gcd

May 20, 2020

[1]:
```
restart;
```

```
--loading configuration for package "FourTiTwo" from file
/home/hegland/.Macaulay2/init-FourTiTwo.m2
--loading configuration for package "Topcom" from file
/home/hegland/.Macaulay2/init-Topcom.m2
```

[6]:
```
--------------------
-- linear algebra --
--------------------
<< "solve Ax=b " << endl;
-- solve a linear system of equations
A = matrix "3,4;2,5" **RR
b = matrix "2;7" **RR   -- not "vector"!
<< "A= " << A << ",  b = " << b <<endl;
x = solve(A,b)
<< "solution x = " << x << ", residual r=Ax-b= " << A*x-b << endl;
```

```
solve Ax=b

A= | 3 4 |,  b = | 2 |
   | 2 5 |       | 7 |

solution x = | -2.57143 |, residual r=Ax-b= | 0 |
             | 2.42857  |                   | 0 |
```

[7]:
```
restart;
```

```
--loading configuration for package "FourTiTwo" from file
/home/hegland/.Macaulay2/init-FourTiTwo.m2
--loading configuration for package "Topcom" from file
/home/hegland/.Macaulay2/init-Topcom.m2
```

[7]:
```
-----------------
-- polynomials --
-----------------
<< "compute gcd(p,q)" << endl;
kk = QQ
```

```
P = kk[x]

p = (x-7/10)^2*(x-23/100)*(x^2+3/10)
q = (x-7/10)^3*(x-3/10)*(x^2+37/100)

g = gcd(p,q)
g/leadCoefficient(g)
```

```
compute gcd(p,q)


      2   7     49
o7 = x  - -x + ---
          5    100

o7 : P
```

[16]:
```
-----------------------------
-- code our own gcd for QQ --
-----------------------------

  -- replace polynomial division/remainder by S polynomial
  -- this version gives wrong results for RR and CC

<< "compute gcdx(p,q)" << endl;
kk = QQ
P = kk[x]

p = (x-7/10)^2*(x-23/100)*(x^2+3/10)
q = (x-7/10)^3*(x-3/10)*(x^2+37/100)

spoly = (p,q) -> (
   ddeg = first(degree(p)) - first(degree(q));
   lcp = leadCoefficient(p);
   lcq = leadCoefficient(q);
   return (lcq*p - lcp*x^ddeg * q)/max(abs(lcp),abs(lcq));
  )

-- gcd --
gcdx = (p,q) -> (
  if first(degree(p)) < first(degree(q)) then return gcdx(q,p)
    else s=spoly(p,q);if s==0 then return q else return gcdx(q,s);
  )
g = gcdx(p,q)
g/leadCoefficient(g)
```

```
compute gcdx(p,q)
```

```
       2   7     49
o16 = x  - -x + ---
           5    100

o16 : P
```

[25]:
```
-----------------------------
-- code our own gcd for RR --
-----------------------------

  -- replace stopping criterium for gcd, spoly is the same

<< "compute gcdx(p,q)" << endl;
kk = RR
P = kk[x]

p = (x-7/10)^2*(x-23/100)*(x^2+3/10)
q = (x-7/10)^3*(x-3/10)*(x^2+37/100)

spoly = (p,q) -> (
   ddeg = first(degree(p)) - first(degree(q));
   lcp = leadCoefficient(p);
   lcq = leadCoefficient(q);
   return (lcq*p - lcp*x^ddeg * q)/max(abs(lcp),abs(lcq));
  )

-- approximate gcd --
gcdx = (p,q) -> (
  if first(degree(p)) < first(degree(q)) then return gcdx(q,p)
    else s=spoly(p,q);if norm(s)<=1e-6 then return q else return gcdx(q,s);
  )
g = gcdx(p,q)
g/leadCoefficient(g)
```

```
compute gcdx(p,q)


       2
o25 = x  - 1.4x + .49

o25 : P
```

[1]:
```
restart;
loadPackage"NumericalAlgebraicGeometry"
```

```
--loading configuration for package "FourTiTwo" from file
/home/hegland/.Macaulay2/init-FourTiTwo.m2
```

```
--loading configuration for package "Topcom" from file
/home/hegland/.Macaulay2/init-Topcom.m2

--loading configuration for package "NumericalAlgebraicGeometry" from file
/home/hegland/.Macaulay2/init-NumericalAlgebraicGeometry.m2
--loading configuration for package "PHCpack" from file
/home/hegland/.Macaulay2/init-PHCpack.m2
--loading configuration for package "Bertini" from file
/home/hegland/.Macaulay2/init-Bertini.m2

o1 = NumericalAlgebraicGeometry

o1 : Package
```

[10]:
```
---------------------------------
-- using homotopy continuation --
-- recover gcv multiplying 0s  --
---------------------------------

<< "solve p(x)=0 and q(x)=0
" << endl;
kk = CC
P = kk[x]

p = (x-7/10)^2*(x-23/100)*(x^2+3/10)
q = (x-7/10)^3*(x-3/10)*(x^2+37/100)

s = solveSystem({p, q})

xsol_1 = (coordinates(s#0))_0
xsol_2 = (coordinates(s#1))_0

g = (x-xsol_1)*(x-xsol_2)
```

```
       2
o10 = x  + (- 1.4 + 3.23171e-14*ii)x + .49 - 2.26219e-14*ii

o10 : P
```