Chapter 6

Pairing-friendly curves

To realise pairing-based cryptography in practice, we need two things [Sco07a]:

- efficient algorithms for computing pairings; and
- suitable elliptic curves.

The former was briefly outlined in the last chapter (and will be taken much further in the next), whilst this chapter is dedicated to the latter.

6.1 A balancing act

Pairings are fundamentally different to traditional number-theoretic primitives, in that they require multiple groups that are defined in different settings. Namely, $\mathbb{G}_1$ and $\mathbb{G}_2$ are elliptic curve groups, whilst $\mathbb{G}_T$ is a multiplicative subgroup of a finite field. All three groups must be secure against the respective instances of the discrete logarithm problem, which means attackers can break the system by solving either the DLP in $\mathbb{G}_T$ or the EDCLP in $\mathbb{G}_1$ or $\mathbb{G}_2$. As we discussed in Section 2.1, elliptic curve groups currently obtain much greater security per bit than finite fields; this is because the best attacks on the ECDLP remain generic attacks like Pollard rho [Pol78] which have exponential complexity, whilst the best attacks on the DLP have sub-exponential complexity. In other words, to achieve the same security, a finite field group needs to have a much greater cardinality than an elliptic curve group. It is standard to state the complexity of asymmetric primitives in terms of the equivalent symmetric key size. For example, the most

recent ECRYPT recommendations (see `http://www.keylength.com/en/3/`) say that to achieve security comparable to AES-128 (i.e. 128-bit security), we need an elliptic curve group of approximately 256 bits[1] and a finite field of approximately 3248 bits. We give an example of a curve in the context of pairings for which $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ meet these particular requirements.

Example 6.1.1 (Magma script). Let $E/\mathbb{F}_q : y^2 = x^3 + 14$ be the curve with order $\#E(\mathbb{F}_q)$ having large prime factor r, where q and r are given as

$$\begin{aligned} q = {} & 421943326900167228539204394914103813941511244153259151125138128777531 \\ & 7505016692408034796798044263154903329667 \quad (369 \text{ bits}), \\ r = {} & 223697061107578678950388273657862788561030003896406213385139113737656 \\ & 9980702677867 \quad (271 \text{ bits}). \end{aligned}$$

The embedding degree is $k = 9$, i.e. $q^9 - 1 \equiv 0 \bmod r$. Thus, the two elliptic curve groups $\mathbb{G}_1 \in E[r]$ and $\mathbb{G}_2 \in E[r]$ have an order of 271 bits, which meets the current requirements for 128-bit security. Although $\mathbb{G}_T$ is a subgroup of order r (in $\mathbb{F}_{q^k}^*$), the attack complexity is determined by the full size of the field $\mathbb{F}_{q^9}$, which is 3248 bits, also meeting the requirements for 128-bit security.

We discuss an important point with reference to the above example. Namely, if we were to use primes q and r of the same bit-sizes as Example 6.1.1, but which corresponded to a curve with a larger embedding degree k, then this would not increase the security level offered by the pairing. For example, even though $k = 18$ gives a finite field of 6496 bits, which on its own corresponds to a much harder DLP ($\approx$ 175-bit security), the overall complexity of attacking the protocol remains the same, because the attack complexity of the ECDLP has not changed. Such an increase in k unnecessarily hinders the efficiency of the pairing, since the most costly operations in Miller's algorithm take place in $\mathbb{F}_{q^k}$. Thus, the ideal approach is to optimise the balance between r and $\mathbb{F}_{q^k}$ so that both can be as small as possible whilst simultaneously meeting the particular security level required. This was achieved successfully in our example, where $\mathbb{F}_{q^k}$ was exactly the recommended size, and r was only a few bits larger than what is needed to claim 128-bit security.

[1] The "half-the-size" principle between elliptic curve groups and the equivalent asymmetric key size is standard [Sma10, §6.1], since attacks against elliptic curves with order r subgroup have running time $O(\sqrt{r})$. Obtaining the equivalent finite field group size is not as trivial – see [Sma10, §6.2].

Nevertheless, we can still obtain a significant improvement on the parameters used in Example 6.1.1; we can keep all three group sizes the same, whilst decreasing the size of the base field $\mathbb{F}_q$. The Hasse bound (see Eq. (2.6)) tells us that the bit-length of $\#E$ and the bit-length of q will be the same. Thus, it is possible that we can find curves defined over smaller fields whose largest prime order subgroup has the same bit-size as that in Example 6.1.1, and whose embedding degree is large enough to offset the decrease in q and therefore that the corresponding full extension field also meets the security requirements. We give a "prime" example.

Example 6.1.2 (Magma script). Let $E/\mathbb{F}_q : y^2 = x^3 + 2$ be the curve with prime order $r = \#E(\mathbb{F}_q)$, where q and r are given as

$$q = 28757880164823737284021204980065523467377219983513098565427519263513769\\ 64733335173 \quad (271 \text{ bits}).$$

$$r = 28757880164823737284021204980065523467376683719770479098963148984065605\\ 60716472109 \quad (271 \text{ bits}).$$

The embedding degree is $k = 12$, i.e. $q^{12} - 1 \equiv 0 \bmod r$, giving $\mathbb{F}_{q^{12}}$ as a 3248-bit field, which is exactly the same size as the $k = 9$ curve in Example 6.1.1. Thus, $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ have orders of the same bit-lengths as before, but using this curve instead means that arithmetic in $\mathbb{F}_q$ will be substantially faster; a 271-bit field in this case, compared to 369-bit field in the last.

In light of the difference between Example 6.1.1 and Example 6.1.2, an important parameter associated with a curve that is suitable for pairings is the ratio between the field size q and the large prime group order r, which we call the ρ-value, computed as

$$\rho = \frac{\log q}{\log r}.$$

Referring back to the two curves above, we have $\rho = \frac{\log q}{\log r} = \frac{369}{271} = 1.36$ in Example 6.1.1, whilst $\rho = \frac{\log q}{\log r} = \frac{271}{271} = 1$ in Example 6.1.2. The ρ-value essentially indicates how much (ECDLP) security a curve offers for its field size, and since we generally prefer the largest prime divisor r of $\#E$ to be as large as possible, $\rho = 1$ is as good as we can get. Indeed, the curve in Example 6.1.2 with $\rho = 1$ belongs to the famous Barreto-Naehrig (BN) family of curves [BN05], which all have $k = 12$ and for which the ratio between the sizes of r and $\mathbb{F}_{q^k}$ make them perfectly suited

to the 128-bit security level. This ratio between these group sizes is $\rho \cdot k$ (i.e. $\frac{\log q^k}{\log r} = k \cdot \frac{\log q}{\log r}$), so for commonly used security levels, Figure 6.1 gives the value of $\rho \cdot k$ that balances the current attack complexities of the DLP and ECDLP. Different information security and/or intelligence organisations from around the

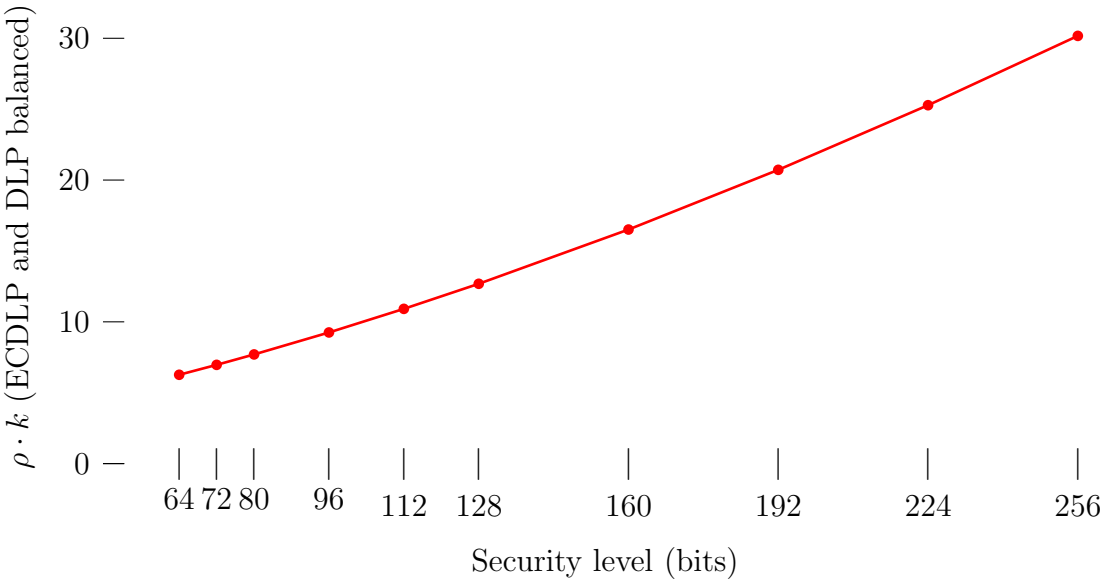

Figure 6.1: The value of $\rho \cdot k$ that balances the complexity of the DLP and ECDLP for commonly used security levels.

globe, such as NIST (the USA) and FNISA (France), give slightly different key size recommendations and complexity evaluations of the algorithms involved; all of this information is conveniently collected at `http://www.keylength.com/`. We have chosen to generate Figure 6.1 according to the numbers in the (most) recent ECRYPT II report [Sma10], which is also summarised there.

Having seen two examples above, we are now in a position to define a *pairing-friendly* curve. Following [FST10], we say that a curve is pairing-friendly if the following two conditions hold:

- there is a prime $r \geq \sqrt{q}$ dividing $\#E(\mathbb{F}_q)$ (i.e. $\rho \leq 2$), and
- the embedding degree k with respect to r is less than $\log_2(r)/8$.

Thus, in their widely cited taxonomy paper, Freeman *et al.* [FST10] consider pairing-friendly curves up to $k = 50$, which is large enough to cover recommended levels of security for some decades yet.

Balasubramanian and Koblitz [BK98] show that, for q of any suitable cryptographic size, the chances of a randomly chosen curve over $\mathbb{F}_q$ being pairing-

friendly is extremely small. Specifically, they essentially show that the embedding degree (with respect to r) of a such a curve is proportional to (and therefore is of the same order as) r, i.e. $k \approx r$. Very roughly speaking, such an argument is somewhat intuitive since (for a random curve) $\#E$ can fall anywhere in the range $[q+1-2\sqrt{q}, q+1+2\sqrt{q}]$, so r can be thought of as independent of q, meaning that the order of q in $\mathbb{Z}_r^*$ is random (but see [BK98] for the correct statements). Therefore, imposing that k is small enough to work with elements in $\mathbb{F}_{q^k}$ is an extremely restrictive criterion, so one can not hope to succeed if randomly searching for pairing-friendly curves amongst arbitrary elliptic curves. Thus, in general, pairing-friendly curves require very special constructions.

In Section 6.2 we briefly discuss supersingular elliptic curves, which always possess embedding degrees $k \leq 6$ [MOV93, §4], and (so long as $r \geq \sqrt{q}$) are therefore always pairing-friendly. Referring back to Figure 6.1 though, we can see that having $k > 6$ is highly desirable for efficient pairings at the widely accepted security levels, and thus in Section 6.3 we focus on the ordinary (non-supersingular) case and outline the constructions that achieve pairing-friendly curves with $k > 6$.

6.2 Supersingular curves

Recall from Section 4.1 that an elliptic curve E is characterised as supersingular if and only if a distortion map exists on E. There are essentially five types of supersingular curves that are of interest in PBC [Gal05, Table IX.1], but here we will only mention two. This is because we are only concerned with prime fields in this text, and the other three are either defined over $\mathbb{F}_{p^2}$, $\mathbb{F}_{2^m}$ or $\mathbb{F}_{3^m}$. As Galbraith mentions, a problem in characteristic 2 and 3 is that there is only a small number of curves and fields to choose from, so there is an element of luck in the search for a curve whose order contains a large prime factor. Another problem in small characteristic is that there exist enhanced algorithms for discrete logarithms (see [Gal05, Ch. IX.13]).

All supersingular curves over large prime fields have $\#E(\mathbb{F}_q) = q+1$, from which it follows that $k = 2$, i.e. regardless of the prime factor $r \neq 2$, $r \mid q+1$ implies $r \nmid q-1$ but $r \mid q^2-1$. We have already seen examples of the two popular supersingular choices in Section 4.1, whose general forms are given in Table 6.1.

We give another example of both cases below, but we choose the parameter

q	E	distortion map ϕ	e.g.
2 mod 3	$y^2 = x^3 + b$	$(x, y) \mapsto (\zeta_3 x, y)$, $\zeta_3^3 = 1$	Eg. 4.1.4 (Fig. 4.5)
3 mod 4	$y^2 = x^3 + ax$	$(x, y) \mapsto (-x, iy)$, $i^2 = -1$	Eg. 4.1.5 (Fig. 4.6)

Table 6.1: The two types of popular supersingular curves over prime fields.

sizes to serve another purpose: to show how important it is to employ ordinary curves with higher embedding degrees.

Example 6.2.1 (Magma script). We will choose $q \equiv 11 \bmod 12$ so we can define both examples in Table 6.1 over the same field, but also so that the security of these curves in the context of PBC matches the security of the curve with $k = 12$ in Example 6.1.2. For the ECDLP security to be 128 bits, r still only needs to be 256 bits in size. However, since $k = 2$, for $\mathbb{F}_{q^k}$ to be around 3248 bits, q needs to be around 1624 bits:

$q =$425708693169757088196017853607835113595127103859429924930531263283244403251872949802982860038531930965867890444658222153407204383584492024637762799391807569669124814253270947366226515064812665901907204494611177526596015257984009814596057160388672298355821309046798841446111721495601835913381835880170934319890420895521320439930666405003725309562669243847766834546592867695533445054256132471093279787853214492986394176521193456205570309658462204234557728373615304193316916440130004424612327.

Consider $E_1/\mathbb{F}_q : y^2 = x^3 + 314159$ and $E_2/\mathbb{F}_q : y^2 = x^3 + 265358x$. Both curves have order $\#E_1(\mathbb{F}_q) = \#E_2(\mathbb{F}_q) = q + 1 = hr$, where h is a 1369-bit cofactor and r is the 256-bit prime given as

$r =$5789604461865809771178549250434395392663499233282028201972879200395656482006 3.

The distortion maps are defined over $\mathbb{F}_{q^2} = \mathbb{F}_q(i)$, where $i^2 + 1 = 0$ – see Table 6.1 or Examples 4.1.4 and 4.1.5. The huge size of q stresses the importance of adhering to the optimal ratio of $\rho \cdot k$ suggested by Figure 6.1. A rough but conservative approximation of the complexity of field multiplications in the 1624-bit field, compared to the 271-bit field in Example 6.1.2 gives a ratio of at least 25 : 1. Referring back to the discussion of pairing types in Section 4.2, this gives some idea of the computational price one pays when insisting on the

computability of ψ (as well as the other desired properties offered by a Type 1 pairing), rather than adopting a Type 3 pairing and trusting in the heuristics of Chatterjee and Menezes in the absence of such a ψ [CM09, Men09].

We round out this section by remarking that although supersingular elliptic curves are limited to $k \leq 6$, Rubin and Silverberg give a practical way to obtain larger values of k using Weil descent [RS02]. Alternatively, one can employ a higher genus supersingular curve to obtain a higher embedding degree [Gal01, RS02]. As Galbraith remarks however, there are severe efficiency limitations in both scenarios, and we achieve faster pairings in practice by using ordinary pairing-friendly elliptic curves [Gal05, Ch. IX.15].

6.3 Constructing ordinary pairing-friendly curves

There are three main methods of constructing ordinary pairing-friendly elliptic curves. The two most general methods, the Cocks-Pinch [CP01] and Dupont-Enge-Morain [DEM05] algorithms, produce curves with $\rho = 2$, which is more often than not undesirable when compared to the ρ-values obtained by the third method. Moreover, the third method encompasses all constructions that produce *families* of pairing-friendly elliptic curves, which have been the most successful methods of producing curves that are suitable for current and foreseeable levels of security.

All of the constructions in the literature essentially follow the same idea: fix k and then compute integers t, r, q such that there is an elliptic curve $E/\mathbb{F}_q$ with trace of Frobenius t, a subgroup of prime order r, and an embedding degree k. The *complex multiplication method* (CM method) of Atkin and Morain [AM93] can then be used to find the equation of E, provided the CM discriminant D of E is not too large: D is the square-free part of $4q - t^2$, i.e.

$$Df^2 = 4q - t^2, \tag{6.1}$$

for some integer f. Equation (6.1) is often called the CM equation of E, and by "D not too large" we mean D is less than, say 10^{12} [Sut12].

In 2001, Miyaji, Nakabayashi and Takano [MNT01] gave the first constructions of ordinary pairing-friendly elliptic curves. Their method has since been greatly extended and generalised, but all of the constructions of families essentially followed from their idea, which is aptly named the MNT strategy or MNT

criteria [FST10, Gal05]. For some special cases, Miyaji *et al.* used the fact that if k is the (desired) embedding degree, then $r \mid q^k - 1$ implies $r \mid \Phi_k(q)$, since the k-th cyclotomic polynomial $\Phi_k(x)$ is the factor of $x^k - 1$ that does not appear as a factor of any polynomial $(x^i - 1)$ with $i < k$ [Gal05, IX.15.2]. For these cases they were also the first to parameterise *families* of pairing-friendly curves, by writing t, r and q as polynomials $t(x)$, $r(x)$ and $q(x)$ in terms of a parameter x. Miyaji *et al.* focussed on embedding degrees $k = 3, 4, 6$ and assumed that the group order was to be prime, i.e. $r(x) = q(x) + 1 - t(x)$ (from (2.6)). They proved that the only possibilities for $t(x)$ and $q(x)$ (and hence $r(x)$) are

$$\begin{aligned} k = 3: &\quad t(x) = -1 \pm 6x &&\text{and} \quad q(x) = 12x^2 - 1; \\ k = 4: &\quad t(x) = -x, x + 1 &&\text{and} \quad q(x) = x^2 + x + 1; \\ k = 6: &\quad t(x) = 1 \pm 2x &&\text{and} \quad q(x) = 4x^2 + 1. \end{aligned}$$

Example 6.3.1 (Magma script). We kick-start a search for a $k = 4$ (toy) MNT curve with $x = 10$, incrementing by 1 until $q(x) = x^2 + x + 1$ and $r(x) = q(x)+1-t(x)$ (with either of $t(x) = -x$ or $t(x) = x+1$) are simultaneously prime. At $x = 14$, both $q = q(x) = 211$ and $r = r(x) = 197$ (with $t(x) = x+1$) are prime, so we are guaranteed an elliptic curve $E/\mathbb{F}_q$ with r points and embedding degree $k = 4$ (notice $q^4 - 1 \equiv 0 \bmod r$). The CM equation yields $Df^2 = 4q - t^2 = 619$, which itself is prime, so $f = 1$ and thus we seek a curve over $\mathbb{F}_q$ with CM discriminant $D = 619$. The CM method produces one such curve as $E/\mathbb{F}_q : y^2 = x^3 + 112x + 19$. Notice that $\phi_4(q(x)) = q(x)^2 + 1 = (x^2 + 1) \cdot (x^2 + 2x + 2)$, both factors being the possibilities for $r(x)$.

Notice that the toy example above has $\rho = \frac{\log q}{\log r} = \frac{\log 211}{\log 197} = 1.01$. For x of cryptographically large size though, we will get $\rho = 1$ since $q(x) = x^2 + x + 1$ and $r(x) = x^2 + 2x + 2$ or $r(x) = x^2 + 1$ have the same degree. In general parameterised families then, we use the degrees of $q(x)$ and $r(x)$ to state ρ as

$$\rho = \frac{\deg(q(x))}{\deg(r(x))}.$$

A number of works followed the MNT paper and gave useful generalisations of their results. In particular, we mention the work by Barreto *et al.* [BLS02], Scott and Barreto [SB06], and Galbraith *et al.* [GMV07], all three of which obtain more parameterised families by relaxing the condition that the group order is prime

and allowing for small cofactors so that $\#E = hr$. Another observation made by Barreto *et al.* that somewhat simplifies the process is the following: $r \mid \Phi_k(q)$ and $q + 1 - t \equiv 0 \bmod r$ combine to give that $\Phi_k(t-1) \equiv 0 \bmod r$ [BLS02, Lemma 1]. Substituting $hr = q + 1 - t$ into the CM equation in (6.1) gives

$$Df^2 = 4hr - (t-2)^2. \tag{6.2}$$

In Section 3.1 of [BLS02], Barreto *et al.* obtain many nice parameterised families for various k by considering a special case of the above equation with $t(x) = x+1$, $D = 3$ and (since $r \mid \Phi_k(x)$) finding $f(x)$ and $m(x)$ to fit

$$3f(x)^2 = 4m(x)\Phi_k(x) - (x-1)^2. \tag{6.3}$$

We note that curves with CM discriminant $D = 3$ are always of the form $y^2 = x^3 + b$. A convenient solution to Equation (6.3) for $k = 2^i \cdot 3$ is $m = (x-1)^2/3$ and $f(x) = (x-1)(2x^4-1)/3$, for which we can take $r = \Phi_k(x)$. Taking $i = 3$, we give a cryptographically useful example of a BLS (Barreto-Lynn-Scott) curve with $k = 24$.

Example 6.3.2 (Magma script). Following the above description, the BLS family with $k = 24$ is parameterised as $q(x) = (x-1)^2(x^8-x^4+1)/3+x$, $r(x) = \Phi_{24}(x) = x^8-x^4+1$, $t(x) = x+1$. The family has $\rho = \frac{\deg(q(x))}{\deg(r(x))} = 10/8 = 1.25$ and therefore $\rho \cdot k = 30$. Referring back to Figure 6.1, we see that such a curve gives a nice balance between the sizes of r and q^k (the ECDLP and DLP) for pairings at the 256-bit security level. Indeed, at present this family remains the front-runner for this particular security level [Sco11, CLN11]. To find a curve suitable for this level, we need r to be about 512 bits, and since $\deg(r(x)) = 8$, we will start the search for q, r both prime with a 64-bit value; note that $x \equiv 1 \bmod 3$ makes $q(x)$ an integer, so the first such value is $x = 2^{63} + 2$. After testing a number of incremental $x \leftarrow x + 3$ values, $x = 9223372036854782449$ gives $q(x)$ and $r(x)$ as 629 and 505 bit primes respectively. Since $D = 3$ and $E/\mathbb{F}_q : y^2 = x^3 + b$, i.e. there is only one curve constant, we do not need to use the CM method. Instead, it is usually quicker to try successive values of b until we find the correct curve. In this case, $b = 1$ gives $E/\mathbb{F}_q : y^2 = x^3 + 1$ as our pairing-friendly $k = 24$ BLS curve.

Barreto *et al.* [BLS02, §3.2] actually give a more general algorithm which, instead of insisting that $t = x + 1$, takes $t = x^i + 1$. Brezing and Weng [BW05]

found even more useful families by searching with more general polynomials for $t(x)$. Several constructions followed by looking for parameterisations that satisfy the following conditions which define a family [FST10, Def. 2.7] (also see [Fre06, Def. 2.5]):

(i) $r(x)$ is nonconstant, irreducible, and integer-valued with a positive leading coefficient.

(ii) $r(x) \mid q(x) + 1 - t(x)$.

(iii) $r(x) \mid \Phi_k(t(x) - 1)$.

(iv) The parameterised CM equation $Df^2 = 4q(x) - t(x)^2$ has infinitely many integer solutions (x, f).

Referring to condition (iv) above, we say that a family parameterised by $(t(x),\ r(x),\ q(x))$ is a *complete* family if there exists $f(x) \in \mathbb{Q}[x]$ such that $Df(x)^2 = 4q(x) - t(x)^2$. Otherwise, we say the family is *sparse*. We have already seen a curve belonging to the popular Barreto-Naehrig (BN) family in Example 6.1.2. In the following example we look at the BN parameterisations in terms of the above conditions.

Example 6.3.3 (Magma script). Barreto and Naehrig [BN05] discovered that, for $k = 12$, setting the trace of Frobenius t to be $t(x) = 6x^2 + 1$ gives $\Phi_{12}(t(x) - 1) = \Phi_{12}(6x^2) = (36x^4 + 36x^3 + 18x^2 + 6x + 1)(36x^4 - 36x^3 + 18x^2 - 6x + 1)$. This facilitates the choice of $r(x)$ as the first factor $r(x) = 36x^4 + 36x^3 + 18x^2 + 6x + 1$, from which taking $q(x)$ as $q(x) = 36x^4 + 36x^3 + 24x^2 + 6x + 1$ means not only that $r(x) \mid q(x) + 1 - t(x)$ (condition (ii) above), but in fact that $r(x) = q(x) + 1 - t(x)$. Thus, when x is found that makes $r(x)$ and $q(x)$ simultaneously prime, we have a pairing-friendly curve with $k = 12$ that has prime order. Not only is the ρ-value $\rho = 1$ ideal, but there are many more reasons why BN curves have received a great deal of attention [DSD07, PJNB11, AKL+11]. Notice that $D = 3$ and $f(x) = 6x^2 + 4x + 1$ satisfies the CM equation (condition (iv) above), so the BN family is a complete family and BN curves are always of the form $y^2 = x^3 + b$.

The last point of Example 6.3.3 is a crucial one. Referring back to Section 4.3, we know that $D = 3$ curves of the form $y^2 = x^3 + b$ admit cubic and sextic twists. Thus, in the case of BN curves where $k = 12$, we can make use of a sextic twist to represent points in $\mathbb{G}_2 \in E(\mathbb{F}_{q^{12}})$ as points in a much smaller subfield on the twist, i.e. in $\Psi^{-1}(\mathbb{G}_2) = \mathbb{G}_2' \in E'(\mathbb{F}_{q^2})$. In general then, when k has

the appropriate factor $d \in \{3, 4, 6\}$, we would like to make use of the highest degree twist possible, so we would prefer our pairing-friendly curves to be of the following two forms:

degree d	curve	j-invariant	CM discriminant	field
$3, 6 \mid k$	$y^2 = x^3 + b$	$j(E) = 0$	$D = 3$	$q \equiv 1 \bmod 3$
$4 \mid k$	$y^2 = x^3 + ax$	$j(E) = 1728$	$D = 1$	$q \equiv 1 \bmod 4$

Table 6.2: Pairing-friendly elliptic curves admitting high-degree twists.

See [Sil09][p. 45] for the definition of the j-invariant of an elliptic curve (and the associated calculations); we simply remark that two elliptic curves $E/\mathbb{F}_q$ and $\tilde{E}/\mathbb{F}_q$ are isomorphic over $\overline{\mathbb{F}}_q$ if and only they have the same j-invariant. Due to the preferences in Table 6.2, our discussion will really only be dealing with curves of j-invariants (respectively CM discriminants) $j \in \{0, 1728\}$ (respectively $D \in \{3, 1\}$). In this respect, we are also very fortunate that most of the best constructions of pairing-friendly families have either $D = 1$ or $D = 3$, depending on the embedding degree they target. In general, a severe loss of efficiency is suffered in pairing computations when choosing a curve that does not offer a high-degree twist, so at any particular security level we tend to focus on the curves whose embedding degrees are suitable, both according to Figure 6.1 and which contain $d \in \{3, 4, 6\}$ as a factor [FST10, §8.2]. Besides, as we will see in the next chapter, there are further efficiency reasons that happily coincide with having $d \mid k$ for $d \in \{3, 4, 6\}$. The equivalence conditions on q in Table 6.2 are to ensure E is ordinary, complementing the supersingular cases in Table 6.1.

Our last example in this chapter belongs to another complete family from the more recent work of Kachisa, Schaefer and Scott [KSS08], who present record-breaking (in terms of the lowest ρ-value) curves for embedding degrees $k \in \{16, 18, 36, 40\}$.

Example 6.3.4 (Magma script). We choose a KSS curve with $k = 16$, which is parameterised by $t(x) = (2x^5 + 41x + 35)/35$, $q(x) = (x^{10} + 2x^9 + 5x^8 + 48x^6 + 152x^5 + 240x^4 + 625x^2 + 2398x + 3125)/980$ and $r(x) = (x^8 + 48x^4 + 625)/61250$. This family has $\rho = 5/4$, so referring back to Figure 6.1 we see that $\rho \cdot k = 20$ is a nice fit for pairings at the 192-bit security level. Thus, r should be around 384 bits, so starting our search with x around 2^{50} should do the trick (we add the extra two bits to account for the 16-bit denominator of $r(x)$). The polynomials for $q(x)$ and $t(x)$ can only take on integers if $x \equiv \pm 25 \bmod 70$, so we start

with $x \equiv 2^{50} + 21 \equiv 25 \bmod 70$ and iterate accordingly. We soon arrive at $x = 1125899907533845$, which gives a 491-bit q as

$$q = 3340194518359587075607904514504348578130581647867654217642899810042867643534741048241225178436682317003010155280705836842596368221341280505964970897,$$

and a 385-bit prime factor r of $\#E(\mathbb{F}_q)$ as

$$r = 4215918189011304280250800671237881596873006793850195934448558095361634092780222932018149564359414764607793390912163 3.$$

Again, we do not need the CM method to find the curve: we simply start with $a = 1$ in $y^2 = x^3 + ax$ and increment until we find $a = 3$ which gives the correct curve as $E/\mathbb{F}_q : y^2 = x^3 + 3x$. E has embedding degree 16 with respect to r, so the full extension field $\mathbb{F}_{q^k}$ is 7842 bits.

We finish this chapter with two important remarks.

Remark 6.3.1 (Curves for ECC vs. curves for PBC). At the highest level, finding curves that are suitable for ECC really imposes only one condition on our search, whilst finding curves that are suitable for PBC imposes two: in ECC we only look for curves with large prime order subgroups, whilst in PBC we have the added stipulation in that we also require a low embedding degree. Whilst one can search for suitable curves for ECC by checking "random" curves until we come across one with almost prime order, in PBC we require very special constructions (like all those discussed in this chapter) that also adhere to the extra criterion – as we have already discussed, we can not expect to find any pairing-friendly curves by choosing curves at random [BK98]. A major consequence is that in ECC we can specify the underlying field $\mathbb{F}_q$ however we like before randomly looking for a suitable curve over that field. In this case fields can therefore be chosen to take advantage of many low-level optimisations; for example, Mersenne primes achieve very fast modular multiplications which blows out the relative cost of inversions. On the other hand, in PBC we are confined to the values taken by the polynomials $q(x)$ and have limited control over the prime fields we find. Thus, we are not afforded the luxury of many low-level optimisations and this drastically affects the ratios between field operations (inversions/multiplications/squarings/additions). For example, whilst $\mathbb{F}_q$-inversions

in ECC are commonly reported to cost more than 80 $\mathbb{F}_q$-multiplications, the ratio in the context of PBC is nowhere near as drastic [LMN10, AKL+11]. This means we often have to rethink trade-offs between field operations that were originally popularised in ECC.

Remark 6.3.2 (Avoiding pairing-friendly curves in ECC). In the previous remark we said that in ECC we only need to satisfy one requirement (the large prime subgroup), but this is not the full story. In fact, in this context we prefer to choose curves that are strictly not pairing-friendly. After all, in ECC there is no need for a low embedding degree, so choosing a curve that (unnecessarily) has one gives an adversary another potential avenue for attack. Indeed, exploiting curves with low embedding degrees in the context of ECC was the first use of pairings in cryptography – the famous Menezes-Okamato-Vanstone (MOV) [MOV93] and Frey-Rück (FR) [FR94] attacks. Thus, so long as we avoid supersingular curves, the heuristic argument [BK98] tells us that the curves we choose at random will have enormous embedding degrees with overwhelmingly high probability, so this is not a restriction in the sense of Remark 6.3.1.

6.4 Chapter summary

We stressed the importance of finding elliptic curves with large prime order subgroups and small embedding degrees, i.e. pairing-friendly curves. We showed that supersingular curves, whilst easy to find, severely limit the efficiency of pairing computations, particularly at moderate to high levels of security, because they are confined to $k \leq 6$ (and $k \leq 2$ over prime fields). Thus, we turned our focus to the more difficult task of constructing ordinary pairing-friendly elliptic curves, and summarised many landmark results that have enhanced this arena over the last decade. In particular, we gave examples of some of the most notable families of pairing-friendly elliptic curves, some of which have already become widespread in real-world implementations of pairings.