

CoCoALib - Feature #143

Buchberger-Moeller (parent task)

01 May 2012 10:17 - John Abbott

Status:	In Progress	Start date:	01 May 2012
Priority:	Urgent	Due date:	
Assignee:	John Abbott	% Done:	38%
Category:	New Function	Estimated time:	0.00 hour
Target version:	CoCoALib-1.0	Spent time:	18.45 hours
Description			
Robbiano wants to have Buchberger-Moeller in C5. It makes most sense to implement it in CoCoALib and then make it visible in C5. There are several versions to implement: these are in the subtasks.			
Subtasks:			
Feature # 144: Buchberger-Moeller: generic impl			Closed
Feature # 145: Buchberger-Moeller: fast modular impl			New
Feature # 147: Buchberger-Moeller: impl via modular reduction			In Progress
Related issues:			
Related to CoCoALib - Feature #146: Buchberger-Moeller: input conversions		New	01 May 2012
Related to CoCoALib - Feature #123: IdealOfPoints with generic coeffs		New	04 Apr 2012

History

#1 - 28 Jan 2013 08:09 - Anna Maria Bigatti

- Category set to New Function

#2 - 08 Feb 2013 18:00 - John Abbott

- Status changed from New to In Progress

- Assignee set to John Abbott

- Target version set to CoCoALib-0.9953

#3 - 08 Apr 2013 12:10 - John Abbott

What output do we want or need from the Buchberger-Moeller implementations?

1. generators of the ideal (list of polynomials)
2. separators (list of polynomials)
3. quotient basis (list of PPs, or perhaps list of polynomials)

The definition of *separator* in the case of projective points is not unique. In C4 we opted for polynomials which evaluate to 1 on the given representatives; should we continue to use this definition in CoCoA-5? Or perhaps there should no way to compute the separators in this case?

Opinions? Comments?

#4 - 08 Apr 2013 12:28 - John Abbott

JAA proposes offering two functions **IdealOfPoints** (for the affine case) and **IdealOfProjectivePoints** (for the projective case) which produce *ideals*.

In CoCoALib (and perhaps also in CoCoA-5) we could offer further functions for computing a complete result (presumably represented as a RECORD). In C4 these had cumbersome names such as IdealAndSeparatorsOfProjectivePoints. JAA proposes simpler names such as BuchbergerMoeller and ProjectiveBuchbergerMoeller. In the result JAA prefers a list of generators rather than an ideal -- it is quite easy for a user to

produce an ideal from a list of generators (or *vice versa*).

Opinions?

#5 - 08 Apr 2013 20:58 - John Abbott

My old impls in CoCoA-4 returned a matrix as the result: the entries in the matrix are the coeffs of the basis elements and of the separators.

The result also contained a list of PPs so that the appropriate polys could be reconstructed easily.

JAA now thinks that this is probably the most sensible type of value to return (for the **internal** BM fns that actually do the work).

The conversion from matrix to list of polys is straightforward, and can be effected by some auxiliary fns.

#6 - 29 May 2013 16:51 - John Abbott

- Target version changed from CoCoALib-0.9953 to CoCoALib-0.99534 Seoul14

The stopgap impls will have to suffice for 0.9953/CoCoA School as there's no chance of completing a proper impl in time.

#7 - 29 Oct 2013 15:17 - Anna Maria Bigatti

- Target version changed from CoCoALib-0.99534 Seoul14 to CoCoALib-0.99532

#8 - 03 Apr 2014 11:29 - John Abbott

- Target version changed from CoCoALib-0.99532 to CoCoALib-0.99533 Easter14

#9 - 04 Apr 2014 17:45 - John Abbott

- Target version changed from CoCoALib-0.99533 Easter14 to CoCoALib-0.99534 Seoul14

#10 - 10 Jul 2014 14:16 - John Abbott

- Target version changed from CoCoALib-0.99534 Seoul14 to CoCoALib-1.0