Curves Over Algebraic Number Fields

General Genus

The case of curves of arbitrary genus is much more difficult than the case of curves of genus 1, and there are no well-developed algorithms for this case. I have not been able to code any significant program to deal with this case because of the large number of subsidiary algorithms for which I do not have programs, though such programs have been written elsewhere, or can readily be written. Presented here, therefore, are the outlines of techniques which will enable one to bound the torsion of curves of arbitrary genus over algebraic number fields.

The matter is complicated by the fact that most of the necessary supporting mathematics is couched in very abstract language and much is contained only in the "folklore" of Algebraic Geometry. For this reason the results stated will be less detailed than in the rest of the monograph, and there will not always be complete references to support them. The theory to be described is true for curves of genus 1 as well as for curves of higher genus, and in fact lies behind much of the theory described in the previous chapter. We will therefore be able to illustrate# most of this work by examples over* curves of genus 1, and we will usually do this for simplicity although the main application of the work will be for curves of genus > 1.

The basic idea is similar to many processes in algebraic geometry: we reduce the problem to one over finite fields, in which we can compute explicitly and over which we

---

There is an excellent illustration of these methods at work in Mazur & Swinnerton-Dyer (1974, p. 21, Lemma 1).

* Also there are many more computations and tables relating to elliptic curves (Swinnerton-Dyer, 1974, for example), so that it is easier to find suitable examples, and to explain their behaviour.

can perform complete searches, and then we can piece together the information from several finite fields in order to solve the original problem. In the terminology of computer algebra, we will adopt a *modular* approach.

Good Reduction

Let K be the common field of definition of the curve C and the divisor D, so that D corresponds to an element of the Jacobian of C as defined over K. Let p' be a prime ideal of K lying over the rational prime p. Let K' be the *residue class field* of K mod p', i.e. the field generated by the elements of the integers of K modulo the ideal p'. Let A be an Abelian variety over K (normally considered to be the Jacobian of C from our point of view, but the general theory does not require this). Let A' be the variety over K' defined by the same equations as A over K (i.e. A' is a *specialisation* of A in the sense of Mumford (1965)). We note that A' is defined over a finite field, and therefore has only a finite number of elements, which must all be torsion elements.

If p is any rational prime, and G is an Abelian group (normally a Jacobian divisor group but this is not necessary), then define the *p-part* of G to be those elements of G which have order a power of p. This is clearly a subgroup of G, and its order is a power of p. Define the *non-p-part* of G to be those elements of G whose (finite) order is coprime to p. This too is a subgroup of G. If every element of G is of finite order, then G is the direct product of its p-part and non-p-part for any prime p. Because of the manipulation of p-parts that we will engage in, we shall require the following algorithm:

MAX_POWER

Input:

P: a positive integer, frequently a prime.

N: a positive integer.

Output:

Q: the largest integral power of $P \leq N$.

We will not bother to describe such a simple algorithm in detail.

Following Serre and Tate (1968) we say that A *has good reduction** at p' iff $A = A' \times K$, where the operation "x" means the taking of tensor products over the valuation ring of p'. The following result is essentially well-known:

Theorem 1 If A has good reduction at p' which lies over p, then the non-p-part of the torsion subgroup of A is injected into the non-p-part of the torsion sub-group of A'.

Corollary 2 The size of the non-p-part of the torsion group of A divides the size of A' (viewed as a group).

We can also state the following result (from Serre and Tate (1968) or Shimura and Taniyama (1961)) as adapted to our circumstances and notation:

Theorem 3 For a fixed K and A, there are only a finite number of primes of bad reduction.

Theorem 2 of Chapter 7 implies that this is still true if we restrict ourselves to unramified primes, which we will do in the future. This means that we can use the following algorithm to reduce the general problem to that of torsion bounds modulo prime ideals.

---

* Roughly speaking, this condition means that A can be reconstructed from a knowledge of A' and K. This is a similar concept to that of *good evaluations* or *lucky primes* (see, e.g., Yun 1973).

BOUND_TORSION

(Version 1)

Input:

K: an algebraic number field.

F(X,Y): the equation of a curve defined over K.

Output:

N: a bound for the torsion of F over K.

[1] For P = 2,3,5,... do:
For each prime ideal P' of K with P' | P do:
if GOOD_REDUCTION(F,K,P')
then do:

[1.1] NON_P_PART := FINITE_BOUND(F,K,P').

The algorithm FINITE_BOUND should give a bound for the torsion modulo the prime ideal P'.

[1.2] Go to [2].

[2] For Q = prime after P,... do:
For each prime ideal Q' of K with
Q' | Q do:
if GOOD_REDUCTION(F,K,Q')
then do:

[2.1] NON_Q_PART := FINITE_BOUND(F,K,Q').

[2.2] Go to [3]

[3] ANSWER1 := NON_Q_PART * MAX_POWER(Q,NON_P_PART).

ANSWER2 := NON__P__PART * MAX__POWER(P,NON__Q__PART).
Return minimum(ANSWER1,ANSWER2).

ANSWER1 splits the torsion group into its non-q-part and its q-part, while ANSWER2 does the converse.

This algorithm is not the only one which can use the mechanism we have developed to find a bound for the torsion of an elliptic curve. One possibility is to take 3 primes of good reduction rather than 2, and then observe that the p-torsion, for any p, will appear in the non-q-parts for 2 primes q, so that the torsion is bounded by the square root of the product of the 3 non-q-torsions.

Later in the Chapter we will see that something much better can be done when we know precisely what the torsion is over the finite field.

Torsion over Finite Fields

The aim of this section is to describe various ways of finding, or at least bounding, the size of Jacobian divisor group over a finite field, i.e. the implementation of the algorithm FINITE__BOUND. Much of this material comes from Lang (1959, chapter V, especially section 3)#. We require a piece of notation: if A is an Abelian variety defined over a field K, let $|A|_K$ be the number of points of A defined over K. Our first remark is that, if C is a curve of genus g, then Jac(C) is an Abelian variety of dimension g (see the discussion at the end of chapter 5). Hence we wish to discover $|\mathrm{Jac}(C)|_K$ for the curve C under investigation. Unfortunately, this is not easily related to $|C|_K$. However we do have the following Lemma:

Lemma 4 (This is proved in Lang (1959, pp. 139-140).) Let A be an Abelian variety of dimension r over a finite field K with q elements. Let $\phi : A=>A$ be the Frobenius

I am grateful to Professor Sir Peter Swinnerton-Dyer for drawing my attention to this work, and for correcting many errors in my understanding of it.

endomorphism induced by the Frobenius* endomorphism on K. Then $|A|_K = \prod_{j=1}^{j=2r}(w_j - 1)$ where the w_j are the characteristic roots of ϕ.

Lemma 5 (Lang (1959, p.138 Lemma 2 and Chapter IV section 3.) With the notation as above, $|w_j| = q^{1/2}$.

Theorem 6 $|\mathrm{Jac}(C)|_K \leq (q^{1/2} + 1)^{2g}$.

Proof: From Lemma 4 and 5 above.

Criteria for Good Reduction

Theorem 7 If C has good reduction, then $\mathrm{Jac}(C)$ does.

The converse is not necessarily true, but it does not seem worth trying to take advantage of those primes at which $\mathrm{Jac}(C)$ has good reduction but C does not. I have managed to avoid any explicit construction of Jacobians in the programming, and I feel that the slight gains would be outweighed by additional complexity.

Clearly we will not have good reduction if the genus of C' is not the same as that of C, for then the Jacobians would have different dimensions. In fact the following is a necessary and sufficient criterion for good reduction to occur for C, and hence for $\mathrm{Jac}(C)$:

Theorem 8 If C and C' have the same genus, and F' is absolutely irreducible (i.e. irreducible in all algebraic extensions of K'), then we have good reduction.

Let us consider the two halves of this test separately: the genus preservation part first. The genus of C' can never be more than that of C, so we have to detect those cases

---

* The *Frobenius endomorphism* is defined by $x \mapsto x^q$. See Eichler (1966, p.249) for further details.

in which the genus decreases. This can happen in one of two ways: a differential of the first kind on C can cease to be a differential of the first kind on C', or the space of differentials of the first kind can contract. As an example of the first, consider $Y^2 = (X-1)(X+1)(X+2)$, which has genus 1 over $\boldsymbol{Q}$ with one differential of the first kind, viz. dX/Y. However, modulo 3 this function has a pole at $X=1=-2$, for $X-1$ is a local parameter there, and $1/Y$ behaves like $1/(X-1)\sqrt{X+1}$. In fact, of course, the curve has genus 0 modulo 3. Since we know the differentials of the first kind as a result of computing the genus, this case is fairly easy to test for. The other possibility is that the differentials of the first kind will cease to be independent, and as an example of this consider the space curve $Y^2 = X^3-1$ and $Z^2 = X^3+2$. Here both dX/Y and dX/Z are independent differentials of the first kind over the rationals, but not when taken modulo 3, as one might expect.

Note that there is a possibility that we will reject some primes as not giving rise to good reductions when in fact they do, since we might inadvertently have an expression for a differential which was divisible by the prime in question. This is especially likely to happen over algebraic number fields, when we are dealing with prime ideals rather than straight primes, since in this case we cannot just divide out the prime. However, this can only happen finitely often, so we still have an infinite number of primes of good reduction available. This point may prove computationally embarrassing, but it does not affect the theory.

Example

We will consider the example of Tate's curve with $D=2$ from Appendix 2 example 4, and we will work over the rationals. This is not necessary, inasmuch as either Mazur's bound or the Lutz-Nagell approach will suffice, but it is a relatively easy case to work and explain. The equation is $y^2 = 4x^3-15x^2+8x+16$. Clearly this does not have good reduction mod 2, because the equation reduces to $y^2 = x^2$ and not only is this clearly not irreducible but also the differential of the first kind, $1/y$, does not remain a differential of the first kind.

Modulo 3, the equation reduces to $y^2 = x^3 + 2x + 1$, which is irreducible and preserves the differentials of the first kind. Therefore the non-3-part of the torsion is at most $(3^{1/2} + 1)^2 = 7$ (after rounding down to an integer). The curve also has good reduction modulo 5, so that the non-5-part is at most $(5^{1/2} + 1)^2 = 10$ (after rounding down). Therefore the curve has torsion at most* minimum(7*9,10*5) = 50.

A Better Algorithm

In this section we will develop the consequences of knowing exactly how many points there are on the Jacobian of the curve over our finite field. Just as in the previous case we required the trivial algorithm MAX__POWER to extract p-parts, here we have an algorithm OCCURRENCES to do the same.

---

* In fact we can do rather better than this if we consider the various cases separately.
Case 1. The curve has 7-torsion. In this case the torsion group must have precisely 7 elements (this is, in fact, what happens).
Case 2. The curve has 3-torsion and 5-torsion. In this case the 3-torsion is bounded by 9, and the 5-torsion by 5, so the whole torsion is bounded by 45 (since introducing 2-torsion decreases the total). In fact this can be ruled out by considering reduction modulo 11, which gives 18 (and hence 15 once we have ensured that the group structure is maintained) as a bound for the non-11-part, and hence for the entire torsion.
Case 3. The curve has 3-torsion (but not 5- or 7-torsion). Then the maximum size of the torsion group is 10.
Case 4. The curve has 5-torsion (but not 3-torsion). In this case there can also be no 2-torsion (consider the non-3-part) so the torsion group must have order 5.
Case 5. The curve has only 2-torsion. In this case the torsion is bounded by 4.
Unfortunately I know of no good way of mechanising this sort of intuition, so we are left with the bound of
minimum(NON__Q__PART * MAX__POWER(Q,NON__P__PART),
NON__P__PART * MAX__POWER(P,NON__Q__PART)).

OCCURRENCES

Input:

P: a positive integer, frequently a prime.

N: a positive integer.

Output:

Q: the largest integral power of P dividing N.

We will not bother to describe such a simple algorithm in detail.

Finding the number of points on the Jacobian of the curve is not easy, and I have no algorithm to suggest for doing it. In the case of curves of genus 1, then the curve is the Jacobian and there is no real problem (except that one has to be careful when counting multiple points, and to distinguish ordinary multiple points from ramified points). Nevertheless we present this algorithm, since it is clearly the correct way to approach the torsion divisor problem (from our current state of knowledge) in the case of algebraic number fields.

BOUND_TORSION

(Version 2)

Input:

K: an algebraic number field.

F(X,Y): the equation of a curve defined over K.

Output:

N: a bound for the torsion of F over K,

such that the true torsion is a factor of N (as opposed to version 1, where we merely knew that it was at least N).

[1] For P = 2,3,5,... do:

For each prime ideal P' of K with

P' | P do:

if GOOD__REDUCTION(F,K,P')

then do:

[1.1] NON__P__PART := FINITE__TORSION(F,K,P').

The algorithm FINITE__TORSION should give the torsion modulo the prime ideal P'.

[1.2] Go to [2].

[2] For Q = prime after P,... do:

For each prime ideal Q' of K with

Q' | Q do:

if GOOD__REDUCTION(F,K,Q')

then do:

[2.1] NON__Q__PART := FINITE__TORSION(F,K,Q').

[2.2] Go to [3]

[3] ANSWER1 := OCCURRENCES(Q,NON__P__PART) * NON__Q__PART

OCCURRENCES(Q,NON__Q__PART)

ANSWER2 := OCCURRENCES(P,NON__Q__PART) * NON__P__PART

OCCURRENCES(P,NON__P__PART)

Return gcd(ANSWER1,ANSWER2).

Furthermore this curve has 7 points on it, viz. $(0,\pm 1)$, $(1,\pm 1)$, $(2,\pm 1)$ and the point at infinity.

Computational Considerations

This section will describe some of the problems involved in the implementation of this "modular" algorithm, and outline some solutions. Since I do not have anything approaching a complete implementation of the work described in this chapter, this section may well not be complete,.

We know that the residue class field is finite of characteristic p, and it is in fact obtained from the integers of $\boldsymbol{Q}(I)$ by identifying elements if their difference lies in the prime ideal. Since p lies in the prime ideal, we need only consider those elements of $\boldsymbol{Z}[I]$ with integer coefficients between 0 and p, i.e. a finite set. Hence we can construct the residue class field by enumeration, though this may not be efficient in all cases. Since the field is of characteristic p, it contains a subfield isomorphic to the integers modulo p. The field is then an extension of the field of integers modulo p, and computation over such fields has been studied by Mignotte (1976).

We need to determine not only irreducibility but also absolute irreducibility in a residue class field in order to test for good reduction. We can factorise univariate polynomials in the subfield which corresponds to the integers modulo p by Berlekamp's Algorithm (Zimmer, 1972), but this is not sufficient. For example, in the residue field of 5 in $\boldsymbol{Q}(\sqrt{2})$ (see the next section for a detailed discussion of this example) the polynomial $X^2 + 2 = 0$ factorises, whereas it does not factorise in the integers modulo 5. Hence, even for polynomials defined over the subfield, reduction to the subfield is not an adequate factorisation strategy.

Berlekamp (1970) presents an algorithm for reducing the problem of factoring univariate polynomials over a field of prime power order to that of factoring over the prime field, but this is not an easy process* and I have not yet implemented it. Alternatively, using the techniques of Trager (1976) generalised (where applicable) to finite fields and to multi-variate expressions of algebraic extensions (see Appendix 3 for details and algor-

* Mignotte (1976) describes the process as "complexe mais efficace".

ithms), we can reduce the problem to that of factoring a much larger polynomial over the field of p elements, which can then be solved readily. Once we have a univariate factorisation, we can grow it up to a multi-variate factorisation in almost# all cases,using essentially the same techniques as are used in p-adic factorising techniques (Wang, 1978 or Yun 1973 and 1976).

An Example over Algebraic Fields

Now let us consider the curve $Y^2 = X^3 + 8$, with differential $1/Y$. This does not have good reduction modulo 2 (since it becomes $Y^2 = X^3$, which is no longer of genus 1) or 3 (since it becomes $Y^2 = (X-1)^3$, also no longer of genus 1). It has good reduction modulo both 5 and 7, and the non-p-parts of the torsion are 6 and 12 respectively. Hence the torsion is at most 6 (in fact, the curve has no torsion over $\boldsymbol{Q}$, but it does have one generator of infinite order (Birch & Swinnerton-Dyer, 1963), which has to map into a torsion divisor over a finite field).

Over the field $\boldsymbol{Q}(\sqrt{2})$, the situation is slightly different. 2 and 3 are still primes of bad reduction (and, more generally, extending the ground field does not get rid of any primes of bad reduction). 5 is a prime in this field; and the residue class field has 25 elements, which we can represent by $\{(i + j\sqrt{2}), 0 \leq i,j \leq 4\}$ The curve has 36 points of finite order in this residue class field.

In $\boldsymbol{Q}(\sqrt{2})$, the rational prime 7 splits into the product of 2 prime ideals, $< 7,4 + \sqrt{2} >$ and $< 7,3 + \sqrt{2} >$. The first of these is a prime of good reduction, and the residue class field has 7 elements, which we can equate with the numbers 0 to 6 modulo 7. The curve has 12 points of finite order over this field (exactly as in the case of $\boldsymbol{Q}$, since the two residue class fields are isomorphic and the isomorphism preserves the curve). Thus

---

it may be that all evaluations of the other variables in a multivariate factorisation are "unlucky" in the sense that the image ceases to be square-free. However, this can only happen for finitely many primes, so at the worst we can afford to treat this as a case of bad reduction and try a different prime.

with 36 for the non-5-torsion and 12 for the non-7-torsion, we are led to a bound of 12 for the torsion of the curve over the rationals as extended by the square root of 2. Although I know of no easy[$] way of discovering the torsion of an elliptic curve over fields other than the rationals, this curve does have a point of order 6 over the field $Q(\sqrt{2})$, viz $X = 4$, $Y = 6\sqrt{2}$, and furthermore twice this point (which is therefore of order 3) crops up in some comparatively simple integrals (see the footnote to Appendix 2 example 3).

Note that, though our point is of order 3 (or possibly 6) we have a computed bound of 12 by the second technique, but 252 by the first technique, so that this technique, while undoubtedly workable and effective in the mathematical sense of the word, has limitations for practical computation. One reason for this is that there are likely to be infinite order divisors as well, and these will tend to map into points of high (but finite) order in the torsion group corresponding to a good reduction, thus giving us unnecessarily high estimates.

Computationally, this is especially embarrassing because we only need the bound when there are divisors of infinite order, since if the divisor is of finite order we will find the order with or without a bound.

$ As far as I know, the algorithms of Birch & Swinnerton-Dyer (1963) have not been implemented over fields other than the rationals.