THEORY OF COMPUTING LIBRARY
GRADUATE SURVEYS 5 (2013), pp. 1–60
www.theoryofcomputing.org

Fast Matrix Multiplication

Markus Bläser*

Received March 7, 2013; Published December 24, 2013

Abstract: We give an overview of the history of fast algorithms for matrix multiplication. Along the way, we look at some other fundamental problems in algebraic complexity like polynomial evaluation.

This exposition is self-contained. To make it accessible to a broad audience, we only assume a minimal mathematical background: basic linear algebra, familiarity with polynomials in several variables over rings, and rudimentary knowledge in combinatorics should be sufficient to read (and understand) this article. This means that we have to treat tensors in a very concrete way (which might annoy people coming from mathematics), occasionally prove basic results from combinatorics, and solve recursive inequalities explicitly (because we want to annoy people with a background in theoretical computer science, too).

ACM Classification: F.2.2

AMS Classification: 68Q17, 68Q25

Key words and phrases: fast matrix multiplication, bilinear complexity, tensor rank

1 Introduction

Given two $n \times n$-matrices $x = (x_{ik})$ and $y = (y_{kj})$ whose entries are indeterminates over some field K, we want to compute their product $xy = (z_{ij})$. The entries z_{ij} are given by the following well-known bilinear forms

$$z_{ij} = \sum_{k=1}^{n} x_{ik} y_{kj}, \qquad 1 \le i, j \le n. \tag{1.1}$$

Each z_{ij} is the sum of n products. Thus every z_{ij} can be computed with n multiplications and $n-1$ additions. This gives an algorithm that altogether uses n^3 multiplications and $n^2(n-1)$ additions.

*Supported by DFG grant BL 511/10-1.

 DOI: 10.4086/toc.gs.2013.005

This algorithm looks so natural and intuitive that it is very hard to imagine that there is better way to multiply matrices. However, in 1969, Strassen [31] found a way to multiply 2×2-matrices with only 7 multiplications but 18 additions.

Let z_{ij}, $1 \le i, j \le 2$, be given by

$$\begin{pmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix}.$$

We compute the seven products

$$\begin{aligned}
p_1 &= (x_{11}+x_{22})(y_{11}+y_{22}), \\
p_2 &= (x_{21}+x_{22})y_{11}, \\
p_3 &= x_{11}(y_{12}-y_{22}), \\
p_4 &= x_{22}(-y_{11}+y_{21}), \\
p_5 &= (x_{11}+x_{12})y_{22}, \\
p_6 &= (-x_{11}+x_{21})(y_{11}+y_{12}), \\
p_7 &= (x_{12}-x_{22})(y_{21}+y_{22}).
\end{aligned}$$

We can express each of the z_{ij} as a linear combination of these seven products, namely,

$$\begin{pmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{pmatrix} = \begin{pmatrix} p_1+p_4-p_5+p_7 & p_3+p_5 \\ p_2+p_4 & p_1+p_3-p_2+p_6 \end{pmatrix}.$$

The number of multiplications in this algorithm is optimal (we will see this later), but already for 3×3-matrices, the optimal number of multiplication is not known. We know that it lies between 19 and 23, cf. [5, 21].

But is it really interesting to save one multiplication but have an additional 14 additions instead?[1] The important point is that Strassen's algorithm does not only work over fields but also over noncommutative rings. In particular, the entries of the 2×2-matrices could also be matrices and we can apply the algorithm recursively. And for matrices, multiplications—at least if we use the naive method—are much more expensive than additions, namely $O(n^3)$ compared to n^2.

Proposition 1.1. *One can multiply $n\times n$-matrices with $O(n^{\log_2 7})$ arithmetic operations (and even without using divisions).*[2]

[1] There is a variant of Strassen's algorithm that uses only 15 additions [38]. However, de Groote [15] showed that, using an appropriate notion of equivalence, there is only one algorithm for multiplying 2×2-matrices using seven multiplications. And one can even show that 15 additions is optimal, i. e., every algorithms that uses only seven multiplications needs at least 15 additions [7].

[2] What is an arithmetic operation? We will make this precise in the next section. For the moment, we compute over the field of rational functions $K(x_{ij}, y_{ij} \mid 1 \le i, j \le n)$. We start with the constants from K and the indeterminates x_{ij} and y_{ij}. Then we can take any two of the elements that we computed so far and compute their product, their quotient (if the second element is not zero), their sum, or their difference. We are done if we have computed all the z_{ij} in (1.1).

Proof. W. l. o. g. let $n = 2^\ell, \ell \in \mathbb{N}$. If this is not the case, then we can embed our matrices into matrices whose size is the next largest power of two and fill the remaining positions with zeros.[3] Since the algorithm does not use any divisions, substituting an indeterminate by a concrete value will not cause a division by zero.

We will show by induction in ℓ that we can multiply with 7^ℓ multiplications and $6 \cdot (7^\ell - 4^\ell)$ additions/subtractions.

Induction start ($\ell = 1$): See above.

Induction step ($\ell - 1 \to \ell$): We think of our matrices as 2×2-matrices whose entries are $2^{\ell-1} \times 2^{\ell-1}$ matrices, i. e., we have the following block structure:

$$\left(\begin{array}{c|c} & \\ \hline & \end{array}\right) \cdot \left(\begin{array}{c|c} & \\ \hline & \end{array}\right) = \left(\begin{array}{c|c} & \\ \hline & \end{array}\right).$$

We can multiply these matrices using Strassen's algorithm with seven multiplications of $2^{\ell-1} \times 2^{\ell-1}$-matrices and 18 additions of $2^{\ell-1} \times 2^{\ell-1}$-matrices.

For the seven multiplications of the $2^{\ell-1} \times 2^{\ell-1}$-matrices, we need $7 \cdot 7^{\ell-1} = 7^\ell$ multiplications by the induction hypothesis. And we need $7 \cdot 6 \cdot (7^{\ell-1} - 4^{\ell-1})$ additions/subtractions for the seven multiplications. The 18 additions of $2^{\ell-1} \times 2^{\ell-1}$-matrices need $18 \cdot (2^{\ell-1})^2$ additions. Thus the total number of additions/subtractions is

$$7 \cdot 6 \cdot (7^{\ell-1} - 4^{\ell-1}) + 18 \cdot (2^{\ell-1})^2 = 6 \cdot (7^\ell - 7 \cdot 4^{\ell-1} + 3 \cdot 4^{\ell-1}) = 6 \cdot (7^\ell - 4^\ell).$$

This finishes the induction step. Since $7^\ell = n^{\log_2 7}$, we are done. □

2 Computations and costs

2.1 Karatsuba's algorithm

Let us start with a very simple computational problem, the multiplication of univariate polynomials of degree one. We are given two polynomials $a_0 + a_1 X$ and $b_0 + b_1 X$ and we want to compute the coefficients c_0, c_1, c_2 of their product, which are given by

$$(a_0 + a_1 \cdot X) \cdot (b_0 + b_1 \cdot X) = \underbrace{a_0 b_0}_{=:c_0} + \underbrace{(a_0 b_1 + a_1 b_0)}_{=:c_1} \cdot X + \underbrace{a_1 b_1}_{=:c_2} \cdot X^2.$$

We here consider the coefficients of the two polynomials to be indeterminates over some field K. The coefficients of the product are rational functions (in fact, bilinear forms) in a_0, a_1, b_0, b_1, so the following model of computation seems to fit well. We have a sequence $(w_1, w_2, \dots, w_\ell)$ of rational functions such that each w_i is either a_0, a_1, b_0, or b_1 (inputs) or a constant from K or can be expressed as $w_i = w_j \text{ op } w_k$ for indices $j, k < i$ and op is one of the arithmetic operations $\cdot$, $/$, $+$, or $-$.

[3] Asymptotically, this is o.k. For practical purposes, it is better to directly recurse if n is even and add a row and column with zeros if n is odd.

Here is one possible computation that computes the three coefficients c_0, c_1, and c_2.

$$\begin{array}{rrcl}
 & w_1 & = & a_0\,, \\
 & w_2 & = & a_1\,, \\
 & w_3 & = & b_0\,, \\
 & w_4 & = & b_1\,, \\
(c_0 =) & w_5 & = & w_1 \cdot w_3\,, \\
(c_2 =) & w_6 & = & w_2 \cdot w_4\,, \\
 & w_7 & = & w_1 + w_2\,, \\
 & w_8 & = & w_3 + w_4\,, \\
 & w_9 & = & w_7 \cdot w_8\,, \\
 & w_{10} & = & w_5 + w_6\,, \\
(c_1 =) & w_{11} & = & w_9 - w_{10}\,.
\end{array}$$

The above computation only uses three multiplications instead of four, which the naive algorithm needs. This is also called *Karatsuba's algorithm* [20].[4] Like Strassen's algorithm, it can be generalized to higher degree polynomials. If we have two polynomials $A(X) = \sum_{i=0}^{n} a_i X^i$ and $B(X) = \sum_{j=0}^{n} b_j X^j$ with $n = 2^\ell - 1$, then we split the two polynomials into halves, that is, $A(X) = A_0(X) + X^{(n+1)/2} A_1(X)$ with $A_0(X) = \sum_{i=0}^{(n+1)/2-1} a_i X^i$ and $A_1(X) = \sum_{i=0}^{(n+1)/2-1} a_{(n+1)/2+i} X^i$ and the same for B. Then we multiply these polynomials using the above scheme with A_0 taking the role of a_0 and A_1 taking the role of a_1 and the same for B. All multiplications of polynomials of degree $(n+1)/2-1$ are performed recursively. Let $N(n)$ denote the number of arithmetic operations that the above algorithm needs to multiply polynomial of degree $\le n$. The algorithm above gives the following recursive equation

$$N(n) = 3 \cdot N((n+1)/2 - 1) + O(n) \qquad \text{and} \qquad N(2) = 7\,.$$

Similarly to the analysis of Strassen's algorithm, one can show that $N(n) = O(n^{\log_2 3})$. Karatsuba's algorithm again trades one multiplication for a bunch of additional additions which is bad for degree one polynomials but good in general, since polynomial addition only needs n operations but polynomial multiplication—at least when using the naive method—is much more expensive, namely, $O(n^2)$.

2.2 A general model

We provide a framework to define computations and costs that is general enough to cover all the examples that we will look at. For a set S, let $\mathrm{fin}(S)$ denote the set of all finite subsets of S.

Definition 2.1 (Computation structure)**.** A computation structure is a set M together with a mapping $\gamma : M \times \mathrm{fin}(M) \to [0;\infty]$ such that

1. $\mathrm{im}(\gamma)$ is well ordered, that is, every non-empty subset of $\mathrm{im}(\gamma)$ has a minimum,
2. $\gamma(w,U) = 0$ if $w \in U$,
3. $U \subseteq V \Rightarrow \gamma(w,V) \le \gamma(w,U)$ for all $w \in M$, $U, V \subseteq \mathrm{fin}(M)$.

[4] See [19] why Ofman is a coauthor and why this paper was not even written by Karatsuba.

M is the set of objects that we are computing with. $\gamma(w,U)$ is the cost of computing w from U "in one step." In the example of polynomial multiplication of degree one in the previous subsection, M is the set of all rational functions in a_0, a_1, b_0, b_1. If we want to count the number of arithmetic operations of Karatsuba's algorithm, then $\gamma(w,U) = 0$ if $w \in U$. ("There is no cost if we already computed w.") We have $\gamma(w,U) = 1$ if there are $u, v \in U$ such that $w = u \operatorname{op} v$. ("$w$ can be computed from u and v with one arithmetical operation.") In all other cases $\gamma(w,U) = \infty$. ("w cannot be computed in one step from U.")

Often, we have a set M together with some operations $\phi : M^s \to M$ of some arity s. If we assign to each such operation a cost, then this induces a computation structure in a very natural way.

Definition 2.2. A structure $(M, \phi_1, \phi_2, \ldots)$ with (partial) operations $\phi_j : M^{s_j} \to M$ and a cost function $¢ : \{\phi_1, \phi_2, \ldots\} \to [0;\infty]$ such that $\operatorname{im}(¢)$ is well ordered induces a computation structure in the following way:

$$\gamma(w,U) := \min\{¢(\phi_j) \mid \exists u_1, \ldots, u_{s_j} \in U : w = \phi_j(u_1, \ldots, u_{s_j})\}.$$

If the minimum is taken over the empty set, then we set $\gamma(w,U) = \infty$. If $w \in U$, then $\gamma(w,U) = 0$.

Remark 2.3 (for hackers)**.** We can always achieve $\gamma(w,U) = 0$ by adding the function $\phi_0 = \operatorname{id}$ to the structure with $¢(\phi_0) = 0$.

Definition 2.4 (Computation)**.**

1. A sequence $\beta = (w_1, ..., w_m)$ of elements in M is a computation with input $X \subseteq M$ if

$$\forall j \le m : \quad w_j \in X \ \vee \ \gamma(w_j, V_j) < \infty \ \text{ where } V_j = \{w_1, ..., w_{j-1}\}$$

2. β computes a set $Y \in \operatorname{fin}(M)$ if in addition $Y \subseteq \{w_1, ..., w_m\}$.

3. The cost of β is $\Gamma(\beta, X) \overset{Def}{=} \sum_{j=1}^{m} \gamma(w_j, V_j)$.

In a computation, every w_i can be computed from elements previously computed, i. e., elements in V_j or from elements in X ("inputs"). The cost of a computation is the sum of the costs of the individuals steps.

Definition 2.5 (Complexity)**.** The complexity of Y given X is defined by

$$C(Y,X) := \min\{\Gamma(\beta, X) \mid \beta \text{ computes } Y \text{ from } X\}.$$

The complexity of a set Y is nothing but the cost of a cheapest computation that computes Y.

Notation 2.6.

1. If we compute only one element y, we will write $C(y,X)$ instead of $C(\{y\},X)$ and so on.

2. If $X = \emptyset$ or X is clear from the context, then we will just write $C(Y)$.

2.3 Examples

The following computation structure will appear quite often in these notes.

Example 2.7 (Ostrowski measure)**.** Our structure is $M = K(X_1,\dots,X_n)$, the field of rational functions in indeterminates $X_1,\dots,X_n$. We have four (or three) operations of arity 2, namely, multiplication, division, addition, and subtraction. Division is a partial operation which is only defined if the second input is nonzero (as a rational function). If we are only interested in computing polynomials, we might occasionally disallow divisions. For every $\lambda \in K$, there is an operation $\lambda\cdot$ of arity 1, the multiplication with the scalar λ. The cost is given by:

Operation	Arity	Cost
$\cdot, /$	2	1
$+, -$	2	0
$\lambda\cdot$	1	0

While in today's computer chips, multiplication takes about the same number of cycles as addition, Strassen's algorithm and also Karatsuba's algorithm show that this is nevertheless a meaningful way of charging cost.

The complexity induced by the Ostrowski measure will be denoted by $\mathrm{C}^{*/}$, or C^{*} if we disallow divisions. In particular, Karatsuba's algorithm yields $\mathrm{C}^{*/}(\{c_0,c_1,c_2\},\{a_0,a_1,b_0,b_1\}) = 3$. (The lower bound follows from the fact, that c_0,c_1,c_2 are linearly independent over K.)

Example 2.8 (Addition chains)**.** Our structure is $M = \mathbb{N}$ with the following operations:

Operation	Arity	Cost
1	0	0
$+$	2	1

$C(n)$ measures how many additions we need to generate n from 1.

Additions chains are motivated by the problem of computing a power X^n from X with as few multiplications as possible. We have $\log n \le C(n) \le 2\log n$. The lower bound follows from the fact that we can at most double the largest number computed so far with one more addition. The upper bound is the well-known "square and multiply" algorithm. This is an old problem from the 1930s, which goes back to Scholz [26] and Brauer [6], but quite some challenging questions still remain open.

Research problem 2.9. Prove the *Scholz-Brauer conjecture*:

$$C(2^n - 1) \le n + C(n) - 1 \qquad \text{for all } n \in \mathbb{N}.$$

Research problem 2.10. Prove *Stolarsky's conjecture* [29]:

$$C(n) \ge \log n + \log(q(n)) \qquad \text{for all } n \in \mathbb{N},$$

where $q(n)$ is the sum of the bits of the binary expansion of n. Schönhage [27] proved that

$$C(n) \ge \log n + \log(q(n)) - 2.13\,.$$

3 Evaluation of polynomials

Let us start with a simple example, the evaluation of univariate polynomials. Our input are the coefficients $a_0, \ldots, a_n$ of the polynomial and the point x at which we want to evaluate the polynomial. We model them as indeterminates, so our set $M = K_0(a_0, \ldots, a_n, x)$. We are interested in determining $C(f, \{a_0, \ldots, a_n, x\})$ where

$$f = a_0 + a_1 x + \cdots + a_n x^n \in K_0(a_0, \ldots, a_n, x) .$$

A well known algorithm to compute f is *Horner's scheme*. We write f as

$$f = ((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_0 .$$

This representation immediately gives a way to compute f with n multiplications and n additions. We will show that this is best possible: Even if we can make as many additions/subtractions as we want, we still need n multiplications/divisions. And even if we are allowed to perform as many multiplications/divisions as we want, n additions/subtractions are required. In the former case, we will use the well-known Ostrowski measure. In the latter case, we will use the so-called *additive complexity*, denoted by C^+, which is "the opposite" of the Ostrowski model. Here multiplications and divisions are for free but additions and subtractions count.

Operation	Costs	
	$\mathrm{C}^{*/}$	C^+
$\cdot, /$	1	0
$+, -$	0	1
$\lambda \cdot$	0	0
$p \in K_0(x)$	0	0

We will even allow that we can get elements from $K := K_0(x)$ for free (operation with arity zero). So we, e. g., can compute arbitrary powers of x at no cost. (This is a special feature of this section. In general, this is neither the case under the Ostrowski measure nor under the additive measure.)

Theorem 3.1. *Let $a_0, \ldots, a_n, x$ be indeterminates over K_0 and $f = a_0 + a_1 x + \cdots + a_n x^n$. Then $\mathrm{C}^{*/}(f) \geq n$ and $\mathrm{C}^+(f) \geq n$. This is even true if all elements from $K_0(x)$ are free of cost.*

The question about the optimality of Horner's scheme was raised by Ostrowski [23]. It is one of the founding problems of algebraic complexity theory. It took one decade until Pan [24] was able to prove that Horner's scheme is optimal with respect to multiplications. Prior to this, Motzkin [22] proved that it is optimal with respect to additions. We will prove both results in the next two subsections.

3.1 Multiplications

The first statement of Theorem 3.1 is implied by the following lower bound due to Winograd [37].

Theorem 3.2. *Let $K_0 \subseteq K$ be fields, $Z = \{z_1, \dots, z_n\}$ be indeterminates and $F = \{f_1, \dots, f_m\}$ where $f_\mu = \sum_{\nu=1}^{n} p_{\mu,\nu} z_\nu + q_\mu$ with $p_{\mu\nu}, q_\mu \in K$, $1 \le \mu \le m$. Then $\mathrm{C}^{*/}(F, Z) \ge r - m$ where*

$$r = \text{col-rk}_{K_0} \begin{pmatrix} p_{11} & \dots & p_{1n} & 1 & \dots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ p_{m1} & \dots & p_{mn} & 0 & \dots & 1 \end{pmatrix}.$$

We get the first part of Theorem 3.1 from Theorem 3.2 as follows: We set

$$\begin{aligned} K &= K_0(x)\,, \\ z_\nu &= a_\nu\,, \\ m &= 1\,, \\ f_1 &= f\,, \\ p_{1\nu} &= x^\nu\,, \qquad 1 \le \nu \le n\,, \\ q_1 &= a_0\,. \end{aligned}$$

Then $P = (x, x^2, \dots, x^n, 1)$ and $\text{col-rk}_{K_0} P = n + 1$.[5] We get $\mathrm{C}^{*/}(f_1, \{a_0, \dots, a_n\}) \ge n + 1 - 1 = n$ by Theorem 3.2.

Proof of Theorem 3.2. The proof is by induction in n.
Induction start ($n = 0$): We have

$$P = \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix}$$

and therefore, $r = m$. Thus $\mathrm{C}^{*/}(F) \ge 0 = r - m$.
Induction step ($n - 1 \to n$): If $r = m$, then there is nothing to show. Thus we can assume that $r > m$. We claim that in this case, $\mathrm{C}^{*/}(F, Z) \ge 1$. This is due to the fact that the set of all rational functions that can be computed with zero cost is

$$W_0 = \{w \in K(z_1, \dots, z_m) \mid C(w, Z) = 0\} = K + K_0 z_1 + K_0 z_2 + \dots + K_0 z_n\,.$$

(Clearly, every element in W_0 can be computed without any cost. But W_0 is also closed under all operations that are free of cost.) If $r > m$, then there are μ and i such that $p_{\mu,i} \notin K_0$ and therefore $f_\mu \notin W_0$.

W. l. o. g. K_0 is infinite, because if we replace K_0 by $K_0(t)$ for some indeterminate t, the complexity cannot go up, since every computation over K_0 is certainly a computation over $K_0(t)$. W. l. o. g. $f_\mu \neq 0$ for all $1 \le \mu \le m$.

Let $\beta = (w_1, \dots, w_\ell)$ be an optimal computation for F and let each $w_\lambda = p_\lambda / q_\lambda$ with $p_\lambda, q_\lambda \in K_0[z_1, \dots, z_n]$. Let j be minimal such that $\gamma(w_j, V_j) = 1$, where $V_j = \{w_1, \dots, w_{j-1}\}$. Then there are $u, v \in W_0$ such that

$$w_j = \begin{cases} u \cdot v & \text{or} \\ u / v\,. \end{cases}$$

[5] Remember that we are talking about the rank over K_0. And over K_0, pairwise distinct powers of x are linearly independent!

By definition of W_0, there exist $\alpha_1,\dots,\alpha_n \in K_0$, $b \in K$ and $\gamma_1,\dots,\gamma_n \in K_0$, $d \in K$ such that

$$u = \sum_{\nu=1}^{n} \alpha_\nu z_\nu + b\,,$$
$$v = \sum_{\nu=1}^{n} \gamma_\nu z_\nu + d\,.$$

Because $b \cdot d, b/d \in W_0$, there is a ν_1 such that $\alpha_{\nu_1} \neq 0$ or there is a ν_2 such that $\gamma_{\nu_2} \neq 0$. W. l. o. g. $\nu_1 = n$ or $\nu_2 = n$.

Now the idea is the following. We define a homomorphism $S : M' \to \bar{M}$ where M' is an appropriate subset of M and $\bar{M} = K[z_1,\dots,z_{n-1}]$ in such a way that

$$C(S(f_1),\dots,S(f_m)) \leq C(f_1,\dots,f_m) - 1\,.$$

Such an S is also called a *substitution* and the proof technique that we are using is called the *substitution method*. Then we apply the induction hypothesis to $S(f_1),\dots,S(f_m)$.

Case 1: $w_j = u \cdot v$. We can assume that $\gamma_n \neq 0$. Our substitution S is induced by

$$z_n \to \frac{1}{\gamma_n}\Big(\underbrace{\lambda}_{\in K_0} - \sum_{\nu=1}^{n-1} \gamma_\nu z_\nu - d\Big)\,,$$
$$z_\nu \to z_\nu \qquad \text{for } 1 \leq \nu \leq n-1\,.$$

The parameter λ will be chosen later. We have $S(z_n) \in W_0$, so there is a computation $(x_1,\dots,x_t)$ computing z_n at no cost. In the following, for an element $g \in K(z_1,\dots,z_n)$, we set $\bar{g} := S(g)$. We claim that the sequence

$$\bar{\beta} = (\underbrace{\bar{x}_1,\dots,\bar{x}_t}_{\text{compute } \bar{z}_n \text{ for free}}, \bar{w}_1,\dots,\bar{w}_\ell)$$

is a computation for $\bar{f}_1,\dots,\bar{f}_m$, since S is a homomorphism. There are two problems that have to be fixed: First z_n (an input) is replaced by something, namely $\bar{z}_n$, that is not an input. But we compute $\bar{z}_n$ in the beginning. Second, the substitution might cause a "division by zero," i. e., there might be an i such that $\bar{q}_i = 0$ and then $\bar{w}_i = \bar{p}_i/\bar{q}_i$ is not defined. But since q_i considered as an element of $K(z_1,\dots,z_{n-1})[z_n]$ can only have finitely many zeros, we can choose the parameter λ in such a way that none of the $\bar{q}_i$ is zero. (K_0 is infinite!)

By definition of S,

$$\bar{w}_j = \bar{u} \cdot \underbrace{\bar{v}}_{=\lambda}\,,$$

thus

$$\gamma(\bar{w}_j, \bar{V}_j) = 0\,.$$

This means that

$$\Gamma(\beta, Z) - 1 \geq \bar{\Gamma}(\bar{\beta}, \bar{Z})$$

and

$$\mathrm{C}^{*/}(F,Z) = \Gamma(\beta,Z) \geq \bar{\Gamma}(\bar{\beta},\bar{Z}) + 1 \underbrace{\geq}_{\substack{\text{induction}\\ \text{hypothesis}}} \text{col-rk}_{K_0} \bar{P} - m + 1\,.$$

It remains to estimate $\text{col-rk}_{K_0} \bar{P}$. We have

$$\bar{f}_\mu = \sum_{\nu=1}^{n-1} \bar{p}_{\mu\nu} z_\nu + \bar{q}_\mu\,,$$

$$\bar{p}_{\mu\nu} = p_{\mu\nu} - \frac{\gamma_\nu}{\gamma_n} p_{\mu n}\,,$$

$$\bar{q}_\mu = q_\mu - \frac{p_{\mu n}}{\gamma_n}(\lambda - d)\,.$$

Thus $\bar{P}$ is obtained from P by adding a K_0-multiple of the nth column to the other ones and then deleting the nth column. Therefore, $\text{col-rk}_{K_0} \bar{P} \geq r-1$ and $\mathrm{C}^{*/}(F,Z) \geq r-m$.

Case 2: $w_j = u/v$. If $\gamma_n \neq 0$, then $\bar{v} = \lambda \in K_0$ and the same substitution as in the first case works. If $\gamma_\nu = 0$ for all ν, then $v = d$ and $\alpha_n \neq 0$. Now we substitute

$$z_n \mapsto \frac{1}{\alpha_n}(\lambda d - \sum_{\nu=1}^{n-1} \alpha_\nu z_\nu - b)\,,$$

$$z_\nu \mapsto z_\nu \qquad \text{for } 1 \leq \nu \leq n-1\,.$$

Then $\bar{u} = \lambda d$ and $\bar{w}_j = \bar{u}/\bar{v} = \lambda \in K_0$. We can now proceed as in the first case. □

3.1.1 Further applications

Here are two other applications of Theorem 3.2.

Several polynomials

We can also look at the evaluation of several polynomials at one point x, i. e., at the complexity of

$$f_\mu(x) = \sum_{\nu=0}^{n_\mu} a_{\mu\nu} x^\nu\,, \qquad 1 \leq \mu \leq m\,.$$

Here the matrix P looks like

$$P = \left(\begin{array}{cccccccccccccc|cccc} x & x^2 & \dots & x^{n_1} & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & x & x^2 & \dots & x^{n_2} & \dots & 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \ddots & \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & x & x^2 & \dots & x^{n_m} & 0 & 0 & \dots & 1 \end{array}\right)$$

and we have $\text{col-rk}_{K_0} P = n_1 + n_2 + \cdots + n_m + m$. Thus

$$\mathrm{C}^{*/}(f_1,\dots,f_m) \geq n_1 + n_2 + \cdots + n_m\,,$$

that is, evaluating each polynomial using the Horner scheme is optimal. On the other hand, if we want to evaluate one polynomial at several points, this can be done much faster, see [8].

Matrix vector multiplication

Here, we consider the polynomials $f_1,\dots,f_m$ given by

$$\begin{pmatrix} a_{11} & \dots & a_{1k} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mk} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix} = \begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix}.$$

The matrix P is given by

$$P = \left(\begin{array}{ccccccccccccccc|cccc} x_1 & x_2 & \dots & x_k & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & x_1 & x_2 & \dots & x_k & \dots & 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \ddots & \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & x_1 & x_2 & \dots & x_k & 0 & 0 & \dots & 1 \end{array}\right).$$

Thus $\text{col-rk}_{K_0}(P) = km+m$ and

$$C^{*/}(f_1,\dots,f_m) \geq mk\,.$$

This means that here—opposed to general matrix multiplication—the trivial algorithm is optimal.

3.2 Additions

The second statement of Theorem 3.1 follows from the Theorem 3.3 below. We need the concept of *transcendence degree*. If we have two fields $K \subseteq L$, then the transcendence degree of L over K, $\text{tr-deg}_K(L)$ is the maximum number t of elements $a_1,\dots,a_t \in L$ such that $a_1,\dots,a_t$ do not fulfill any algebraic relation over K, that is, there is no t-variate polynomial p with coefficients from K such that $p(a_1,\dots,a_t) = 0$.[6]

Theorem 3.3. *Let K_0 be a field and $K = K_0(a_0,\dots,a_n)$. Let $f = a_0 + \dots + a_n x^n$. Then*

$$C^{+}(f) \geq \text{tr-deg}_{K_0}(a_0,a_1,\dots,a_n) - 1\,.$$

Proof. Let $\beta = (w_1,\dots,w_\ell)$ be a computation that computes f. W. l. o. g. $w_\lambda \neq 0$ for all $1 \leq \lambda \leq \ell$.

We want to characterize the set W_m of all elements that can be computed with m additions. We claim that there are polynomials $g_i(x,z_1,\dots,z_i)$ and elements $\zeta_i \in K$, $1 \leq i \leq m$ such that

$$W_0 = \{bx^{t_0} \mid t_0 \in \mathbb{Z}, b \in K\} \qquad \text{and}$$
$$W_m = \{bx^{t_0} f_1(x)^{t_1} \dots f_m(x)^{t_m} \mid t_i \in \mathbb{Z}, b \in K\}$$

where $f_i(x) = g_i(x,z_1,\dots,z_i)\,|_{z_1 \to \zeta_1,\dots,z_i \to \zeta_i}$, $1 \leq i \leq m$. The proof of this claim is by induction in m.
Induction start ($m = 0$): clear by construction.
Induction step ($m \to m+1$): Let $w_i = u \pm v$ be the last addition/subtraction in our computation with $m+1$ additions/subtractions. u,v can be computed with m additions/subtractions, therefore $u,v \in W_m$ by the induction hypothesis. This means that

$$w_i = bx^{t_0} f_1(x)^{t_1} \dots f_m(x)^{t_m} \pm cx^{s_0} f_1(x)^{s_1} \dots f_m(x)^{s_m}\,.$$

[6]Note the similarity to dimension of vector spaces. Here the dimension is the maximum number of elements that do not fulfill any *linear* relation.

W. l. o. g. $b \neq 0$, otherwise we would add 0. Therefore,

$$w_i = b(x^{t_0} g_1^{t_1} \cdots g_m^{t_m} \pm \frac{c}{b} \cdot x^{s_0} g_1^{s_1} \cdots g_m^{s_m}) \mid_{z_1 \to \zeta_1, \ldots, z_m \to \zeta_m} .$$

We set

$$g_{m+1} := (x^{t_0} g_1^{t_1} \cdots g_m^{t_m} \pm z_{m+1} x^{s_0} g_1^{s_1} \cdots g_m^{s_m}) .$$

Then

$$w_i = b g_{m+1} \mid_{z_1 \to \zeta_1, \ldots, z_{m+1} \to \zeta_{m+1}} \quad \text{with} \quad \zeta_{m+1} = \frac{c}{b}.$$

This shows the claim.

Since w_i was the last addition/subtraction in β for every $j > i$, w_j can be computed using only multiplications and is therefore in W_{m+1}. Since the g_i depend on $m+1$ variables $z_1, \ldots, z_{m+1}$, the polynomials f_i have transcendence degree at most $m+1$. Henceforth, the transcendence degree of the coefficients of f is at most $m+1$, since they are polynomials in the f_i. $\square$

Exercise 3.4. Show that the additive complexity of matrix-vector multiplication is $m(k-1)$ (multiplication of an $m \times k$-matrix with a vector of size k, see the specification in the previous section). Thus the trivial algorithm is optimal.

4 Bilinear problems

Let K be a field and let $M = K(x_1, \ldots, x_N)$. We will use the Ostrowski measure in the following. We will ask questions of the form

$$\mathrm{C}^{*/}(F) = ?$$

where $F = \{f_1, \ldots, f_k\}$ is a set of *quadratic forms*,

$$f_\kappa = \sum_{\mu,\nu=1}^{N} t_{\kappa\mu\nu} x_\mu x_\nu , \qquad 1 \leq \kappa \leq k .$$

Most of the time, we will consider the special case of bilinear forms, that is, our variables are divided into two disjoint sets and only products of one variable from the first set with one variable of the second set appear in f_κ.

The "three dimensional array" $t := (t_{\kappa\mu\nu})_{\kappa=1,\ldots,k;\mu,\nu=1,\ldots,N} \in K^{k \times N \times N}$ is called the *tensor corresponding to* F. Since $x_\mu x_\nu = x_\nu x_\mu$, there are several tensors that represent the same set F. A tensor s is *symmetrically equivalent* to t if

$$s_{\kappa\mu\nu} + s_{\kappa\nu\mu} = t_{\kappa\mu\nu} + t_{\kappa\nu\mu} \qquad \text{for all } \kappa,\, \mu,\, \nu.$$

Two tensors describe the same set of quadratic forms if they are symmetrically equivalent.

The two typical problems that we will deal with in the following are:

	a_0	a_1	a_2	a_3
b_0	1	2	3	4
b_1	2	3	4	5
b_2	3	4	5	6
b_3	4	5	6	7

Figure 1: The tensor of the multiplication of polynomials of degree three. The rows correspond to the coefficients of the first polynomial, the columns to the coefficients of the second. The tensors consist of 7 layers. The entries of the tensor are from $\{0,1\}$. The entry ℓ in position (i,j) means that $t_{i,j,\ell}=1$, i. e., $a_i \cdot b_j$ occurs in c_ℓ.

	$x_{1,1}$	$x_{1,2}$	$x_{2,1}$	$x_{2,2}$
$y_{1,1}$	$(1,1)$		$(2,1)$	
$y_{2,1}$		$(1,1)$		$(2,1)$
$y_{1,2}$	$(1,2)$		$(2,2)$	
$y_{2,2}$		$(1,2)$		$(2,2)$

Figure 2: The tensor of 2×2-matrix multiplication. Again, it is $\{0,1\}$-valued. An entry (κ,ν) in the row (κ,μ) and column (μ,ν) means that $x_{\kappa,\mu}y_{\mu,\nu}$ appears in $f_{\kappa,\nu}$.

Matrix multiplication: We are given two $n\times n$-matrices $x=(x_{ij})$ and $y=(y_{ij})$ with indeterminates as entries. The entries of xy are given by the well-known quadratic (in fact bilinear) forms

$$f_{ij}=\sum_{k=1}^{n} x_{ik}y_{kj}\,, \qquad 1\le i,j\le n\,.$$

Polynomial multiplication: Here our input consists of two polynomials $p(z)=\sum_{i=0}^{m} a_i z^i$ and $q(z)=\sum_{j=0}^{n} b_j z^j$. The coefficients are again indeterminates over K. The coefficients c_ℓ, $0\le \ell\le m+n$ of their product pq are given by the bilinear forms

$$c_\ell=\sum_{i+j=\ell} a_i b_j\,, \qquad 0\le \ell\le m+n\,.$$

Figure 1 shows the tensor of multiplication of degree 3 polynomials. It is shown as an element of $K^{4\times 4\times 7}$. Strictly speaking, it would be an element of $K^{8\times 8\times 7}$. But since polynomial multiplication is a bilinear map, the rest of the entries are zero. We will look at tensors of bilinear maps in this way in the following. Figure 2 shows the tensor of 2×2-matrix multiplication. It lives in $K^{4\times 4\times 4}$.

4.1 Vermeidung von Divisionen

Strassen [32] showed that for computing sets of quadratic forms, divisions do not help (provided that the field of scalars is large enough). For a polynomial $g\in K[x_1,\dots,x_N]$, $H_j(g)$ denotes the *homogenous part* of degree j of g, that is, the sum of all monomials of degree j of g.

Theorem 4.1. *Let* $F_\kappa = \sum_{\mu,\nu=1}^{N} t_{\kappa\mu\nu} x_\mu x_\nu$, $1 \le \kappa \le k$, *and* $F = \{F_1, \dots, F_k\}$. *If* $\#K = \infty$ *and* $\mathrm{C}^{*/}(F) \le \ell$ *then there are products*

$$P_\lambda = \Big(\sum_{i=1}^{N} u_{\lambda i} x_i\Big)\Big(\sum_{i=1}^{N} v_{\lambda i} x_i\Big), \qquad 1 \le \lambda \le \ell,$$

such that $F \subseteq \mathrm{lin}_K\{P_1, \dots, P_\ell\}$. *In particular,* $\mathrm{C}^*(F) = \mathrm{C}^{*/}(F)$.

Note that each factor of the products is a linear form in the variables that is free of cost. We can write each F_κ as a linear combination of the products, again at no cost.

Proof. Let $\beta = (w_1, \dots, w_L)$ be an optimal computation for F, w. l. o. g. $0 \notin F$ and $w_i \ne 0$ for all $1 \le i \le L$. Let $w_i = g_i / h_i$ with $g_i, h_i \in K[x_1, \dots, x_N]$, $h_i, g_i \ne 0$.

As a first step, we want to achieve that

$$H_0(g_i) \ne 0 \ne H_0(h_i), \qquad 1 \le i \le L.$$

We substitute

$$x_i \to \bar{x}_i - \alpha_i, \qquad 1 \le i \le N,$$

for some $\alpha_i \in K$. Let the resulting computation be $\bar{\beta} = (\bar{w}_1, \dots, \bar{w}_L)$ where $\bar{w}_i = \bar{g}_i / \bar{h}_i$,

$$\bar{g}_i(\bar{x}_1, \dots, \bar{x}_N) = g_i(x_1 + \alpha_1, \dots, x_N + \alpha_N) \qquad \text{and} \qquad \bar{h}_i(\bar{x}_1, \dots, \bar{x}_N) = h_i(x_1 + \alpha_1, \dots, x_N + \alpha_N).$$

Since $f_\kappa \in \{w_1, \dots, w_L\}$,

$$\bar{f}_\kappa(\bar{x}_1, \dots, \bar{x}_N) = f_\kappa(\bar{x}_1 + \alpha_1, \dots, \bar{x}_N + \alpha_N) \in \{\bar{w}_1, \dots, \bar{w}_L\}.$$

Because

$$\bar{f}_\kappa(\bar{x}_1, \dots, \bar{x}_N) = \sum_{\mu,\nu=1}^{N} t_{\kappa\mu\nu} \bar{x}_\mu \bar{x}_\nu = \sum_{\mu,\nu=1}^{N} t_{\kappa\mu\nu} x_\mu x_\nu + \text{ terms of degree} \le 1,$$

we can extend the computation $\bar{\beta}$ without increasing the cost such that the new computation computes $f_\kappa(x_1, \dots, x_N)$, $1 \le \kappa \le k$. All we have to do is to compute the terms of degree one, which is free of cost, and subtract them from the $\bar{f}_\kappa(\bar{x}_1, \dots, \bar{x}_N)$, which is again free of cost. We call the resulting computation again $\bar{\beta}$.

By the following well-known fact, we can choose the α_i in such a way that all $H_0(\bar{g}_i) \ne 0 \ne H_0(\bar{h}_i)$, since $H_0(\bar{g}_i) = g_i(\alpha_1, \dots, \alpha_N)$ and $H_0(\bar{h}_i) = h_i(\alpha_1, \dots, \alpha_N)$.

Fact 4.2. *For any finite set of polynomials* $\phi_1, \dots, \phi_n$, $\phi_i \ne 0$ *for all i, there are* $\alpha_1, \dots, \alpha_N \in K$ *such that* $\phi_i(\alpha_1, \dots, \alpha_N) \ne 0$ *for all i provided that* $\#K = \infty$.[7]

[7]Hint:

if type = mathematician **then**
    return "It's an open set!"
else if type = theoretical computer scientist **then**
    use the Schwartz-Zippel lemma
else
    prove it by induction on *n*
end if

Next, we substitute

$$\bar{x}_i \to x_i z, \qquad 1 \le i \le N.$$

Let $\tilde{\beta} = (\tilde{w}_1, \dots, \tilde{w}_{\tilde{L}})$ be the resulting computation. We view the $\tilde{w}_i$ as elements of $K(x_1, \dots, x_N)[[z]]$, that is, as formal power series in z with rational functions in $x_1, \dots, x_N$ as coefficients. This is possible, since every $\bar{w}_i = \bar{g}_i / \bar{h}_i$. The substitution above transforms $\bar{g}_i$ and $\bar{h}_i$ into the power series

$$\tilde{g}_i = H_0(\bar{g}_i) + H_1(\bar{g}_i)z + H_2(\bar{g}_i)z^2 + \cdots \quad \text{and}$$
$$\tilde{h}_i = H_0(\bar{h}_i) + H_1(\bar{h}_i)z + H_2(\bar{h}_i)z^2 + \cdots .$$

By the fact below, $\tilde{h}_i$ has in inverse in $K(x_1, \dots, x_N)[[z]]$ because $H_0(\bar{h}_i) \neq 0$. Thus $\tilde{w}_i = \tilde{g}_i / \tilde{h}_i$ is an element of $K(x_1, \dots, x_N)[[z]]$ and we can write it as

$$\tilde{w}_i = c_i + c_i^{'} z + c_i^{''} z^2 + \cdots .$$

Fact 4.3. *A formal power series $\sum_{i=0}^{\infty} a_i z^i \in L[[z]]$ is invertible iff $a_0 \neq 0$. Its inverse is given by*

$$\frac{1}{a_0}(1 + q + q^2 + \cdots)$$

where $q = -\sum_{i=1}^{\infty} \frac{a_i}{a_0} z^i$.[8]

Since in the end, we compute a set of quadratic forms, it is sufficient to compute only $\tilde{w}_i$ up to degree two in z. Because c_i and c_i' can be computed for free in the Ostrowski model, we only need to compute c_i'' in every step.

First case: ith step is a multiplication. We have

$$\tilde{w}_i = \tilde{u} \cdot \tilde{v} = (u + u'z + u''z^2 + \dots)(v + v'z + v''z^2 + \dots).$$

We can compute

$$c_i'' = \underbrace{\underbrace{u}_{\in K} v''}_{\text{free of cost}} + u'v' + \underbrace{u'' \underbrace{v}_{\in K}}_{\text{free of costs}} .$$

with one bilinear multiplication.

Second case: ith step is a division. Here,

$$\begin{aligned} \tilde{w}_i &= \frac{\tilde{u}}{\tilde{v}} \\ &= \frac{u + u^{'}z + u^{''}z + \dots}{1 + v^{'}z + v^{''}z^2 + \dots} \\ &= (u + u^{'}z + u^{''}z^2 + \dots)(1 - (v^{'}z + v^{''}z^2 + \dots) + (v^{'}z + \dots)^2 - (v^{'}z + \dots)^3 + \dots). \end{aligned}$$

Thus

$$c_i'' = u^{''} - u^{'}v^{'} - u(-v^{''} + (v^{'})^2) = u^{''} - (u^{'} - \underbrace{uv^{'}}_{\text{free of costs}})v^{'} + \underbrace{uv^{''}}_{\text{free of costs}}$$

can be computed with one costing operation. □

[8] Hint: $\frac{1}{1-q} = \sum_{i=0}^{\infty} q^i$.

4.2 Rank of bilinear problems

Polynomial multiplication and matrix multiplication are bilinear problems. We can separate the variables into two sets $\{x_1,\dots,x_M\}$ and $\{y_1,\dots,y_N\}$ and write the quadratic forms as

$$f_\kappa = \sum_{\mu=1}^{M}\sum_{\nu=1}^{N} t_{\kappa\mu\nu} x_\mu y_\nu\,, \qquad 1 \le \kappa \le k\,.$$

The tensor $(t_{\kappa\mu\nu}) \in K^{k\times M\times N}$ is unique once we fix a ordering of the variables and quadratic forms and we do not need the notion of symmetric equivalence.

Theorem 4.1 tell us that under the Ostrowski measure, we only have to consider products of linear forms. When computing bilinear forms, it is a natural to restrict ourselves to products of the form linear form in $\{x_1,\dots,x_M\}$ times a linear form in $\{y_1,\dots,y_N\}$.

Definition 4.4. The minimal number of products

$$P_\lambda = \Big(\sum_{\mu=1}^{M} u_{\lambda\mu} x_\mu\Big)\Big(\sum_{\nu=1}^{N} v_{\lambda\nu} y_\nu\Big)\,, \qquad 1 \le \lambda \le \ell$$

such that $F \subseteq \operatorname{lin}\{P_1,\dots,P_l\}$ is called *rank* of $F = \{F_1,\dots,F_k\}$ or *bilinear complexity* of F. We denote it by $R(F)$.

We can define the rank in terms of tensors, too. Let $t = (t_{\kappa\mu\nu})$ be the tensor of F as above. We have

$$\begin{aligned} R(F) \le \ell \Leftrightarrow{} & \text{there are linear forms } u_1,\dots,u_\ell \text{ in } x_1,\dots,x_M \\ & \text{and } v_1,\dots,v_\ell \text{ in } y_1,\dots,y_N \text{ such that } F \subseteq \operatorname{lin}\{u_1v_1,\dots,u_\ell v_\ell\} \\ \Leftrightarrow{} & \text{there are } w_{\lambda\kappa} \in K,\ 1 \le \lambda \le \ell,\ 1 \le \kappa \le k, \\ & \text{such that } f_\kappa = \sum_{\lambda=1}^{l} w_{\lambda\kappa} u_\lambda v_\lambda = \sum_{\lambda=1}^{\ell} w_{\lambda\kappa}\Big(\sum_{\mu=1}^{M} u_{\lambda\mu} x_\mu\Big)\Big(\sum_{\nu=1}^{N} v_{\lambda\nu} y_\nu\Big),\ 1 \le \kappa \le k\,. \end{aligned}$$

Comparing coefficients, we get

$$t_{\kappa\mu\nu} = \sum_{\lambda=1}^{l} w_{\lambda\kappa} u_{\lambda\mu} v_{\lambda\nu}\,, \qquad 1 \le \kappa \le k,\ 1 \le \mu \le M,\ 1 \le \nu \le N\,.$$

Definition 4.5. Let $w \in K^k$, $u \in K^M$, $v \in K^N$. The tensor $w \otimes u \otimes v \in K^{k\times M\times N}$ with entry $w_\kappa u_\mu v_\nu$ in position (κ,μ,ν) is called a *triad*.

From the calculation above, we get

$$\begin{aligned} R(F) \le \ell \Leftrightarrow{} & \text{there are } w_1,\dots w_\ell \in K^k,\ u_1 \dots u_\ell \in K^M, \text{ and } v_1 \dots v_\ell \in K^N \text{ such that} \\ & t = (t_{\kappa\mu\nu}) = \sum_{\lambda=1}^{\ell} \underbrace{w_\lambda \otimes u_\lambda \otimes v_\lambda}_{\text{triad}}\,. \end{aligned}$$

We define the rank $R(t)$ of a tensor t to be the minimal number of triads such that t is the sum of these triads.[9] To every set of bilinear forms F there is a corresponding tensor t and vice versa. As we have seen, their rank is the same.

Example 4.6 (Complex multiplication)**.** Consider the multiplication of complex number viewed as an $\mathbb{R}$-algebra. Its multiplication is described by the two bilinear forms f_0 and f_1 defined by

$$(x_0 + x_1 i)(y_0 + y_1 i) = \underbrace{x_0 y_0 - x_1 y_1}_{f_0} + \underbrace{(x_0 y_1 + x_1 y_0)}_{f_1} i \,.$$

It is clear that $R(f_0, f_1) \leq 4$. But also $R(f_0, f_1) \leq 3$ holds. Let

$$\begin{aligned} P_1 &= x_0 y_0 \,, \\ P_2 &= x_1 y_1 \,, \\ P_3 &= (x_0 + x_1)(y_0 + y_1) \,. \end{aligned}$$

Then

$$\begin{aligned} f_0 &= P_1 - P_2 \,, \\ f_1 &= P_3 - P_1 - P_2 \,. \end{aligned}$$

This is essentially Karatsuba's algorithm. Note that $\mathbb{C} \cong K[X]/(X^2+1)$. We first multiply the two polynomials $x_0 + x_1 X$ and $y_0 + y_1 X$ and then reduce modulo $X^2 - 1$, which is free of cost in the bilinear model.

Multiplicative complexity and rank are linearly related.

Theorem 4.7. *Let $F = \{f_1, \ldots, f_k\}$ be a set of bilinear forms in variables $\{x_1, \ldots, x_M\}$ and $\{y_1, \ldots, y_N\}$. Then*

$$\mathrm{C}^{*/}(F) \leq R(F) \leq 2\,\mathrm{C}^{*/}(F) \,.$$

Proof. The first inequality is clear. For the second, assume that $\mathrm{C}^{*/}(F) = \ell$ and consider an optimal computation. We have

$$\begin{aligned} f_\kappa &= \sum_{\lambda=1}^{\ell} w_{\lambda\kappa} \Big(\sum_{\mu=1}^{M} u_{\lambda\mu} x_\mu + \sum_{\nu=1}^{N} u'_{\lambda\nu} y_\nu \Big) \Big(\sum_{\mu=1}^{M} v'_{\lambda\mu} x_\mu + \sum_{\nu=1}^{N} v_{\lambda\nu} y_\nu \Big) \\ &= \sum_{\lambda=1}^{\ell} w_{\lambda\kappa} \Big(\sum_{\mu=1}^{M} u_{\lambda\mu} x_\mu \Big) \Big(\sum_{\nu=1}^{N} v_{\lambda\nu} y_\nu \Big) + \sum_{\lambda=1}^{\ell} w_{\lambda\kappa} \Big(\sum_{\mu=1}^{M} v'_{\lambda\mu} x_\mu \Big) \Big(\sum_{\nu=1}^{N} u'_{\lambda\nu} y_\nu \Big) \,. \end{aligned}$$

The terms of the form $x_i x_j$ and $y_i y_j$ have to cancel each other, since they do not appear in f_κ. □

[9] Note the similarity to the definition of rank of a matrix. The rank of a matrix M is the minimum number of rank-1 matrices ("diads") such such that M is the sum of these rank-1 matrices.

Example 4.8 (Winograd's algorithm [36])**.** Do products that are not bilinear help in for the computation of bilinear forms? Here is an example. We consider the multiplication of $M \times 2$ matrices with $2 \times N$ matrices. Then entries of the product are given by

$$f_{\mu\nu} = x_{\mu 1} y_{1\nu} + x_{\mu 2} y_{2\nu} .$$

Consider the following MN products

$$(x_{\mu 1} + y_{2\nu})(x_{\mu 2} + y_{1\nu}) \qquad 1 \le \mu \le M, \ \ 1 \le \nu \le N .$$

We can write

$$f_{\mu\nu} = (x_{\mu 1} + y_{2\nu})(x_{\mu 2} + y_{1\nu}) - x_{\mu 1} x_{\mu 2} - y_{1\nu} y_{2\nu} ,$$

thus a total of $MN + M + N$ products suffice. Setting $M = 2$, we can multiply 2×2 matrices with $2 \times n$ matrices with $3N + 2$ multiplications. For the rank, the best we know is $\lceil 3\frac{1}{2} N \rceil$ multiplications, which we get by repeatedly applying Strassen's algorithm and possibly one matrix-vector multiplication if N is odd.

Waksman [34] showed that if $\operatorname{char} K \neq 2$, then even $MN + M + N - 1$ products suffice. We get that the multiplicative complexity of 2×2 with 2×3 matrix multiplication is ≤ 10. On the other hand, Alekseyev [1] proved that the rank is 11.

5 The exponent of matrix multiplication

In the following $\langle k, m, n \rangle : K^{k \times m} \times K^{m \times n} \to K^{k \times n}$ denotes the bilinear map that maps a $k \times m$-matrix A and an $m \times n$-matrix B to their product AB. Since there is no danger of confusion, we will also use the same symbol for the corresponding tensor and for the set of bilinear forms

$$\left\{ \sum_{\mu=1}^{m} X_{\kappa\mu} Y_{\mu\nu} \;\middle|\; 1 \le \kappa \le k, \ \ 1 \le \nu \le n \right\} .$$

Definition 5.1. $\omega = \inf\{\beta \mid R(\langle n, n, n \rangle) \le O(n^\beta)\}$ is called the *exponent of matrix multiplication*.

In the definition of ω above, we only count bilinear products. For the asymptotic growth, it does not matter whether we count all operations or only bilinear products. Let $\tilde{\omega} = \inf\{\beta \mid C(\langle n, n, n \rangle) \le O(n^\beta)\}$ with $¢(\pm) = ¢(*/) = ¢(\lambda\cdot) = 1$.

Theorem 5.2. *$\omega = \tilde{\omega}$, if K is infinite.*

Proof. $\omega \le \tilde{\omega}$ is obvious. For the other inequality, note that from the definition of ω, it follows that there is an α such that

$$\forall \varepsilon > 0 : \exists m_0 > 1 : \forall m \ge m_0 : R(\langle m, m, m \rangle) \le \alpha \cdot m^{w+\varepsilon} .$$

Let $\varepsilon > 0$ be given and choose such an m that is large enough. Let $r = R(\langle m, m, m \rangle)$.

To multiply $m^i \times m^i$-matrices we decompose them into blocks of $m^{i-1} \times m^{i-1}$-matrices and apply recursion. Let $A(i)$ be the number of arithmetic operations for the multiplication of $m^i \times m^i$-matrices with this approach. We obtain

$$A(i) \le r A(i-1) + c\, m^{2(i-1)}$$

where c is the number of additions and scalar multiplications that are performed by the chosen bilinear algorithm for $\langle m,m,m\rangle$ with r bilinear multiplications. Expanding this, we get

$$\begin{aligned}
A(i) &\leq r^i A(0) + cm^{2(i-1)} \left(\sum_{j=0}^{i-2} \frac{r^j}{m^{2j}} \right) \\
&= r^i A(0) + c\, m^{2(i-1)} \frac{\left(\frac{r}{m^2}\right)^{i-1} - 1}{\frac{r}{m^2} - 1} \\
&= r^i A(0) + c\, m^2\ \frac{r^{i-1} - m^{2(i-1)}}{r - m^2} \\
&= \underbrace{\left(A(0) + \frac{c\, m^2}{r(r-m^2)} \right)}_{\text{constant}} r^i - \frac{c}{r-m^2}\, m^2 \,.
\end{aligned}$$

(Obviously, $r \geq m^2$. But it is also very easy to show that $r > m^2$, so we are not dividing by zero.) We have $C(\langle n',n',n'\rangle) \leq C(\langle n,n,n\rangle)$ if $n' \leq n$. (Recall that we can eliminate divisions, so we can fill up with zeros.) Therefore,

$$\begin{aligned}
C(\langle n,n,n\rangle) &\leq C(\left\langle m^{\lceil \log_m n\rceil}, m^{\lceil \log_m n\rceil}, m^{\lceil \log_m n\rceil} \right\rangle) \\
&\leq A(\lceil \log_m n \rceil) \\
&= O(r^{\lceil \log_m n\rceil}) \\
&= O(r^{\log_m n}) \\
&= O(n^{\log_m r})\,.
\end{aligned}$$

Since $r \leq \alpha \cdot m^{\omega+\varepsilon}$, we have $\log_m r \leq \omega + \varepsilon + \log_m \alpha$. With $\varepsilon' = \varepsilon + \log_m \alpha$,

$$C(\langle n,n,n\rangle) = O(n^{\log_m r}) = O(n^{\omega+\varepsilon'})\,.$$

Thus

$$\tilde{\omega} \leq \omega + \varepsilon \qquad \text{for all } \varepsilon > 0\,,$$

since $\log_m \alpha \to 0$ if $m \to \infty$. This means $\tilde{\omega} = \omega$, since $\tilde{\omega}$ is an infimum. □

To prove good upper bounds for ω, we introduce some operation on tensors and analyze the behavior of the rank under these operations.

5.1 Permutations (of tensors)

Let $t \in K^{k\times m\times n}$ and $t = \sum_{j=1}^{r} t_j$ with triads $t_j = a_{j1} \otimes a_{j2} \otimes a_{j3}$, $1 \leq j \leq r$. Let $\pi \in S_3$, where S_3 denotes the symmetric group on $\{1,2,3\}$. For a triad t_j, let $\pi t_j = a_{j\pi^{-1}(1)} \otimes a_{j\pi^{-1}(2)} \otimes a_{j\pi^{-1}(3)}$ and $\pi t = \sum_{j=1}^{r} \pi t_j$.

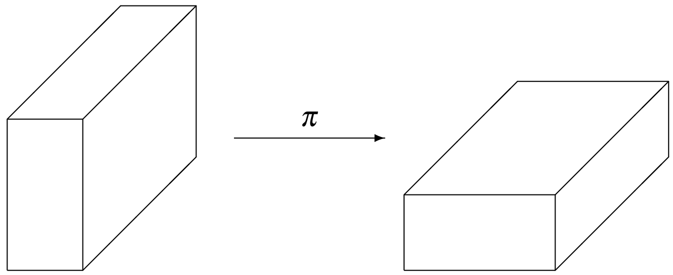

Figure 3: Permutation of the dimensions.

πt is well-defined. To see this, let $t = \sum_{i=1}^{s} b_{i1} \otimes b_{i2} \otimes b_{i3}$ be a second decomposition of t. We claim that

$$\sum_{j=1}^{r} a_{j\pi^{-1}(1)} \otimes a_{j\pi^{-1}(2)} \otimes a_{j\pi^{-1}(3)} = \sum_{i=1}^{s} b_{i\pi^{-1}(1)} \otimes b_{i\pi^{-1}(2)} \otimes b_{i\pi^{-1}(3)} .$$

Let $a_{j1} = (a_{j11}, \dots, a_{j1k})$ and $b_{i1} = (b_{i11}, \dots, b_{i1k})$ and let a_{j2}, a_{j3}, b_{i2}, and b_{i3} be given analogously.

We have

$$t_{e_1 e_2 e_3} = \sum_{j=1}^{r} a_{j1e_1} \cdot a_{j2e_2} \cdot a_{j3e_3} = \sum_{i=1}^{s} b_{i1e_1} \cdot b_{i2e_2} \cdot b_{i3e_3} .$$

Thus

$$\begin{aligned} \pi t_{e_1 e_2 e_3} &= \sum_{j=1}^{r} a_{j\pi^{-1}(1)e_{\pi^{-1}(1)}} \cdot a_{j\pi^{-1}(2)e_{\pi^{-1}(2)}} \cdot a_{j\pi^{-1}(3)e_{\pi^{-1}(3)}} \\ &= \sum_{i=1}^{s} b_{i\pi^{-1}(1)e_{\pi^{-1}(1)}} \cdot b_{i\pi^{-1}(2)e_{\pi^{-1}(2)}} \cdot b_{i\pi^{-1}(3)e_{\pi^{-1}(3)}} . \end{aligned}$$

The proof of the following lemma is obvious.

Lemma 5.3. $R(t) = R(\pi t)$.

Instead of permuting the dimensions, we can also permute the slices of a tensor. Let $t = (t_{ij\ell}) \in K^{k \times m \times n}$ and $\sigma \in S_k$. Then, for $t' = (t_{\sigma(i)j\ell})$, $R(t') = R(t)$.

More generally, let $A : K^k \to K^{k'}$, $B : K^m \to K^{m'}$, and $C : K^n \to K^{n'}$ be homomorphisms. Let $t = \sum_{j=1}^{r} t_j$ with triads $t_j = a_{j1} \otimes a_{j2} \otimes a_{j3}$. We set

$$(A \otimes B \otimes C)t_j = A(a_{j1}) \otimes B(a_{j2}) \otimes C(a_{j3})$$

and

$$(A \otimes B \otimes C)t = \sum_{j=1}^{r} (A \otimes B \otimes C)t_j .$$

By looking at a particular entry of t, it is easy to see that this is well-defined.

The proof of the following lemma is again obvious.

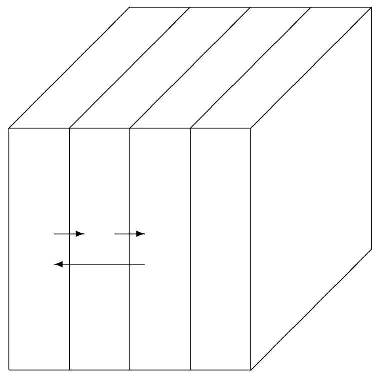

Figure 4: Permutation of the slices.

Lemma 5.4. $R((A\otimes B\otimes C)t)\leq R(t)$.

Equality holds if A, B, and C are isomorphisms. How does the tensor of matrix multiplication look like? Recall that the bilinear forms are given by

$$Z_{\kappa\nu}=\sum_{\mu=1}^{m}X_{\kappa\mu}Y_{\mu\nu}\,,\quad 1\leq\kappa\leq k\,,\;\;1\leq\nu\leq n\,.$$

The entries of the corresponding tensor

$$(t_{\kappa\bar{\mu},\mu\bar{\nu},\nu\bar{\kappa}})=t\in K^{(k\times m)\times(m\times n)\times(n\times k)}$$

are given by

$$t_{\kappa\bar{\mu},\mu\bar{\nu},\nu\bar{\kappa}}=\delta_{\bar{\kappa}\kappa}\delta_{\bar{\mu}\mu}\delta_{\bar{\nu}\nu}$$

where δ_{ij} is Kronecker's delta. (Here, each dimension of the tensor is addressed with a two-dimensional index, which reflects the way we number the entries of matrices. If you prefer it, you can label the entries of the tensor with indices from $1,\dots km$, $1,\dots mn$, and $1,\dots,nk$. We also "transposed" the indices in the third slice, to get a symmetric view of the tensor.)

Let $\pi=(123)$. Then for $\pi t=:t'\in K^{(n\times k)\times(k\times m)\times(m\times n)}$, we have

$$\begin{aligned}t'_{\nu\bar{\kappa},\kappa\bar{\mu},\mu\bar{\nu}}&=\delta_{\bar{\nu}\nu}\delta_{\bar{\kappa}\kappa}\delta_{\bar{\mu}\mu}\\&=\delta_{\bar{\kappa}\kappa}\delta_{\bar{\mu}\mu}\delta_{\bar{\nu}\nu}\\&=t_{\kappa\bar{\mu},\mu\bar{\nu},\nu\bar{\kappa}}\,.\end{aligned}$$

Therefore,

$$R(\langle k,m,n\rangle)=R(\langle n,k,m\rangle)=R(\langle m,n,k\rangle)\,.$$

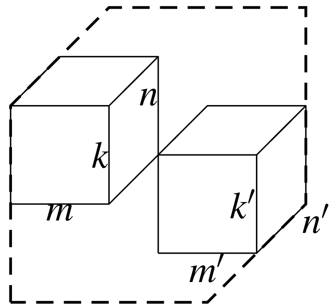

Figure 5: Sum of two tensors.

Now, let $t'' = (t_{\mu\bar{\kappa},\nu\bar{\mu},\bar{\kappa}\nu})$. We have $R(t) = R(t'')$, since permuting the "inner" indices corresponds to permuting the slices of the tensor.

Next, let $\pi = (12)(3)$. Let $\pi t'' =: t''' \in K^{(n\times m)\times(m\times k)\times(k\times n)}$. We have,

$$\begin{aligned} t'''_{\nu\bar{\mu},\mu\bar{\kappa},\kappa\bar{\nu}} &= \delta_{\mu,\bar{\mu}}\delta_{\kappa,\bar{\kappa}}\delta_{\nu,\bar{\nu}} \\ &= t_{\bar{\kappa}\mu,\bar{\mu}\nu,\bar{\nu}\kappa}\,. \end{aligned}$$

Therefore,

$$R(\langle k,m,n\rangle) = R(\langle n,m,k\rangle)\,.$$

The second transformation corresponds to the well-known fact that $AB = C$ implies $B^T A^T = C^T$.

To summarize:

Lemma 5.5. $R(\langle k,m,n\rangle) = R(\langle n,k,m\rangle) = R(\langle m,n,k\rangle) = R(\langle m,k,n\rangle) = R(\langle n,m,k\rangle) = R(\langle k,n,m\rangle)$.

5.2 Products and sums

Let $t \in K^{k\times m\times n}$ and $t' \in K^{k'\times m'\times n'}$. The *direct sum* of t and t', $s := t \oplus t' \in K^{(k+k')\times(m+m')\times(n+n')}$, is defined as follows:

$$s_{\kappa\mu\nu} = \begin{cases} t_{\kappa\mu\nu} & \text{if } 1 \le \kappa \le k,\ 1 \le \mu \le m,\ 1 \le \nu \le n\,, \\ t'_{\kappa-k,\mu-m,\nu-n} & \text{if } k+1 \le \kappa \le k+k',\ m+1 \le \mu \le m+m',\ n+1 \le \nu \le n+n'\,, \\ 0 & \text{otherwise.} \end{cases}$$

Lemma 5.6. $R(t \oplus t') \le R(t) + R(t')$.

Proof. Let $t = \sum_{i=1}^{r} u_i \otimes v_i \otimes w_i$ and $t' = \sum_{i=1}^{r} u'_i \otimes v'_i \otimes w'_i$. Let

$$\begin{aligned} \hat{u}_i &= (\underbrace{u_{i1},\cdots,u_{i_k}}_{u_i},\underbrace{0,\cdots,0}_{k'}) \quad \text{and} \\ \hat{u}'_i &= (\underbrace{0,\cdots,0}_{k},\underbrace{u'_{i1},\cdots,u'_{ik}}_{u'_i}) \end{aligned}$$

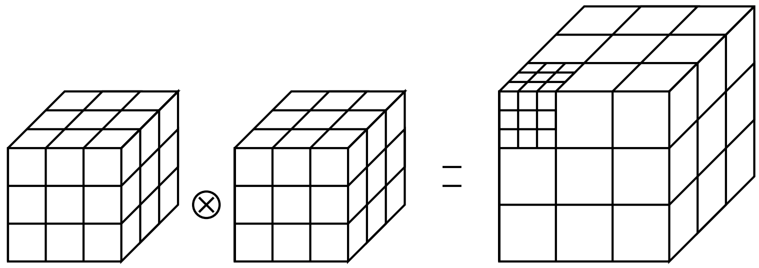

Figure 6: Product of two tensors.

and define $\hat{v}_i$, $\hat{w}_i$ and $\hat{v}'_i$, $\hat{w}'_i$ analogously. An easy calculation shows that

$$t \oplus t' = \sum_{i=1}^{r} \hat{u}_i \otimes \hat{v}_i \otimes \hat{w}_i + \sum_{j=1}^{r'} \hat{u}'_i \otimes \hat{v}'_i \otimes \hat{w}'_i,$$

which proves the lemma. □

Research problem 5.7 (Strassen's additivity conjecture)**.** Show that for all tensors t and t', $R(t \oplus t') = R(t) + R(t')$, that is, equality always holds in the lemma above.

The *tensor product* $t \otimes t' \in K^{kk' \times mm' \times nn'}$ of two tensors $t \in K^{k \times m \times n}$ and $t' \in K^{k' \times m' \times n'}$ is defined by

$$t \otimes t' = \left(t_{\kappa\mu\nu}\ t'_{\kappa'\mu'\nu'} \right)_{\substack{1 \le \kappa \le k, 1 \le \kappa' \le k' \\ 1 \le \mu \le m, 1 \le \mu' \le m' \\ 1 \le \nu \le n, 1 \le \nu' \le n'}} .$$

It is very convenient to use double indices κ, κ' to "address" the slices $1, \ldots, kk'$ of the tensor product. The same is true for the other two dimensions.

Lemma 5.8. $R(t \otimes t') \le R(t)R(t')$.

Proof. Let $t = \sum_{i=1}^{r} u_i \otimes v_i \otimes w_i$ and $t' = \sum_{i=1}^{r'} u'_i \otimes v'_i \otimes w'_i$. Let

$$u_i \otimes u'_j := (u_{i\kappa} u'_{j\kappa'})_{1 \le \kappa \le k, 1 \le \kappa' \le k'} \in K^{kk'} .$$

In the same way we define $v_i \otimes v'_j$, $w_i \otimes w'_j$. We have

$$\begin{aligned} (u_i \otimes u'_j) \otimes (v_i \otimes v'_j) \otimes (w_i \otimes w'_j) &= (u_{i\kappa} u'_{j\kappa'} \cdot v_{i\mu} v'_{j\mu'} \cdot w_{i\nu} w'_{j\nu'})_{\substack{1 \le \kappa \le k, 1 \le \kappa' \le k' \\ 1 \le \mu \le m, 1 \le \mu' \le m' \\ 1 \le \nu \le n, 1 \le \nu' \le n'}} \\ &\in K^{kk' \times mm' \times nn'} \cong K^{(k \times k') \times (m \times m') \times (n \times n')} \end{aligned}$$

and

$$\sum_{i=1}^{r}\sum_{j=1}^{r'}(u_i\otimes u'_j)\otimes(v_i\otimes v'_j)\otimes(w_i\otimes w'_j) = \Big(\sum_{i=1}^{r}\sum_{j=1}^{r'} u_{i\kappa}u'_{j\kappa'}v_{i\mu}v'_{j\mu'}w_{i\nu}w'_{i\nu'}\Big)_{\substack{1\le\kappa\le k,1\le\kappa'\le k'\\ 1\le\mu\le m,1\le\mu'\le m'\\ 1\le\nu\le n,1\le\nu'\le n'}}$$

$$= \Bigg(\underbrace{\Big(\sum_{i=1}^{r} u_{i\kappa}v_{i\mu}w_{i\nu}\Big)}_{t_{\kappa\mu\nu}}\cdot\underbrace{\Big(\sum_{j=1}^{r'} u'_{j\kappa}v'_{j\mu}w'_{j\nu'}\Big)}_{t'_{\kappa'\mu'\nu'}}\Bigg)_{\substack{1\le\kappa\le k,1\le\kappa'\le k'\\ 1\le\mu\le m,1\le\mu'\le m'\\ 1\le\nu\le n,1\le\nu'\le n'}}$$

$$= t\otimes t',$$

which proves the lemma. □

For the tensor product of matrix multiplications, we have

$$\begin{aligned}\langle k,m,n\rangle\otimes\langle k',m',n'\rangle &= (\delta_{\kappa\bar\kappa}\delta_{\mu\bar\mu}\delta_{\nu\bar\nu}\delta_{\kappa'\bar\kappa'}\delta_{\mu'\bar\mu'}\delta_{\nu'\bar\nu'})\\ &= (\delta_{\kappa\bar\kappa}\delta_{\kappa'\bar\kappa'}\delta_{\mu\bar\mu}\delta_{\mu'\bar\mu'}\delta_{\nu\bar\nu}\delta_{\nu'\bar\nu'})\\ &= \big(\delta_{(\kappa,\kappa'),(\bar\kappa,\bar\kappa')}\delta_{(\mu,\mu'),(\bar\mu,\bar\mu')}\delta_{(\nu,\nu'),(\bar\nu,\bar\nu')}\big)\\ &= \langle kk',mm',nn'\rangle .\end{aligned}$$

Thus, the tensor product of two matrix tensors is a bigger matrix tensor. This corresponds to the well known identity $(A\otimes B)(A'\otimes B') = (AA'\otimes BB')$ for the Kronecker product of matrices. (Note that we use quadruple indices to address the entries of the Kronecker products and also of the slices of $\langle k,m,n\rangle\otimes\langle k',m',n'\rangle$.) It follows that the inequality in Lemma 5.8 can be strict. We have $R(\langle 2,2,2\rangle) = 7$, but there are faster ways to multiply matrices than Strassen's algorithm.

Using this machinery, we can show that whenever we can multiply matrices of a fixed format efficiently, then we get good bounds for ω.

Theorem 5.9. *If $R(\langle k,m,n\rangle)\le r$, then $\omega\le 3\cdot\log_{kmn} r$.*

Proof. If $R(\langle k,m,n\rangle)\le r$, then $R(\langle n,k,m\rangle)\le r$ and $R(\langle m,n,k\rangle)\le r$ by Lemma 5.5. Thus, by Lemma 5.8,

$$R(\underbrace{\langle k,m,n\rangle\otimes\langle n,k,m\rangle\otimes\langle m,n,k\rangle}_{=\langle kmn,kmn,kmn\rangle})\le r^3$$

and, with $N = kmn$,

$$R(\langle N^i,N^i,N^i\rangle\le r^{3i} = (N^{3\log_N r})^i = (N^i)^{3\log_N r}$$

for all $i\ge 1$. Therefore, $\omega\le 3\log_N r$. □

Example 5.10 (Matrix tensors of small format)**.** What do we know about the rank of matrix tensors of small formats?

- $R(\langle 2,2,2\rangle)\le 7 \implies \omega\le 3\cdot\log_{2^3} 7 = \log_2 7\approx 2.81$.

- $R(\langle 2,2,3\rangle) \le 11$. (This is achieved by doing Strassen once and one trivial matrix-vector product.) This gives a worse bound than 2.81. A lower bound of 11 is shown by [1].
- $14 \le R(\langle 2,3,3\rangle) \le 15$, see [8] for corresponding references.
- $19 \le R(\langle 3,3,3\rangle) \le 23$. The lower bound is shown in [5], the upper bound is due to Laderman [21]. (We would need ≤ 21 to get an improvement.)
- $R(\langle 70,70,70\rangle) \le 143.640$ [25]. This gives $\omega \le 2.80$. (Don't panic, there is a structured way to come up with this algorithm.)

Research problem 5.11. What is the complexity of tensor rank? Håstad [17] has shown that this problem is NP-complete over $\mathbb{F}_q$ and NP-hard over $\mathbb{Q}$. What upper bounds can we show over $\mathbb{Q}$? Over $\mathbb{R}$, the problem is decidable, even in PSPACE, since it reduces to the existential theory over the reals.

6 Border rank

Over $\mathbb{R}$ or $\mathbb{C}$, the rank of matrices is semi-continuous. Let

$$\mathbb{R}^{n\times n} \ni A_j \to A = \lim_{j\to\infty} A_j \,.$$

If for all j, $\mathrm{rk}(A_j) \le r$, then $\mathrm{rk}(A) \le r$ as $\mathrm{rk}(A_j) \le r$ means all $(r+1)\times(r+1)$ minors vanish. But since minors are continuous functions, all $(r+1)\times(r+1)$ minor of A vanish, too.

The same is not true for 3-dimensional tensors. Consider the multiplication of univariate polynomials of degree one modulo X^2:

$$(a_0 + a_1 X)(b_0 + b_1 X) = a_0 b_0 + (a_1 b_0 + a_0 b_1)X + a_1 b_1 X^2 \,.$$

The tensor corresponding to the two bilinear forms $a_0 b_0$ and $a_1 b_0 + a_0 b_1$ has rank 3:

$$\begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & 0 \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline 0 & 1 \\ \hline 1 & 0 \\ \hline\end{array}$$

To show the lower bound, we use the substitution method. We first set $a_0 = 0$, $b_0 = 1$. Then we still compute a_1. Thus there is a product that depends on a_1, say one factor is $\alpha a_0 + \beta a_1$ with $\beta \ne 0$. When we replace a_1 by $-\frac{\alpha}{\beta} a_0$, we kill one product. We still compute

$$a_0 b_0 \qquad \text{and} \qquad -\frac{\alpha}{\beta} a_0 b_0 + a_0 b_1 \,.$$

Next, set $a_0 = 1$, $b_0 = 0$. Then we still compute b_1. We can kill another product by substituting b_1 as above. After this, we still compute $a_0 b_0$, which needs one product.

However, we can approximate the tensor above by tensors of rank two. Let

$$t(\varepsilon) = (1,\varepsilon)\otimes(1,\varepsilon)\otimes(0,\tfrac{1}{\varepsilon}) + (1,0)\otimes(1,0)\otimes(1,-\tfrac{1}{\varepsilon}) \,.$$

$t(\varepsilon)$ obviously has rank two for every $\varepsilon > 0$. The slices of $t(\varepsilon)$ are

1	0
0	0

0	1
1	ε

Thus $t(\varepsilon) \to t$ if $\varepsilon \to 0$.

Bini, Capovani, Lotti and Romani [4] used this effect to design better matrix multiplication algorithms. They started with the following partial matrix multiplication:

$$\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} \left(\begin{array}{c|c} y_{11} & y_{12} \\ y_{21} & y_{22} \end{array}\right) = \left(\begin{array}{c|c} z_{11} & z_{12} \\ z_{21} & \cancel{z_{22}} \end{array}\right)$$

where we only want to compute three entries of the result. We have $R(\{z_{11}, z_{12}, z_{21}\}) = 6$ but we can approximate $\{z_{11}, z_{12}, z_{21}\}$ with only five products.

That the rank is six can be shown using the substitution method. Consider z_{12}. It clearly depends on y_{12}, so there is (after appropriate scaling) a product with one factor being $y_{12} + \ell(y_{11}, y_{21}, y_{22})$ where ℓ is a linear form. Substitute $y_{12} \to -\ell(y_{11}, y_{21}, y_{22})$. This substitution only affects z_{12}. After this substitution we still compute $\bar{z}_{12} = x_{11}(-\ell(y_{11}, y_{21}, y_{22})) + x_{12}y_{22}$. $\bar{z}_{12}$ still depends on y_{22}. Thus we can substitute again $y_{22} \to -\ell'(y_{11}, y_{21})$. This kills two products and we still compute z_{11}, z_{21}. But this is nothing else than $\langle 2, 2, 1 \rangle$, which has rank four.

Consider the following five products:

$$\begin{aligned}
p_1 &= (x_{12} + \varepsilon x_{22})y_{21}\,, \\
p_2 &= x_{11}(y_{11} + \varepsilon y_{12})\,, \\
p_3 &= x_{12}(y_{11} + y_{21} + \varepsilon y_{22})\,, \\
p_4 &= (x_{11} + x_{12} + \varepsilon x_{21})y_{11}\,, \\
p_5 &= (x_{12} + \varepsilon x_{21})(y_{11} + \varepsilon y_{22})\,.
\end{aligned}$$

We have

$$\begin{aligned}
\varepsilon z_{11} &= \varepsilon p_1 + \varepsilon p_2 + O(\varepsilon^2)\,, \\
\varepsilon z_{12} &= p_2 - p_4 + p_5 + O(\varepsilon^2)\,, \\
\varepsilon z_{21} &= p_1 - p_3 + p_5 + O(\varepsilon^2)\,.
\end{aligned}$$

Here, $O(\varepsilon^i)$ collects terms of degree i or higher in ε. Now we take a second copy of the partial matrix multiplication above, with new variables. With these two copies, we can multiply 2×2-matrices with 2×3-matrices (by identifying some of the variables in the copy). So we can approximate $\langle 2, 2, 3 \rangle$ with 10 multiplications. If approximation would be as good as exact computation, then we would get $\omega \leq 2.78$ out of this, an improvement over Strassen's algorithm.

We will formalize the concept of approximation. Let K be a field and $K[[\varepsilon]] =: \hat{K}$. The role of the small quantity ε in the beginning of this section is now taken by the indeterminate ε.

Definition 6.1. Let $h \in \mathbb{N}$, $t \in K^{k \times m \times n}$.

1. $R_h(t) = \min\{r \mid \exists u_\rho \in K[\varepsilon]^k, v_\rho \in K[\varepsilon]^m, w_\rho \in K[\varepsilon]^n : \sum_{\rho=1}^{r} u_\rho \otimes v_\rho \otimes w_\rho = \varepsilon^h t + O(\varepsilon^{h+1})\}$.

2. $\underline{R}(t) = \min_h R_h(t)$. $\underline{R}(t)$ is called the *border rank* of t.

Remark 6.2.

1. $R_0(t) = R(t)$.
2. $R_0(t) \geq R_1(t) \geq \cdots = \underline{R}(t)$.
3. For $R_h(t)$ it is sufficient to consider powers up to ε^h in u_ρ, v_ρ, w_ρ.

Theorem 6.3. *Let $t \in K^{k\times m\times n}$, $t' \in K^{k'\times m'\times n'}$. We have*

1. $\forall \pi \in S_3 : R_h(\pi t) = R_h(t)$.
2. $R_{\max\{h,h'\}}(t \oplus t') \leq R_h(t) + R_{h'}(t')$.
3. $R_{h+h'}(t \otimes t') \leq R_h(t) \cdot R_{h'}(t')$.

Proof.

1. Clear.
2. W. l. o. g. $h \geq h'$. There are approximate computations such that

$$\sum_{\rho=1}^{r} u_\rho \otimes v_\rho \otimes w_\rho = \varepsilon^h t + O(\varepsilon^{h+1}) \quad \text{and} \tag{6.1}$$

$$\sum_{\rho=1}^{r'} \varepsilon^{h-h'} u'_\rho \otimes v'_\rho \otimes w'_\rho = \varepsilon^{\overset{h}{\cancel{h'}}} t' + O(\varepsilon^{\overset{h}{\cancel{h'}}+1})\,. \tag{6.2}$$

   Now we can combine these two computations as we did in the case of rank.

3. Let $t = (t_{ijl})$ and $t' = (t'_{i'j'l'})$. We have $t \otimes t' = (t_{ijl} \cdot t'_{i'j'l'}) \in K^{kk'\times mm'\times nn'}$. Take two approximate computations for t and t' as above. Viewed as exact computations over $K[[\varepsilon]]$, their tensor product computes over the following:

$$T = \varepsilon^h t + \varepsilon^{h+1} s, \qquad T' = \varepsilon^{h'} t' + \varepsilon^{h'+1} s'$$

   with $s \in K[\varepsilon]^{k\times m\times n}$ and $s' \in K[\varepsilon]^{k'\times m'\times n'}$. The tensor product of these two computations computes:

$$\begin{aligned} T \otimes T' &= (\varepsilon^h t_{ijl} + \varepsilon^{h+1} s_{ijl})(\varepsilon^{h'} t'_{i'j'l'} + \varepsilon^{h'+1} s'_{i'j'l'}) \\ &= (\varepsilon^{h+h'} t_{ijl} t'_{i'j'l'} + O(\varepsilon^{h+h'+1})) \\ &= \varepsilon^{h+h'} t \otimes t' + O(\varepsilon^{h+h'+1})\,. \end{aligned}$$

   But this is an approximate computation for $t \otimes t'$. $\square$

The next lemma shows that we can turn approximate computations into exact ones.

Lemma 6.4. *There is a constant c_h such that for all t : $R(t) \leq c_h R_h(t)$. c_h depends polynomially on h, in particular $c_h \leq \binom{h+2}{2}$.*

Remark 6.5. Over infinite fields, even $c_h = 1+2h$ works.

Proof. Let t be a tensor with border rank r and let

$$\sum_{\rho=1}^{r} \underbrace{\left(\sum_{\alpha=0}^{h} \varepsilon^{\alpha} u_{\rho\alpha}\right)}_{\in K[\varepsilon]^k} \otimes \left(\sum_{\beta=0}^{h} \varepsilon^{\beta} v_{\rho\beta}\right) \otimes \left(\sum_{\gamma=0}^{h} \varepsilon^{\gamma} w_{\rho\gamma}\right) = \varepsilon^h t + O(\varepsilon^{h+1}).$$

The left-hand side of the equation can be rewritten as follows:

$$\sum_{\rho=1}^{r} \sum_{\alpha=0}^{h} \sum_{\beta=0}^{h} \sum_{\gamma=0}^{h} \varepsilon^{\alpha+\beta+\gamma} u_{\rho\alpha} \otimes v_{\rho\beta} \otimes w_{\rho\gamma}.$$

By comparing the coefficients of ε powers, we see that t is the sum of all $u_{\rho\alpha} \otimes v_{\rho\beta} \otimes w_{\rho\gamma}$ with $\alpha+\beta+\gamma = h$. Thus to compute t exactly, it is sufficient to compute $\binom{h+2}{2}$ products for each product in the approximate computation. □

A first attempt to apply the results above is to proceed as follows: We have $R_1(\langle 2,2,3\rangle) \leq 10$. $R_1(\langle 3,2,2\rangle) \leq 10$ and $R_1(\langle 2,3,2\rangle) \leq 10$ follows by Theorem 6.3(1). By Theorem 6.3(3), we may conclude $R_3(\langle 12,12,12\rangle) \leq 1000$. By Lemma 6.4

$$R(\langle 12,12,12\rangle) \leq \binom{3+2}{2} \cdot 1000 = 10 \cdot 1000 = 10000.$$

But trivially, $R(\langle 12,12,12\rangle) \leq 12^3 = 1728$. It turns out that it is better to first "tensor up" and then turn the approximate computation into the exact one.

Theorem 6.6. *If $\underline{R}(\langle k,m,n\rangle) \leq r$ then $\omega \leq 3\log_{kmn} r$.*

Proof. Let $N = kmn$ and let $R_h(\langle k,m,n\rangle) \leq r$. By Theorem 6.3, we get $R_{3h}(\langle N,N,N\rangle) \leq r^3$ and $R_{3hs}(\langle N^s,N^s,N^s\rangle) \leq r^{3s}$ for all s. By Lemma 6.4, this yields $R(\langle N^s,N^s,N^s\rangle) \leq c_{3hs} r^{3s}$. Therefore,

$$\omega \leq \log_{N^s}(c_{3hs} r^{3s}) = 3s\log_{N^s}(r) + \log_{N^s}(c_{3hs}) = 3\log_N(r) + \underbrace{\frac{1}{s}\log_N(\mathrm{poly}(s))}_{\to 0}.$$

Since ω is an infimum, we get $\omega \leq 3\log_N(r)$. □

Corollary 6.7. $\omega \leq 2.78$.

7 Schönhage's τ-Theorem

Strassen "just" gave a clever algorithm for multiplying 2×2-matrices to obtain a fast algorithm for multiplying matrices. Bini et al. showed that is sufficient to approximate a fixed size matrix tensor instead of computing it exactly. In this section, we will show how to make use of a fast algorithm that approximates a tensor that is not a matrix tensor at all! In in the subsequent two sections, we will see the same with tensors that are even "less" matrix tensors than the one in this section.

Note that Bini et al. start with a tensor corresponding to a partial matrix multiplication. They glue two of them together to get a matrix tensor. Schönhage [28] observed that it is better to take the partial matrix multiplication, tensor up first, and then try to get a large total matrix multiplication out of the resulting tensor. The interested reader is referred to Schönhage's original paper. We will not deal with this method here, since the same paper contains a second, related method that gives even better results, the so-called *τ-Theorem*.[10]

We will consider an extreme case of a partial matrix multiplication, namely direct sums of matrix tensors. Direct sums of matrix tensors correspond to independent matrix multiplications and we can view them as partial matrix multiplications by embedding the factors in large block diagonal matrices. In particular, we will look at sums of the form $R(\langle k,1,n\rangle \oplus \langle 1,m,1\rangle)$. The first summand is the product of a vector of length k with a vector of length n, forming a rank-one matrix. The second summand is a scalar product of two vectors of length m.

Lemma 7.1.

1. $R(\langle k,1,n\rangle \oplus \langle 1,m,1\rangle) = k \cdot n + m.$

2. $\underline{R}(\langle k,1,n\rangle) = k \cdot n$ *and* $\underline{R}(\langle 1,m,1\rangle) = m.$

3. $\underline{R}(\langle k,1,n\rangle \oplus \langle 1,m,1\rangle) \leq k \cdot n + 1$ *with* $m = (n-1)(k-1).$

The first statement is shown by using the substitution method. We first substitute m variables belonging to one vector of $\langle 1,m,1\rangle$. Then we set the variables of the other vector to zero. We still compute $\langle k,1,n\rangle$.

For the second statement, it is sufficient to note that both tensors consist of kn and m linearly independent slices, respectively, even over $K(\varepsilon)$.

For the third statement, we just prove the case $k = n = 3$. From this, the general construction becomes

[10] According to Schönhage, the term τ-Theorem was coined by Hans F. de Groote in his lecture notes [16].

obvious. So we want to approximate $a_i b_j$ for $1 \le i, j \le 3$ and $\sum_{\mu=1}^{4} u_\mu v_\mu$. Consider the following products

$$\begin{aligned}
p_1 &= (a_1 + \varepsilon u_1)(b_1 + \varepsilon v_1)\,, \\
p_2 &= (a_1 + \varepsilon u_2)(b_2 + \varepsilon v_2)\,, \\
p_3 &= (a_2 + \varepsilon u_3)(b_1 + \varepsilon v_3)\,, \\
p_4 &= (a_2 + \varepsilon u_4)(b_2 + \varepsilon v_4)\,, \\
p_5 &= (a_3 - \varepsilon u_1 - \varepsilon u_3) b_1\,, \\
p_6 &= (a_3 - \varepsilon u_2 - \varepsilon u_4) b_2\,, \\
p_7 &= a_1 (b_3 - \varepsilon v_1 - \varepsilon v_2)\,, \\
p_8 &= a_2 (b_3 - \varepsilon v_3 - \varepsilon v_4)\,, \\
p_9 &= a_3 b_3\,.
\end{aligned}$$

These nine product obviously compute $a_i b_j$ up to terms of order ε, $1 \le i, j \le 3$. Furthermore,

$$\varepsilon^2 \sum_{\mu=1}^{4} u_\mu v_\mu = p_1 + \cdots + p_9 - (a_1 + a_2 + a_3)(b_1 + b_2 + b_3)\,.$$

Thus ten products are sufficient to approximate $\langle 3,1,3\rangle \oplus \langle 1,4,1\rangle$.[11]

The second and the third statement together show, that the additivity conjecture is *not* true for the border rank.

Definition 7.2. Let $t \in K^{k \times m \times n}$ and $t' \in K^{k' \times m' \times n'}$.

1. t is called a *restriction* of t' if there are homomorphisms $\alpha : K^{k'} \to K^k$, $\beta : K^{m'} \to K^m$, and $\gamma : K^{n'} \to K^n$ such that $t = (\alpha \otimes \beta \otimes \gamma) t'$. We write $t \le t'$.

2. t and t' are isomorphic if α, β, γ are isomorphisms ($t \cong t'$).

In the following, $\langle r\rangle$ denotes the tensor in $K^{r \times r \times r}$ that has a 1 in the positions (ρ, ρ, ρ), $1 \le \rho \le r$, and 0s elsewhere (a "diagonal," the three-dimensional analogue of the identity matrix). This tensor corresponds to the r bilinear forms $x_\rho y_\rho$, $1 \le \rho \le r$ (r independent products).

Lemma 7.3. $R(t) \le r \Leftrightarrow t \le \langle r\rangle$.

Proof. "$\Leftarrow$": follows immediately from Lemma 5.4.

[11] Note how amazing this is: Asume that in the good old times, when computers were rare and expensive, you were working at the computer center of your university. A chemistry professor approaches you and tells you that he has some data and needs to compute a large rank one matrix from it. He needs the results the next day. Since computers were not only rare and expensive, but also slow, the computing capacity of the center barely suffices to compute the product in one day. But then a physics professor calls you: She needs to compute a scalar product of a similar size and again, she wants the result the next day. When you compute exactly, you have to upset one of them, no matter what. But if you are willing to approximate the results, and, hey, they will not recognize this anyway because of measurement errors, then you can satisfy both of them!

"$\Rightarrow$": $\langle r \rangle = \sum_{\rho=1}^{r} e_\rho \otimes e_\rho \otimes e_\rho$, where e_ρ is the ρth unit vector. If the rank of t is $\leq r$, then we can write t as the sum of r triads,

$$t = \sum_{\rho=1}^{r} u_\rho \otimes v_\rho \otimes w_\rho \,.$$

We define three homomorphisms

$$\begin{aligned} \alpha &: e_\rho \mapsto u_\rho, \ \ 1 \leq \rho \leq r, \\ \beta &: e_\rho \mapsto v_\rho, \ \ 1 \leq \rho \leq r, \\ \gamma &: e_\rho \mapsto w_\rho, \ \ 1 \leq \rho \leq r. \end{aligned}$$

By construction,

$$(\alpha \otimes \beta \otimes \gamma)\langle r \rangle = \sum_{\rho=1}^{r} \underbrace{\alpha(e_\rho)}_{=u_\rho} \otimes \underbrace{\beta(e_\rho)}_{=v_\rho} \otimes \underbrace{\gamma(e_\rho)}_{=w_\rho} = t \,.$$

□

Observation 7.4.

1. $t \otimes t' \cong t' \otimes t$,
2. $t \otimes (t' \otimes t'') \cong (t \otimes t') \otimes t''$,
3. $t \oplus t' \cong t' \oplus t$,
4. $t \oplus (t' \oplus t'') \cong (t \oplus t') \oplus t''$,
5. $t \otimes \langle 1 \rangle \cong t$,
6. $t \oplus \langle 0 \rangle \cong t$,
7. $t \otimes (t' \oplus t'') \cong t \otimes t' \oplus t \otimes t''$.

Above, $\langle 0 \rangle$ is the empty tensor in $K^{0\times 0\times 0}$. So the (isomorphism classes of) tensors form a semi-ring.[12]

The main result of this section is the following theorem due to Schönhage [28]. It is often called the τ-theorem in the literature, because the letter τ has a leading role in the original proof. But in our proof, it only has a minor one.

Theorem 7.5 (Schönhage's τ-theorem)**.** *If $\underline{R}(\bigoplus_{i=1}^{p} \langle k_i, m_i, n_i \rangle) \leq r$ with $r > p$ then $\omega \leq 3\tau$ where τ is defined by*

$$\sum_{i=1}^{p} (k_i \cdot m_i \cdot n_i)^\tau = r \,.$$

[12] If two tensors are isomorphic, then the live in they same space $K^{k\times m\times n}$. If t is any tensor and n is a tensor that is completely filled with zeros, then t is not isomorphic to $t \oplus n$. But from a computational viewpoint, these tensors are the same. So it is also useful to use this wider notion of equivalence: Two tensors t and t' are isomorphic, if there are tensors n and n' completely filled with zeros such that $t \oplus n$ and $t' \oplus n'$ are isomorphic.

Notation 7.6. Let $f \in \mathbb{N}$ and t be a tensor. $f \odot t := \underbrace{t \oplus \cdots \oplus t}_{f \text{ times}}$.

Lemma 7.7. *If* $R(f \odot \langle k,m,n \rangle) \leq g$, *then* $\omega \leq 3 \cdot \dfrac{\log \left\lceil \frac{g}{f} \right\rceil}{\log(kmn)}$.

Proof. We first show that for all s,

$$R(f \odot \langle k^s, m^s, n^s \rangle) \leq \left\lceil \frac{g}{f} \right\rceil^s \cdot f\,.$$

The proof is by induction on s. If $s = 1$, this is just the assumption of the lemma. For the induction step $s \mapsto s+1$, note that

$$\begin{aligned} f \odot \left\langle k^{s+1}, m^{s+1}, n^{s+1} \right\rangle &= \underbrace{(f \odot \langle k,m,n \rangle)}_{\leq \langle g \rangle} \otimes \langle k^s, m^s, n^s \rangle \\ &\leq \langle g \rangle \otimes \langle k^s, m^s, n^s \rangle \\ &= g \odot \langle k^s, m^s, n^s \rangle. \end{aligned}$$

Therefore,

$$\begin{aligned} R(f \odot \left\langle k^{s+1}, m^{s+1}, n^{s+1} \right\rangle) &\leq R\left(g \odot \langle k^s, m^s, n^s \rangle\right) \\ &\leq R\Big(\left\lceil \frac{g}{f} \right\rceil \cdot f \odot \langle k^s, m^s, n^s \rangle \Big) \\ &= \left\lceil \frac{g}{f} \right\rceil \cdot \left\lceil \frac{g}{f} \right\rceil^s \cdot f \\ &= \left\lceil \frac{g}{f} \right\rceil^{s+1} f. \end{aligned}$$

This shows the claim. Now use the claim to prove our lemma: $R(f \odot \langle k^s, m^s, n^s \rangle) \leq \left\lceil \frac{g}{f} \right\rceil^s \cdot f$ implies

$$\omega \leq \frac{3s \log \left\lceil \frac{g}{f} \right\rceil + \log(f) \cdot 3}{s \cdot \log(kmn)} = \frac{3 \log \left\lceil \frac{g}{f} \right\rceil + \overbrace{\log(f) \cdot \frac{3}{s}}^{\to 0 \text{ for } s \to \infty}}{\log(kmn)}.$$

Since ω is an infimum, we get $\omega \leq \dfrac{3 \log \left\lceil \frac{g}{f} \right\rceil}{\log(kmn)}$. □

Proof of Theorem 7.5. There is an h such that

$$R_h\Big(\bigoplus_{i=1}^{p} \langle k_i, m_i, n_i \rangle\Big) \leq r\,.$$

By taking tensor powers and using the fact that the tensors form a ring, we get

$$R_{hs}\left(\bigoplus_{\sigma_1+\ldots+\sigma_p=s}\frac{s!}{\sigma_1!\cdot\ldots\cdot\sigma_p!}\odot\left\langle\underbrace{\prod_{i=1}^{p}k_i^{\sigma_i}}_{=k'},\underbrace{\prod_{i=1}^{p}m_i^{\sigma_i}}_{=m'},\underbrace{\prod_{i=1}^{p}n_i^{\sigma_i}}_{=n'}\right\rangle\right)\leq r^s\,.$$

k',m',n' depend on $\sigma_1,\ldots,\sigma_p$. Next, we convert the approximate computation into an exact one and get

$$R\left(\bigoplus_{\sigma_1+\ldots+\sigma_p=s}\frac{s!}{\sigma_1!\cdot\ldots\cdot\sigma_p!}\odot\langle k',m',n'\rangle\right)\leq r^s\cdot c_{hs}\,.$$

Recall that c_{hs} is a polynomial in h and s. By raising the defining equation for τ in the statement of the theorem to the sth power, we see that

$$\sum_{s=\sigma_1+\ldots+\sigma_p}\underbrace{\frac{s!}{\sigma_1!\cdot\ldots\cdot\sigma_p!}(k'\cdot m'\cdot n')^{\tau}}_{=(*)}=r^s\,.$$

Fix $\sigma_1,\ldots,\sigma_p$ such that (*) is maximized. Then k', m', and n' are constant. To apply Lemma 7.7, we set

$$\begin{aligned}
f&=\frac{s!}{\sigma_1!\cdot\ldots\cdot\sigma_p!}<p^s\,,\\
g&=r^s\cdot c_{hs}\,,\\
m&=m'\,,\\
k&=k'\,,\\
n&=n'\,.
\end{aligned}$$

The number of all $\vec{\sigma}$ with $\sigma_1+\ldots+\sigma_p=s$ is

$$\binom{s+p-1}{p-1}=\frac{s+p-1}{p-1}\cdot\frac{s+p-2}{p-2}\cdots\leq(s+1)^{p-1}\,.$$

Thus

$$f\cdot(kmn)^{\tau}\geq\frac{r^s}{(s+1)^{p-1}}.$$

We get that

$$\left\lceil\frac{g}{f}\right\rceil\leq\frac{r^s\cdot c_{hs}}{f}+1\leq(kmn)^{\tau}\cdot(s+1)^{p-1}\cdot c_{hs}\,.$$

Furthermore,

$$(kmn)^{\tau}\geq\frac{r^s}{(s+1)^{p-1}f}\geq\frac{r^s}{(s+1)^{p-1}p^s}. \tag{7.1}$$

By Lemma 7.7,

$$\begin{aligned}\omega &\le 3\cdot\frac{\tau\cdot\log(kmn)+(p-1)\cdot\log(s+1)+\log(c_{hs})}{\log(kmn)}\\ &= 3\tau+\frac{(p-1)\log(s+1)+\log(c_{hs})}{\log(kmn)}\underset{s\to\infty}{\to} 3\tau\,.\end{aligned}$$

because

$$\log(kmn)\ge\frac{s}{\tau}\cdot\underbrace{(\log r-\log p)}_{>0\ \text{ by assumption}}-O(\log(s))$$

by (7.1). □

By using the example at the beginning of this section with $k=4$ and $n=3$, we get the following bound out of the τ-theorem.

Corollary 7.8. $\omega\le 2.55$.

What is the algorithmic intuition behind the τ-theorem? If we take the sth tensor power of a sum of N independent matrix products, we get a sum of N^s independent matrix products. From these matrix products, we choose a subset with isomorphic tensors. In the proof of the theorem, this is done when maximizing the quantity (*). Assume we get ℓ matrix products of the form $\langle k,m,n\rangle$. What can we do with this? Well, we can compute a large matrix product $\langle tk,tm,tn\rangle$ with $t^3\le\ell$ by using the trivial algorithm for multiplying $\langle t,t,t\rangle$ together with the ℓ independent products for $\langle k,m,n\rangle$, each of them replacing one of the multiplications in the trivial algorithm. We get a new improved algorithm for multiplying matrices. If we use this new algorithm for computing $\langle t,t,t\rangle$, we get an even better algorithm, and so on. The bound on the exponent that we get in the limit is the one given by the τ-theorem. Along with this, we also get an algorithm to compute the value of τ, see the original paper by Schönhage.

Coppersmith and Winograd [12] optimize this approach by introducing the concept of *null-like* tensors. They were able to get an upper bound <2.5 with their approach. Before this result, according to Schönhage, quite a few researchers conjectured that ω might be 2.5, since there were some further improvements, for instance by V. Pan, by using better starting algorithms, moving the upper bounds close to 2.5 (see the original paper by Schönhage).

8 Strassen's Laser method

Consider the following tensor (see Figure 7 for a pictorial description)

$$\mathrm{Str}=\sum_{i=1}^{q}(\underbrace{e_i\otimes e_0\otimes e_i}_{\langle q,1,1\rangle}+\underbrace{e_0\otimes e_i\otimes e_i}_{\langle 1,1,q\rangle})\,.$$

This tensor is similar to $\langle 1,2,q\rangle$, only the "directions" of the two scalar products are not the same. But Strassen's tensor can be approximated very efficiently. We have

$$\sum_{i=1}^{q}(e_0+\varepsilon e_i)\otimes(e_0+\varepsilon e_i)\otimes e_i=\sum_{i=1}^{q}e_0\otimes e_0\otimes e_i+\varepsilon\sum_{i=1}^{q}(e_i\otimes e_0\otimes e_i+e_0\otimes e_i\otimes e_i)+O(\varepsilon^2)\,.$$

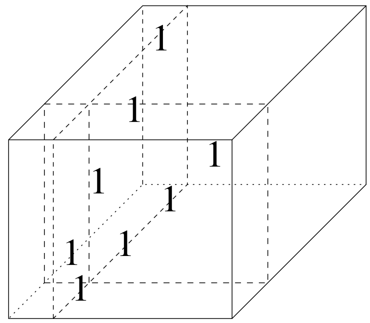

Figure 7: Strassen's tensor.

If we subtract the triad $e_0 \otimes e_0 \otimes \sum_{i=1}^{q} e_i$, we get an approximation of Str. Thus $\underline{R}(\mathrm{Str}) \leq q+1$. On the other hand, $\underline{R}(\langle 1,2,q \rangle) = 2q$. Can we make use of this very cheap tensor?

Definition 8.1. Let $t \in K^{k \times m \times n}$ be a tensor. Let $I_1, \dots, I_p$, $J_1, \dots, J_q$, and $L_1, \dots, L_s$ be sets such that

$$\begin{aligned} I_i &\subseteq \{1,\dots,k\}, & 1 \leq i \leq p, \\ J_j &\subseteq \{1,\dots,m\}, & 1 \leq j \leq q, \\ L_\ell &\subseteq \{1,\dots,n\}, & 1 \leq \ell \leq s. \end{aligned}$$

1. The sets are called a *decomposition* $\mathcal{D}$ of format $k \times m \times n$ if

$$\begin{aligned} I_1 \dot\cup I_2 \dot\cup \cdots \dot\cup I_p &= \{1,\dots,k\}, \\ J_1 \dot\cup J_2 \dot\cup \cdots \dot\cup J_q &= \{1,\dots,m\}, \\ L_1 \dot\cup L_2 \dot\cup \cdots \dot\cup L_s &= \{1,\dots,n\}. \end{aligned}$$

2. $t_{I_i,J_j,L_\ell} \in K^{|I_i| \times |J_j| \times |L_\ell|}$ is the tensor that one gets when restricting t to the slices in I_i, J_j, L_ℓ, i. e.,

$$t_{I_i,J_j,L_\ell}(a,b,c) = t(\hat{a},\hat{b},\hat{c})$$

where $\hat{a} =$ the ath smallest element in I_i and $\hat{b}$ and $\hat{c}$ are defined analogously.[13]

3. $t_{\mathcal{D}} \in K^{p \times q \times s}$ is defined by

$$t_{\mathcal{D}}(i,j,l) = \begin{cases} 1 & \text{if } t_{I_i,J_j,L_\ell} \neq 0, \\ 0 & \text{otherwise.} \end{cases}$$

4. Finally, $\mathrm{supp}_{\mathcal{D}}\, t = \{(i,j,\ell) \mid t_{I_i,J_j,L_\ell} \neq 0\}$.

We can think of giving the tensors an "inner" and an "outer" structure. A decomposition cuts the tensor into (combinatorial) cuboids t_{I_i,J_j,L_ℓ}, these cuboids need not be connected. The cuboids form the inner structure. For the outer structure $t_{\mathcal{D}}$, we interpret each set I_i or J_j or L_ℓ as a single index. If the corresponding inner tensor t_{I_i,J_j,L_ℓ} is nonzero, we put a 1 into position (i,j,ℓ). The support is just the set of all places where we put a 1 in $t_{\mathcal{D}}$.

[13] To avoid multiple indices, we here use the notation $t(a,b,c)$ to access the element in position (a,b,c) instead of $t_{a,b,c}$.

Definition 8.2. Let $\mathcal{D}$ and $\mathcal{D}'$ be two decompositions for format $k \times m \times n$ and $k' \times m' \times n'$ consisting of sets $I_1, \dots, I_p$, $J_1, \dots, J_q$, and $L_1, \dots, L_s$ and $I'_1, \dots, I'_{p'}$, $J'_1, \dots, J'_{q'}$, and $L'_1, \dots, L'_{s'}$. Their product $\mathcal{D} \otimes \mathcal{D}'$ is a decomposition of format $kk' \times mm' \times nn'$ and is given by the sets

$$\begin{array}{lll} I_i \times I'_{i'}\,, & 1 \le i \le p\,, & 1 \le i' \le p'\,, \\ J_j \times J'_{j'}\,, & 1 \le j \le q\,, & 1 \le j' \le q'\,, \\ L_\ell \times L'_{\ell'}\,, & 1 \le l \le s\,, & 1 \le l' \le s'\,. \end{array}$$

Lemma 8.3. *Let ρ and ρ' be two sets of tensors. Let $t \in K^{k \times m \times n}$ and $t' \in K^{k' \times m' \times n'}$ with decompositions $\mathcal{D}$ and $\mathcal{D}'$ be given. Assume that $t_{I_i, J_j, L_\ell} \in \rho$ for all $(i, j, \ell) \in supp_{\mathcal{D}} t$ and the same for t'. Then $\mathcal{D} \otimes \mathcal{D}'$ is a decomposition of $t \otimes t'$ such that*

$$(t \otimes t')_{\mathcal{D} \otimes \mathcal{D}'} \cong t_{\mathcal{D}} \otimes t'_{\mathcal{D}'}\,.\text{[14]}$$

Furthermore, $(t \otimes t')_{I_i \times I'_{i'}, J_j \times J'_{j'}, L_\ell \times L'_{\ell'}} \in \rho \otimes \rho'$ for all $(i, j, \ell) \in supp_{\mathcal{D}} t$ and $(i', j', \ell') \in supp_{\mathcal{D}'} t'$, where $\rho \otimes \rho'$ is defined elementwisely.

The proof of the lemma is a somewhat tedious but easy exercise which we leave to the reader.

Next, we decompose Strassen's tensor and analyse its outer structure. We define a decomposition $\mathcal{D}$ as follows:

$$\begin{array}{ccccl} \underset{I_0}{\{0\}} & \dot\cup & \underset{I_1}{\{1, \dots, q\}} & = \{0, \dots, q\}\,, \\ \underset{J_0}{\{0\}} & \dot\cup & \underset{J_1}{\{1, \dots, q\}} & = \{0, \dots, q\}\,, \\ & & \underset{L_1}{\{1, \dots, q\}} & = \{1, \dots, q\}\,. \end{array}$$

With respect to $\mathcal{D}$, we have

$$\mathrm{Str}_{\mathcal{D}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \langle 1, 2, 1 \rangle\,,$$

$$\mathrm{Str}_{I_i, J_j, L_l} \in \{\langle 1, 1, q \rangle, \langle q, 1, 1 \rangle\} \subseteq \{\langle k, m, n \rangle \mid k \cdot m \cdot n = q\}.$$

The format of Str is $(q+1) \times (q+1) \times q$. Next, we make Str symmetric. Take the permutation $\pi = (1\ 2\ 3)$. We have

$$\pi\, \mathrm{Str}_{\pi\mathcal{D}} = \langle 1, 1, 2 \rangle \qquad \text{and} \qquad \pi^2\, \mathrm{Str}_{\pi^2\mathcal{D}} = \langle 2, 1, 1 \rangle\,,$$

where $\pi\mathcal{D}$ and $\pi^2\mathcal{D}$ are the defined by permuting the sets accordingly. Let

$$\text{Sym-Str} = \mathrm{Str} \otimes \pi\, \mathrm{Str} \otimes \pi^2\, \mathrm{Str}\,.$$

By Lemma 8.3, $\hat{\mathcal{D}} = \mathcal{D} \otimes \pi\mathcal{D} \otimes \pi^2\mathcal{D}$ is a decomposition of Sym-Str such that

$$\text{Sym-Str}_{\mathcal{D}} = \langle 2, 2, 2 \rangle$$

and every inner tensor is in

$$\{\langle k, m, n \rangle \mid k \cdot m \cdot n = q^3\}\,.$$

[14]The order of the indices, when building $t \otimes t'$ and $\mathcal{D} \otimes \mathcal{D}'$ should be the same.

Definition 8.4. Let $t \in K^{k\times m\times n}$, $t' \in K^{k'\times m'\times n'}$.

1. Let $t' = \sum_{\rho=1}^{r} u_\rho \otimes v_\rho \otimes w_\rho$ as well as $A(\varepsilon) \in K[\varepsilon]^{k\times k'}$, $B(\varepsilon) \in K[\varepsilon]^{m\times m'}$, and $C(\varepsilon) \in K[\varepsilon]^{n\times n'}$. Define
$$(A(\varepsilon)\otimes B(\varepsilon)\otimes C(\varepsilon))t' = \sum_{\rho=1}^{r} A(\varepsilon)u_\rho \otimes B(\varepsilon)v_\rho \otimes C(\varepsilon)w_\rho\,.$$
(This is well-defined.)

2. t is a *degeneration* of t' if there are $A(\varepsilon) \in K[\varepsilon]^{k\times k'}$, $B(\varepsilon) \in K[\varepsilon]^{m\times m'}$, $C(\varepsilon) \in K[\varepsilon]^{n\times n'}$, and $q \in \mathbb{N}$ such that
$$\varepsilon^q t = (A(\varepsilon)\otimes B(\varepsilon)\otimes C(\varepsilon))t' + O(\varepsilon^{q+1})\,.$$
We will write $t \trianglelefteq_q t'$ or $t \trianglelefteq t'$.

Remark 8.5. $\underline{R}(t) \le r \Leftrightarrow t \trianglelefteq \langle r\rangle$.

The remark above can be interpreted as follows: If you want to "buy" a tensor, then it costs r multiplications. Then next lemma is a kind of a converse. It tells you, that when you bought a matrix tensor $\langle n,n,n\rangle$, then you can "resell" it and get $\Omega(n^2)$ single multiplications back.

Lemma 8.6.
$$\left\langle \left\lceil \frac{3}{4}n^2 \right\rceil \right\rangle \trianglelefteq \langle n,n,n\rangle\,.$$

Proof. First assume that n is odd, $n = 2\nu+1$. We label rows and columns from $-\nu,\dots,\nu$. We define the linear mappings $A,B,C : K^{n\times n} \to K[\varepsilon]^{n\times n}$ by
$$\begin{array}{lll} A: & e_{ij} & \mapsto e_{ij}\cdot\varepsilon^{i^2+2ij}\,,\\ B: & e_{jk} & \mapsto e_{jk}\cdot\varepsilon^{j^2+2jk}\,,\\ C: & e_{ki} & \mapsto e_{ki}\cdot\varepsilon^{k^2+2ki}\,,\end{array}$$
where $e_{i,j}$ denotes the standard basis. A, B, and C define matrices in $K[\varepsilon]^{n^2\times n^2}$. Recall that
$$\langle n,n,n\rangle = \sum_{i,j,k=-\nu}^{\nu} e_{ij}\otimes e_{jk}\otimes e_{ki}\,.$$
We have
$$(A\otimes B\otimes C)\langle n,n,n\rangle = \sum_{i,j,k=-nu}^{\nu} \underbrace{\varepsilon^{i^2+2ij+j^2+2jk+k^2+2ki}}_{=\varepsilon^{(i+j+k)^2}} e_{ij}\otimes e_{jk}\otimes e_{ki}\,.$$

If $i+j+k=0$ then
$$\begin{array}{ccc} i,k & & j\\ i,j & \text{determine} & k\\ j,k & & i \end{array}\,.$$

So all terms with exponent 0 form a set of independent products. It is easy to see that there are $\ge (3/4)n^2$ triples (i,j,k) with $i+j+k=0$. The case when n is even is treated in a similar way. $\square$

Definition 8.7. Let $t \in K^{k\times m\times n}$, $t' \in K^{k'\times m'\times n'}$. t is a monomial degeneration of t' if the entries of the matrices A, B, and C in Definition 8.4 are monomials.

The matrices constructed in Lemma 8.6 are monomial matrices. Therefore,

$$\left\langle \left\lceil \frac{3}{4}n^2 \right\rceil \right\rangle$$

is a monomial degeneration of $\langle n,n,n\rangle$.

Now we want to apply Lemma 8.6 to Sym-Str$_{\hat{\mathfrak{D}}}$. First, we raise Sym-Str to the sth tensorial power. We get

$$\left\langle \frac{3}{4}2^{2s} \right\rangle \underbrace{\trianglelefteq}_{\text{Lemma 8.6}} (\text{Sym-Str})^{\otimes s}_{\hat{\mathfrak{D}}^{\otimes s}} \trianglelefteq_{6s} \left\langle (q+1)^{3s} \right\rangle .$$

The inner tensors or Sym-Str$^{\otimes s}$ are $\in \{\langle k,m,n\rangle \mid k\cdot m\cdot n = q^{3s}\}$. How does this inner structure behave with respect to the degeneration

$$\left\langle \frac{3}{4}2^{2s} \right\rangle \trianglelefteq (\text{Sym-Str})^{\otimes s}_{\hat{\mathfrak{D}}^{\otimes s}} ?$$

Since this degeneration is a monomial degeneration, every 1 in the tensor $\left\langle \frac{3}{4}2^{2s}\right\rangle$ will correspond to *one* tensor in $\{\langle k,m,n\rangle \mid k\cdot m\cdot n = q^{3s}\}$.[15] So we get a direct sum of $\frac{3}{4}2^{2s}$ tensors each of them being in $\{\langle k,m,n\rangle \mid k\cdot m\cdot n = q^{3s}\}$. The border rank of this sum is bound by $(q+1)^{3s}$. But in this situation, we can apply the τ-theorem! We get

$$(q^{3s})^{\tau}\frac{3}{4}2^{2s} \le (q+1)^{3s},$$

$$q^{3\tau}\underbrace{\sqrt[s]{\frac{3}{4}}}_{\to 1}2^2 \le (q+1)^3,$$

$$\omega \le \log_q \frac{(q+1)^3}{4}.$$

The right-hand side is minimal for $q = 5$ and gives us the result $\omega \le 2.48$.

Corollary 8.8 (Strassen [33])**.** $\omega \le 2.48$.

Research problem 8.9. What is $\underline{R}$(Sym-Str)? It is quite easy to see that $\underline{R}(\text{Str}) = q+1$, since it consists of $q+1$ linearly independent slices. But the format of Sym-Str is $q(q+1)^2 \times q(q+1)^2 \times q(q+1)^2$, so it is not clear whether the upper bound $(q+1)^3$ is tight.

Why is the laser method called laser method? Here is an explanation I heard from Amin Shokrollahi who claimed to have heard it from Volker Strassen: In a laser, one generates coherent light. You can think of the two inner tensors in Strassen's tensor as light waves having different polarization. In the end we obtain a diagonal with "light waves" having the same polarization.

[15] If the degeneration were not monomial, then every 1 in $\left\langle \frac{3}{4}2^{2s}\right\rangle$ would be linear combination of several entries of the tensor $(\text{Sym-Str})^{\otimes s}_{\hat{\mathfrak{D}}^{\otimes s}}$. Per se, this is fine. But when looking at the inner structures, then every 1 will correspond to a linear combination of matrix tensor of formats that do not match.

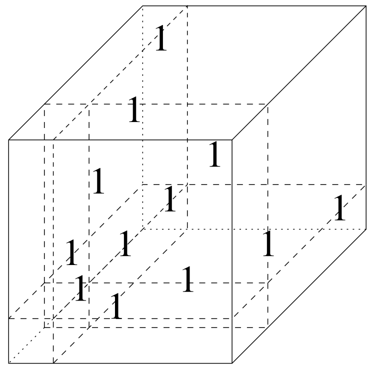

Figure 8: Coppersmith and Winograd's tensor.

9 Coppersmith and Winograd's method

Strassen's tensor is asymmetric, its format is $(q+1)\times(q+1)\times q$. For only one additional multiplication, we can compute the following symmetric variant (see Figure 8 for a pictorial description)

$$\mathrm{CW} = \sum_{i=1}^{q} (\underbrace{e_i \otimes e_0 \otimes e_i}_{\langle q,1,1\rangle} + \underbrace{e_0 \otimes e_i \otimes e_i}_{\langle 1,1,q\rangle} + \underbrace{e_i \otimes e_i \otimes e_0}_{\langle 1,q,1\rangle}).$$

This tensor can be approximated efficiently. We have

$$\begin{aligned}
\varepsilon^5 \cdot \mathrm{CW} = \sum_{i=1}^{q} & \varepsilon \cdot (e_0 + \varepsilon^2 e_i) \otimes (e_0 + \varepsilon^2 e_i) \otimes (e_0 + \varepsilon^2 e_i) \\
& - (e_0 + \varepsilon^3 \sum_{i=1}^{q} e_i) \otimes (e_0 + \varepsilon^3 \sum_{i=1}^{q} e_i) \otimes (e_0 + \varepsilon^3 \sum_{i=1}^{q} e_i) \\
& + (1 - q\varepsilon) \cdot e_0 \otimes e_0 \otimes e_0 \\
& + O(\varepsilon^6).
\end{aligned}$$

Thus, $\underline{R}(\mathrm{CW}) \le q+2$. We define a decomposition $\mathcal{D}$ as follows:

$$\begin{aligned}
\underbrace{\{0\}}_{I_0} \,\dot\cup\, \underbrace{\{1,\dots,q\}}_{I_1} &= \{0,\dots,q\}, \\
\underbrace{\{0\}}_{J_0} \,\dot\cup\, \underbrace{\{1,\dots,q\}}_{J_1} &= \{0,\dots,q\}, \\
\underbrace{\{0\}}_{L_0} \,\dot\cup\, \underbrace{\{1,\dots,q\}}_{L_1} &= \{0,\dots,q\}.
\end{aligned}$$

With respect to $\mathcal{D}$, we have

$$\mathrm{CW}_{\mathcal{D}} = \begin{pmatrix} 2 & 1 \\ 1 & \end{pmatrix},$$
$$\mathrm{CW}_{I_i,J_j,L_\ell} \in \{\langle 1,1,q\rangle, \langle q,1,1\rangle, \langle 1,q,1\rangle\}.$$

The right-hand side of the first equation represents a tensor of format $2\times 2\times 2$. An entry k in position (i,j) means that the (i,j,k)th entry of the tensor is 1. All other entries are 0.

The inner structures with respect to $\mathcal{D}$ are the same as in the previous section. However, $\mathrm{CW}_{\mathcal{D}}$ is not a matrix product anymore. Therefore, we cannot apply the machinery of the previous section. Coppersmith and Winograd [13] found a way to get fast matrix multiplication algorithms from the bound $\underline{R}(\mathrm{CW}) \le q+2$. The proof of their bound that we present here is due to Strassen, see also [8, Sect. 15.7, 15.8]. We follow the proof in the book [8] quite closely. In particular, we use the same notation.

9.1 Tight sets

The question that we have to deal with is the following: Given a tensor t, what is the largest N for which we can show that $\langle N\rangle \trianglelefteq t^{\otimes s}$ by a monomial degeneration? Strassen gave an answer for tensors $t = \langle n,n,n\rangle$. Next, we want to develop a general method.

Definition 9.1. Let I, J, and L be finite sets. Let $A,B \subseteq I\times J\times L$. A is called a *combinatorial degeneration* of B if there are functions $a : I \to \mathbb{Z}$, $b : J \to \mathbb{Z}$, and $c : L \to \mathbb{Z}$ such that

1. $\forall (i,j,\ell) \in A : a(i)+b(j)+c(\ell) = 0$,
2. $\forall (i,j,\ell) \in B\backslash A : a(i)+b(j)+c(\ell) > 0$.

Combinatorial degenerations can be turned into monomial degenerations, this is essentially done in Lemma 8.6. Let $t_A \in K^{|I|\times|J|\times|L|}$ be the tensor that has a 1 in every positions corresponding to tuples in A and 0 elsewhere. Define t_B in the same way. Define mappings X, Y, and Z by

$$\begin{array}{rccl} X: & e_i & \mapsto & \varepsilon^{a(i)} e_i\,, \\ Y: & e_j & \mapsto & \varepsilon^{b(j)} e_j\,, \\ Z: & e_\ell & \mapsto & \varepsilon^{c(k)} e_\ell\,. \end{array}$$

Then $t_A = (X\otimes Y\otimes Z)t_B + O(\varepsilon)$, i. e., t_A is a monomial degeneration of t_B.

Definition 9.2.

1. $A \subseteq I\times J\times L$ is called *tight* if there are an $r\ge 1$ and injective maps $a : I \to \mathbb{Z}^r$, $b : J \to \mathbb{Z}^r$, and $c : L \to \mathbb{Z}^r$ such that for all $(i,j,\ell) \in A$, $a(i)+b(j)+c(\ell) = 0$.
2. A set $\Delta \subseteq I\times J\times L$ is called *diagonal* if the three canonical projections $p_I : \Delta \to I$, $p_J : \Delta \to J$, and $p_L : \Delta \to L$ are injective. This means that $\Delta = \{(1,1,1),(2,2,2),\ldots\}$ up to permutations.

If a (combinatorial) diagonal is a combinatorial degeneration of some set A, then we can get a diagonal from t_A via a monomial degeneration.

Let $\mathbb{Z}_M = \mathbb{Z}/M\mathbb{Z}$. The set ψ_M below will play an important role in the following.

Lemma 9.3. *Let $M \in \mathbb{N}$. Let $\psi_M = \{(i,j,\ell) \in \mathbb{Z}_M^3 \mid i+j+\ell = 0$ in $\mathbb{Z}_M\}$. ψ_M contains a diagonal Δ with $|\Delta| \geq M/2$, which is a combinatorial degeneration of ψ_M.*

Proof. By shifting one of the indices, we can assume that $\psi_M = \{(i,j,\ell) \in \mathbb{Z}_M^3 \mid i+j+\ell+1 = 0 \mod M\}$. We write $\psi_M = A \cup B$ with

$$\begin{aligned} A &= \{(i,j,\ell) \mid i+j+\ell = M-1 \text{ in } \mathbb{Z}\}\,, \\ B &= \{(i,j,\ell) \mid i+j+\ell = 2M-1 \text{ in } \mathbb{Z}\}\,. \end{aligned}$$

$\Delta = \{(i,i,M-1-2i) \mid 0 \leq i \leq (M-1)/2\}$ is a diagonal with $|\Delta| \geq M/2$.

We define functions $a,b,c : \mathbb{Z}_M \to \mathbb{Z}$ by

$$\begin{aligned} a(i) &= 4i^2\,, \\ b(j) &= 4j^2\,, \\ c(\ell) &= -2(M-1-\ell)^2\,. \end{aligned}$$

For $(i,j,\ell) \in A$,

$$a(i)+b(j)+c(\ell) = 4i^2+4j^2-2\underbrace{(M-1-\ell)}_{i+j}{}^2 = 2i^2+2j^2-4ij = 2(i-j)^2 \geq 0\,.$$

Equality holds iff $(i,j,\ell) \in \Delta$, because if $i = j$, then $\ell = M-1-2i$ since $(i,j,\ell) \in A$.

For $(i,j,\ell) \in B$,

$$\begin{aligned} a(i)+b(j)+c(\ell) &= 4i^2+4j^2-2\underbrace{(M-1-\ell)}_{i+j-M}{}^2 \\ &= 4i^2+4j^2-2(i+j)^2+4M\underbrace{(i+j)}_{\geq M}-2M^2 \\ &\geq 2(i-j)^2+2M^2 > 0\,. \end{aligned}$$

This proves the lemma. ∎

The next lemma gives a simple lower bound on the size of a diagonal by just removing "collisions."

Lemma 9.4. *Let $\Phi \subseteq I \times J \times L$ and*

$$\Pi = \left\{\{(i,j,\ell),(i',j',\ell')\} \in \binom{\Phi}{2} \;\middle|\; i = i' \vee j = j' \vee \ell = \ell'\right\}.$$

Then there are $I' \subseteq I$, $J' \subseteq J$, and $L' \subseteq L$ such that

$$\Delta := (I' \times J' \times L') \cap \Phi$$

is a diagonal of size $\geq |\Phi| - |\Pi|$ and $\Delta \trianglelefteq \Phi$.

$$\begin{array}{ccc}
\Phi & \overset{F_w}{\to} & \Psi_M \\
\triangledown| & & \triangledown| \\
\Phi_w & \overset{F_w}{\to} & D \\
|| & & \\
\bigcup_{d \in D} \Phi_w(d) & & \\
\triangledown| & & \\
\bigcup_{d \in D} \Delta_d & &
\end{array}$$

Figure 9: The construction in the proof of Theorem 9.6.

Proof. We interpret $G = (\Phi, \Pi)$ as a graph. G has $\geq |\Phi| - |\Pi|$ connected components, since every edge in Π can connect at most two components when adding the edges of Π to the empty graph one after another. Choose one node of every connected component. These nodes form the set Δ. We set $I' = p_I(\Delta)$, and $J' = p_J(\Delta)$, and $L' = p_L(\Delta)$, where p_I, p_J, and p_L are the canonical projections.

It remains to show that Δ is a combinatorial degeneration of Φ. Define the mappings a, b and c by

$$a(i) = \begin{cases} 0 & i \in I', \\ 1 & i \in I \backslash I', \end{cases}$$
$$b(j) = \begin{cases} 0 & j \in J', \\ 1 & j \in J \backslash J', \end{cases}$$
$$c(\ell) = \begin{cases} 0 & l \in L', \\ 1 & \ell \in L \backslash L'. \end{cases}$$

By the definition of Φ and the choice of Δ,

- $\forall (i, j, \ell) \in \Delta : a(i) + b(j) + c(\ell) = 0$,
- $\forall (i, j, \ell) \in \Phi \backslash \Delta : a(i) + b(j) + c(\ell) > 0$.

This shows that Δ is a combinatorial degeneration of Φ. □

Definition 9.5. Let $\beta \in \mathbb{Z}$. $A \subseteq I \times J \times L$ is called β*-tight* if it is tight and if there are function a, b, and c like in Definition 9.2 such that in addition, $a(I), b(J), c(L) \subseteq \{-\beta, \ldots, \beta\}^r$.

The following theorem is the main result of this subsection. It provides a way to find a large diagonal in a tight set.

Theorem 9.6. *Let $\Phi \subseteq I \times J \times L$ be β-tight, $|I| \leq |J| \leq |L|$ and assume that the projections $p_I : \Phi \to I$, $p_J : \Phi \to J$, and $p_L : \Phi \to L$ are surjective. Let $c > 1$ such that*

$$\max_{i \in I} |p_I^{-1}(i)|, \ \max_{j \in J} |p_J^{-1}(j)|, \ \max_{\ell \in L} |p_L^{-1}(\ell)| \leq c \cdot \frac{|\Phi|}{|L|}.$$

Then there is a diagonal $\Delta \trianglelefteq \Phi$ *with*

$$|\Delta| \geq \frac{2}{27c\beta} \cdot |I| .$$

Proof. Let $a : I \to \{-\beta, 0, \beta\}^r$, $b : J \to \{-\beta, 0, \beta\}^r$, and $c : L \to \{-\beta, 0, \beta\}^r$ be injective such that $a(i) + b(j) + c(\ell) = 0$ for all $(i, j, \ell) \in \Phi$. Let $M \geq 2\beta + 1$ be a prime to be chosen later and let $w_1, \ldots, w_{r+3} \in \mathbb{Z}_M$. Let $w = (w_1, \ldots, w_{r+3})$. We define the following functions $A_w : I \to \mathbb{Z}_M$, $B_w : J \to \mathbb{Z}_M$, and $C_w : L \to \mathbb{Z}_M$ by

$$\begin{array}{rclllll} A_w(i) & = & \sum_{\rho=1}^r a_\rho(i) w_\rho & + w_{r+1} & - w_{r+2} & & \bmod M , \\ B_w(j) & = & \sum_{\rho=1}^r b_\rho(j) w_\rho & & + w_{r+2} & - w_{r+3} & \bmod M , \\ C_w(\ell) & = & \sum_{\rho=1}^r c_\rho(\ell) w_\rho & - w_{r+1} & & + w_{r+3} & \bmod M . \end{array}$$

It is straightforward to check that for all $(i, j, \ell) \in \Phi$, $A_w(i) + B_w(j) + C_w(\ell) = 0$, even when not computing modulo M.

Let $F_w : I \times J \times L \to \mathbb{Z}_M^3$ be defined by $(i, j, \ell) \mapsto (A_w(i), B_w(j), C_w(\ell))$. By construction,

$$F_w(\Phi) \subseteq \Psi_M = \{(x, y, z) \in \mathbb{Z}_M^3 \mid x + y + z = 0\} .$$

By Lemma 9.3, there exists a diagonal $D \trianglelefteq \Psi_M$ with $|D| \geq M/2$. Let $\Phi_w = F_w^{-1}(D) \cap \Phi$.

We claim that Φ_w is a degeneration of Φ. Since D is a degeneration of Ψ_M there are functions a_D, b_D, and c_D such that

- $\forall (i, j, \ell) \in D : a_D(i) + b_D(j) + c_D(\ell) = 0$ and
- $\forall (i, j, \ell) \in \Psi_M \setminus D : a_D(i) + b_D(j) + c_D(\ell) > 0$.

The functions $a_D \circ \hat{A}_w$, $b_D \circ \hat{B}_w$, and $c_D \circ \hat{C}_w$ prove the claim above, where $\hat{A}_w$ is defined like A_w but not reduced modulo M. $\hat{B}_w$ and $\hat{C}_w$ are defined in the same way.

For $d \in D$, set $\Phi_w(d) = F_w^{-1}(d) \cap \Phi$. Then:

$$\Phi_w = \bigcup_{d \in D} \Phi_w(d) .$$

Since D is a diagonal, the sets $p_I(\Phi_w(d))$ with $d \in D$ are pairwise disjoint. (Note that F_w operates on the three coordinates separately.) The same holds for p_J and p_L. From this it follows that if $\Delta_d \trianglelefteq \Phi_w(d)$ are diagonals, then

$$\Delta = \bigcup_{d \in D} \Delta_d$$

is a diagonal and $\Delta \trianglelefteq \Phi_w$. (We can glue the functions $p_I(\Phi_w(d)) \to \mathbb{Z}^r$ for $d \in D$ in the definition of combinatorial degeneration together, since the sets $p_I(\Phi_w(d))$ are pairwise disjoint. The same is true for the other two coordinates.) Figure 9 shows the construction we built so far.

Let

$$\Pi_w(d) = \left\{ \{(i, j, \ell), (i', j', \ell')\} \in \binom{\Phi_w(d)}{2} \;\middle|\; i = i' \vee j = j' \vee \ell = \ell' \right\} .$$

By Lemma 9.4 there exists $\Delta_d \trianglelefteq \Phi_w(d)$ with $|\Delta_d| \geq |\Phi_w(d)| - |\Pi_w(d)|$.

It remains to show the following claim:

Claim: We can choose M and $w_1,\dots,w_{r+3}$ in such a way that

$$S_w := \sum_{d\in D}(|\Phi_w(d)| - |\Pi_w(d)|) \geq \frac{2}{27c\beta}\cdot |I|\,.$$

The proof of the claim is by the probabilistic method. We choose $w_1,\dots,w_{r+3}$ uniformly at random (and M depending on $w_1,\dots,w_{r+3}$) and show that

$$E[S_w] \geq \frac{2}{27c\beta}\cdot |I|\,.$$

In particular, for at least one choice of $w_1,\dots,w_{r+3}$, S_w is large enough.

Fix $(i,j,\ell)\in I\times J\times L$. The random variables $w\to A_w(i)$, $w\to B_w(j)$, and $w\to C_w(\ell)$ are uniformly distributed and pairwise independent since $w\to(A_w(i),B_w(j))$ is surjective (as a mapping from $\mathbb{Z}_M^{r+3}\to\mathbb{Z}_M^2$). This is due to the fact that w_{r+1} only appears in A_w and w_{r+3} only appears in B_w. The same is true for the other two pairs.

Furthermore $A_w(i), A_w(i')$ and $C_w(\ell)$ are pairwise independent for $i\neq i'$, since

$$w\to(A_w(i),A_w(i'),C_w(\ell))$$

is surjective because

$$\begin{pmatrix} a_1(i) & \dots & a_r(i) & 1 & -1 & 0\\ a_1(i') & \dots & a_r(i') & 1 & -1 & 0\\ c_1(\ell) & \dots & c_r(\ell) & -1 & 0 & 1\end{pmatrix}$$

has rank three over $\mathbb{Z}_M$. If one writes the zero vector as a linear combination of these three rows, then the coefficient of the last row will be zero because of the 1 in the last column of the matrix. a is injective as a mapping to $\mathbb{Z}^r$. But since $M\geq 2\beta+1$, it is also injective as a mapping to $\mathbb{Z}_M^r$. Therefore, the first two rows are not identical, since $i\neq i'$. Thus the coefficients of the first two rows must be zero, too.

The expected value of $|\Phi_w(d)|$ for $d=(x,y,z)$ is the probability that we hit (x,y,z), i. e.,

$$\begin{aligned} E[|\Phi_w(d)|] &= \sum_{(i,j,\ell)\in\Phi}\Pr_w[A_w(i)=x, B_w(j)=y, C_w(\ell)=z]\\ &= \sum_{(i,j,\ell)\in\Phi}\Pr_w[A_w(i)=x, B_w(j)=y]\\ &= |\Phi|\cdot\frac{1}{M^2}.\end{aligned}$$

We can drop the event $C_w(\ell)=z$, since it is implied by the other two events for $(i,j,\ell)\in\Phi$ and $(x,y,z)\in\Psi_M$.

To estimate the expected value of $|\Pi_w(d)|$, we decompose it into three sets. Let

$$\begin{aligned} U_w(d) &:= \left\{\{(i,j,\ell),(i',j',\ell')\}\in\binom{\Phi_w(d)}{2}\;\middle|\;\ell=\ell'\right\}\\ &= \bigcup_{\ell\in L}\left\{\{(i,j,\ell),(i',j',\ell')\}\in\binom{p_L^{-1}(\ell)}{2}\;\middle|\;A_w(i)=x=A_w(i'), C_w(\ell)=z\right\}.\end{aligned}$$

Note that as above, $A_w(i) = x = A_w(i')$ and $C_w(\ell) = z$ imply $B_w(j) = y = B_w(j')$. As we have seen, $A_w(i)$, $A_w(i')$, and $C_w(\ell)$ are independent. Therefore,

$$\begin{aligned} E(|U_w(d)|) &= \sum_{\ell \in L} \frac{|p_L^{-1}(\ell)|(|p_L^{-1}(\ell)| - 1)}{2} M^{-3} \\ &\le \frac{1}{2M^3} \sum_{\ell \in L} |p_L^{-1}(\ell)|^2 \\ &\le \frac{c|\Phi|^2}{2M^3|L|} . \end{aligned}$$

For the last inequality, we used that $\sum_{\ell \in L} |p_L^{-1}(\ell)| = |\Phi|$ and the assumption that $|p_L^{-1}(\ell)| \le c|\Phi|/|L|$. We do the same for the other two coordinates and get

$$E[|\Pi_w(d)|] \le \frac{3c|\Phi|^2}{2M^3|I|} .$$

Recall that $|I| \le |J|, |L|$.

Now we can finish the proof of the claim:

$$\begin{aligned} E(S_w) &= \sum_{d \in D} (|\Phi_w(d)| - |\Pi_w(d)|) \\ &\ge |D| \cdot \left(\frac{|\Phi|}{M^2} - \frac{3c|\Phi|^2}{2M^3|I|} \right) \\ &\ge \frac{|I|}{2c} \left(\frac{c|\Phi|}{M|I|} - \frac{3}{2} \cdot \left(\frac{c|\Phi|}{M|I|} \right)^2 \right) . \end{aligned}$$

Now we choose the prime M such that

$$\frac{9}{4} \cdot \frac{\beta c|\Phi|}{|I|} \le M \le \frac{9}{2} \cdot \frac{\beta c|\Phi|}{|I|} .$$

Such an M exists by Bertrand's postulate. Since $|I| \le |\Phi|$, $M \ge 2\beta + 1$, as required. It is easy to check that with this choice of M,

$$E(S_w) \ge \frac{|I|}{2c} \cdot \frac{4}{27\beta} = \frac{2|I|}{27c\beta} ,$$

and we are done. □

9.2 First construction

The support Φ of CW with respect to $\mathcal{D}$ is

$$\{(1,1,0), (1,0,1), (0,1,1)\} \subseteq \{0,1\}^3 .$$

It is obviously tight, since it fulfills $i+j+\ell=2$. Take the Nth tensor power $\mathrm{CW}^{\otimes N}$. All inner tensors of $\mathrm{CW}^{\otimes N}$ with respect to $\mathcal{D}^{\otimes N}$ are tensors $\langle x,y,z\rangle$ with $xyz=q^N$. By Theorem 9.6, the support Φ^N of $\mathrm{CW}^{\otimes N}$ contains a diagonal of size $2|I|^N/(27c)$ where c is chosen such that

$$|p_{I^N}^{-1}(i)| \le c\frac{|\Phi|^N}{|I|^N}.$$

Since $p_I^{-1}(1)=\{(1,1,0),(1,0,1)\}$, $|p_{I^N}^{-1}(1,\dots,1)|=2^N$. (We only need to check this for I^N since the situation is completely symmetric.) Therefore,

$$c \ge \frac{|I|^N 2^N}{|\Phi|^N} = \frac{4^N}{3^N}.$$

Thus, we get a diagonal of size $(2/27)\cdot(3/2)^N$. We now can apply the τ-Theorem (recall that combinatorial degenerations give rise to monomial degenerations) and get

$$\frac{2}{27}\cdot\left(\frac{3}{2}\right)^N q^{\omega/3\cdot N} \le (q+2)^N.$$

Taking Nth roots and letting N go to infinity, we get

$$\omega \le 3\log_q\left(\frac{2(q+2)}{3}\right).$$

For $q=18$, this gives $\omega \le 2.69$. 2.69? Really, 2.69!

So what went wrong? It turns out, that it is better to restrict Φ^N. Let I' be the set of all vectors in I^N with $2N/3$ 1's. We assume that N is divisible by 3. We define J' and L' in the same way. Let $\Phi'=\Phi^N\cap I'\times J'\times L'$. Φ' is nonempty, since the product containing $N/3$ factors of each of the 3 elements in Φ is in $I'\cap J'\cap L'$.

Now, $|p_{I'}^{-1}(i)|$ have the same size for all i by construction. Then trivially,

$$|p_{I'}^{-1}(i)| \le \frac{|\Phi'|}{|I'|},$$

so we can choose $c=1$ in Theorem 9.6. We get a diagonal of size $\frac{2}{27}\binom{N}{2N/3}$. We apply the τ-theorem once again and get this time

$$\frac{2}{27}\cdot\binom{N}{2N/3}q^{\omega/3\cdot N} \le (q+2)^N.$$

By Stirling's formula,

$$\frac{1}{N}\ln\binom{N}{2N/3} \to -\frac{2}{3}\ln\frac{2}{3}-\frac{1}{3}\ln\frac{1}{3} = -\frac{2}{3}\ln(2)+\ln 3$$

for $N\to\infty$. Therefore, we get

$$\omega \le 3\cdot\log_q\left(\frac{2^{2/3}(q+2)}{3}\right) = \log_q\left(\frac{4(q+2)^3}{27}\right).$$

For $q=8$, we obtain the following result.

Corollary 9.7 (Coppersmith & Winograd). $\omega \leq 2.41$.

It can be shown that $\underline{R}(\mathrm{CW}) = q+2$. So is this the end of this approach? Note that in the above calculation, we always compute a huge power $\mathrm{CW}^{\otimes N}$. The format of this tensor is $(q+1)^N \times (q+1)^N \times (q+1)^N$. So it could be the case that $\underline{R}(\mathrm{CW}^{\otimes N}) = (q+1)^N$. The *asymptotic rank* $\underset{\sim}{R}(t)$ of a tensor t is defined as

$$\underset{\sim}{R}(t) := \lim_{N\to\infty} R(t^{\otimes N})^{1/N}.$$

This is well-defined. All the bounds that we have shown so far are still valid if we replace border rank by asymptotic rank. If $\underset{\sim}{R}(\mathrm{CW}) = q+1$, then $\omega = 2$ would follow (from the construction above for $q=2$).

Problem 9.8. What is $\underset{\sim}{R}(\mathrm{CW})$? Even simpler: Is $\underline{R}(\mathrm{CW}^{\otimes 2}) < (q+2)^2$?

9.3 Main theorem

Next we prove a general theorem, that formalizes the method used to prove Corollary 9.7. We will work with arbitrary probability distributions on the support, since in this case, we can even handle the case when the inner tensors are matrix tensors of different sizes.

Let $P : I \to [0;1]$ be a probability distribution. The entropy $H(P)$ of P is defined as

$$H(P) := -\sum_{i\in I : P(i)>0} P(i)\cdot \ln P(i).$$

Fact 9.9. *For all $\mu : I \to \mathbb{N}$ with $\sum_{i\in I}\mu(i) = N$,*

$$\left|\frac{1}{N}\cdot \ln\binom{N}{\mu} - H\left(\frac{\mu}{N}\right)\right| \to 0.$$

The fact can be easily shown using Stirling's formula.

Let $P : I\times J\times L \to [0;1]$ be a probability distribution. Then

$$P_1(i) := \sum_{(j,\ell)\in J\times L} P(i,j,\ell)$$

is a probability distribution, the first marginal distribution. In the same way, we define $P_2(j)$ and $P_3(\ell)$.

Theorem 9.10 (Coppersmith & Winograd). *Let $\mathcal{D}$ be a decomposition of a tensor $t \in K^{k\times m\times n}$ with sets $I_1,\dots,I_p$, $J_1,\dots,J_q$, and $L_1,\dots,L_s$ such that*

1. $\mathrm{supp}_{\mathcal{D}}\, t$ *is tight,*
2. t_{I_i,J_j,L_ℓ} *is a matrix tensor for all* $(i,j,\ell)\in \mathrm{supp}_{\mathcal{D}}\, t$.

Then

$$\min_{1\leq m\leq 3} H(P_m) + \omega\cdot \sum_{(i,j,\ell)\in\mathrm{supp}_{\mathcal{D}}\, t} P(i,j,\ell)\cdot \ln(\zeta(t_{I_i,J_j,L_\ell})) \leq \ln \underline{R}(t)$$

for all probability distributions P on $\mathrm{supp}_{\mathcal{D}}\, t$, where $\zeta(\langle x,y,z\rangle) = (xyz)^{1/3}$.

Proof. Assume that $\mathrm{supp}_{\mathcal{D}}\,t$ is β-tight. We choose a function $Q : \mathrm{supp}_{\mathcal{D}}\,t \to \mathbb{N}$ and let

$$N = \sum_{(i,j,\ell)\in\mathrm{supp}_{\mathcal{D}}\,t} Q(i,j,\ell)\,.$$

(Think of Q being a discretization of our probability distribution P.) Let

$$\mu(i) = \sum_{j,\ell} Q(i,j,\ell)\,.$$

We define $\nu(j), \pi(\ell)$ analogously. Obviously $\sum \mu(i) = N$. We say that $x = (x_1,\dots,x_N) \in I^N$ has distribution μ if for all $i \in I$, i appears in exactly $\mu(i)$ positions.

It is easy to check that the support of $t^{\otimes N}$ with respect to the decomposition $\mathcal{D}^{\otimes N}$ is again β-tight. Let

$$\begin{aligned}
I_\mu &:= \{x \in I^N \mid x \text{ has distribution } \mu\}\,,\\
J_\nu &:= \{y \in J^N \mid y \text{ has distribution } \nu\}\,,\\
L_\pi &:= \{z \in L^N \mid z \text{ has distribution } \pi\}\,,\\
\Phi &:= I_\nu \times J_\nu \times L_\pi \cap (\mathrm{supp}_{\mathcal{D}}\,t)^N\,.
\end{aligned}$$

We have $|I_\mu| = \binom{N}{\mu}$, $|J_\nu| = \binom{N}{\nu}$, and $|L_\pi| = \binom{N}{\pi}$. Furthermore, Φ is not empty. The projection $p_1 : \Phi \to I_\mu$ is surjective with $|p_1^{-1}(i)| = |\Phi|/|I_\mu|$. All fibers $p_1^{-1}(i)$ have the same size, namely $|\Phi|/|I_\mu|$. The same holds for J_ν and L_π.

What do the inner tensors of $t^{\otimes N}$ with respect to the decomposition $t^{\otimes N}$ look like? They are tensor products of the inner tensors of t, i. e., matrix tensors itself. Take $(x,y,z) \in \Phi$. The inner tensor corresponding to (x,y,z) is

$$t^{\otimes N}_{I_{x_1}\times\cdots\times I_{x_N}, J_{y_1}\times\cdots\times J_{y_N}, L_{z_1}\times\cdots\times J_{z_N}} = \bigotimes_{s=1}^{N} t_{I_{x_s},J_{y_s},L_{z_s}}\,.$$

Assume that $t_{I_i,J_j,L_\ell} \in U_i \otimes V_j \otimes W_\ell$ with $\dim U_i = k_i$, $\dim V_j = m_j$, and $\dim W_\ell = n_\ell$. Then $\zeta(t_{I_i,J_j,L_\ell}) = (k_i m_j n_\ell)^{1/6}$. Thus,

$$\begin{aligned}
\zeta(t^{\otimes N}_{I_{x_1}\times\cdots\times I_{x_N}, J_{y_1}\times\cdots\times y_{x_N}, L_{z_1}\times\cdots\times L_{z_N}}) &= \prod_{s=1}^{N} (k_{x_s} m_{y_s} n_{z_s})^{1/6}\\
&= \prod_{i\in I} k_i^{\mu(i)/6} \prod_{j\in J}]m_j^{\nu(j)/6} \prod_{\ell L} n_\ell^{\pi(\ell)/6}\\
&= \prod_{(i,j,\ell)\in\mathrm{supp}_{\mathcal{D}}\,t} (k_i m_j n_\ell)^{Q(i,j,\ell)/6}\\
&= \prod_{(i,j,\ell)\in\mathrm{supp}_{\mathcal{D}}\,t} \zeta(t_{I_i,J_j,L_\ell})^{Q(i,j,\ell)}\,.
\end{aligned}$$

This means that all inner tensors of $t^{\otimes N}$ restricted to Φ have the same ζ-value. This is another reason for restricting the situation to the invariant sets I_μ, J_ν, and L_π.

Next, we apply Theorem 9.6 to the β-tight set $\Phi \subseteq I_\mu \times J_\nu \times L_\pi$. We get a diagonal Δ of size

$$|\Delta| \geq \frac{2}{27\beta} \min\{|I_\mu|, |J_\nu|, |L_\pi|\}.$$

Note that we can choose the constant $c = 1$. Δ is a degeneration of $\Phi \trianglelefteq (\operatorname{supp}_{\mathcal{D}} t)^N$. Therefore,

$$\bigoplus_{(x,y,z)\in\Delta} t^{\otimes N}_{I_{x_1}\times\cdots\times I_{x_N}, J_{y_1}\times\cdots\times J_{y_N}, L_{z_1}\times\cdots\times L_{z_N}} \trianglelefteq t^{\otimes N}.$$

We apply the τ-theorem and obtain

$$|\Delta| \prod_{(i,j,\ell)\in\operatorname{supp}_{\mathcal{D}} t} \zeta(t^{Q(i,j,\ell)}_{I_i,J_j,L_\ell})^\omega \leq \underline{R}(t^{\otimes N}) \leq \underline{R}(t)^N.$$

Taking logarithms, we get

$$\frac{1}{N}\ln|\Delta| + \omega \sum_{(i,j,\ell)\in\operatorname{supp}_{\mathcal{D}} t} \frac{1}{N} Q(i,j,\ell) \ln\zeta(t_{L_i,J_j,L_\ell}) \leq \underline{R}(t).$$

Now we approximate the given probability distribution P by the function Q such that

$$|P(i,j,\ell) - \frac{1}{N}Q(i,j,\ell)| \leq \varepsilon.$$

ε solely depends on N and goes to 0 as N goes to ∞.

By Fact 9.9 we can approximate $\frac{1}{N}\ln|\Delta|$ by $\min_{1\leq m\leq 3} H(P_m)$. Therefore, we get

$$\min_{1\leq m\leq 3} H(P_m) + \omega \sum_{(i,j,\ell)\in\operatorname{supp}_{\mathcal{D}} t} P(i,j,\ell)\log\zeta(t_{I_i,J_j,L_\ell}) \leq \ln\underline{R}(t) + C\cdot\varepsilon$$

for some constant C. The result follows by letting ε tend to zero. □

Remark 9.11. The theorem above generalizes Strassen's laser method, since matrix tensors are tight.

Consider the following enhanced Coppersmith and Winograd tensor

$$\mathrm{CW}_+ = \sum_{i=1}^{q}(\underbrace{e_i\otimes e_0\otimes e_i}_{\langle q,1,1\rangle} + \underbrace{e_0\otimes e_i\otimes e_i}_{\langle 1,1,q\rangle} + \underbrace{e_i\otimes e_i\otimes e_0}_{\langle 1,q,1\rangle}) + e_{q+1}\otimes e_0\otimes e_0 + e_0\otimes e_{q+1}\otimes e_0 + e_0\otimes e_0\otimes e_{q+1}.$$

Astonishingly, this larger tensor has border rank $q+2$, too:

$$\begin{aligned}
\varepsilon^5\,\mathrm{CW}_+ = &\sum_{i=1}^{q}\varepsilon\cdot(e_0+\varepsilon^2 e_i)\otimes(e_0+\varepsilon^2 e_i)\otimes(e_0+\varepsilon^2 e_i)\\
&-(e_0+\varepsilon^3\sum_{i=1}^{q}e_i)\otimes(e_0+\varepsilon^3\sum_{i=1}^{q}e_i)\otimes(e_0+\varepsilon^3\sum_{i=1}^{q}e_i)\\
&+(1-q\varepsilon)\cdot(e_0+\varepsilon^5 e_{q+1})\otimes(e_0+\varepsilon^5 e_{q+1})\otimes(e_0+\varepsilon^5 e_{q+1})\\
&+O(\varepsilon^6).
\end{aligned}$$

Thus, $\underline{R}(\mathrm{CW}_+) \le q+2$. We define a decomposition $\mathcal{D}$ as follows:

$$\begin{array}{ccccccl}
\underbrace{\{0\}}_{I_0} & \dot\cup & \underbrace{\{1,\dots,q\}}_{I_1} & \dot\cup & \underbrace{\{q+1\}}_{I_2} & = \{0,\dots,q+1\}, \\
\underbrace{\{0\}}_{J_0} & \dot\cup & \underbrace{\{1,\dots,q\}}_{J_1} & \dot\cup & \underbrace{\{q+1\}}_{J_2} & = \{0,\dots,q+1\}, \\
\underbrace{\{0\}}_{L_0} & \dot\cup & \underbrace{\{1,\dots,q\}}_{L_1} & \dot\cup & \underbrace{\{q+1\}}_{L_2} & = \{0,\dots,q+1\}.
\end{array}$$

With respect to $\mathcal{D}$, we have

$$\mathrm{CW}_{\mathcal{D}} = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 1 & \\ 1 & & \end{pmatrix},$$

$$\mathrm{CW}_{I_i,J_j,L_\ell} \in \begin{cases} \{\langle 1,1,q\rangle, \langle q,1,1\rangle, \langle 1,q,1\rangle\} & \text{if } (i,j,\ell) \in \{(1,1,0),(1,0,1),(0,1,1)\}, \\ \{\langle 1,1,1\rangle\} & \text{if } (i,j,\ell) \in \{(0,0,2),(0,2,0),(2,0,0)\}. \end{cases}$$

The support of t with respect to $\mathcal{D}$ is tight, since it is given by $i+j+\ell = 2$.

To apply Theorem 9.10, we distribute the probability $\beta/3$ over the "small" products and $(1-\beta/3)$ over the "large" products uniformly. Then we get:

$$H\left(1-\frac{\beta}{3}+2\frac{\beta}{3}, 2\frac{1-\beta}{3}, \frac{\beta}{3}\right) + \frac{\omega}{3}\cdot(\beta \log 1 + (1-\beta)\cdot \log q) \le \log(q+2).$$

Setting $q = 6$ and $\beta = 0.048$ yields $\omega \le 2.39$.

Corollary 9.12 (Coppersmith & Winograd)**.** $\omega \le 2.39$.

9.4 Further improvements

Instead of starting with CW_+ we can also start with $\mathrm{CW}_+^{\otimes 2}$ as our starting tensor. While this does not give anything new when we take $\mathcal{D}^{\otimes 2}$ as the decomposition, we can gain something by choosing a new decomposition. The elements of $\mathrm{supp}_{\mathcal{D}^{\otimes}}(\mathrm{CW}_+^{\otimes 2})$ are contained in $\{0,1,2\}^2 \times \{0,1,2\}^2 \times \{0,1,2\}^2$. Coppersmith and Winograd build a new decomposition with support $\subseteq \{0,\dots,4\}^3$ by identifying $((i,i'),(j,j'),(\ell,\ell'))$ with $(i+i', j+j', \ell+\ell')$. This gives a coarser outer structure. Tensors of the old inner structure are now grouped together. Funnily, the new inner tensors are still matrix tensors with one exception. To analyse this exception, Coppersmith an Winograd introduced the *value* of a tensor t: Suppose that $\omega = 3\tau$ is the exponent of matrix multiplication. If $\bigoplus_{i=1}^n \langle k_i, m_i, n_i\rangle \trianglelefteq t^{\otimes N}$, then the value of t is at least $(\sum_{i=1}^n (k_i m_i n_i)^\tau)^{1/N}$. Intuitively, the value is the contribution of t to the τ-theorem, when we construct the diagonal in the proof of Theorem 9.10. Theorem 9.10 can be generalized to this more general situation.

Coppersmith and Winograd do the analysis for $\mathrm{CW}_+^{\otimes 2}$. Andrew Stothers [30] (see also [14]) does it for $\mathrm{CW}_+^{\otimes 4}$ ($\mathrm{CW}_+^{\otimes 3}$ does not seem to give any improvement) and Virginia Vassilevska-Williams [35] for $\mathrm{CW}_+^{\otimes 8}$ with the help of a computer program. We get the upper bounds 2.376, 2.3737, and 2.3727 for ω, respectively.

10 Group-theoretic approach

While the bounds on ω mentioned in the previous section are the best currently known, we present an interesting approach due to Cohn and Umans [10]. We follow their exposition quite closely.

Let G be a finite group and $\mathbb{C}[G]$ denote the *group algebra* over $\mathbb{C}$. The elements of $\mathbb{C}[G]$ are formal sums of the form

$$\sum_{g \in G} a_g g \qquad \text{with } a_g \in \mathbb{C} \text{ for all } g \in G.$$

Addition and scalar multiplication is defined component-wisely. Multiplication is defined such that it distributes over addition:

$$\left(\sum_{g \in G} a_g g\right)\left(\sum_{h \in H} b_g g\right) = \sum_{f \in G} \sum_{\substack{g,h \in G: \\ g+h=f}} a_g b_h f\,.$$

Let C_n be the cyclic group of order n and g be a generator. The product of two elements $\sum_{i=1}^{n-1} a_i g^i$ and $\sum_{i=1}^{n-1} b_i g^i$ in $\mathbb{C}[C_n]$ is the cyclic convolution

$$\sum_{i=0}^{n-1} \sum_{j,k: j+k=i \mod n} a_j b_k g^i\,.$$

Wedderburn's theorem for group algebras of finite groups states that every group algebra $\mathbb{C}[G]$ of a finite group G is isomorphic to the direct product of square matrices over $\mathbb{C}$:

$$\mathbb{C}[G] \cong \mathbb{C}^{d_1 \times d_1} \times \cdots \times \mathbb{C}^{d_k \times d_k}\,.$$

The numbers $d_1, \ldots, d_k$ are called the *character degrees*. k is the number of conjugacy classes. By comparing dimensions, it follows that $|G| = d_1^2 + \cdots + d_k^2$. See [18] for an introduction to representation theory. For the cyclic group of order n, $\mathbb{C}[C_n] \cong \mathbb{C}^n$ because $\mathbb{C}[C_n]$ is commutative. Since on the other hand, $\mathbb{C}[C_n] \cong \mathbb{C}[X]/(X^n - 1)$—in both algebras, multiplication is cyclic convolution—multiplication of polynomials of degree $\le (n-1)/2$ can be performed by a cyclic convolution which in turn can performed by n pointwise multiplications. Since an isomorphism $\mathbb{C}[C_n] \to \mathbb{C}^n$ is a linear transformation and hence, can be performed with scalar multiplications, this shows that the rank of multiplication of polynomials of degree $\le (n-1)/2$ is bounded by n.

An isomorphism $\mathbb{C}[G] \to \mathbb{C}^{d_1 \times d_1} \times \cdots \times \mathbb{C}^{d_k \times d_k}$ is called a *discrete Fourier transform*. For the cyclic group C_n of order n, there are discrete Fourier transforms what can be implemented fast, even under the total cost measure. Using one of the fast Fourier transform algorithms, polynomial multiplication of polynomials of degree d can be done with $O(d \log d)$ total operations. Also other group algebras allow fast Fourier transformations, see [3].[16]

[16] But note that in our setting, discrete Fourier transforms are free of cost, since they are linear transformations. So there is no need for fast Fourier transforms for fast matrix multiplication But there is no cheating involved here, since it does not matter for the exponent whether we only count all operations or only bilinear multiplications.

10.1 Matrix multiplication via groups

In the light of this success for polynomial multiplication, it is now natural to try the same approach for matrix multiplication. For a subset S of a finite group, let

$$Q(S) = \{st^{-1} \mid s,t \in S\}$$

denote the right quotient of S. Note that if S is a subgroup, then $Q(S) = S$.

Definition 10.1. A group G *realizes* $\langle n_1,n_2,n_3\rangle$ if there are subsets $S_1,S_2,S_3 \subseteq G$ such that $|S_i| = n_i$ for $1 \le i \le 3$ and for all $q_i \in Q(S_i)$, $1 \le i \le 3$,

$$q_1q_2q_3 = 1 \qquad \text{implies} \qquad q_1 = q_2 = q_3 = 1\,.$$

We call this condition on S_1,S_2,S_3 the *triple product property*.

As a first example, consider the product of cyclic groups $C_k \times C_m \times C_n$. This group realizes $\langle k,m,n\rangle$ through the subgroups $C_k \times \{1\} \times \{1\}$, $\{1\} \times C_m \times \{1\}$, and $\{1\} \times \{1\} \times C_n$.

It is rather easy to verify that when G realizes $\langle n_1,n_2,n_3\rangle$, then it realizes $\left\langle n_{\pi(1)},n_{\pi(2)},n_{\pi(3)}\right\rangle$ for every $\pi \in S_3$, too (see [10, Lem. 2.1] for a proof).

Lemma 10.2. *Let G and G' be groups. If G realizes $\langle k,m,n\rangle$ and G' realizes $\langle k',m',n'\rangle$, then $G \times G'$ realizes $\langle kk',mm',nn'\rangle$.*

Proof. Assume that G realizes $\langle k,m,n\rangle$ through S_1, S_2, and S_3 and G' realizes $\langle k',m',n'\rangle$ through T_1, T_2, and T_3.

$G \times G'$ realizes $\langle kk',mm',nn'\rangle$ through $S_1 \times T_1$, $S_2 \times T_2$, and $S_3 \times T_3$. To prove this, we need to verify that for $s_i,s_i' \in S_i$ and $t_i,t_i' \in T_i$,

$$(s_1',t_1')(s_1,t_1)^{-1}(s_2',t_2')(s_2,t_2)^{-1}(s_3',t_3')(s_3,t_3)^{-1} = 1 \tag{10.1}$$

implies $(s_i',t_i')(s_i,t_i)^{-1} = 1$ for all i. (10.1) is equivalent to

$$\begin{aligned} s_1's_1^{-1}s_2's_2^{-1}s_3's_3^{-1} &= 1\,, \\ t_1't_1^{-1}t_2't_2^{-1}t_3't_3^{-1} &= 1\,. \end{aligned}$$

By the triple product property, $s_i's_i^{-1} = 1$ and $t_i't_i^{-1} = 1$ for all i. Thus

$$(s_i',t_i')(s_i,t_i)^{-1} = (s_i',t_i')(s_i^{-1},t_i^{-1}) = (1,1)\,,$$

as desired. □

Multiplication in a group algebra $\mathbb{C}[G]$ is a bilinear mapping. By abuse of notation, we call the tensor of this mapping $\mathbb{C}[G]$ again. We say that a tensor s is a *restriction* of a tensor t if $(A \otimes B \otimes C)s = t$. We write $s \le t$ in this case. If s is a restriction of t, then it is a degeneration of t, too.

Theorem 10.3. *Let G be a finite group. If G realizes $\langle k,m,n\rangle$, then $\langle k,m,n\rangle \leq \mathbb{C}[G]$. In particular, $R(\langle k,m,n\rangle) \leq R(\mathbb{C}[G])$.*

Proof. Assume that G realizes $\langle k,m,n\rangle$ through S, T, and U. Let $A \in \mathbb{C}^{k\times m}$ and $B \in \mathbb{C}^{m\times n}$. We index the rows and columns of A with elements from S and T, respectively. In the same way, we index the rows and columns of B with T and U and the rows and columns of the result AB by S and U, respectively.

We have

$$\left(\sum_{s\in S,t'\in T} A_{s,t'}s^{-1}t'\right)\left(\sum_{t\in T,u'\in U} B_{t,u'}t^{-1}u'\right) = \sum_{s\in S,u'\in U}\left(\sum_{t,t'\in T} A_{s,t'}B_{t,u'}\right)s^{-1}t't^{-1}u'$$
$$= \sum_{s'\in S,u\in U} (AB)_{s,u'}s'^{-1}u\,,$$

since $(s^{-1}t')(t^{-1}u') = s'^{-1}u$ is equivalent to $s's^{-1}t't^{-1}u'u^{-1} = 1$. The triple product property now yields $s = s'$, $t = t'$, and $u = u'$. □

The group algebra $\mathbb{C}[G]$ is isomorphic to a product of matrix algebras. Therefore, when G realizes $\langle k,m,n\rangle$, Theorem 10.3 reduces the multiplication of $k\times m$-matrices with $m\times n$-matrices to many small matrix multiplications.

10.2 The pseudo-exponent

The pseudo-exponent of a group measures the quality of the embedding provided by Theorem 10.3.

Definition 10.4. The *pseudo-exponent* $\alpha(G)$ of a nontrivial finite group G is

$$\alpha(G) = \min\left\{\frac{3\log|G|}{\log kmn} \;\middle|\; G \text{ realizes } \langle k,m,n\rangle,\ \max\{k,m,n\} > 1\right\}.$$

The pseudo-exponent of the trivial group is 3.

Lemma 10.5. *Let G be a finite group.*

1. $2 < \alpha(G) \leq 3$.
2. *If G is abelian, then* $\alpha(G) = 3$.

Proof. The upper bound of 3 follows directly from the observation above that every group realizes $\langle |G|,1,1\rangle$. Note that any group G realizes $\langle |G|,1,1\rangle$ by choosing subgroups $H_1 = G$, $H_2 = \{1\}$, and $H_3 = \{1\}$. For the lower bound, suppose that G realizes $\langle k,m,n\rangle$ through sets S, T, and U. The map $Q(S)\times Q(T) \to G$ defined by $(x,y) \mapsto xy$ is injective. Its image intersects $Q(U)$ only in $\{1\}$. This follows from the definition of "realizes": Assume that $st = u$ with $s \in Q(S)$, $t \in Q(T)$, and $u \in Q(U)$. Then $s = t = u = 1$. Therefore,

$$|G| \geq |Q(S)\times Q(T)| \geq km$$

where the last inequality is strict if $|U| = n > 1$. The same is true for the pairs T,U and S,U. Thus, $|G|^3 > (kmn)^2$, which implies $\alpha(G) > 2$.

If G is abelian, then the map $Q(S) \times Q(T) \times Q(U) \to G$ given by $(x,y,z) \mapsto xyz$ is injective, because $x'y'z' = xyz$ implies $x^{-1}x'y^{-1}y'z^{-1}z' = 1$. Now, injectivity follows from the definition of "realizes." Therefore, $|G| \geq kmn$, if G is abelian. □

Example 10.6. The symmetric group $S_{\binom{n}{2}}$ has pseudo-exponent $2 + O(1/\log n)$. To see this, we think of $S_{\binom{n}{2}}$ acting on triples (a,b,c) with $a+b+c = n-1$ and $a,b,c \geq 0$. Let H_i be the subgroup of $S_{\binom{n}{2}}$ that fixes the ith coordinate. We claim that $S_{\binom{n}{2}}$ realizes $\langle N,N,N\rangle$ via H_1,H_2,H_3 where $N = |H_i| = 1!2!\cdots n!$. If this were true, then

$$\alpha(S_{\binom{n}{2}}) = \frac{\log \binom{n}{2}!}{\log N} = 2 + O\left(\frac{1}{\log n}\right).$$

So it remains to show that H_1,H_2,H_3 satisfy the triple product property: Let $h_1h_2h_3 = 1$. Order the triples (a,b,c) lexicographically. Let (a,b,c) be the smallest triple such that $h_i(a,b,c) \neq (a,b,c)$ for some i. Since (a,b,c) is the smallest such triple, $h_3(a,b,c) = (a+j,b-j,c)$ for some $j \geq 0$. (Note that h_i fixes (a,b,c) iff h_i^{-1} fixes (a,b,c).) Next, $h_2(a+j,b-j,c) = (a+j+k,b-j,c-k)$ for some k. Since h_1 fixes the first coordinate, we have $j+k = 0$. Since (a,b,c) was the smallest triple, h_1 fixes $(a,b-j,c+j)$, thus $j = 0$. Therefore, $h_i(a,b,c) = (a,b,c)$, a contradiction. Hence, $h_i = 1$ for all i.

10.3 Bounds on ω

Unfortunately, if a group has pseudo exponent close to 2 it does not mean that we get a good bound on ω from it. The group needs to have small character degrees in addition.

Theorem 10.7. *Suppose G has pseudo exponent α and its character degrees are $d_1,\dots,d_t$. Then*

$$|G|^{\omega/\alpha} \leq \sum_{i=1}^{t} d_i^{\omega}.$$

Proof. By the definition of pseudo exponent, there are k, m, and n such that G realizes $\langle k,m,n\rangle$ with $kmn = |G|^{3/\alpha}$. By Theorem 10.3,

$$\langle k,m,n\rangle \leq \mathbb{C}[G] \cong \bigoplus_{i=1}^{t} \langle d_i,d_i,d_i\rangle.$$

If we take the ℓth tensor power of this, we get

$$\left\langle k^\ell,m^\ell,n^\ell\right\rangle \leq \left(\bigoplus_{i=1}^{t} \langle d_i,d_i,d_i\rangle\right)^{\otimes \ell} = \bigoplus_{i_1,\dots,i_\ell=1}^{t} \langle d_{i_1}\cdots d_{i_\ell}, d_{i_1}\cdots d_{i_\ell}, d_{i_1}\cdots d_{i_\ell}\rangle.$$

Taking ranks on both sides, we get

$$R(\left\langle k^\ell,m^\ell,n^\ell\right\rangle) \leq c\cdot\left(\sum_{i=1}^{t} d_i^{\omega+\varepsilon}\right)^{\ell},$$

where $\varepsilon > 0$ and c is a constant such that $R(\langle s,s,s\rangle) \le c \cdot s^{\omega+\varepsilon}$ for all s. Since $(xyz)^{\omega/3} \le R(\langle x,y,z\rangle)$ for all x,y,z, we get by taking ℓth roots

$$|G|^{\omega/\alpha} = (kmn)^{\omega/3} \le \sum_{i=1}^{t} d_i^{\omega+\varepsilon}\,.$$

Since $\varepsilon > 0$ was arbitrary, the claim of the theorem follows. □

Corollary 10.8. *Suppose G has pseudo exponent α and its largest character degree is $d_{\max}$. Then $|G|^{\omega/\alpha} \le |G| d_{\max}^{\omega-2}$.*

Proof. Use $\sum_{i=1}^{t} d_i^2 = |G|$. □

10.4 Applications

So is there a group that gives a nontrivial bound on the exponent? While in the first paper, no such example was given, Cohn et al. [9] in a second paper gave several such examples. It is also possible to match the upper bound by Coppersmith and Winograd within this group theoretic framework. To this aim, they generalize the triple product property to a simultaneous triple product property. It is quite easy to prove analogues of Lemma 10.2, Theorem 10.3, and of Theorem 10.7 with matrix tensors replaced by sums of matrix tensors. The interested reader is referred to [9].

Furthermore, Cohn et al. [9] make two conjectures, both of which would imply $\omega = 2$. One of them, however, contradicts a variant of the sunflower conjecture [2].

Definition 10.9. Let G and H be two groups, with a left action of G on H. The *semidirect product $H \rtimes G$* is the set $H \times G$ with the multiplication law

$$(h_1,g_1)(h_2,g_2) = (h_1(g_1 \cdot h_2), g_1 g_2)$$

where $g_1 \cdot h_2$ denotes the action of g_1 on h_2.

Example 10.10. Let C_n be the cyclic group of order n and set $H = C_n^3$. Let $G = H^2 \rtimes C_2$ where C_2 acts on H^2 by switching the two factors. Let z be the generator of C_2. We write elements of G as $(a,b)z^i$ with $a,b \in H$ and $i \in \{0,1\}$. Let H_1,H_2,H_3 be the three factors of H viewed as subgroups. We define subsets

$$S_i = \{(a,b)z^j \mid a \in H_i \setminus \{1\},\ \ b \in H_{i+1},\ \ j \in \{0,1\}\}\,,$$

where the index of H_{i+1} is taken cyclically.

The character degrees of G are at most 2, because H^2 is an Abelian subgroup of index 2. The sum of the squares of the character degrees is $|G|$, therefore, the sum of their cubes is $\le 2|G|$, which is $4n^6$.

We will show below, that G realizes $\langle |S_1|,|S_2|,|S_3|\rangle$. Each S_i has size $2n(n-1)$. Thus the pseudo exponent is

$$\frac{3\log|G|}{\log(|S_1|^3)} = \frac{\log 2n^6}{\log 2n(n-1)}\,.$$

By Corollary 10.8,

$$|G|^{\omega/\alpha} = (2n(n-1))^6 \le |G| \cdot 2^{\omega-2} = 2^{\omega-2} 2n^6\,.$$

If we set $n = 17$, we get the bound $\omega \leq 2.91$.

It remains to show that S_1, S_2 and S_3 satisfy the triple product property. Let $q_i \in Q(S_i)$. We have $q_i = (a_i, b_i)(c_i^{-1}, d_i^{-1})$ or $q_i = (a_i, b_i)z(c_i^{-1}, d_i^{-1})$. In a product $q_1q_2q_3 = 1$, there are either two appearances of z or none; since otherwise, $q_1q_2q_3 = (x, y)z \neq 1$.

First assume that there are none. Then

$$q_1q_2q_3 = (a_1c_1^{-1}a_2c_2^{-1}a_3c_3^{-1}, b_1d_1^{-1}b_2d_2^{-1}b_3d_3^{-1})\,.$$

Thus $q_1q_2q_3 = 1$ iff $q_1 = q_2 = q_3 = 1$, since the triple product property holds for each factor H separately.

Now assume that there are two appearances of z. Assume that it appears in q_1 and q_2. The other cases are treated similarly. We have

$$q_1q_2q_3 = (a_1d_1^{-1}b_2c_2^{-1}a_3c_3^{-1}, b_1c_1^{-1}a_2d_2^{-1}b_3d_3^{-1})$$

a_1 is the only element from $C_n \times \{1\} \times \{1\}$ in the first product on the right-hand side. Since $a_1 \neq 1$, the product $q_1q_2q_3 \neq 1$.

11 Support rank

Finally, we consider another relaxation of rank, introduced in [11].

Definition 11.1.

1. Two tensors $t, t' \in K^{k \times m \times n}$ are *support equivalent* if for all h, i, j,

$$t_{h,i,j} \neq 0 \iff t'_{h,i,j} \neq 0\,.$$

   We write $t \sim_{\mathrm{s}} t'$.

2. The *support rank* (or s-rank for short) of a tensor t is defined by

$$R_{\mathrm{s}}(t) = \min\{R(t') \mid t' \sim_{\mathrm{s}} t\}\,.$$

By definition, the s-rank is a lower bound for the rank. But the s-rank can be much lower.

Example 11.2. Let I be the identity matrix and J be the all-ones matrix of size $n \times n$. Then $R(J - I) = n$. Let $M = (\zeta^{i-j})$ for some primitive nth root of unity ζ. M is a rank-one matrix. $M - I$ and $J - I$ are support equivalent. But $R_{\mathrm{s}}(M - I) \leq 2$, since s-rank is subadditive.

Like border rank, s-rank is a relaxation of rank. These two relaxations are however incomparable. In the example above, $J - I$ has border rank n, too. On the other hand, then tensor at the beginning of Section 6 has s-rank 3 by the same proof given there. (Most lower bound proofs for the rank based on substitution method also work for s-rank.)

Definition 11.3. The *s-rank exponent* of matrix multiplication is defined as

$$\omega_{\mathrm{s}} = \inf\{\tau \mid R_{\mathrm{s}}(\langle n, n, n\rangle) = O(n^\tau)\}\,.$$

Note that s-rank behaves like rank: It is subadditive and submultiplicative. We have $(kmn)^{\omega_s} \leq R_s(\langle k,m,n\rangle)$. We can define border s-rank and get a similar relation to s-rank. The asymptotic sum inequality holds for the s-rank, too, and the laser methods works as well, provided that we replace ω by ω_s. How are ω and ω_s related?

Theorem 11.4. $\omega \leq (3\omega_s - 2)/2$.

Proof. Given $\varepsilon > 0$, choose C such that $R_s(\langle n,n,n\rangle) \leq C \cdot n^{\omega_s+\varepsilon}$. Let t be a tensor with $t \sim_s \langle n,n,n\rangle$ and $R(t) \leq Cn^{\omega_s+\varepsilon}$. Decompose $\langle n,n,n\rangle = \langle n,n,1\rangle \otimes \langle 1,1,n\rangle$. This induces a decomposition of $t = t_1 \otimes t_2$ with $t_1 \sim_s \langle n,n,1\rangle$ and $t_2 \sim_s \langle 1,1,n\rangle$. Now think of t having inner structure t_1 and outer structure t_2. By Lemma 11.6 below, t_1 is isomorphic to $\langle n,n,1\rangle$ and t_2 is isomorphic to $\langle 1,1,n\rangle$. But this is exactly the situation we were in when applying the laser method to Str. In the same way, we get

$$n^2 n^{2\omega} \leq n^{3(\omega_s+\varepsilon)}.$$

Since this is true for any ε, we get the desired bound. □

In other words, if $\omega_s \leq 2+\varepsilon$, then $\omega \leq 2+\frac{3}{2}\varepsilon$. In particular, if $\omega_s = 2$, then $\omega = 2$.

Problem 11.5. Can the factor $\frac{3}{2}$ above be improved?

Lemma 11.6. *Let t be a tensor with slices $t_1,\dots,t_n$. such that each t_i has only one nonzero entry. If $t' \sim_s t$, then t' is isomorphic to t.*

Proof. Assume that w. l. o. g. $t_1,\dots,t_n$ are the 1-slices of t. We can assume that they are all nonzero. Let t' be a tensor with $t' \sim_s t$. Let $t'_1,\dots,t'_n$ be the slices of t'. Then $t_i = \alpha_i t'_i$ for some $\alpha_i \in K$, $1 \leq i \leq n$. Let $A : K^n \to K^n$ be the isomorphism defined by multiplying the ith coordinate by α_i, $1 \leq i \leq n$. Then $(A \otimes I \otimes I)t = t'$. □

How to make use out of s-rank? Cohn and Umans [11] generalize their group theoretic approach by replacing groups by coherent configurations and group algebras by adjacency algebras. The s-rank comes into play because of the structural constants. In group algebras, these are either 0 or 1, in adjacency algebras, they can be arbitrary. Because of the structural constants, adjacency algebras yield bounds on ω_s instead of ω. The interested reader is referred to the original paper [11]. Furthermore, Cohn and Umans currently do not get any bound on ω_s that is better then the current best upper bounds on ω. So a lot of challenging open problems are waiting out there!

Acknowledgement 11.7. This article is based on the course material of the course "Bilinear Complexity" which I held at Saarland University in summer term 2009 and 2013. I would like to thank Fabian Bendun who typed my lecture notes. I would also like to thank all other participants of the course. I learnt most of the results presented in this article from Arnold Schönhage when I was a student at the University of Bonn in the nineties of the last century. The way I present the results and many of the proofs are inspired by what I learnt from him. Amir Shpilka forced me to write and publish this article. He was very patient.

References

[1] VALERY B. ALEKSEYEV: On the complexity of some algorithms of matrix multiplication. *J. Algorithms*, 6(1):71–85, 1985. [doi:10.1016/0196-6774(85)90019-7] 18, 25

[2] NOGA ALON, AMIR SHPILKA, AND CHRISTOPHER UMANS: On sunflowers and matrix multiplication. *Comput. Complexity*, 22(2):219–243, 2013. Preliminary version in CCC'12. See also at ECCC. [doi:10.1007/s00037-013-0060-1] 55

[3] ULRICH BAUM AND MICHAEL CLAUSEN: *Fast Fourier Transforms*. Spektrum Akademischer Verlag, 1993. 51

[4] DARIO BINI, MILVIO CAPOVANI, FRANCESCO ROMANI, AND GRAZIA LOTTI: $O(n^{2.7799})$ complexity for $n \times n$ approximate matrix multiplication. *Inform. Process. Lett.*, 8(5):234–235, 1979. [doi:10.1016/0020-0190(79)90113-3] 26

[5] MARKUS BLÄSER: On the complexity of the multiplication of matrices of small formats. *J. Complexity*, 19(1):43–60, 2003. [doi:10.1016/S0885-064X(02)00007-9] 2, 25

[6] ALFRED T. BRAUER: On addition chains. *Bull. Amer. Math. Soc.*, 45(2):736–739, 1939. [doi:10.1090/S0002-9904-1939-07068-7] 6

[7] NADER H. BSHOUTY: On the additive complexity of 2×2-matrix multiplication. *Inform. Process. Lett.*, 56(6):329–335, 1995. Preliminary version in SAC'91. [doi:10.1016/0020-0190(95)00176-X] 2

[8] PETER BÜRGISSER, MICHAEL CLAUSEN, AND MOHAMMAD AMIN SHOKROLLAHI: *Algebraic Complexity Theory*. Springer, 1997. [doi:10.1007/978-3-662-03338-8] 10, 25, 40

[9] HENRY COHN, ROBERT D. KLEINBERG, BALÁZS SZEGEDY, AND CHRISTOPHER UMANS: Group-theoretic algorithms for matrix multiplication. In *Proc. 46th FOCS*, pp. 379–388. IEEE Comp. Soc. Press, 2005. [doi:10.1109/SFCS.2005.39] 55

[10] HENRY COHN AND CHRISTOPHER UMANS: A group-theoretic approach to fast matrix multiplication. In *Proc. 44th FOCS*, pp. 438–449. IEEE Comp. Soc. Press, 2003. [doi:10.1109/SFCS.2003.1238217] 51, 52

[11] HENRY COHN AND CHRISTOPHER UMANS: Fast matrix multiplication using coherent configurations. In *Proc. 24th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'13)*, pp. 1074–1087. ACM Press, 2013. SIAM. See also at arXiv. 56, 57

[12] DON COPPERSMITH AND SHMUEL WINOGRAD: On the asymptotic complexity of matrix multiplication. *SIAM J. Comput.*, 11(3):472–492, 1982. Preliminary version in FOCS'81. [doi:10.1137/0211038] 34

[13] DON COPPERSMITH AND SHMUEL WINOGRAD: Matrix multiplication via arithmetic progressions. *J. Symbolic Comput.*, 9(3):251–280, 1990. Preliminary version in STOC'87. [doi:10.1016/S0747-7171(08)80013-2] 40

[14] ALEXANDER M. DAVIE AND ANDREW JAMES STOTHERS: Improved bound for complexity of matrix multiplication. *Proc. Royal Soc. Edinburgh: Sect. A Math.*, 143(2):351–369, 2013. [doi:10.1017/S0308210511001648] 50

[15] HANS F. DE GROOTE: On the varieties of optimal algorithms for the computation of bilinear mappings II: Optimal algorithms for 2×2–matrix multiplication. *Theoret. Comput. Sci.*, 7(2):127–148, 1978. [doi:10.1016/0304-3975(78)90045-2] 2

[16] HANS F. DE GROOTE: *Lectures on the Complexity of Bilinear Problems*. Volume 245 of *Lecture Notes in Computer Science*. Springer, 1987. [doi:10.1007/BFb0020719] 29

[17] JOHAN HÅSTAD: Tensor rank is NP-complete. *J. Algorithms*, 11(4):644–654, 1990. Preliminary version in ICALP'89. [doi:10.1016/0196-6774(90)90014-6] 25

[18] GORDON JAMES AND MARTIN LIEBECK: *Representations and Characters of Groups*. Cambridge Univ. Press, 2001. [doi:10.1017/CBO9780511814532] 51

[19] ANATOLII ALEXEEVICH KARATSUBA: The complexity of computations. *Proc. Steklov Institute of Mathematics*, 211(169–183), 1995. Translation from Trudy Mat. Inst. Steklova, 207, 180–196 (1994). Available at `http://www.cs.ru/personal/karatsuba/divcen.pdf`. 4

[20] ANATOLII ALEXEEVITCH KARATSUBA AND YURI PETROVICH OFMAN: Multiplication of many-digit numbers by automatic computers. *Proc. USSR Academy of Sciences*, 145(293–294), 1962. 4

[21] JULIAN D. LADERMAN: A noncommutative algorithm for multiplying 3×3–matrices using 23 multiplications. *Bull. Amer. Math. Soc.*, 82(1):126–128, 1976. [doi:10.1090/S0002-9904-1976-13988-2] 2, 25

[22] THEODORE S. MOTZKIN: Evaluation of polynomials. *Bull. Amer. Math. Soc.*, 61(2):113–184, 1955. [doi:10.1090/S0002-9904-1955-09897-5] 7

[23] ALEXANDER M. OSTROWKSI: On two problems in abstract algebra connected with Horner's rule. In *Studies in Mathematics and Mechanics presented to Richard von Mises*, pp. 40–48. Academic Press, 1954. 7

[24] VICTOR YA. PAN: Methods of computing values of polynomials. *Russ. Math. Surv.*, 21(1):105–136, 1966. [doi:10.1070/RM1966v021n01ABEH004147] 7

[25] VICTOR YA. PAN: New fast algorithms for matrix operations. *SIAM J. Comput.*, 9(2):321–342, 1980. [doi:10.1137/0209027] 25

[26] ARNOLD SCHOLZ: Aufgabe 253. *Jahresberichte der deutschen Mathematiker-Vereinigung*, 47(4):41–42, 1937. 6

[27] ARNOLD SCHÖNHAGE: A lower bound for the length of addition chains. *Theoret. Comput. Sci.*, 1(1):1–12, 1975. [doi:10.1016/0304-3975(75)90008-0] 6

[28] ARNOLD SCHÖNHAGE: Partial and total matrix multiplication. *SIAM J. Comput.*, 10(3):434–455, 1981. [doi:10.1137/0210032] 29, 31

[29] KENNETH B. STOLARSKY: A lower bound for the Scholz–Brauer problem. *Canad. J. Math.*, 21:675–683, 1969. [doi:10.4153/CJM-1969-077-x] 6

[30] ANDREW JAMES STOTHERS: *On the complexity of matrix multiplication*. Ph. D. thesis, The University of Edinburgh, 2010. Edinburgh Research Archive. 50

[31] VOLKER STRASSEN: Gaussian elimination is not optimal. *Numer. Math.*, 13(4):354–356, 1969. [doi:10.1007/BF02165411] 2

[32] VOLKER STRASSEN: Vermeidung von Divisionen. *J. Reine Angew. Math.*, 264:184–202, 1973. EUMDL. 13

[33] VOLKER STRASSEN: Relative bilinear complexity and matrix multiplication. *J. Reine Angew. Math.*, 375/376:406–443, 1987. EUDML. 38

[34] ABRAHAM WAKSMAN: On Winograd's algorithm for inner products. *IEEE Trans. Comput.*, C–19(4):360–361, 1970. [doi:10.1109/T-C.1970.222926] 18

[35] VIRGINIA VASSILEVSKA WILLIAMS: Multiplying matrices faster than Coppersmith-Winograd. In *Proc. 44th STOC*, pp. 887–898. ACM Press, 2012. [doi:10.1145/2213977.2214056] 50

[36] SHMUEL WINOGRAD: A new algorithm for inner product. *IEEE Trans. Comput.*, C–17(7):693–694, 1968. [doi:10.1109/TC.1968.227420] 18

[37] SHMUEL WINOGRAD: On the number of multiplications necessary to compute certain functions. *Comm. Pure and Appl. Math*, 23(2):165–179, 1970. Preliminary version in PNAS 1967. [doi:10.1002/cpa.3160230204] 7

[38] SHMUEL WINOGRAD: On multiplication of 2×2–matrices. *Lin. Alg. Appl.*, 4(4):381–388, 1971. [doi:10.1016/0024-3795(71)90009-7] 2

AUTHOR

Markus Bläser
full professor
Saarland University, Saarbrücken, Germany
mblaeser@cs.uni-saarland.de

ABOUT THE AUTHOR

MARKUS BLÄSER is notorious for not putting his CV anywhere. The explanations in the ToC-Style file what to put here made him almost switch to software engineering.