Quick solvers for endomorphisms of modules

Chris Liu, Colorado State University
Joint work with James B. Wilson and Josh Maglione

April 29, 2023

Simultaneous Sylvester System

Given $A_i \in K^{s\times b},\ B_i \in K^{a\times t},\ C_i \in K^{a\times b}$

Find X, Y satisfying $\forall\ i \in \{1, \ldots, c\}$

$$X \; A_i + B_i \; Y = C_i$$

Endomorphism of modules

- Ω a matrix algebra over K with n generators.

Endomorphism of modules

- Ω a matrix algebra over K with n generators.
- $M = \langle M_1, \ldots, M_n \rangle$ an Ω-module, $M_i \in K^{d \times d}$.

Endomorphism of modules

- Ω a matrix algebra over K with n generators.
- $M = \langle M_1, \ldots, M_n \rangle$ an Ω-module, $M_i \in K^{d \times d}$.

$$\begin{aligned} \mathrm{End}_\Omega(M) &= \{X \in \mathrm{End}_K(M) \mid \forall A \in \Omega, XA = AX\} \\ &= \{X \in K^{d \times d} \mid \forall i \in [n], XM_i = M_i X\} \end{aligned}$$

Other Applications

- **Adjoints**
  $t : K^n \times K^n \to K^n$ bilinear, $t(u, v) = (u^\intercal T_1 v, \ldots, u^\intercal T_n v)$

$$\begin{aligned}\mathrm{Adj}(t) &= \{(\varphi, \psi) \in \mathrm{End}(K^n) \times \mathrm{End}(K^n)^{\mathsf{op}} \mid t(\varphi(\cdot), \cdot) = t(\cdot, \psi(\cdot))\} \\ &= \{(X, Y) \in K^{n\times n} \times K^{n\times n} \mid \forall i \in [n], T_i X = Y^\intercal T_i\}\end{aligned}$$

Other Applications

- **Adjoints**
  $t : K^n \times K^n \to K^n$ bilinear, $t(u, v) = (u^\intercal T_1 v, \ldots, u^\intercal T_n v)$

$$\begin{aligned}\mathrm{Adj}(t) &= \{(\varphi, \psi) \in \mathrm{End}(K^n) \times \mathrm{End}(K^n)^{\mathsf{op}} \mid t(\varphi(\cdot), \cdot) = t(\cdot, \psi(\cdot))\} \\ &= \{(X, Y) \in K^{n\times n} \times K^{n\times n} \mid \forall i \in [n], T_i X = Y^\intercal T_i\}\end{aligned}$$

- **Wedderburn complements**

$$\left\langle \begin{bmatrix} B_i & C_i \\ & A_i \end{bmatrix} \right\rangle \sim \left\langle \begin{bmatrix} B_i & \\ & A_i \end{bmatrix} \right\rangle$$

if there exists X such that $XA_i - B_iX = C_i$

Other Applications

- **Adjoints**
  $t : K^n \times K^n \to K^n$ bilinear, $t(u,v) = (u^\intercal T_1 v, \ldots, u^\intercal T_n v)$

$$\begin{aligned}\mathrm{Adj}(t) &= \{(\varphi,\psi) \in \mathrm{End}(K^n) \times \mathrm{End}(K^n)^{\mathsf{op}} \mid t(\varphi(\cdot),\cdot) = t(\cdot,\psi(\cdot))\} \\ &= \{(X,Y) \in K^{n\times n} \times K^{n\times n} \mid \forall i \in [n], T_i X = Y^\intercal T_i\}\end{aligned}$$

- **Wedderburn complements**

$$\left\langle \begin{bmatrix} B_i & C_i \\ & A_i \end{bmatrix} \right\rangle \sim \left\langle \begin{bmatrix} B_i & \\ & A_i \end{bmatrix} \right\rangle$$

if there exists X such that $XA_i - B_iX = C_i$

- Engineering applications in control theory, PDEs, and robotics

$A \in K^{s\times b\times c}, B \in K^{a\times t\times c}, C \in K^{a\times b\times c}$

$(\forall i)\ XA_i + B_iY = C_i$

$$
\left[\begin{array}{c|c}
\begin{matrix} A^{\intercal}_{*1*} & & \\ & \ddots & \\ & & A^{\intercal}_{*1*} \end{matrix} & \begin{matrix} B^{\intercal}_{1**} & & \\ \vdots & & \\ B^{\intercal}_{a**} & & \end{matrix} \\
\hline
\begin{matrix} A^{\intercal}_{*2*} & & \\ & \ddots & \\ & & A^{\intercal}_{*2*} \end{matrix} & \begin{matrix} & B^{\intercal}_{1**} & \\ & \vdots & \\ & B^{\intercal}_{a**} & \end{matrix} \\
\hline
\vdots & \ddots \\
\hline
\begin{matrix} A^{\intercal}_{*b*} & & \\ & \ddots & \\ & & A^{\intercal}_{*b*} \end{matrix} & \begin{matrix} & & B^{\intercal}_{1**} \\ & & \vdots \\ & & B^{\intercal}_{a**} \end{matrix}
\end{array}\right]
\left[\begin{array}{c}
X_{11} \\ \vdots \\ X_{1s} \\ \hline \vdots \\ \hline X_{a1} \\ \vdots \\ X_{as} \\ \hline Y_{11} \\ \vdots \\ Y_{t1} \\ \hline \vdots \\ \hline Y_{1b} \\ \vdots \\ Y_{tb}
\end{array}\right]
=
\left[\begin{array}{c}
C_{111} \\ \vdots \\ C_{11c} \\ \hline \vdots \\ \hline C_{a11} \\ \vdots \\ C_{a1c} \\ \hline \vdots \\ \hline C_{ab1} \\ \vdots \\ C_{abc}
\end{array}\right].
$$

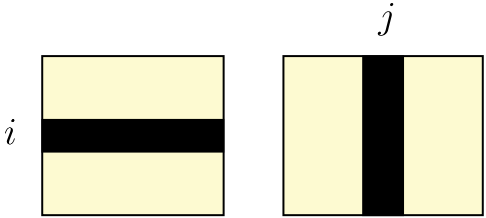
- $O(n^6)$, breaks on my laptop after $n = 50$
- Schneider conjectures can do no better than $O(n^6)$
- $n = 1000$ means 10^{18} FLOPs, exascale computing

Squeezing 5 indices into a single matrix should be a crime.

Squeezing 5 indices into a single matrix should be a crime.

Came from a tensor problem, discovered as a tensor solution, implement as a tensor algorithm.

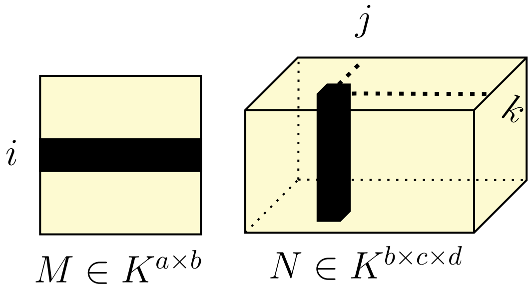
Tensor Notation



$M \in K^{a \times b}$ $\qquad$ $N \in K^{b \times c}$

$$(MN)_{ij} = (M \times_b N)_{ij} = \sum_{k=1}^{b} M_{ik} N_{kj}$$

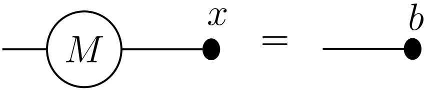
Tensor Notation



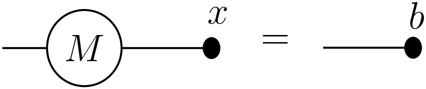
$$(M \times_b N)_{ijk} = \sum_{l=1}^{b} M_{il} N_{ljk}$$

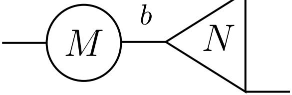
Simultaneous Sylvester System: $X \times_s A + B \times_t Y = C$

Tensor Network Diagrams

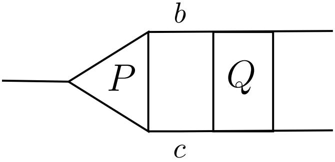


Tensor Network Diagrams

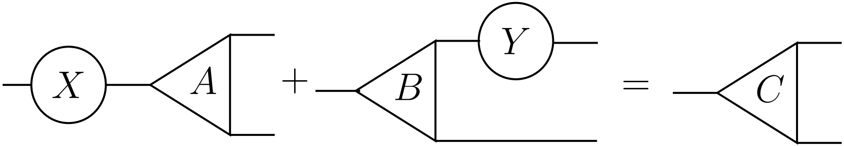




$M \times_b N$



$P \times_{bc} Q$


X
A
+
B
Y
=
C

Sylvester Equation (1884)

$$XA + BX = C$$

Sylvester Equation (1884)

$$
\left[\begin{array}{ccc|c|ccc}
a_{11}+b_{11} & \cdots & b_{1n} & & a_{1n} & & \\
\vdots & \ddots & \vdots & \cdots & & \ddots & \\
b_{n1} & \cdots & a_{11}+b_{nn} & & & & a_{1n} \\
\hline
 & & & \ddots & & & \\
\hline
a_{n1} & & & & a_{nn}+b_{11} & \cdots & b_{1n} \\
 & \ddots & & \cdots & \vdots & \ddots & \vdots \\
 & & a_{n1} & & b_{n1} & \cdots & a_{nn}+b_{nn}
\end{array}\right]
\left[\begin{array}{c}
X_{11} \\ \vdots \\ X_{n1} \\ \hline \vdots \\ \hline X_{1n} \\ \vdots \\ X_{nn}
\end{array}\right]
=
\left[\begin{array}{c}
C_{11} \\ \vdots \\ C_{n1} \\ \hline \vdots \\ \hline C_{1n} \\ \vdots \\ C_{nn}
\end{array}\right]
$$

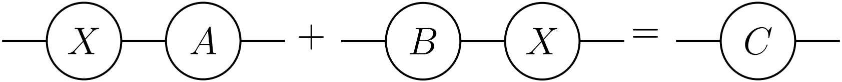
n^2 variables in n^2 equations. Naively, $O(n^6)$ as well

Sylvester Equation (1884)

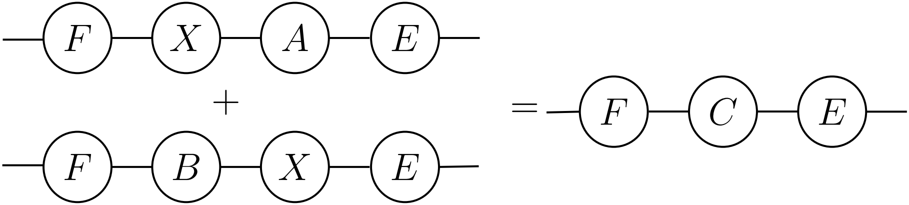
Row operations on this matrix

Act on left and right

Bartels-Stewart (1972)



Bartels-Stewart (1972)



$$E^* A E = \begin{bmatrix} * & & \\ \vdots & \ddots & \\ * & \cdots & * \end{bmatrix}, F^* B F = \begin{bmatrix} * & \cdots & * \\ & \ddots & \vdots \\ & & * \end{bmatrix}$$

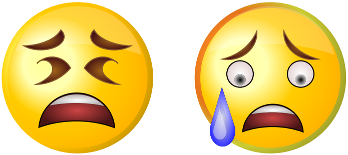
Computing E, F is $O(n^3)$

Simultaneous Sylvester System

$$X \times_s A + B \times_t Y = C$$

$$
\left[\begin{array}{c|c}
\begin{matrix} A^\intercal_{*1*} & & \\ & \ddots & \\ & & A^\intercal_{*1*} \end{matrix} & \begin{matrix} B^\intercal_{1**} & & & \\ \vdots & & & \\ B^\intercal_{a**} & & & \end{matrix} \\
\hline
\begin{matrix} A^\intercal_{*2*} & & \\ & \ddots & \\ & & A^\intercal_{*2*} \end{matrix} & \begin{matrix} & B^\intercal_{1**} & & \\ & \vdots & & \\ & B^\intercal_{a**} & & \end{matrix} \\
\hline
\vdots & \ddots \\
\hline
\begin{matrix} A^\intercal_{*b*} & & \\ & \ddots & \\ & & A^\intercal_{*b*} \end{matrix} & \begin{matrix} & & & B^\intercal_{1**} \\ & & & \vdots \\ & & & B^\intercal_{a**} \end{matrix}
\end{array}\right]
\left[\begin{array}{c}
X_{11} \\ \vdots \\ X_{1s} \\ \hline \vdots \\ \hline X_{a1} \\ \vdots \\ X_{as} \\ \hline Y_{11} \\ \vdots \\ Y_{t1} \\ \hline \vdots \\ \hline Y_{1b} \\ \vdots \\ Y_{tb}
\end{array}\right]
=
\left[\begin{array}{c}
C_{111} \\ \vdots \\ C_{11c} \\ \hline \vdots \\ \hline C_{a11} \\ \vdots \\ C_{a1c} \\ \hline \vdots \\ \hline C_{ab1} \\ \vdots \\ C_{abc}
\end{array}\right].
$$

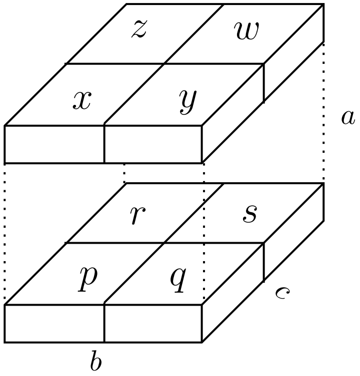
Row operations on this matrix

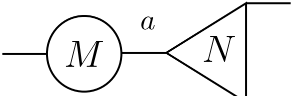


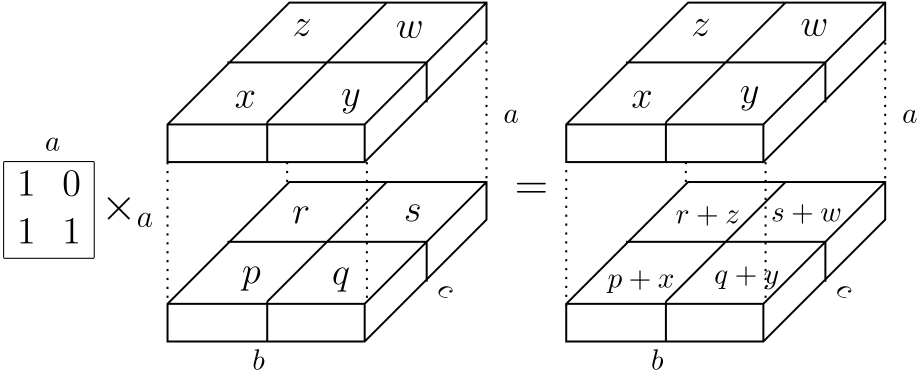
Slice operations on A and B

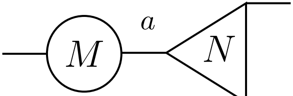
$$M = \begin{array}{|cc|} \hline 1 & 0 \\ 1 & 1 \\ \hline \end{array}$$

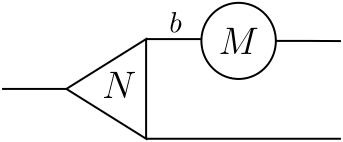
$N =$

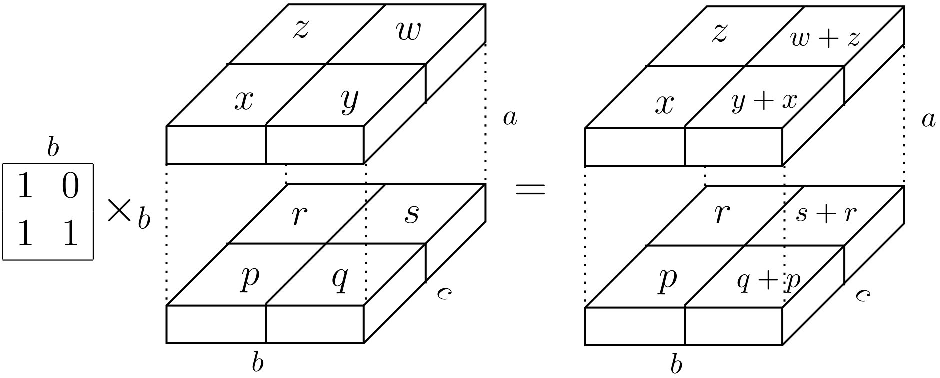


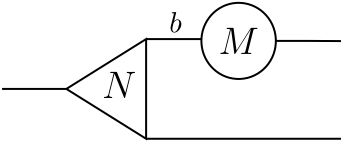

M
a
N

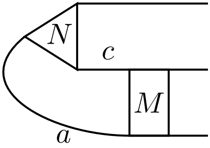

a
1 0
1 1
×a
z
w
x
y
r
s
p
q
a
c
b
=
z
w
x
y
r + z
s + w
p + x
q + y
a
c
b


M
a
N

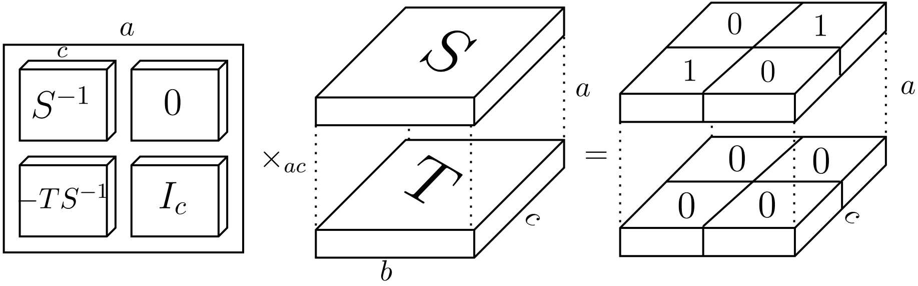

b
M
N


b
1 0
1 1
×_b
z
w
x
y
a
r
s
p
q
c
b
=
z
w + z
x
y + x
a
r
s + r
p
q + p
c
b

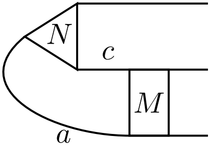

b
M
N



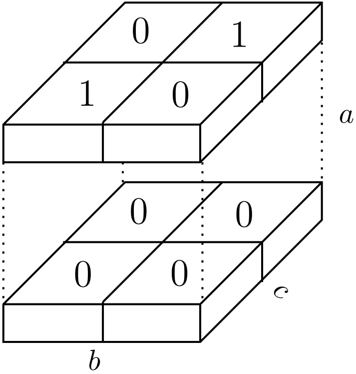
N
c
M
a



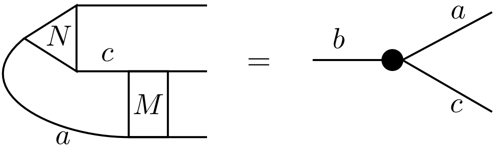
a
c
S^{-1}
0
-TS^{-1}
I_c
×_{ac}
S
T
a
b
c
=
0
1
1
0
a
0
0
0
0
c

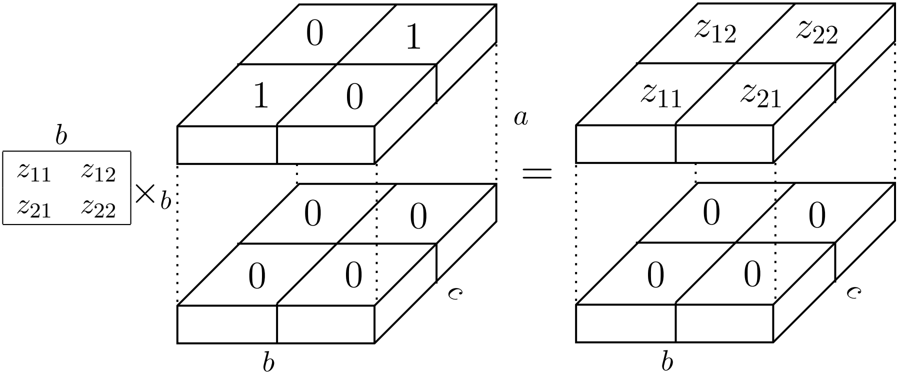


N
c
M
a

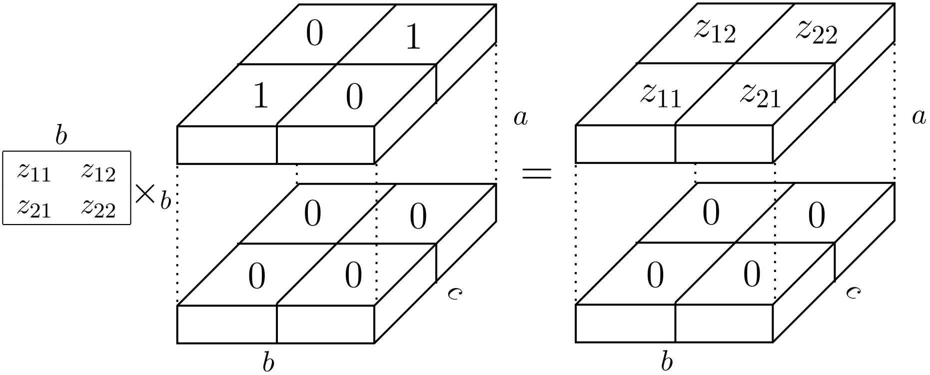


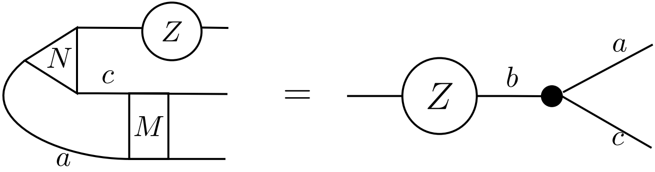
Inert

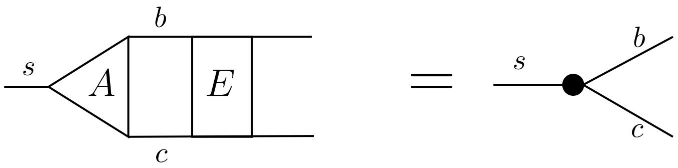




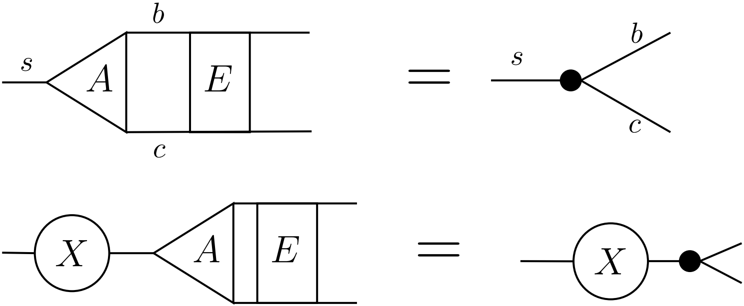
b
z_{11} z_{12}
z_{21} z_{22}
$\times_b$
0 1
1 0
0 0
0 0
a
c
b
=
z_{12} z_{22}
z_{11} z_{21}
0 0
0 0
a
c
b


b
z_{11} z_{12}
z_{21} z_{22}
$\times_b$
0 1
1 0
a
0 0
0 0
c
b
=
z_{12} z_{22}
z_{11} z_{21}
a
0 0
0 0
c
b

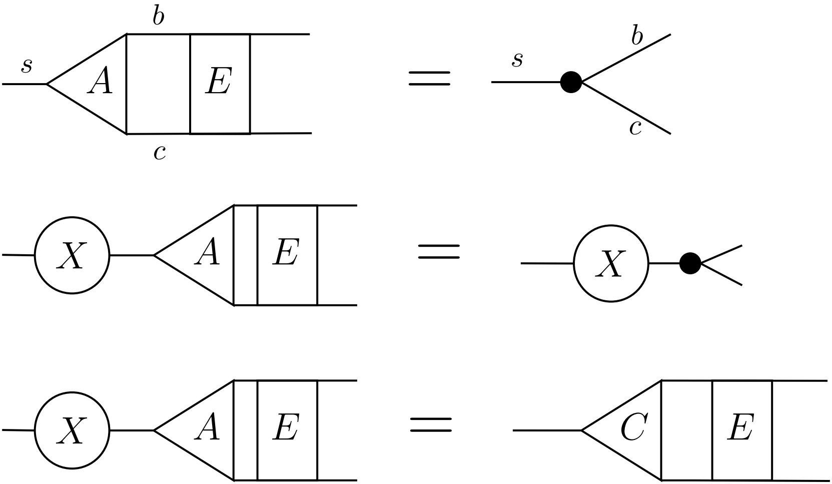

N
Z
c
M
a
=
Z
b
a
c



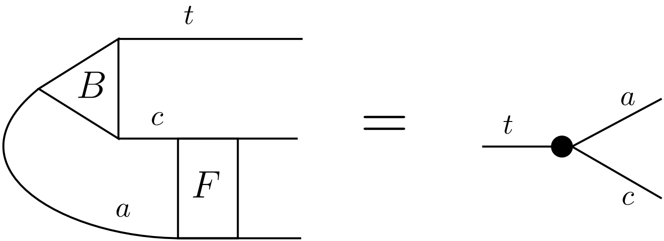
b
s
A
E
c
s
b
c



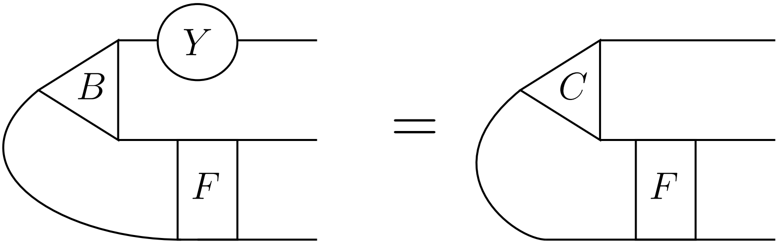
s
b
c
A
E
=
s
b
c
X
A
E
=
X



$B = 0$ case solved

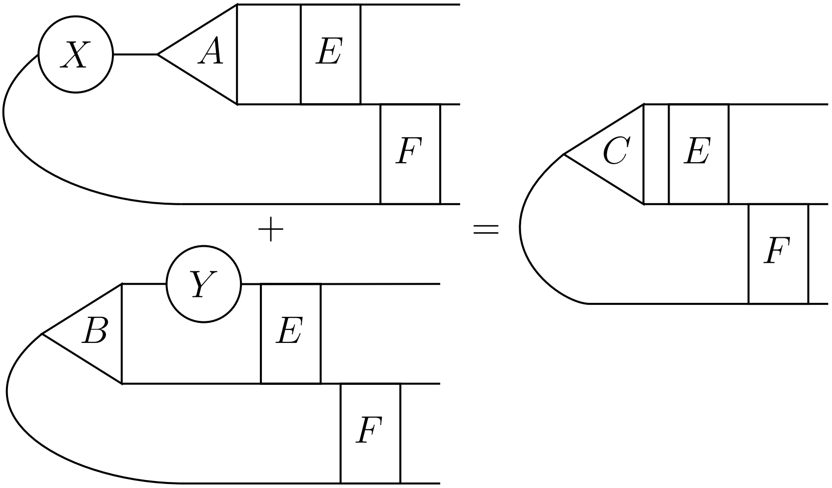


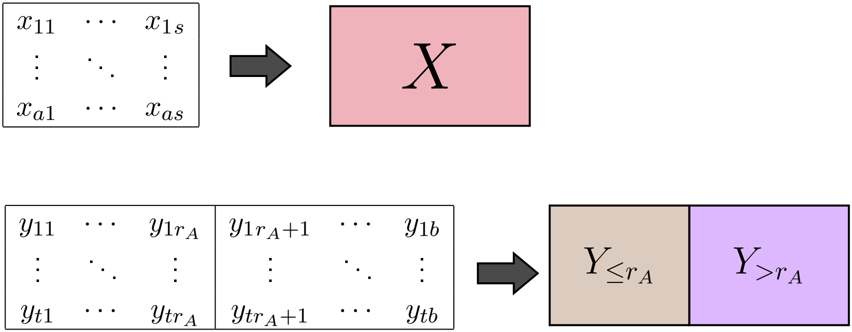
t
B
c
F
a
=
t
a
c

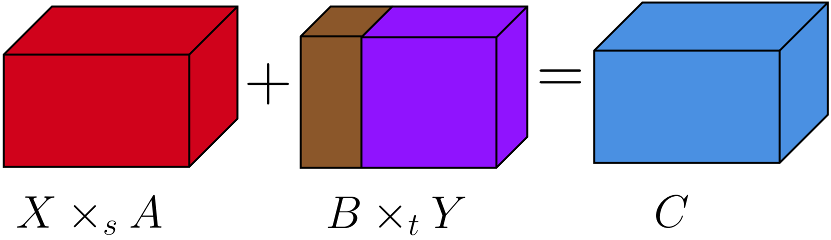


$A = 0$ case solved

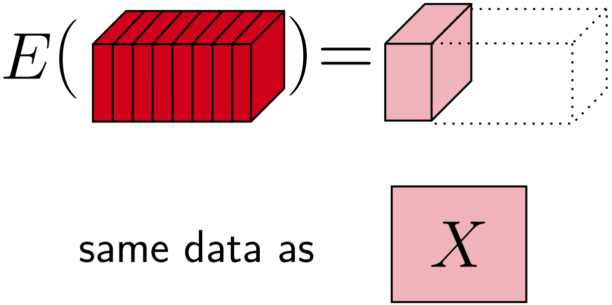
QuickSylver

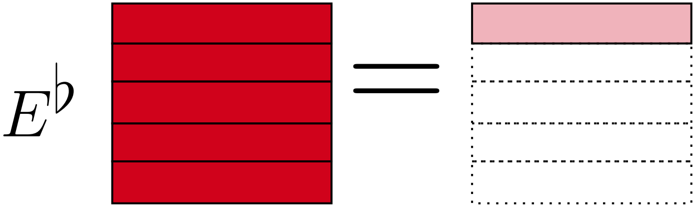



$x_{11} \cdots x_{1s}$
$x_{a1} \cdots x_{as}$
X
$y_{11} \cdots y_{1r_A}$
$y_{1r_A+1} \cdots y_{1b}$
$y_{t1} \cdots y_{tr_A}$
$y_{tr_A+1} \cdots y_{tb}$
$Y_{\leq r_A}$
$Y_{>r_A}$



$X \times_s A$
$+$
$B \times_t Y$
$=$
C

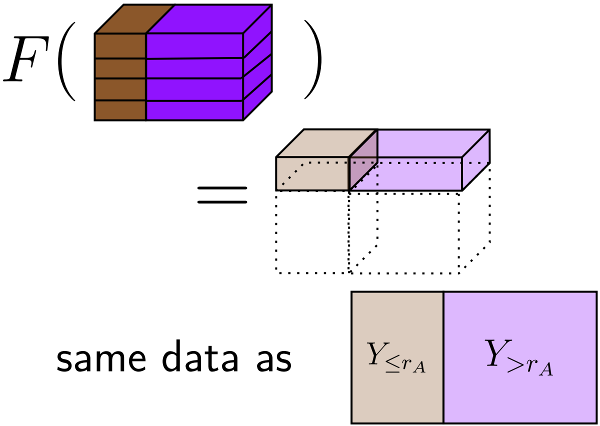

E( ) =
same data as
X



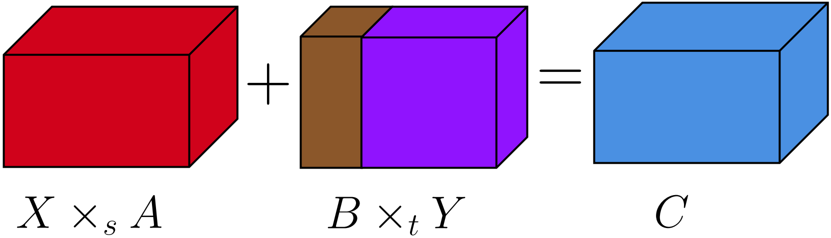
$$\mathcal{A} = [A_{*1*}^{\mathsf{T}} \ \cdots \ A_{*r_A*}^{\mathsf{T}}]^{\mathsf{T}}, \mathcal{A}^{\#}\mathcal{A} = I_s, \mathcal{A}^{\perp}\mathcal{A} = 0$$

$$\left[\begin{array}{c|ccc} \mathcal{A}^{\#} & & & \\ \mathcal{A}^{\perp} & & & \\ \hline -A_{*r_A+1*}^{\mathsf{T}}\mathcal{A}^{\#} & I_c & & \\ \cdots & & \ddots & \\ -A_{*b*}^{\mathsf{T}}\mathcal{A}^{\#} & & & I_c \end{array}\right] \left(\left[\begin{array}{c} A_{*1*}^{\mathsf{T}} \\ \vdots \\ A_{*r_A*}^{\mathsf{T}} \\ \vdots \\ A_{*b*}^{\mathsf{T}} \end{array}\right] X^{\mathsf{T}} \right)$$

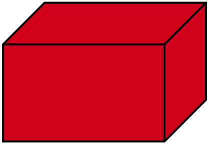
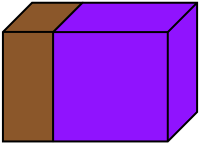
$$= \left[\begin{array}{c} \mathcal{A}^{\#}\mathcal{A} = I_s \\ \mathcal{A}^{\perp}\mathcal{A} = 0 \\ (-A_{*r_A+1*}^{\mathsf{T}}\mathcal{A}^{\#}\mathcal{A} + A_{*r_A+1*}^{\mathsf{T}}) = 0 \\ \vdots \\ (-A_{*b*}^{\mathsf{T}}\mathcal{A}^{\#}\mathcal{A} + A_{*b*}^{\mathsf{T}}) = 0 \end{array}\right] X^{\mathsf{T}} = \left[\begin{array}{c} X^{\mathsf{T}} \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}\right]$$

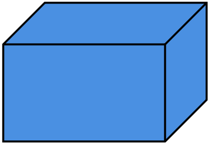

$F($ $)$
$=$
same data as
$Y_{\leq r_A}$
$Y_{>r_A}$

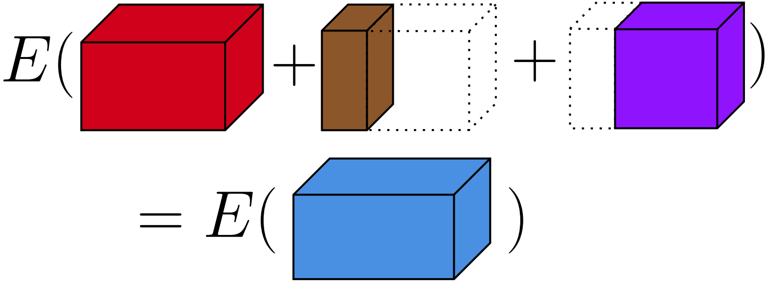
Instead of solving

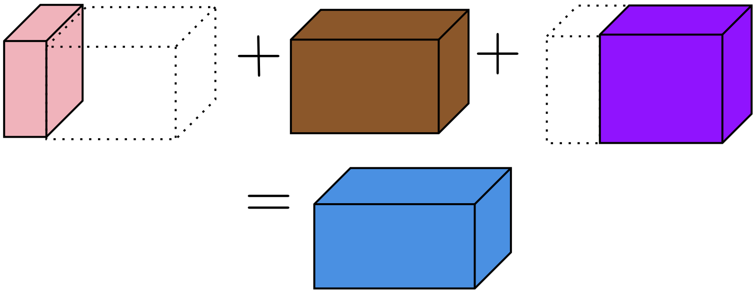


Solve this instead

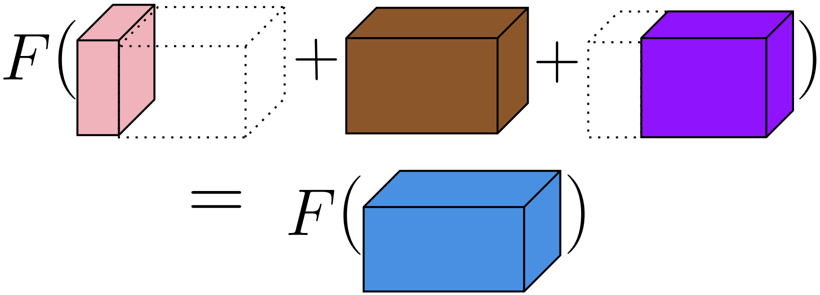
$F(E($  $+$  $))$

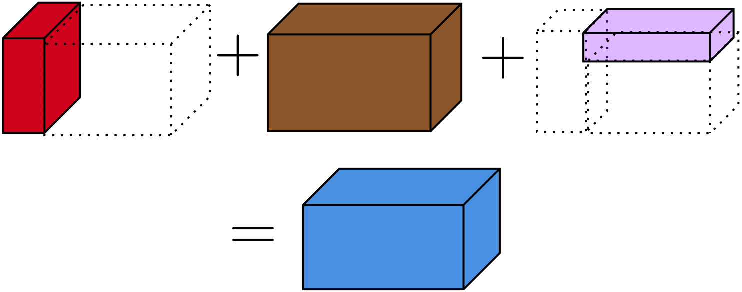
$= F(E($  $))$


E(
+
+
)
= E(
)



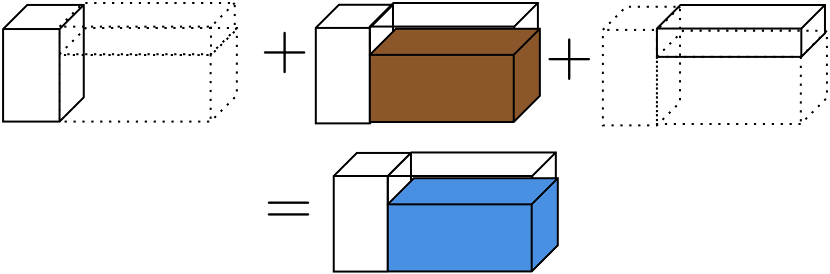
$$E \times_{bc} (X \times_s A + B \times_t Y) = E \times_{bc} C$$


F
F



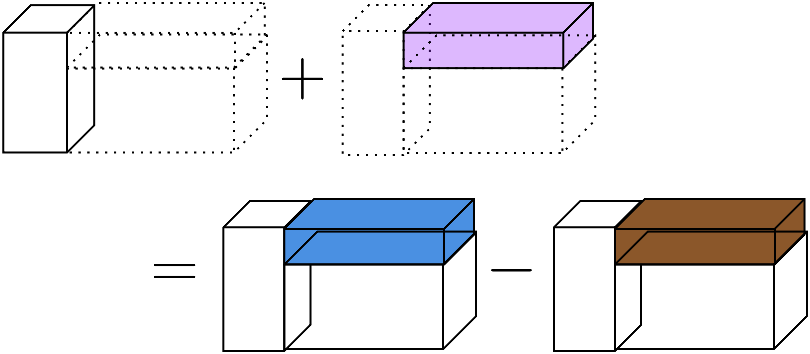
$$F \times_{ac} E \times_{bc} (X \times_s A + B \times_t Y) = F \times_{ac} E \times_{bc} C$$

Solve linear system for $Y_{\leq r_A}$



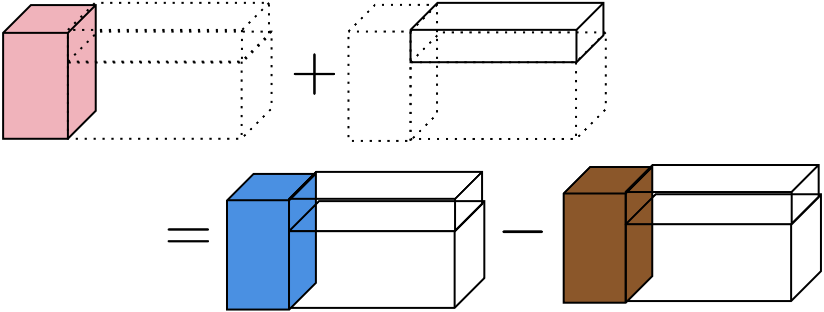
$abc - sa - b(t - r_A)$ equations in tr_A variables. If $r_A \approx s/c$ and s, c are $O(n)$, equation of $O(n)$ variables, solved in $O(n^3)$.

Back-substitute for $Y_{>r_A}$



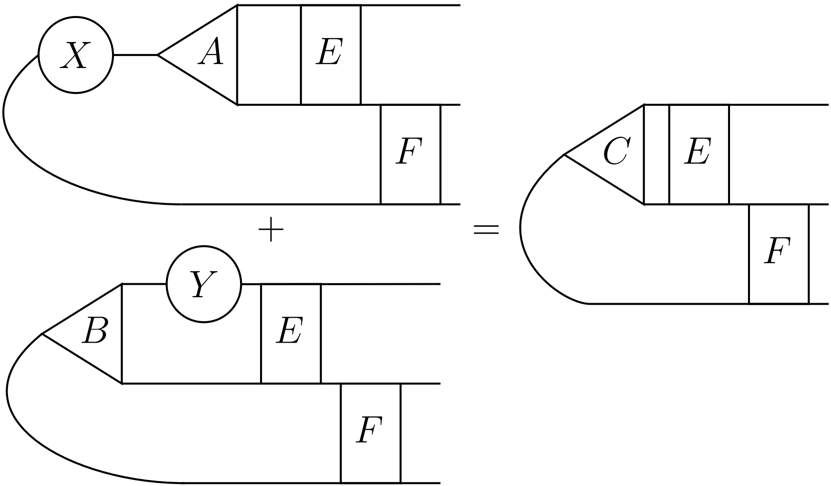
Total of $t(b - r_A)$ coefficients to calculate, each costing $2(r_A + r_B)c$, is $O(n^3)$ under same assumptions.

Back-substitute for X

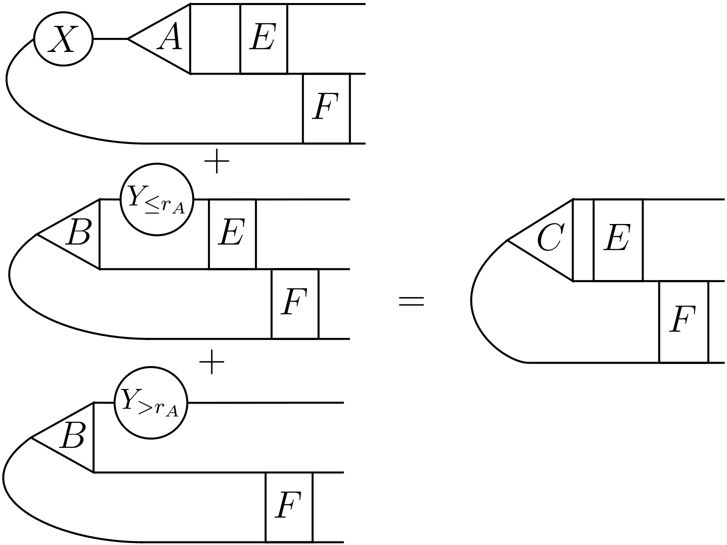


Total of sa coefficients to calculate, each costing $2(r_A + r_B)c$, is $O(n^3)$ under same assumptions.

QuickSylver



QuickSylver - What actually happens



A zoo of tensor software

ID	Package Name	Functionality					Tensor Type	Platform	Language
		DatM	EWOps	SpecCon	Con	Decomp			
0	Acrotensor [28]	–	–	✓	✓	–	D	C, G	C++
1	AdaTM [54]	–	–	✓	–	✓	S	C	C
2	Boost.uBlas.Tensor [6]	✓	✓	✓	✓	–	D	C	C++
3	BTAS [73]	✓	✓	✓	✓	✓	nan	C	C++
4	COGENT [48]	–	–	✓	✓	–	D	G	Python → CUDA
5	COMET [86]	–	–	✓	✓	–	S	C	C++ → C++
6	CoTenGra [34]	–	–	✓	✓	–	D	C, D, G	Python
7	CP-CALS [75]	–	–	✓	–	✓	D	C, G	C++, Mat[i]
8	CSTF [9]	–	–	–	–	✓	S	D	Scala
9	CuTensor [64]	✓	✓	✓	✓	–	D	G	C, CUDA
10	cuTT [41]	✓	–	–	–	–	D	G	C++, CUDA
11	Cyclops [81]	✓	✓	✓	✓	–	S, D	C, D, G	C++
12	D-Tucker [43]	–	–	–	–	✓	D	C	Matlab
13	DFacTo [15]	–	–	–	–	✓	S	C, D	C++
14	Eigen Tensor [16]	✓	✓	✓	✓	–	D	C, G	C++
15	ExaTN [60]	✓	✓	✓	✓	✓	D	C, D, G	C++, Py[i]
16	Fastor [74]	✓	✓	✓	✓	–	D	C	C++
17	FTensor [52]	✓	✓	✓	✓	–	D	C	C++
18	Genten [72]	–	–	–	–	✓	D, S	C, G	C++
19	GigaTensor [45]	–	–	–	–	✓	S	C, D	Unknown
20	HPTT [85]	✓	–	–	–	–	D	C	C++, Python[i], C[i]
21	ITensor [29]	–	✓	✓	✓	✓	D, BS	C, G[x]	C++, Julia
22	libtensor [42]	–	–	✓	✓	–	D, BS	C	C++
23	Ltensor [2]	–	–	✓	✓	–	D	C	C++
24	MATLAB [58]	✓	✓	✓	✓	–	D	C	Matlab
25	MultiArray [30]	✓	–	–	–	–	D	C	C++
26	multiway [38]	–	–	–	–	✓	D	C	R
27	N-way toolbox [4]	–	–	–	–	✓	D	C	Matlab
28	NCON [69]	–	–	✓	✓	–	D	C	Matlab
29	netcon [70]	–	–	✓	✓	–	D	C	Matlab

With ITensor

```
FEB = contract(F, E, B)
FEA = contract(F, E, A)
FEC = contract(F, E, C)

Y_dense = solve(dense_part(FEB), dense_part(FEC))
Y_backsub = rest_of_Y_part(FEC) -
    contract(rest_of_Y_part(FEB), Y_dense)
X_backsub = contract(E, C) - contract(E, B, Y)
```

Results

- $A, B, C \in K^{500\times 500\times 500}$ solved on my laptop in about 20 seconds
- Thinner tensor requires solving bigger dense system. (i.e $A, B, C \in K^{400\times 400\times 20}$ solved on my laptop in about 80 seconds)

Results

- $A, B, C \in K^{500 \times 500 \times 500}$ solved on my laptop in about 20 seconds
- Thinner tensor requires solving bigger dense system. (i.e $A, B, C \in K^{400 \times 400 \times 20}$ solved on my laptop in about 80 seconds)
- Features for free: contraction sequence optimization, concurrency, GPUs
- Caveat: ITensor do not support finite fields, so below are done with Float64.

Next steps

- Implement bits of the tensor network abstractions in Magma
- QuickSylver in Magma for module endomorphisms

Next steps

- Implement bits of the tensor network abstractions in Magma
- QuickSylver in Magma for module endomorphisms
- Generalize to higher valence and derivations
- Investigate into tensor computation software (survey paper lists 79 packages!!)

Tensor problems deserve tensor solutions.