

CoCoALib - Feature #961

New function: ReducedGBasis

03 Nov 2016 08:50 - Anna Maria Bigatti

Status:	Closed	Start date:	03 Nov 2016
Priority:	Normal	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	New Function	Estimated time:	6.00 hours
Target version:	CoCoALib-0.99550 spring 2017	Spent time:	4.55 hours
Description Currently ReducedGBasis is implemented in CoCoA_5. Add it to CoCoALib, and add the flag HasReducedGBasis. Supposing we already have GBasis and later we compute ReducedGBasis, should we replace the TidyGens field? One should think so, but there are bad (non-homogeneous) examples like $(x-y^{100}, y-z-1)$.			
Related issues:			
Related to CoCoALib - Feature #957: New function: HasGBasis		Closed	27 Oct 2016
Related to CoCoALib - Feature #1016: ReducedGBasis for RingWeyl (and other no...		Closed	02 Mar 2017
Related to CoCoA-5 - Slug #405: ReducedGBasis not memorized in an ideal		Closed	09 Oct 2013

History

#1 - 23 Nov 2016 08:34 - Anna Maria Bigatti

- % Done changed from 0 to 70

I realized, after trying to implement it, that CoCoALib myDoGBasis already does it!!
In fact the only operation non done was to make it monic (for historical CoCoA-4 communication).
Now I made it monic, so that GBasis is actually identical to ReducedGBasis.
Is this a good idea or not?

#2 - 23 Nov 2016 08:35 - Anna Maria Bigatti

- Status changed from New to In Progress
- Target version changed from CoCoALib-0.99560 to CoCoALib-0.99550 spring 2017

#3 - 02 Dec 2016 09:13 - Anna Maria Bigatti

the GBasis is not reduced if the ring is not commutative
(ideal(x,dx) is homogeneous, so the algorithm does not expect degree drops)
For the moment I fixed this in CoCoA-5, but I should set the right flag inside cocoalib.

#4 - 12 Jan 2017 15:12 - Anna Maria Bigatti

- Related to Feature #957: New function: HasGBasis added

#5 - 02 Mar 2017 11:24 - John Abbott

I prefer to have two distinct functions **GBasis** and **ReducedGBasis** because:

- the name **ReducedGBasis** gives a guarantee for the future... the caller knows that it should return a result with known properties
- the name **GBasis** gives the caller a potentially weaker guarantee, and would allow us (as developers) the freedom to return non reduced bases (e.g. with integer coeffs rather than made monic)

Also I think someone reading code which uses ReducedGBasis knows that the result should be "nice".

#6 - 02 Mar 2017 13:46 - Anna Maria Bigatti

John Abbott wrote:

I prefer to have two distinct functions **GBasis** and **ReducedGBasis** because:

done.
Should pass it to cocoa-5.

#7 - 29 Mar 2017 18:21 - Anna Maria Bigatti

- Status changed from In Progress to Closed
- % Done changed from 70 to 100

In conclusion: all done for the commutative case.
To do for non-commutative case (new issue).
Closing this one.

#8 - 29 Mar 2017 18:22 - Anna Maria Bigatti

- Related to Feature #1016: ReducedGBasis for RingWeyl (and other non-commutative rings) added

#9 - 21 Apr 2017 11:09 - Anna Maria Bigatti

- Status changed from Closed to Resolved

Found bug in interreduction.

```
ReducedGBasis(ideal(x*y, y^3+x, y^3));
```

(thanks to strange error in GroebnerFan! <https://cocoa.dima.unige.it/redmine/issues/780#note-6>)

Fixing it.

#10 - 21 Apr 2017 15:14 - Anna Maria Bigatti

Subtle.... I think there were two problems: I fixed one (now repeating the cycle if a new LPP is found during interreduction), but the bug is still there.
Now I think the problem is a dangling iterator (aaaargh!).

#11 - 21 Apr 2017 15:32 - Anna Maria Bigatti

I checked the example in <https://cocoa.dima.unige.it/redmine/issues/418>.
I convinced myself that this cannot happen in the final interreduction of a GB computation, i.e. no new LPP can appear at this stage (its a GB!!).
So my earlier suspicion was wrong.

Also the suspicion of a dangly iterator was wrong: it just happened that, in the unlikely event that **the first** poly in the GB has to be removed, the

second is skipped by the ++it in the for loop. I put the increment in the right place of the cycle, and it works.

#12 - 21 Apr 2017 17:08 - Anna Maria Bigatti

- *Status changed from Resolved to Closed*

#13 - 27 Apr 2017 14:56 - Anna Maria Bigatti

- *Related to Slug #405: ReducedGBasis not memorized in an ideal added*