CoCoALib - Slug #675

Matrix determinant over multivariate poly ring

28 Mar 2015 09:16 - John Abbott

Status:	In Progress	**Start date:**	28 Mar 2015
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	10%
Category:	Tidying	**Estimated time:**	0.00 hour
Target version:	CoCoALib-1.0	**Spent time:**	0.75 hour

Description

CoCoA can be too slow and require too much RAM for computing determinant of a matrix of multivariate polynomials.

In my case M := MakeMatByRows(8,8, [x[i] | i in 1..64]);

Related issues:

Related to CoCoALib - Bug #15: Adjoint of a non-invertible matrix	**Closed**	**28 Oct 2011**
Related to CoCoALib - Slug #129: Better GCD	**New**	**15 Apr 2012**

History

#1 - 28 Mar 2015 09:18 - John Abbott

In my specific instance the direct algorithm (as alternating sum of products) would be best.

The problem with multivariate polynomials is the GCDs (needed when computing in the fraction field) which are currently computed slowly using G-bases.

#2 - 20 May 2015 10:22 - Anna Maria Bigatti

in the cocoa manual there is this example

```
/**/ discriminant((x+1)^20+2);
```

which is far too slow with the direct algorithm.
So I removed "|| (IsPolyRing(myR) && NumIndets(myR) > 1))" in

```
  void DenseMatImpl::myDet(RingElem& d) const
```

and it goes very fast. What should we do?

#3 - 22 May 2015 13:21 - John Abbott

- Status changed from New to In Progress

- % Done changed from 0 to 10

I have changed myDet in DenseMatrix.C so that it calls DetDirect only if there are at least 3 indets and the matrix is small (max 9x9) and the entries are all monomials.
This means it runs "fast" for the cases I need for Bettina, and should still be fast in other cases.

Not an ideal solution, but a tolerable hack for the time being.

#4 - 22 May 2015 13:45 - John Abbott

A Google search for "division free algorithm for determinant" produces several hits. Possibly something there may be useful for computing determinants of multivariate polynomials. I vaguely recall having seen the abstract of a paper which claimed to have a clever way of computing determinants using the n! summation by coalescing products (but recall no details now).