

## CoCoALib - Design #1460

### Name of TensorMat; change to KroneckerProd?

23 May 2020 14:43 - John Abbott

<b>Status:</b>	Closed	<b>Start date:</b>	23 May 2020
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	John Abbott	<b>% Done:</b>	100%
<b>Category:</b>	Renaming	<b>Estimated time:</b>	1.33 hour
<b>Target version:</b>	CoCoALib-0.99800	<b>Spent time:</b>	1.35 hour
<b>Description</b>			
CoCoALib has a function called TensorMat. The operation is also known as <b>Kronecker product</b> of matrices. The wikipedia entry suggests that KroneckerProd would be a better name.  What do you think?			

### History

#### #1 - 23 May 2020 14:45 - John Abbott

- Subject changed from Name of TensorMat to Name of TensorMat; change to KroneckerProd?

I re-implemented the fn because I had forgotten (or didn't realise) that it was already in CoCoALib.  
At the very least, there should be mention of "Kronecker product" in the documentation.

We could have both names. My current preference is to use the name **KroneckerProd**, and make **TensorMat** obsolescent.

#### #2 - 25 May 2020 09:32 - John Abbott

Another matter is **where** to put the defn of the function.  
Currently TensorMat is defined in MatrixSpecial.C; I had expected to find it in MatrixOps.

Most of the other fns defined in MatrixSpecial are pseudo-ctors.  
Is KroneckerProd a pseudo-ctor or an op? At the moment I think it is an op.

Maybe the original idea was to impl a pseudo-ctor where the original matrices would be stored (or pointed to), and then each entry computed on demand. The advantage is that almost no extra space is needed to store the KroneckerProd. This disadvantages are that the original matrices must be stored (at least via a ref count increment?) and that repeated access to an entry requires repeated computation of the value (just a multiplication). It seems hard to imagine a situation where the "lazy" impl really brings a real benefit (except perhaps for sparse matrices?)

At the moment I favour viewing it as an op (rather than a "smart" lazy ctor), so think that it is better moved to MatrixOps.

Comments?

#### #3 - 25 May 2020 10:10 - Anna Maria Bigatti

John Abbott wrote:

Another matter is **where** to put the defn of the function.

Currently TensorMat is defined in MatrixSpecial.C; I had expected to find it in MatrixOps.

I agree. I expect the function was older than the file.

Most of the other fns defined in MatrixSpecial are pseudo-ctors.  
Is KroneckerProd a pseudo-ctor or an op? At the moment I think it is an op.

Maybe most of them are op's

#### **#4 - 06 Oct 2020 15:16 - John Abbott**

- Status changed from New to Feedback
- Assignee set to John Abbott
- % Done changed from 0 to 90

This has already been done.  
TensorMat is now in obsolete.C

Was this already in 0.99710?

#### **#5 - 29 Oct 2020 15:01 - John Abbott**

- Status changed from Feedback to Closed
- % Done changed from 90 to 100
- Estimated time set to 1.33 h

This has been in feedback for a month or so. Closing:

Conclusion: fn name changed KroneckerProd, impl creates new dense matrix, impl is in **MatrixOps-KroneckerProd.C**  
Old fn TensorMat moved to obsolete.C

**BTW** KroneckerProd requires both mats to be over same ring (see issue [#635](#)).

**PS** I suppose I could have implemented the "view" version, and then made KroneckerProd use the view as initializer for a dense mat... (can't be bothered now)