IVRG WEEK 5
FROM GROUPS TO EQUATIONS

DAVID SWINARSKI

Setup

First, make sure the file "DecomposeGAction.txt" is saved in your home directory. We will load it later. It contains a `Magma` program I wrote for you to use.

I learned a way to save your `Magma` session, but it works best if you run these commands at the beginning. If you type

```
> SetLogFile("ivrg9-13-2010.txt");
```

then `Magma` will start a log file with that string as its filename. At the end of the day, we'll run

```
> UnsetLogFile();
```

1. Introduction

Last time, we defined studied the curve C given by the points $[x : y : z] \in \mathbb{P}^2$ such that $x^4+y^4+z^4 = 0$. We used `Magma` to compute its automorphism group. Over $\mathbb{Q}$, we found $\mathrm{Aut}(C) \cong S_4$. Over $\mathbb{Q}[i]$, we found $\mathrm{Aut}(C) \cong \langle 96, 64\rangle$.

This week we will reverse the process. Suppose you know that the group $\langle 96, 64\rangle$ is the automorphism group of a curve of genus 3. How can we find polynomial equations for C?

This is the goal for the semester. As mentioned before, we have lists and lists of groups which are the automorphism groups of curves. I know how to compute lots of things given equations for a curve. So given a group, I want to be able to find equations.

This week we'll consider the example $\langle 96, 64\rangle$ in detail. Again, we already know the answer is $x^4 + y^4 + z^4 = 0$.

Step 1: Determine how many and what kind of polynomials you are looking for

There are two types of curves of genus 3: *hyperelliptic* and *non-hyperelliptic*.

Definition 1.1 *A curve C is* hyperelliptic *if there exists a morphism $f : C \to \mathbb{P}^1$ such that f is $2:1$ at all but finitely many points.*

Every hyperelliptic curve has a nontrivial automorphism called the hyperelliptic involution*, which we denote σ. If P and Q are two points such that $f(P) = f(Q)$, then we define σ by $\sigma(P) = Q$ and $\sigma(Q) = P$. Then $\sigma^2 = \mathrm{Id}$.*

If C is hyperelliptic and genus 3, then we seek an equation of the form $y^2 = f(x)$ where f has degree 7 or 8.

If C is non-hyperelliptic and genus 3, then we seek an equation of the form $F(x, y, z) = 0$, where F is homogeneous of degree 4.

So we want to know: is the curve determined by the group $\langle 96, 64\rangle$ hyperelliptic or non-hyperelliptic?

In our situation there is an easy test for hyperellipticity that we can apply. First another definition:

Date: September 13, 2010.

Definition 1.2 *The* center *of a group G is the set of elements which commute with all elements of G:*

$$\text{Center}(G) := \{g \in G : gh = hg \forall h \in G\}$$

Note from Dave: the following proposition was corrected on Oct. 1, 2010.

Proposition 1.3 ([2, p. 736]) *If C is hyperelliptic, then there exists an element $g \in \text{Aut}(C)$ such that $g^2 = \text{Id}$ and $g \in \text{Center}(\text{Aut}(C))$.*

Note: Lewittes proves that $\langle \sigma \rangle$ is a normal subgroup of $\text{Aut}(C)$, where σ is the hyperelliptic involution. But you can easily show that a normal subgroup of order 2 is central.

Thus, we can compute $\text{Center}(\text{Aut}(C))$ in `Magma` and see if it contains any elements of order 2. If it does not, then we know C is not hyperelliptic. (I'll show you a way to verify that a curve is hyperelliptic in a few weeks).

```
> SmallGroupDatabase();           //loads the small group database
> H:=SmallGroup(96,64);           //loads <96,64>
> Center(H);                      //computes the center of <96,64>
GrpPC of order 1
PC-Relations:
> Order(Center(H));
1
```

Notice the output from `Center(H)` already tells us that the center has order 1. That is, the center consists of only one element, the identity (since clearly Id commutes with any element of G). Therefore, we see that in $\langle 96, 64 \rangle$ there are no order two elements in the center. We conclude that our curve must be non-hyperelliptic, and therefore given by an equation of the form $F(x, y, z) = 0$, where F is homogeneous of degree 4 (which fits with the answer, which we secretly know).

Step 2: Find out how $\text{Aut}(C)$ acts on the variables

We know $\langle 96, 64 \rangle$ is supposed to act on x, y, and z, but how?

For genus $g = 3, 4, 5$ these matrices are given in the literature. (For curves of genus $g \geq 6$, we'll have to do a little more work.) For instance, for genus 3, see [1, Proposition 1.3, p. 281]. This proposition gives matrix generators for automorphism groups whose orders are of the form $2^p 3^q$ (and 96 is of this form). In the last line, we see $G(96) = \langle B(1, i, i), C(1, 1, 1) \rangle$. Refer to page 279 where the notation $B(1, i, i)$ and $C(1, 1, 1)$ is explained. We see that Kuribayashi and Kuribayashi are asserting that the action of $\langle 96, 64 \rangle$ on the variables x, y, and z is given by two matrices, which I call A and B below:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ i & 0 & 0 \\ 0 & 0 & i \end{pmatrix}, B = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

We want to put these in `Magma`. Notice that we will want to be able to use the complex number i, and $i^4 = 1$. Recall from last week that anytime we want to use roots of unity, we want to work over a *cyclotomic field*.

```
> K<i>:=CyclotomicField(4);              //create the field Q[i], and use the
                                         //letter i for the new variable
> GL3K:=GeneralLinearGroup(3,K);         //create the group of 3x3 invertible
                                         //matrices with coefficients in the field K
> A:=elt<GL3K | 0,1,0, i,0,0, 0,0,i>;    //enter the matrix A
> A;                                     //peek at A to check it
[0 1 0]
[i 0 0]
[0 0 i]
```

```
> B:=elt<GL3K | 0,1,0, 0,0,1, 1,0,0>;     //enter the matrix B
> B;
[0 1 0]
[0 0 1]
[1 0 0]
> G:=sub<GL3K | A,B>;
> Order(G);                               //compute the order of G to check it
96
> IdentifyGroup(G);                       //Check that the group G really is <96,64>
<96, 64>
```

We have now loaded $\langle 96, 64\rangle$ in Magma. In the next three sections we will outline three different ways to find $F(x, y, z)$.

Finding equations, method one: Computing invariants with Magma

Definition 1.4 *Suppose a group G acts on a set X. If x has the property that $gx = x$ for all $g \in G$, then we say x is G-invariant.*

From the previous section, we know how $G \cong \langle 96, 64\rangle$ acts on x, y, and z. That means G acts on monomials of every degree. For example, we can define the action of g on x^2 to be $gx^2 := (gx)(gx)$. More generally, $gx^a y^b z^c = (gx)^a (gy)^b (gz)^c$. Then we can extend the G-action from monomials to polynomials.

If there is a G-invariant degree 4 homogeneous polynomial, then that could be the answer we seek. Let's see if there are any. In Magma:

```
> R:=InvariantRing(G);    //Compute the invariant ring
> R;                      //Peeking at R still doesn't display the invariant polynomials
Invariant Ring
Group:
    MatrixGroup(3, K) of order 96 = 2^5 * 3
    Generators:
        [0 1 0]
        [i 0 0]
        [0 0 i]

        [0 1 0]
        [0 0 1]
        [1 0 0]
Coefficient ring:
    Cyclotomic Field of order 4 and degree 2
> PrimaryInvariants(R);   //Shows us the generators of the invariant ring
[
    x1^4 + x2^4 + x3^4,
    x1^2*x2^2*x3^2,
    x1^8 + x2^8 + x3^8
]
```

We see that there is one homogeneous degree 4 invariant polynomial. Magmaused the variables x_1, x_2, and x_3 for it, but we can easily see that this is the same as $x^4 + y^4 + z^4$, as we expected.

This method is easy to try, but it may not always work. Here are two possible reasons it could fail:

(1) There might not be enough invariant polynomials in the degrees you need.

(2) I don't know what algorithm Magmauses, so it could be possible that the calculation will be too slow to complete.

What is special

Finding equations, method two: Computing invariants by hand

In this example, we can find the invariant $x^4 + y^4 + z^4$ by hand.

Let's start by examining the structure of G a little more closely. I don't know a built-in Magma command to see the 96 matrices in G, so here is a way we can do this ourselves. Note: if you think of G as a list of 96 matrices stored in order, the numbering map is the map which takes a matrix A to its position on the list. Also, note that when we begin the for loop, we don't use a semicolon at the end of the line; that tells Magma the commands are continued on the next line.

```
> f:=Inverse(NumberingMap(G));
> for i:=1 to 96 do
for> print(f(i));
for> print("");
for> end for;
[1 0 0]
[0 1 0]
[0 0 1]

.
.
.

[ 0  0 -i]
[ 0  i  0]
[-1  0  0]

[ 0  0  1]
[-1  0  0]
[ 0 -1  0]
```

Scanning over this list, we recognize a few of these matrices. For instance, the list includes

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix},$$

which we recognize as permutations of the variables x, y, and z. The two matrices above generate a copy of S_3 inside G.

I want to find G-invariant polynomials. I'll do it in two stages. First, I'll find some S_3-invariant polynomials. Then I'll check to see if any of these are actually G-invariant.

One way to find some S_3-invariant polynomials is to take the sum of the monomials in an orbit. For instance, we can consider how S_3 acts on the monomials $\{x^4, x^3y, x^3z, x^2y^2, x^2yz, x^2z^2, xy^3, xy^2z, xyz^2, xz^3, y^4, y^3z, y^2z^2, yz^3, z^4\}$. Start with x^4. Under the permutations of the variables x, y, z, the monomial x^4 might get mapped to itself, to y^4, or to z^4. Then $\{x^4, y^4, z^4\}$ is one S_3-orbit, and $x^4 + y^4 + z^4$ is an S_3-invariant polynomial.

We can continue splitting up the remaining 12 monomials into orbits. We get $\{x^3y, x^3z, y^3z, y^3x, xz^3, yz^3\}$ (the orbit of monomials of the form "one variable cubed times a variable to the first power"), $\{x^2y^2, x^2z^2, y^2z^2\}$, $\{x^2yz, xy^2z, xyz^2\}$. Thus we have found four S_3-invariant polynomials as follows:

$$\begin{aligned} F_1(x,y,z) &= x^4+y^4+z^4 \\ F_2(x,y,z) &= x^3y+x^3z+y^3z+y^3x+xz^3+yz^3 \\ F_3(x,y,z) &= x^2y^2+x^2z^2+y^2z^2 \\ F_4(x,y,z) &= x^2yz+xy^2z+xyz^2 \end{aligned}$$

(Question: are there any other degree 4 S_3-invariant polynomials?)

Now we can ask whether any of the four S_3-invariant polynomials are actually G-invariant. For each polynomial $F_i(x,y,z)$ on the list above, we can test whether $AF_i = F_i$ and $BF_i = F_i$. Out of these four polynomials, we see that only $F_1(x,y,z)$ is G-invariant. Again, we have found the answer.

I like this method because we got to think a little bit, but it is easy to see some obstacles to applying it in general. This method could break down at the beginning: you might stare at the list of matrices and not see any which generate a meaningful subgroup of G. Then of course the broader problem is that there might not be enough invariant polynomials anyway...

Finding equations, method three: Decomposing the action of G on polynomials

The third method is to decompose the G-action on polynomials. This is more general than finding invariants. What does it mean to decompose a G-action on polynomials? Let me illustrate with a slightly simpler example.

Let's look at how the matrix A acts on degree 2 monomials. Recall that

$$A = \begin{pmatrix} 0 & 1 & 0 \\ i & 0 & 0 \\ 0 & 0 & i \end{pmatrix},$$

and so the associated linear transformation is (read off the columns of the matrix above)

$$\begin{aligned} x &\mapsto iy \\ y &\mapsto x \\ z &\mapsto iz \end{aligned}$$

Then we can compute the action on degree 2 monomials. We get:

$$\begin{aligned} x^2 &\mapsto (iy)(iy) = -y^2 \\ y^2 &\mapsto (x)(x) = x^2 \\ z^2 &\mapsto (iz)(iz) = -z^2 \\ xy &\mapsto (iy)(x) = ixy \\ xz &\mapsto (iy)(iz) = -yz \\ yz &\mapsto (x)(iz) = ixz \end{aligned}$$

Then the matrix of this linear transformation with respect to the ordered basis $\{x^2, y^2, z^2, xy, xz, yz\}$ is

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & i \\ 0 & 0 & 0 & 0 & -1 & 0 \end{pmatrix}.$$

We see that the upper right 3×3 block is zero, and the lower left 3×3 block is zero.

We do a similar calculation for the matrix B and find the matrix below for its action on degree 2 monomials:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

We see that the upper right 3×3 block is zero, and the lower left 3×3 block is zero.

Since A and B both have this block form, and G is generated by A and B, we can conclude that for any matrix in G, the associated matrix for the action on degree 2 monomials has this block form. That is, the two vector subspaces $\mathrm{Span}\{x^2, y^2, z^2\}$ and $\mathrm{Span}\{xy, xz, yz\}$ do not get "mixed" under the action of G.

That is what the function `DecomposeGAction` computes. We run this example in `Magma`:

```
> load "DecomposeGAction.txt";          //Load my program.  One of the only
Loading "DecomposeGAction.txt"          //Magma commands that starts lowercase.
> S<x,y,z>:=PolynomialRing(K,3);        //Tell Magma what variables you want it
                                        //to use for polynomials.
> DecomposeGAction(G,S,2);
[
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 5,
        Dimension := 3,
        Elements := [
            x^2,
            y^2,
            z^2
        ]
        >,
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 7,
        Dimension := 3,
        Elements := [
            x*y,
            x*z,
            y*z
        ]
        >
]
```

If we apply this to degree four polynomials, we get the following:

```
> DecomposeGAction(G,S,4);
[
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 1,
        Dimension := 1,
        Elements := [
            x^4 + y^4 + z^4
        ]
        >,
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 3,
        Dimension := 2,
        Elements := [
            x^4 - z^4,
            y^4 - z^4
        ]
        >,
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 5,
        Dimension := 3,
        Elements := [
            x^2*y^2,
            x^2*z^2,
            y^2*z^2
        ]
        >,
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 6,
        Dimension := 3,
        Elements := [
            x^2*y*z,
            x*y^2*z,
            x*y*z^2
        ]
        >,
    rec<recformat<CharacterRow, Dimension, Elements> |
        CharacterRow := 10,
        Dimension := 6,
        Elements := [
            x^3*y,
            x^3*z,
            x*y^3,
            x*z^3,
            y^3*z,
            y*z^3
        ]
        >
]
```

and from the first entry in the output, we see that there is a one-dimensional subspace of the degree four homogeneous polynomials spanned by $x^4 + y^4 + z^4$ which does not get mixed with any other polynomials under G, and we have seen before that this is indeed the answer we are looking for.

There are a few problems with this method:

(1) You might get lots of candidate polynomials. Then you need to study each one.
(2) The polynomials you see might not give you a smooth curve.

We will see both of these issues arise next week.

The end

Don't forget to save your Magma session!

```
> UnsetLogFile();
> exit;
```

References

[1] Izumi Kuribayashi and Akikazu Kuribayashi, *Automorphism groups of compact Riemann surfaces of genera three and four*, J. Pure Appl. Algebra **65** (1990), no. 3, 277–292, DOI 10.1016/0022-4049(90)90107-S. MR1072285 (92a:30041) ←2

[2] Joseph Lewittes, *Automorphisms of compact Riemann surfaces*, Amer. J. Math. **85** (1963), 734–752. MR0160893 (28 #4102) ←2

Software Packages Referenced

[3] School of Mathematics and Statistics Computational Algebra Research Group University of Sydney, MAGMA *computational algebra system* (2008), available at http://magma.maths.usyd.edu.au/magma/. Version 2.15-1. ←