Vasconcelas example 3.67

Test example I_7 of the Greuel, Laplagne, Seelich 2009 paper, *Normalization of rings*, is

$$x^2 + zw,\ y^3 + xwt,\ xw^3 + z^3t + ywt^2,\ y^2w^4 - xy^2z^2t - w^3t^3$$

which seems to be the homogeneous form of example 3.67 on page 95 of Vasconcelos book, *Computational Methods in Commutative Algebra and Algebraic Geometry*, where there is no further attribution.

In the paper, there is timing data for `normal` against `normalP` and the old normal function, now called `normalC`. In char 0, 12 sec, *, 582 sec. In char 2, 11 sec, 0 sec, 35 sec. In char 5, 12 sec, 3 sec, 318 sec. In char 11, 11 sec, 43 sec, 503 sec. In char 32003, 11 sec,*, 617 sec.

This leads to conclusions that, among other things, `normal` is fast, and `normalP` slows down with larger characteristic. In terms of timing, this is a biased view (as is the present annotated one), in that the authors are comparing their `normal` to a possibly not well-written stripped-down version of the qth-power algorithm `normalP`.

But what they do not mention is the actual presentation of the integral closure gotten.

1 Singular

So it is only fair to start by running this example in Singular.

```
                     SINGULAR                                 /
 A Computer Algebra System for Polynomial Computations       /   version 3-1-3
                                                           0<
 by: W. Decker, G.-M. Greuel, G. Pfister, H. Schoenemann     \   March 2011
FB Mathematik der Universitaet, D-67653 Kaiserslautern        \
> LIB "normal.lib";
> ring r=0,(x,y,z,w,t),dp;
> ideal i=x2+zw,y3+xwt,xw3+z3t+ywt2,y2w4-xy2z2t-w3t3;
> list nor=normal(i);nor;
//   characteristic : 0
//   number of vars : 26
//        block   1 : ordering dp
//                  : names    T(1) T(2) T(3) T(4) T(5) T(6) T(7) T(8)
                               T(9) T(10) T(11) T(12) T(13) T(14) T(15)
                               T(16) T(17) T(18) T(19) T(20) T(21)
//        block   2 : ordering dp
//                  : names    x y z w t
//        block   3 : ordering C
[2]:
   [1]:
      _[1]=zw3t2
      _[2]=z4t3+yzwt4
      _[3]=xz3t3+xywt4
      _[4]=yz2w2t2
      _[5]=y2zw2t2
      _[6]=xyzw2t2
      _[7]=z3w2t3
      _[8]=xz2w2t3
      _[9]=xy2w2t3
      _[10]=y2z3wt2
      _[11]=xyz3wt2
      _[12]=xy2z2wt2
      _[13]=y2z4t2-xzw2t4
      _[14]=xy2z3t2
      _[15]=yz3t5+y2wt6
      _[16]=y2z3t4-xw2t6
      _[17]=z3t7+ywt8
      _[18]=y2w3t6
      _[19]=yw3t8
      _[20]=xy2z2t7+w3t9
      _[21]=w4t9
      _[22]=y6
```

```
> def R=nor[1][1];
> setring R;
> option(redSB);
> ideal s=std(norid);s;
s[1]=x^2+z*w
s[2]=y^3+x*w*t
s[3]=x*w^3+z^3*t+y*w*t^2
s[4]=z*w^4-x*z^3*t-x*y*w*t^2
s[5]=y^2*w^4-x*y^2*z^2*t-w^3*t^3
s[6]=T(21)*z+w*t^7
s[7]=T(21)*y-y*w^7-y*z^5*t^2-2*y^2*z^2*w*t^3-z^2*t^6
s[8]=T(21)*w^2-w^9-z^5*w^2*t^2-2*y*z^2*w^3*t^3-x*y^2*z*t^5
s[9]=T(21)*x*w+y*w^6*t^2-x*y*z^2*w^2*t^3+y^2*z^2*t^5
s[10]=T(20)-w^6+x*z^2*w^2*t-y*z^2*t^3
s[11]=T(19)*z+y*t^6
s[12]=T(19)*w*t-y*w^7-y*z^5*t^2-2*y^2*z^2*w*t^3-z^2*t^6
s[13]=T(19)*y*w-w^6*t^2+x*z^2*w^2*t^3-y*z^2*t^5
s[14]=T(19)*x*w+w^4*t^4-x*z^2*t^5
s[15]=T(19)*y^2+T(21)*x
s[16]=T(18)*w-w^6+x*z^2*w^2*t-y*z^2*t^3
s[17]=T(18)*z+y^2*t^4
s[18]=T(18)*t^2-T(19)*y
s[19]=T(18)*y^2-w^4*t^3+x*z^2*t^4
s[20]=T(18)*x*y+w^2*t^5
s[21]=T(17)*z-x*t^4
s[22]=T(17)*x+w*t^4
s[23]=T(17)*w*t-T(18)*y
s[24]=T(17)*y*w-w^4*t^2+x*z^2*t^3
s[25]=T(17)*y^2+T(18)*x
s[26]=T(17)*y*t^2+T(19)*x
s[27]=T(17)*w^3-T(19)*w+z^2*t^5
s[28]=T(16)*z-x*y^2*t
s[29]=T(16)*y-w^2*t^2
s[30]=T(16)*x+y^2*w*t
s[31]=T(16)*w*t-y*w^4+x*y*z^2*t
s[32]=T(16)*t^3+T(18)*x
s[33]=T(16)*w^3-T(17)*w^2+y^2*z^2*t^2
s[34]=T(15)*w-w^4+x*z^2*t
s[35]=T(15)*z-x*y*t^2
s[36]=T(15)*y-T(16)*t
s[37]=T(15)*x+y*w*t^2
s[38]=T(15)*t^2-T(17)*y
s[39]=T(14)*t+y^2*w-t^3
s[40]=T(14)*y-x*w^2-y*t^2
s[41]=T(14)*w^3+x*y^2*z^2
s[42]=T(14)*x*w^2-y^2*z^3
```

```
s[43]=T(13)*t-x*y^2
s[44]=T(13)*w+T(14)*x-x*t^2
s[45]=T(13)*y-z*w^2
s[46]=T(13)*x-T(14)*z+z*t^2
s[47]=T(12)*z-T(14)*w
s[48]=T(12)*y+z^2*t
s[49]=T(12)*w^2+x*y^2*z
s[50]=T(12)*x*w-y^2*z^2
s[51]=T(12)*x*t^2+T(16)*w^2-T(17)*w
s[52]=T(12)*t^5+T(21)-w^7-z^5*t^2-2*y*z^2*w*t^3
s[53]=T(11)*y-T(14)*w
s[54]=T(11)*w*t-T(16)*t+y*w^3
s[55]=T(11)*x*t+x*y*w^2+y^2*t^2
s[56]=T(11)*w^2+x*y*z^2
s[57]=T(11)*x*w-y*z^3
s[58]=T(11)*z^2*t+T(12)*w*t^2+y*z^2*w^2
s[59]=T(10)*w+T(12)*x
s[60]=T(10)*z+T(14)*x
s[61]=T(10)*y+T(15)-w^3
s[62]=T(10)*x-T(14)*w
s[63]=T(10)*t^2-T(16)*w+T(17)
s[64]=T(9)*t+T(11)*t+y*w^2
s[65]=T(9)*w+T(16)
s[66]=T(9)*y+w*t^2
s[67]=T(9)*x-y^2*t
s[68]=T(9)*z^2-T(12)*w*t
s[69]=T(8)*w+x*z*t
s[70]=T(8)*z-T(15)+w^3
s[71]=T(8)*x-z^2*t
s[72]=T(8)*y^2-T(12)*w*t
s[73]=T(7)*w+z^2*t
s[74]=T(7)*z-x*w^2-y*t^2
s[75]=T(7)*x-T(15)+w^3
s[76]=T(7)*y^2+T(12)*x*t
s[77]=T(7)*y*t^2+T(18)-w^5+x*z^2*w*t
s[78]=T(7)*t^4-T(17)*w^2+T(19)
s[79]=T(6)*w+x*y
s[80]=T(6)*x-y*z
s[81]=T(6)*z*t-T(8)*y
s[82]=T(6)*y*t-T(9)*z
s[83]=T(6)*z^2-T(11)*w
s[84]=T(6)*y*z-T(12)*w
s[85]=T(6)*y^2+z*w*t
s[86]=T(5)*w+y^2
s[87]=T(5)*y-x*t
s[88]=T(5)*x-T(6)*y
```

```
s[89]=T(5)*z^2+T(12)*x
s[90]=T(4)*w+y*z
s[91]=T(4)*y-T(5)*z
s[92]=T(4)*x-T(6)*z
s[93]=T(4)*z*t-T(7)*y
s[94]=T(4)*z^2+T(11)*x
s[95]=T(3)+w
s[96]=T(2)-x
s[97]=T(1)+1
s[98]=T(21)^2+T(11)*z*t^11-w^14+z^10*t^4+4*y*z^7*w*t^5
      +6*y^2*z^4*w^2*t^6+4*z^4*w*t^9+6*y*z*w^2*t^10
s[99]=T(19)*T(21)-T(19)*w^7+y*z^4*t^8+2*y^2*z*w*t^9+z*t^12
s[100]=T(18)*T(21)-w^12-x*z^7*w*t^3-3*x*y*z^4*w^2*t^4+3*y^2*z^4*t^6
      -4*x*z*w^2*t^8+y*z*t^10
s[101]=T(17)*T(21)-T(5)*z*t^9-T(19)*w^5-x*y*z*w*t^7
s[102]=T(16)*T(21)-T(19)*w^4-x*y*z*t^7
s[103]=T(15)*T(21)-w^10-z^5*w^3*t^2-2*x*y*z^4*t^4-3*x*y^2*z*w*t^5-x*z*t^8
s[104]=T(14)*T(21)-T(21)*t^2+w^7*t^2+z^5*t^4+2*y*z^2*w*t^5
s[105]=T(13)*T(21)+y*w^4*t^4-x*y*z^2*t^5
s[106]=T(12)*T(21)-T(11)*z*t^6-y*z*w^2*t^5
s[107]=T(11)*T(21)+T(8)*y*t^6
s[108]=T(10)*T(21)+T(5)*z*t^7
s[109]=T(9)*T(21)-T(8)*y*t^6+T(19)*w^3
s[110]=T(8)*T(21)-x*t^8
s[111]=T(7)*T(21)-z*t^8
s[112]=T(6)*T(21)+T(19)*x*t
s[113]=T(5)*T(21)+T(19)*y*t
s[114]=T(4)*T(21)-y*t^7
s[115]=T(19)^2-T(4)*t^11-w^11*t-x*z^7*t^4-3*x*y*z^4*w*t^5
      -3*x*y^2*z*w^2*t^6-x*z*w*t^9
s[116]=T(18)*T(19)-T(5)*z*t^9-T(19)*w^5-x*y*z*w*t^7
s[117]=T(17)*T(19)+T(8)*t^8-w^9*t-z^5*w^2*t^3-2*y*z^2*w^3*t^4-x*y^2*z*t^6
s[118]=T(16)*T(19)-T(21)*w*t
s[119]=T(15)*T(19)+T(8)*y*t^6-T(19)*w^3
s[120]=T(14)*T(19)-T(19)*t^2+y*w^6*t-x*y*z^2*w^2*t^2+y^2*z^2*t^4
s[121]=T(13)*T(19)+w^2*t^6
s[122]=T(12)*T(19)-z*t^7
s[123]=T(11)*T(19)-T(21)*t+w^7*t+z^5*t^3+2*y*z^2*w*t^4
s[124]=T(10)*T(19)-T(8)*t^6
s[125]=T(9)*T(19)+T(21)*t
s[126]=T(8)*T(19)+T(6)*t^7
s[127]=T(7)*T(19)+T(4)*t^7
s[128]=T(6)*T(19)-T(11)*t^5-y*w^2*t^4
s[129]=T(5)*T(19)+T(17)*t^3
s[130]=T(4)*T(19)+T(5)*t^6
s[131]=T(18)^2-w^10-z^5*w^3*t^2-2*x*y*z^4*t^4-3*x*y^2*z*w*t^5-x*z*t^8
```

```
s[132]=T(17)*T(18)+T(8)*y*t^6-T(19)*w^3
s[133]=T(16)*T(18)-T(19)*w^2
s[134]=T(15)*T(18)-T(21)*w
s[135]=T(14)*T(18)-T(19)*y+w^5*t^2-x*z^2*w*t^3
s[136]=T(13)*T(18)+y*w^2*t^4
s[137]=T(12)*T(18)-y*z*t^5
s[138]=T(11)*T(18)-z^2*t^5
s[139]=T(10)*T(18)-T(8)*y*t^4
s[140]=T(9)*T(18)+T(19)*w
s[141]=T(8)*T(18)-T(11)*z*t^4-y*z*w^2*t^3
s[142]=T(7)*T(18)+T(5)*z*t^5
s[143]=T(6)*T(18)-w*t^5
s[144]=T(5)*T(18)+T(17)*y*t
s[145]=T(4)*T(18)+x*t^5
s[146]=T(17)^2-T(21)*t
s[147]=T(16)*T(17)-w^6*t+x*z^2*w^2*t^2-y*z^2*t^4
s[148]=T(15)*T(17)-T(19)*w
s[149]=T(14)*T(17)-T(17)*t^2+y*w^4*t-x*y*z^2*t^2
s[150]=T(13)*T(17)+y^2*w*t^3
s[151]=T(12)*T(17)+T(5)*z*t^4
s[152]=T(11)*T(17)-T(18)*t+w^5*t-x*z^2*w*t^2
s[153]=T(10)*T(17)-T(12)*t^4
s[154]=T(9)*T(17)+T(18)*t
s[155]=T(8)*T(17)-z*t^5
s[156]=T(7)*T(17)-T(8)*t^4
s[157]=T(6)*T(17)-y*t^4
s[158]=T(5)*T(17)+T(11)*t^3+y*w^2*t^2
s[159]=T(4)*T(17)-T(6)*t^4
s[160]=T(16)^2-w^5*t+x*z^2*w*t^2
s[161]=T(15)*T(16)-T(17)*w^2
s[162]=T(14)*T(16)-T(16)*t^2+y*w^3*t
s[163]=T(13)*T(16)-x*y*w^2*t
s[164]=T(12)*T(16)+x*y*z*t^2
s[165]=T(11)*T(16)+x*z^2*t^2
s[166]=T(10)*T(16)+y*z^2*t^2
s[167]=T(9)*T(16)+w^4*t-x*z^2*t^2
s[168]=T(8)*T(16)-y^2*z*t^2
s[169]=T(7)*T(16)-T(12)*w*t^2
s[170]=T(6)*T(16)+x*w*t^2
s[171]=T(5)*T(16)+y*w*t^2
s[172]=T(4)*T(16)+z*w*t^2
s[173]=T(15)^2-w^6+x*z^2*w^2*t-y*z^2*t^3
s[174]=T(14)*T(15)-T(17)*y+w^3*t^2
s[175]=T(13)*T(15)-x*w^2*t^2
s[176]=T(12)*T(15)+x*z*t^3
s[177]=T(11)*T(15)+T(16)*w^2-T(17)*w
```

```
s[178]=T(10)*T(15)+z^2*t^3
s[179]=T(9)*T(15)+T(17)*w
s[180]=T(8)*T(15)-y*z*t^3
s[181]=T(7)*T(15)-T(8)*y*t^2
s[182]=T(6)*T(15)-y^2*t^2
s[183]=T(5)*T(15)+w*t^3
s[184]=T(4)*T(15)+T(11)*z*t+y*z*w^2
s[185]=T(14)^2-y*z^3+y^2*w*t-t^4
s[186]=T(13)*T(14)-T(11)*z*w
s[187]=T(12)*T(14)-T(12)*t^2-y*z^2*w
s[188]=T(11)*T(14)-T(11)*t^2-z^3*w
s[189]=T(10)*T(14)-T(16)*w+T(17)+x*y*z^2
s[190]=T(9)*T(14)+T(11)*t^2
s[191]=T(8)*T(14)-T(8)*t^2-x*y^2*z
s[192]=T(7)*T(14)-T(7)*t^2-y^2*z^2
s[193]=T(6)*T(14)-T(6)*t^2-z*w^2
s[194]=T(5)*T(14)-T(5)*t^2+x*y*w
s[195]=T(4)*T(14)-T(4)*t^2+x*z*w
s[196]=T(13)^2+T(11)*x*z+y^2*z*t
s[197]=T(12)*T(13)+x*y*z^2
s[198]=T(11)*T(13)+x*z^3
s[199]=T(10)*T(13)+y*z^3
s[200]=T(9)*T(13)+x*y*w*t
s[201]=T(8)*T(13)-y^2*z^2
s[202]=T(7)*T(13)-T(14)*w^2
s[203]=T(6)*T(13)+x*z*w
s[204]=T(5)*T(13)+y*z*w
s[205]=T(4)*T(13)+z^2*w
s[206]=T(12)^2+T(11)*z*t
s[207]=T(11)*T(12)-T(8)*y*t-z^2*w^2
s[208]=T(10)*T(12)-T(5)*z*t^2+x*y*z*w
s[209]=T(9)*T(12)+T(8)*y*t
s[210]=T(8)*T(12)-x*y^2*w+x*t^3
s[211]=T(7)*T(12)-y^2*z*w+z*t^3
s[212]=T(6)*T(12)+T(15)-w^3
s[213]=T(5)*T(12)+T(7)*y
s[214]=T(4)*T(12)+x*w^2+y*t^2
s[215]=T(11)^2-T(13)*z^2-T(15)*t+w^3*t
s[216]=T(10)*T(11)-T(7)*y*t+x*z^2*w
s[217]=T(9)*T(11)+T(15)*t-w^3*t
s[218]=T(8)*T(11)+T(5)*z*t^2-x*y*z*w
s[219]=T(7)*T(11)-T(12)*t^2-y*z^2*w
s[220]=T(6)*T(11)+T(14)*x
s[221]=T(5)*T(11)+x*w^2+y*t^2
s[222]=T(4)*T(11)+T(14)*z
s[223]=T(10)^2+T(12)*t^2+y*z^2*w
```

```
s[224]=T(9)*T(10)+T(7)*y*t
s[225]=T(8)*T(10)-y^2*z*w+z*t^3
s[226]=T(7)*T(10)+T(8)*t^2+x*y^2*z
s[227]=T(6)*T(10)+x*w^2+y*t^2
s[228]=T(5)*T(10)-T(11)*t
s[229]=T(4)*T(10)+T(6)*t^2+z*w^2
s[230]=T(9)^2-T(15)*t
s[231]=T(8)*T(9)-T(5)*z*t^2
s[232]=T(7)*T(9)+T(12)*t^2
s[233]=T(6)*T(9)-x*t^2
s[234]=T(5)*T(9)-y*t^2
s[235]=T(4)*T(9)-z*t^2
s[236]=T(8)^2-x*w^2*t-y*t^3
s[237]=T(7)*T(8)-T(6)*t^3-z*w^2*t
s[238]=T(6)*T(8)-T(7)*y
s[239]=T(5)*T(8)+T(12)*t
s[240]=T(4)*T(8)+T(11)*t
s[241]=T(7)^2-T(4)*t^3+x*z*w*t
s[242]=T(6)*T(7)+T(11)*t
s[243]=T(5)*T(7)+T(10)*t
s[244]=T(4)*T(7)-T(5)*t^2+x*y*w
s[245]=T(6)^2-T(5)*z
s[246]=T(5)*T(6)-z*t
s[247]=T(4)*T(6)+T(12)
s[248]=T(5)^2-T(6)*t
s[249]=T(4)*T(5)-T(8)
s[250]=T(4)^2+T(10)
```

This produces a *strict, affine r-algebra* presentation meaning that relations s[1] through s[5] form a Gröbner basis for the input ring, r, relations s[6] through s[97] are r-linear relations in the 21 new variables found, and s[98] through s[250] are r-quadratic relation defining an r-multiplication. Notice that relations s[10], s[95], s[96], s[97] are tacit admissions that T(20), T(3), T(2), and T(1) respectively are completely unnecessary variables.

As in the paper, this seems to be fairly independent of characteristic in that char 32003 runs just as fast with the same presentation.

Now let's try the unhomogenized version that was in the Vasconcelos book.

```
> ring r=0,(x,y,z,w),dp;
> ideal i=x2+zw,y3+xw,xw3+z3+yw,y2w4-xy2z2-w3;
> list nor=normal(i);nor;
//   characteristic : 0
//   number of vars : 14
//        block   1 : ordering dp
//                  : names    T(1) T(2) T(3) T(4) T(5) T(6) T(7) T(8) T(9) T(10)
//        block   2 : ordering dp
//                  : names    x y z w
//        block   3 : ordering C
[2]:
   [1]:
      _[1]=z3+yw
      _[2]=z2w2
      _[3]=yzw2
      _[4]=xzw2
      _[5]=y2w2
      _[6]=xyw2
      _[7]=y2z2w
      _[8]=xyz2w
      _[9]=xy2zw
      _[10]=xy2z2
      _[11]=w3
> def R=nor[1][1];
> setring R;
> normap;
normap[1]=x
normap[2]=y
normap[3]=z
normap[4]=w
> option(redSB);
> ideal s=std(norid);s;
s[1]=x^2+z*w
s[2]=y^3+x*w
s[3]=x*w^3+z^3+y*w
s[4]=z*w^4-x*z^3-x*y*w
s[5]=y^2*w^4-x*y^2*z^2-w^3
s[6]=T(10)-y^2*w+1
s[7]=T(9)*z-y^2*w^2+w
s[8]=T(9)*y-z^2
s[9]=T(9)*w^2-x*y^2*z
s[10]=T(9)*x*w+y^2*z^2
s[11]=T(8)*y-y^2*w^2+w
s[12]=T(8)*x-x*y*w^2-y^2
```

```
s[13]=T(8)*w^2-x*y*z^2
s[14]=T(8)*z*w-y*z*w^3+x*y^2
s[15]=T(8)*z^2+T(9)*w-y*z^2*w^2
s[16]=T(7)*w+T(9)*x
s[17]=T(7)*z+x*y^2*w-x
s[18]=T(7)*x-y^2*w^2+w
s[19]=T(7)*y^2+T(8)*w
s[20]=T(6)*w-x*y
s[21]=T(6)*y+T(8)*z-y*z*w^2
s[22]=T(6)*x+y*z
s[23]=T(6)*z^2-T(8)*w
s[24]=T(5)*w-y^2
s[25]=T(5)*y+x
s[26]=T(5)*x+T(8)*z-y*z*w^2
s[27]=T(5)*z^2+T(9)*x
s[28]=T(4)*w-x*z
s[29]=T(4)*z+T(7)*y
s[30]=T(4)*y-T(6)*z
s[31]=T(4)*x+z^2
s[32]=T(3)*w-y*z
s[33]=T(3)*y-T(5)*z
s[34]=T(3)*x-T(6)*z
s[35]=T(3)*z^2+x*y*w^2+y^2
s[36]=T(2)*w-z^2
s[37]=T(2)*z+x*w^2+y
s[38]=T(2)*y-T(3)*z
s[39]=T(2)*x+T(7)*y
s[40]=T(1)+x
s[41]=T(9)^2-T(8)*z
s[42]=T(8)*T(9)+T(6)*z-z^2*w^2
s[43]=T(7)*T(9)+T(5)*z+x*y*z*w
s[44]=T(6)*T(9)+T(7)*y
s[45]=T(5)*T(9)-T(3)*z
s[46]=T(4)*T(9)-x*y^2*w+x
s[47]=T(3)*T(9)+x*w^2+y
s[48]=T(2)*T(9)-y^2*z*w+z
s[49]=T(8)^2-T(7)*y-x*y^2*z^2
s[50]=T(7)*T(8)+T(3)*z+x*z^2*w
s[51]=T(6)*T(8)-x*y^2*w+x
s[52]=T(5)*T(8)+x*w^2+y
s[53]=T(4)*T(8)-T(5)*z-x*y*z*w
s[54]=T(3)*T(8)-y^2*z*w+z
s[55]=T(2)*T(8)+T(9)-y*z^2*w
s[56]=T(7)^2-T(9)+y*z^2*w
s[57]=T(6)*T(7)+x*w^2+y
s[58]=T(5)*T(7)+T(8)
```

```
s[59]=T(4)*T(7)-y^2*z*w+z
s[60]=T(3)*T(7)-T(6)+z*w^2
s[61]=T(2)*T(7)-T(4)+x*y^2*z
s[62]=T(6)^2+T(5)*z
s[63]=T(5)*T(6)-z
s[64]=T(4)*T(6)+T(3)*z
s[65]=T(3)*T(6)-T(9)
s[66]=T(2)*T(6)-T(8)
s[67]=T(5)^2+T(6)
s[68]=T(4)*T(5)-T(9)
s[69]=T(3)*T(5)+T(4)
s[70]=T(2)*T(5)-T(7)
s[71]=T(4)^2-x*w^2-y
s[72]=T(3)*T(4)-T(8)
s[73]=T(2)*T(4)+T(6)-z*w^2
s[74]=T(3)^2-T(7)
s[75]=T(2)*T(3)+T(5)+x*y*w
s[76]=T(2)^2+T(3)+x*z*w
```

Well, this again gives a *strict affine r-algebra* presentation, but this time with only 10 new variables and 76 relations. Clearly again T(10) and T(1) are completely unnecessary.

Is even this better presentation good? Well, let's consider that the monomial ordering produced by `normal` is a product order, with a default `dp` monomial ordering on the new variables corresponding to the fractions found, and the input monomial ordering used on the old variables. That is, there is no concept that the input monomial ordering might induce a monomial ordering on the output. If it should, then this information will not be readily seen from the presentation.

2 Macaulay2

```
Macaulay2, version 1.4
with packages: ConwayPolynomials, Elimination, IntegralClosure, LLLBases,
               PrimaryDecomposition, ReesAlgebra, TangentCone

i1 : A=QQ[x,y,z,w,t]/ideal(x^2+z*w,y^3+x*w*t,x*w^3+z^3*t+y*w*t^2,
                          y^2*w^4-x*y^2*z^2*t-w^3*t^3);

i2 : time icf=icFractions A;
     -- used 1.64329 seconds

i3 : toString icf
o3 = {(z*w^2)/(y), (-z^3*t)/(y^2), (-y^2*w)/(t), (-y*z)/(w), (-y^2)/(w),
     (x*y)/(w), (w*t^2)/(y), (x*w^2+y*t^2)/(z), x, y, z, w, t}

i4 : G=transpose gens gb presentation integralClosure A

o4 = {-2} | x2+zw                            |
     {-3} | y3+xwt                           |
     {-4} | xw3+z3t+ywt2                     |
     {-5} | zw4-xz3t-xywt2                   |
     {-6} | y2w4-xy2z2t-w3t3                 |
     {-2} | w_(2,1)w-xy                      |
     {-2} | w_(2,1)x+yz                      |
     {-3} | w_(2,1)y2-zwt                    |
     {-5} | w_(2,1)z3t-yzw3+xy2t2            |
     {-5} | w_(2,1)yz2t-y2w3+w2t3            |
     {-2} | w_(2,0)w+y2                      |
     {-2} | w_(2,0)y-xt                      |
     {-2} | w_(2,0)x+w_(2,1)y                |
     {-5} | w_(2,0)z3t-xy2w2+xwt3            |
     {-3} | w_(2,6)w+z2t                     |
     {-3} | w_(2,6)z-xw2-yt2                 |
     {-4} | w_(2,6)y2-w_(2,0)z2t             |
     {-4} | w_(2,6)xy+w_(2,1)z2t             |
     {-3} | w_(2,2)z-w_(2,1)yt               |
     {-3} | w_(2,2)y-wt2                     |
     {-3} | w_(2,2)x+y2t                     |
     {-4} | w_(2,2)wt+w_(2,1)z2t-yw3         |
     {-2} | w_(2,1)^2-w_(2,0)z               |
     {-2} | w_(2,0)w_(2,1)+zt                |
     {-2} | w_(2,0)^2+w_(2,1)t               |
     {-3} | w_(2,1)w_(2,6)-w_(2,2)t+yw2      |
     {-3} | w_(2,1)w_(2,2)-xt2               |
     {-3} | w_(2,0)w_(2,2)+yt2               |
```

```
{-5} | w_(2,6)^2y-w_(2,0)zt3+xyzwt          |
{-5} | w_(2,6)^2x+w_(2,1)zt3-z2w2t          |
{-4} | w_(2,2)^2-w_(2,6)xt-w3t              |
{-6} | w_(2,6)^3-w_(2,0)t5-xz3t2+xywt3     |
{-6} | w_(2,2)w_(2,6)^2-y2zwt2+zt5         |
{-2} | w_(3,0)w+yz                         |
{-2} | w_(3,0)y-w_(2,0)z                   |
{-2} | w_(3,0)x+w_(2,1)z                   |
{-3} | w_(3,0)zt-w_(2,6)y                  |
{-4} | w_(3,0)t3-w_(2,6)^2-xzwt            |
{-4} | w_(3,0)w_(2,1)t2-w_(2,2)w_(2,6)     |
{-3} | w_(3,0)w_(2,0)z-w_(2,6)x            |
{-3} | w_(3,0)w_(2,6)-w_(2,0)t2+xyw        |
{-3} | w_(3,0)w_(2,2)+zt2                  |
{-3} | w_(3,0)^2t-w_(2,0)w_(2,6)           |
{-3} | w_(3,0)^2w_(2,1)+xw2+yt2            |
{-3} | w_(3,0)^2w_(2,0)+w_(2,2)t-yw2       |
{-3} | w_(3,0)^3+w_(2,1)t2-zw2             |
{-3} | w_(4,0)t+y2w                        |
{-3} | w_(4,0)w-w_(3,0)w_(2,1)z+wt2        |
{-3} | w_(4,0)y-xw2                        |
{-3} | w_(4,0)x-w_(3,0)^2z+xt2             |
{-3} | w_(4,0)w_(2,1)+zw2                  |
{-3} | w_(4,0)w_(2,0)+xyw                  |
{-4} | w_(4,0)w_(2,6)-y2z2                 |
{-4} | w_(4,0)w_(2,2)+yw2t                 |
{-3} | w_(4,0)w_(3,0)+xzw                  |
{-4} | w_(4,0)^2-yz3-y2wt                  |
{-3} | w_(5,1)t-w_(2,2)t+yw2               |
{-3} | w_(5,1)w+w_(2,1)z2                  |
{-3} | w_(5,1)y-w_(3,0)w_(2,1)z            |
{-3} | w_(5,1)x+w_(3,0)z2                  |
{-3} | w_(5,1)w_(2,1)-w_(3,0)^2z           |
{-3} | w_(5,1)w_(2,0)+xw2+yt2              |
{-4} | w_(5,1)w_(2,6)-w_(2,2)w_(2,6)-yz2w |
{-4} | w_(5,1)w_(2,2)-w_(2,6)xt            |
{-3} | w_(5,1)w_(3,0)+w_(4,0)z+zt2         |
{-4} | w_(5,1)w_(4,0)-z3w                  |
{-3} | w_(5,0)t-xy2                        |
{-3} | w_(5,0)w+w_(3,0)^2z-xt2             |
{-3} | w_(5,0)y-zw2                        |
{-3} | w_(5,0)x-w_(4,0)z                   |
{-3} | w_(5,0)w_(2,1)-xzw                  |
{-3} | w_(5,0)w_(2,0)+yzw                  |
{-4} | w_(5,0)w_(2,6)+w_(2,1)yz2           |
{-4} | w_(5,0)w_(2,2)-xywt                 |
```

```
{-3} | w_(5,0)w_(3,0)+z2w                     |
{-4} | w_(5,0)w_(4,0)+w_(2,1)z3+xy2t          |
{-4} | w_(5,1)^2-w_(5,0)z2-w_(2,6)xt          |
{-4} | w_(5,0)w_(5,1)+xz3                     |
{-4} | w_(5,0)^2-w_(3,0)z3+y2zt               |
```

This Macaulay2 run on the homogeneous input produced an *affine A-algebra* presentation with only 8 new variables and 79 relations, with the first 5 again defining the input ring, and reasonably independent of characteristic.

The monomial ordering is again a product ordering (lex-over-input) But again there is no concept that the monomial ordering on the input should induce a monomial ordering on the output.

```
Macaulay2, version 1.4
with packages: ConwayPolynomials, Elimination, IntegralClosure, LLLBases,
               PrimaryDecomposition, ReesAlgebra, TangentCone

i1 : A=QQ[x,y,z,w]/ideal(x^2+z*w,y^3+x*w,x*w^3+z^3+y*w,y^2*w^4-x*y^2*z^2-w^3);

i2 : time ic=transpose gens gb presentation integralClosure A;
     -- used 0.837211 seconds

i3 : ic

o3 = {-2}  | x2+zw                                              |
     {-3}  | y3+xw                                              |
     {-4}  | xw3+z3+yw                                          |
     {-5}  | zw4-xz3-xyw                                        |
     {-6}  | y2w4-xy2z2-w3                                      |
     {-2}  | w_(2,0)w-z2                                        |
     {-3}  | w_(2,0)z+xw2+y                                     |
     {-5}  | w_(2,0)xy2-y2w3+w2                                 |
     {-5}  | w_(2,0)^2y2+xy2zw-xz                               |
     {-5}  | w_(2,0)^3y+xyz3-xy2w+x                             |
     {-6}  | w_(2,0)^4x-z6+yz3w-y2w2+w                          |
     {-12} | w_(2,0)^9-xz10w+3xyz7w2+6y2z7+4xz4w2+5yz4-y2zw+z |
```

The inhomogeneous version runs faster, and produces an *affine A-algebra* presentation with only 1 new variable and 12 relations.

Vasconcelos, in his book, implicitly suggests a different ordering. That is, one could easily view the first input relation as defining $z := -x^2/w$ and the second as defining $x := -y^3/w$. So it would make sense to change the order of z and y, expecting this to end up as an extension of $k[w]$ in the variable y, with this z and x as given.

So, let's go back and try this.

```
> ring a=0,(x,z,y,w),dp;
> ideal i=x2+zw,y3+xw,xw3+z3+yw,y2w4-xy2z2-w3;
> int time=timer;
> list nor=normal(i);nor;

// 'normal' created a list, say nor, of two elements.
// To see the list type
      nor;

// * nor[1] is a list of 1 ring(s).
// To access the i-th ring nor[1][i], give it a name, say Ri, and type
     def R1 = nor[1][1]; setring R1; norid; normap;
// For the other rings type first (if R is the name of your base ring)
     setring R;
// and then continue as for R1.
// Ri/norid is the affine algebra of the normalization of R/P_i where
// P_i is the i-th component of a decomposition of the input ideal id
// and normap the normalization map from R to Ri/norid.

// * nor[2] is a list of 1 ideal(s). Let ci be the last generator
// of the ideal nor[2][i]. Then the integral closure of R/P_i is
// generated as R-submodule of the total ring of fractions by
// 1/ci * nor[2][i].
[1]:
   [1]:
      //   characteristic : 0
//   number of vars : 14
//        block   1 : ordering dp
//                  : names    T(1) T(2) T(3) T(4) T(5) T(6) T(7) T(8) T(9) T(10)
//        block   2 : ordering dp
//                  : names    x z y w
//        block   3 : ordering C
[2]:
   [1]:
      _[1]=z3+yw
      _[2]=y2w2
      _[3]=zyw2
      _[4]=xyw2
      _[5]=z2w2
```

```
      _[6]=xzw2
      _[7]=z2y2w
      _[8]=xzy2w
      _[9]=xz2yw
      _[10]=xz2y2
      _[11]=w3
> timer-time;
0
> def R=nor[1][1];
> setring R;
> normap;
normap[1]=x
normap[2]=z
normap[3]=y
normap[4]=w
> option(redSB);
> ideal s=std(norid);s;
s[1]=x^2+z*w
s[2]=y^3+x*w
s[3]=x*w^3+z^3+y*w
s[4]=z*w^4-x*z^3-x*y*w
s[5]=y^2*w^4-x*z^2*y^2-w^3
s[6]=T(10)-y^2*w+1
s[7]=T(9)*y-y^2*w^2+w
s[8]=T(9)*x-x*y*w^2-y^2
s[9]=T(9)*w^2-x*z^2*y
s[10]=T(9)*z*w-z*y*w^3+x*y^2
s[11]=T(9)*z^3-z^3*y*w^2+y^2*w^3-w^2
s[12]=T(8)*w+T(9)*z^2-z^2*y*w^2
s[13]=T(8)*y-z^2
s[14]=T(8)*z-y^2*w^2+w
s[15]=T(7)*w+T(8)*x
s[16]=T(7)*z+x*y^2*w-x
s[17]=T(7)*x-y^2*w^2+w
s[18]=T(7)*y^2+T(9)*w
s[19]=T(6)*w-x*z
s[20]=T(6)*z+T(7)*y
s[21]=T(6)*x+z^2
s[22]=T(6)*y^2+T(9)*z^2-z^2*y*w^2
s[23]=T(5)*w-z^2
s[24]=T(5)*z+x*w^2+y
s[25]=T(5)*x+T(7)*y
s[26]=T(5)*y^2+T(8)*x
s[27]=T(4)*w-x*y
s[28]=T(4)*y+T(9)*z-z*y*w^2
s[29]=T(4)*z-T(6)*y
```

```
s[30]=T(4)*x+z*y
s[31]=T(3)*w-z*y
s[32]=T(3)*z-T(5)*y
s[33]=T(3)*x-T(6)*y
s[34]=T(3)*y^2+x*z
s[35]=T(2)*w-y^2
s[36]=T(2)*y+x
s[37]=T(2)*z-T(3)*y
s[38]=T(2)*x+T(9)*z-z*y*w^2
s[39]=T(1)+x
s[40]=T(9)^2-T(7)*y-x*z^2*y^2
s[41]=T(8)*T(9)+T(6)*y-z^2*w^2
s[42]=T(7)*T(9)+T(5)*y+x*z^2*w
s[43]=T(6)*T(9)-T(3)*y-x*z*y*w
s[44]=T(5)*T(9)+T(8)-z^2*y*w
s[45]=T(4)*T(9)-x*y^2*w+x
s[46]=T(3)*T(9)-z*y^2*w+z
s[47]=T(2)*T(9)+x*w^2+y
s[48]=T(8)^2-T(9)*z
s[49]=T(7)*T(8)+T(3)*y+x*z*y*w
s[50]=T(6)*T(8)-x*y^2*w+x
s[51]=T(5)*T(8)-z*y^2*w+z
s[52]=T(4)*T(8)+T(7)*y
s[53]=T(3)*T(8)+x*w^2+y
s[54]=T(2)*T(8)-T(5)*y
s[55]=T(7)^2-T(8)+z^2*y*w
s[56]=T(6)*T(7)-z*y^2*w+z
s[57]=T(5)*T(7)-T(6)+x*z*y^2
s[58]=T(4)*T(7)+x*w^2+y
s[59]=T(3)*T(7)-T(4)+z*w^2
s[60]=T(2)*T(7)+T(9)
s[61]=T(6)^2-x*w^2-y
s[62]=T(5)*T(6)+T(4)-z*w^2
s[63]=T(4)*T(6)+T(5)*y
s[64]=T(3)*T(6)-T(9)
s[65]=T(2)*T(6)-T(8)
s[66]=T(5)^2+T(3)+x*z*w
s[67]=T(4)*T(5)-T(9)
s[68]=T(3)*T(5)+T(2)+x*y*w
s[69]=T(2)*T(5)-T(7)
s[70]=T(4)^2+T(3)*y
s[71]=T(3)*T(4)-T(8)
s[72]=T(2)*T(4)-z
s[73]=T(3)^2-T(7)
s[74]=T(2)*T(3)+T(6)
s[75]=T(2)^2+T(4)
```

This clearly has almost the same result.

```
Macaulay2, version 1.4
with packages: ConwayPolynomials, Elimination, IntegralClosure, LLLBases,
               PrimaryDecomposition, ReesAlgebra, TangentCone

i1 : A=QQ[x,z,y,w]/ideal(x^2+z*w,y^3+x*w,x*w^3+z^3+y*w,y^2*w^4-x*y^2*z^2-w^3);

i2 : time ic=transpose gens gb presentation integralClosure A;
     -- used 0.769979 seconds
i3 : ic

o3 = {-2}  | x2+zw                                           |
     {-3}  | y3+xw                                           |
     {-4}  | xw3+z3+yw                                       |
     {-5}  | zw4-xz3-xyw                                     |
     {-6}  | y2w4-xz2y2-w3                                   |
     {-2}  | w_(2,3)w-z2                                     |
     {-3}  | w_(2,3)z+xw2+y                                  |
     {-5}  | w_(2,3)xy2-y2w3+w2                              |
     {-5}  | w_(2,3)^2y2+xzy2w-xz                            |
     {-5}  | w_(2,3)^3y+xz3y-xy2w+x                          |
     {-6}  | w_(2,3)^4x-z6+z3yw-y2w2+w                       |
     {-12} | w_(2,3)^9-xz10w+3xz7yw2+6z7y2+4xz4w2+5z4y-zy2w+z |
```

```
  ring a=11,(x,z,y,w),dp;
> ideal i=x2+zw,y3+xw,xw3+z3+yw,y2w4-xy2z2-w3;
> list norp=normalP(i,"withRing");norp;
//   characteristic : 11
//   number of vars : 4
//        block   1 : ordering dp
//                  : names    T(1) T(4)
//        block   2 : ordering dp
//                  : names    y w
//        block   3 : ordering C
[2]:
   [1]:
      _[1]=y2w
      _[2]=zyw
      _[3]=xyw
      _[4]=z2w
      _[5]=xzw
      _[6]=z2y2
      _[7]=xzy2
      _[8]=xz2y
      _[9]=w2
> def Rp=norp[1][1];
> setring Rp;
> normap;
normap[1]=-T(1)*y
normap[2]=-T(1)^3
normap[3]=y
normap[4]=w
> option(redSB);
> ideal s=std(norid);s;
s[1]=y^17+y^2*w^11-w^10
s[2]=T(4)*y^5+y^2*w^4-w^3
s[3]=T(4)*w^7-y^12
s[4]=T(1)*w-y^2
s[5]=T(1)*y^10-T(4)*w^6
s[6]=T(4)^2*w^3-T(1)*y^5+y^9
s[7]=T(1)*T(4)*y^3+y^2*w^3-w^2
s[8]=T(1)^2*y^3-T(4)^2*w^2-T(1)*y^7
s[9]=T(4)^3-T(1)-y^6*w+2*y^4
s[10]=T(1)^2*T(4)*y+y^2*w^2-w
s[11]=T(1)^3*y-T(4)^2*y^2*w^2-T(4)^2*w-T(1)*y^9
s[12]=T(1)^3*T(4)+y^3*w-y
s[13]=T(1)^4-T(4)^2*y^3*w-T(4)^2*y-T(4)*w^5
```

Usually `normalP` takes too long and reduces varaibles that just happen to show up linearly in some relation. But here it reduces z and x as we had hoped for. And it found a useful cube root of $-z$, namely $T(1)$, giving a reduced presentation showing indeed that y was integral over $k[w]$. Of course this is no longer a *strict affine a-algebra* presentation as described in chapter 3 of the SINGULAR book. But at least we can see now that this is an integral closure of a curve (the curve defined by s[1]).

3 Qth-power

My *qth-power algorithm* is meant to run this with structure. That usually means that one must think about the input and preprocess it so that both input and output will have information in them.

This is clearly homogeneous, maybe not immediately clear that it is a curve. Dehomogenize it by using $Z_{21} := z/t$, $X_{18} := x/t$, $W_{15} := w/t$, $Y_{11} := y/t$, and look at the divisors:

$$div(Z_{21}) = -21 \cdot P_\infty + 12 \cdot Q + 9 \cdot R$$

$$div(X_{18}) = -18 \cdot P_\infty + 5 \cdot Q + 13 \cdot R$$

$$div(W_{15}) = -15 \cdot P_\infty - 2 \cdot Q + 17 \cdot R$$

$$div(Y_{11}) = -11 \cdot P_\infty + 1 \cdot Q + 10 \cdot R$$

Maybe it is no surprise that there exists V_7 with

$$div(V_7) = -7 \cdot P_\infty + 4 \cdot Q + 3 \cdot R$$

and $Z_{21} = V_7^3$, $X_{18} = Y_{11}V_7$. The one-point version would use $U_{37} := W_{15}Y_{11}^2$ in place of W_{15}.

The two-point version can be done using:

$$Y_{11}^{17} + Y_{11}^2 W_{15}^{11} - W_{15}^{10} = 0;$$

while the one-point version can be done using:

$$Y_{11}^7 + Y_{11}^3 V_7 - V_7^{11} = 0.$$

```
i1 : loadPackage "QthPower";

i2 : wtr=matrix{{11,15}};

i3 : R=QQ[y,w,Weights=>entries weightGrevlex(wtr)];

i4 : I={y^17+y^2*w^11-w^10};

i5 : time ic=rationalIntegralClosure(wtr,R,I);

i6 : time ic1=minimization(ic);

i7 : toString ic1

o7 = matrix {{27,   17,   11,   15,    7}},
         QQ[p_7, p_12, p_14, p_15, p_16],

    {p_14^2-p_15*p_16,
     p_12*p_14-p_16^4,
     p_12*p_15-p_14*p_16^3,
     p_12^2-p_7*p_16,
     p_7*p_14-p_12*p_16^3,
     p_7*p_15-p_16^6,
     p_7*p_12+p_15^2*p_16^2-p_16,
     p_14*p_15^2*p_16+p_7*p_16^3-p_14,
     p_15^3*p_16+p_12*p_16^5-p_15,
     p_7^2+p_14*p_15*p_16^4-p_12}}
```

`minimization` doesn't flesh the reult out to a strict affine P-algebra, which would require variables for $p_{14}p_{15}$, p_{15}^2, and p_{15}^3.

4 Compact description

The following is a reasonably small alternative two-point description.

```
Macaulay2, version 1.4
with packages: ConwayPolynomials, Elimination, IntegralClosure, LLLBases,
               PrimaryDecomposition, ReesAlgebra, TangentCone

i1 : loadPackage "QthPower";

i2 : wtr=matrix{{7,4}};

i3 : R=QQ[u,s,Weights=>entries weightGrevlex(wtr)];

i4 : I={u^7-u^3*s^7-s^3};

i5 : time ic=rationalIntegralClosure(wtr,R,I);
     -- used 0.603232 seconds

i6 : toString ic

o6 = ({s^2,
     u*s^2,
   u^2*s^2,
   u^3*s,
   u^4*s,
   u^5,
   u^6},

{    p_0^2-p_1*p_6-p_2*p_6^11-p_5*p_6^6,
     p_0*p_1-p_2-p_3*p_6^11-p_6^6,
     p_0*p_2-p_0*p_6^6-p_3*p_6,
     p_0*p_3-p_1*p_6^6-p_4,
     p_0*p_4-p_2*p_6^6-p_5*p_6,
     p_0*p_5-p_3*p_6^6-p_6,
     p_1^2-p_0*p_6^5-p_3,
     p_1*p_2-p_1*p_6^6-p_4,
     p_1*p_3-p_2*p_6^5-p_5,
     p_1*p_4-p_3*p_6^6-p_6,
     p_1*p_5-p_0,
     p_2^2-p_2*p_6^6-p_5*p_6,
     p_2*p_3-p_3*p_6^6-p_6,
     p_2*p_4-p_0*p_6,
     p_2*p_5-p_1*p_6,
     p_3^2-p_0,
     p_3*p_4-p_1*p_6,
```

```
     p_3*p_5-p_2,
     p_4^2-p_2*p_6,
     p_4*p_5-p_3*p_6,
     p_5^2-p_4},
     QQ[p_0, p_1, p_2, p_3, p_4, p_5, p_6],
matrix {{34,  27,  24,  17,  14,   7,   4}})
```

But if one buys into the *units* viewpoint perhaps one should try using

$$V_7^{37} - U_{37}^4(U_{37} - 1)^3 = 0$$

with U_{37} a unit to get both one-point and two-point answers at the same time, though with a larger presentation.

5 Timings

Try the one-point version:

$$A := \overline{\mathbf{F}}[Y_{11}, V_7]/\langle Y_{11}^7 + Y_{11}^3 V_7 - v_7^{11} \rangle.$$

(* means that the integral closure mod q is larger than in characteristic 0, — means the method is not applicable, *** means it ran out of (time and/or storage) resources.)

char	*normal*	*normalP*	*icFract.*	*icFRacP*	*Normalis.*	*Int.Clos.*	*QthPower*	*Qth*
0	0	$---$	0.5	$---$	0.6	0.02	0.57	0
2	0	0	0.5	0.1	0.5	0.07	0.15	0
3	0	0	0.5	0.1	0.5	0.06	0.11	0
5	1	0	0.5	0.0	0.5	0.05	0.11	0
7	0	0	0.5	0.0	0.5	0.08	0.15	0
11	0	0	0.5	0.1	0.5	0.03	0.17	0
13	0	0	0.5	0.5	0.5	0.04	0.16	0
17	0	$***$	0.5	2.6	0.5	0.03	0.17	0
19	0	0	0.5	1.9	0.5	0.03	0.17	0
23	0	0	0.5	23.	0.5	0.03	0.20	0
29	0	0	0.5	8.	0.5	0.06	0.18	0
31	1	0	0.6	7.	0.5	0.07	0.19	0
$*37$	1	$***$	41.	404.	1.1	0.09	29.	0
41	1	9	0.5	$***$	0.5	0.02	0.21	0
101	1	$***$			0.5	0.08	0.27	0
4001	0				0.5	0.02	57.0	

The failings of both `icFracP` and `normalP` for even relatively small characteristics is based on poor computational implementation of the crucial $f^q \; mod \; I$ step. [That is, if I interrupt the computation, it will undoubtedly be at `preimage(Q,phi,L)` in `normalP` and `K=intersect(kernel f,U)` in `icFracP`.] I have no explanation for the glitch at $q = 17$ in `normalP`.

6 Timings

Try the two-point version:

$$A := \overline{\mathbf{F}}[Y_{11}, W_{15}]/\langle Y_{11}^{17} + Y_{11}^{2} W_{15}^{11} - W_{15}^{10} \rangle.$$

(* means that the integral closure mod q is larger than in characteristic 0, — means the method is not applicable, *** means it ran out of (time and/or storage) resources.)

char	*normal*	*normalP*	*icFract.*	*icFRacP*	*Normalis.*	*Int.Clos.*	*QthPower*	*Qth*
0	1	−−−		−−−	468	0.6	3.9	
2	1	1	∗∗∗	0.1	846	0.6	0.9	
3	1	0		0.0	396	0.6	0.7	
5	0	0		0.1		0.5	0.8	
7	1	0		0.2		0.6	0.9	
11	1	0		3.0		0.7	0.9	
13	1	0		1.0		0.7	0.9	
17	1	0		∗∗∗		0.5	1.1	
19	1	1				0.5	1.0	
23	1	72				0.5	1.1	
29	0	495				0.6	1.1	
31	1					0.6	1.1	
∗37	298					1.1	∗∗∗	
41	1					0.5		
101	1					0.6		
4001	0					0.5		

I guess `icFractions` is known to perform poorly when there are many steps between input and output, `Normalisation` has known problems with bad monomial orderings and large rational coefficients. Here the lex ordering seems to be the culprit in all characteristics.

7 Timings

Try the original non-homogeneous version:

$$A := \overline{\mathbf{F}}[x, y, z, w]/\langle x^2 + zw, y^3 + xw, xw^3 + z^3 + yw, y^2w^4 - xy^2z^2 - w^3\rangle.$$

(* means that the integral closure mod q is larger than in characteristic 0, — means the method is not applicable, *** means it ran out of (time and/or storage) resources.)

char	*normal*	*normalP*	*icFract.*	*icFRacP*	*Normalis.*	*Int.Clos.*	*QthPower*	*Qth*
0	0	−−−		−−−		−−−	−−−	−−−
2	0	0	0.5	0.2	* * *	−−−	−−−	−−−
3	1	0	0.5	0.1		−−−	−−−	−−−
5	0	0	0.6	0.3		−−−	−−−	−−−
7	0	1	0.5	107.		−−−	−−−	−−−
11	0	2	0.6			−−−	−−−	−−−
13	1	37				−−−	−−−	−−−
17	0	* * *				−−−	−−−	−−−
19	0					−−−	−−−	−−−
23	0					−−−	−−−	−−−
29	1					−−−	−−−	−−−
31	0					−−−	−−−	−−−
*37	351					−−−	−−−	−−−
41	0					−−−	−−−	−−−
101	1					−−−	−−−	−−−
4001	0					−−−	−−−	−−−

Could I have predicted that $q = 37$ would be different? Maybe from the computation of a conductor element over **Z**. Of course the real problem here is not so much the timings but rather what information the fractions and/or presentation give. It should be noted that it is easy to make up simple examples on which the qth-power algorithm my way is not applicable. One can take a curve and append a few extra functions and the related relations without simplifying back to the original curve. But this is exactly why I suggest preprocessing various inputs.

8 Timings

Try the original homogeneous version:

$A := \overline{\mathbf{F}}[x,y,z,w,t]/\langle x^2+zw, y^3+xwt, xw^3+z^3t+ywt^2, y^2w^4-xy^2z^2t-w^3t^3\rangle.$

(* means that the integral closure mod q is larger than in characteristic 0, — means the method is not applicable, *** means it ran out of (time and/or storage) resources.)

char	*normal*	*normalP*	*icFract.*	*icFRacP*	*Normalis.*	*Int.Clos.*	*QthPower*	*Qth*
0	5	−−−		−−−		−−−	−−−	−−−
2	5					−−−	−−−	−−−
3	4					−−−	−−−	−−−
5	5					−−−	−−−	−−−
7	5					−−−	−−−	−−−
11	5					−−−	−−−	−−−
13	5					−−−	−−−	−−−
17	4					−−−	−−−	−−−
19	5					−−−	−−−	−−−
23	5					−−−	−−−	−−−
29	5					−−−	−−−	−−−
31	5					−−−	−−−	−−−
*37	***					−−−	−−−	−−−
41	4					−−−	−−−	−−−
101	5					−−−	−−−	−−−
4001	5					−−−	−−−	−−−

And is it worth it for some reason to give up, say an order of magnitude in timing, to get a much more complicated answer? I certainly don't see the advantage. But then again, it makes absoultely no sense to homogenize weigthed problems; so why should it make sense to homogenize this?