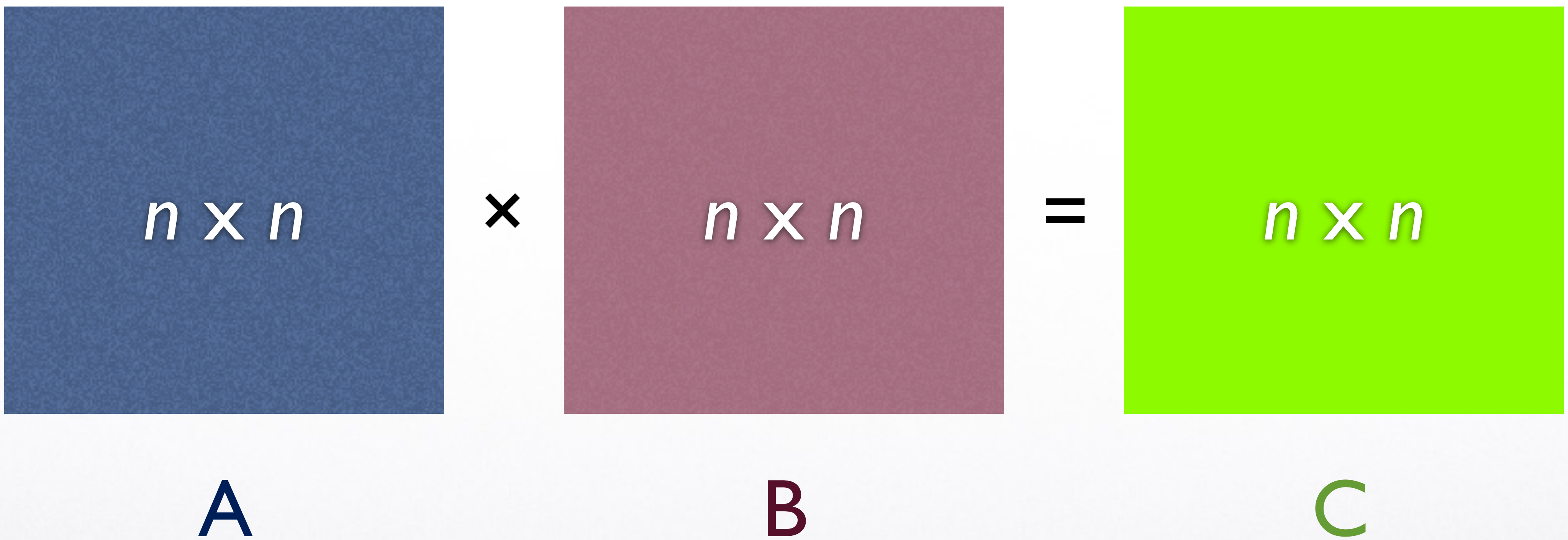
Fast matrix multiplication: Limitations of C.–W. method

Yuval Filmus, Institute for Advanced Study (Princeton)
Joint work with Andris Ambainis (U. of Latvia) and François Le Gall (U. of Tokyo)

Matrix multiplication



How fast can we do it?

Why do we care?

Why do we care?

Equivalent to:

- Matrix inverse
- Solving linear equations
- Determinant
- Diagonalization

Why do we care?

Equivalent to:

- Matrix inverse
- Solving linear equations
- Determinant
- Diagonalization

Used to solve:

- Testing graphs for triangles
- All pairs shortest paths
- Parsing of context-free languages

Why do we care?

Equivalent to:

- Matrix inverse
- Solving linear equations
- Determinant
- Diagonalization

Used to solve:

- Testing graphs for triangles
- All pairs shortest paths
- Parsing of context-free languages

Forms inner loop in:

- Linear programming

High-school algorithm

```
for (i=0; i<n; i++)
  for (j=0; j<n; j++)
    for (k=0; k<n; k++)
      C[i][k] += A[i][j] * B[j][k];
```

High-school algorithm

```
for (i=0; i<n; i++)
  for (j=0; j<n; j++)
    for (k=0; k<n; k++)
      C[i][k] += A[i][j] * B[j][k];
```

Complexity: $O(n^3)$

Can we do better?

Can we do better?

- Strassen (1969): $O(n^{2.81})$

Can we do better?

- Strassen (1969): $O(n^{2.81})$
- Schönhage (1981): $O(n^{2.55})$
- Strassen (1986): $O(n^{2.48})$
- Coppersmith & Winograd (1987): $O(n^{2.376})$

Can we do better?

- Strassen (1969): $O(n^{2.81})$
- Schönhage (1981): $O(n^{2.55})$
- Strassen (1986): $O(n^{2.48})$
- Coppersmith & Winograd (1987): $O(n^{2.376})$
- Stothers (2010), Williams (2012), Le Gall (2014): $O(n^{2.373})$

Exponent of matrix multiplication

Exponent of matrix multiplication

- Definition: ω is best exponent
  $O(n^{\rho})$ algorithm $\Rightarrow \omega \leq \rho$

Exponent of matrix multiplication

- Definition: ω is best exponent
  $O(n^\rho)$ algorithm $\Rightarrow \omega \leq \rho$
- Le Gall (2014): $\omega < 2.3728639$

Exponent of matrix multiplication

- Definition: ω is best exponent
  $O(n^\rho)$ algorithm $\Rightarrow \omega \leq \rho$

- Le Gall (2014): $\omega < 2.3728639$
- Conjecture: $\omega = 2$

Exponent of matrix multiplication

- Definition: ω is best exponent
  $O(n^\rho)$ algorithm $\Rightarrow \omega \leq \rho$
- Le Gall (2014): $\omega < 2.3728639$
- Conjecture: $\omega = 2$
- Best lower bound: $\Omega(n^2 \log n)$ [Raz]

What we show

What we show

- All algorithms since 1987 use a single method: analyze powers of the CW identity (higher powers lead to better bounds)

What we show

- All algorithms since 1987 use a single method: analyze powers of the CW identity (higher powers lead to better bounds)
- We show: this method cannot prove $\omega < 2.3725$ (cf. best known bound $\omega < 2.3728$)

What we show

- All algorithms since 1987 use a single method: analyze powers of the CW identity (higher powers lead to better bounds)
- We show: this method cannot prove $\omega < 2.3725$ (cf. best known bound $\omega < 2.3728$)
- We suggest generalized method which could break this limit (but cannot prove $\omega < 2.3078$)

Proof outline

Proof outline

- Formally express what it means to analyze *n*th power of CW identity

Proof outline

- Formally express what it means to analyze *n*th power of CW identity
- Develop a framework encompassing analysis of all powers at once

Proof outline

- Formally express what it means to analyze *n*th power of CW identity
- Develop a framework encompassing analysis of all powers at once
- Prove a limitation on new framework

What is computation?

What is computation?

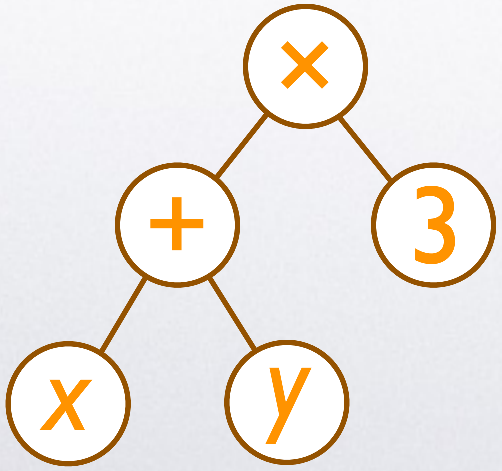
- Different computation models have different power

What is computation?

- Different computation models have different power
- Natural model for us – algebraic:

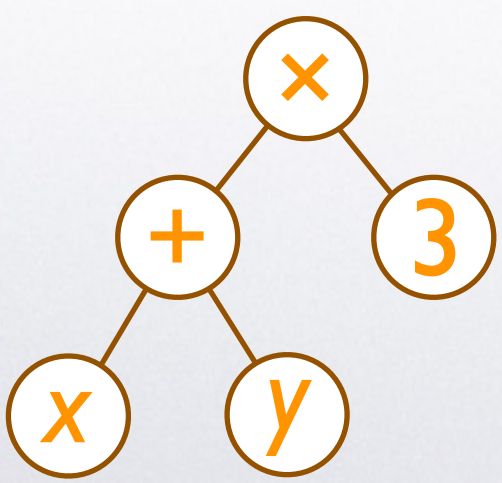
What is computation?

- Different computation models have different power
- Natural model for us – algebraic:
  - Algebraic circuits over +,−,×,÷ with constants



What is computation?

- Different computation models have different power
- Natural model for us – algebraic:
  - Algebraic circuits over +,−,×,÷ with constants
  - Straight-line programs



$t = x + y$

$out = t \times 3$

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

$$\ell_1 = a + b$$
$$\ell_2 = c + d$$

$$m_1 = a \times c$$
$$m_2 = b \times d$$
$$m_3 = \ell_1 \times \ell_2$$

$$x = m_1 - m_2$$
$$t = m_1 + m_2$$
$$y = m_3 - t$$

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

$$\left.\begin{aligned} \ell_1 &= a + b \\ \ell_2 &= c + d \end{aligned}\right\}$$ Linear combinations of input

$$m_1 = a \times c$$
$$m_2 = b \times d$$
$$m_3 = \ell_1 \times \ell_2$$

$$x = m_1 - m_2$$
$$t = m_1 + m_2$$
$$y = m_3 - t$$

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

$$\left.\begin{aligned} \ell_1 &= a + b \\ \ell_2 &= c + d \end{aligned}\right\}$$ Linear combinations of input

$$\left.\begin{aligned} m_1 &= a \times c \\ m_2 &= b \times d \\ m_3 &= \ell_1 \times \ell_2 \end{aligned}\right\}$$ Products of ℓ_js and inputs

$$\begin{aligned} x &= m_1 - m_2 \\ t &= m_1 + m_2 \\ y &= m_3 - t \end{aligned}$$

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

$$\ell_1 = a + b$$
$$\ell_2 = c + d$$
} Linear combinations of input

$$m_1 = a \times c$$
$$m_2 = b \times d$$
$$m_3 = \ell_1 \times \ell_2$$
} Products of ℓ_js and inputs

$$x = m_1 - m_2$$
$$t = m_1 + m_2$$
$$y = m_3 - t$$
} Linear combinations of m_js

Bilinear algorithms

Example: $(a + b\,i) \times (c + d\,i) = x + y\,i$

$$\left.\begin{aligned} \ell_1 &= a + b \\ \ell_2 &= c + d \end{aligned}\right\}$$ Linear combinations of input

$$\left.\begin{aligned} m_1 &= a \times c \\ m_2 &= b \times d \\ m_3 &= \ell_1 \times \ell_2 \end{aligned}\right\}$$ Products of ℓ_js and inputs

$$\left.\begin{aligned} x &= m_1 - m_2 \\ t &= m_1 + m_2 \\ y &= m_3 - t \end{aligned}\right\}$$ Linear combinations of m_js

Strassen: normal form for bilinear functions

Strassen's algorithm

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

Strassen's algorithm

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$m_1 = (a_{11}+a_{22})(b_{11}+b_{22})$

$m_2 = (a_{21}+a_{22})(b_{11})$

$m_3 = (a_{11})(b_{12}-b_{22})$

$m_4 = (a_{22})(b_{21}-b_{11})$

$m_5 = (a_{11}+a_{12})(b_{22})$

$m_6 = (a_{21}-a_{11})(b_{11}+b_{12})$

$m_7 = (a_{12}-a_{22})(b_{21}+b_{22})$

Strassen's algorithm

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$m_1 = (a_{11}+a_{22})(b_{11}+b_{22})$
$m_2 = (a_{21}+a_{22})(b_{11})$
$m_3 = (a_{11})(b_{12}-b_{22})$
$m_4 = (a_{22})(b_{21}-b_{11})$
$m_5 = (a_{11}+a_{12})(b_{22})$
$m_6 = (a_{21}-a_{11})(b_{11}+b_{12})$
$m_7 = (a_{12}-a_{22})(b_{21}+b_{22})$

$c_{11} = m_1+m_4-m_5+m_7$
$c_{12} = m_3+m_5$
$c_{21} = m_2+m_4$
$c_{22} = m_1-m_2+m_3+m_6$

Strassen's algorithm

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix}$$

Strassen's algorithm

Strassen's algorithm

$M_1 = (A_{11}+A_{22})(B_{11}+B_{22})$

•

•

•

2x2 algorithm

$M_7 = (A_{12}-A_{22})(B_{21}+B_{22})$

Strassen's algorithm

$M_1 = (A_{11}+A_{22})(B_{11}+B_{22})$

•

•

•

$M_7 = (A_{12}-A_{22})(B_{21}+B_{22})$

2x2 algorithm

$C_{11} = M_1+M_4-M_5+M_7$

$C_{12} = M_3+M_5$

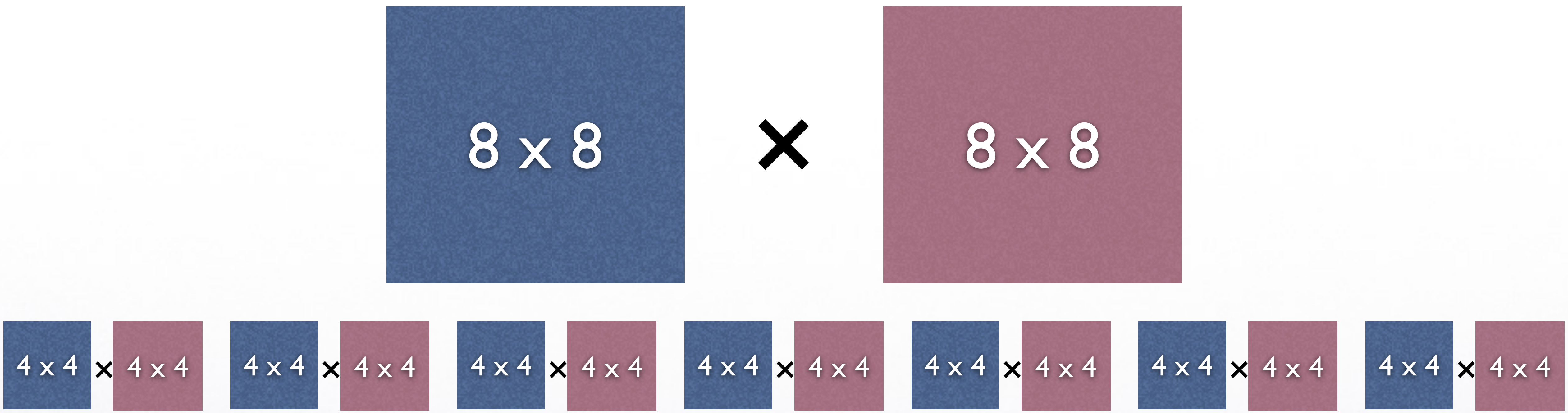
$C_{21} = M_2+M_4$

$C_{22} = M_1-M_2+M_3+M_6$

Strassen's algorithm

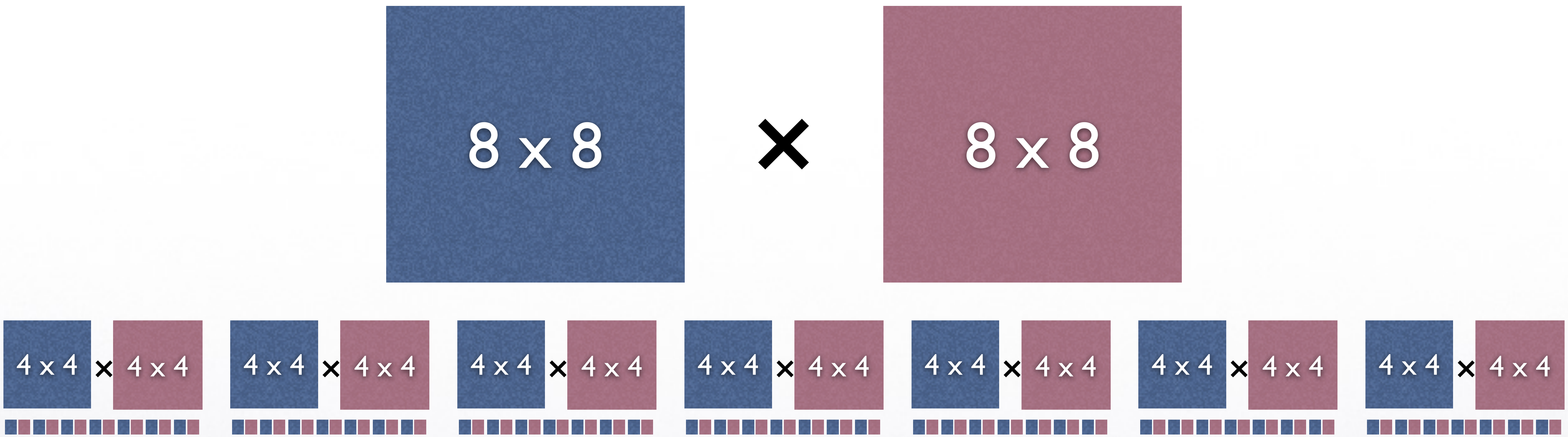
8 x 8 × 8 x 8

Strassen's algorithm



7 multiplications 4x4

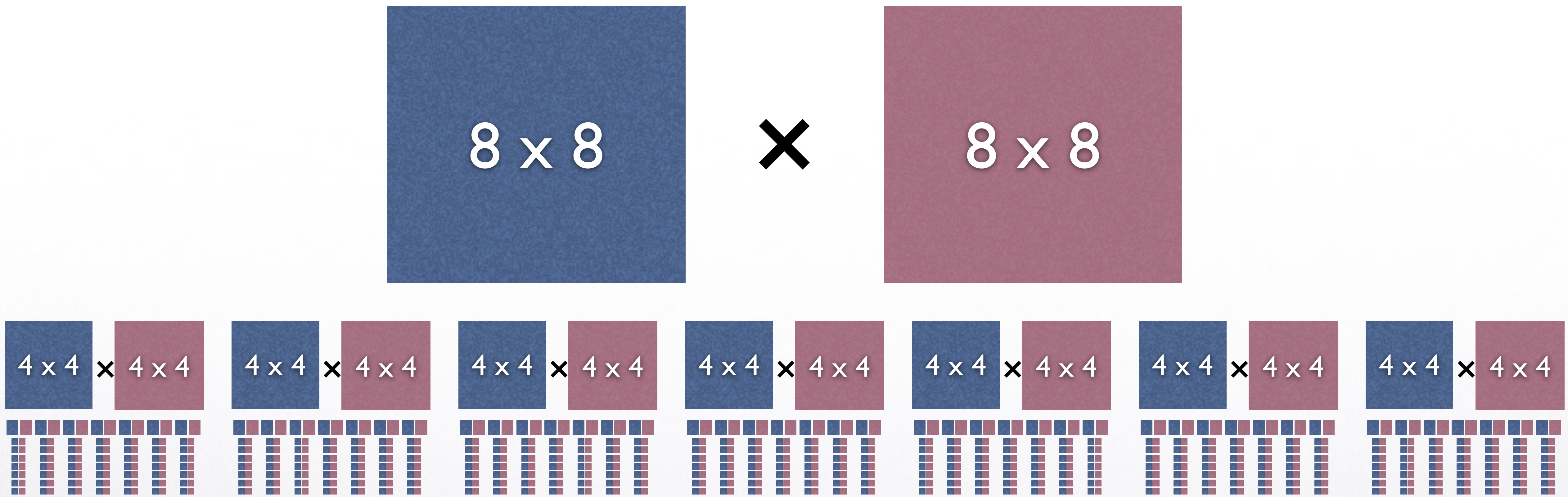
Strassen's algorithm



7 multiplications 4x4

7x7 multiplications 2x2

Strassen's algorithm



7 multiplications 4x4

7x7 multiplications 2x2

7x7x7 multiplications 1x1

Strassen’s algorithm

Strassen's algorithm

- Multiplying 2x2 matrices: 7 multiplications

Strassen's algorithm

- Multiplying 2x2 matrices: 7 multiplications
- Multiplying 4x4 matrices: 49 multiplications

Strassen's algorithm

- Multiplying 2x2 matrices: 7 multiplications
- Multiplying 4x4 matrices: 49 multiplications
- Multiplying 2^dx2^d matrices:
  $7^d=(2^d)^{\log_2 7}\approx(2^d)^{2.81}$ multiplications

Strassen's algorithm

- Multiplying 2x2 matrices: 7 multiplications
- Multiplying 4x4 matrices: 49 multiplications
- Multiplying 2^dx2^d matrices:
  $7^d=(2^d)^{\log_2 7}\approx(2^d)^{2.81}$ multiplications
- Multiplying $n\times n$ matrices: $O(n^{2.81})$

Tensor notation

Strassen's 2x2 algorithm

$$m_1 = (a_{11}+a_{22})(b_{11}+b_{22})$$
$$m_2 = (a_{21}+a_{22})(b_{11})$$
$$m_3 = (a_{11})(b_{12}-b_{22})$$
$$m_4 = (a_{22})(b_{21}-b_{11})$$
$$m_5 = (a_{11}+a_{12})(b_{22})$$
$$m_6 = (a_{21}-a_{11})(b_{11}+b_{12})$$
$$m_7 = (a_{12}-a_{22})(b_{21}+b_{22})$$
$$c_{11} = m_1+m_4-m_5+m_7$$
$$c_{12} = m_3+m_5$$
$$c_{21} = m_2+m_4$$
$$c_{22} = m_1-m_2+m_3+m_6$$

Tensor notation

Strassen's 2x2 algorithm

$m_1 = (a_{11}+a_{22})(b_{11}+b_{22})$
$m_2 = (a_{21}+a_{22})(b_{11})$
$m_3 = (a_{11})(b_{12}-b_{22})$
$m_4 = (a_{22})(b_{21}-b_{11})$
$m_5 = (a_{11}+a_{12})(b_{22})$
$m_6 = (a_{21}-a_{11})(b_{11}+b_{12})$
$m_7 = (a_{12}-a_{22})(b_{21}+b_{22})$
$c_{11} = m_1+m_4-m_5+m_7$
$c_{12} = m_3+m_5$
$c_{21} = m_2+m_4$
$c_{22} = m_1-m_2+m_3+m_6$

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$$
$$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}$$
$$=$$
$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Strassen's identity

Tensor notation

Strassen's 2x2 algorithm

$m_1 = (a_{11}+a_{22})(b_{11}+b_{22})$
$m_2 = (a_{21}+a_{22})(b_{11})$
$m_3 = (a_{11})(b_{12}-b_{22})$
$m_4 = (a_{22})(b_{21}-b_{11})$
$m_5 = (a_{11}+a_{12})(b_{22})$
$m_6 = (a_{21}-a_{11})(b_{11}+b_{12})$
$m_7 = (a_{12}-a_{22})(b_{21}+b_{22})$
$c_{11} = m_1+m_4-m_5+m_7$
$c_{12} = m_3+m_5$
$c_{21} = m_2+m_4$
$c_{22} = m_1-m_2+m_3+m_6$

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+ a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}$$

$$=$$

$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+ (a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+ (a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+ (a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+ (a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+ (a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+ (a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Strassen's identity

$R(\langle 2,2,2\rangle) \leq 7$

Bilinear view of matrices

Bilinear view of matrices

Matrices are bilinear operators:

$$M_{n \times m} = \sum_{i=1}^{n} \sum_{j=1}^{m} M_{ij}\, a_i\, b_j$$

Bilinear view of matrices

Matrices are bilinear operators:

$$M_{n \times m} = \sum_{i=1}^{n} \sum_{j=1}^{m} M_{ij}\, a_i\, b_j$$

Rank one matrices:

$$M_{n \times m} = \sum_{i=1}^{n} \sum_{j=1}^{m} A_i\, B_j\, a_i\, b_j$$

Bilinear view of matrices

Matrices are bilinear operators:

$$M_{n\times m} = \sum_{i=1}^{n} \sum_{j=1}^{m} M_{ij}\, a_i\, b_j$$

Rank one matrices:

$$M_{n\times m} = \sum_{i=1}^{n} \sum_{j=1}^{m} A_i\, B_j\, a_i\, b_j$$

Rank = min no. rank one matrices summing to M

Bilinear view of matrices

Matrices are bilinear operators:

$$M_{n\times m} = \sum_{i=1}^{n}\sum_{j=1}^{m} M_{ij}\, a_i\, b_j$$

Rank one matrices:

$$M_{n\times m} = \sum_{i=1}^{n}\sum_{j=1}^{m} A_i\, B_j\, a_i\, b_j$$

Rank = min no. rank one matrices summing to M

Row rank = Rank = Column rank

More on tensors

More on tensors

Tensors: three-dimensional matrices

More on tensors

Tensors: three-dimensional matrices

$$\langle n,m,p\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p} a_{ij}b_{jk}c_{ik}$$

$nm \times mp \times np$ tensor

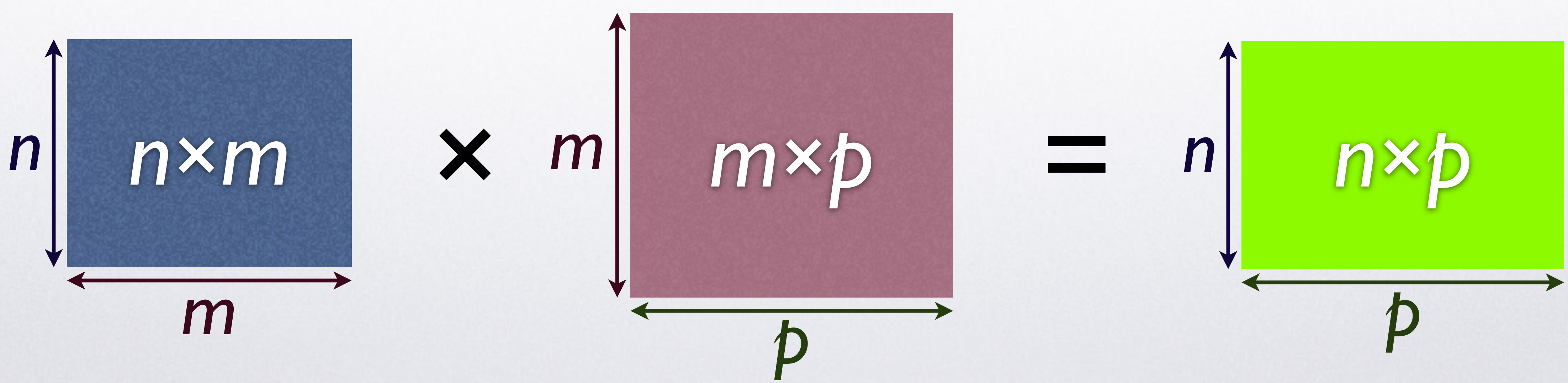
More on tensors

Tensors: three-dimensional matrices

$$\langle n,m,p\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p} a_{ij}b_{jk}c_{ik}$$

$nm \times mp \times np$ tensor

Matrix multiplication tensor



More on tensors

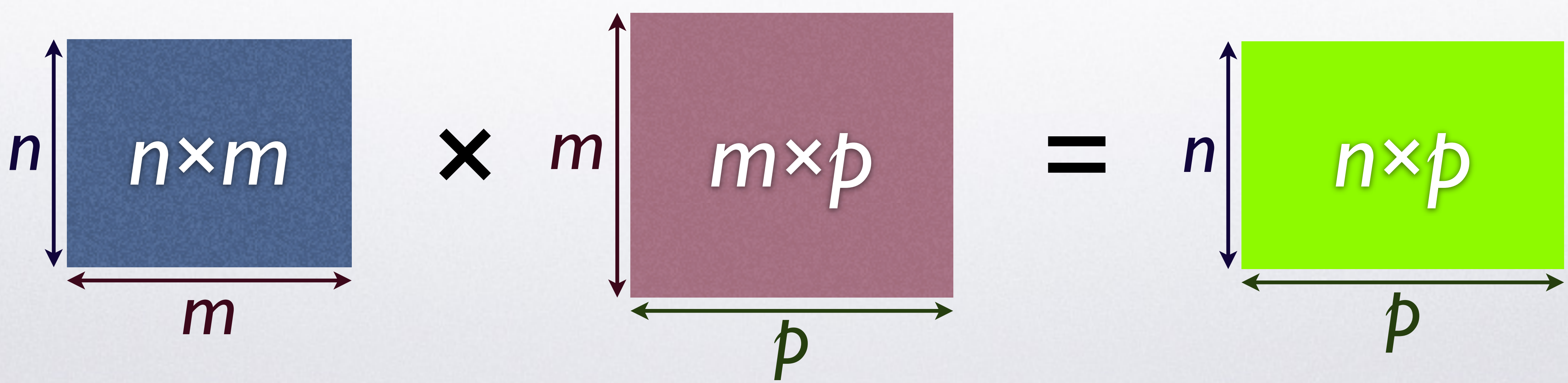
Tensors: three-dimensional matrices

$$\langle n,m,p\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p} a_{ij}b_{jk}c_{ik}$$

$nm \times mp \times np$ tensor

1 at row (i,j)
column (j,k)
depth (i,k)

Matrix multiplication tensor



Tensor rank

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$$
$$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}=$$
$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Strassen's identity

Tensor rank

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$$
$$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}=$$

$\Big\}\ \langle 2,2,2\rangle$

$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Strassen's identity

Tensor rank

$$\left.\begin{array}{l} a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+ \\ a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}= \end{array}\right\} \langle 2,2,2\rangle$$

$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Rank one tensors

Strassen's identity

Tensor rank

$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$
$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}=$ $\Big\}\ \langle 2,2,2\rangle$

$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$
$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$
$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$
$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$
$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$
$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$
$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$

Strassen's identity

$R(T)$ = tensor rank of T: min no. rank one tensors summing to T

Tensor rank

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$$
$$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}=$$

$\Big\}$ $\langle 2,2,2\rangle$

$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

Strassen's identity

R(T) = tensor rank of T:
min no. rank one tensors summing to T

NP-hard to compute! (Håstad)

Tensor rank

$$a_{11}b_{11}c_{11}+a_{12}b_{21}c_{11}+a_{11}b_{12}c_{12}+a_{12}b_{22}c_{12}+$$
$$a_{21}b_{11}c_{21}+a_{22}b_{21}c_{21}+a_{21}b_{12}c_{22}+a_{22}b_{22}c_{22}=$$

$\Big\}$ $\langle 2,2,2\rangle$

$$(a_{11}+a_{22})(b_{11}+b_{22})(c_{11}+c_{22})+$$
$$(a_{21}+a_{22})(b_{11})(c_{21}-c_{22})+$$
$$(a_{11})(b_{12}-b_{22})(c_{12}+c_{22})+$$
$$(a_{22})(b_{21}-b_{11})(c_{11}+c_{21})+$$
$$(a_{11}+a_{12})(b_{22})(c_{12}-c_{11})+$$
$$(a_{21}-a_{11})(b_{11}+b_{12})(c_{22})+$$
$$(a_{12}-a_{22})(b_{21}+b_{22})(c_{11})$$

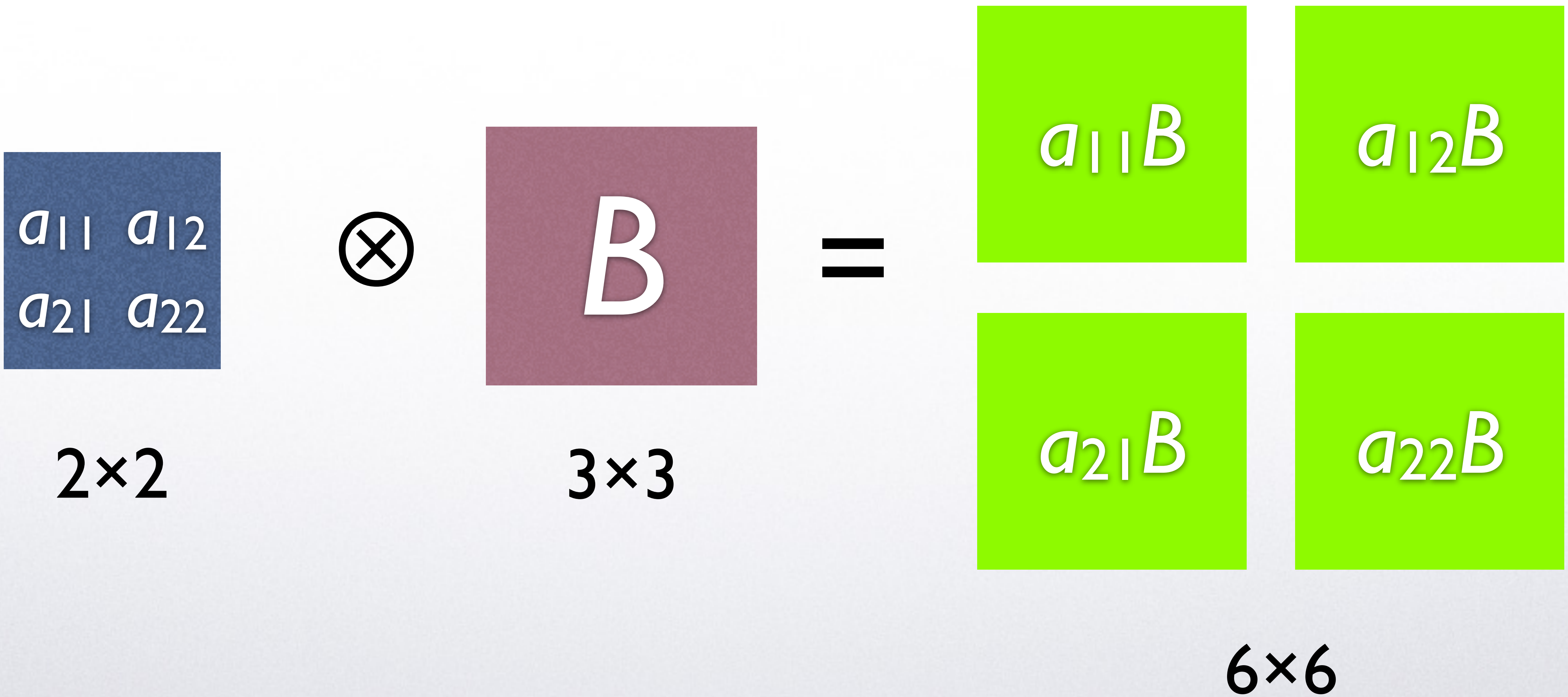
R(T) = tensor rank of T: min no. rank one tensors summing to T

NP-hard to compute! (Håstad)

Strassen's identity ⟶ $R(\langle 2,2,2\rangle)\leq 7$

Tensor product

Kronecker product of matrices:



Tensor product

Kronecker product of matrices:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \otimes B = \begin{bmatrix} a_{11}B & a_{12}B \\ a_{21}B & a_{22}B \end{bmatrix}$$

2×2 $\quad$ 3×3 $\quad$ 6×6

Tensor product of tensors – 3D analog

Tensor product

Tensor product

- Example: $\langle n_1,m_1,p_1\rangle\otimes\langle n_2,m_2,p_2\rangle=\langle n_1n_2,m_1m_2,p_1p_2\rangle$

Tensor product

- Example: $\langle n_1,m_1,p_1\rangle\otimes\langle n_2,m_2,p_2\rangle=\langle n_1n_2,m_1m_2,p_1p_2\rangle$
- Corresponds to recursion: Strassen's identity $\Rightarrow$

$$R(\langle 2^d,2^d,2^d\rangle)=R(\langle 2,2,2\rangle^{\otimes d})\leq R(\langle 2,2,2\rangle)^d=7^d$$

Tensor product

- Example: $\langle n_1,m_1,p_1\rangle\otimes\langle n_2,m_2,p_2\rangle=\langle n_1n_2,m_1m_2,p_1p_2\rangle$
- Corresponds to recursion: Strassen's identity $\Rightarrow$

$$R(\langle 2^d,2^d,2^d\rangle)=R(\langle 2,2,2\rangle^{\otimes d})\leq R(\langle 2,2,2\rangle)^d=7^d$$

- Strassen showed $R(\langle n,n,n\rangle)=O(n^{\omega+\varepsilon})$ and v.v.

Tensor product

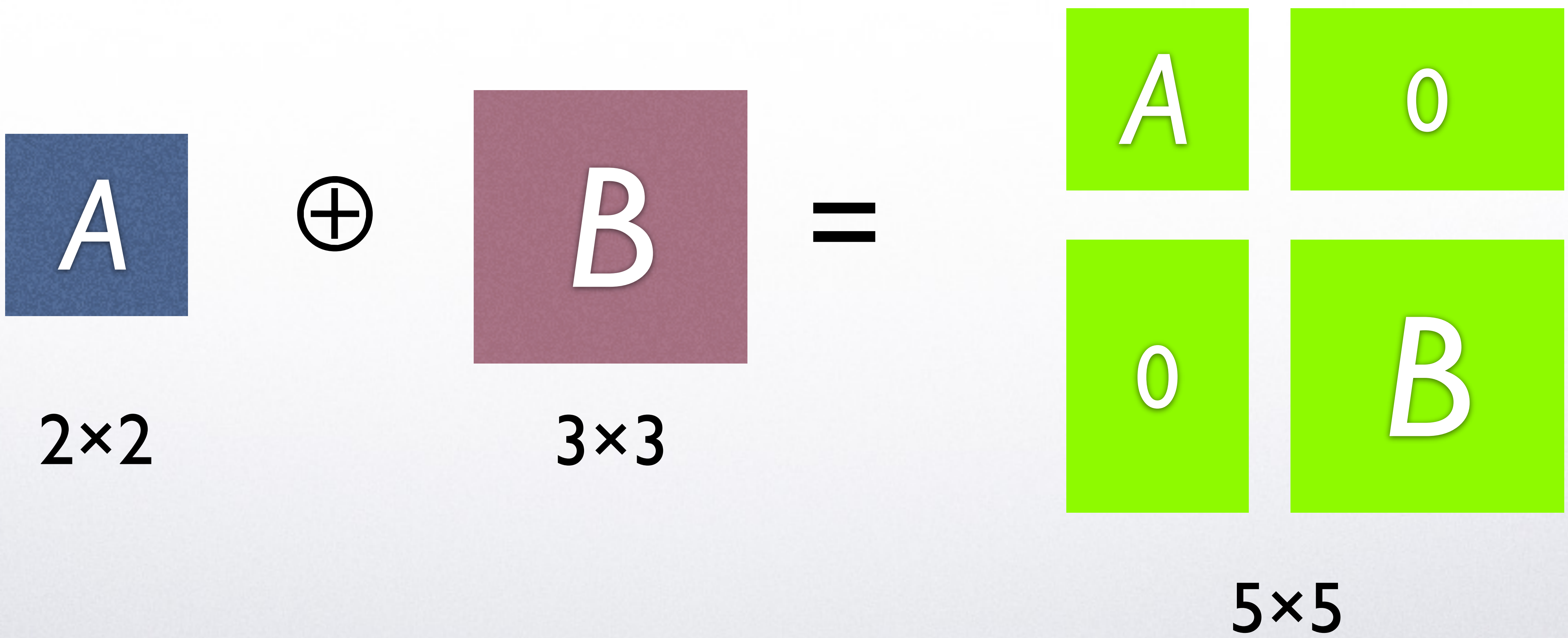
- Example: $\langle n_1,m_1,p_1\rangle\otimes\langle n_2,m_2,p_2\rangle=\langle n_1n_2,m_1m_2,p_1p_2\rangle$
- Corresponds to recursion: Strassen's identity $\Rightarrow$

$$R(\langle 2^d,2^d,2^d\rangle)=R(\langle 2,2,2\rangle^{\otimes d})\leq R(\langle 2,2,2\rangle)^d=7^d$$

- Strassen showed $R(\langle n,n,n\rangle)=O(n^{\omega+\varepsilon})$ and v.v.
- Algebraic definition of ω

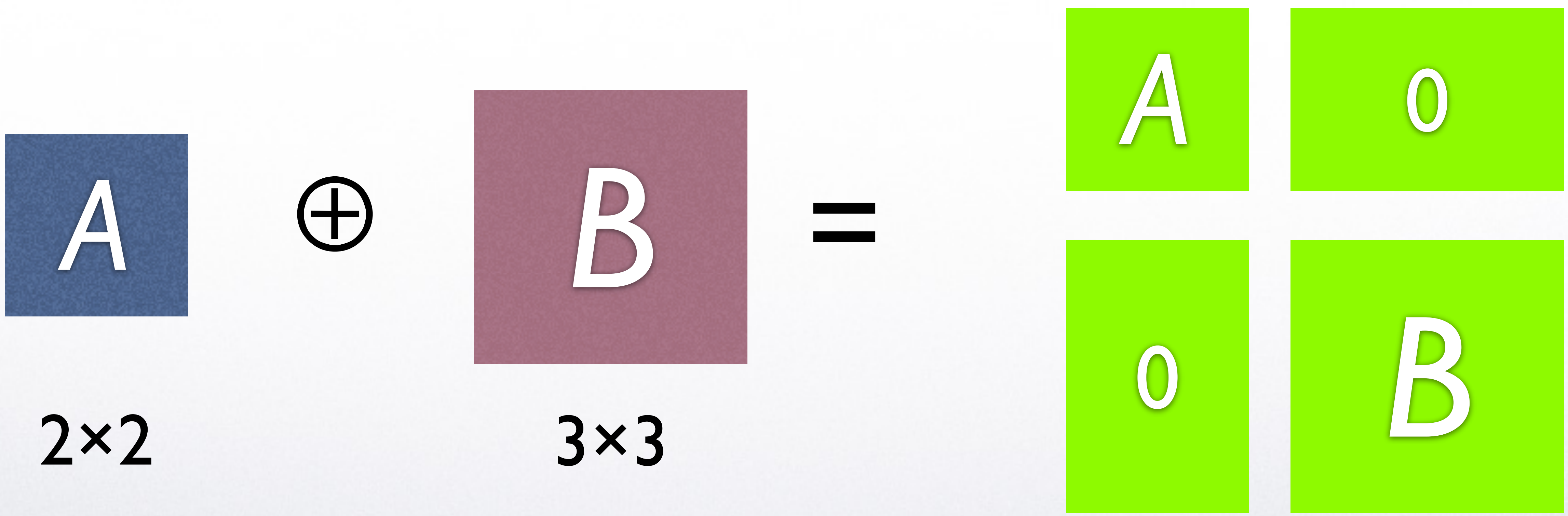
Tensor sum

Direct sum of matrices:



Tensor sum

Direct sum of matrices:



Tensor sum of tensors – 3D analog

5×5

Asymptotic sum inequality

Asymptotic sum inequality

Border rank

- Schönhage proved: $\underline{R}(\langle 4,1,4\rangle \oplus \langle 1,9,1\rangle) \leq 17$

Outer product Inner product

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} [y_1 y_2 y_3 y_4] \qquad [z_1 \ldots z_9] \begin{bmatrix} w_1 \\ \vdots \\ w_9 \end{bmatrix}$$

Asymptotic sum inequality

Border rank

- Schönhage proved: $\underline{R}(\langle 4,1,4\rangle \oplus \langle 1,9,1\rangle) \leq 17$
- Schönhage's asymptotic sum inequality:
  $\omega \leq 3\tau$ where $16^{\tau}+9^{\tau}=17$

Asymptotic sum inequality

Border rank

- Schönhage proved: $\underline{R}(\langle 4,1,4\rangle \oplus \langle 1,9,1\rangle) \leq 17$
- Schönhage's asymptotic sum inequality: $\omega \leq 3\tau$ where $16^{\tau}+9^{\tau}=17$
- Gives the bound $\omega < 2.55$

Asymptotic sum inequality

Border rank

- Schönhage proved: $\underline{R}(\langle 4,1,4\rangle \oplus \langle 1,9,1\rangle) \leq 17$
- Schönhage's asymptotic sum inequality: $\omega \leq 3\tau$ where $16^{\tau}+9^{\tau}=17$
- Gives the bound $\omega < 2.55$
- Applies to any tensor sum of matrix multiplication tensors

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\left(\sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0\right) \leq q+2$$

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\left(\sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0\right) \leq q+2$$

$$\underbrace{}_{\langle 1,1,q\rangle} \quad \underbrace{}_{\langle q,1,1\rangle} \quad \underbrace{}_{\langle 1,q,1\rangle}$$

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\left(\sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0\right) \leq q+2$$

$$\underbrace{}_{\langle 1,1,q \rangle} \quad \underbrace{}_{\langle q,1,1 \rangle} \quad \underbrace{}_{\langle 1,q,1 \rangle}$$

Problem: tensors not disjoint!

Laser method

What about more general tensors?

Coppersmith–Winograd "Easy identity"

$$\underline{R}\Big(\underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q\rangle} + \underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1\rangle} + \underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1\rangle}\Big) \leq q+2$$

Strassen's laser method:

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\Big(\underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q\rangle} + \underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1\rangle} + \underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1\rangle}\Big) \le q+2$$

Strassen's laser method:

- Take high tensor power

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\left(\sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0\right) \leq q+2$$

$$\underbrace{}_{\langle 1,1,q\rangle} \quad \underbrace{}_{\langle q,1,1\rangle} \quad \underbrace{}_{\langle 1,q,1\rangle}$$

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\Big(\underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q\rangle}+\underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1\rangle}+\underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1\rangle}\Big)\leq q+2$$

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method

What about more general tensors?

Coppersmith–Winograd
"Easy identity"

$$\underline{R}\left(\sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0\right) \leq q+2$$

$$\underbrace{}_{\langle 1,1,q\rangle} \quad \underbrace{}_{\langle q,1,1\rangle} \quad \underbrace{}_{\langle 1,q,1\rangle}$$

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

$\omega < 2.404$

Coppersmith–Winograd "Complicated identity"

$\underline{R}(T_{CW}) \leq q+2$, where

Coppersmith–Winograd tensor

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0$$

Coppersmith–Winograd "Complicated identity"

$\underline{R}(T_{CW}) \leq q+2$, where

Coppersmith–Winograd tensor

$$T_{CW} = \underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q \rangle} + \underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1 \rangle} + \underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1 \rangle} + \underbrace{x_0 y_0 z_{q+1}}_{\langle 1,1,1 \rangle} + \underbrace{x_0 y_{q+1} z_0}_{\langle 1,1,1 \rangle} + \underbrace{x_{q+1} y_0 z_0}_{\langle 1,1,1 \rangle}$$

Basis of all algorithms since 1987!

Coppersmith–Winograd "Complicated identity"

$\underline{R}(T_{CW}) \leq q+2$, where

$\omega < 2.388$

Coppersmith–Winograd tensor

$$T_{CW} = \underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q \rangle} + \underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1 \rangle} + \underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1 \rangle} + \underbrace{x_0 y_0 z_{q+1}}_{\langle 1,1,1 \rangle} + \underbrace{x_0 y_{q+1} z_0}_{\langle 1,1,1 \rangle} + \underbrace{x_{q+1} y_0 z_0}_{\langle 1,1,1 \rangle}$$

Basis of all algorithms since 1987!

Recursive laser method

$\underline{R}(T_{CW}{}^{\otimes 2}) \leq (q+2)^2$, where

$T_{CW}{}^{\otimes 2}$ = sum of 15 non-disjoint tensors:
12 matrix multiplication tensors &
3 complicated tensors

Recursive laser method

$\underline{R}(T_{CW}^{\otimes 2}) \leq (q+2)^2$, where

$T_{CW}^{\otimes 2}$ = sum of 15 non-disjoint tensors:
12 matrix multiplication tensors &
3 complicated tensors

Some resulting from merging!

Recursive laser method

$\underline{R}(T_{CW}^{\otimes 2}) \leq (q+2)^2$, where

Some resulting from merging!

$T_{CW}^{\otimes 2}$ = sum of 15 non-disjoint tensors:
12 matrix multiplication tensors &
3 complicated tensors

$$T_{112} = \underbrace{\sum_{i=1}^{q} x_{i0} y_{i0} z_{0(q+1)}}_{\langle 1,q,1 \rangle} + \underbrace{\sum_{i=1}^{q} x_{0i} y_{0i} z_{(q+1)0}}_{\langle 1,q,1 \rangle} + \underbrace{\sum_{i=1}^{q}\sum_{j=1}^{q} x_{i0} y_{0j} z_{ij}}_{\langle q,1,q \rangle} + \underbrace{\sum_{i=1}^{q}\sum_{j=1}^{q} x_{0j} y_{i0} z_{ij}}_{\langle q,1,q \rangle}$$

Recursive laser method

$\underline{R}(T_{CW}{}^{\otimes 2}) \leq (q+2)^2$, where

Some resulting from merging!

$T_{CW}{}^{\otimes 2}$ = sum of 15 non-disjoint tensors:
12 matrix multiplication tensors &
3 complicated tensors

Problem: asymptotic sum inequality only handles matrix multiplication tensors

Recursive laser method

$\underline{R}(T_{CW}^{\otimes 2}) \leq (q+2)^2$, where

Some resulting from merging!

$T_{CW}^{\otimes 2}$ = sum of 15 non-disjoint tensors:
12 matrix multiplication tensors &
3 complicated tensors

Problem: asymptotic sum inequality only handles matrix multiplication tensors

Solution: generalized asymptotic sum inequality handles tensors analyzed by laser method

Recursive laser method

- Analyzing $T_{CW}^{\otimes 2}$:

Recursive laser method

- Analyzing $T_{CW}^{\otimes 2}$:
  - Analyze T_{112} using laser method

Recursive laser method

- Analyzing $T_{CW}^{\otimes 2}$:
  - Analyze T_{112} using laser method
  - Analyze $T_{CW}^{\otimes 2}$ using generalized laser method

Recursive laser method

- Analyzing $T_{CW}{}^{\otimes 2}$:
  - Analyze T_{112} using laser method
  - Analyze $T_{CW}{}^{\otimes 2}$ using generalized laser method

$\omega < 2.376$

Analyzing powers of T_{CW}

Analyzing powers of T_{CW}

C & W	1987	T_{CW}	$\omega < 2.3871900$
C & W	1987	$T_{CW}^{\otimes 2}$	$\omega < 2.3754770$
Stothers	2010	$T_{CW}^{\otimes 4}$	$\omega < 2.3729269$
Williams	2012	$T_{CW}^{\otimes 8}$	$\omega < 2.3728642$
Le Gall	2014	$T_{CW}^{\otimes 16}$	$\omega < 2.3728640$
Le Gall	2014	$T_{CW}^{\otimes 32}$	$\omega < 2.3728639$

Source: Le Gall 2014

Analyzing powers of T_{CW}

C & W	1987	T_{CW}	$\omega < 2.3871900$
C & W	1987	$T_{CW}^{\otimes 2}$	$\omega < 2.3754770$
Stothers	2010	$T_{CW}^{\otimes 4}$	$\omega < 2.3729269$
Williams	2012	$T_{CW}^{\otimes 8}$	$\omega < 2.3728642$
Le Gall	2014	$T_{CW}^{\otimes 16}$	$\omega < 2.3728640$
Le Gall	2014	$T_{CW}^{\otimes 32}$	$\omega < 2.3728639$

Source: Le Gall 2014

What is the limit of this approach?

Our work

Our work

- Formalize recursive laser method
  Define $L(T^{\otimes k})$ = resulting bound on ω

Our work

- Formalize recursive laser method
  Define $L(T^{\otimes k})$ = resulting bound on ω
- Formulate "laser method with merging"
  Define $L_M(T)$ = resulting bound on ω

Our work

- Formalize recursive laser method
  Define $L(T^{\otimes k})$ = resulting bound on ω
- Formulate "laser method with merging"
  Define $L_M(T)$ = resulting bound on ω
- Prove that $L_M(T) \leq L(T^{\otimes k})$ for all k

Our work

Our work

- Main theorem: lower bound on $L_M(T)$

Our work

- Main theorem: lower bound on $L_M(T)$
- Application 1: $L_M(T_{CW^{\otimes 16}}) \geq 2.3725$ (cf. $L(T_{CW^{\otimes 32}}) \approx 2.3728$)

Our work

- Main theorem: lower bound on $L_M(T)$
- Application I: $L_M(T_{CW^{\otimes 16}}) \geq 2.3725$ (cf. $L(T_{CW^{\otimes 32}}) \approx 2.3728$)
- Application II: $L_M(T_{CW}) \geq 2.3078$ (do not expect to be tight)

Laser method with merging

Laser method with merging

Strassen's laser method:

Laser method with merging

Strassen's laser method:

- Take high tensor power

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging:

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging:

- Take high tensor power

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging:

- Take high tensor power
- Zero variables and merge tensors to obtain disjoint tensors

Laser method with merging

Strassen's laser method:

- Take high tensor power
- Zero variables to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging:

- Take high tensor power
- Zero variables and merge tensors to obtain disjoint tensors
- Apply asymptotic sum inequality

Laser method with merging

Example: T_{CW}

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0$$

Laser method with merging

Example: T_{CW}

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + \underbrace{x_0 y_{q+1} z_0}_{\langle 1,1,1 \rangle} + \underbrace{x_{q+1} y_0 z_0}_{\langle 1,1,1 \rangle}$$

Laser method with merging

Example: T_{CW}

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + \underbrace{x_0 y_{q+1} z_0}_{\langle 1,1,1 \rangle} + \underbrace{x_{q+1} y_0 z_0}_{\langle 1,1,1 \rangle}$$

$$x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0 \approx a_{11} \underbrace{b_{11} c_{11} + a_{12} b_{21}}_{\langle 1,2,1 \rangle} c_{11}$$

Laser method with merging

Example: T_{CW}

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + \underbrace{x_0 y_{q+1} z_0}_{\langle 1,1,1 \rangle} + \underbrace{x_{q+1} y_0 z_0}_{\langle 1,1,1 \rangle}$$

$$x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0 \approx a_{11} \underbrace{b_{11} c_{11} + a_{12} b_{21}}_{\langle 1,2,1 \rangle} c_{11}$$

Not representative!

Main lower bound

Main lower bound

- Goal: lower bound $L_M(T)$

Main lower bound

- Goal: lower bound $L_M(T)$
- Reduce to problem in extremal combinatorics: Maximal weight of structure satisfying certain constraints

Main lower bound

- Goal: lower bound $L_M(T)$
- Reduce to problem in extremal combinatorics: Maximal weight of structure satisfying certain constraints
  - *Structure*: components of $T^{\otimes N}$ after zeroing and merging

Main lower bound

- Goal: lower bound $L_M(T)$
- Reduce to problem in extremal combinatorics: Maximal weight of structure satisfying certain constraints
  - *Structure*: components of $T^{\otimes N}$ after zeroing and merging
  - *Constraints*: components are disjoint

Main lower bound

- Goal: lower bound $L_M(T)$
- Reduce to problem in extremal combinatorics: Maximal weight of structure satisfying certain constraints
  - *Structure*: components of $T^{\otimes N}$ after zeroing and merging
  - *Constraints*: components are disjoint
  - *Weight*: quantity in asymptotic sum inequality

Main lower bound

- Goal: lower bound $L_M(T)$
- Reduce to problem in extremal combinatorics: Maximal weight of structure satisfying certain constraints
  - *Structure*: components of $T^{\otimes N}$ after zeroing and merging
  - *Constraints*: components are disjoint
  - *Weight*: quantity in asymptotic sum inequality
- Upper bound using volume argument (inspired by Cohn, Kleinberg, Szegedy, Umans)

The lower bound

Coppersmith & Winograd showed

$$L(T_S) \leq 3\log_q \frac{q+2}{2^{h(1/3)}}$$

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 \qquad \underline{R}(T_S) \leq q+2$$

The lower bound

Coppersmith & Winograd showed

$$L(T_S) \leq 3\log_q \frac{q+2}{2^{h(1/3)}} \qquad q=8 \Rightarrow \omega<2.404$$

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 \qquad \underline{R}(T_S) \leq q+2$$

The lower bound

Coppersmith & Winograd showed

$$L(T_S) \leq 3\log_q \frac{q+2}{2^{h(1/3)}}$$

$q=8 \Rightarrow \omega<2.404$

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0$$

$\underline{R}(T_S) \leq q+2$

Cohn, Kleinberg, Szegedy, Umans: bound is tight

The lower bound

Coppersmith & Winograd showed

$$L(T_S) \leq 3\log_q \frac{q+2}{2^{h(1/3)}} \qquad q=8 \Rightarrow \omega<2.404$$

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 \qquad \underline{R}(T_S) \leq q+2$$

Cohn, Kleinberg, Szegedy, Umans: bound is tight

Our lower bound is extension of their methods

Partitioned tensors

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0$$

Partitioned tensors

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0$$

x variables

x_0 $X^{[0]}$

$x_1,\ldots,x_q$ $X^{[1]}$

Partitioned tensors

Coppersmith–Winograd "simple" tensor

$$T_S = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0$$

x variables		*y* variables		*z* variables	
x_0	$X^{[0]}$	y_0	$Y^{[0]}$	z_0	$Z^{[0]}$
$x_1,\ldots,x_q$	$X^{[1]}$	$y_1,\ldots,y_q$	$Y^{[1]}$	$z_1,\ldots,z_q$	$Z^{[1]}$

Partitioned tensors

Coppersmith–Winograd "simple" tensor

$$T_S = \underbrace{\sum_{i=1}^{q} x_0 y_i z_i}_{\langle 1,1,q\rangle^{[0,1,1]}} + \underbrace{\sum_{i=1}^{q} x_i y_0 z_i}_{\langle q,1,1\rangle^{[1,0,1]}} + \underbrace{\sum_{i=1}^{q} x_i y_i z_0}_{\langle 1,q,1\rangle^{[1,1,0]}}$$

x variables		y variables		z variables	
x_0	$X^{[0]}$	y_0	$Y^{[0]}$	z_0	$Z^{[0]}$
$x_1,\ldots,x_q$	$X^{[1]}$	$y_1,\ldots,y_q$	$Y^{[1]}$	$z_1,\ldots,z_q$	$Z^{[1]}$

Partitioned tensors

Can represent powers of T_S using *index triples*:

Partitioned tensors

Can represent powers of T_S using *index triples*:

T_S = [0,1,1] [1,0,1] [1,1,0]

Partitioned tensors

Can represent powers of T_S using *index triples*:

$$T_S = [0,1,1]\ [1,0,1]\ [1,1,0]$$

$$T_S^{\otimes 2} = \begin{matrix} [00,11,11] & [01,10,11] & [01,11,10] \\ [10,01,11] & [11,00,11] & [11,01,10] \\ [10,11,01] & [11,10,01] & [11,11,00] \end{matrix}$$

Partitioned tensors

Can represent powers of T_S using *index triples*:

$$T_S \quad = [0,1,1]\ [1,0,1]\ [1,1,0]$$

$$T_S^{\otimes 2} = \begin{matrix} [00,11,11]\ [01,10,11]\ [01,11,10] \\ [10,01,11]\ [11,00,11]\ [11,01,10] \\ [10,11,01]\ [11,10,01]\ [11,11,00] \end{matrix}$$

Zeroing variables: retain only index triples $[i,j,k]$ where $i \in I$, $j \in J$, $k \in K$

Partitioned tensors

Can represent powers of T_S using *index triples*:

$$T_S = [0,1,1]\ [1,0,1]\ [1,1,0]$$

$$T_S^{\otimes 2} = \begin{matrix} [00,11,11]\ [01,10,11]\ [01,11,10] \\ [10,01,11]\ [11,00,11]\ [11,01,10] \\ [10,11,01]\ [11,10,01]\ [11,11,00] \end{matrix}$$

Zeroing variables: retain only index triples $[i,j,k]$ where $i \in I$, $j \in J$, $k \in K$

Resulting sum is *disjoint* if no i,j,k repeats

Partitioned tensors

Can represent powers of T_S using *index triples*:

$$T_S \quad = [0,1,1]\ [1,0,1]\ [1,1,0]$$

$$T_S^{\otimes 2} = \begin{array}{l} [00,11,11]\ [01,10,11]\ [01,11,10] \\ [10,01,11]\ [11,00,11]\ [11,01,10] \\ [10,11,01]\ [11,10,01]\ [11,11,00] \end{array}$$

Zeroing variables: retain only index triples $[i,j,k]$ where $i \in I$, $j \in J$, $k \in K$

Resulting sum is *disjoint* if no i,j,k repeats

Capacity C_N: max no. of disjoint index triples for $T_S^{\otimes N}$

Capacity

Capacity

Capacity C_N: max no. of disjoint index triples for $T_S{}^{\otimes N}$

Capacity

Capacity C_N: max no. of disjoint index triples for $T_S{}^{\otimes N}$

Example: $C_2 \geq 2$

Capacity

Capacity C_N: max no. of disjoint index triples for $T_S{}^{\otimes N}$

Example: $C_2 \geq 2$

$$T_S{}^{\otimes 2} = \begin{matrix} [00,11,11] & [01,10,11] & [01,11,10] \\ [10,01,11] & [11,00,11] & [11,01,10] \\ [10,11,01] & [11,10,01] & [11,11,00] \end{matrix}$$

$I = \{00,11\}$ $J = \{01,11\}$ $K = \{10,11\}$

Capacity

Capacity C_N: max no. of disjoint index triples for $T_S{}^{\otimes N}$

Example: $C_2 \geq 2$

$$T_S{}^{\otimes 2} = \begin{matrix} [00,11,11] & [01,10,11] & [01,11,10] \\ [10,01,11] & [11,00,11] & [11,01,10] \\ [10,11,01] & [11,10,01] & [11,11,00] \end{matrix}$$

$I = \{00,11\}$ $J = \{01,11\}$ $K = \{10,11\}$

Asymptotic sum inequality: $\omega \leq 3\log_{q^N} \frac{(q+2)^N}{C_N}$

Capacity

Capacity C_N: max no. of disjoint index triples for $T_S{}^{\otimes N}$

Example: $C_2 \geq 2$

$$T_S{}^{\otimes 2} = \begin{matrix} [00,11,11] & [01,10,11] & [01,11,10] \\ [10,01,11] & [11,00,11] & [11,01,10] \\ [10,11,01] & [11,10,01] & [11,11,00] \end{matrix}$$

$I = \{00,11\}$ $J = \{01,11\}$ $K = \{10,11\}$

Asymptotic sum inequality: $\omega \leq 3\log_{q^N} \frac{(q+2)^N}{C_N}$

Taking the limit $N \to \infty$: $\omega \leq 3\log_q \frac{q+2}{C}$, where $C = \lim_{N\to\infty} C_N{}^{1/N}$

Capacity

Capacity

- Consider all index triples arising from $N/3$ [0,1,1]s, $N/3$ [1,0,1]s, $N/3$ [1,1,0]s

Capacity

- Consider all index triples arising from $N/3$ [0,1,1]s, $N/3$ [1,0,1]s, $N/3$ [1,1,0]s
- At most $\binom{N}{N/3} \leq 2^{h(1/3)N}$ different x indices

Capacity

- Consider all index triples arising from $N/3$ [0,1,1]s, $N/3$ [1,0,1]s, $N/3$ [1,1,0]s
- At most $\binom{N}{N/3} \leq 2^{h(1/3)N}$ different x indices
- So contribution is at most $2^{h(1/3)N}$

Capacity

- Consider all index triples arising from $N/3$ [0,1,1]s, $N/3$ [1,0,1]s, $N/3$ [1,1,0]s
- At most $\binom{N}{N/3} \leq 2^{h(1/3)N}$ different x indices
- So contribution is at most $2^{h(1/3)N}$
- Same is true for all $O(N^2)$ *types*

Capacity

- Consider all index triples arising from $N/3$ [0,1,1]s, $N/3$ [1,0,1]s, $N/3$ [1,1,0]s
- At most $\binom{N}{N/3} \leq 2^{h(1/3)N}$ different x indices
- So contribution is at most $2^{h(1/3)N}$
- Same is true for all $O(N^2)$ *types*
- So $C_N \leq O(N^2\ 2^{h(1/3)N}) \Rightarrow C \leq 2^{h(1/3)}$

Capacity

Capacity

- We proved $C_N \leq O(N^2\ 2^{h(1/3)N}) \Rightarrow C \leq 2^{h(1/3)}$

Capacity

- We proved $C_N \leq O(N^2\, 2^{h(1/3)N}) \Rightarrow C \leq 2^{h(1/3)}$
- Since $L(T_S) = 3\log_q \frac{q+2}{C}$, we deduce $L(T_S) \geq \frac{q+2}{2^{h(1/3)}}$

Capacity

- We proved $C_N \leq O(N^2\, 2^{h(1/3)N}) \Rightarrow C \leq 2^{h(1/3)}$
- Since $L(T_S) = 3\log_q \frac{q+2}{C}$, we deduce $L(T_S) \geq \frac{q+2}{2^{h(1/3)}}$
- Coppersmith and Winograd showed $C \geq 2^{h(1/3)}$ using a complicated combinatorial construction

Capacity

- We proved $C_N \leq O(N^2\, 2^{h(1/3)N}) \Rightarrow C \leq 2^{h(1/3)}$
- Since $L(T_S) = 3\log_q \frac{q+2}{C}$, we deduce $L(T_S) \geq \frac{q+2}{2^{h(1/3)}}$
- Coppersmith and Winograd showed $C \geq 2^{h(1/3)}$ using a complicated combinatorial construction
- It follows that $L(T_S) = \frac{q+2}{2^{h(1/3)}}$

More on merging

Coppersmith–Winograd tensor $T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0$

x variables		y variables		z variables	
x_0	$X[0]$	y_0	$Y[0]$	z_0	$Z[0]$
$x_1,\ldots,x_q$	$X[1]$	$y_1,\ldots,y_q$	$Y[1]$	$z_1,\ldots,z_q$	$Z[1]$
x_{q+1}	$X[2]$	y_{q+1}	$Y[2]$	z_{q+1}	$Z[2]$

More on merging

Coppersmith–Winograd tensor

$$T_{CW} = \sum_{i=1}^{q} x_0 y_i z_i + \sum_{i=1}^{q} x_i y_0 z_i + \sum_{i=1}^{q} x_i y_i z_0 + x_0 y_0 z_{q+1} + x_0 y_{q+1} z_0 + x_{q+1} y_0 z_0$$

$$\underbrace{}_{\langle 1,1,q\rangle^{[0,1,1]}} \underbrace{}_{\langle q,1,1\rangle^{[0,1,1]}} \underbrace{}_{\langle 1,q,1\rangle^{[0,1,1]}} \underbrace{}_{\langle 1,1,1\rangle^{[0,0,2]}} \underbrace{}_{\langle 1,1,1\rangle^{[0,2,0]}} \underbrace{}_{\langle 1,1,1\rangle^{[2,0,0]}}$$

x variables		y variables		z variables	
x_0	$X[0]$	y_0	$Y[0]$	z_0	$Z[0]$
$x_1,\ldots,x_q$	$X[1]$	$y_1,\ldots,y_q$	$Y[1]$	$z_1,\ldots,z_q$	$Z[1]$
x_{q+1}	$X[2]$	y_{q+1}	$Y[2]$	z_{q+1}	$Z[2]$

More on merging

Actual merging in $T_{CW^{\otimes 2}}$:

$$x_0y_0z_{q+1}\otimes x_0y_{q+1}z_0+x_0y_{q+1}z_0\otimes x_0y_0z_{q+1}=$$

$$x_{00}y_{0q+1}z_{q+10}+x_{00}y_{0q+1}z_{q+10}\approx\langle 1,1,2\rangle$$

More on merging

Actual merging in $T_{CW^{\otimes 2}}$:

$$x_0 y_0 z_{q+1} \otimes x_0 y_{q+1} z_0 + x_0 y_{q+1} z_0 \otimes x_0 y_0 z_{q+1} =$$

$$x_{00} y_{0q+1} z_{q+10} + x_{00} y_{0q+1} z_{q+10} \approx \langle 1,1,2 \rangle$$

Corresponds to merging [00,02,20] and [00,20,02]

More on merging

Actual merging in $T_{CW^{\otimes 2}}$:

$$x_0 y_0 z_{q+1} \otimes x_0 y_{q+1} z_0 + x_0 y_{q+1} z_0 \otimes x_0 y_0 z_{q+1} =$$

$$x_{00} y_{0q+1} z_{q+10} + x_{00} y_{0q+1} z_{q+10} \approx \langle 1,1,2 \rangle$$

Corresponds to merging [00,02,20] and [00,20,02]

In any merging, for each *t*: either

i_t = 0 for all [*i*,*j*,*k*] (*x-constant*) or

j_t = 0 for all [*i*,*j*,*k*] (*y-constant*) or

k_t = 0 for all [*i*,*j*,*k*] (*z-constant*)

The lower bound

The lower bound

- Line = collection of merged index triples

The lower bound

- Line = collection of merged index triples
- Consider lines with $n/3$ x-constant, $n/3$ y-constant, $n/3$ z-constant coordinates

The lower bound

- Line = collection of merged index triples
- Consider lines with $n/3$ x-constant, $n/3$ y-constant, $n/3$ z-constant coordinates
- Consider index triples with $\alpha/3$ each of [110],[101],[011], $\beta/3$ each of [200],[020],[002]

The lower bound

- Line = collection of merged index triples
- Consider lines with $n/3$ x-constant, $n/3$ y-constant, $n/3$ z-constant coordinates
- Consider index triples with $\alpha/3$ each of [110],[101],[011], $\beta/3$ each of [200],[020],[002]
- Upper bound no. of index triples in line

The lower bound

- Line = collection of merged index triples
- Consider lines with $n/3$ x-constant, $n/3$ y-constant, $n/3$ z-constant coordinates
- Consider index triples with $\alpha/3$ each of [110],[101],[011], $\beta/3$ each of [200],[020],[002]
- Upper bound no. of index triples in line
- Upper bound no. of index triples

The lower bound

- Line = collection of merged index triples
- Consider lines with $n/3$ x-constant, $n/3$ y-constant, $n/3$ z-constant coordinates
- Consider index triples with $\alpha/3$ each of [110],[101],[011], $\beta/3$ each of [200],[020],[002]
- Upper bound no. of index triples in line
- Upper bound no. of index triples
- Deduce upper bound on capacity

What now?

What now?

- Find better ways of analyzing T_{CW}
  (cannot prove anything better than $\omega < 2.3078$)

What now?

- Find better ways of analyzing T_{CW}
  (cannot prove anything better than $\omega < 2.3078$)
- Find better identities

What now?

- Find better ways of analyzing T_{CW} (cannot prove anything better than $\omega < 2.3078$)
- Find better identities
- Group-theoretic method of Cohn & Umans

What now?

- Find better ways of analyzing T_{CW}
(cannot prove anything better than $\omega < 2.3078$)
- Find better identities
- Group-theoretic method of Cohn & Umans
- Completely different methods?

Questions?