SingularM2Sage

Manoj Kummini

August 9, 2021

We give some examples of using SageMath to mix calculations that use Macaulay2 and Singular. For a specific example, we compute some modular invariants, for which there is a Singular library, but no Macaulay2 library.

1 Initial computations in Singular

In this section, we start with Singular objects, and then pass the results of the computation to Macaulay2 objects.

[1]:
```
R = singular.ring(3, '(x4,x3,x2,x1)', 'dp')
```

[2]:
```
g1 = singular.matrix(4,4,'1,0,0,1,0,1,0,0,0,0,1,1,0,0,0,1')
g2 = singular.matrix(4,4,'1,0,1,0,0,1,1,0,0,0,1,0,0,0,0,1')
```

Now printing the Singular object and its SageMath conversion.

[3]:
```
print(g1)
```

```
1,0,0,1,
0,1,0,0,
0,0,1,1,
0,0,0,1
```

[4]:
```
print(g1.sage())
```

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

Import the library for computing invariants.

[5]:
```
singular.lib('finvar.lib')
```

There is a Singular command invariant_ring which computes the full invariant ring (i.e., a set of primary invariants and a set of secondary invariants). However there is a possible bug in Sage reading this matrix/tuple output of Singular. Hence we call only the primary_invariants command from Sage. (One can check from working within Singular that the secondary invariants for this action are trivial.)

[6]:
```
P1 = singular.primary_invariants(g1,g2)
```

[7]:
```
print(P1)
```

```
x4-x3-x2,              x1,                    x2^3-x4*x1^2+x3*x1^2,
x3^9-x4^3*x2^6+x2^..
```

[8]:
```
print(P1.sage())
```

```
[
x4 - x3 - x2
x1
x2^3 - x4*x1^2 + x3*x1^2 x3^9 - x4^3*x2^6 + x2^9 - x4^3*x2^4*x1^2 + x4*x2^6*x1^2
- x4^3*x2^2*x1^4 + x4*x2^4*x1^4 - x4^3*x1^6 + x4*x2^2*x1^6]
```

[9]:
```
print(P1.sage()[0])
```

```
(x4 - x3 - x2, x1, x2^3 - x4*x1^2 + x3*x1^2, x3^9 - x4^3*x2^6 + x2^9 -
x4^3*x2^4*x1^2 + x4*x2^6*x1^2 - x4^3*x2^2*x1^4 + x4*x2^4*x1^4 - x4^3*x1^6 +
x4*x2^2*x1^6)
```

[10]:
```
f = P1.sage()[0][3]
```

[11]:
```
print(f)
```

```
x3^9 - x4^3*x2^6 + x2^9 - x4^3*x2^4*x1^2 + x4*x2^6*x1^2 - x4^3*x2^2*x1^4 +
x4*x2^4*x1^4 - x4^3*x1^6 + x4*x2^2*x1^6
```

We would now like to manipulate f in `Macaulay2`. As an example, we show that the coefficient of $x_4^{p^e}$ is an invariant for each e, since the action is triangular. First, convert the ring and the group elements.

[12]:
```
RM2 = macaulay2(R.sage())
```

[13]:
```
g1m2 = macaulay2(g1.sage()).substitute(RM2).transpose()
g2m2 = macaulay2(g2.sage()).substitute(RM2).transpose()
```

[14]:
```
print(g1m2)
```

```
| 1 0 0 0 |
| 0 1 0 0 |
| 0 0 1 0 |
| 1 0 1 1 |
```

We define the two ring maps defined by the two matrices above.

[15]:
```
phi1 = macaulay2.map(RM2, RM2, macaulay2.matrix(RM2.vars())*g1m2)
phi2 = macaulay2.map(RM2, RM2, macaulay2.matrix(RM2.vars())*g2m2)
```

[16]:
```
x4, x3, x2, x1 = RM2.gens()
```

Rewrite the ring as $\mathbb{Z}/3[x_1, x_2, x_3][x_4]$ and make f and element of this ring.

[17]:
```
RM2b = macaulay2.ring('ZZ/3', '[x3,x2,x1]', order='Lex')
```

```
[18]: RM2s = macaulay2.ring(RM2b.name(), '[x4]')
```

```
[19]: fs = macaulay2(f).substitute(RM2s)
```

```
[20]: print(fs)
```

```
         6      4  2      2  4      6    3       6  2      4  4      2  6           9      9
(- x2  - x2 x1  - x2 x1  - x1 )x4  + (x2 x1  + x2 x1  + x2 x1 )x4 + x3  + x2
```

Extract the coefficients of x_4^i.

```
[21]: f0, f1, f3 = [(macaulay2.coefficient(x4^i, fs)).substitute(RM2) for i in [0,1,3]]
```

```
[22]: [macaulay2.toString(v) for v in [f0, f1, f3]]
```

```
[22]: [x3^9+x2^9, x2^6*x1^2+x2^4*x1^4+x2^2*x1^6, -x2^6-x2^4*x1^2-x2^2*x1^4-x1^6]
```

As expected, the coefficients of x_4 and of x_4^3 are invariants.

```
[23]: [v == phi1(v) for v in [f0, f1, f3]]
```

```
[23]: [False, True, True]
```

```
[24]: [v == phi2(v) for v in [f0, f1, f3]]
```

```
[24]: [False, True, True]
```