

Exposition of OpenAI's Matrix Multiplication in Exponent $9/4$ (and an explicit algorithm)

Amir Shpilka*

Abstract

A recent manuscript of OpenAI proves that the exponent of matrix multiplication satisfies $\omega \leq 9/4$. We give an elementary, self-contained proof of this bound. We follow the ideas of the original manuscript, but implement them differently to obtain explicit algorithms. Specifically, for every $\varepsilon > 0$, we give a matrix multiplication algorithm with exponent below $9/4 + \varepsilon$ and base matrix size at most $2^{(1/\varepsilon)^{O(1)}}$.

We also show that techniques similar to those used in the square case apply to rectangular matrix multiplication. Starting from Strassen's tensor, we give elementary proofs of the bounds $\omega(1, \frac{2}{3 \log 3}, 1) = 2$ and $\omega(1, 0.539, 2) = 3$. Stronger rectangular bounds are known, but we include these proofs for their simplicity.

In an appendix we describe the original proof, which goes through Strassen's theory of the asymptotic spectrum.

*Blavatnik School of Computer Science and AI, and School of Mathematical Sciences, Tel Aviv University, shpilka@tauex.tau.ac.il. This research was funded by the European Union (ERC, EACTP, 101142020). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Contents

1	Introduction	1
1.1	The high-level approach	1
1.2	The main theorem	2
1.3	Earlier work	3
1.4	Comparison with the original proof	4
1.5	Concurrent expositions	5
1.6	AI disclosure	6
1.7	Acknowledgment	6
1.8	Organization	6
2	Matrix multiplication, tensors and the exponent ω	7
2.1	Strassen's algorithm	7
2.2	Trilinear forms and tensors	8
2.3	Operations on tensors	9
2.4	Matrix multiplication tensors and dot products	12
2.5	From border rank to algorithms: Bini's theorem	13
3	Polynomial multiplication, the seed tensor and the rate	14
3.1	Polynomial multiplication	14
3.2	The seed tensor	15
3.3	The rate of a tensor	16
4	Proving the main theorem	17
4.1	The basic inequalities	17
4.2	The diagonal-growth argument	18
5	Proving the inequalities	20
5.1	Ideas behind the proofs	20
5.2	Extracting specific factors from tensor powers	21
5.3	Discrete concavity	23
5.4	Shifted tripling	28
6	Constructing an explicit sequence of algorithms	31
6.1	Proof of Lemma 6.2	35
7	(In)Dependence on the field	40
8	Rectangular matrix multiplication	41
8.1	The rectangular exponent	41
8.2	Strassen's tensor	42
8.3	Two exact values of the rectangular exponent	43
9	Concluding remarks and open problems	45
A	The proof of [Ope26a] via spectral points	49

1 Introduction

Multiplying two $n \times n$ matrices by the school method requires $O(n^3)$ multiplications and additions. In his seminal 1969 breakthrough, Strassen showed that one can do better [Str69]: two 2×2 matrices can be multiplied with 7 multiplications instead of 8, and applying this construction recursively to block matrices gives an algorithm with $O(n^{\log_2 7}) = O(n^{2.81})$ operations. The *exponent of matrix multiplication* ω is the infimum of achievable exponents (Definition 2.1). It is known that $2 \leq \omega \leq 3$, and a long line of work has pushed the upper bound down. Recent progress mainly improved the third digit after the point, going from $\omega \leq 2.376$ to $\omega \leq 2.372$. A recent manuscript of OpenAI [Ope26a] made a breakthrough, proving

$$\omega \leq \frac{9}{4},$$

using new and elegant ideas. In this paper we aim to present the result without assuming prior knowledge of matrix multiplication algorithms. Apart from the background, which can be found in the excellent survey [Blä13] and in the more comprehensive book [BCS97], the main ideas and the basic constructions are from [Ope26a]. However, the way we combine them and the overall presentation are different (in Section 1.4 we compare the two proofs).

Readers unfamiliar with tensors are advised to jump straight to Section 2 and return to read the rest of the introduction after becoming familiar with the notions of tensors and degenerations.

1.1 The high-level approach

Suppose we can multiply two $u \times u$ matrices using r products of linear forms in their entries. The same identity applies to matrices whose entries are themselves square matrices, where now each product multiplies two matrices of smaller dimension. Applying this recursively gives an algorithm using $r^{\log_u(n)} = n^{\log_u(r)}$ products, i.e., $\omega \leq \log_u(r)$, in a similar manner to the way that Strassen extended his 2×2 improvement to arbitrary n (Proposition 2.3). Most subsequent algorithms do not start from the matrix multiplication tensor itself, but rather from another tensor T and obtain a large matrix multiplication tensor from a high tensor power of T , using restrictions or degenerations (Section 2.3). If T has rank at most r ,¹ $R(T) \leq r$, and $T^{\otimes m} \geq M_n$, then Bini’s theorem (Theorem 2.23) gives

$$\omega \leq \frac{\log(r^m)}{\log n} = \frac{\log r}{\log(n^{1/m})}. \quad (1.1)$$

One can think of the degeneration as saying that we can extract a matrix multiplication tensor of dimension $n^{1/m}$ per copy of T . Measuring and tracking the sizes of extractable matrix multiplication tensors is the basis for obtaining $\omega \leq 9/4$.

The starting computation in [Ope26a] is the polynomial multiplication tensor. Two polynomials with a coefficients each (that is, each of degree at most $a - 1$) can be multiplied with $2a - 1$ products using the standard approach: evaluate the polynomials at $2a - 1$ points, multiply their values, and interpolate (which is a linear operation given the products). More generally, we denote

$$C(a, b) = \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} x_i y_j z_{i+j} \quad (1.2)$$

for the tensor corresponding to multiplying a polynomial of degree $< a$ with a polynomial of degree $< b$: the coefficient of z_k is the k th coefficient of the product of the polynomials with coefficient vectors (x_i) and (y_j) .

¹Roughly, rank is the number of multiplications needed to compute the tensor (Definition 2.6).

Let K_a be the tensor formed from three copies of this computation, with the output of one copy placed in each of the three variable groups (i.e., one as (1.2), one with $x_{i+j}y_iz_j$ as the summand and one with $x_jy_{i+j}z_i$), see (3.1). By basic properties of tensors, the rank of this tensor product satisfies $R(K_a) \leq (2a-1)^3$. If powers of K_a produce matrix products whose dimension grows like $a^{4/3}$ per copy, then (1.1) gives

$$\omega \leq \frac{3 \log(2a-1)}{(4/3) \log a} = \frac{9}{4} \cdot \frac{\log(2a-1)}{\log a} \rightarrow \frac{9}{4},$$

which is how we get the exponent $\frac{9}{4}$.

1.2 The main theorem

In Definition 3.5 we introduce the rate function $F(a, b)$, which measures the matrix dimension obtainable from powers of the balanced tensor $Q(a, b) = \bigotimes_{\pi \in S_3} C(a, b)^\pi$. More precisely, $F(a, b)$ is the supremum of $n^{1/(2m)}$ over degenerations $Q(a, b)^{\otimes m} \supseteq M_n$. The factor 2 in this normalization accounts for the fact that $Q(a, a) \cong K_a^{\otimes 2}$. Thus, on the diagonal,

$$F(a, a) = \sup\{n^{1/m} : K_a^{\otimes m} \supseteq M_n\}.$$

We prove that the following quantity is a lower bound on $F(a, a)$.

$$\Lambda_a = \frac{3a-1}{2} \prod_{i=1}^{a-1} \left(1 + \frac{1}{3i}\right), \quad (1.3)$$

so that, e.g., $\Lambda_1 = 1$, $\Lambda_2 = 10/3$, and $\Lambda_3 = 56/9$.

Theorem 1.1. *For every integer $a \geq 1$, $F(a, a) \geq \Lambda_a$, and $\Lambda_a > a^{4/3}$ for $a \geq 2$. Consequently, for every $a \geq 2$,*

$$\omega \leq \frac{3 \log(2a-1)}{\log \Lambda_a} < \frac{9}{4} \cdot \frac{\log(2a-1)}{\log a},$$

and therefore $\omega \leq 9/4$.

Theorem 1.1 bounds ω , but its proof does not exhibit an algorithm. By carrying out explicitly a variant of the constructions behind the proof, we also obtain an explicit algorithm.

Theorem 1.2. *For every $0 < \varepsilon \leq 1$, we can explicitly construct an exact matrix multiplication algorithm with exponent below $9/4 + \varepsilon$. Its base matrix size is at most*

$$2^{(1/\varepsilon)^{O(1)}}.$$

The proof of Theorem 1.2 can be seen as giving an alternative proof of $\omega \leq 9/4$. However, we chose to give the proof of Theorem 1.1 separately, as the proof of Theorem 1.2 is a bit more technical and harder to follow compared to the simpler approach of bounding the rate using the two rate inequalities, without having to keep track of all the degenerations.

Finally, we give two examples showing how the same ideas also give algorithms for rectangular matrices. Write $\omega(a, b, c)$ for the exponent of multiplying an $n^a \times n^b$ matrix by an $n^b \times n^c$ matrix. In Section 8 we prove

$$\omega\left(1, \frac{2}{3 \log 3}, 1\right) = 2, \quad \text{and} \quad \omega(1, 0.539, 2) = 3.$$

In both cases the exponent matches the lower bound given by the number of outputs. The proofs apply the same ideas as in the square case to small tensors that are sums of two dot products. The first bound uses the four-term tensor T_2 , and the second combines the constructions for T_2 and T_4 , where T_q is the tensor used by Strassen in the paper [Str87] in which he introduced the laser method. Better results are known for rectangular matrix multiplication, but in this case we chose simplicity over optimality. See Remark 8.10.

1.3 Earlier work

Upper bounds. After Strassen’s 2×2 algorithm there was no progress for almost a decade. Then came a sequence of conceptual advances, each of which found a new way of turning a “cheap” bilinear computation into a fast algorithm for matrix multiplication. Pan [Pan78] introduced the method of trilinear aggregating and obtained $\omega < 2.796$. Bini, Capovani, Romani and Lotti [BCRL79] noticed that one can save multiplications by allowing *approximate* algorithms, whose output is correct only in the limit when a parameter ε tends to zero, and obtained $\omega < 2.78$ (for computing an approximation to matrix multiplication). Bini [Bin80] then showed that approximate algorithms can be turned into exact ones with the same exponent. This is the notion of border rank that we use throughout (Section 2.3). Schönhage [Sch81] proved the *asymptotic sum inequality*: if p independent matrix products, where the i th multiplies an $n_i \times m_i$ matrix by an $m_i \times \ell_i$ matrix, can be computed together by an approximate bilinear algorithm with r multiplications, then we obtain the following upper bound on ω :

$$\sum_{i=1}^p (n_i m_i \ell_i)^{\omega/3} \leq r.$$

For example, p independent products of $n \times n$ matrices with r multiplications give $p n^\omega \leq r$. So to bound ω it suffices to compute many independent matrix products at once, possibly of different shapes. Using this result Schönhage obtained $\omega < 2.522$.

Another breakthrough of Strassen is the *laser method* [Str87] that became the basis for all later advances (before [Ope26a]). It starts from a tensor that is not a matrix multiplication tensor, but that is composed of a combination of small matrix multiplication tensors. We then take a large tensor power of that tensor and zero out variables, so that the remaining tensor is a direct sum of many independent matrix products, to which the asymptotic sum inequality of Schönhage applies. Strassen obtained $\omega < 2.479$ in this way. Coppersmith and Winograd [CW90] applied the laser method to a carefully designed tensor, now called the Coppersmith–Winograd tensor, using sets without three-term arithmetic progressions to decide which variables to zero out (in a high tensor power), and obtained $\omega < 2.376$.

It took more than twenty years until subsequent work, which applied the laser method to tensor powers of the Coppersmith–Winograd tensor with more and more refined analyses [Sto10, VW12, DS13, LG14, AVW21, DWZ23, VWXXZ24], managed to improve the third digit after the point. The best published bound before [Ope26a] was $\omega < 2.37134$ [ADVW⁺25], and a recent preprint [DEK⁺26] reports $\omega < 2.37118$, obtained by better optimization within the same framework. Shortly before [Ope26a], OpenAI released another paper [Ope26b] proving $\omega < 2.258$ over $\mathbb{C}$, together with new bounds for rectangular matrix multiplication.

A different, group-theoretic approach was suggested by Cohn and Umans [CU03, CKSU05]. It generalizes the way the fast Fourier transform is used to multiply polynomials: one embeds matrix multiplication in the multiplication of a group algebra, performs that multiplication using Fourier analysis (for noncommutative groups, using irreducible representations), and reads off the entries of the matrix product. It led to interesting conjectures that would imply $\omega = 2$, but so far it has not improved on the laser method.

Barriers. The slow progress was partly explained by several barrier results. Ambainis, Filmus and Le Gall [AFLG15] showed that a large class of variants of the laser method, applied to powers of the Coppersmith–Winograd tensor with $q = 5$,² cannot give an algorithm with running time $O(n^{2.3078})$. Alman and Vassilevska Williams [AVW18] considered more general methods, which may use arbitrary monomial degenerations (this includes the laser method and the group-theoretic approach), and showed that when they start from any tensor in a large family generalizing the Coppersmith–Winograd tensor, they cannot prove a bound better than some constant $\hat{\omega} > 2$. Alman [Alm21] and Christandl, Vrana and Zuiddam [CVZ21] proved barriers that hold even for arbitrary degenerations. For example, [CVZ21] shows that if a method passes through a tensor T , then the bound it gives is at least twice a quantity called the *irreversibility* of T .

It is important to note that these barriers concern what can be obtained from powers of a *fixed* starting tensor, and need not give a uniform bound above 2 if we change the starting tensor. Christandl, Vrana and Zuiddam [CVZ21, Section 5.2] give an example of this: for reduced polynomial multiplication tensors, which compute multiplication modulo x^n , their barrier tends to 2 as $n \rightarrow \infty$.

Lower bounds. The only known lower bound on ω is the trivial $\omega \geq 2$. For concrete matrix sizes more is known. Winograd [Win71] proved that seven multiplications are needed to multiply 2×2 matrices, and Landsberg [Lan06] proved that even approximate algorithms need seven, i.e., that the border rank of M_2 is 7. For $n \times n$ matrices, Bläser [Blä99] proved that the rank of M_n is at least $\frac{5}{2}n^2 - 3n$ over any field. This was improved to $3n^2 - o(n^2)$ over $\mathbb{F}_2$ in [Shp03], and later to $3n^2 - o(n^2)$ over $\mathbb{C}$ by Landsberg [Lan14] and Massarenti and Raviolo [MR13, MR14]. For the border rank, Strassen [Str83] proved a lower bound of $\frac{3}{2}n^2$, which was improved to $2n^2 - n$ by Landsberg and Ottaviani [LO15] and to $2n^2 - \lceil \log_2 n \rceil - 1$ by Landsberg and Michałek [LM18]. Raz [Raz03] proved an $\Omega(n^2 \log n)$ lower bound for arithmetic circuits over $\mathbb{C}$ whose scalar multiplication coefficients are bounded by an absolute constant. Raz and Shpilka [RS03] proved that constant-depth arithmetic circuits over any field, and constant-depth circuits with arbitrary gates over any finite field, require a number of edges superlinear in n^2 to compute M_n . For general arithmetic circuits nothing better than $\Omega(n^2)$ is known. We note, however, that such lower bounds say nothing nontrivial about ω .

1.4 Comparison with the original proof

The proof of [Ope26a] works with *spectral points*, which are the points of Strassen’s asymptotic spectrum³ [Str88, CVZ23]. See the survey of Wigderson and Zuiddam [WZ26] for an introduction to the asymptotic spectrum and its uses. Roughly, these are maps from tensors to the nonnegative reals that are additive under direct sums, multiplicative under tensor products, monotone under degenerations, and equal to 1 on the unit tensor $I_1 = xyz$. Strassen showed that a spectral point λ must satisfy $\lambda(M_n) = n^{3\theta_\lambda}$, for some $\theta_\lambda > 0$ that depends on λ [Str88, Section 4]. For a fixed spectral point λ , [Ope26a] defines a quantity $P_\lambda(a, b)$ from the values of λ on $C(a, b)$ and its leg permutations, and proves that P_λ satisfies *concavity* and *tripling* inequalities. It is a simple calculation to prove that these inequalities imply $P_\lambda(a, a) \geq a^{4/3}$. On the other hand, the rank of $C(a, a)$ gives $P_\lambda(a, a) \leq (2a - 1)^{1/\theta_\lambda}$, which implies $\theta_\lambda \leq 3/4$. To get back from spectral points to a bound on ω , [Ope26a] uses a special case of a theorem of Strassen [Str88, Theorem 3.8]: the *asymptotic rank* of a tensor (Exercise 2.26) is the maximum of its values over all spectral points. Since $R(M_m) \geq m^\omega$ for every $m \geq 2$, the asymptotic rank of M_n is at least n^ω . Hence, for every

²The Coppersmith–Winograd tensor is actually a family of tensors CW_q , parametrized by q .

³[Ope26a] uses the term *characters* for spectral points, but since spectral points is the common term in the CS literature we use it here.

$n \geq 2$ and every integer $k < n^\omega$ there is a spectral point with $\lambda(M_n) \geq k$ (this special case is stated in [Ope26a, Lemma 2.2]). Taking $k = \lceil n^\omega \rceil - 1$, we get

$$n^\omega - 1 \leq k \leq \lambda(M_n) = n^{3\theta_\lambda} \leq n^{9/4},$$

implying $\omega \leq 9/4$.

Our approach, which is inspired by the proof of [Ope26a], is to measure, for every tensor, the largest size of a matrix multiplication tensor that we can extract from high powers of that tensor, per copy. Roughly, $F(a, b)$ is this measure for the tensor $Q(a, b)$.⁴ Thus, this notion is defined via degeneration of tensors, unlike the function P_λ which depends on the spectral point λ . We prove the same concavity and tripling inequalities as [Ope26a] proves for P_λ . Consequently, the same calculation implies that $F(a, a) > a^{4/3}$. Since F is defined directly via degenerations, this immediately implies $\omega \leq 9/4$.

This point of view has the added advantage that it allows us to obtain explicit algorithms with exponent approaching $9/4$. The idea is to follow the degenerations behind the proofs of the concavity and tripling inequalities. Basically, [Ope26a] shows that two copies of $C(a, b)$ that share the first input degenerate to $C(a, b-1) + C(a, b+1)$, and that $C(a, 3c+a-1)$ degenerates to three copies of $C(a, c)$. Together with some extra machinery used to separate tensors that share a leg into a direct sum (Lemma 5.1), these degenerations imply the concavity and tripling inequalities

$$2F(a, b) \geq F(a, b-1) + F(a, b+1), \quad \text{and} \quad F(a, 3c+a-1) \geq 3F(a, c).$$

It is also not hard to see that $F(a, b) \geq \max(a, b)$. Using these facts alone, one can prove $F(a, a) \geq \Lambda_a > a^{4/3}$, which implies the bound on ω .

To obtain an algorithm, we first derive a sufficiently large lower bound on $F(A, A)$ by finitely many applications of the concavity and tripling inequalities, starting from the baseline bounds $F(a, b) \geq \max(a, b)$. Whenever an inequality is applied we perform the appropriate degeneration. For example, if we obtained a lower bound on $F(a, b)$ from bounds on $F(a, b-1)$ and $F(a, b+1)$ using concavity, then we will take high power $Q(a, b)^{\otimes N}$ and apply the degeneration to obtain $(Q(a, b-1) + Q(a, b+1))^{\otimes N}$, then separating the legs we will get $Q(a, b-1)^{\otimes N_1} \otimes Q(a, b+1)^{\otimes N-N_1}$.⁵ Thus, we start from a large tensor $Q(a, a)$ and walk our way “down” using the degenerations used in the proof of the inequalities. The process itself is more complicated and involves using auxiliary tensors. Section 6 explains how to compose these constructions and bound the resulting tensor powers and auxiliary tensors. This gives, for every $\varepsilon > 0$, a prescribed sequence of applications of these constructions, composed of $\text{poly}(1/\varepsilon)$ many steps, that yields an explicit restriction $Q(A, A)^{\otimes m} \geq M_n$, for a suitable A , with exponent below $9/4 + \varepsilon$, where m and $\log n$ are polynomial in $1/\varepsilon$. The algorithm then follows by recursion.

We note that unlike the laser method of [Str87], which starts from a single fixed tensor and zeroes out variables in a high tensor power, our algorithmic version of the proof of [Ope26a] works with the whole family of polynomial multiplication tensors $C(a, b)$ and relates their rates for different input lengths (in our language, the numbers $F(a, b)$).

For a more detailed account of the proof see Sections 4 and 5.

1.5 Concurrent expositions

During the final stages of preparing this exposition, two other papers presenting proofs of the result of [Ope26a] appeared [Pas26, AvdBZ26]. We briefly compare their approaches with ours.

⁴It is actually the square root of this number for reasons we explain in Section 3.3.

⁵In this description we considerably simplified what is actually going on. In particular, we start from $C(a, b)$ and not $Q(a, b)$, and later have to symmetrize its legs. We also ignored auxiliary tensors that enter the construction.

Amanov, van den Berg, and Zuiddam [AvdBZ26] follow the proof of [Ope26a] in the language of spectral points. As in [Ope26a], their proof that $\omega \leq 9/4$ uses Strassen’s theory of the asymptotic spectrum of tensors, and they do not give an explicit matrix multiplication algorithm. They also obtain bounds for rectangular matrix multiplication and prove a related barrier result.

Pass [Pas26] takes a very different approach. He gives a constructive proof in the language of *polynomial packings*, which is a notion defined in his paper. He performs the different steps of his algorithm using the language of packings, but his steps are parallel to rate inequalities such as those we prove here. Tracking the parameters in his construction shows that he obtains a base matrix of size $2^{(1/\varepsilon)^{O(1)}}$, when the desired exponent is at most $9/4 + \varepsilon$.

Our exposition works directly with tensors and defines the notion of rate of a tensor, which roughly captures the matrix multiplication tensors obtainable from powers of the tensor. We prove inequalities for these rates using explicit degenerations (similar to [Ope26a]), and construct an algorithm by showing how to turn a finite derivation of a rate bound, from the inequalities, into an explicit algorithm. Our construction and that of [Pas26] both give base matrices of size $2^{\text{poly}(1/\varepsilon)}$, but the proofs are different. We also include applications to rectangular matrix multiplication, choosing examples for their simplicity rather than seeking the strongest available bounds. Our aim is to make the proof accessible to students with no prior background in matrix multiplication algorithms.

1.6 AI disclosure

This is an exposition of a result obtained by an internal model of OpenAI [Ope26a], and it gives a slightly different proof of the main results there. I used ChatGPT Astra to help me understand the original paper and work out this presentation. In particular, the idea of organizing the proof around the rate $F(a, b)$ rather than spectral points came out of these interactions. The (simple) idea of obtaining an explicit algorithm by following the degenerations is mine.

Throughout the work I used ChatGPT Astra and Claude Opus 5.5 to discuss and check arguments, work out technical details (especially simplifying calculations), draft and revise parts of the text, and find and check relevant references. Some technical points also came from these interactions, for example the use of $2M$ copies in the separation lemma and the bound of Corollary 8.8. I revised the initial drafts myself and have verified all the arguments. I take full responsibility for the contents of this paper.

1.7 Acknowledgment

I thank Itai Leigh and Ben Lee Volk for comments that improved the presentation.

1.8 Organization

To make this paper self-contained we give classical background in Section 2 that includes tensors and degenerations. Readers familiar with tensor rank, border rank and Bini’s theorem can skip it. A much more detailed and thorough presentation can be found in [Blä13, BCS97]. Section 3 introduces the polynomial multiplication tensor, the seed tensor, and the rate $F(a, b)$. Section 4 states the two main rate inequalities and shows how they imply the bound $\omega \leq 9/4$. In Section 5 we give the basic constructions that are the backbone of the proof of the rate inequalities, and combine them in tensor powers to prove the inequalities, and in Section 6 we turn the proof into an explicit sequence of algorithms. Section 7 extends the argument to arbitrary fields. In Section 8 we apply the same techniques and give short proofs of two exact values of the rectangular exponent. We conclude and give some open questions in Section 9. Finally, in Appendix A we give an overview of the

original proof via spectral points. In particular, we track some statements proved in [Ope26a] back to Strassen. We note that [AvdBZ26] give a more detailed account of the proof given in [Ope26a].

Conventions. Unless stated otherwise, we work over $\mathbb{F} = \mathbb{C}$. The extension to arbitrary fields is explained in Section 7. Logarithms are taken in base 2. Indices of coefficients start at 0, so that, e.g., x_i is the coefficient of ξ^i . Indices that only enumerate terms or objects start at 1.

2 Matrix multiplication, tensors and the exponent ω

We count *arithmetic operations*: additions, subtractions, and multiplications of two numbers (including multiplication by a fixed constant), each at unit cost. An algorithm for $n \times n$ matrix multiplication receives the $2n^2$ entries of X and Y and outputs the n^2 entries of XY .

Definition 2.1. *The exponent of matrix multiplication is*

$$\omega = \inf\{\tau : \text{two } n \times n \text{ complex matrices can be multiplied with } O(n^\tau) \text{ arithmetic operations}\}.$$

The school method computes each of the n^2 entries $(XY)_{ik} = \sum_j X_{ij}Y_{jk}$ with n multiplications and $n - 1$ additions, so $\omega \leq 3$. On the other hand, every algorithm must produce n^2 linearly independent quadratic polynomials, so it needs at least $\Omega(n^2)$ operations and $\omega \geq 2$. Whether $\omega = 2$ is the main open problem in the area, and the focus of [Ope26a].

2.1 Strassen's algorithm

Write 2×2 matrices as $X = \begin{pmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{pmatrix}$, and similarly Y . Strassen's seven products are

$$\begin{aligned} P_1 &= (X_{11} + X_{22})(Y_{11} + Y_{22}), & P_5 &= (X_{11} + X_{12})Y_{22}, \\ P_2 &= (X_{21} + X_{22})Y_{11}, & P_6 &= (X_{21} - X_{11})(Y_{11} + Y_{12}), \\ P_3 &= X_{11}(Y_{12} - Y_{22}), & P_7 &= (X_{12} - X_{22})(Y_{21} + Y_{22}), \\ P_4 &= X_{22}(Y_{21} - Y_{11}), \end{aligned}$$

and then

$$(XY)_{11} = P_1 + P_4 - P_5 + P_7, \quad (XY)_{12} = P_3 + P_5, \quad (XY)_{21} = P_2 + P_4, \quad (XY)_{22} = P_1 - P_2 + P_3 + P_6.$$

It is a simple exercise to verify Strassen's algorithm.

We note the following simple but important fact. Every product has the form $L(X) \cdot M(Y)$: a linear form in the entries of X times a linear form in the entries of Y , with the X -factor on the left. Hence, the identities can be thought of as being over *noncommuting variables*. They therefore remain true when the entries are themselves matrices, which do not commute.

Definition 2.2. *A bilinear algorithm with r multiplications for a family of bilinear forms $f_1, \dots, f_K$ in variables $x = (x_i)$ and $y = (y_j)$ consists of linear forms $L_1, \dots, L_r$ in x , linear forms $M_1, \dots, M_r$ in y , and scalars $\alpha_{k\rho}$ such that*

$$f_k(x, y) = \sum_{\rho=1}^r \alpha_{k\rho} L_\rho(x) M_\rho(y) \quad (1 \leq k \leq K).$$

Proposition 2.3. *Let $u \geq 2$. If there is a bilinear algorithm with r multiplications for $u \times u$ matrix multiplication, then $\omega \leq \log_u r$.*

Proof. Let $n = u^k$. View $n \times n$ matrices as $u \times u$ matrices whose entries are matrices of size n/u . Apply the bilinear algorithm to the outer $u \times u$ matrix: each $L_\rho(X)$ and $M_\rho(Y)$ is a linear combination of matrices of dimension $n/u = u^{k-1}$, each product $L_\rho(X)M_\rho(Y)$ is a product of two $(n/u) \times (n/u)$ matrices, and each output block is a linear combination of these products. Here we use the observation above that the algorithm does not rely on commutativity of inputs. Computing the linear combinations costs at most $O((n/u)^2)$ operations. Hence the number $T(n)$ of operations satisfies

$$T(u^k) \leq r T(u^{k-1}) + O(u^{2(k-1)}), \quad T(1) = 1.$$

Hence, $T(u^k) \leq r^k + O\left(\sum_{j=0}^{k-1} r^j u^{2(k-1-j)}\right) \leq O(k)r^k$, where we used $u^2 \leq r$ (Corollary 2.11 below). Since $r^k = n^{\log_u r}$ we get $T(n) = O(n^{\log_u r} \log n)$ for n a power of u . The result for general n follows by simple padding. Thus for every $\varepsilon > 0$ we can multiply $n \times n$ matrices using $O(n^{\log_u r + \varepsilon})$ operations, and $\omega \leq \log_u r$. $\square$

Strassen's algorithm thus gives $\omega \leq \log_2 7 \approx 2.807$. We shall soon see how to interpret Strassen's algorithm using tensor products of matrix multiplication tensors.

2.2 Trilinear forms and tensors

It is convenient to view a bilinear computation, together with its outputs, as a single trilinear form. In this language the inputs and the outputs play symmetric roles, and this symmetry will be used throughout this paper.

Definition 2.4. A tensor over a field $\mathbb{F}$ is a trilinear form $T : \mathbb{F}^{n_1} \times \mathbb{F}^{n_2} \times \mathbb{F}^{n_3} \rightarrow \mathbb{F}$. Equivalently,

$$T = \sum_{i < n_1, j < n_2, k < n_3} c_{ijk} x_i y_j z_k, \quad c_{ijk} \in \mathbb{F}.$$

The three groups of variables $x = (x_i)$, $y = (y_j)$, $z = (z_k)$ are called the legs X, Y, Z of T . The number of variables of a leg is its dimension. The scalars $\{c_{ijk}\}$ are called the coefficients of T .

A family of bilinear forms $f_k(x, y) = \sum_{i,j} c_{ijk} x_i y_j$ can be arranged into a single tensor $T = \sum_k f_k(x, y) z_k$: the z -variables label the outputs, and f_k is the coefficient of z_k . Conversely, every tensor defines such a family. We often read a tensor as a bilinear map $(x, y) \mapsto T(x, y)$ whose k th output is the coefficient of z_k .

Example 2.5. Matrix multiplication of an $n \times m$ matrix X by an $m \times \ell$ matrix Y is the tensor

$$\langle n, m, \ell \rangle = \sum_{i < n} \sum_{j < m} \sum_{k < \ell} x_{ij} y_{jk} z_{ki}, \quad \mathbf{M}_n = \langle n, n, n \rangle. \quad (2.1)$$

The coefficient of z_{ki} is $\sum_j x_{ij} y_{jk} = (XY)_{ik}$. In particular, $\langle n, m, \ell \rangle = \text{Tr}(XYZ)$, where $Z = (z_{ki})$ is an $\ell \times n$ matrix.

For a permutation π of the three legs, the tensor T^π is obtained by moving the variables of the leg X to the leg $\pi(X)$, and similarly for Y and Z . For instance, if τ exchanges the X - and Y -legs, then $(\sum c_{ijk} x_i y_j z_k)^\tau = \sum c_{ijk} x_j y_i z_k$. We write $(T^\pi)^\rho = T^{\rho\pi}$.

Definition 2.6. The rank $R(T)$ of a tensor T is the least r such that

$$T = \sum_{\rho=1}^r L_\rho(x) M_\rho(y) N_\rho(z)$$

with linear forms L_ρ, M_ρ, N_ρ in the variables of the respective legs, over the underlying field.

Proposition 2.7. $R(T)$ is the least number of multiplications in a bilinear algorithm for the bilinear forms defined by T . Moreover $R(T^\pi) = R(T)$.

Proof. If $f_k = \sum_\rho \alpha_{k\rho} L_\rho(x) M_\rho(y)$, then $T = \sum_k f_k z_k = \sum_\rho L_\rho(x) M_\rho(y) N_\rho(z)$ with $N_\rho(z) = \sum_k \alpha_{k\rho} z_k$. Conversely, given such a decomposition of T , the coefficient of z_k is $\sum_\rho \alpha_{k\rho} L_\rho M_\rho$, where $\alpha_{k\rho}$ is the coefficient of z_k in N_ρ . Permuting the legs permutes the three factors of each term. $\square$

Combining [Propositions 2.3](#) and [2.7](#), if $R(M_u) \leq r$ then $\omega \leq \log_u r$.

[Proposition 2.7](#) shows that discussing tensor-rank or bilinear complexity is completely equivalent. Hence, from now on we shall focus on upper bounding the rank of the tensor M_n , or more generally, of $\langle n, m, \ell \rangle$.

One way to prove lower bounds on tensor rank is to view the tensor as a matrix.

Definition 2.8 (Flattening). The X -flattening of $T = \sum c_{ijk} x_i y_j z_k : \mathbb{F}^{n_1} \times \mathbb{F}^{n_2} \times \mathbb{F}^{n_3} \rightarrow \mathbb{F}$ is the matrix $T^X \in \mathbb{F}^{n_1 \times n_2 n_3}$ with rows indexed by i , columns indexed by pairs (j, k) , and entries c_{ijk} . The flattenings T^Y and T^Z are defined similarly.

Lemma 2.9. The rank of T^X satisfies $\text{rank}(T^X) \leq R(T)$, and likewise for T^Y, T^Z .

Proof. The map $T \mapsto T^X$ is linear, and the flattening of one term $L(x)M(y)N(z)$ is a rank-one matrix: its $(i, (j, k))$ entry is $L_i M_j N_k$, where L_i, M_j, N_k are the coefficients of the three linear forms. The claim follows as a sum of r rank-one matrices has rank at most r . $\square$

Lemma 2.10. Every flattening of M_n has rank n^2 .

Proof. In M_n^X the row of x_{ij} has its nonzero entries in the columns (y_{jk}, z_{ki}) , $k < n$. Distinct rows have disjoint supports, so the n^2 rows are linearly independent. The other legs are similar. $\square$

Corollary 2.11. $R(M_n) \geq n^2$. In particular, $r \geq u^2$ in [Proposition 2.3](#).

Exercise 2.12. The multiplication of complex numbers, viewed as a bilinear map $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$, is the tensor $T = x_0 y_0 z_0 - x_1 y_1 z_0 + x_0 y_1 z_1 + x_1 y_0 z_1$. Show that its rank is 3 over $\mathbb{R}$ and 2 over $\mathbb{C}$.

2.3 Operations on tensors

As described in [Section 1.1](#), to obtain upper bounds on the rank of M_n we start with other tensors and manipulate them to eventually get the tensor M_n . For this we introduce operations on tensors that enable creating new tensors out of existing ones while controlling the change in rank.

Definition 2.13 (Restriction). Let $U = \sum u_{ijk} x_i y_j z_k$ have legs of dimensions n, m, ℓ , and let V be a tensor in variables $x' = (x'_{i'})_{i' < n'}$, $y' = (y'_{j'})_{j' < m'}$, $z' = (z'_{k'})_{k' < \ell'}$. We say that V is a restriction of U , and write $U \geq V$, if there are matrices $A \in \mathbb{F}^{n \times n'}$, $B \in \mathbb{F}^{m \times m'}$ and $C \in \mathbb{F}^{\ell \times \ell'}$ such that the substitution

$$x_i \mapsto (Ax')_i = \sum_{i'} A_{ii'} x'_{i'}, \quad y_j \mapsto (By')_j, \quad z_k \mapsto (Cz')_k$$

turns U into V , that is,

$$V = \sum_{i,j,k} u_{ijk} (Ax')_i (By')_j (Cz')_k.$$

If this holds with square invertible matrices A, B, C we say that U and V are isomorphic, and write $U \cong V$.

Thus, a restriction applies a linear map to the variables of each leg separately. Setting some variables to zero is a restriction (the corresponding rows of A , B or C are zero), and renaming the variables of a leg is an isomorphism. Restriction is transitive: if $U \geq V$ with matrices A, B, C and $V \geq W$ with matrices A', B', C' , then $U \geq W$ with AA', BB', CC' . Restriction models a reduction between computations: if $U = \sum_{\rho=1}^r L_{\rho}(x)M_{\rho}(y)N_{\rho}(z)$, then the substitution gives $V = \sum_{\rho=1}^r L_{\rho}(Ax')M_{\rho}(By')N_{\rho}(Cz')$, so a bilinear algorithm for U yields a bilinear algorithm for V with the same number of multiplications.

Definition 2.14 (Direct sum, tensor product, unit tensor). *Let $U = \sum u_{ijk}x_i y_j z_k : \mathbb{F}^{n_1} \times \mathbb{F}^{n_2} \times \mathbb{F}^{n_3} \rightarrow \mathbb{F}$ and $V = \sum v_{pqr}x'_p y'_q z'_r : \mathbb{F}^{m_1} \times \mathbb{F}^{m_2} \times \mathbb{F}^{m_3} \rightarrow \mathbb{F}$.*

(a) *The direct sum $U \oplus V$ is the tensor $\sum u_{ijk}x_i y_j z_k + \sum v_{pqr}x'_p y'_q z'_r : \mathbb{F}^{n_1+m_1} \times \mathbb{F}^{n_2+m_2} \times \mathbb{F}^{n_3+m_3} \rightarrow \mathbb{F}$, in which the variables of U and of V are distinct on every leg. The dimension of each leg is the sum of the dimensions of the corresponding legs of U and V . As a collection of bilinear maps, $U \oplus V$ computes U and V on disjoint inputs.*

(b) *The tensor product $U \otimes V$ is the tensor in the variables $X_{(i,p)}, Y_{(j,q)}, Z_{(k,r)}$, indexed by pairs,*

$$U \otimes V = \sum_{i,j,k,p,q,r} u_{ijk}v_{pqr} X_{(i,p)} Y_{(j,q)} Z_{(k,r)} : \mathbb{F}^{n_1 m_1} \times \mathbb{F}^{n_2 m_2} \times \mathbb{F}^{n_3 m_3} \rightarrow \mathbb{F}.$$

Equivalently, multiply the two trilinear forms and replace every product $x_i x'_p$ by the new variable $X_{(i,p)}$, and similarly on the other two legs. The dimension of each leg is the product of the dimensions of the corresponding legs of U and V .

(c) *For $r \geq 1$, the unit tensor is $I_r = \sum_{\rho < r} x_{\rho} y_{\rho} z_{\rho} : \mathbb{F}^r \times \mathbb{F}^r \times \mathbb{F}^r \rightarrow \mathbb{F}$. We refer to the tensor $I_r \otimes U$ as being r independent copies of U . Note that $R(T) \leq r$ if and only if $I_r \geq T$ (see [Lemma 2.17](#)).*

For matrices we use the same symbol for the *Kronecker product*: if A is an $n \times m$ matrix and A' is a $n' \times m'$ matrix, then $A \otimes A'$ is the $nn' \times mm'$ matrix with rows indexed by pairs (i, i') , columns indexed by pairs (j, j') , and entries $(A \otimes A')_{(i,i'),(j,j')} = A_{ij}A'_{i'j'}$. It satisfies $(A \otimes A')(B \otimes B') = AB \otimes A'B'$, and $\text{rank}(A \otimes A') = \text{rank } A \cdot \text{rank } A'$. This is the tensor product of matrices viewed as tensors with two legs.

2.3.1 Degenerations and border rank

Bini, Capovani, Romani, and Lotti suggested to use approximately correct computation to compute matrix multiplication. That is, computations that compute at every output coordinate a bilinear form whose coefficients are arbitrarily close to the desired coefficients. They proved that such algorithms can be much cheaper than exact ones [BCRL79]. Bini [Bin80] later showed that such approximation algorithms can actually be turned into exact algorithms with the same exponent ω . Degenerations formalize the idea of approximations, and most reductions in this paper are degenerations.

Roughly speaking, a degeneration is a restriction in which the substitution matrices may depend on a parameter t (that can be thought of as an infinitesimal added to the field), and in which, intuitively, we keep only the limit as $t \rightarrow 0$. We note that this notion is well defined also in positive characteristic.

Definition 2.15 (Degeneration). *Let U and V be tensors, and let $A(t), B(t), C(t)$ be matrices of the sizes in [Definition 2.13](#) whose entries are Laurent polynomials in t , that is, finite sums*

$\sum_{e \in \mathbb{Z}} c_e t^e$. Substituting $x_i \mapsto (A(t)x')_i$, $y_j \mapsto (B(t)y')_j$, $z_k \mapsto (C(t)z')_k$ in U gives a tensor $U'(t) = U \circ (A \otimes B \otimes C)$ whose coefficients are Laurent polynomials in t . We say that U degenerates to V , and write $U \succeq V$, if for some choice of $A(t), B(t), C(t)$ no negative powers of t appear in $U'(t)$ and its constant term is V :

$$U'(t) = V + tV_1 + t^2V_2 + \cdots + t^dV_d$$

for some tensors $V_1, \dots, V_d$. The least such d is the degree of the degeneration.

We note that we cannot substitute $t = 0$ in $A(t), B(t), C(t)$ as t may appear with negative powers there.

Over $\mathbb{C}$, for every fixed number $t \neq 0$ the matrices $A(t), B(t), C(t)$ are ordinary complex matrices, so $U'(t)$ is a restriction of U , and $V = \lim_{t \rightarrow 0} U'(t)$ is a limit of restrictions of U . The definition borrows its intuition from this language of approximations, but is defined for every characteristic. Using degenerations we define the border rank of a tensor, which roughly captures the minimal rank at which we can approximate the tensor.

Definition 2.16 (Border rank). *The border rank of a tensor V is $\underline{R}(V) = \min\{r : I_r \succeq V\}$.*

Intuitively, $\underline{R}(V) \leq r$ means that V is the limit, as $t \rightarrow 0$, of restrictions of I_r , that is, of tensors of rank at most r .

We use the following standard facts [Bin80, Str87], [BCS97, Chapter 15].

Lemma 2.17 (Basic facts). *For all tensors U, U', V, V', W :*

Restrictions and rank.

- (R1) $U \succeq V$ and $U' \succeq V'$ imply $U \otimes U' \succeq V \otimes V'$ and $U \oplus U' \succeq V \oplus V'$. Reordering the factors of a tensor product gives an isomorphic tensor.
- (R2) $\underline{R}(U) \leq r$ if and only if $I_r \succeq U$. Consequently $U \succeq V$ implies $\underline{R}(V) \leq \underline{R}(U)$.
- (R3) $I_r \otimes U \cong U \oplus \cdots \oplus U$ (r summands). In particular $I_r \otimes I_s \cong I_{rs}$.
- (R4) $\underline{R}(U \otimes V) \leq \underline{R}(U) \underline{R}(V)$ and $\underline{R}(U \oplus V) \leq \underline{R}(U) + \underline{R}(V)$.
- (R5) $I_r \succeq I_s$ for $r \geq s$, and every nonzero tensor U satisfies $U \succeq I_1$.

Flattenings.

- (F1) If $U \succeq V$ then $\text{rank } V^X \leq \text{rank } U^X$, and likewise for V^Y, V^Z .
- (F2) $(U \otimes V)^X$ is the Kronecker product $U^X \otimes V^X$ (after reordering columns), so $\text{rank}(U \otimes V)^X = \text{rank } U^X \cdot \text{rank } V^X$.
- (F3) Every flattening of I_r has rank r .

Degenerations.

- (D1) If $U \succeq V$ then $U \succeq V$, with degree 0. If $U_1 \succeq V_1$ with degree d_1 and $U_2 \succeq V_2$ with degree d_2 , then $U_1 \otimes U_2 \succeq V_1 \otimes V_2$ with degree at most $d_1 + d_2$, and $U_1 \oplus U_2 \succeq V_1 \oplus V_2$ with degree at most $\max(d_1, d_2)$.
- (D2) If $U \succeq V$ and $V \succeq W$, then $U \succeq W$.

(D3) Degenerations do not increase the rank of any flattening.

Since restrictions are degenerations, Lemma 2.17(R2) gives $\underline{R}(V) \leq R(V)$. The next example shows that the inequality can be strict.

Example 2.18 (Border rank can be smaller than rank). Consider the so called W -tensor $W = x_0y_0z_1 + x_0y_1z_0 + x_1y_0z_0$. In $I_2 = a_1b_1c_1 + a_2b_2c_2$ substitute

$$a_1 \mapsto x_0 + tx_1, \quad b_1 \mapsto y_0 + ty_1, \quad c_1 \mapsto t^{-1}(z_0 + tz_1), \quad a_2 \mapsto x_0, \quad b_2 \mapsto y_0, \quad c_2 \mapsto -t^{-1}z_0.$$

Expanding, we obtain the degeneration $W + t(x_0y_1z_1 + x_1y_0z_1 + x_1y_1z_0) + t^2x_1y_1z_1$. So $I_2 \supseteq W$.

On the other hand, $R(W) = 3$. The upper bound is clear. For the lower bound, suppose $W = \sum_{\rho=1}^2 L_\rho M_\rho N_\rho$. At least one L_ρ has a nonzero coefficient of x_1 . Choose c so that substituting $x_1 = cx_0$ makes this linear form vanish. The resulting tensor is

$$x_0(y_0z_1 + y_1z_0 + cy_0z_0).$$

It is not hard to see that the bilinear form in parentheses has rank two, in contradiction.

A special type of degeneration that is both useful and simpler to understand has the following form.

Lemma 2.19 (Weighted degenerations). Apply fixed linear substitutions to each leg of U (for instance a change of basis), and then multiply every new variable v by $t^{w(v)}$, where $w(v) \in \mathbb{Z}$ is some weight function. If every term with a nonzero coefficient has total weight $w(x) + w(y) + w(z) \geq 0$, then letting V be the sum of the terms of total weight 0, we obtain that V is a degeneration of U .

Proof. A term xyz becomes $t^{w(x)+w(y)+w(z)}xyz$. Grouping terms by their total weight $k \geq 0$, the result is $\sum_k t^k V_k$, where V_k consists of the terms of total weight k and $V = V_0$. $\square$

Note that individual weights may be negative, as long as all total weights are nonnegative.

2.4 Matrix multiplication tensors and dot products

The following simple lemma captures the recursive idea in Strassen's algorithm.

Lemma 2.20. (a) $\langle n, m, \ell \rangle \otimes \langle n', m', \ell' \rangle \cong \langle nn', mm', \ell\ell' \rangle$. In particular $M_r \otimes M_s \cong M_{rs}$.

(b) $M_r \geq M_s$ for $r \geq s$, and $M_r \geq I_r$.

Proof. (a) The product of the terms $x_{ij}y_{jk}z_{ki}$ and $x'_{i'j'}y'_{j'k'}z'_{k'i'}$ is $X_{(i,i'),(j,j')} Y_{(j,j'),(k,k')} Z_{(k,k'),(i,i')}$: we pair the row indices, the middle indices, and the column indices. In other words, an $rs \times rs$ matrix is an $r \times r$ matrix whose entries are $s \times s$ matrices, and multiplying these block matrices corresponds to multiplication of the $rs \times rs$ matrices.

(b) For the first claim set to zero all variables with an index at least s . For the second set to zero all variables x_{ij}, y_{ij}, z_{ij} with $i \neq j$, and note that the result is $\sum_i x_{ii}y_{ii}z_{ii}$. $\square$

Observe that the recursion in Proposition 2.3 corresponds to the isomorphism $M_{u^k} \cong M_u^{\otimes k}$ given by Lemma 2.20(a). In particular, Lemma 2.17(R4) gives $R(M_{u^k}) \leq R(M_u)^k$.

Exercise 2.21. Let σ move the legs cyclically ($X \rightarrow Y \rightarrow Z \rightarrow X$). Show that $\langle n, m, \ell \rangle^\sigma \cong \langle \ell, n, m \rangle$, and that $\langle n, m, \ell \rangle$ and $\langle \ell, m, n \rangle$ have the same rank. Conclude that if $R(\langle n, m, \ell \rangle) \leq r$ then $\omega \leq 3 \log r / \log(nm\ell)$.

Dot products. The following three dot-product tensors will play an important role in the proof and reductions:

$$B_X(r) = x \sum_{i=0}^{r-1} y_i z_i, \quad B_Y(r) = y \sum_{i=0}^{r-1} x_i z_i, \quad B_Z(r) = z \sum_{i=0}^{r-1} x_i y_i.$$

As bilinear maps, with outputs labeled by Z variables, $B_X(r)$ multiplies a scalar by a vector of length r and outputs the r results. More generally, if T computes a (multi-output) bilinear map $(x, y) \mapsto T(x, y)$, then $T \otimes B_X(r)$ computes

$$(x, y^{(0)}, \dots, y^{(r-1)}) \mapsto (T(x, y^{(0)}), \dots, T(x, y^{(r-1)})),$$

that is, T applied to r independent second inputs with one *common* first input.

Lemma 2.22. $B_X(r) \cong \langle 1, 1, r \rangle$, $B_Y(r) \cong \langle r, 1, 1 \rangle$ and $B_Z(r) \cong \langle 1, r, 1 \rangle$. Hence

$$B_X(r) \otimes B_Y(r) \otimes B_Z(r) \cong \mathbf{M}_r, \quad (2.2)$$

and

$$\mathbf{M}_n \otimes B_X(r) = \sum_{i,j,k=0}^{n-1} \sum_{u=0}^{r-1} X_{ij} Y_{j,(k,u)} Z_{(k,u),i} \cong \langle n, n, nr \rangle. \quad (2.3)$$

Proof. The first three isomorphisms are clear. The other claims follow from Lemma 2.20(a). $\square$

Equation (2.3) says that $\mathbf{M}_n \otimes B_X(r)$ is the multiplication of one $n \times n$ matrix by an $n \times nr$ matrix. It is important to distinguish this from $I_r \otimes \mathbf{M}_n$, which computes r independent products of $n \times n$ matrices.

2.5 From border rank to algorithms: Bini's theorem

Bini proved that asymptotically, degenerations can be transformed into exact algorithms without changing the exponent. Specifically, it is enough to upper bound the border rank of the matrix multiplication tensor to obtain an upper bound on ω .

Theorem 2.23 (Bini [Bin80]). *If $\underline{\mathbf{R}}(\mathbf{M}_n) \leq r$ with $n \geq 2$, then $\omega \leq \log_n r$.*

The idea is that the error terms depending on t can be removed using interpolation. Under tensor powers, the matrix dimension grows exponentially, whereas the approximation degree grows only linearly. Thus the cost of interpolation does not affect the resulting exponent.

Lemma 2.24 (Interpolation). *If $U \supseteq V$ with degree d , then $I_{d+1} \otimes U \supseteq V$. In particular, $\mathbf{R}(V) \leq (d+1) \mathbf{R}(U)$.*

Proof. We use the notation of Definition 2.15. Pick distinct nonzero numbers $t_0, \dots, t_d$. The Vandermonde matrix $(t_j^k)_{0 \leq j, k \leq d}$ is invertible, so there are numbers c_j with $\sum_j c_j t_j^k = 1$ for $k = 0$ and $\sum_j c_j t_j^k = 0$ for $1 \leq k \leq d$. Then

$$\sum_{j=0}^d c_j U'(t_j) = \sum_{k=0}^d \left(\sum_j c_j t_j^k \right) V_k = V \quad (V_0 = V).$$

It remains to show that this sum is a restriction of $I_{d+1} \otimes U$. By Lemma 2.17(R3), $I_{d+1} \otimes U \cong U^{(0)} \oplus \dots \oplus U^{(d)}$, where $U^{(r)} = \sum u_{ijk} x_i^{(r)} y_j^{(r)} z_k^{(r)}$ is a copy of U in its own variables. Let x', y', z'

be the variables of V , and let $A_r = A(t_r)$, $B_r = B(t_r)$, $C_r = C(t_r)$ be the substitution matrices of the degeneration evaluated at t_r . In $U^{(r)}$ substitute

$$x_i^{(r)} \mapsto c_r(A_r x')_i, \quad y_j^{(r)} \mapsto (B_r y')_j, \quad z_k^{(r)} \mapsto (C_r z')_k.$$

This is one linear substitution on each leg of $U^{(0)} \oplus \dots \oplus U^{(d)}$, and it maps the variables of all the copies to the same variables x', y', z' . It turns $U^{(r)}$ into $c_r U'(t_r)$, and hence the whole direct sum into $\sum_r c_r U'(t_r) = V$. This proves $I_{d+1} \otimes U \geq V$, and the rank bound follows from [Lemma 2.17\(R2\),\(R4\)](#). $\square$

Proof of Theorem 2.23. Let $I_r \supseteq M_n$ with degree d . By [Lemma 2.17\(D1\)](#), $I_{r^N} \cong I_r^{\otimes N} \supseteq M_n^{\otimes N} \cong M_{n^N}$ with degree at most Nd , so [Lemma 2.24](#) gives $R(M_{n^N}) \leq (Nd + 1)r^N$. By [Proposition 2.3](#),

$$\omega \leq \frac{\log((Nd + 1)r^N)}{\log n^N} = \log_n r + \frac{\log(Nd + 1)}{N \log n} \quad \text{for every } N \geq 1.$$

Letting $N \rightarrow \infty$ gives $\omega \leq \log_n r$. $\square$

The factor $Nd + 1$ disappears because it grows only linearly in N (equivalently, $\log(Nd + 1)/N \rightarrow 0$). The fact that polynomial losses do not affect the exponent is used repeatedly in later sections, and especially in the proof of [Theorem 1.2](#).

Lemma 2.25. *If $R(T) \leq r$ and $T^{\otimes m} \supseteq M_n$ with $n \geq 2$, then $\omega \leq m \log r / \log n$.*

Proof. $I_{r^m} \cong I_r^{\otimes m} \supseteq T^{\otimes m} \supseteq M_n$ ([Lemma 2.17](#)), so $\underline{R}(M_n) \leq r^m$ by [\(D2\)](#), and [Theorem 2.23](#) applies. $\square$

Bini's theorem says that as far as the exponent is concerned, approximate algorithms for matrix multiplication are as good as exact ones. As we saw in [Example 2.18](#), approximations can save rank, so this principle is very important for matrix multiplication algorithms, and indeed, all modern algorithms rely on it.

Exercise 2.26 (Asymptotic rank). *Show that the limit $\tilde{R}(T) = \lim_{N \rightarrow \infty} R(T^{\otimes N})^{1/N}$ exists and equals $\inf_N R(T^{\otimes N})^{1/N}$ (this is Fekete's lemma), and that $\tilde{R}(T) = \lim_{N \rightarrow \infty} \underline{R}(T^{\otimes N})^{1/N}$. Show also that $\omega \leq \log_2 \tilde{R}(M_2)$. (Equality holds as well, but this requires showing that general algorithms are not much more efficient than bilinear ones, see [[BCS97](#), Proposition 15.1].)*

Hint. To prove existence of the limit, fix m and write $N = qm + i$ with $0 \leq i < m$. Apply submultiplicativity to the q groups of m factors and the remaining i factors. To compare rank and border rank, fix a degeneration $I_r \supseteq T^{\otimes m}$ and apply interpolation to its k th tensor power. Let k tend to infinity before varying m .

3 Polynomial multiplication, the seed tensor and the rate

3.1 Polynomial multiplication

Let $\mathbb{F}[\xi]_n$ be the space of polynomials of degree less than n in a variable ξ over the field $\mathbb{F}$. If $f = \sum_{i < a} x_i \xi^i \in \mathbb{F}[\xi]_a$ and $g = \sum_{j < b} y_j \xi^j \in \mathbb{F}[\xi]_b$, then $fg \in \mathbb{F}[\xi]_{a+b-1}$. Its k th coefficient, $\sum_{i+j=k} x_i y_j$, is the coefficient of z_k in $C(a, b)$. Thus $C(a, b)$ is the tensor of the bilinear map

$$\mathbb{F}[\xi]_a \times \mathbb{F}[\xi]_b \longrightarrow \mathbb{F}[\xi]_{a+b-1}, \quad (f, g) \longmapsto fg.$$

Its legs have dimensions a , b and $a + b - 1$.

For example, $C(2, 2) = x_0 y_0 z_0 + (x_0 y_1 + x_1 y_0) z_1 + x_1 y_1 z_2$, and Karatsuba's identity

$$C(2, 2) = (x_0 + x_1)(y_0 + y_1)z_1 + x_0 y_0(z_0 - z_1) + x_1 y_1(z_2 - z_1)$$

uses three products instead of four.

Theorem 3.1. $R(C(a, b)) = a + b - 1$.

Proof. Let $n = a + b - 1$ and let $\eta_1, \dots, \eta_n$ be distinct scalars. Let $\mathbf{h} = (h_0, \dots, h_{n-1}) \in \mathbb{F}^n$ be the coefficient vector of $h = fg$ and let $V = (\eta_l^k)_{l,k}$ be the Vandermonde matrix. Then $V\mathbf{h} = (f(\eta_l)g(\eta_l))_l$, and as V is invertible,

$$C(a, b) = \sum_k h_k z_k = \sum_{l=1}^n f(\eta_l) g(\eta_l) \left(\sum_k (V^{-1})_{kl} z_k \right),$$

where $f(\eta_l) = \sum_i x_i \eta_l^i$ and $g(\eta_l) = \sum_j y_j \eta_l^j$ are linear forms. This decomposition has n terms. For the lower bound, note that the coefficient matrices of $z_0, \dots, z_{n-1}$ are nonzero and are supported on the disjoint sets $\{(i, j) : i + j = k\}$, so the Z -flattening has rank n , and [Lemma 2.9](#) gives $R(C(a, b)) \geq n$. $\square$

3.2 The seed tensor

Polynomial multiplication treats its legs asymmetrically: for example, in $C(a, a)$ they have dimensions $a, a, 2a - 1$. As M_n is symmetric under cyclic permutations of its legs ([Exercise 2.21](#)), it is natural to symmetrize. Let σ be the cyclic permutation $X \rightarrow Y \rightarrow Z \rightarrow X$. Applying σ we obtain the polynomial multiplication tensors

$$C(a, a)^\sigma = \sum_{k,l} x_{k+l} y_k z_l, \quad C(a, a)^{\sigma^2} = \sum_{p,r} x_p y_{p+r} z_r,$$

in which the output sits on the X -leg and on the Y -leg, respectively. Their product with $C(a, a)$ is the *seed* tensor

$$K_a = C(a, a) \otimes C(a, a)^\sigma \otimes C(a, a)^{\sigma^2} = \sum_{i,j,k,l,p,r=0}^{a-1} X_{i,k+l,p} Y_{j,k,p+r} Z_{i+j,l,r}. \quad (3.1)$$

Note that by considering the indices of a given term we can tell where it came from: $(i, j) \mapsto i + j$ has its output on the Z -leg, $(k, l) \mapsto k + l$ on the X -leg, and $(p, r) \mapsto p + r$ on the Y -leg. Each leg has dimension $a^2(2a - 1)$. By [Theorem 3.1](#), [Proposition 2.7](#) and [Lemma 2.17\(R4\)](#),

$$R(K_a) \leq (2a - 1)^3. \quad (3.2)$$

It is convenient to symmetrize further, over all six leg permutations:

$$Q(a, b) = \bigotimes_{\pi \in S_3} C(a, b)^\pi. \quad (3.3)$$

This tensor will be the major player in the upcoming proofs.

Remark 3.2 (Why Q and not K). *In the three-sector degeneration, applied to $C(a, 3c + a - 1)$, that we discuss in [Lemma 5.2](#), we obtain three copies of $C(a, c)$, where the middle block has its second and third legs exchanged. Taking all six leg permutations makes the product invariant under this exchange.*

Lemma 3.3. $Q(a, b) \cong Q(b, a)$, $Q(a, b)^\rho \cong Q(a, b)$ for every $\rho \in S_3$, and $Q(a, a) \cong K_a^{\otimes 2}$. In particular $R(Q(a, a)) \leq (2a - 1)^6$.

Proof. Let τ exchange the X - and Y -legs. Renaming $i \leftrightarrow j$ in (1.2) shows that $C(b, a) \cong C(a, b)^\tau$. Hence $Q(b, a) \cong \bigotimes_\pi C(a, b)^{\pi\tau}$ and $Q(a, b)^\rho \cong \bigotimes_\pi C(a, b)^{\rho\pi}$. As π runs over S_3 , so do $\pi\tau$ and $\rho\pi$, and we can reorder the factors. For $a = b$ we have $C(a, a)^\tau \cong C(a, a)$, so $C(a, a)^{\sigma^k\tau} \cong C(a, a)^{\sigma^k}$, and since $S_3 = \{1, \sigma, \sigma^2, \tau, \sigma\tau, \sigma^2\tau\}$ we get $Q(a, a) \cong K_a \otimes K_a$. The rank bound follows from (3.2). $\square$

Lemma 3.4.

$$\bigotimes_{\pi \in S_3} B_X(r)^\pi \cong (B_X(r) \otimes B_Y(r) \otimes B_Z(r))^{\otimes 2} \cong M_{r^2}. \quad (3.4)$$

Proof. $B_X(r)^\pi$ is the dot product whose scalar sits on the leg $\pi(X)$, and each of the three legs equals $\pi(X)$ for exactly two permutations π . The claim follows from (2.2) and the fact that $M_r \otimes M_r \cong M_{r^2}$. $\square$

3.3 The rate of a tensor

Next we define the rate of a tensor U and our function F . Intuitively, the rate measures how large a matrix multiplication tensor high powers of U degenerate to, per copy.

Definition 3.5 (Rate). For a nonzero tensor U put

$$\mu(U) = \sup\{n^{1/m} : m, n \geq 1, U^{\otimes m} \supseteq M_n\}, \quad F(a, b) = \sqrt{\mu(Q(a, b))}.$$

A degeneration $Q(a, b)^{\otimes m} \supseteq M_n$ is called a witness, and $n^{1/(2m)}$ is its rate.

Thus $F(a, b)$ is the supremum of the rates of witnesses for $Q(a, b)$.⁶ In particular, for every $\gamma < F(a, b)$ there is a witness of rate greater than γ .

To see how a witness bounds ω , note that by Lemma 3.3 $R(Q(a, a)) \leq (2a - 1)^6$. Lemma 2.25 shows that a witness $Q(a, a)^{\otimes m} \supseteq M_n$ of rate $\gamma = n^{1/(2m)} > 1$ gives $\omega \leq 6m \log(2a - 1)/(2m \log \gamma)$, that is,

$$\omega \leq \frac{3 \log(2a - 1)}{\log \gamma}. \quad (3.5)$$

We will show how to get large matrix multiplication from high tensor powers of $Q(a, a)$, which will yield $\gamma > a^{4/3}$ in (3.5).

We next prove some basic properties of the rate function.

Lemma 3.6. $\mu(U^{\otimes 2}) = \mu(U)^2$. Consequently $F(a, a) = \mu(K_a)$.

Proof. If $U^{\otimes m} \supseteq M_n$, then $(U^{\otimes 2})^{\otimes m} \supseteq M_n \otimes M_n \cong M_{n^2}$ by (D1), so $\mu(U^{\otimes 2}) \geq (n^2)^{1/m} = (n^{1/m})^2$. Conversely, a degeneration $(U^{\otimes 2})^{\otimes m} = U^{\otimes 2m} \supseteq M_n$ shows that $\mu(U) \geq n^{1/(2m)}$, i.e., $\mu(U)^2 \geq n^{1/m}$. Finally, $F(a, a)^2 = \mu(Q(a, a)) = \mu(K_a^{\otimes 2}) = \mu(K_a)^2$ by Lemma 3.3. $\square$

Exercise 3.7. Let U, V be nonzero tensors. Show that

$$\mu(U \otimes V) \geq \mu(U)\mu(V).$$

Show that the inequality can be strict. More precisely, for every integer $r \geq 2$, prove that

$$\mu(B_X(r)) = \mu(B_Y(r)) = 1, \quad \mu(B_X(r) \otimes B_Y(r)) = \sqrt{r}.$$

⁶Our rate is closely related to the relative exponent of [CVZ21, Definition 2.4] and to the asymptotic relative value of [WZ26]. In the terminology of [WZ26], $2 \log F(a, b)$ is the asymptotic relative value of $Q(a, b)$ in terms of M_2 , that is, the number of copies of M_2 obtained asymptotically per copy of $Q(a, b)$. The relative exponent of [CVZ21] counts in the other direction: for $\max(a, b) \geq 2$, the number of copies of $Q(a, b)$ needed per copy of M_2 is $1/(2 \log F(a, b))$.

The following simple bounds on F will be used in [Section 6](#) to bound the size of the explicit construction.

Lemma 3.8. *There are restrictions $Q(a, b) \geq M_{\max(a, b)^2}$. Moreover,*

$$F(1, b) = b, \quad F(a, b) = F(b, a), \quad \max(a, b) \leq F(a, b) \leq \sqrt{ab(a + b - 1)},$$

and $Q(a, b) \geq I_2$ whenever $(a, b) \neq (1, 1)$.

Proof. Suppose $b \geq a$. In each of the six factors of $Q(a, b)$, set the first-input variables x_i with $i > 0$ to zero. In $C(a, b)$ this leaves $x_0 \sum_{j < b} y_j z_j = B_X(b)$, and in $C(a, b)^\pi$ it leaves $B_X(b)^\pi$, so [\(3.4\)](#) gives M_{b^2} . The case $a \geq b$ follows from $Q(a, b) \cong Q(b, a)$, which also gives $F(a, b) = F(b, a)$. Hence $\mu(Q(a, b)) \geq \max(a, b)^2$.

For the upper bound, note that each $C(a, b)^\pi$ has leg dimensions $a, b, a + b - 1$ in some order, and each of these lands on each leg for two of the six permutations. Hence every leg of $Q(a, b)$ has dimension $[ab(a + b - 1)]^2$, and every leg of $Q(a, b)^{\otimes m}$ has dimension $[ab(a + b - 1)]^{2m}$. A flattening of M_n has rank n^2 ([Lemma 2.10](#)), and degenerations do not increase the ranks of flattenings ([D3](#)), so $Q(a, b)^{\otimes m} \supseteq M_n$ implies $n^2 \leq [ab(a + b - 1)]^{2m}$, i.e., $n^{1/(2m)} \leq \sqrt{ab(a + b - 1)}$. For $a = 1$ the two bounds coincide.

Finally, if $\max(a, b) \geq 2$ then $Q(a, b) \geq M_4 \geq M_2 \geq I_2$ ([Lemma 2.20\(b\)](#)). $\square$

Exercise 3.9. *Show that [\(3.5\)](#), together with the upper bound of [Lemma 3.8](#), can never give a bound below 2. Show that if $F(a, a) \geq a^{3/2 - o(1)}$ then [\(3.5\)](#) gives $\omega = 2$, and that the lower bound $F(a, a) \geq a$ gives nothing better than $\omega \leq 3$.*

4 Proving the main theorem

In this section we prove [Theorem 1.1](#), assuming two inequalities for the rate, which we prove in [Section 5](#).

4.1 The basic inequalities

The proof is based on two inequalities that enable lower bounding $F(a, b)$. The first is a slight generalization of [[Ope26a](#), Lemma 4.1], which proves it for $s = 1$. We prove both in [Section 5](#).

Theorem 4.1 (Discrete concavity [[Ope26a](#), Lemma 4.1]). *For all integers $a \geq 1$ and $1 \leq s < b$,*

$$2F(a, b) \geq F(a, b - s) + F(a, b + s).$$

Theorem 4.2 (Shifted tripling [[Ope26a](#), Lemma 4.2]). *For all integers $a, c \geq 1$,*

$$F(a, 3c + a - 1) \geq 3F(a, c).$$

We next show that these two inequalities, when combined with [Lemma 3.8](#), force $F(a, a)$ to be large. Consider $F(2, 2)$. Tripling with $a = c = 2$ gives $F(2, 7) \geq 3F(2, 2)$. Viewing F as a function on $\mathbb{N}^2$, the five increments in row 2 from column 2 to column 7 add up to at least $2F(2, 2)$. By concavity, each of them is at most the increment from column 1 to column 2, which is $F(2, 2) - F(2, 1) = F(2, 2) - 2$. Hence

$$2F(2, 2) \leq F(2, 7) - F(2, 2) \leq 5(F(2, 2) - 2), \quad \text{so} \quad F(2, 2) \geq \frac{10}{3} = \Lambda_2.$$

The same reasoning works along the whole diagonal. The general case is captured in the following lemma. The lemma only requires $s = 1$ in [Theorem 4.1](#) and we will use the general case later when we construct a sequence of algorithms.

Lemma 4.3 (Diagonal-growth argument of [[Ope26a](#), Lemma 5.1]). *Let $A \geq 2$, and let $\Phi(a, b)$, for $1 \leq a, b \leq 4A - 1$, be real numbers with $\Phi(a, b) = \Phi(b, a)$ and $\Phi(1, b) = b$. Suppose that for every $2 \leq a \leq A$,*

$$2\Phi(a, b) \geq \Phi(a, b-1) + \Phi(a, b+1) \quad (a \leq b \leq 4a-2) \quad \text{and} \quad \Phi(a, 4a-1) \geq 3\Phi(a, a).$$

Then, for every $2 \leq a \leq A$,

$$\Phi(a, a) \geq \Lambda_a > a^{4/3}. \tag{4.1}$$

The proof of [Theorem 1.1](#) immediately follows.

Proof of Theorem 1.1. For $a = 1$, [Lemma 3.8](#) gives $F(1, 1) = 1 = \Lambda_1$. For $a \geq 2$, the initial bounds, concavity and shifted tripling show that F satisfies the hypotheses of [Lemma 4.3](#). Hence $F(a, a) \geq \Lambda_a > a^{4/3}$.

For every $1 < \gamma < \Lambda_a$, there is a witness $Q(a, a)^{\otimes m} \succeq M_n$ with $n^{1/(2m)} > \gamma$. By (3.5),

$$\omega \leq \frac{3 \log(2a-1)}{\log \gamma}.$$

Letting γ approach Λ_a gives

$$\omega \leq \frac{3 \log(2a-1)}{\log \Lambda_a} < \frac{9}{4} \frac{\log(2a-1)}{\log a}.$$

The right-hand side tends to $9/4$ as $a \rightarrow \infty$. □

4.2 The diagonal-growth argument

Before giving the formal proof of [Lemma 4.3](#), let us explain informally why concavity and shifted tripling force $F(a, a)$ to grow (at least) like $a^{4/3}$. Fix the first argument a and think of $F(a, b)$ as a function of b . Shifted tripling with $c = a$ tells us that $F(a, 4a-1) \geq 3F(a, a)$, that is, along the $3a-1$ steps from $b = a$ to $b = 4a-1$ the function increases by at least $2F(a, a)$. The average increment is therefore at least $2F(a, a)/(3a-1)$.

Concavity says that these increments never increase as b grows. Hence the first increment, from $b = a$ to $b = a+1$, is at least their average. Indeed, if it were smaller, then all the later increments would be smaller as well, and their sum could not reach $2F(a, a)$. Applying concavity once more, the increment from $b = a-1$ to $b = a$ is at least as large. Thus both steps next to the diagonal increase F by at least roughly $2F(a, a)/(3a)$.

For example, for $a = 2$ we have $F(2, 1) = 2$, and the increment $F(2, 2) - F(2, 1)$ must be at least $\frac{2}{5}F(2, 2)$. Rearranging gives $F(2, 2) \geq 10/3$, which is the bound we obtained in [Section 4.1](#).

Now move along the diagonal from $(a-1, a-1)$ to (a, a) . This changes both arguments, and, by the symmetry $F(a, b) = F(b, a)$, each change contributes at least roughly $2F/(3a)$. Hence $F(a, a)$ roughly increases by at least $\frac{4}{3} \cdot F(a, a)/a$ in each step. In other words, $F(a, a)/F(a-1, a-1)$ is roughly at least $1 + \frac{4}{3a}$, and multiplying these ratios along the diagonal gives

$$F(a, a) \gtrsim \prod_{i=2}^a \left(1 + \frac{4}{3i}\right) \approx a^{4/3}.$$

Lemma 4.3 makes this calculation precise. To see what this gives for ω , recall that $F(a, a) = \mu(K_a)$ (Lemma 3.6) is the matrix dimension that we extract per copy of the seed K_a , and that $R(K_a) \leq (2a - 1)^3$. By (3.5), a witness of rate γ gives $\omega \leq 3 \log(2a - 1) / \log \gamma$. Taking γ close to $a^{4/3}$ we get

$$\omega \lesssim \frac{3 \log(2a - 1)}{\frac{4}{3} \log a} = \frac{9}{4} \cdot \frac{\log(2a - 1)}{\log a},$$

which tends to $9/4$ as $a \rightarrow \infty$, since $\log(2a - 1) / \log a \rightarrow 1$.

We note that the $2/3$ in the step size $2F(a, a)/(3a)$ comes from the tripling inequality: we gain $2F(a, a)$ over about $3a$ steps. A different decomposition of polynomial multiplication might give a stronger rate inequality and improve this lower bound. To get $\omega = 2$ we would need $F(a, a) \geq a^{3/2-o(1)}$ (Exercise 3.9). Exercise 4.4 shows that for $a = 2$ the two inequalities, in the form used in Lemma 4.3, cannot give more than $F(2, 2) \geq 10/3$.

Proof of Lemma 4.3. By concavity, the successive differences in each row are nonincreasing. Let $1 \leq a < A$. The first difference between columns a and $4a - 1$ is therefore at least their average. Using tripling, we get

$$\Phi(a, a + 1) - \Phi(a, a) \geq \frac{\Phi(a, 4a - 1) - \Phi(a, a)}{3a - 1} \geq \frac{2\Phi(a, a)}{3a - 1}$$

(for $a = 1$ this holds since $\Phi(1, b) = b$). In particular

$$\Phi(a, a + 1) \geq \frac{3a + 1}{3a - 1} \Phi(a, a).$$

The same bound holds in row $a + 1$, and by concavity again the difference immediately before the diagonal is no smaller:

$$\Phi(a + 1, a + 1) - \Phi(a + 1, a) \geq \Phi(a + 1, a + 2) - \Phi(a + 1, a + 1) \geq \frac{2\Phi(a + 1, a + 1)}{3a + 2}.$$

Thus, by symmetry,

$$\frac{3a + 1}{3a - 1} \Phi(a, a) \leq \Phi(a, a + 1) = \Phi(a + 1, a) \leq \frac{3a}{3a + 2} \Phi(a + 1, a + 1).$$

Rearranging, we obtain

$$\Phi(a + 1, a + 1) \geq \frac{(3a + 1)(3a + 2)}{3a(3a - 1)} \Phi(a, a).$$

Starting with $\Phi(1, 1) = 1$ and multiplying these inequalities, we conclude that for every $2 \leq a \leq A$,

$$\Phi(a, a) \geq \prod_{i=1}^{a-1} \frac{(3i + 1)(3i + 2)}{3i(3i - 1)} = \frac{3a - 1}{2} \prod_{i=1}^{a-1} \left(1 + \frac{1}{3i}\right) = \Lambda_a,$$

where the product of the factors $(3i + 2)/(3i - 1)$ telescopes.

Finally,

$$\left(1 + \frac{1}{3i}\right)^3 = 1 + \frac{1}{i} + \frac{1}{3i^2} + \frac{1}{27i^3} > 1 + \frac{1}{i},$$

so multiplying over $1 \leq i < a$ gives

$$\prod_{i=1}^{a-1} \left(1 + \frac{1}{3i}\right) > a^{1/3} \quad (a \geq 2). \quad (4.2)$$

Since $(3a - 1)/2 \geq a$, we have $\Lambda_a > a^{4/3}$ for every $a \geq 2$. □

Exercise 4.4 (The diagonal-growth bound is tight for $A = 2$). Find numbers $\Phi(a, b)$, $1 \leq a, b \leq 7$, satisfying the hypotheses of [Lemma 4.3](#) with $A = 2$ and $\Phi(2, 2) = 10/3$. Conclude that concavity and tripling as used in that lemma cannot prove $F(2, 2) > 10/3$.

Hint. Define $\Phi(2, b) = 2 + \frac{4}{3}(b - 1)$.

Exercise 4.5 (A three-line bound). Use [Theorems 4.1](#) and [4.2](#) and the symmetry $F(a, b) = F(b, a)$ to show that for every integer $c \geq 1$,

$$F(4c - 1, 4c - 1) \geq 6F(c, c).$$

Conclude that $F(a_k, a_k) \geq 6^k$ for $a_k = (2 \cdot 4^k + 1)/3$, and hence that $\omega \leq 3 \log 4 / \log 6 = 6 / \log_2 6 < 2.33$.

Hint. Let $a = 4c - 1$ and $s = 3c - 1$, so that $a - s = c$ and $a + s = 3c + a - 1$. Apply [Theorem 4.1](#) and then use tripling in row a and symmetry.

5 Proving the inequalities

5.1 Ideas behind the proofs

Let us say a few words about the ideas behind the proofs of [Theorems 4.1](#) and [4.2](#).

For [Theorem 4.1](#), consider the case $s = 1$ for simplicity. The idea is to consider two polynomial products with the same first input, represented by $C(a, b) \otimes B_X(2)$ and to show a degeneration to $C(a, b + 1) + C(a, b - 1)$. To see why this can be useful we note that by [Lemmas 3.4](#) and [3.8](#) we have

$$Q(2, 2) \geq M_4, \quad Q(2, 3) \geq M_9, \quad Q(2, 1) \cong M_4.$$

So changing the lengths $(2, 2)$ to $(2, 3)$ improves the extraction, while changing them to $(2, 1)$ does not hurt it.

The problem is that the two tensors composing $C(a, b + 1) + C(a, b - 1)$ share the first leg. That is, as polynomial multiplication tensors we have the map $(f, g, h) \rightarrow (fg, fh)$ where $\deg(f) < a$, $\deg(g) < b + 1$, $\deg(h) < b - 1$. The new idea is that by investing a few copies of the unit tensor we can separate the two dependent tensors into a direct sum and additionally obtain for free another copy of $B_X(M)$ (which is itself a matrix multiplication tensor). This is captured in the following important separation lemma.

Lemma 5.1 (Shared-leg separation; [[Ope26a](#), Proposition 3.1]). Let $M \geq 1$ and $T = \sum_{i=0}^{M-1} T_i$. Suppose that the Y -variables are partitioned into groups $Y_0, \dots, Y_{M-1}$, the Z -variables into groups $Z_0, \dots, Z_{M-1}$, and that T_i involves only the X -variables and the variables in Y_i and Z_i . Then there is an explicit weighted degeneration, of degree at most $(M - 1)^2$,

$$I_{2M} \otimes T \geq \bigoplus_{i=0}^{M-1} (T_i \otimes B_X(M)). \quad (5.1)$$

To see how we apply this lemma note that by taking high tensor powers of $C(a, b)$ and symmetrizing, we obtain a degeneration

$$I_{(2M)^6} \otimes Q(a, b)^{\otimes N} \otimes M_{4^N} \geq I_{M^6} \otimes Q(a, b + 1)^{\otimes N_+} \otimes Q(a, b - 1)^{\otimes N_-} \otimes M_{M^2}, \quad (5.2)$$

where $M = \binom{N}{N_+}$ for any $N_+ + N_- = N$, from which we deduce [Theorem 4.1](#).

The proof of [Theorem 4.2](#) partitions the second-input and output variables into three intervals and uses a degeneration to remove terms joining different intervals. Specifically, we prove

Lemma 5.2 (Three-sector degeneration; [Ope26a, proof of Lemma 4.2]). *For integers $a, c \geq 1$, there is an explicit weighted degeneration of degree at most one,*

$$\mathsf{C}(a, 3c + a - 1) \supseteq T_0 + T_1 + T_2,$$

in which the three blocks use the same X -variables and disjoint Y - and Z -variables. The outer blocks T_0, T_2 are copies of $\mathsf{C}(a, c)$. The middle block T_1 is a copy of $\mathsf{C}(a, c)$ with its second and third legs exchanged and its first-input indices reversed.

Then, symmetrizing over the six permutations of the legs (to compensate for the exchange of legs), taking high powers and using again Lemma 5.1, we obtain a degeneration

$$I_{(2M)^6} \otimes \mathsf{Q}(a, 3c + a - 1)^{\otimes N} \supseteq I_{M^6} \otimes \mathsf{Q}(a, c)^{\otimes N} \otimes \mathsf{M}_{M^2}, \quad (5.3)$$

where N is a multiple of 3, and $M = N!/((N/3)!)^3$, from which Theorem 4.2 follows.

To deduce Theorem 4.1 and Theorem 4.2 from (5.2) and (5.3) we note that the two degenerations allow us to “reuse” part of the invested tensors $I_{(2M)^6} \otimes \mathsf{M}_{4N}$ and $I_{(2M)^6}$, respectively. This allows us to repeat the construction without having to “invest” large tensors again. This is captured in the next simple *reuse* lemma.

Lemma 5.3 (Reusing the auxiliary tensor). *Let $U \geq I_2$, $E \neq 0$, and u, p, r be positive integers. If*

$$I_u \otimes U^{\otimes p} \otimes E \supseteq \mathsf{M}_r \otimes E,$$

then $\mu(U) \geq r^{1/(p + \lceil \log_2 u \rceil)}$.

The proof can be thought of as an amortized-cost argument: we first prepare E using some fixed number of copies of U , and then, in each round, we use $p + \lceil \log_2 u \rceil$ additional copies of U to produce a factor M_r , while returning E for the next round. Thus, the cost of preparing E is paid only once, and by running sufficiently many rounds its amortized cost per round becomes negligible. In other words, we essentially pay $p + \lceil \log_2 u \rceil$ copies of U for each factor M_r .

Proof. Let $q = p + \lceil \log_2 u \rceil$. Since $U^{\otimes(q-p)} \geq I_{2^{q-p}} \geq I_u$, (D1) and (D2) give $U^{\otimes q} \otimes E \supseteq \mathsf{M}_r \otimes E$. By induction on k ,

$$U^{\otimes kq} \otimes E = U^{\otimes q} \otimes (U^{\otimes(k-1)q} \otimes E) \supseteq U^{\otimes q} \otimes E \otimes \mathsf{M}_{r^{k-1}} \supseteq \mathsf{M}_r \otimes E \otimes \mathsf{M}_{r^{k-1}} \cong \mathsf{M}_{r^k} \otimes E.$$

Choose ℓ with $2^\ell \geq R(E)$. Then $U^{\otimes \ell} \geq I_{2^\ell} \geq E$, and $E \geq I_1$. Hence $U^{\otimes(kq+\ell)} \supseteq \mathsf{M}_{r^k}$, and so $\mu(U) \geq r^{k/(kq+\ell)}$ for every k . The result follows from letting k go to infinity. $\square$

5.2 Extracting specific factors from tensor powers

In this subsection we prove Lemma 5.1 and show how to use it in order to isolate specific terms in tensor powers. We note that selecting products with prescribed numbers of factors is a standard step in tensor-power arguments that goes back to Schönhage’s theorem [Sch81] and Strassen’s laser method [Str87] (see also [Blä13, Section 9.3]). To ease the reading we repeat the statement of the lemma.

Lemma 5.1 (Shared-leg separation; [Ope26a, Proposition 3.1]). *Let $M \geq 1$ and $T = \sum_{i=0}^{M-1} T_i$. Suppose that the Y -variables are partitioned into groups $Y_0, \dots, Y_{M-1}$, the Z -variables into groups $Z_0, \dots, Z_{M-1}$, and that T_i involves only the X -variables and the variables in Y_i and Z_i . Then there is an explicit weighted degeneration, of degree at most $(M-1)^2$,*

$$I_{2M} \otimes T \supseteq \bigoplus_{i=0}^{M-1} (T_i \otimes B_X(M)). \quad (5.1)$$

Proof. Let $\zeta = \zeta_{2M}$ be a primitive $2M$ th root of unity. Write $x_u, y_{i,v}$ and $z_{i,w}$ for the variables of T , where $y_{i,v} \in Y_i$ and $z_{i,w} \in Z_i$. The factor I_{2M} supplies $2M$ independent copies of T , indexed by $\rho = 0, \dots, 2M - 1$. In copy ρ substitute

$$\begin{aligned} x_u^{(\rho)} &\mapsto \sum_{j=0}^{M-1} \zeta^{\rho j} X_{u,j}, \\ y_{i,v}^{(\rho)} &\mapsto \sum_{k=0}^{M-1} \zeta^{\rho(k-i)} Y_{i,v,k}, \\ z_{i,w}^{(\rho)} &\mapsto \frac{1}{2M} \sum_{l=0}^{M-1} \zeta^{-\rho l} Z_{i,w,l}. \end{aligned}$$

All copies are mapped into the same new variables, so this is a restriction of $I_{2M} \otimes T$. A term $x_u y_{i,v} z_{i,w}$ of T produces the new term $X_{u,j} Y_{i,v,k} Z_{i,w,l}$ with coefficient

$$\frac{1}{2M} \sum_{\rho=0}^{2M-1} \zeta^{\rho(j+(k-i)-l)}$$

times its original coefficient. This is 1 if $k - l + (j - i)$ is divisible by $2M$, and 0 otherwise. Since $|k - l + j - i| \leq 2(M - 1) < 2M$, the surviving terms are exactly those with

$$k - l + j - i = 0. \tag{5.4}$$

Now multiply each new variable by a power of t : multiply $X_{u,j}$ by t^{j^2} , $Y_{i,v,k}$ by t^{2ik-i^2} , and $Z_{i,w,l}$ by t^{-2il} . Each exponent uses only the indices of its own variable. On every surviving term, by (5.4), the total exponent is

$$j^2 + (2ik - i^2) - 2il = j^2 - i^2 + 2i(k - l) = j^2 - i^2 - 2i(j - i) = (j - i)^2 \geq 0.$$

Hence the transformed tensor is a polynomial in t of degree at most $(M - 1)^2$, even though some individual exponents are negative. Its constant coefficient consists of the terms with $j = i$, and for them (5.4) gives $k = l$. Let us write out what remains. Denote $T_i = \sum_{u,v,w} c_{i,uvw} x_u y_{i,v} z_{i,w}$. A term of T_i produced the new terms $X_{u,j} Y_{i,v,k} Z_{i,w,l}$, and the surviving ones are those with $j = i$ and $k = l$. Hence the limit tensor is

$$\sum_{i=0}^{M-1} \sum_{u,v,w} c_{i,uvw} \sum_{k=0}^{M-1} X_{u,i} Y_{i,v,k} Z_{i,w,k}.$$

Observe that for each fixed i , $\sum_{u,v,w} c_{i,uvw} \sum_{k=0}^{M-1} X_{u,i} Y_{i,v,k} Z_{i,w,k}$ is isomorphic to $T_i \otimes B_X(M)$. Since every variable of this sum carries the index i , different blocks use disjoint variables on all three legs. Thus, the limit tensor is isomorphic to the direct sum $\bigoplus_i (T_i \otimes B_X(M))$. $\square$

The number $2M$ of copies is smaller than the $5M$ used in [Ope26a, Proposition 3.1]. We note that the proof of Lemma 5.1 would work also with $2M - 1$ copies rather than $2M$.

Exercise 5.4 (Separation needs at least M copies). *Suppose that for some L , $I_L \otimes T \supseteq \bigoplus_i (T_i \otimes B_X(M))$ holds for every $T = \sum_{i < M} T_i$ as in Lemma 5.1. Show that $L \geq M$.*

We now show how to use Lemma 5.1 to keep specific terms in a tensor power. We shall use an explicit index set $\mathcal{I}$ for indexing the components of T . Let $T = \sum_{i \in \mathcal{I}} T_i$ satisfy the assumptions of

Lemma 5.1: the blocks share their X -variables and use disjoint groups Y_i, Z_i on the other two legs. Write

$$T^{\otimes N} = \sum_{w \in \mathcal{I}^N} T_w, \quad T_w = T_{w_1} \otimes \cdots \otimes T_{w_N}.$$

The groups of the coordinates of a Y -variable determine a word $w \in \mathcal{I}^N$, and similarly for a Z -variable. The *type* of w is its vector of letter counts $(N_i)_{i \in \mathcal{I}}$.

Lemma 5.5 (Extracting a fixed type class). *Let $(N_i)_{i \in \mathcal{I}}$ be nonnegative integers with sum N , and put $M = N! / \prod_{i \in \mathcal{I}} N_i!$. Then there is an explicit degeneration*

$$I_{2M} \otimes T^{\otimes N} \supseteq I_M \otimes \left(\bigotimes_{i \in \mathcal{I}} T_i^{\otimes N_i} \right) \otimes B_X(M). \quad (5.5)$$

Proof. On the Y - and Z -legs, set to zero every variable whose word does not have type (N_i) . This keeps exactly M blocks. By Lemma 5.1, using $2M$ copies makes these blocks fully independent and attaches $B_X(M)$ to each of them. Reordering the tensor factors within each block identifies it with

$$\left(\bigotimes_{i \in \mathcal{I}} T_i^{\otimes N_i} \right) \otimes B_X(M).$$

The M independent copies give the factor I_M . □

In both proofs below, we apply this lemma in all six orientations. The six factors I_M combine into I_{M^6} . Also, each orientation of the dot product occurs twice, so (3.4) gives

$$\bigotimes_{\pi \in S_3} B_X(M)^\pi \cong (B_X(M) \otimes B_Y(M) \otimes B_Z(M))^{\otimes 2} \cong \mathbf{M}_{M^2}.$$

5.3 Discrete concavity

We prove Theorem 4.1 by first constructing a degeneration between polynomial multiplication tensors of related lengths.⁷

Starting with two products fg_0 and fg_1 , where $f \in \mathbb{F}[\xi]_a$ and $g_0, g_1 \in \mathbb{F}[\xi]_b$, we merge the second inputs into $g_0 + \xi^s g_1 \in \mathbb{F}[\xi]_{b+s}$. The pairs in the kernel of this map have the form $(-\xi^s h, h)$ with $h \in \mathbb{F}[\xi]_{b-s}$. We use a degeneration to separate multiplication on the kernel from the induced multiplication on the direct complement.

Lemma 5.6 (Midpoint degeneration). *For integers $a \geq 1$ and $1 \leq s < b$, there is a weighted degeneration of degree at most one, with integer coefficients,*

$$\mathbf{C}(a, b) \otimes B_X(2) \supseteq \mathbf{C}(a, b+s) + \mathbf{C}(a, b-s),$$

where the two blocks share the X -variables and have disjoint Y - and Z -variables.

Proof. The tensor $\mathbf{C}(a, b) \otimes B_X(2)$ computes the bilinear map

$$(f, (g_0, g_1)) \longmapsto (fg_0, fg_1),$$

where $f \in \mathbb{F}[\xi]_a$ and $g_0, g_1 \in \mathbb{F}[\xi]_b$. Fix f and consider the linear map

$$T_f : \mathbb{F}[\xi]_b \oplus \mathbb{F}[\xi]_b \longrightarrow \mathbb{F}[\xi]_{a+b-1} \oplus \mathbb{F}[\xi]_{a+b-1}, \quad T_f(g_0, g_1) = (fg_0, fg_1).$$

⁷In [Ope26a] these were neighboring tensors $\mathbf{C}(a, b)$ and $\mathbf{C}(a, b \pm 1)$. Here we take larger steps and consider $\mathbf{C}(a, b \pm s)$.

We shall choose bases, independently of f , in which all these maps have the same block triangular form. For $n > s$, define the merging map

$$\mu_n : \mathbb{F}[\xi]_n \oplus \mathbb{F}[\xi]_n \longrightarrow \mathbb{F}[\xi]_{n+s}, \quad \mu_n(g_0, g_1) = g_0 + \xi^s g_1.$$

It is easy to see that this map is onto, and

$$\ker \mu_n = \{(-\xi^s h, h) : h \in \mathbb{F}[\xi]_{n-s}\}.$$

Choose a basis whose first $n + s$ elements are

$$(\xi^j, 0) \quad (0 \leq j < n), \quad (0, \xi^j) \quad (n - s \leq j < n).$$

Their images under μ_n are the monomial basis of $\mathbb{F}[\xi]_{n+s}$. Complete this basis with the kernel vectors

$$(-\xi^{j+s}, \xi^j) \quad (0 \leq j < n - s).$$

Use these bases for the second-input space, with $n = b$, and for the output space, with $n = a + b - 1$. Multiplication by f respects the kernel, since

$$T_f(-\xi^s h, h) = (-\xi^s f h, f h),$$

and commutes with merging:

$$\mu_{a+b-1}(T_f(g_0, g_1)) = \mu_{a+b-1}(f g_0, f g_1) = f \mu_b(g_0, g_1).$$

Consequently, in the chosen bases the matrix of T_f has the form

$$\begin{pmatrix} A_f & 0 \\ C_f & R_f \end{pmatrix}.$$

Here A_f is multiplication by f from $\mathbb{F}[\xi]_{b+s}$ to $\mathbb{F}[\xi]_{a+b+s-1}$, and R_f is multiplication by f from $\mathbb{F}[\xi]_{b-s}$ to $\mathbb{F}[\xi]_{a+b-s-1}$. The zero block reflects the fact that multiplication maps the kernel into the kernel.

It remains to remove C_f . Multiply the Y -variables corresponding to the kernel by t^{-1} and the Z -variables corresponding to the kernel by t , leaving all other variables unchanged. The matrix becomes

$$\begin{pmatrix} A_f & 0 \\ tC_f & R_f \end{pmatrix}.$$

Thus, this is a weighted degeneration of degree at most one. Its constant term consists of $\mathbb{C}(a, b + s)$ and $\mathbb{C}(a, b - s)$. They share the first input f , while their second-input and output variables are disjoint.

Finally, the basis matrices above have integer entries and determinant ± 1 , so their inverses also have integer entries. This proves the claim about the coefficients. $\square$

We note that the insistence on integer coefficients comes from the wish to extend the result to any characteristic. See [Section 7](#).

Exercise 5.7 (Why a degeneration is needed). *Let*

$$U = \mathbb{C}(2, 2) \otimes B_X(2), \quad V = \mathbb{C}(2, 3) + \mathbb{C}(2, 1),$$

where the two blocks of V share their X -variables and use disjoint Y - and Z -variables. By [Lemma 5.6](#), $U \supseteq V$. Show that there is no restriction $U \geq V$. Hint. First show that both tensors have flattening ranks $(2, 4, 6)$, so a restriction would have to be an isomorphism. Fixing the first input f gives a linear map from the second-input space to the output space. For three pairwise nonproportional first inputs f_1, f_2, f_3 , compare the dimensions of the intersections of the images of these three maps.

Example 5.8. Let $a = b = 2$, $f = x_0 + x_1\xi$, $K = \ker \mu_2$ and $K' = \ker \mu_3$. The complement of K has basis $(1, 0), (\xi, 0), (0, \xi)$, and K is spanned by $(-\xi, 1)$. In the output space $\mathbb{F}[\xi]_3 \oplus \mathbb{F}[\xi]_3$, use the complement basis $(1, 0), (\xi, 0), (\xi^2, 0), (0, \xi^2)$ and the kernel basis $(-\xi, 1), (-\xi^2, \xi)$.

The only nonzero entry of C_f comes from the input vector $(0, \xi)$, since

$$T_f(0, \xi) = (0, x_0\xi + x_1\xi^2) = x_0(\xi^2, 0) + x_1(0, \xi^2) + x_0(-\xi^2, \xi).$$

The last summand lies in K' and gives the entry x_0 of C_f . The degeneration removes this entry, leaving $C(2, 3)$ and $C(2, 1)$ with a common first input.

Now that we have the appropriate construction we can prove [Theorem 4.1](#).

Theorem 4.1 (Discrete concavity [[Ope26a](#), Lemma 4.1]). For all integers $a \geq 1$ and $1 \leq s < b$,

$$2F(a, b) \geq F(a, b - s) + F(a, b + s).$$

The idea is simple. Denote $T_+ = C(a, b + s)$ and $T_- = C(a, b - s)$. Then the expansion of $(T_+ + T_-)^{\otimes 3}$ contains the three products

$$T_+ \otimes T_+ \otimes T_-, \quad T_+ \otimes T_- \otimes T_+, \quad T_- \otimes T_+ \otimes T_+,$$

which are all isomorphic to $T_+^{\otimes 2} \otimes T_-$, as they differ only in the order of their factors. Similarly, in the N th power there are $\binom{N}{N_+}$ products that contain N_+ copies of T_+ and $N_- = N - N_+$ copies of T_- . We pick some fixed N_+ and keep only these products (by setting variables corresponding to different terms to zero) and apply [Lemma 5.1](#) to separate these $\binom{N}{N_+}$ products, making them independent. In addition, this attaches a dot product to each of these products. As we shall see, after balancing the three legs (tensoring all six permutations) these dot products give an additional matrix dimension.

Proof. We first fix witnesses for the two tensors $Q(a, b \pm s)$ (recall, these are degenerations whose rates can be taken arbitrarily close to $F(a, b \pm s)$). By taking tensor powers we may assume that both witnesses use the same number m of copies (if they use m_1 and m_2 copies, raise them to the powers m_2 and m_1 , respectively). So let

$$Q(a, b + s)^{\otimes m} \supseteq M_{n_+}, \quad Q(a, b - s)^{\otimes m} \supseteq M_{n_-},$$

where n_+ and n_- are the sizes of the matrix products that we get from $Q(a, b + s)$ and $Q(a, b - s)$, respectively, and denote their rates by

$$\alpha = n_+^{1/(2m)}, \quad \beta = n_-^{1/(2m)}.$$

We shall prove that $F(a, b) \geq (\alpha + \beta)/2$. As the rates of witnesses can be taken arbitrarily close to $F(a, b + s)$ and $F(a, b - s)$, this proves the theorem. Note that by [Lemma 3.8](#) we may assume that $\alpha \geq b + s \geq 3$, and in particular $\alpha + \beta > 2$.

The idea is to take a high power of the degeneration of [Lemma 5.6](#), and to keep only the products that contain a prescribed number N_+ of copies of $C(a, b + s)$ and N_- of copies of $C(a, b - s)$ (chosen

in a way that maximizes the resulting rate). As we shall see, the best choice is to take N_+/N close to $\alpha/(\alpha + \beta)$. This is similar to the idea behind the proof of Schönhage's theorem mentioned in the introduction [Sch81].

Recall that $Q(a, b) = \bigotimes_{\pi \in S_3} C(a, b)^\pi$ is the tensor product of the six copies of $C(a, b)$ obtained by permuting its legs (see (3.3)). We call these the six orientations. Applying a permutation π of the legs to the degeneration of Lemma 5.6 gives

$$C(a, b)^\pi \otimes B_X(2)^\pi \supseteq C(a, b + s)^\pi + C(a, b - s)^\pi,$$

where now the two blocks share the leg $\pi(X)$. Taking the tensor product over the six orientations, and using Lemma 3.4 for the six dot products $B_X(2)^\pi$, we get

$$Q(a, b) \otimes M_4 \cong \bigotimes_{\pi \in S_3} (C(a, b)^\pi \otimes B_X(2)^\pi) \supseteq \bigotimes_{\pi \in S_3} (C(a, b + s)^\pi + C(a, b - s)^\pi).$$

For an arbitrary integer $N \geq 1$, taking the N th power gives

$$Q(a, b)^{\otimes N} \otimes M_{4N} \supseteq \bigotimes_{\pi \in S_3} (C(a, b + s)^\pi + C(a, b - s)^\pi)^{\otimes N}. \quad (5.6)$$

Choose nonnegative integers N_+, N_- with $N_+ + N_- = N$, and put

$$M = \binom{N}{N_+}.$$

Fix an orientation π . The two blocks $C(a, b + s)^\pi$ and $C(a, b - s)^\pi$ share the leg $\pi(X)$ and use disjoint variables on the other two legs. Thus Lemma 5.5, with $\pi(X)$ as the shared leg and with counts (N_+, N_-) , gives

$$I_{2M} \otimes (C(a, b + s)^\pi + C(a, b - s)^\pi)^{\otimes N} \supseteq I_M \otimes (C(a, b + s)^\pi)^{\otimes N_+} \otimes (C(a, b - s)^\pi)^{\otimes N_-} \otimes B_X(M)^\pi.$$

We apply this in each of the six orientations and take the tensor product. On the left-hand side this uses $I_{2M}^{\otimes 6} \cong I_{(2M)^6}$, and on the right-hand side it returns I_{M^6} . Since reordering the factors of a tensor product gives an isomorphic tensor,

$$\bigotimes_{\pi \in S_3} \left((C(a, b + s)^\pi)^{\otimes N_+} \otimes (C(a, b - s)^\pi)^{\otimes N_-} \right) \cong Q(a, b + s)^{\otimes N_+} \otimes Q(a, b - s)^{\otimes N_-},$$

and by Lemma 3.4 the six dot products $B_X(M)^\pi$ give M_{M^2} . Combining this with (5.6) we get

$$I_{(2M)^6} \otimes Q(a, b)^{\otimes N} \otimes M_{4N} \supseteq I_{M^6} \otimes Q(a, b + s)^{\otimes N_+} \otimes Q(a, b - s)^{\otimes N_-} \otimes M_{M^2}. \quad (5.7)$$

Remark 5.9. We record the degrees of the degenerations, which will be needed for the explicit construction. There are two degenerations, together with restrictions. The first is a tensor product of $6N$ degree-one degenerations, so its degree is at most $6N$. The second applies Lemma 5.5 in each of the six orientations. Each application has degree at most $(M - 1)^2$, so their tensor product has degree at most $6(M - 1)^2$.

We continue with the proof and show how to pick N and N_+, N_- . For a positive integer k let j be the index of a largest term in the binomial expansion of $(\alpha + \beta)^k$. As there are $k + 1$ terms we get that

$$\binom{k}{j} \alpha^j \beta^{k-j} \geq \frac{(\alpha + \beta)^k}{k + 1}. \quad (5.8)$$

Denote

$$N = mk, \quad N_+ = mj, \quad N_- = m(k - j).$$

We shall also use the simple combinatorial estimate

$$M = \binom{N}{N_+} = \binom{mk}{mj} \geq \binom{k}{j}^m. \quad (5.9)$$

The factors $Q(a, b + s)^{\otimes N_+}$ on the right-hand side of (5.7) form j groups of m copies of $Q(a, b + s)$, and the factors $Q(a, b - s)^{\otimes N_-}$ form $k - j$ groups of m copies of $Q(a, b - s)$. We apply the degeneration $Q(a, b + s)^{\otimes m} \supseteq M_{n_+}$ to each group of the first kind, and $Q(a, b - s)^{\otimes m} \supseteq M_{n_-}$ to each group of the second kind. By (D1),

$$Q(a, b + s)^{\otimes N_+} \otimes Q(a, b - s)^{\otimes N_-} \supseteq M_{n_+}^{\otimes j} \otimes M_{n_-}^{\otimes (k-j)} \cong M_{n_+^j n_-^{k-j}}.$$

Composing with (5.7) and using (D2) gives

$$I_{(2M)^6} \otimes Q(a, b)^{\otimes N} \otimes M_{4N} \supseteq I_{M^6} \otimes M_{D_N}, \quad (5.10)$$

where

$$D_N = M^2 n_+^j n_-^{k-j} = (M \alpha^{mj} \beta^{m(k-j)})^2.$$

By (5.8) and (5.9) we have $D_N^{1/(2N)} \geq (\alpha + \beta)/(k + 1)^{1/k}$, and on the other hand, the binomial theorem applied to $(\alpha + \beta)^N$ gives $M \alpha^{mj} \beta^{m(k-j)} \leq (\alpha + \beta)^N$. Hence,

$$D_N^{1/(2N)} = M^{1/N} \alpha^{j/k} \beta^{(k-j)/k} \geq \left(\binom{k}{j} \alpha^j \beta^{k-j} \right)^{1/k} \geq \frac{\alpha + \beta}{(k + 1)^{1/k}}.$$

It remains to pay for the auxiliary tensors. The point is that (5.10) returns them, so we can use Lemma 5.3. Denote

$$E_N = I_{M^6} \otimes M_{4N}, \quad r_N = \left\lfloor \frac{D_N}{4^N} \right\rfloor.$$

As $\alpha + \beta > 2$, the ratio $D_N/4^N$ grows exponentially, so $r_N \geq 1$ for large N , and $\liminf_{k \rightarrow \infty} r_N^{1/(2N)} \geq (\alpha + \beta)/2$. On the left-hand side of (5.10) we have $I_{(2M)^6} \cong I_{64} \otimes I_{M^6}$, and on the right-hand side we can crop the matrix and get $M_{D_N} \geq M_{r_N} \otimes M_{4N}$. Thus

$$I_{64} \otimes Q(a, b)^{\otimes N} \otimes E_N \supseteq M_{r_N} \otimes E_N, \quad (5.11)$$

i.e., both the independent copies I_{M^6} and the matrix product M_{4N} that we invested are returned. As $Q(a, b) \geq I_2$ by Lemma 3.8, Lemma 5.3 with $u = 64$ and $p = N$ gives, for every fixed N ,

$$F(a, b) \geq r_N^{1/(2(N+6))} = \left\lfloor \frac{D_N}{4^N} \right\rfloor^{1/(2(N+6))} \geq \left(\frac{1}{2} \frac{D_N}{4^N} \right)^{1/(2(N+6))} \geq \frac{1}{2^{1/(2(N+6))}} \left(\frac{\alpha + \beta}{2(k + 1)^{1/k}} \right)^{\frac{N}{N+6}}.$$

Letting k go to infinity we get that $F(a, b) \geq (\alpha + \beta)/2$. $\square$

Exercise 5.10 (Merging several polynomials). *Let $a \geq 1$ and $b, r \geq 2$ be integers. Show that there is a degeneration of degree at most one,*

$$C(a, b) \otimes B_X(r) \supseteq C(a, b + r - 1) + \underbrace{C(a, b - 1) + \cdots + C(a, b - 1)}_{r-1 \text{ copies}},$$

where all the blocks share their X -variables and use disjoint Y - and Z -variables. What inequality for F does this construction suggest? Show that

$$rF(a, b) \geq F(a, b + r - 1) + (r - 1)F(a, b - 1)$$

already follows from concavity.

Hint. Replace the merging map in [Lemma 5.6](#) by

$$\mu : (\mathbb{F}[\xi]_b)^{\oplus r} \rightarrow \mathbb{F}[\xi]_{b+r-1}, \quad \mu(g_0, \dots, g_{r-1}) = \sum_{j=0}^{r-1} \xi^j g_j.$$

Identify its kernel and check that multiplication by f commutes with the resulting maps.

5.4 Shifted tripling

As described in [Section 5.1](#), we first prove [Lemma 5.2](#) and then obtain [Theorem 4.2](#) from it.

Lemma 5.2 (Three-sector degeneration; [[Ope26a](#), proof of Lemma 4.2]). *For integers $a, c \geq 1$, there is an explicit weighted degeneration of degree at most one,*

$$C(a, 3c + a - 1) \geq T_0 + T_1 + T_2,$$

in which the three blocks use the same X -variables and disjoint Y - and Z -variables. The outer blocks T_0, T_2 are copies of $C(a, c)$. The middle block T_1 is a copy of $C(a, c)$ with its second and third legs exchanged and its first-input indices reversed.

The idea is to find several copies of a shorter polynomial multiplication tensor, corresponding to polynomials of smaller degree, inside a longer one. A first attempt is to split the second input into three polynomials g_0, g_1, g_2 of length c , placed far apart, leaving gaps so that the output intervals of fg_0, fg_1, fg_2 do not overlap. Since each product fg_i has $a - 1$ more coefficients than g_i , the gaps must have length $a - 1$, so this requires a second input of length $3c + 2(a - 1)$. The construction below needs only $3c + a - 1$, because it also uses the indices in between: in the middle piece, the roles of the second input and the output are exchanged.

To see the pattern, consider the case $a = 2, c = 1$, where we start with

$$C(2, 4) = \sum_{j=0}^3 (x_0 y_j z_j + x_1 y_j z_{j+1}).$$

We split the Y -indices into a left, a middle and a right interval, $\{0\}, \{1, 2\}, \{3\}$, and the Z -indices into $\{0, 1\}, \{2\}, \{3, 4\}$. The outer intervals have sizes 1 and 2, the lengths of the second input and of the output of $C(2, 1)$, and the middle intervals have the same sizes in the opposite order. We multiply the middle Y -variables y_1, y_2 by t and the middle Z -variable z_2 by t^{-1} . The result is

$$(x_0 y_0 z_0 + x_1 y_0 z_1) + (x_1 y_1 z_2 + x_0 y_2 z_2) + (x_0 y_3 z_3 + x_1 y_3 z_4) + t(x_0 y_1 z_1 + x_1 y_2 z_3).$$

The terms multiplied by t are exactly those that join a middle Y -variable with an outer Z -variable. No term joins a middle Z -variable with an outer Y -variable, so no negative power of t appears.

The constant term consists of three pieces. They use separate y - and z -variables, share x_0, x_1 , and each of them is a dot product of length two. The first piece is $y_0(x_0 z_0 + x_1 z_1)$. As a bilinear map, its outputs, which are the coefficients of z_0 and z_1 , are $x_0 y_0$ and $x_1 y_0$. That is, it multiplies $f = x_0 + x_1 \xi$ by the constant polynomial y_0 , so it is a copy of $C(2, 1)$, which is the dot product

$B_Y(2)$. The same holds for the third piece $y_3(x_0z_3 + x_1z_4)$. In the middle piece $z_2(x_1y_1 + x_0y_2)$ the scalar sits on the Z -leg. After reversing the order of x_0, x_1 it is $B_Z(2)$, which is $\mathbb{C}(2, 1)$ with its second and third legs exchanged.

The general construction follows the same pattern. We split the Y - and Z -indices into a left, a middle and a right interval, give weight $+1$ to the middle Y -variables and weight -1 to the middle Z -variables, and keep the terms of weight zero. The outer pieces are copies of $\mathbb{C}(a, c)$, and in the middle piece the middle Y -interval, of length $c + a - 1$, plays the role of the output.

Proof of Lemma 5.2. We view $\mathbb{C}(a, 3c + a - 1)$ as the bilinear map

$$(f, g) \mapsto fg, \quad \deg f < a, \quad \deg g < 3c + a - 1.$$

Decompose the second input as

$$g = g_0 + \xi^c g_1 + \xi^{2c+a-1} g_2,$$

where

$$\deg g_0, \deg g_2 < c, \quad \deg g_1 < c + a - 1.$$

Decompose the output into blocks of the corresponding lengths:

$$fg = h_0 + \xi^{c+a-1} h_1 + \xi^{2c+a-1} h_2,$$

where

$$\deg h_0, \deg h_2 < c + a - 1, \quad \deg h_1 < c.$$

The products fg_0 and $\xi^{2c+a-1} fg_2$ lie entirely in the first and last output blocks, respectively. To see how the middle input contributes to all three output blocks, write

$$fg_1 = u_0 + \xi^{a-1} u_1 + \xi^{c+a-1} u_2,$$

where

$$\deg u_0, \deg u_2 < a - 1, \quad \deg u_1 < c.$$

Consequently, the multiplication map in these block coordinates is

$$(f; g_0, g_1, g_2) \mapsto (fg_0 + \xi^c u_0, u_1, fg_2 + u_2) = (h_0, h_1, h_2).$$

Now multiply the middle input block g_1 by t and the middle output block h_1 by t^{-1} . Since each u_i is bilinear in f, g_1 , the resulting map is

$$(f; g_0, g_1, g_2) \mapsto (fg_0 + t\xi^c u_0, u_1, fg_2 + tu_2).$$

The part not involving positive powers of t is the map

$$(f; g_0, g_1, g_2) \mapsto (fg_0, u_1, fg_2). \tag{5.12}$$

Thus the first and last blocks are copies of $\mathbb{C}(a, c)$.

By its definition, u_1 consists of the coefficients of fg_1 in degrees $a - 1, \dots, a + c - 2$. Writing $f = \sum_{i=0}^{a-1} f_i \xi^i$ and $g_1 = \sum_{j=0}^{c+a-2} (g_1)_j \xi^j$, we have

$$(u_1)_k = \sum_{i=0}^{a-1} f_i (g_1)_{a-1+k-i}, \quad 0 \leq k < c.$$

We now observe that this is again a polynomial multiplication tensor, in which the coefficients of f are reversed. Let $\tilde{f}$ be the polynomial f with coefficients reversed. That is,

$$\tilde{f}(\xi) = \xi^{a-1} f(\xi^{-1}),$$

so that $\tilde{f}_i = f_{a-1-i}$. Let $z_0, \dots, z_{c-1}$ be independent variables labeling the coefficients of u_1 . The corresponding trilinear form is

$$\sum_{k=0}^{c-1} \sum_{i=0}^{a-1} f_i(g_1)_{a-1+k-i} z_k = \sum_{i=0}^{a-1} \sum_{k=0}^{c-1} \tilde{f}_i(g_1)_{i+k} z_k.$$

The last expression is $C(a, c)$ with its second and third legs exchanged and its first-input coefficients reversed.

The three blocks in (5.12) share the coefficients of f and use disjoint second-input and output coordinates. This proves the claimed degree-one degeneration. $\square$

We now prove [Theorem 4.2](#). We first recall the theorem.

Theorem 4.2 (Shifted tripling [[Ope26a](#), Lemma 4.2]). *For all integers $a, c \geq 1$,*

$$F(a, 3c + a - 1) \geq 3F(a, c).$$

The proof idea is similar to the proof of [Theorem 4.1](#). We take tensor powers of the degeneration of [Lemma 5.2](#) and use [Lemma 5.5](#) to extract the relevant factors.

Proof. Fix a witness

$$Q(a, c)^{\otimes m} \supseteq M_n, \quad \text{of rate} \quad \alpha = n^{1/(2m)}.$$

We shall prove that $F(a, 3c + a - 1) \geq 3\alpha$, and the theorem follows by taking the supremum over all witnesses.

By [Lemma 5.2](#),

$$C(a, 3c + a - 1) \supseteq T_0 + T_1 + T_2,$$

where the three blocks share their X -variables and use disjoint Y - and Z -variables, the outer blocks are copies of $C(a, c)$, and the middle block is a copy of $C(a, c)$ with its last two legs exchanged and its first-input indices reversed.

Let $N \geq 1$ and let j_0, j_1, j_2 be nonnegative integers with $j_0 + j_1 + j_2 = N$. In the N th power of $T_0 + T_1 + T_2$, keep the products in which T_i appears exactly j_i times. There are

$$M = \frac{N!}{j_0! j_1! j_2!}$$

such products. We apply [Lemma 5.5](#) in each of the six orientations, using the same counts j_0, j_1, j_2 in every orientation, and take the tensor product.

The outer blocks give $Q(a, c)^{\otimes(j_0+j_2)}$. After separation, the factors no longer share variables, so we may reverse the first-input indices in each middle factor separately. Exchanging its last two legs permutes the six orientations. Thus, the middle blocks give $Q(a, c)^{\otimes j_1}$, and all the polynomial multiplication factors together give $Q(a, c)^{\otimes N}$. The six dot products give M_{M^2} , and the returned unit tensors give I_{M^6} . Hence

$$I_{(2M)^6} \otimes Q(a, 3c + a - 1)^{\otimes N} \supseteq I_{M^6} \otimes Q(a, c)^{\otimes N} \otimes M_{M^2}. \quad (5.13)$$

Remark 5.11. *This construction also uses two degenerations, together with restrictions. The first tensors $6N$ copies of the degree-one three-sector degeneration, so its degree is at most $6N$. The second applies separation in the six orientations and has degree at most $6(M-1)^2$.*

We now take N to be a multiple of $3m$ and choose $j_0 = j_1 = j_2 = N/3$, so that

$$M = \frac{N!}{((N/3)!)^3}.$$

Applying our witness to the N/m groups of m copies of $\mathbf{Q}(a, c)$ gives

$$I_{(2M)^6} \otimes \mathbf{Q}(a, 3c + a - 1)^{\otimes N} \supseteq I_{M^6} \otimes \mathbf{M}_{r_N}, \quad \text{where} \quad r_N = M^2 n^{N/m}. \quad (5.14)$$

This time the auxiliary tensor is just $E_N = I_{M^6}$, and as $I_{(2M)^6} \cong I_{64} \otimes I_{M^6}$ we get

$$I_{64} \otimes \mathbf{Q}(a, 3c + a - 1)^{\otimes N} \otimes E_N \supseteq \mathbf{M}_{r_N} \otimes E_N. \quad (5.15)$$

As $3c + a - 1 \geq 3$, [Lemma 3.8](#) gives $\mathbf{Q}(a, 3c + a - 1) \geq I_2$, and so [Lemma 5.3](#) implies that

$$F(a, 3c + a - 1) \geq r_N^{1/(2(N+6))}.$$

It remains to estimate M . Since the balanced multinomial coefficient is the largest one,

$$\frac{3^N}{(N+1)^2} \leq M \leq 3^N, \quad \text{and so} \quad M^{1/N} \rightarrow 3.$$

Consequently, as $n = \alpha^{2m}$,

$$r_N^{1/(2(N+6))} = (M^{1/N} \alpha)^{N/(N+6)} \rightarrow 3\alpha,$$

which proves the claim. $\square$

6 Constructing an explicit sequence of algorithms

In this section we prove [Theorem 1.2](#) by explaining how to choose the witnesses.

The proof of [Theorem 1.1](#) shows that high powers of polynomial multiplication tensors degenerate to large matrix multiplication tensors via a derivation of a lower bound on $F(a, a)$. However, $F(a, a)$ is defined as a supremum of rates of witnesses, and the proof only shows that good witnesses exist, and does not have any guarantee on the size of witnesses. Our goal is to construct such a witness explicitly and bound its size. By following the steps of the derivation carefully, we show that, for every $\varepsilon > 0$, we can explicitly construct a restriction $\mathbf{Q}(A, A)^{\otimes m} \geq \mathbf{M}_n$ that gives an algorithm with exponent below $9/4 + \varepsilon$.

One way to realize a derivation of a lower bound on $F(a, a)$ is to construct a witness after every step. The proofs of [Theorems 4.1](#) and [4.2](#) show how to turn witnesses for the bounds used in a step into a witness for the new bound. The problem is that each application takes high tensor powers of the witnesses it starts from, so these powers multiply along the derivation.

We improve this in two ways. First, we make larger jumps along the diagonal. Each jump uses finitely many midpoint and tripling steps and incurs a small loss, which we bound in [Lemma 6.5](#). Second, we compose the finite constructions while *keeping their auxiliary tensors*, and pay for these tensors only once. This is what [Lemma 6.2](#) captures.

We next give some terminology that will be used in the proof.

The proof of [Lemma 4.3](#) shows that $F(a, a)$ is large by using values of F at nearby points: We start with the baseline values $\max(a, b)$ at every pair (a, b) ([Lemma 3.8](#)), and allow four ways of obtaining further values. Each comes from the proof of a rate inequality or from a trivial observation:

1. Midpoint step: If α and β are available at $(a, b - s)$ and $(a, b + s)$, respectively, we may put $(\alpha + \beta)/2$ at (a, b) , where $1 \leq s < b$ (this comes from the proof of [Theorem 4.1](#)).
2. Tripling step: From α at (a, c) we may put 3α at $(a, 3c + a - 1)$ (this comes from the proof of [Theorem 4.2](#)).
3. Symmetry step: We may transfer a value from (a, b) to (b, a) .
4. Restriction step: We may transfer a value from (a, b) to (a', b') when $a' \geq a$ and $b' \geq b$.

Definition 6.1. A derivation is a finite list of entries, each consisting of a pair (a, b) and a value. Each entry is either a baseline value $\max(a, b)$, or is obtained from earlier entries by one of the four steps. We record the steps together with the earlier entries they use. Note that a pair (a, b) may appear in several entries, with values recorded at different points in the derivation. Each step refers to particular earlier entries, using the values recorded in those entries. The number of steps of a derivation is the number of its entries that are not baseline values.

By [Lemma 3.8](#), the baseline values at (a, b) are at most $F(a, b)$. Also $F(a, b) = F(b, a)$, and $Q(a', b') \geq Q(a, b)$ when $a' \geq a$ and $b' \geq b$ (set the extra variables to zero). Together with [Theorems 4.1](#) and [4.2](#), it follows that every value obtained at (a, b) is at most $F(a, b)$.

The following lemma captures the description given at the beginning of this section on how to transform a derivation into a restriction. Since the proof of the lemma does not introduce new tools and is mainly technical, we postpone its proof to [Section 6.1](#).

Lemma 6.2. Suppose a derivation of S steps gives a value $\phi \geq 2$ at (a, b) . Suppose all lengths occurring in the derivation are at most L . Then we can explicitly construct a restriction

$$Q(a, b)^{\otimes m} \geq M_n, \quad n^{1/(2m)} \geq \phi/2,$$

where m and $\log n$ are bounded by a polynomial in $S + \log L$.

We next show how to start with a value at (a, b) and obtain a value at a farther point in the same row (that is, with the same a).

Lemma 6.3. Let $a, b, k \geq 1$ and $1 \leq s \leq a + 2b$ be integers. Suppose a derivation contains a value $\alpha > 0$ at (a, b) . Then one tripling step, at most k midpoint steps and at most two restriction steps extend it to a derivation that contains a value at least

$$(1 - 2^{1-k}) \left(1 + \frac{2s}{a + 2b}\right) \alpha$$

at $(a, b + s)$. All new entries lie in row a , in columns between b and $3b + a$.

Ideally, we would have liked to obtain the value $\left(1 + \frac{2s}{a + 2b}\right) \alpha$, but we suffer a small loss due to the finite nature of our construction. We shall keep track of these small losses and show that they do not affect the final exponent.

Proof. A tripling step gives 3α at $(a, 3b + a - 1)$, and restriction gives the same value at $(a, 3b + a)$. Let

$$\psi(c) = \left(1 + \frac{2(c - b)}{a + 2b}\right) \alpha$$

be the linear function with $\psi(b) = \alpha$ and $\psi(3b + a) = 3\alpha$. The proof will follow from the following claim.

Claim 6.4. *For every integer $k \geq 0$ and every integer c with $b \leq c \leq 3b + a$, at most k midpoint steps and at most one further restriction step give a value at least $\psi(c) - 2^{1-k}\alpha$ at (a, c) .*

Proof. If $c = b$ or $c = 3b + a$ then we already have the value $\psi(c)$.

So assume $b < c < 3b + a$. We prove the claim by induction on k . For $k = 0$ restriction from (a, b) gives α at (a, c) . As $\psi(c) \leq 3\alpha$, this is at least $\psi(c) - 2\alpha$ as claimed.

Let $k \geq 1$ and $b < c < 3b + a$. Set

$$e = \begin{cases} b & 2c \leq 4b + a \\ 3b + a & \text{otherwise} \end{cases}, \quad \text{and} \quad c' = 2c - e.$$

It is easy to verify that $b \leq c' \leq 3b + a$, and that c is the midpoint of e and c' , at distance $|c - e|$ from both. Clearly $|c - e| \geq 1$, and it holds that $|c - e| \leq c - b < c$, which is again easy to verify in each of the cases.

By the induction hypothesis, at most $k - 1$ midpoint steps and one restriction step give a value at least $\psi(c') - 2^{2-k}\alpha$ at (a, c') . A midpoint step with the value $\psi(e)$ at (a, e) then gives at (a, c) a value that is at least

$$\frac{\psi(e) + (\psi(c') - 2^{2-k}\alpha)}{2} = \psi(c) - 2^{1-k}\alpha,$$

since ψ is linear. □

Now take $c = b + s$. Then

$$\psi(b + s) = \left(1 + \frac{2s}{a + 2b}\right) \alpha.$$

Since $\psi(b + s) \geq \alpha$, an additive loss of $2^{1-k}\alpha$ is at most a fraction 2^{1-k} of $\psi(b + s)$, implying the lemma. □

Next, we do the same for doubling both coordinates. That is, we move from (qd, qd) to $(2qd, 2qd)$ in q diagonal moves of length d . Each move consists of two applications of [Lemma 6.3](#), one before and one after symmetry.

Lemma 6.5. *Let $q \geq 16$ and $d \geq 1$ be integers. Suppose a derivation contains a value $\alpha > 0$ at (qd, qd) . Then at most $2q^2$ midpoint steps and at most $7q$ other steps extend it to a derivation that contains a value strictly greater than*

$$2^{4/3} \left(1 - \frac{1}{2q}\right) \alpha$$

at $(2qd, 2qd)$. All lengths used are less than $8qd$.

Proof. Fix $q \leq i < 2q$, and let $a_i = id$, and suppose we derived a value β_i at (a_i, a_i) . Apply [Lemma 6.3](#) with $b = a_i$, $s = d$ and $k = q$. At $(a_i, a_i + d)$ we obtain a value at least

$$(1 - 2^{1-q}) \left(1 + \frac{2d}{3a_i}\right) \beta_i = (1 - 2^{1-q}) \left(1 + \frac{2}{3i}\right) \beta_i.$$

A symmetry step gives the same value at $(a_i + d, a_i)$ as well. Next, we apply the lemma again, in row $a_i + d$, with $b = a_i$ and the same s and k . This gives a value at $(a_i + d, a_i + d)$ and gains a further factor of at least

$$(1 - 2^{1-q}) \left(1 + \frac{2d}{(a_i + d) + 2a_i}\right) = (1 - 2^{1-q}) \left(1 + \frac{2}{3i + 1}\right),$$

over the value at $(a_i + d, a_i)$. Thus the move from (id, id) to $((i+1)d, (i+1)d)$ gains at least the factor

$$(1 - 2^{1-q})^2 \left(1 + \frac{2}{3i}\right) \left(1 + \frac{2}{3i+1}\right) = (1 - 2^{1-q})^2 \frac{i+1}{i} \left(1 + \frac{1}{3i+1}\right).$$

Multiplying over $q \leq i < 2q$, the gain in value from moving from (qd, qd) to $(2qd, 2qd)$ is at least

$$\prod_{i=q}^{2q-1} (1 - 2^{1-q})^2 \frac{i+1}{i} \left(1 + \frac{1}{3i+1}\right) = 2(1 - 2^{1-q})^{2q} \prod_{i=q}^{2q-1} \left(1 + \frac{1}{3i+1}\right).$$

To bound the product, observe that

$$\left(1 + \frac{1}{3i+1}\right)^3 \geq 1 + \frac{3}{3i+1} = \frac{3(i+1)+1}{3i+1}.$$

Consequently,

$$\prod_{i=q}^{2q-1} \left(1 + \frac{1}{3i+1}\right) \geq \left(\frac{6q+1}{3q+1}\right)^{1/3} = 2^{1/3} \left(1 - \frac{1}{6q+2}\right)^{1/3} \geq 2^{1/3} \left(1 - \frac{1}{4q}\right).$$

Also, since $q \geq 16$ we have $8q^2 \leq 2^{q-1}$, and therefore

$$(1 - 2^{1-q})^{2q} \geq 1 - 2q 2^{1-q} \geq 1 - \frac{1}{4q}.$$

The total gain is therefore at least

$$2^{4/3} \left(1 - \frac{1}{4q}\right)^2 > 2^{4/3} \left(1 - \frac{1}{2q}\right).$$

By [Lemma 6.3](#), each diagonal move uses two sets of at most q midpoint steps, two tripling steps and at most four restriction steps, and in addition we performed one symmetry step, so each diagonal move uses at most $2q + 7$ steps. Its largest length is $4id + d < 8qd$. $\square$

We are now ready to prove [Theorem 1.2](#).

Proof of Theorem 1.2. Let

$$q = \left\lceil \frac{16}{\varepsilon^2} \right\rceil, \quad A = q2^q.$$

Start with the baseline value q at (q, q) . Apply the doubling lemma, [Lemma 6.5](#), q times, with $d = 1, 2, 4, \dots, 2^{q-1}$. The final value B at (A, A) satisfies

$$B \geq \left(2^{4/3} \left(1 - \frac{1}{2q}\right)\right)^q \cdot q \geq \frac{q}{2} 2^{4q/3}.$$

There are $S \leq q(2q^2 + 7q) = O(q^3)$ steps, and all lengths are less than $4A$.

Apply [Lemma 6.2](#) to this entire derivation. We obtain an exact restriction

$$\mathbf{Q}(A, A)^{\otimes m} \geq \mathbf{M}_n, \quad \gamma = n^{1/(2m)} \geq \frac{B}{2} \geq \frac{q}{4} 2^{4q/3},$$

with $m, \log n = q^{O(1)}$, since $\log(4A) = O(q)$. The rank bound $\mathbf{R}(\mathbf{Q}(A, A)) \leq (2A - 1)^6$ and [Proposition 2.3](#) give an algorithm with exponent at most

$$\frac{6 \log(2A - 1)}{\frac{1}{m} \log n} = \frac{3 \log(2A - 1)}{\log \gamma} < \frac{3(q + \log q + 1)}{4q/3 + \log q - 2}.$$

Note that

$$\frac{3(q + \log q + 1)}{4q/3 + \log q - 2} - \frac{9}{4} = \frac{3 \log q + 30}{4(4q/3 + \log q - 2)} \leq \frac{3 \log q + 30}{16q/3} \leq \frac{4}{\sqrt{q}} \leq \varepsilon,$$

where we have used $q \geq 16$. This proves the exponent bound. The base matrix size satisfies $n \leq 2^{q^{O(1)}} = 2^{\text{poly}(1/\varepsilon)}$. $\square$

6.1 Proof of Lemma 6.2

Fix a completed derivation of S steps, giving a value $\phi \geq 2$ at a point (a, b) . We must construct a restriction $Q(a, b)^{\otimes m} \geq M_n$ with $n^{1/(2m)} \geq \phi/2$, and prove the appropriate upper bounds on m and $\log n$.

We first recall the finite construction for a single step. We then prove by induction that these constructions can be composed while keeping an auxiliary tensor, whose rank is not too large. Finally, we apply interpolation as in the proof of Bini's theorem (Theorem 2.23), and then prepare and reuse the auxiliary tensor. To prove that we can perform the interpolation efficiently, we also keep track of the degree of degenerations that occur in the construction.

Analyzing a single step. The following are the degenerations used in the proofs of Theorems 4.1 and 4.2.

Claim 6.6. *For positive integers a, N , the following hold.*

1. *Midpoint step: For integers $1 \leq s < b$ and $0 \leq j \leq N$, let $M = \binom{N}{j}$. Then*

$$I_{64M^6} \otimes Q(a, b)^{\otimes N} \otimes M_{4N} \supseteq I_{M^6} \otimes Q(a, b-s)^{\otimes j} \otimes Q(a, b+s)^{\otimes (N-j)} \otimes M_{M^2}.$$

2. *Tripling step: For $b \geq 1$ and nonnegative integers $j_0 + j_1 + j_2 = N$, let $M = N!/(j_0!j_1!j_2!)$. Then*

$$I_{64M^6} \otimes Q(a, 3b+a-1)^{\otimes N} \supseteq I_{M^6} \otimes Q(a, b)^{\otimes N} \otimes M_{M^2}.$$

Each construction is a composition of two degenerations, of degrees at most $6N$ and $6(M-1)^2$, together with restrictions. Since $M \leq 3^N$, each of these degrees is at most $6 \cdot 9^N$.

Proof. These are the finite constructions in the proofs of Theorems 4.1 and 4.2. For the midpoint construction, take $N_- = j$ and $N_+ = N - j$. The degree bounds follow from Remarks 5.9 and 5.11. $\square$

We distinguish between the steps of the numerical derivation and the stages of the tensor construction. A step is one application of a midpoint, tripling, symmetry or restriction rule. A stage is one of the individual degenerations used in the tensor construction. As Claim 6.6 shows, each midpoint or tripling step is implemented by two stages, together with restrictions.

Composing the steps of the derivation. Once the values are calculated, we use the recorded steps to construct a degeneration. We can describe the dependencies between the bounds by a directed graph: The source nodes are the baseline bounds. Each step adds a node recording the position and the new bound, with incoming edges from the nodes whose bounds were used in that step. Thus, several nodes may correspond to the same position, recording bounds obtained at different steps.

When we calculated the values we followed the direction of the edges: we used bounds at the incoming neighbors to obtain a bound at the new node. The tensor construction goes in the reverse

direction. Starting at the node for the final bound, we use [Claim 6.6](#) to replace copies of its tensor by copies of the tensors at its incoming neighbors, together with matrix multiplication and auxiliary factors. We then continue with the earlier nodes.

To understand what the inductive statement should say, consider a midpoint step. Suppose that bounds α and β at $(a, b - s)$ and $(a, b + s)$ give the bound $(\alpha + \beta)/2$ at (a, b) . The corresponding finite construction, applied to k copies of $\mathbf{Q}(a, b)$ with suitable auxiliary tensors, produces the factors

$$\mathbf{Q}(a, b - s)^{\otimes j} \otimes \mathbf{Q}(a, b + s)^{\otimes (k-j)}$$

for a suitable $0 \leq j \leq k$, together with matrix multiplication and auxiliary factors. To continue, we must apply the earlier constructions to these j copies of $\mathbf{Q}(a, b - s)$ and $k - j$ copies of $\mathbf{Q}(a, b + s)$. Thus, even if we are ultimately interested in copies of a single tensor, the induction must allow products of different tensors.

We number the nodes of the graph, listing the baseline nodes first and then the nodes produced by the successive steps. Write (c_i, d_i) for the position of node i and α_i for its bound. Our inductive statement concerns a product

$$\bigotimes_i \mathbf{Q}(c_i, d_i)^{\otimes n_i}, \quad N = \sum_i n_i.$$

Here n_i is the number of copies of $\mathbf{Q}(c_i, d_i)$ to which we will apply the construction for the bound α_i , and N is the total number of these tensor copies. In the midpoint step, k copies at one node are replaced by j and $k - j$ copies at its two incoming neighbors, so this total is preserved.

Lemma 6.7. *Assume the notation above. Let ℓ be the number of nodes in the graph, including the baseline nodes. For nonnegative integers $n_1, \dots, n_\ell$ with $N = \sum_i n_i \geq 1$, we can explicitly construct a nonzero tensor E and an integer $r \geq 1$ such that*

$$I_{64^S} \otimes \left(\bigotimes_{i=1}^{\ell} \mathbf{Q}(c_i, d_i)^{\otimes n_i} \right) \otimes E \geq \mathbf{M}_r \otimes E$$

and

$$\sqrt{r} \geq \frac{\prod_{i=1}^{\ell} \alpha_i^{n_i}}{(N+1)^{2S}}.$$

If all c_i, d_i are at most L , the construction supplies a rank decomposition of E with at most 2^{16SN} terms, and

$$r \leq 3^{2SN} L^{2N}.$$

The construction is given as a composition of at most $2S$ degenerations, each of degree at most $6 \cdot 9^N$, together with restrictions.

Proof. The proof is by induction on the number of steps S in the derivation. At $S = 0$, all entries are baseline entries, so

$$\bigotimes_i \mathbf{Q}(c_i, d_i)^{\otimes n_i} \geq \bigotimes_i \mathbf{M}_{\alpha_i^{2n_i}} \cong \mathbf{M}_{\prod_i \alpha_i^{2n_i}}.$$

We take $E = I_1$ and $r = \prod_i \alpha_i^{2n_i}$. Since each baseline value is at most L , this also gives $r \leq L^{2N}$.

Suppose $S \geq 1$. If $n_\ell = 0$, then the claim follows by applying the induction hypothesis to the first $S - 1$ steps and discarding the extra factor I_{64} .

So assume $n_\ell > 0$. Set

$$U_0 = \bigotimes_{i < \ell} \mathbf{Q}(c_i, d_i)^{\otimes n_i}.$$

Thus U_0 is the product of the input factors other than the n_ℓ factors belonging to the last entry.

We distinguish cases according to the step that produced the last node ℓ :

1. *Midpoint step:* Suppose the last entry is at (a, b) , and is obtained from values α, β at $(a, b - s), (a, b + s)$. Thus $\alpha_\ell = (\alpha + \beta)/2$. Choose j maximizing $\binom{n_\ell}{j} \alpha^j \beta^{n_\ell - j}$ and set $M = \binom{n_\ell}{j}$. By choice of j it holds that

$$\frac{M}{2^{n_\ell}} \alpha^j \beta^{n_\ell - j} \geq \frac{1}{n_\ell + 1} \left(\frac{\alpha + \beta}{2} \right)^{n_\ell} = \frac{1}{n_\ell + 1} \alpha_\ell^{n_\ell}. \quad (6.1)$$

Define

$$U' = U_0 \otimes \mathbf{Q}(a, b - s)^{\otimes j} \otimes \mathbf{Q}(a, b + s)^{\otimes (n_\ell - j)}.$$

The tensor U' still has N factors of the form $\mathbf{Q}(c, d)$, and each is associated with an earlier node. We may therefore apply the induction hypothesis to the first $S - 1$ steps, with the corresponding multiplicities, which gives

$$I_{64^{S-1}} \otimes U' \otimes E' \supseteq \mathbf{M}_{r'} \otimes E',$$

where

$$\sqrt{r'} \geq \frac{\alpha^j \beta^{n_\ell - j} \prod_{i=1}^{\ell-1} \alpha_i^{n_i}}{(N + 1)^{2(S-1)}}.$$

By [Claim 6.6](#), we invested $I_{64M^6} = I_{64} \otimes I_{M^6}$ and $\mathbf{M}_{4^{n_\ell}}$ and obtained back I_{M^6} and $\mathbf{M}_{M^2}$. Since the factor I_{64} is absorbed into I_{64^S} , we define

$$E = I_{M^6} \otimes E' \otimes \mathbf{M}_{4^{n_\ell}}.$$

Consider the following degeneration:

$$\begin{aligned} I_{64^S} \otimes \mathbf{Q}(a, b)^{\otimes n_\ell} \otimes U_0 \otimes E \\ &\cong I_{64^{S-1}} \otimes (I_{64M^6} \otimes \mathbf{Q}(a, b)^{\otimes n_\ell} \otimes \mathbf{M}_{4^{n_\ell}}) \otimes U_0 \otimes E' \\ &\supseteq I_{64^{S-1}} \otimes U' \otimes E' \otimes I_{M^6} \otimes \mathbf{M}_{M^2} \\ &\supseteq I_{M^6} \otimes E' \otimes \mathbf{M}_{r'} \otimes \mathbf{M}_{M^2} \\ &\cong I_{M^6} \otimes E' \otimes \mathbf{M}_{M^2 r'}. \end{aligned}$$

To return the factor $\mathbf{M}_{4^{n_\ell}}$ as part of E , set

$$r = \max \left\{ 1, \left\lfloor \frac{M^2 r'}{4^{n_\ell}} \right\rfloor \right\}.$$

If $r = 1$, we do not use the degeneration above, since $M^2 r'$ may be smaller than 4^{n_ℓ} , and then the factor $\mathbf{M}_{4^{n_\ell}}$ is not returned. Instead, we restrict all polynomial multiplication factors in the statement of the lemma to I_1 and discard I_{64^S} , which proves the claim with $r = 1$.

If $r > 1$ then the above gives

$$I_{64^S} \otimes \mathbf{Q}(a, b)^{\otimes n_\ell} \otimes U_0 \otimes E \supseteq I_{M^6} \otimes E' \otimes \mathbf{M}_{M^2 r'} \supseteq \mathbf{M}_r \otimes E.$$

In either case $r \geq M^2 r' / (2 \cdot 4^{n_\ell})$. Thus (6.1) gives

$$\begin{aligned} \sqrt{r} &\geq \frac{M}{2^{n_\ell} \sqrt{2}} \sqrt{r'} \\ &\geq \frac{\prod_{i \leq \ell} \alpha_i^{n_i}}{\sqrt{2} (n_\ell + 1) (N + 1)^{2(S-1)}} \\ &\geq \frac{\prod_{i \leq \ell} \alpha_i^{n_i}}{(N + 1)^{2S}}, \end{aligned}$$

since $n_\ell \leq N$ and $\sqrt{2}(n_\ell + 1) \leq (N + 1)^2$. This concludes the case of a midpoint step.

2. *Tripling step*: Suppose the last entry is at $(a, 3b + a - 1)$ and comes from a value α at (a, b) , so $\alpha_\ell = 3\alpha$. Choose a largest trinomial coefficient

$$M = \frac{n_\ell!}{j_0!j_1!j_2!} \geq \frac{3^{n_\ell}}{(n_\ell + 1)^2}.$$

As before, applying the induction hypothesis to

$$U' = Q(a, b)^{\otimes n_\ell} \otimes U_0$$

gives

$$I_{64^{S-1}} \otimes U' \otimes E' \supseteq M_{r'} \otimes E', \quad \sqrt{r'} \geq \frac{\alpha^{n_\ell} \prod_{i < \ell} \alpha_i^{n_i}}{(N + 1)^{2(S-1)}}.$$

Set $E = I_{M^6} \otimes E'$ and $r = M^2 r'$. By [Claim 6.6](#) we have

$$\begin{aligned} I_{64^S} \otimes Q(a, 3b + a - 1)^{\otimes n_\ell} \otimes U_0 \otimes E \\ &\cong I_{64^{S-1}} \otimes (I_{64} \otimes I_{M^6} \otimes Q(a, 3b + a - 1)^{\otimes n_\ell}) \otimes E' \otimes U_0 \\ &\supseteq I_{64^{S-1}} \otimes (I_{M^6} \otimes Q(a, b)^{\otimes n_\ell} \otimes M_{M^2}) \otimes E' \otimes U_0 \\ &\cong I_{64^{S-1}} \otimes U' \otimes E' \otimes I_{M^6} \otimes M_{M^2} \\ &\supseteq M_{M^2 r'} \otimes E' \otimes I_{M^6} = M_{M^2 r'} \otimes E. \end{aligned}$$

Consequently,

$$\sqrt{r} = M \sqrt{r'} \geq \frac{(3\alpha)^{n_\ell} \prod_{i < \ell} \alpha_i^{n_i}}{(n_\ell + 1)^2 (N + 1)^{2(S-1)}} \geq \frac{\prod_{i \leq \ell} \alpha_i^{n_i}}{(N + 1)^{2S}}.$$

This completes the proof for the case of a tripling step.

3. *Symmetry and restriction steps*: These steps are trivial; we simply replace the n_ℓ factors of the last entry by n_ℓ factors of its predecessor and apply the induction hypothesis (and discard the extra I_{64}). Thus, in this case we take $E = E'$ and $r = r'$.

It remains to check the rank and degree bounds. In a midpoint step, the new part of E has a rank decomposition with at most

$$M^6 (4^{n_\ell})^3 \leq 3^{6n_\ell} 4^{3n_\ell} < 2^{16n_\ell}$$

terms. In a tripling step it has rank M^6 , and the other steps add no auxiliary tensor. Additionally, in every case $r \leq 3^{2n_\ell} r'$. Since $n_\ell \leq N$, the claimed bounds follow by induction.

As for the degree, each induction step adds at most two degenerations. By [Claim 6.6](#), each has degree at most $6 \cdot 9^{n_\ell} \leq 6 \cdot 9^N$. Thus the entire construction uses at most $2S$ degenerations, with the claimed degree bounds. $\square$

For [Lemma 6.2](#) itself, we take N copies of the tensor at the final node and set all other multiplicities to zero.

Proof of Lemma 6.2. If $S = 0$, the baseline restriction suffices, so assume $S \geq 1$.

We now apply [Lemma 6.7](#) to the completed derivation, with N factors of the final tensor and no other input factors. For every $N \geq 1$, the lemma gives

$$I_{64^S} \otimes Q(a, b)^{\otimes N} \otimes E \supseteq M_r \otimes E, \quad \sqrt{r} \geq \frac{\phi^N}{(N + 1)^{2S}}. \quad (6.2)$$

The supplied rank decomposition of E has at most 2^{16SN} terms.

Taking the N th tensor power of both sides of (6.2) gives

$$I_{64SN} \otimes Q(a, b)^{\otimes N^2} \otimes E^{\otimes N} \geq M_{rN} \otimes E^{\otimes N}.$$

We wish to turn the entire degeneration into a restriction. Taking the N th tensor power of the construction replaces each of these degenerations by its N th tensor power. We apply Lemma 2.24 to each one separately and then compose the resulting restrictions.

Consider a degeneration stage. If its original degree was d , the degree of this tensor power is at most Nd , so interpolation requires an additional factor I_{Nd+1} .

By Lemma 6.7, there are at most $2S$ degeneration stages, each of degree at most $6 \cdot 9^N$. After taking their N th tensor powers, each stage has degree at most $6N9^N$. Interpolating each stage separately therefore requires a total additional unit factor of dimension at most

$$(6N9^N + 1)^{2S} = 2^{O(SN)}.$$

In conclusion,

$$I_{2^{O(SN)}} \otimes Q(a, b)^{\otimes N^2} \otimes E^{\otimes N} \geq M_{rN} \otimes E^{\otimes N}.$$

The unit tensor $I_{2^{O(SN)}}$ can be supplied using $O(NS)$ additional copies of $Q(a, b)$ (since $\phi \geq 2$, we have $(a, b) \neq (1, 1)$, and the baseline restriction gives $Q(a, b) \geq I_2$). Thus,

$$Q(a, b)^{\otimes (N^2 + O(SN))} \otimes E^{\otimes N} \geq M_{rN} \otimes E^{\otimes N}. \quad (6.3)$$

Remark 6.8. *It is important to note that the order matters here. If we first interpolated each degeneration and then took its N th tensor power, then a degeneration of degree d would require the additional factor*

$$I_{d+1}^{\otimes N} = I_{(d+1)^N}.$$

Using our degree bounds, this would give only an $O(SN^2)$ bound on the number of additional copies of $Q(a, b)$. This estimate does not make the overhead small compared with the N^2 input copies by increasing N . By postponing the interpolation we require only $O(SN)$ copies of $Q(a, b)$.

We now prepare $E^{\otimes N}$ once and reuse it. By Lemma 6.7 its rank is at most 2^{16SN^2} , so

$$Q(a, b)^{\otimes 16SN^2} \geq I_{2^{16SN^2}} \geq E^{\otimes N}.$$

Apply (6.3) N times, returning the same $E^{\otimes N}$ after every application. Then discard it. Explicitly,

$$\begin{aligned} Q(a, b)^{\otimes (N^3 + O(SN^2))} &\geq Q(a, b)^{\otimes N(N^2 + O(SN))} \otimes E^{\otimes N} \\ &\geq M_{rN^2} \otimes E^{\otimes N} \\ &\geq M_{rN^2}. \end{aligned}$$

For a sufficiently large absolute integer c , accounting for both the repeated applications and the preparation of $E^{\otimes N}$, we take

$$m = N^3 + cSN^2, \quad n = r^{N^2}.$$

It remains to choose N . The bound in (6.2) gives

$$\log n \geq 2N^3 \log \phi - 4SN^2 \log(N + 1).$$

Therefore

$$\log \phi - \frac{\log n}{2m} \leq \frac{cS \log \phi + 2S \log(N+1)}{N}.$$

By [Lemma 3.8](#),

$$\phi \leq F(a, b) \leq \sqrt{ab(a+b-1)} \leq L^2.$$

Since $\log(N+1) \leq 2\sqrt{N}$, we obtain

$$\log \phi - \frac{\log n}{2m} \leq \frac{2cS \log L}{N} + \frac{4S}{\sqrt{N}}.$$

For $S \geq 1$, take

$$N = \lceil 8cS \log L + 64S^2 \rceil.$$

Since $N \geq 8cS \log L$ and $\sqrt{N} \geq 8S$, the two terms are at most $1/4$ and $1/2$, respectively. Therefore

$$\log \phi - \frac{\log n}{2m} \leq \frac{3}{4} < 1,$$

and hence $n^{1/(2m)} \geq \phi/2$. This choice of N is polynomial in $S + \log L$. Moreover, [Lemma 6.7](#) gives

$$\log n = N^2 \log r \leq 2N^3(S \log 3 + \log L).$$

Thus both m and $\log n$ are polynomial in $S + \log L$, as required. $\square$

7 (In)Dependence on the field

We now explain why the same argument proves $\omega_{\mathbb{F}} \leq 9/4$ over every field $\mathbb{F}$, where $\omega_{\mathbb{F}}$ is the exponent of matrix multiplication over $\mathbb{F}$. The adaptation of the construction to positive characteristic follows [\[Zui26\]](#).

Let $\mathbb{K}$ be the prime field of $\mathbb{F}$, i.e., $\mathbb{K} = \mathbb{Q}$ if the characteristic is zero and $\mathbb{K} = \mathbb{F}_p$ if it is p . We first carry out all the constructions over the algebraic closure $\overline{\mathbb{K}}$ of $\mathbb{K}$, and then explain how to get back to $\mathbb{K}$.

As $\overline{\mathbb{K}}$ is infinite, evaluation and interpolation work as before. The only place where we have to be a bit careful is the separation lemma, which uses roots of unity. Here we let the number of copies L be either $2M - 1$ or $2M$, where in characteristic $p > 0$ we pick the one that is not divisible by p (in characteristic zero both choices are fine). Then $\overline{\mathbb{K}}$ contains a primitive L th root of unity, L is invertible in $\overline{\mathbb{K}}$, and the proof of [Lemma 5.1](#) goes through since $|k - l + j - i| \leq 2M - 2 < L$.

To return to the prime field, we use the standard argument from the proof of Schönhage's theorem that the matrix multiplication exponent depends only on the characteristic of the field [\[Sch81, Theorem 2.8\]](#); see also [\[BCS97, Corollary 15.18\]](#). We include the argument for completeness.

Fix $\varepsilon > 0$. The construction, followed by interpolation, gives an exact decomposition of some M_u with $r < u^{9/4+\varepsilon}$ terms. As this decomposition has finitely many coefficients, they all lie in some finite extension $\mathbb{E}$ of $\mathbb{K}$, say of degree d . Let $e_1, \dots, e_d$ be a basis of $\mathbb{E}$ over $\mathbb{K}$, and let $\varphi : \mathbb{E} \rightarrow \mathbb{K}$ be a $\mathbb{K}$ -linear map with $\varphi(1) = 1$. We can write each linear form that appears in a rank-one term as $\sum_i e_i \ell_i$, where the ℓ_i have coefficients in $\mathbb{K}$. Expanding the product of the three linear forms and applying φ to the coefficients, we replace each rank-one term by at most d^3 rank-one terms over $\mathbb{K}$. Since the coefficients of M_u are in $\mathbb{K}$, and φ is the identity on $\mathbb{K}$, this does not change the tensor.

Applying the same conversion to the N th tensor power of our decomposition, we get that

$$R_{\mathbb{K}}(M_{u^N}) \leq d^3 r^N.$$

As every algorithm over $\mathbb{K}$ is also an algorithm over $\mathbb{F}$, [Proposition 2.3](#) gives

$$\omega_{\mathbb{F}} \leq \omega_{\mathbb{K}} \leq \frac{\log(d^3 r^N)}{N \log u} \longrightarrow \log_u r < 9/4 + \varepsilon,$$

and the claim follows by letting $\varepsilon \rightarrow 0$.

8 Rectangular matrix multiplication

The ideas outlined in the previous sections can be applied to multiplication algorithms for rectangular matrices. Here we do not balance the three dimensions into a square product, and instead keep track of each of them. Using tensors that are sums of two dot products, we will prove

$$\omega\left(1, \frac{2}{3 \log 3}, 1\right) = 2, \quad \omega(1, 0.539, 2) = 3. \quad (8.1)$$

While these bounds are weaker than the best known ones (see [Remark 8.10](#)), their proofs are short and they give a nice illustration of the approach.

8.1 The rectangular exponent

For nonnegative reals a, b, c we denote by $\omega(a, b, c)$ the infimum over all τ such that an $n^a \times n^b$ matrix can be multiplied by an $n^b \times n^c$ matrix using $O(n^\tau)$ arithmetic operations. In particular, $\omega = \omega(1, 1, 1)$. As in the square case, it is enough to bound the border rank.

Lemma 8.1 (Rectangular border rank). *If $\underline{R}(\langle n, m, \ell \rangle) \leq r$ then $\omega(\log n, \log m, \log \ell) \leq \log r$.*

Proof. The proof is identical to the square case. If $I_r \supseteq \langle n, m, \ell \rangle$ has degree d , then by taking the N th tensor power and interpolating we get that $\underline{R}(\langle n^N, m^N, \ell^N \rangle) \leq (dN + 1)r^N$. As the recursion of [Proposition 2.3](#) works for rectangular blocks as well, we get that $\omega(\log n, \log m, \log \ell) \leq \log r + \log(dN + 1)/N$, and the claim follows by letting N go to infinity. $\square$

We shall use the following simple properties of $\omega(a, b, c)$ which we leave as an exercise to the reader.

Exercise 8.2. *The following properties hold:*

1. Symmetry: $\omega(a, b, c)$ does not change when we permute a, b, c (see [Exercise 2.21](#)).
2. Monotonicity: ω is non-decreasing in each argument, as a smaller matrix product is a restriction of a larger one.
3. Homogeneity: $\omega(\lambda a, \lambda b, \lambda c) = \lambda \omega(a, b, c)$ for $\lambda \geq 0$, which follows by replacing n with n^λ .
4. Subadditivity:

$$\omega(a + a', b + b', c + c') \leq \omega(a, b, c) + \omega(a', b', c'), \quad (8.2)$$

which follows by taking the tensor product of the two algorithms.

5. Trivial lower bound: $\omega(a, b, c) \geq \max\{a + b, b + c, a + c\}$, since the outputs are linearly independent quadratic forms.
6. Blocking: increasing one of the arguments by δ increases ω by at most δ , e.g., $\omega(a + \delta, b, c) \leq \delta + \omega(a, b, c)$. Indeed, we can cut the larger dimension into n^δ blocks and handle each block separately.

Note that the last property implies that $\omega(a, b, c)$ is continuous.

8.2 Strassen's tensor

Let $q \geq 2$. We now consider the tensor that Strassen used in his laser method [Str87] (see also [Blä13, Section 8]):

$$T_q := \sum_{i=1}^q x_i y_0 z_i + \sum_{i=1}^q x_0 y_i z_i = B_Y(q) + B_X(q). \quad (8.3)$$

While T_q has $2q$ terms, its border rank is only $q + 1$.

Lemma 8.3 ([Str87, Lemma 7.4]). $\underline{R}(T_q) \leq q + 1$.

For completeness we give the proof.

Proof. It is not hard to verify that

$$\sum_{i=1}^q (x_0 + t x_i)(y_0 + t y_i) z_i - x_0 y_0 \sum_{i=1}^q z_i = t T_q + t^2 \sum_{i=1}^q x_i y_i z_i.$$

Dividing by t we get T_q plus terms that vanish when $t \rightarrow 0$, as a sum of $q + 1$ rank-one tensors. $\square$

Consider now the N th tensor power of T_q . It is a sum of 2^N products, where each product is a tensor product of k copies of $B_Y(q)$ and $N - k$ copies of $B_X(q)$, in some order. We keep only the $M = \binom{N}{k}$ products with a fixed k .

Lemma 8.4. For integers $0 \leq k \leq N$ and $M = \binom{N}{k}$,

$$I_{2M} \otimes T_q^{\otimes N} \supseteq I_M \otimes \langle q^k, M, q^{N-k} \rangle. \quad (8.4)$$

Consequently,

$$\omega(k \log q, \log M, (N - k) \log q) \leq N \log(q + 1) + 1. \quad (8.5)$$

Proof. Each of the M products that we keep is determined by its X -variables (and also by its Y -variables), while all of them share the Z -variables. Thus, Lemma 5.5, with Z as the shared leg, makes them into M independent tensors and attaches $B_Z(M)$ to each of them. After reordering the factors, each of these tensors is

$$B_Y(q^k) \otimes B_Z(M) \otimes B_X(q^{N-k}) \cong \langle q^k, M, q^{N-k} \rangle, \quad (8.6)$$

which gives (8.4).

For the second claim, note that Lemma 8.3 gives $I_{(q+1)^N} \supseteq T_q^{\otimes N}$, and hence $I_{2(q+1)^N} \otimes I_M \supseteq I_M \otimes \langle q^k, M, q^{N-k} \rangle$. The claim now follows from Lemma 8.5, which is a rectangular analog of Lemma 5.3. $\square$

Lemma 8.5 (Rectangular reuse). Let C, M, n, m, ℓ be positive integers such that

$$I_C \otimes I_M \supseteq I_M \otimes \langle n, m, \ell \rangle.$$

Then $\omega(\log n, \log m, \log \ell) \leq \log C$.

Proof. We apply the degeneration N times, where in each round we use the copy of I_M that was returned in the previous round, and at the end we drop the final I_M . This gives $I_{MC^N} \supseteq \langle n^N, m^N, \ell^N \rangle$, and so, by Lemma 8.1 and homogeneity,

$$\omega(\log n, \log m, \log \ell) \leq \log C + \frac{\log M}{N}.$$

The claim follows by letting N go to infinity. $\square$

In order to let N grow we need to estimate the binomial coefficient. Recall that the entropy function is $H(p) = -p \log p - (1-p) \log(1-p)$.⁸

The following is a special case of the conditional-label entropy bound of [Ope26b, Theorem 9.2]. We give a direct proof.

Theorem 8.6. *For every integer $q \geq 2$ and $0 \leq p \leq 1$,*

$$\omega(p \log q, H(p), (1-p) \log q) \leq \log(q+1). \quad (8.7)$$

Proof. For every N let $k_N = \lfloor pN \rfloor$ and $p_N = k_N/N$, and note that $p_N \rightarrow p$ as N goes to infinity. It is not hard to verify that

$$\binom{N}{k_N} p_N^{k_N} (1-p_N)^{N-k_N}$$

is a largest term among the $N+1$ terms in the binomial expansion of $(p_N + (1-p_N))^N = 1$. Hence it is at least $1/(N+1)$ and at most 1. Taking logarithms and dividing by N we get

$$H(p_N) - \frac{\log(N+1)}{N} \leq \frac{1}{N} \log \binom{N}{k_N} \leq H(p_N).$$

Since H is continuous, this shows that $\frac{1}{N} \log \binom{N}{k_N} \rightarrow H(p)$.

Now apply (8.5) with $k = k_N$, so that $M = \binom{N}{k_N}$, and divide by N . By homogeneity,

$$\omega\left(p_N \log q, \frac{1}{N} \log \binom{N}{k_N}, (1-p_N) \log q\right) \leq \log(q+1) + \frac{1}{N}.$$

As N goes to infinity, the three arguments on the left tend to $p \log q$, $H(p)$ and $(1-p) \log q$, and the right-hand side tends to $\log(q+1)$. The claim now follows from the continuity of ω . $\square$

8.3 Two exact values of the rectangular exponent

We now choose the parameters. Put

$$u = \frac{2}{3 \log 3} = 0.4206 \dots$$

Corollary 8.7. $\omega(1, u, 1) = 2$. In other words, $\alpha \geq u$, where $\alpha = \sup\{a : \omega(1, a, 1) = 2\}$.

Proof. Let $q = 2$ and $p = 2/3$. As $H(2/3) = \log 3 - \frac{2}{3}$, we get from Theorem 8.6, by dividing by $\log 3$, $\omega(u, 1-u, u/2) \leq 1$. Hence, by symmetry,

$$\omega(u, u/2, 1-u) \leq 1 \quad \text{and} \quad \omega(1-u, u/2, u) \leq 1.$$

Adding the two bounds using (8.2) gives $\omega(1, u, 1) \leq 2$, and the reverse inequality is the trivial lower bound. $\square$

We next consider the family $\omega(1, a, 2)$. Here the trivial lower bound is 3, and the question is how large a can be while keeping $\omega(1, a, 2) = 3$. The same computation as above, with T_2 alone, gives $\omega(1, 1/2, 2) = 3$. We can do better by combining the bounds obtained from T_2 and from T_4 .

⁸As usual, $0 \log 0 = 0$.

Corollary 8.8. Let $\gamma = \frac{8}{15 \log 5 - 20} \geq 0.539$. Then $\omega(1, \gamma, 2) = 3$.

Proof. Let $q = 2$ and $p = 2/3$, as in the previous proof. After permuting the arguments,

$$\omega\left(\frac{2}{3}, \frac{1}{3}, \log 3 - \frac{2}{3}\right) \leq \log 3.$$

With $q = 4$ and $p = 1/5$ we have $H(1/5) = \log 5 - \frac{8}{5}$, and after permuting,

$$\omega\left(\log 5 - \frac{8}{5}, \frac{2}{5}, \frac{8}{5}\right) \leq \log 5.$$

In both bounds, the right-hand side equals the sum of the first and third arguments. The idea is to add the two bounds with weights for which the third argument becomes (almost) twice the first one, and then to use blocking ([Exercise 8.2\(6\)](#)). Specifically, we add twice the first bound and five times the second, and get, by homogeneity and subadditivity, that for

$$c_1 = 5 \log 5 - \frac{20}{3}, \quad c_2 = \frac{8}{3}, \quad c_3 = 2 \log 3 + \frac{20}{3},$$

it holds that

$$\omega(c_1, c_2, c_3) \leq 2 \log 3 + 5 \log 5.$$

Clearly $c_1 + c_3 = 2 \log 3 + 5 \log 5$. Moreover, calculating we see that $c_3 \leq 2c_1$. Thus, by [Exercise 8.2\(6\)](#), we can increase the third argument from c_3 to $2c_1$ at a cost of at most $2c_1 - c_3$, and get

$$\omega(c_1, c_2, 2c_1) \leq c_1 + c_3 + (2c_1 - c_3) = 3c_1.$$

Dividing by c_1 we get that $\omega(1, c_2/c_1, 2) \leq 3$. The claim follows as $c_2/c_1 = \gamma$, and, as before, 3 is also a lower bound. $\square$

Exercise 8.9 (Better choice of parameters). (a) Fix an integer $q \geq 2$. For which $p \in [0, 1]$ does the upper bound

$$\omega(p \log q, H(p), (1-p) \log q) \leq \log(q+1)$$

equal the flattening lower bound for the same three dimensions? Use your answer to explain the choices $p = 2/3$ for $q = 2$ and $p = 1/5$ for $q = 4$.

(b) Using only the two bounds in the proof of [Corollary 8.8](#), improve the conclusion to

$$\omega(1, 0.54, 2) = 3.$$

Hints. For (a), maximize $H(p) + p \log q$ (or $H(p) + (1-p) \log q$, depending on the case). For (b), allow positive real weights in place of 2 and 5, and choose them so that the third argument of the sum is exactly twice the first.

Remark 8.10. We note that stronger rectangular bounds are known. Shortly before [\[Ope26a\]](#), OpenAI [\[Ope26b\]](#) proved $\alpha > 0.465$ over fields of characteristic zero. Over $\mathbb{C}$, [Uznański \[Uzn26, Corollary 1.2\]](#) proved

$$\omega(1, a, 1) \leq \begin{cases} 2, & 0 \leq a \leq \frac{1}{2} \\ 1 + a + \frac{1}{4a}, & a \geq \frac{1}{2} \end{cases}$$

and in particular $\alpha \geq 1/2$. For the second family, [Uznański \[Uzn26, Theorem 1.1\]](#) also obtains

$$\omega(1, 2/3, 2) = 3$$

over fields of characteristic zero. Over every fixed field, Song [Son26, Theorem 3.9] proved that $\omega(1, 0.635269, 2) = 3$.

These results use further properties of the tensor constructions, whereas our goal here was to show that a short application of separation and reuse already gives nontrivial bounds. For example, Corollary 8.7 improves on the bound $\alpha > 0.321334$ [VWXXZ24], which was the best known before the recent work of OpenAI [Ope26b, Ope26a].

9 Concluding remarks and open problems

We end with a few remarks and questions. For the square case, the numerical argument uses symmetry, the boundary values $F(1, b) = b$, concavity, and shifted tripling with $c = a$. These imply that $F(a, a) \geq \Lambda_a$, which grows like $a^{4/3}$. On the other hand, the upper bound coming from flattenings is $F(a, a) \leq a\sqrt{2a-1}$. If we could prove a lower bound of the form $F(a, a) \geq a^{3/2-o(1)}$, then this would imply $\omega = 2$. There is a gap already for $a = 2$, as the bounds proved here only give

$$\frac{10}{3} \leq F(2, 2) \leq 2\sqrt{3}.$$

Are there other decompositions of polynomial multiplication that give stronger inequalities for F , or that determine $F(2, 2)$? Note that by Exercise 4.4, improving the lower bound requires inequalities beyond those used in Lemma 4.3.

The rectangular constructions use the same separation and reuse arguments, except that we keep track of the three dimensions separately. Already the four-term tensor T_2 gives $\omega(1, u, 1) = 2$ for $u = 2/(3 \log 3)$, and combining the constructions for T_2 and T_4 gives $\omega(1, 0.539, 2) = 3$. It would be interesting to understand how far these elementary arguments can go when starting from other small tensors.

References

- [ADVW⁺25] Josh Alman, Ran Duan, Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. More asymmetry yields faster matrix multiplication. In *Proceedings of the 2025 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2005–2039, 2025. doi:10.1137/1.9781611978322.63.
- [AFLG15] Andris Ambainis, Yuval Filmus, and François Le Gall. Fast matrix multiplication: limitations of the Coppersmith–Winograd method. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing (STOC)*, pages 585–593, 2015.
- [Alm21] Josh Alman. Limits on the universal method for matrix multiplication. *Theory of Computing*, 17(1):1–30, 2021.
- [AvdBZ26] Alimzhan Amanov, Maxim van den Berg, and Jeroen Zuiddam. Multiplying matrices via polynomials: exposition and barriers, 2026. URL: <https://arxiv.org/abs/2610.30054>, arXiv:2610.30054.
- [AVW18] Josh Alman and Virginia Vassilevska Williams. Limits on all known (and some unknown) approaches to matrix multiplication. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 580–591, 2018.

- [AVW21] Josh Alman and Virginia Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 522–539, 2021.
- [BCRL79] Dario Bini, Milvio Capovani, Francesco Romani, and Grazia Lotti. $O(n^{2.7799})$ complexity for $n \times n$ approximate matrix multiplication. *Information Processing Letters*, 8(5):234–235, 1979.
- [BCS97] Peter Bürgisser, Michael Clausen, and M. Amin Shokrollahi. *Algebraic Complexity Theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften*. Springer, 1997.
- [Bin80] Dario Bini. Relations between exact and approximate bilinear algorithms. Applications. *Calcolo*, 17(1):87–97, 1980. doi:[10.1007/BF02575865](https://doi.org/10.1007/BF02575865).
- [Blä99] Markus Bläser. A $\frac{5}{2}n^2$ -lower bound for the rank of $n \times n$ -matrix multiplication over arbitrary fields. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 45–50, 1999.
- [Blä13] Markus Bläser. Fast Matrix Multiplication. Theory of Computing Library, Graduate Surveys, no. 5, 2013. doi:[10.4086/toc.gs.2013.005](https://doi.org/10.4086/toc.gs.2013.005).
- [CKSU05] Henry Cohn, Robert Kleinberg, Balázs Szegedy, and Christopher Umans. Group-theoretic algorithms for matrix multiplication. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 379–388, 2005.
- [CU03] Henry Cohn and Christopher Umans. A group-theoretic approach to fast matrix multiplication. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 438–449, 2003.
- [CVZ21] Matthias Christandl, Péter Vrana, and Jeroen Zuiddam. Barriers for fast matrix multiplication from irreversibility. *Theory of Computing*, 17(2):1–32, 2021.
- [CVZ23] Matthias Christandl, Péter Vrana, and Jeroen Zuiddam. Universal points in the asymptotic spectrum of tensors. *Journal of the American Mathematical Society*, 36:31–79, 2023. doi:[10.1090/jams/996](https://doi.org/10.1090/jams/996).
- [CW90] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9(3):251–280, 1990. doi:[10.1016/S0747-7171\(08\)80013-2](https://doi.org/10.1016/S0747-7171(08)80013-2).
- [DEK⁺26] Emilien Dupont, Marvin Eisenberger, Borislav Kozlovskii, Abbas Mehrabian, Francisco J. R. Ruiz, Abigail See, Renfei Zhou, Josh Alman, Virginia Vassilevska Williams, and Matej Balog. Improving the matrix multiplication exponent with modern optimization and AlphaEvolve, 2026. URL: <https://arxiv.org/abs/2608.16884>, arXiv:2608.16884.
- [DS13] Alexander M. Davie and Andrew J. Stothers. Improved bound for complexity of matrix multiplication. *Proceedings of the Royal Society of Edinburgh Section A*, 143:351–369, 2013. doi:[10.1017/S0308210511001648](https://doi.org/10.1017/S0308210511001648).

- [DWZ23] Ran Duan, Hongxun Wu, and Renfei Zhou. Faster matrix multiplication via asymmetric hashing. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 2129–2138, 2023. doi:[10.1109/FOCS57990.2023.00130](https://doi.org/10.1109/FOCS57990.2023.00130).
- [Lan06] Joseph M. Landsberg. The border rank of the multiplication of 2×2 matrices is seven. *Journal of the American Mathematical Society*, 19(2):447–459, 2006.
- [Lan14] Joseph M. Landsberg. New lower bounds for the rank of matrix multiplication. *SIAM Journal on Computing*, 43(1):144–149, 2014.
- [LG14] François Le Gall. Powers of tensors and fast matrix multiplication. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, ISSAC '14, pages 296–303, New York, NY, USA, 2014. Association for Computing Machinery. URL: <https://arxiv.org/abs/1401.7714>, doi:[10.1145/2608628.2608664](https://doi.org/10.1145/2608628.2608664).
- [LM18] Joseph M. Landsberg and Mateusz Michałek. A $2n^2 - \log_2(n) - 1$ lower bound for the border rank of matrix multiplication. *International Mathematics Research Notices*, 2018(15):4722–4733, 2018.
- [LO15] Joseph M. Landsberg and Giorgio Ottaviani. New lower bounds for the border rank of matrix multiplication. *Theory of Computing*, 11:285–298, 2015.
- [MR13] Alex Massarenti and Emanuele Raviolo. The rank of $n \times n$ matrix multiplication is at least $3n^2 - 2\sqrt{2}n^{3/2} - 3n$. *Linear Algebra and its Applications*, 438(11):4500–4509, 2013.
- [MR14] Alex Massarenti and Emanuele Raviolo. Corrigendum to "The rank of $n \times n$ matrix multiplication is at least $3n^2 - 2\sqrt{2}n^{3/2} - 3n$ " [Linear Algebra Appl. 438 (11) (2013) 4500–4509]. *Linear Algebra and its Applications*, 445:369–371, 2014. doi:[10.1016/j.laa.2013.12.009](https://doi.org/10.1016/j.laa.2013.12.009).
- [Ope26a] OpenAI. An Upper Bound of 9/4 for the Matrix Multiplication Exponent. OpenAI Math Release preprint [OAI:Matrix-Multiplication-Nine-Fourths-October-2-2026](https://arxiv.org/abs/2608.11111), 2026.
- [Ope26b] OpenAI. Complex Matrix Multiplication Below 2.258 and Rectangular Bounds. OpenAI Math Release preprint [OAI:Complex-Matrix-Multiplication-Below-2.258-and-Rectangular-Bounds-September-24-2026](https://arxiv.org/abs/2608.11111), 2026.
- [Pan78] Victor Ya. Pan. Strassen’s algorithm is not optimal: trilinear technique of aggregating, uniting and canceling for constructing fast algorithms for matrix operations. In *Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 166–176, 1978.
- [Pas26] Rafael Pass. Matrix multiplication from polynomial packings: A constructive and elementary proof of $\omega \leq 9/4$, 2026. URL: <https://arxiv.org/abs/2610.28627>, arXiv:[2610.28627](https://arxiv.org/abs/2610.28627).
- [Raz03] Ran Raz. On the complexity of matrix product. *SIAM Journal on Computing*, 32(5):1356–1369, 2003.

- [RS03] Ran Raz and Amir Shpilka. Lower bounds for matrix product in bounded depth circuits with arbitrary gates. *SIAM Journal on Computing*, 32(2):488–513, 2003. doi:[10.1137/S009753970138462X](https://doi.org/10.1137/S009753970138462X).
- [Sch81] Arnold Schönhage. Partial and total matrix multiplication. *SIAM Journal on Computing*, 10(3):434–455, 1981. doi:[10.1137/0210032](https://doi.org/10.1137/0210032).
- [Shp03] Amir Shpilka. Lower bounds for matrix product. *SIAM Journal on Computing*, 32(5):1185–1200, 2003.
- [Son26] Zhao Song. Faster asymmetric rectangular matrix multiplication. SSRN preprint 7600178, October 2026. URL: <https://ssrn.com/abstract=7600178>.
- [Sto10] Andrew J. Stothers. *On the complexity of matrix multiplication*. PhD thesis, University of Edinburgh, 2010.
- [Str69] Volker Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13:354–356, 1969. doi:[10.1007/BF02165411](https://doi.org/10.1007/BF02165411).
- [Str83] Volker Strassen. Rank and optimal computation of generic tensors. *Linear Algebra and its Applications*, 52/53:645–685, 1983.
- [Str87] Volker Strassen. Relative bilinear complexity and matrix multiplication. *Journal für die reine und angewandte Mathematik*, 375/376:406–443, 1987. doi:[10.1515/crll.1987.375-376.406](https://doi.org/10.1515/crll.1987.375-376.406).
- [Str88] Volker Strassen. The asymptotic spectrum of tensors. *Journal für die reine und angewandte Mathematik*, 384:102–152, 1988. doi:[10.1515/crll.1988.384.102](https://doi.org/10.1515/crll.1988.384.102).
- [Uzn26] Przemysław Uznański. Rectangular matrix multiplication from polynomial products, October 2026. URL: <https://arxiv.org/abs/2610.10494>, arXiv:2610.10494.
- [VW12] Virginia Vassilevska Williams. Multiplying matrices faster than Coppersmith–Winograd. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*, pages 887–898, 2012. doi:[10.1145/2213977.2214056](https://doi.org/10.1145/2213977.2214056).
- [VWXXZ24] Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. New bounds for matrix multiplication: from alpha to omega. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3792–3835, 2024. doi:[10.1137/1.9781611977912.134](https://doi.org/10.1137/1.9781611977912.134).
- [Win71] Shmuel Winograd. On multiplication of 2×2 matrices. *Linear Algebra and its Applications*, 4(4):381–388, 1971.
- [WZ26] Avi Wigderson and Jeroen Zuiddam. Asymptotic spectra: theory, applications, and extensions. *Bulletin of the American Mathematical Society*, 63(2):199–331, 2026. doi:[10.1090/bull/1880](https://doi.org/10.1090/bull/1880).
- [Zui26] Jeroen Zuiddam. $\omega(F) \leq 9/4$ over every field, in Lean. Lean formalization and accompanying documentation, GitHub, October 2026. URL: <https://github.com/jzuiddam/omega-nine-quarters-all-fields>.

A The proof of [Ope26a] via spectral points

In this appendix we sketch the proof of [Ope26a], and point out which of the general facts about spectral points that it uses are due to Strassen [Str88]. Many of these facts have short and elementary proofs, but here we only describe the high-level approach.

The basic constructions are the same as in the main body of the paper: the separation lemma (Lemma 5.1; [Ope26a, Proposition 3.1]), the midpoint degeneration with $s = 1$ (Lemma 5.6; [Ope26a, proof of Lemma 4.1]) and the three-sector degeneration (Lemma 5.2; [Ope26a, proof of Lemma 4.2]). The difference is in how they are used to bound ω . [Ope26a] evaluates every tensor under a *spectral point*. Each degeneration then becomes an inequality between two real numbers, which are the values of the spectral point on the two tensors. These inequalities are used to prove that $\lambda(M_n) \leq n^{9/4}$ for every spectral point λ . A theorem of Strassen provides a spectral point whose value on M_n is close to n^ω , and this implies $\omega \leq 9/4$.

We start by introducing the notion of a spectral point of Strassen [Str88, Section 3]. These are the points of his asymptotic spectrum. In [Ope26a, Definition 2.1] they are called tensor characters.

Spectral points. A *spectral point* is a map λ from tensors to $\mathbb{R}_{\geq 0}$ such that for all tensors U, V

$$\lambda(U \oplus V) = \lambda(U) + \lambda(V), \quad \lambda(U \otimes V) = \lambda(U)\lambda(V), \quad \lambda(I_1) = 1,$$

and $\lambda(U) \geq \lambda(V)$ whenever $U \geq V$. For example, the rank of the X -flattening is a spectral point (it is one of Strassen's gauge points [Str88, Section 3]): it is multiplicative and monotone by Lemma 2.17(F2),(F1), and additive since the flattening of a direct sum is block diagonal. Clearly $\lambda(I_r) = r$, and since $I_{R(U)} \geq U \geq I_1$ for every nonzero U (Lemma 2.17) we have $1 \leq \lambda(U) \leq R(U)$. Spectral points are also monotone under degenerations [Str88, Theorem 3.1].

Fix a spectral point λ . The following characterization of the values of λ on matrix multiplication tensors is due to Strassen [Str88, Proposition 4.3]. There are $p_X, p_Y, p_Z \in [0, 1]$ with $\lambda(B_X(r)) = r^{p_X}$, $\lambda(B_Y(r)) = r^{p_Y}$, $\lambda(B_Z(r)) = r^{p_Z}$ for all r . Hence, by Lemma 2.22,

$$\lambda(M_r) = r^{p_X + p_Y + p_Z} = r^{3\theta}, \quad \theta = \theta_\lambda = \frac{p_X + p_Y + p_Z}{3}.$$

As $M_r \geq I_r$ we have $\theta \geq 1/3$. (In fact $\theta \geq 2/3$, since $\min_\lambda \lambda(M_2) = 4$ [Str88, Proposition 4.2].)

For a permutation π of the legs, $\lambda_\pi(U) = \lambda(U^\pi)$ is also a spectral point. We denote its X -exponent by $p_X^{(\pi)}$. As π runs over S_3 , the scalar leg of $B_X(r)^\pi$ occupies each of the three positions twice. Thus each of p_X, p_Y, p_Z appears twice, giving [Ope26a, Equation (4.3)]

$$\sum_{\pi \in S_3} p_X^{(\pi)} = 6\theta. \tag{A.1}$$

The profile P . For the fixed spectral point λ , the profile defined in [Ope26a, Equation (4.4)] can be written in our notation as

$$P(a, b) = \lambda(Q(a, b))^{1/(6\theta)} = \left(\prod_{\pi \in S_3} \lambda_\pi(C(a, b)) \right)^{1/(6\theta)}.$$

Its basic properties are recorded in [Ope26a, Equation (4.5)]. By Lemma 3.3, $P(a, b) = P(b, a)$. Since $C(1, b) = B_X(b)$, Lemma 3.4 gives $\lambda(Q(1, b)) = \lambda(M_{b^2}) = b^{6\theta}$, that is, $P(1, b) = b$. Finally, $\lambda(Q(a, b)) \leq R(Q(a, b)) \leq (a + b - 1)^6$, so

$$P(a, b) \leq (a + b - 1)^{1/\theta}. \tag{A.2}$$

Note the relation with our rate. If $\mathbf{Q}(a, b)^{\otimes m} \supseteq \mathbf{M}_n$ then $\lambda(\mathbf{Q}(a, b))^m \geq \lambda(\mathbf{M}_n) = n^{3\theta}$, that is, $P(a, b) \geq n^{1/(2m)}$. Thus $P(a, b) \geq F(a, b)$ for every spectral point. In fact, Strassen's duality theorem implies that $F(a, b)$ is the minimum of $P(a, b)$ over all spectral points (this is the rate formula of [WZ26, Corollary 3.23]). So the two proofs give $F(a, a) \geq \Lambda_a$, but ours obtains it directly from degenerations, without spectral points.

The entropy inequality. The separation lemma becomes the following inequality [Ope26a, Corollary 3.2]. Let $T = \sum_{i \in \mathcal{I}} T_i$, where the T_i are nonzero, share the X -variables and use disjoint groups of Y - and Z -variables, as in Lemma 5.1. Then for every probability vector $q = (q_i)_{i \in \mathcal{I}}$,

$$\lambda(T) \geq 2^{p_X H(q)} \prod_{i \in \mathcal{I}} \lambda(T_i)^{q_i}, \quad \text{where } H(q) = - \sum_i q_i \log q_i, \quad (\text{A.3})$$

is the entropy function.

We use our $2M$ -copy separation lemma. First suppose that q is rational and let N run through multiples of a common denominator of the q_i . Put $N_i = Nq_i$ and $M = N! / \prod_i N_i!$. By Lemma 5.5,

$$I_{2M} \otimes T^{\otimes N} \supseteq I_M \otimes \left(\bigotimes_i T_i^{\otimes N_i} \right) \otimes B_X(M).$$

Applying λ gives

$$2M \cdot \lambda(T)^N \geq M \cdot M^{p_X} \cdot \prod_i \lambda(T_i)^{Nq_i}.$$

Taking N th roots, we note that since $\frac{1}{N} \log M \rightarrow H(q)$, (A.3) for rational q follows. Since $\lambda(T_i) \geq 1$, the right-hand side is continuous in q , so the general case follows by continuity.

Concavity [Ope26a, Lemma 4.1] and tripling [Ope26a, Lemma 4.2]. For $a \geq 1$, $b \geq 2$ and $s = 1$, Lemma 5.6 gives $\mathbf{C}(a, b) \otimes B_X(2) \supseteq \mathbf{C}(a, b+1) + \mathbf{C}(a, b-1)$, where the two blocks share the X -variables. Apply (A.3) with the spectral point λ_π and $q = (q_+, q_-)$:

$$2^{p_X^{(\pi)}} \lambda_\pi(\mathbf{C}(a, b)) \geq 2^{p_X^{(\pi)} H(q)} \lambda_\pi(\mathbf{C}(a, b+1))^{q_+} \lambda_\pi(\mathbf{C}(a, b-1))^{q_-}. \quad (\text{A.4})$$

We use the same probability vector q for all six permutations. By (A.1) and the definition of P ,

$$\sum_{\pi \in S_3} p_X^{(\pi)} = 6\theta, \quad \prod_{\pi \in S_3} \lambda_\pi(\mathbf{C}(a, b)) = P(a, b)^{6\theta}.$$

Thus, multiplying the six inequalities gives

$$2^{6\theta} P(a, b)^{6\theta} \geq 2^{6\theta H(q)} P(a, b+1)^{6\theta q_+} P(a, b-1)^{6\theta q_-}.$$

Taking the power $1/(6\theta)$ on both sides, we obtain

$$2P(a, b) \geq 2^{H(q)} P(a, b+1)^{q_+} P(a, b-1)^{q_-}.$$

For $A, B > 0$ and $q = (A, B)/(A+B)$ we have $2^{H(q)} A^{q_+} B^{q_-} = A+B$, since the logarithm of the left-hand side is $q_+ \log(A/q_+) + q_- \log(B/q_-) = \log(A+B)$. Taking $A = P(a, b+1)$ and $B = P(a, b-1)$ gives $2P(a, b) \geq P(a, b+1) + P(a, b-1)$ as in Theorem 4.1.

Next, we derive the tripling inequality from the three-sector degeneration. For positive integers a, c , Lemma 5.2 gives $\mathbf{C}(a, 3c+a-1) \supseteq T_0 + T_1 + T_2$, where T_0 and T_2 are copies of $\mathbf{C}(a, c)$ and $T_1 \cong \mathbf{C}(a, c)^\tau$, with τ exchanging the Y - and Z -legs. (The reversal of the X -indices in T_1 does not

matter here, since (A.3) only uses the values $\lambda(T_i)$ of the individual blocks.) Applying (A.3) with $q = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ gives

$$\lambda_\pi(\mathbf{C}(a, 3c + a - 1)) \geq 3^{p_X^{(\pi)}} \lambda_\pi(\mathbf{C}(a, c))^{2/3} \lambda_\pi(\mathbf{C}(a, c)^\tau)^{1/3}.$$

Composition with τ permutes the six leg permutations, hence $\prod_\pi \lambda_\pi(\mathbf{C}(a, c)^\tau) = \prod_\pi \lambda_\pi(\mathbf{C}(a, c))$. Multiplying the six inequalities and raising to the power $1/(6\theta)$ gives $P(a, 3c + a - 1) \geq 3P(a, c)$, as we proved for F in Theorem 4.2.

Bounding θ . We have shown that P is symmetric, satisfies $P(1, b) = b$, and satisfies the concavity and tripling inequalities. These are exactly the hypotheses of Lemma 4.3, so $P(a, a) \geq \Lambda_a > a^{4/3}$ for $a \geq 2$. Together with (A.2) we get $a^{4/3} < (2a - 1)^{1/\theta}$, that is,

$$\theta < \frac{3}{4} \cdot \frac{\log(2a - 1)}{\log a} \quad \text{for every } a \geq 2.$$

Letting $a \rightarrow \infty$ gives $\theta \leq 3/4$. Hence $\lambda(\mathbf{M}_n) = n^{3\theta} \leq n^{9/4}$ for every spectral point λ and every n . This is the first part of the proof of Theorem 1.1 in [Ope26a, Section 5].

From spectral points to ω . To conclude, one needs spectral points that are large on $\mathbf{M}_n$. Such spectral points are provided by a theorem of Strassen [Str88, Theorem 3.8]: the asymptotic rank $\tilde{\mathbf{R}}(U)$ of every tensor U (Exercise 2.26) is the maximum of $\lambda(U)$ over all spectral points λ . Since $\mathbf{R}(\mathbf{M}_d^{\otimes N}) = \mathbf{R}(\mathbf{M}_{d^N}) \geq d^{N\omega}$ for every N , we have $\tilde{\mathbf{R}}(\mathbf{M}_d) \geq d^\omega$. Hence the following lemma, which in [Ope26a, Lemma 2.2] is called the *detecting characters* lemma, is a special case of Strassen's theorem.

Lemma A.1 (Strassen [Str88, Theorem 3.8]). *Let $d \geq 2$ and $k \geq 0$ be integers with $k < d^\omega$. Then there is a spectral point λ with $\lambda(\mathbf{M}_d) \geq k$.*

Given the lemma, take $k = \lceil d^\omega \rceil - 1$. Then $d^\omega - 1 \leq k \leq \lambda(\mathbf{M}_d) \leq d^{9/4}$, and letting $d \rightarrow \infty$ gives $\omega \leq 9/4$. This completes the argument in [Ope26a].

We also note that, by Strassen's theorem, $\omega = 3 \max_\lambda \theta_\lambda$. Indeed, since $\omega = \inf_{n \geq 2} \log_n \mathbf{R}(\mathbf{M}_n)$, we have $\tilde{\mathbf{R}}(\mathbf{M}_2) = 2^\omega$ (see Exercise 2.26), and so $2^\omega = \max_\lambda \lambda(\mathbf{M}_2) = 2^{3 \max_\lambda \theta_\lambda}$ [Str88, Proposition 4.2]. Thus bounding θ_λ for every spectral point, as done above, is exactly what is needed in order to bound ω .

A self-contained proof of Lemma A.1 is given in [WZ26, Sections 2 and 3], where the statement used here appears as [WZ26, Corollary 3.30]. [Ope26a, Appendix A] gives a different, direct proof of the special case needed here.

We refer the reader to [AvdBZ26] for a detailed exposition of the spectral point approach.