CoCoALib - Slug #1324

Improve RootBound

30 Sep 2019 19:40 - John Abbott

Status:	In Progress	Start date:	30 Sep 2019
Priority:	Normal	Due date:	
Assignee:		% Done:	10%
Category:	Improving	Estimated time:	0.00 hour
Target version:	CoCoALib-1.0	Spent time:	1.00 hour
Description			
It maybe possible to improve RootBound (in some cases) by using SqfreeFactor.			

Also the CoCoA-5 prototype GoodShiftForRootBound should be translated into C++.

History

#1 - 30 Sep 2019 19:42 - John Abbott

CAREFUL with the suggestions below: it is likely the RootBound is used internally when factorizing or computing sqfr factors.

If we can obtain quickly a factorization of f then RootBound(f) is just max of the RootBound for each factor.

I think that SqfreeFactor should be a good candidate for being quick enough.

It would be nice to have the GoodShiftForRootBound in C++; what name should it have? And what result should it give? Result could be the shift (BigInt) and the improved root bound (BigRat); should there also be the shifted poly?

#2 - 30 Oct 2019 21:44 - John Abbott

- Status changed from New to In Progress

- % Done changed from 0 to 10

Here is a family of examples where factorization may not work so well: let f be a product of x^{k-1} where the k values are chosen so that CoeffHeight(f) is small (*e.g.* 1). Sqfr factorization will then give a high power of x-1 and a factor with large coeffs.

I did try a couple of examples: there was some penalty, but it was much less than expected. Anyway, one could take min of RootBound(f) and max([RootBound(fac) | fac in SqfreeFactor(f).factors]). Though this is obviously more costly than just computing RootBound(f).

Perhaps the sqfr factors can be used only if they "small" compared to the original poly?