ORBIFOLD POINTS ON PRYM-TEICHMÜLLER CURVES IN GENUS FOUR – PARI FILE

DAVID TORRES-TEIGELL AND JONATHAN ZACHHUBER

Abstract. The following PARI code computes the orbifold type (i.e. number and type of orbifold points, cusps and Euler characteristic) of Prym-Teichmüller curves in $\mathcal{M}_4$.

Contents

1. Introduction

The following PARI code calculates the number and type of orbifold points on Prym-Teichmüller curves W_D in $\mathcal{M}_4$, see [TTZ16] for details. By implementing the algorithm of Lanneau-Nguyen [LN14] for the number of cusps and using Möller's formula for the Euler characteristic [Möl14] (see e.g. [Bai07] for details how to compute this), we can give the complete orbifold classification for all nonsquare D.

As neither of the authors can honestly call himself a programmer, we hold no claims for efficiency or "clean" programming. Moreover, we are grateful for any comments or feedback.

While there can be no doubt that the efficiency can be improved, the programme runs in a reasonable time for D up to around 10000.

The code should work with [Par] and (hopefully) all newer versions.

2. Usage

To extract the PARI file (and this `pdf` file), run

```
pdflatex ptmcorbiptsg4.dtx
bibtex ptmcorbiptsg4
pdflatex ptmcorbiptsg4.dtx
```

and then in PARI, e.g.

```
? \r ptmcorbiptsg4.pari
? printsignatureuntil(50)
```

```
Signature for W_D(6) for non-square discriminant up to 50
D g p2 p3 c chi
5 0 0 1 1 -7/15
8 0 1 1 2 -7/6
12 0 1 0 3 -7/3
13 0 0 2 3 -7/3
17 0 0 1 6 -14/3
20 0 2 1 5 -14/3
21 1 0 1 4 -14/3
24 1 2 0 6 -7
28 1 2 2 7 -28/3
29 1 0 3 5 -7
32 1 2 2 7 -28/3
33 2 0 0 12 -14
37 1 0 4 9 -35/3
40 2 2 2 12 -49/3
41 3 0 1 14 -56/3
44 3 4 2 9 -49/3
45 4 0 0 8 -14
48 4 2 1 11 -56/3
? ##
  ***   last result computed in 3 ms.
```

The default value for `printsignatureuntil` is 100.

To obtain the signature for a single D:

```
? printsignature(10540)
10540 34838 48 24 594 -70308
? ##
  ***   last result computed in 5 ms.
```

3. Implementation

We assume always that D is a (non-square!) quadratic discriminant, i.e. 0 or 1 mod 4.

3.1. **Cusps.** We begin by computing the number of cusps, $C(W_D)$ of W_D. By [LN14, Proposition D.4], the number of cusps in genus 4 is the same as those obtained by McMullen for the curves in genus 2, namely

$$C(W_D) = \#\left\{(a,b,c,e) \in \mathbb{Z}^4 : \begin{array}{l} b > 0, c > 0, 0 \leq a < \gcd(b,c), c+e < b, \\ \gcd(a,b,c,e) = 1 \text{ and } D = e^2 + 4bc \end{array}\right\}.$$

To enumerate these prototypes, we loop over possible values of e and b (divisors of $(D-e^2)/4$). To count the possible values of a, we consider $\gcd(b,c)$. However, we consider only those a such that $\gcd(a,b,c,e) = 1$. This is achieved by the function `magic`.

```
magic(n,l)={\\ think of better name
  local(r);
  r=0;
  for(a=1,n%l-1,if(gcd(a,l)==1,r+=1));
  return(n\l*eulerphi(l)+r);
}

cusps(D)={\\ direct gcd computation of cusps
```

```
  local(n,c,m,f);
  n=0;
  if ( (D%4!=0) && (D%4!=1), print("Invalid Discriminant!"); return());
  forstep (e=D%2,sqrt(D),2,
    fordiv((D-e^2)/4,b,
      c=(D-e^2)/(4*b);
      m=gcd(b,c);
      f=gcd(m,e);
      if(e==0,
              n+=magic(m,f),
              n+=2*magic(m,f));
    );
  );
  return (n/2); \\ c+e<b
}
```

3.2. **Euler characteristic.** By [Möl14, Thm 4.1], for nonsquare D, the Euler characteristic is given by

$$\chi(W_D) = -7\chi(X_D),$$

where X_D is the Hilbert modular surface parametrising abelian twofolds admitting proper real multiplication by the order $\mathcal{O}_D$.

```
geteulercharg4(D) = { \\ calculate chi(W_D(6) (genus 4)
  return(-7*geteulercharX(D));
}
```

The Euler characteristic of the "normal" Hilbert modular surface X_D is known. Siegel and, later, Cohen gave explicit expressions for it in terms of zeta functions and divisor sums, see [Bai07, Thm 2.12, 2.15 and 2.16] for a detailed discussion. First, we must calculate certain divisor sums

$$H(D) = -\frac{1}{5}\sum_{e\equiv D(2)} \sigma_1\left(\frac{D-e^2}{4}\right).$$

Here:

```
H (D) = { \\ calculate H(2,D)
  local(HH,e);
  HH=0;
  forstep (e=D%2,sqrt(D),2,
    if (e==0,
      HH+=mysigma((D-e^2)/4),
      HH+=2*mysigma((D-e^2)/4));
  );
  HH/=-5;
  if (issquare(D),HH-=D/10);
  return(HH);
}
```

Note that we slightly modified PARI's sigma function to accept 0:

```
mysigma (n) = {\\ return sigma_1 for >0, -1/24 for 0
  if (n==0, return (-1/24));
  return(sigma(n));
}
```

By exploiting the fact that ζ_D, the Dedekind zeta function of $\mathbb{Q}(\sqrt{D})$ satisfies

$$\zeta_D(-1) = -\frac{1}{12}H(D),$$

we can finally calculate the Euler characteristic:

```
geteulercharX(D) = { \\ calculate chi(X_D) for D
  local (DD,f,z,chi);
  z=coredisc(D,1);
  DD=z[1]; f=z[2]; \\ split into fundamental discriminant and square
  chi=sumdiv(f,r,kronecker(DD,r)*moebius(r)/r^2);
  chi*=-H(DD)/12;
  chi*=2*f^3;
  return(chi);
}
```

3.3. **Orbifold points.** By [TTZ16, Theorem 1.1], W_D has orbifold points of order 2 or 3 unless $D = 5, 12$. Moreover,

$$e_2(D) = \begin{cases} 0\,, & \text{if } D \text{ is odd,} \\ h(-D) + h(-\frac{D}{4})\,, & \text{if } D \equiv 12 \mod 16, \\ h(-D)\,, & \text{if } D \equiv 0, 4, 8 \mod 16, \end{cases}$$

where $h(-D)$ is the class number of the (imaginary quadratic) order $\mathcal{O}_{-D}$. It is well-known that class numbers can be computed by enumerating quadratic forms. The following algorithm is given in [Coh93, Alg. 5.3.5].

```
imagh(C)={ \\ class number for imaginary quadratic
       \\ order O_C. C discriminant, C < 0 !!!
       \\ counting reduced quadratic forms (cf. Cohen, Alg. 5.3.5)
  local(h,q);
  h=0;
  forstep (b=C%2,sqrt(abs(C)/3),2,
    q=(b^2-C)/4;
    for(a=b,sqrt(q),
      if(a==0,next());
      if(q%a==0 && gcd(a,gcd(b,q/a))==1,
          if(a==b || a^2==q || b==0,h+=1,h+=2));
    );
  );
  return(h);
}
```

Note that we disregard extra units in $\mathcal{O}_{-C}$, as this is only relevant for $D = 12$ and is actually a point of order 6, see [TTZ16] for details. Computing the number of points of order 2 is now easy.

```
e2(D)={
  local(e);
  e=0;
  if(D%4==1,return(e));
  e+=imagh(-D);
  if(D/4%4==3, return(e+imagh(-D/4)), return(e));
}
```

The points of order 3 are given by integral solutions of a ternary quadratic form. More precisely:

$$e_3(D) = \#\{a, i, j \in \mathbb{Z} : a^2 + 3j^2 + (2i - j)^2 = D,\ \gcd(a, i, j) = 1\}/12.$$

To enumerate these solutions, we simply loop over a and j, checking that the square root of $q = D - 4a^2 - 3j^2$ is an even integer. Finally, we loop only through the positive values of a, j, i, hence we must count with multiplicities whenever they are non-zero.

```
e3(D)={
  local(e,s,i,q);
  s=0;
  forstep(a=D%2,sqrt(D),2,
    for(j=0,sqrt((D-a^2)/3),
      e=0;
      q=D-a^2-3*j^2;
      if(issquare(q),
        i=sqrtint(q)+j;
        if(i%2==0,
          if(gcd([a,j,i/2])==1,
            if(i*j==0,
              if(i==j, e+=1,e+=2),
              if(i==j, e+=2,e+=4)))));
      if(a==0, s+=e, s+=2*e);
    );
  );
  return(s/12);
}
```

3.4. **Orbifold signature.** We combine all this data to give the orbifold signature, using the formula

$$2 - 2g = \chi + C + \sum_d h_d \left(1 - \frac{1}{d}\right),$$

where g denotes the genus of W_D, χ the Euler characteristic, C the number of cusps and h_d the number of orbifold points of order d. Note that the cases $D = 5$ and $D = 12$ are hard-coded and the points of order 5 and 6 are ommitted, see [TTZ16] for details.

```
getsignatureg4(D) = {
  local(chi,c,p2,p3,g);
  if(D==5,return([5,0,0,1,1,-7/15]));
  if(D==12,return([12,0,1,0,3,-7/3]));
  if ( (D%4!=0) && (D%4!=1), print("Invalid Discriminant!"); return());
  if ( issquare(D), print("Square Discriminants not permitted!"); return());
  chi=geteulercharg4(D);
  c=cusps(D);
  p2=e2(D);
  p3=e3(D);
  g=1-chi/2-c/2-p2/4-p3/3;
  return([D,g,p2,p3,c,chi]);
}
```

The following macro provides a frontend, printing the signature as a row of a table.

```
printsignature(D)={
```

```
  local(v);
  v=getsignatureg4(D);
  for(i=1,6,print1(v[i],"\t"));print();
}
```

Finally, we provide a macro printing a table of the orbifold signature of W_D for D up to n (default $n = 100$).

```
printsignatureuntil(n=100)={
  print("Signature for W_D(6) for non-square discriminant up to ",n);
  print("D\tg\tp2\tp3\tc\tchi");
  forstep(D=5,n,[3,1],
    if(issquare(D),next());
    printsignature(D);
  );
}
```

References

[Bai07] Matt Bainbridge. “Euler characteristics of Teichmüller curves in genus two”. In: *Geom. Topol.* 11 (2007), pp. 1887–2073.

[Coh93] Henri Cohen. *A Course in Computational Algebraic Number Theory.* Graduate Texts in Mathematics 138. Springer, 1993.

[LN14] Erwan Lanneau and Duc-Manh Nguyen. “Teichmüller curves generated by Weierstrass Prym eigenforms in genus 3 and genus 4”. In: *J. Topol.* 7.2 (2014), pp. 475–522.

[Möl14] Martin Möller. “Prym covers, theta functions and Kobayashi geodesics in Hilbert modular surfaces”. In: *Amer. Journal. of Math.* 135 (2014), pp. 995–1022.

[Par] *PARI/GP version* `2.3.5`. The PARI Group. Bordeaux, 2010. URL: http://pari.math.u-bordeaux.fr/.

[TTZ16] David Torres-Teigell and Jonathan Zachhuber. “Orbifold Points on Prym-Teichmüller Curves in Genus Four”. In: (2016). arXiv: 1609.00144 [math.AG].

Fachrichtung Mathematik, Universität des Saarlandes, Campus E24, 66123 Saarbrücken, Germany

E-mail address: torres@math.uni-sb.de

FB 12 – Institut für Mathematik, Johann Wolfgang Goethe-Universität, Robert-Mayer-Str. 6–8, D-60325 Frankfurt am Main

E-mail address: zachhuber@math.uni-frankfurt.de