

## CoCoA-5 - Support #977

### "universal denominator" (related with GroebnerFanIdeals)

21 Nov 2016 09:33 - Anna Maria Bigatti

<b>Status:</b>	In Progress	<b>Start date:</b>	17 Nov 2016
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	20%
<b>Category:</b>	CoCoA-4 function to be added	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	CoCoA-5.4.2	<b>Spent time:</b>	1.65 hour
<b>Description</b>			
[Coming from our recent work on MinPoly] It would be interesting to investigate the "universal denominator" of an ideal.			
[21-nov-2016: separated this from issue about GroebnerFanIdeals output]			
<b>Related issues:</b>			
Related to CoCoA-5 - Feature #903: New function CallOnGroebnerFanIdeals: call...		<b>Closed</b>	<b>04 Jul 2016</b>
Related to CoCoA-5 - Feature #978: CommonDenom: for polys and lists?		<b>Closed</b>	<b>21 Nov 2016</b>
Related to CoCoALib - Feature #979: SmallestNonDivisor -- new fn		<b>Closed</b>	<b>21 Nov 2016</b>
Related to CoCoA-5 - Slug #405: ReducedGBasis not memorized in an ideal		<b>Closed</b>	<b>09 Oct 2013</b>
Related to CoCoA-5 - Design #984: GroebnerFanIdeals: order matrices sometimes...		<b>New</b>	<b>26 Nov 2016</b>
Copied from CoCoA-5 - Support #973: GroebnerFanIdeals: verbosity and output s...		<b>Closed</b>	<b>17 Nov 2016</b>

### History

#### #1 - 21 Nov 2016 09:33 - Anna Maria Bigatti

- Copied from Support #973: GroebnerFanIdeals: verbosity and output style added

#### #2 - 21 Nov 2016 09:35 - Anna Maria Bigatti

By John Abbott:

Does this code correctly compute the "universal denominator" for the ideal I?

```
use P:=QQ[x,y,z];
I := ideal(x^4-3*y^2*z+x*y*z-2, y^2-2*z+5, z^2-2*x+7*y);
println "IsZeroDim: ", IsZeroDim(I);
GF := GroebnerFanIdeals(I);
J := [ReducedGBasis(I) | I in GF];

define DEN(L)
  if type(L) = RINGELEM then
    return lcm([AsINT(den(c)) | c in coefficients(L)]);
  endif;
  return lcm([DEN(f) | f in L]);
enddefine; -- DEN

DENJ := [DEN(RGB) | RGB in J];

N := lcm(DENJ);
FloatStr(N);
facs := SmoothFactor(N,10000000);
println "RemainingFactor = ", FloatStr(facs.RemainingFactor);
IsProbPrime(facs.RemainingFactor); --> false

println "Known bad primes: ", facs.factors;
```

If the code is correct then the "universal denominator" is very large -- I have already made some attempt to make it smaller (the first ideal I tried, not much more complicated than this one, gave a "universal denominator" with almost 7000 digits!)

**#3 - 21 Nov 2016 09:36 - Anna Maria Bigatti**

By John Abbott:

Robbiano suggested that it could be interesting to find the first (or at least a smallish) good prime.

An interim solution could be to use SmoothFactor: call it first with a limit of (say) 100, and if no good primes are found, double the limit and call SmoothFactor again. Asymptotically this is not so efficient, but it seems unlikely that one will often encounter such highly factorizable numbers that several iterations of the loop will be needed.

**#4 - 21 Nov 2016 09:47 - Anna Maria Bigatti**

- Subject changed from GroebnerFanIdeals: universal denominator to GroebnerFanIdeals and "universal denominator"

I copied John's code for further experiments in our working dir MinPoly2016, file Deltone.cocoa5. CVS-ed

**#5 - 21 Nov 2016 09:48 - Anna Maria Bigatti**

- Related to Feature #903: New function CallOnGroebnerFanIdeals: call function on GFan ideals added

**#6 - 21 Nov 2016 14:34 - John Abbott**

- Related to Feature #978: CommonDenom: for polys and lists? added

**#7 - 21 Nov 2016 17:51 - John Abbott**

- Related to Feature #979: SmallestNonDivisor -- new fn added

**#8 - 22 Nov 2016 07:42 - Anna Maria Bigatti**

- % Done changed from 10 to 20

The other functions should be called UniversalDenominator(l) and SmallestExcellentPrime(l).

**#9 - 22 Nov 2016 15:34 - John Abbott**

I have implemented SmallestNonDivisor (see [#979](#)).

I'm still hoping to find a better name than UniversalDenominator; I feel that it ought to contain "Groebner" somewhere, but then the name becomes even longer :-/

**#10 - 23 Nov 2016 07:41 - Anna Maria Bigatti**

John Abbott wrote:

I'm still hoping to find a better name than UniversalDenominator; I feel that it ought to contain "Groebner" somewhere, but then the name becomes even longer :-/

UniversalRGBDenominator?

#11 - 23 Nov 2016 12:19 - John Abbott

The name **UniversalRGBDenominator** is OK for me.

I note that there are already two "denominator" functions: **den** and **CommonDenom**.  
So we have not been consistent about use of an abbreviated form :-/

Also using the abbreviation **RGB** to mean "reduced Groebner basis" is inconsistent with **ReducedGBasis**.  
We could offer **RGBasis** but that is definitely less clear than **ReducedGBasis**.

How about **UniversalRGBDenom**?

#12 - 23 Nov 2016 13:32 - Anna Maria Bigatti

John Abbott wrote:

The name **UniversalRGBDenominator** is OK for me.

I note that there are already two "denominator" functions: **den** and **CommonDenom**.  
So we have not been consistent about use of an abbreviated form :-/

That's not too bad: the first is a single word.

Also using the abbreviation **RGB** to mean "reduced Groebner basis" is inconsistent with **ReducedGBasis**.  
We could offer **RGBasis** but that is definitely less clear than **ReducedGBasis**.

Now **ReducedGBasis** (I'm about to check in) is **identical** to **GBasis**.

(I just needed to make it monic). This is handy, but I'm not 100% sure we don't risk badly in some examples (obviously there are trivial pathological example, but in general?)

How about **UniversalRGBDenom**?

perfect!

**#13 - 23 Nov 2016 16:44 - John Abbott**

I wonder how "universal" UniversalRGBDenom is? Does it also apply to all Janet Bases? And Pommaret Basis (if it exists)? Border bases may be different; though perhaps they are covered if they contain a G-basis?

**#14 - 24 Nov 2016 13:02 - John Abbott**

I now think that the "universal" denominator is valid for any monic basis which contains a RGB as subset (since the other elements of the basis are of the form LPP - NF, and we know that the coeffs of the LPP are in  $\mathbb{Z}\langle \Delta \rangle$ ).

I presume it is not true in general for border bases, since there are ideals having border bases which do not contain a RGB as subset. Nevertheless there are only finitely many distinct border bases (assuming we require the set of PPs outside the span of the LPPs of the basis elements to be factor-closed or connected to 1), so we could also define a "universal denominator" for border bases -- this will be a (possibly trivial) multiple of the universal RGB denominator.

**#15 - 24 Nov 2016 13:05 - John Abbott**

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

**#16 - 26 Nov 2016 17:52 - John Abbott**

- Related to Design #984: *GroebnerFanIdeals: order matrices sometimes have "large" entries added*

**#17 - 26 Apr 2017 16:58 - Anna Maria Bigatti**

- Target version changed from CoCoA-5.2.0 spring 2017 to CoCoA-5.2.2

I think we are going to do more work on this topic. Postponing to next release.

**#18 - 09 Oct 2017 12:53 - Anna Maria Bigatti**

Should UniversalDenominator return INT or RingElem?  
We should also choose its name: UniversalDen?

Currently UniversalRGBDenom returns a RingElem, used to be INT.  
Related functions: CommonDenom, DEN

**#19 - 09 Oct 2017 15:19 - John Abbott**

JAA prefers INT to RINGELEM (at least for the moment).

If the coeff ring is not QQ but is FracField then I'm not sure how sensible it is to make a GFan.  
Shouldn't it be a Comprehensive GFan? Or something similar?

Anyway INT is more convenient for further operations (at least on the examples we have tried so far).

**#20 - 09 Oct 2017 15:23 - Anna Maria Bigatti**

John Abbott wrote:

JAA prefers INT to RINGELEM (at least for the moment).

OK, I'll change it back.

If the coeff ring is not QQ but is FracField then I'm not sure how sensible it is to make a GFan. Shouldn't it be a Comprehensive GFan? Or something similar?

Not really, that's another setting.

Anyway INT is more convenient for further operations (at least on the examples we have tried so far).

That's true (that's why I realized I made the change ;-)

**#21 - 15 Nov 2017 17:35 - Anna Maria Bigatti**

- Subject changed from GroebnerFanIdeals and "universal denominator" to "universal denominator" (related with GroebnerFanIdeals)
- Target version changed from CoCoA-5.2.2 to CoCoA-5.2.4

**#22 - 16 Jan 2018 16:11 - John Abbott**

I wonder whether it might be nice to have a function which returns a "partly factorized" universal denom. I used the partly factorized form when trying to understand the prime factors of the UD of the large example in the paper. Actually we are interested in the radical of the UD rather than the UD itself.

This is related to ~~GGDFreeBasis~~ **CoprimeFactorBasis**.

**#23 - 25 Jul 2018 15:35 - John Abbott**

- Target version changed from CoCoA-5.2.4 to CoCoA-5.3.0

**#24 - 01 Oct 2019 14:30 - John Abbott**

- Target version changed from CoCoA-5.3.0 to CoCoA-5.4.0

**#25 - 09 Jan 2020 20:39 - John Abbott**

What is the status of this issue? It is written in a way which suggests that an implementation exists somewhere, but I do not seem to have it.

**#26 - 03 Feb 2022 19:11 - John Abbott**

- Target version changed from CoCoA-5.4.0 to CoCoA-5.4.2