Magma Codes

August 23, 2023

[4]:
```
// Setting up K = Q(sqrt(5)) and all the constants.
// We do not need to work over Q_2 for these computations since all the
 ↪constants are integral.
// alphaijk = alpha_{i,j}^{(k)}.
K<z> := QuadraticField(5);
phi := (1+z)/2;
phibar := (1-z)/2;
Kx<x> := PolynomialRing(K);
x*(x-(phi - 2^3))*(x-(phi + 2^3))*(x-(phibar - 2^3))*(x-(phibar + 2^3));
beta1 := phi- 2^3;
beta2 := phibar - 2^3;
beta11 := phi - 2^3;
beta12 := phi + 2^3;
beta21 := phibar - 2^3;
beta22 := phibar + 2^3;
gamma := 0;
gamma1 := -beta1;
gamma2 := -beta2;
alpha111 := beta11 - beta1;
alpha112 := beta11 - beta2;
alpha121 := beta12 - beta1;
alpha122 := beta12 - beta2;
alpha211 := beta21 - beta1;
alpha212 := beta21 - beta2;
alpha221 := beta22 - beta1;
alpha222 := beta22 - beta2;
alpha11 := 0;
alpha22 := 0;
alpha12 := beta1 - beta2;
alpha21 := beta2 - beta1;
```

```
x^5 - 2*x^4 - 129*x^3 + 130*x^2 + 3905*x
```

[7]:
```
// Creating the extension L/K.
f := x^2+(alpha111 + alpha121 + gamma1);
L0 := ExtensionField<K, x | f>;
L0y<y> := PolynomialRing(L0);
g := y^2+(alpha212 + alpha222 + gamma2);
```

```
L := ExtensionField<L0, y | g>;
OL := Integers(L);
```

[9]:
```
Factorization(2*OL);
```

```
[
<Prime Ideal of OL
Two element generators:
[[[2, 0], [0, 0]], [[0, 0], [0, 0]]]
[[[3, 2], [1, -1]], [[2, 3], [1, -1]]], 4>
]
```

[22]:
```
k,f := ResidueClassField(Factorization(2*OL)[1,1]);
k;
```

```
Finite field of size 2^2
```

[15]:
```
// Gamma_{r_i}.
flag := true;
F := FiniteField(4);
A<X,s,t> := AffineSpace(F,3);
for a1,a2 in F do
    C:=Scheme(A, [X*(t^2+a1*s*t-a2*s^2)-1,X*s]);
    Dim := Dimension(C);
    Red := IsReduced(C);
    Sing := IsSingular(C);
    if not(Dim eq 1) or not(Red) or Sing then
        flag := false;
        a1,a2;
    end if;
end for;
flag;
```

```
true
```

[16]:
```
// Gamma_{s_i}.
flag := true;
F := FiniteField(4);
A<u,v,x,r,t> := AffineSpace(F,5);
for a1,a2,a4,a6 in F do
    if a1 ne 0 and a2 eq 0 and a6 eq 0 then
        C:=Scheme(A, [
        u*(t^2+a1*t-a2)-(1 + a4*u^2+a6*u^3),
        x-u*r^2,
        u*r,
        v-u*t]);
        Dim := Dimension(C);
```

```
        Red := IsReduced(C);
        Sing := IsSingular(C);
        if not(Dim eq 1) or not(Red) or Sing then
            a1,a2,a4,a6;
        end if;
    end if;
end for;
flag;
```

```
true
```

[19]:

```
// Gamma_{t_i}.
F := FiniteField(4);
A<u,v,x,r,s> := AffineSpace(F,5);
for a1,a2,a4,a6 in F do
    if a1 ne 0 and a2 eq 0 and a6 eq 0 then
        C:=Scheme(A,
↪[v*(1+a1*s-a2*s^2)-(s+a4*s^3*v^2+a6*s^4*v^3),x-r^2*(1-a1*v+a2*s*v+a4*s^2*v^2+a6*s^3*v^3),u-
↪
        Red := IsReduced(C);
        Sing := IsSingular(C);
        if not(Red) or Sing then
            Dimension(C);
            a1,a2,a4,a6;
            S := SingularSubscheme(C);
            PointsOverSplittingField(S);
        end if;
    end if;
end for;
```

```
1
1 0 0 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
1 0 1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
1 0 F.1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
1 0 F.1^2 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1 0 0 0
```

```
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1 0 1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1 0 F.1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1 0 F.1^2 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1^2 0 0 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1^2 0 1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1^2 0 F.1 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
1
F.1^2 0 F.1^2 0
{@ (0, 0, 0, 0, 0) @}
Finite field of size 2^2
```

[36]:

```
f(OL!(L0.1));
f(OL!(L.1));
```

```
k.1^2
k.1
```

[37]:

```
// Gamma_{t_i} for i = 1.
A<U,V,X,r,s> := AffineSpace(k,5);
a1 := k.1^2;
a2 := 0;
a4 := 0;
a6 := 0;
C:=Scheme(A, [
V*(1+a1*s-a2*s^2)-(s+a4*s^3*V^2+a6*s^4*V^3),
X-r^2*(1-a1*V+a2*s*V+a4*s^2*V^2+a6*s^3*V^3),
U-s*V,
V*r]);
```

```
Dimension(C);
IsReduced(C);
IsSingular(C);
```

```
1
true
true
```

[38]:
```
Irred := IrreducibleComponents(C);
Irred[1];
Irred[2];
Dimension(Irred[1]);
IsReduced(Irred[1]);
IsSingular(Irred[1]);
Dimension(Irred[1]);
IsReduced(Irred[2]);
IsSingular(Irred[2]);
```

```
Scheme over GF(2^2) defined by
U + k.1*V + k.1*s,
V*s + k.1*V + k.1*s,
X,
r
Scheme over GF(2^2) defined by
U,
V,
X + r^2,
s
1
true
false
1
true
false
```

[39]:
```
S := SingularSubscheme(C);
P := PointsOverSplittingField(S);
P[1];
IsNode(C,P[1]);
```

```
(0, 0, 0, 0, 0)
true
```

[40]:
```
// Gamma_{t_i} for i = 2.
A<U,V,X,r,s> := AffineSpace(k,5);
a1 := k.1;
a2 := 0;
a4 := 0;
a6 := 0;
```

```
C:=Scheme(A, [
V*(1+a1*s-a2*s^2)-(s+a4*s^3*V^2+a6*s^4*V^3),
X-r^2*(1-a1*V+a2*s*V+a4*s^2*V^2+a6*s^3*V^3),
U-s*V,
V*r]);
Dimension(C);
IsReduced(C);
IsSingular(C);
```

```
1
true
true
```

[41]:
```
Irred := IrreducibleComponents(C);
Irred[1];
Irred[2];
Dimension(Irred[1]);
IsReduced(Irred[1]);
IsSingular(Irred[1]);
Dimension(Irred[1]);
IsReduced(Irred[2]);
IsSingular(Irred[2]);
```

```
Scheme over GF(2^2) defined by
U + k.1^2*V + k.1^2*s,
V*s + k.1^2*V + k.1^2*s,
X,
r
Scheme over GF(2^2) defined by
U,
V,
X + r^2,
s
1
true
false
1
true
false
```

[42]:
```
S := SingularSubscheme(C);
P := PointsOverSplittingField(S);
P[1];
IsNode(C,P[1]);
```

```
(0, 0, 0, 0, 0)
true
```

[45]:
```
// Showing Spec B_i is irreducible for i = 1,2.
LL<X> := PolynomialRing(L,1);
FF := FieldOfFractions(LL);
x := FF!X;
RR<V> := PolynomialRing(FF,1);
Factorization(V^2 + (4*L0.1/x)*V + (4*x^3 + (4*z - 64)*x^2 - 1024*x + 992*z -
  ↪7840)/(x^3*(x+z)));
Factorization(V^2 + (4*L.1/x)*V + (4*x^3 - (4*z + 64)*x^2 - 1024*x - 992*z -
  ↪7840)/(x^3*(x+z)));
```

```
[
<V^2 + 4*L0.1/$.1*V + (4*$.1^3 + (4*z - 64)*$.1^2 - 1024*$.1 + (992*z -
7840))/($.1^4 + z*$.1^3), 1>
]
[
<V^2 + 4*L.1/$.1*V + (4*$.1^3 + (-4*z - 64)*$.1^2 - 1024*$.1 + (-992*z -
7840))/($.1^4 + z*$.1^3), 1>
]
```

[9]:
```
// Computing the irreducible components of Gamma_{t_i} for generic a_1 and a_4.
// Case a_4 is nonzero in k_L.
F<a1,a4> := PolynomialRing(FiniteField(2^2),2);
FF := FieldOfFractions(F);
A<u,v,x,r,s> := AffineSpace(FF,5);
C:=Scheme(A, [v*(1+a1*s)-(s+a4*s^3*v^2),x-r^2*(1-a1*v+a4*s^2*v^2),u-s*v,v*r]);
IrreducibleComponents(C);
```

```
[
Scheme over Multivariate rational function field of rank 2 over GF(2^2) defined
by
u + v*s,
v^2*s^3 + $.1/$.2*v*s + 1/$.2*v + 1/$.2*s,
x,
r,
Scheme over Multivariate rational function field of rank 2 over GF(2^2) defined
by
u,
v,
x + r^2,
s
]
```

[10]:
```
// Case a_4 is zero in k_L.
F<a1> := PolynomialRing(FiniteField(2^2),1);
FF := FieldOfFractions(F);
A<u,v,x,r,s> := AffineSpace(FF,5);
C:=Scheme(A, [v*(1+a1*s)-s,x-r^2*(1-a1*v),u-s*v,v*r]);
IrreducibleComponents(C);
```

```
[
Scheme over Multivariate rational function field of rank 1 over GF(2^2) defined
by
u + 1/$.1*v + 1/$.1*s,
v*s + 1/$.1*v + 1/$.1*s,
x,
r,
Scheme over Multivariate rational function field of rank 1 over GF(2^2) defined
by
u,
v,
x + r^2,
s
]
```