# Coleman Extensions Examples

Sachi Hashimoto and Travis Morrison

November 29, 2019

**Abstract**

This shows some example calculations using the Coleman Extensions Magma library for computing Coleman integrals to number field points and run effective Chabauty for curves with known infinite order divisors.

## 1 A curve without an infinite order rational point

To load the code you must load both the Coleman library of Balakrishnan and Tuitman as well as the Coleman Extensions library:

```
> load "coleman.m";
> load "colemanextensions.m";
```

In this example we consider the Chabauty-Coleman calculation for the curve with the plane model $y^3 = x^4 + 7x^3 + 3x^2 - x$. First we compute the rank using `RankBounds`.

```
> f:=x^4 + 7*x^3 + 3*x^2 - x;
> RankBounds(f,3:ReturnGenerators:=true)
1 1 [*
    x^3 + 3*x^2 + 3*x - 1,
    x^3 + 3/4*x^2 + 3/16*x - 1/64,
    x,
    x^3 + 7*x^2 + 3*x + 1,
    x^3 + 3*x^2 - 2*x + 1,
    x^3 - x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1,
    x^3 + 6*x^2 + 3*x - 1
*]
```

Since $x$ represents a torsion divisor, we have to choose an infinite order divisor which is a sum of points defined over a degree three extension of $\mathbb{Q}$. Let $g = x^3 + 6x^2 + 3x - 1$. We will run Chabauty-Coleman using this divisor. To choose an appropriate prime of which the roots of $g$ are defined we ask for the first split prime:

```
> first_split_prime(g,33264270000);
17
```

where 33264270000 is the discriminant of $f$.

We can construct the splitting field of $g$ and set the divisor corresponding to $g$ manually by computing the $b_0^i$ vector $[1, y, y^2]$ using the following code. We start by defining the Coleman data from the Balakrishnan and Tuitman code, then we compute the points to set in the divisor.

```
> p:=17;
> N:=10;
> data:=coleman_data(y^3-f,p,N);
> L, roots:=SplittingField(x^3 + 6*x^2 + 3*x - 1);
> bvals:=[];
> for r in roots do
for> bool,b:=IsPower(Evaluate(f,r),3);
for> bvals:=Append(bvals,[1,b,b^2]);
for> end for;
> bvals;
[
    [
        1,
        r.1,
        r.1^2
    ],
    [
        1,
        r.1^2 + 5*r.1 - 2,
        -6*r.1^2 - 31*r.1 + 8
    ],
    [
        1,
        -r.1^2 - 6*r.1 - 4,
        5*r.1^2 + 31*r.1 + 22
    ]
]
> OL:=MaximalOrder(L);
> p_splitting:=Factorization(p*OL);
> prime_over_p:=p_splitting[1][1];
> inf:=false;
> divisor:=true;
> D:=set_bad_div(roots,bvals,inf,data,L,prime_over_p,divisor);
> D;
rec<recformat<x, b, inf, xt, bt, index, divisor, L> |
    x :=[ -573834685435 + O(17^10), -66446288765 + O(17^10), 640280974194 +
        O(17^10) ],
    b := [
        [ 1 + O(17^10), -573834685435 + O(17^10), 786733237963 + O(17^10) ],
        [ 1 + O(17^10), -66446288765 + O(17^10), 972512418021 + O(17^10) ],
        [ 1 + O(17^10), 640280974194 + O(17^10), 256748244495 + O(17^10) ]
    ],
    inf := false,
    divisor := true,
    L := Number Field with defining polynomial x^3 + 6*x^2 + 3*x - 1 over the
        Rational Field>
```

To automatically find rational points up to a bound on $f$ and set $\mathrm{Div}(g) - 3\infty$ as a divisor on $f$ without going through the steps above we can simply define the usual Coleman data from Balakrishnan and Tuitman and then run the `Q_divs` function:

```
> g:=x^3 + 6*x^2 + 3*x - 1;
> bound:=1000;
> Qdivs:=Q_divs(data, bound, g);
```

Then some points of `Qdivs` are rational points, for example

```
> Qdivs[1];
```

```
rec<recformat<x, b, inf, xt, bt, index, divisor, L> |
    x := [ 0 ],
    b := [
        [ 1 + O(17^10), 0, 0 ]
    ],
    inf := false,
    divisor := false,
    L := Rational Field>
```

corresponds to the rational point $[0, 0, 1]$. But,

```
> Qdivs[3];
rec<recformat<x, b, inf, xt, bt, index, divisor, L> |
    x := [ -573834685435 + O(17^10), 640280974194 + O(17^10), -66446288765 +
        O(17^10) ],
    b := [
        [ 1 + O(17^10), -573834685435 + O(17^10), 786733237963 + O(17^10) ],
        [ 1 + O(17^10), 640280974194 + O(17^10), 256748244495 + O(17^10) ],
        [ 1 + O(17^10), -66446288765 + O(17^10), 972512418021 + O(17^10) ]
    ],
    inf := false,
    divisor := true,
    L := Number Field with defining polynomial x^3 + 6*x^2 + 3*x - 1 over the
        Rational Field>
```

corresponds to the divisor $Div(g) - 3\infty$ defined above manually. We can run effective Chabauty using this known infinite order divisor to compute the vanishing differential using the function

```
> L, v:=effective_chabauty_with_Qdiv(data:Qpoints:=Qdivs,e:=150);
```

As in the code of Balakrishnan and Tuitman this returns a list of candidate points and annihilating differentials:

```
> L;
[
    rec<recformat<x, b, inf, xt, bt, index> |
        x := O(17^6),
        b := [ 1 + O(17^10), O(17^4), O(17^2) ],
        inf := true>,
    rec<recformat<x, b, inf, xt, bt, index> |
        x := O(17^10),
        b := [ 1 + O(17^10), O(17^6), O(17^10) ],
        inf := false>,
    rec<recformat<x, b, inf, xt, bt, index> |
        x := 539378708825 + O(17^10),
        b := [ 1 + O(17^10), O(17^6), O(17^10) ],
        inf := false>,
    rec<recformat<x, b, inf, xt, bt, index> |
        x := 743576159772 + O(17^10),
        b := [ 1 + O(17^10), O(17^6), O(17^10) ],
        inf := false>,
    rec<recformat<x, b, inf, xt, bt, index> |
        x := 733039031845 + O(17^10),
        b := [ 1 + O(17^10), O(17^6), O(17^10) ],
        inf := false>,
    rec<recformat<x, b, inf, xt, bt, index> |
        x := 1 + O(17^6),
        b := [ 1 + O(17^10), 10462466 + O(17^6), -4761912 + O(17^6) ],
```

```
        inf := false>
]
> v;
[
    [ 1 + O(17^7), O(17^7), O(17^7) ],
    [ O(17^7), 1 + O(17^7), O(17^7) ],
    [ O(17^7), O(17^7), 1 + O(17^7) ]
]
```

The vanishing differentials are computed by finding regular one-forms which vanishes on the integral over $D$ as well as all known rational points; this integral over $D$ is computed by summing over the integrals to all points in the support of the divisor $Div(g)$ from the basepoint $\infty$. In this case we get an unusually large space of vanishing differentials for a rank 1 curve.

# 2 An endomorphism example

One can also use the code to compute integrals to number field points, for example to check if a point is torsion. Here we exhibit another example of points on a Picard curve which are not explained by torsion or linearity, but instead come from the presence of extra endomorphisms of the curve. The curve is a rank 2 Picard curve given by the plane model $y^3 - (x^4 - 2)$ and we can run Chabauty as in the example above at $p = 5$ to get 5-adic points which can be recognized as global points $Q_1 = [i, -1, 1]$ and $Q_2 = [-i, -1, 1]$.

There is a function which computes basis integrals to number field points, allowing one to quickly test relations with Coleman integrals:

```
> data:=coleman_data(y^3 - (x^4 - 2), 5, 10);
> K<i>:=NumberField(x^2+1);
> Kpoints:=[[ i,-1,1], [-i,-1,1]];
> ratpts:=[];
> e:=20;
> compute_integrals(data, ratpts, Kpoints, e);
[
    (-57*5^4 + O(5^7) -8*5^2 + O(5^7) -7764*5 + O(5^7) 7398*5 + O(5^7) -4876 +
        O(5^7) -21441 + O(5^7)),
    (57*5^4 + O(5^7) -8*5^2 + O(5^7) 7764*5 + O(5^7) -7398*5 + O(5^7) -4876 +
        O(5^7) 21441 + O(5^7))
]
```

For example, this shows that $Q_1 + Q_2 - 2\infty$ is torsion while $Q_i - \infty$ is not torsion. More generally, one can use `compute_integrals` to easily compute all basis integrals between an enumerated sequence of rational points (replacing the empty sequence) and an enumerated sequence of points over a number field $K$.