

## CoCoALib - Design #832

### Generalize grading matrix

04 Dec 2015 15:08 - Anna Maria Bigatti

<b>Status:</b>	New	<b>Start date:</b>	04 Dec 2015
<b>Priority:</b>	Low	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	20%
<b>Category:</b>	Data Structures	<b>Estimated time:</b>	6.00 hours
<b>Target version:</b>	CoCoALib-1.0	<b>Spent time:</b>	0.95 hour
<b>Description</b>			
Currently matrices for gradings must be full rank and have no negative entries. In principle we could accept negative entries and "redundant" weights just adding to the PPOrdering a "converting matrix" from the internal non-negative format to any format. Probably not so difficult, but have no requests in all these years, so I'm setting this to low-priority.			
<b>Related issues:</b>			
Related to CoCoALib - Bug #1789: GradingMat with negative weights should com...		<b>Closed</b>	<b>12 Mar 2024</b>
Related to CoCoA-5 - Feature #823: NewPolyRing with weights -- better interface?		<b>Closed</b>	<b>26 Nov 2015</b>

### History

#### #1 - 14 Jun 2018 21:04 - John Abbott

I have just started working on an implementation of isomorphic-to-subring. For example, given  $QQ[x,y,z]$  it should be possible to create an isomorphic copy of the subring generated by  $z$ : effectively a poly ring  $QQ[\#1]$  is created together with two homomorphisms. One hom sends  $x \mapsto 0$  and  $y \mapsto 0$  but  $z \mapsto \#1$ ; the reverse hom sends  $\#1 \mapsto z$ .

The question is what term-ordering to put on the newly created ring. The default is to have an ordering which is the restriction of the term-ordering in the big ring. However there seems to be a "problem" with gradings.

Suppose the grading of  $z$  is  $(0,1)$ ; this means that a compatible grading on  $Q[\#1]$  also has to be  $(0,1)$  but the ring has only 1 indet, and we have supposed that the term-ordering is compatible with the grading **and** the term-ordering is given by a square matrix... not possible in this case! What to do? Just forget the grading?

#### #2 - 13 Mar 2024 20:48 - John Abbott

- Related to Bug #1789: GradingMat with negative weights should complain (or deal with them properly!!) added

#### #3 - 13 Mar 2024 21:11 - John Abbott

My current preference is to allow gradings with negative weights, and possibly linearly dependent weight matrices.

This would mean that the internal repr of the grading differs from that which the user expects to see, but the two are in 1--1 correspondence (via a matrix, not nec invertible!).

Taking the example at the end of comment 1 ([#832#note-1](#)). The internal grading would be  $[1]$  for  $z$ , but this would be mapped into  $ZZ^2$  by multiplying by the  $2 \times 1$  matrix  $ColMat([0,1])$ . This does make functions such as `wdeg` more complicated, and slower, but that may not matter. Also there would have to be functions to map from the grading the user expects to the internal representation, and *vice versa*.

#### #4 - 14 Mar 2024 11:22 - Anna Maria Bigatti

- Related to Feature #823: NewPolyRing with weights -- better interface? added

#5 - 14 Mar 2024 11:29 - Anna Maria Bigatti

- % Done changed from 0 to 20

John Abbott wrote:

My current preference is to allow gradings with negative weights, and possibly linearly dependent weight matrices.

This would mean that the internal repr of the grading differs from that which the user expects to see, but the two are in 1--1 correspondence (via a matrix, not nec invertible!).

Taking the example at the end of comment 1 ([#832#note-1](#)). The internal grading would be [1] for  $z$ , but this would be mapped into  $\mathbb{Z}^2$  by multiplying by the  $2 \times 1$  matrix  $\text{ColMat}([0,1])$ . This does make functions such as `wdeg` more complicated, and slower, but that may not matter. Also there would have to be functions to map from the grading the user expects to the internal representation, and *vice versa*.

I totally agree, just a bit tedious to implement.

Meanwhile, I'm making a CoCoA-5 preliminary-package with Robbiano, to explore the applications/consequences of such gradings.