1975.

[Kno79] John Knopfmacher. *Analytic Arithmetic of Algebraic Function Fields*, volume 50 of *Lecture notes in pure and applied mathematics*. Marcel Dekker, New York, 1979.

[Knu81] Donald Ervin Knuth. *The Art of Computer Programming*, volume 2 - Seminumerical Algorithms. Addison-Wesley, Reading (Massachusetts), 2nd edition, 1981.

[Kob87] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209, January 1987.

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

[Kob98] Neal Koblitz. *Algebraic Aspects of Cryptography*, volume 3 of *Algorithms and Computations in Mathematics*. Springer-Verlag, Berlin, 1998.

[Kri97] Uwe Krieger. signature.c — Anwendung hyperelliptischer Kurven in der Kryptographie. Master's thesis, Universität Essen, Deutschland, 1997.

[KS91] Erich Kaltofen and B. David Saunders. On Wiedemann's method of solving sparse linear systems. In *[MMR91]*, pages 29–38, 1991.

[Küc97] Wolfgang W. Küchlin, editor. *ISSAC 97 — Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation*. ACM Press, 1997.

[Len88] A. K. Lenstra. Fast and rigorous factorization under the generalized Riemann hypothesis. *Indagationes Mathematicae*, 50:443–454, 1988.

[Lew71] Donald Lewis, editor. *Proceedings of Symposia in Pure Mathematics*, volume 10, Providence (Rhode Island), 1971. American Mathematical Society.

[LMO79] J. C. Lagarias, H. L. Montgomery, and A. M. Odlyzko. A bound for the least prime ideal in the Chebotarev density theorem. *Inventiones mathematicae*, 54:271–296, 1979.

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite Fields*, volume 20 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, Cambridge, 2nd edition, 1997.

[LO91] B. A. LaMacchia and A. M. Odlyzko. Solving large sparse linear systems over finite fields. In *[MV91]*, pages 109–133, 1991.

[LP92] H. W. Lenstra Jr. and Carl Pomerance. A rigorous time bound for factoring integers. *Journal of the American Mathematical Society*, 5(3):483–516, 1992.

[LPP93] H. W. Lenstra Jr., J. Pila, and Carl Pomerance. A hyperelliptic smoothness test. I. *Philosophical Transactions of the Royal Society of London, Series A*, 345:397–408, 1993.

[LT82] H. W. Lenstra Jr. and R. Tijdeman, editors. *Computational Methods in Number Theory. Part II*, Amsterdam, 1982. Mathematisch Centrum.

[LV00] Arjen K. Lenstra and Eric R. Verheul. Selecting cryptographic key sizes (extended abstract). In *[IZ00]*, pages 446–465, 2000.

[Ma87] Keju Ma. Analysis of polynomial gcd computations over finite fields. Master's thesis, University of Toronto, Toronto, 1987.

[Man92a] E. Manstavičius. Remarks on the semigroup elements free of large prime factors. *Lithuanian Mathematical Journal*, 32(4):400–409, 1992.

[Man92b] E. Manstavičius. Semigroup elements free of large prime factors. In *[SM92]*, pages 135–153, 1992.

[McC89] Kevin S. McCurley. Cryptographic key distribution and computation in class groups. In *[Mol89]*, pages 459–479, 1989.

[McN99] David McNett. US government's encryption standard broken in less than a day. Formal Press Release, January 1999. Available at http://-www.distributed.net/des/release-desiii.txt.

[MG90] Keju Ma and Joachim von zur Gathen. Analysis of Euclidian algorithms for polynomials over finite fields. *Journal of Symbolic Computation*, 9:429–455, 1990.

[Mil86] Victor S. Miller. Use of elliptic curves in cryptography. In *[Wil86]*, pages 417–426, 1986.

[MMR91] H. F. Mattson, T. Mora, and T. R. N. Rao, editors. *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 539 of *Lecture Notes in Computer Science*, Berlin, 1991. Springer-Verlag.

[Mol89] Richard A. Mollin, editor. *Number Theory and Applications*, volume 265 of *NATO ASI Series C: Mathematical and Physical Sciences*, Dordrecht, 1989. Kluwer Academic Publishers.

[MOV93] Alfred J. Menezes, Tatsuaki Okamoto, and Scott A. Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Transactions on Information Theory*, 39(5):1639–1646, September 1993.

[MOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, Boca Raton, 1997.

[MST99] Volker Müller, Andreas Stein, and Christoph Thiel. Computing discrete logarithms in real quadratic congruence function fields of large genus. *Mathematics of Computation*, 68(226):807–822, 1999.

[MV91] A. J. Menezes and S. A. Vanstone, editors. *Advances in Cryptology — CRYPTO '90*, volume 537 of *Lecture Notes in Computer Science*, Berlin, 1991. Springer-Verlag.

[MVZ98] Volker Müller, Scott Vanstone, and Robert Zuccherato. Discrete logarithm based cryptosystems in quadratic function fields of characteristic 2. *Designs, Codes and Cryptography*, 14(2):159–178, May 1998.

[MW99] Ueli M. Maurer and Stefan Wolf. The relationship between breaking the Diffie–Hellman protocol and computing discrete logarithms. *SIAM Journal on Computing*, 28(5):1689–1721, 1999.

[MWZ98] Alfred J. Menezes, Yi-Hong Wu, and Robert J. Zuccherato. An elementary introduction to hyperelliptic curves. In *[Kob98]*, pages 155–178. Springer-Verlag, 1998.

[Nec94] V. I. Nechaev. Complexity of a determinate algorithm for the discrete logarithm. *Mathematical Notes*, 55(2):165–172, 1994.

[NIS97] NIST. Announcing request for candidate algorithm nominations for the advanced encryption standard (AES). National Institute of Standards and Technology, September 1997. Available at http://csrc.nist.gov/encryption/aes/-pre-round1/aes_9709.htm.

[OW99] P. C. van Oorschot and M. J. Wiener. Parallel collision search with cryptanalytic applications. *Journal of Cryptology*, 12(1):1–28, 1999.

[PGF98] Daniel Panario, Xavier Gourdon, and Philippe Flajolet. An analytic approach to smooth polynomials over finite fields. In *[Buh98]*, pages 226–236, 1998.

[PH78] Stephen C. Pohlig and Martin E. Hellman. An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance. *IEEE Transactions on Information Theory*, 24(1):106–110, January 1978.

[Pil90] J. Pila. Frobenius maps of Abelian varieties and finding roots of unity in finite fields. *Mathematics of Computation*, 55(192):745–763, 1990.

[Pol74] J. M. Pollard. Theorems on factorization and primality testing. *Proc. Camb. Phil. Soc.*, 76:521–528, 1974.

[Pol78] J. M. Pollard. Monte Carlo methods for index computation (mod p). *Mathematics of Computation*, 32(143):918–924, July 1978.

[Pom87] Carl Pomerance. Fast, rigorous factorization and discrete logarithm algorithms. In *[JNNW87]*, pages 119–143, 1987.

[Poo96] Bjorn Poonen. Computational aspects of curves of genus at least 2. In *[Coh96]*, pages 283–306, 1996.

[PR99] Sachar Paulus and Hans-Georg Rück. Real and imaginary quadratic representations of hyperelliptic function fields. *Mathematics of Computation*, 68(227):1233–1241, 1999.

[Pre00] Bart Preneel, editor. *Advances in Cryptology — EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, Berlin, 2000. Springer-Verlag.

[PS98] Sachar Paulus and Andreas Stein. Comparing real and imaginary arithmetics for divisor class groups of hyperelliptic curves. In *[Buh98]*, pages 576–591, 1998.

[RS62] J. Barkley Rosser and Lowell Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6:64–94, 1962.

[SA98] Takakazu Satoh and Kiyomichi Araki. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. *Commentarii Mathematici Universitatis Sancti Pauli*, 47(1):81–92, 1998. Errata in vol. 48 (2):211–213, 1999.

[Sch31] Friedrich Karl Schmidt. Analytische Zahlentheorie in Körpern der Charakteristik p. *Mathematische Zeitschrift*, 33:1–32, 1931.

[Sch82] R. J. Schoof. Quadratic fields and factorization. In *[LT82]*, pages 235–286, 1982.

[Sem95] I. A. Semaev. Computation of discrete logarithms in an arbitrary finite field. *Discrete Mathematics and Applications*, 5(2):107–116, 1995.

[Sem98] I. A. Semaev. Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p. *Mathematics of Computation*, 67(221):353–356, 1998.

[Sey87] Martin Seysen. A probabilistic factorization algorithm with quadratic forms of negative discriminant. *Mathematics of Computation*, 48(178):757–780, 1987.

[Sha71] D. Shanks. Class number, a theory of factorization and genera. In *[Lew71]*, pages 415–440, 1971.

[Sho97] Victor Shoup. Lower bounds for discrete logarithms and related problems. In *[Fum97]*, pages 256–266, 1997.

[Sie36] Carl Ludwig Siegel. Über die Classenzahl quadratischer Zahlkörper. *Acta Arithmetica*, 1:83–86, 1936.

[SM92] F. Schweiger and E. Manstavičius, editors. *New Trends in Probability and Statistic*, 1992.

[Sma99] Nigel P. Smart. The discrete logarithm problem on elliptic curves of trace one. *Journal of Cryptology*, 12(3):193–196, 1999.

[Spa94] Anne-Monika Spallek. *Kurven vom Geschlecht 2 und ihre Anwendung in Public-Key-Kryptosystemen*. PhD thesis, Universität Gesamthochschule Essen, 1994.

[SSW96] R. Scheidler, A. Stein, and H. C. Williams. Key-exchange in real quadratic congruence function fields. *Designs, Codes and Cryptography*, 7(1/2):153–174, 1996.

[Ste96] Andreas Stein. *Algorithmen in reell-quadratischen Kongruenzfunktionenkörpern*. PhD thesis, Universität des Saarlandes, Saarbrücken, 1996.

[Sti93] Henning Stichtenoth. *Algebraic Function Fields and Codes*. Springer-Verlag, Berlin, 1993.

[Str76] Volker Strassen. Einige Resultate über Berechnungskomplexität. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 78:1–8, 1976.

[Ten90] Gérald Tenenbaum. *Introduction à la théorie analytique et probabiliste des nombres*. Institut Elie Cartan, Nancy, 1990.

[Ten95] Gérald Tenenbaum. *Introduction to Analytic and Probabilistic Number Theory*, volume 46 of *Cambridge studies in advanced mathematics*. Cambridge University Press, Cambridge, 1995.

[Tes01] Edlyn Teske. On random walks for Pollard's rho method. *Mathematics of Computation*, 70(234):809–825, 2001.

[Wei48] André Weil. Sur les courbes algébriques et les variétés qui s'en déduisent. In *[Wei71]*. Hermann, Paris, 1948.

[Wei71] André Weil. *Courbes algébriques et variétés abéliennes*. Hermann, Paris, 1971.

[Wil86] Hugh C. Williams, editor. *Advances in Cryptology — CRYPTO '85*, volume 218 of *Lecture Notes in Computer Science*, Berlin, 1986. Springer-Verlag.

[Zuc97a] Robert Zuccherato. *New Applications of Elliptic Curves and Function Fields in Cryptography*. PhD thesis, University of Waterloo, 1997.

[Zuc97b] Robert J. Zuccherato. The continued fraction algorithm and regulator for quadratic function fields of characteristic 2. *Journal of Algebra*, 190:563–587, 1997.

Index

λ-algorithm, 9–12
μ, 91
$\pi_{\mathcal{O}}$, 28
$\psi(N, 2^S)$, 93
ρ-algorithm, 9–12
$\mathrm{A}^{\#}$, 93
$O^{\sim}(f)$, 85
$\mathrm{Div}(F/K)$, 24
$\mathrm{Div}^0(F/K)$, 25
$\mathrm{Div}_\infty(\mathcal{O})$, 28
$\mathrm{Div}^0_\infty(\mathcal{O})$, 28
$\mathrm{div}(a, b)$, 54
$\mathrm{div}(z)$, 25
$\mathrm{div}(\mathfrak{a})$, 27
$\mathrm{div}_0(z)$, 25
$\mathrm{div}_\infty(z)$, 25
$\mathfrak{H}(\mathcal{O})$, 27
$\mathcal{I}(\mathcal{O})$, 27
$J(F/K)$, 25
$\mathcal{L}(D)$, 25
$l(D)$, 25
N, 47
$o(1)$, 91
$\mathrm{Prin}(F/K)$, 25
$\mathrm{Prin}(\mathcal{O})$, 27
Tr, 47

absolute irreducibility, 19, 22, 33, 35
additive arithmetical semigroup, 84
affine
    curve, 18
    plane, 18
almost prime, 9
arithmetical formation, 84, 104, 124, 133
arithmetical semigroup, 84, 92
Artin–Schreier extension, 33, 38, 39, 41
asymmetric cryptography, 5
Axiom $\mathrm{A}^{\#}$, 93

baby step, 9
basis of group, 133, 134
binary quadratic form, 57
binomial distribution, 69, 114
birational equivalence, 19
bisection, 7
brute force, 7

canonical divisor, 26, 56
Cantor reduction, 60, 61, 77
character, 105
    principal, 105
Chinese Remainder Theorem, 8, 67, 126
class group, 24, 25, 27, 46, 87, 95, 105, 116, 123, 129, 132
class number, 26, 28
class number formula, 107
Clifford's theorem, 56
closed point, 20, 23
closure, projective, 19
collision, 9, 12
column echelon form, 109
complement, 133
composition, 57, 58, 76
conjugation, 47, 48, 51
constant field, 22
coordinate ring, 18, 20

cryptography
  asymmetric, 5
  public key, 5
  symmetric, 5, 8
cube, 111
curve
  affine, 18
  elliptic, *see* elliptic curve
  hyperelliptic, *see* hyperelliptic curve
  non-singular, 18, 19
  projective, 19
  real quadratic, *see* real quadratic curve
  smooth, 18, 19
cyclic extension, 31

decomposition law, 23, 24
Dedekind ring, 17, 23, 24, 27
degree
  of closed point, 20, 23
  of divisor, 25
  of map, 6
  of prime divisor, 23, 89
dehomogenisation, 19, 28
determinant, 109
Diffie–Hellman
  key exchange, 6
  problem, 7
digital signature, 5, 6, 103
discrete logarithm, 5, 7–12
discrete valuation, 18–20, 24
discrete valuation ring, 20, 21
discriminant, 105, 107, 116, 121
distinguished point, 11
distributed algorithm, *see* parallel algorithm
distribution
  binomial, *see* binomial distribution
  uniform, *see* uniform distribution
divisor, 24
  canonical, 26
  effective, 26
  of ideal, 27
  of poles, *see* pole divisor
  of zeroes, *see* zero divisor
  positive, 26
  prime, *see* prime divisor
  principal, *see* principal divisor
  reduced, *see* reduced divisor
  semireduced, *see* semireduced divisor
divisor class group, 25, 27
divisor class number, 26
double and add, 114

effective divisor, 26
elementary divisor, 29, 104, 109, 113, 118–120, 127
ElGamal signature, 6
elliptic curve, 26, 38, 107, 135
Euclidean algorithm, 7, 52, 57–59, 61, 62, 67–74, 78, 126
extended Euclidean algorithm, *see* Euclidean algorithm
extension
  cyclic, 31
  Galois, *see* Galois extension
  of prime divisor, 23

factor base, 14, 104, 110, 117, 119, 124
field
  of constants, 22
  of definition, 22
  of numbers, *see* number field
  of residue classes, *see* residue class field
  perfect, 19
finite point, 18
flea, 11
formation, arithmetical, 84, 104, 124, 133
Frobenius automorphism, 20
function

polynomial, 18
rational, 18
subexponential, 13
function field, 18, 20
hyperelliptic, 38
rational, 20, 22, 26, *see* rational function field
function field sieve, 13
fundamental unit, 29

Galois extension, 19, 23, 31–33, 47
Gauß reduction, 59, 77
generalised Riemann hypothesis, *see* Riemann hypothesis
generating function, 92, 96
genus, 26, 33
giant step, 9
greatest common divisor of divisors, 49
group invariants, 104
group structure, 104, 109, 119, 123

Hadamard's bound, 114
half-extended Euclidean algorithm, 58, 78
hash function, 6, 10
Hasse–Weil bound, *see* Weil's theorem
Hermite normal form, 109, 112
heuristic, 103
hyperelliptic
curve, 39, 86, 96, 105, 116, 129, 130, 132
function field, 38
involution, 47, 48, 119

ideal class group, 27, 46, 87, 105, 116, 129, 132
ideal class number, 28
imaginary quadratic
curve, 39, 40, 46
number field, 57, 87, 95, 105, 107, 116, 119, 121, 129, 132
inertia degree, 23, 28, 33, 35
infinite point, 18
infrastructure, 118
integral ideal, 27
integral power basis, 47, 87
invariants of group, 104
involution, hyperelliptic, 47, 48, 119
irreducibility, *see* absolute irreducibility

Jacobian, 25, 86, 105, 107, 116, 130, 132

kangaroo, 10
key
public, 6
secret, 6
key exchange, 6
Kummer extension, 32, 38, 39
Kummer's theorem, 46, 48, 50, 87–89

L–polynomial, 30
Lagrange reduction, 63, 78
Lanczos algorithm, 125
lattice, 109, 111, 113
lifting, 8, 126
line, projective, 22
local integral power basis, 47
local parameter, 21, 89

Möbius inversion, 90
matrix
sparse, 123, 125
unimodular, 109, 113
monoid, 84
multiplier, 58, 68

non-singular
curve, 18, 19
point, 18
norm, 47, 51, 88, 105
normal basis, 75
number field, 24
number field sieve, 13

parallel algorithm, 8, 11, 121

partial fraction decomposition, 34
perfect field, 19
periodic sequence, 10
place, 21
plane
    affine, 18
    projective, 18
Pohlig–Hellman attack, 8, 66, 103, 124
point
    closed, 20
    distinguished, 11
    finite, 18
    infinite, 18
    non-singular, 18
    projective, 18
    rational, 20
    singular, 18, 19, 40, 42
pole, 21
pole divisor, 25, 28, 34
Pollard's algorithms, 9–12
polynomial function, 18
positive divisor, 26
prime, 84, 104
    almost, 9
prime divisor, 21, 25, 87, 89
prime divisor theorem, 90, 93, 96, 100, 107
prime number theorem, 84, 90, 93
prime polynomial theorem, 94
principal character, 105
principal divisor, 25
projection, 28
projective
    closure, 19
    curve, 19
    line, 22
    plane, 18
    point, 18
public key, 6
public key cryptography, 5

ramification index, 23, 33, 35
random walk, 9
rational function, 18
rational function field, 20, 22, 26, 30
rational point, 20
real quadratic curve, 39, 40, 46
reciprocal root, 31
reduced divisor, 55, 87
reduction, 57, 59
    Cantor, 60, 61, 77
    Gauß, 59, 77
    Lagrange, 63, 78
regulator, 28
relation, 14, 109, 110, 113
    useful, 113, 114
residue class field, 23, 24, 33
Riemann hypothesis, 17, 31, 88, 96, 105, 107, 116, 132
Riemann's theorem, 25, 55
Riemann–Roch Theorem, 26, 56, 57
root, reciprocal, 31

secret key, 6
semigroup, arithmetical, 84, 92
semireduced divisor, 54, 88, 96
Shanks's algorithm, 9
sieving, 121
signature, *see* digital signature
singularity, 18, 19, 40, 42
size, 84, 104
Smith normal form, 109
smoothness, 14, 84, 85, 92, 95, 100
    curve, 18, 19
smoothness bound, 14, 85, 124
sparse matrix, 123, 125
splitting prime divisor, 23
square and multiply, *see* double and add
strict triangle inequality, 21, 47, 50
subexponential function, 13, 91
subexponentiality, 13, 115, 130
Sylow subgroup, 8
symmetric cryptography, 5, 8

trace, 47
trial division, 85, 114
triangle inequality, 21, 50
    strict, 21, 47, 50
Tschebyscheff's inequality, 114

uniform distribution, 66, 70, 110, 111, 117, 124, 126, 128, 133, 134
uniformising parameter, 21
unimodal function, 116
unimodular matrix, 109, 113
unit, fundamental, 29
useful relation, 113, 114

valuation, *see* discrete valuation

Weil's theorem, 30, 85, 89, 105

zero, 21
zero divisor, 25
zeta function, 30

By the same author

ELLIPTIC CURVES AND THEIR APPLICATIONS TO CRYPTOGRAPHY

An Introduction

Since their invention in the late seventies, public key cryptosystems have become an indispensable asset in establishing private and secure electronic communication, and this need, given the tremendous growth of the Internet, is likely to continue growing. Elliptic curve cryptosystems represent the state of the art for such systems.

Elliptic Curves and Their Applications to Cryptography: An Introduction provides a comprehensive and self-contained introduction to elliptic curves and how they are employed to secure public key cryptosystems. Even though the elegant mathematical theory underlying these cryptosystems is considerably more involved than for other systems, this text requires the reader to have only an elementary knowledge of basic algebra. The text nevertheless leads to problems at the forefront of current research, featuring chapters on point counting algorithms and security issues. The adopted unifying approach treats with equal care elliptic curves over fields of even characteristic, which are especially suited for hardware implementations, and curves over fields of odd characteristic, which have traditionally received more attention.

Elliptic Curves and Their Applications to Cryptography: An Introduction has been used successfully for teaching advanced undergraduate courses. It will be of greatest interest to mathematicians, computer scientists, and engineers who are curious about elliptic curve cryptography in practice, without losing the beauty of the underlying mathematics.

Contents

1. Public Key Cryptography
2. The Group Law on Elliptic Curves
3. Elliptic Curves Over Finite Fields
4. The Discrete Logarithm Problem
5. Counting Points On Elliptic Curves

184 pages
Hardbound
Kluwer Academic Publishers 1999
ISBN 0-7923-8589-6