



Designing Strassen's Algorithm

Joshua A. Grochow* and Cristopher Moore†

August 30, 2017

Abstract

In 1969, Strassen shocked the world by showing that two $n \times n$ matrices could be multiplied in time asymptotically less than $O(n^3)$. While the recursive construction in his algorithm is very clear, the key gain was made by showing that 2×2 matrix multiplication could be performed with only 7 multiplications instead of 8. The latter construction was arrived at by a process of elimination and appears to come out of thin air. Here, we give the simplest and most transparent proof of Strassen's algorithm that we are aware of, using only a simple unitary 2-design and a few easy lines of calculation. Moreover, using basic facts from the representation theory of finite groups, we use 2-designs coming from group orbits to generalize our construction to all $n \geq 2$ (although the resulting algorithms aren't optimal for $n \geq 3$).

1 Introduction

The complexity of matrix multiplication is a central question in computational complexity, bearing on the complexity not only of most problems in linear algebra, but also of myriad combinatorial problems, e.g., various shortest path problems [Zwi02] and bipartite matching problems [San09]. The main question around matrix multiplication is whether two $n \times n$ matrices can be multiplied in time $O(n^{2+\varepsilon})$ for every $\varepsilon > 0$. The current best upper bound on this exponent is 2.3728639 [LG14], narrowly beating [DS13, Wil12]. The best known lower bound is still only $3n^2 - o(n)$ [Lan14].

Since Strassen's 1969 paper [Str69], which showed how to beat the standard $O(n^3)$ time algorithm, it has been understood that one way to get asymptotic improvements in algorithms for matrix multiplication is to find algebraic algorithms for multiplying small matrices using only a few multiplications, and then to apply these algorithms recursively.

While the recursive construction in Strassen's algorithm is very clear—treat a $2n \times 2n$ matrix as a 2×2 matrix each of whose entries is an $n \times n$ matrix—the base case, which accounts for how Strassen was able to beat $O(n^3)$, seems to come out of thin air. Indeed, Strassen was trying to prove, by process of (intelligently exhaustive) elimination, that such an algorithm could *not* exist (e.g., [Lan08, Remark 1.1.1] or [LR13]). In his paper it is presented as follows, which “one easily sees” [Str69, p. 355] correctly computes 2×2 matrix multiplication $C = AB$:

$$\begin{array}{rcl} C_{11} & = & I + IV - V + VII \\ C_{21} & = & II + IV \end{array} \quad \begin{array}{rcl} C_{12} & = & III + V \\ C_{22} & = & I + III - II + VI \end{array} ,$$

*Departments of Computer Science and Mathematics, University of Colorado, Boulder, CO
jgrochow@colorado.edu

†The Santa Fe Institute, Santa Fe, NM moore@santafe.edu

where

$$\begin{array}{ll}
 I &= (A_{11} + A_{22})(B_{11} + B_{22}) & V &= (A_{11} + A_{12})B_{22} \\
 II &= (A_{21} + A_{22})B_{11} & VI &= (-A_{11} + A_{21})(B_{11} + B_{12}) \\
 III &= A_{11}(B_{12} - B_{22}) & VII &= (A_{12} - A_{22})(B_{21} + B_{22}) \\
 IV &= A_{22}(-B_{11} + B_{21}) & &
 \end{array}
 .$$

While verifying the above by calculation is not difficult—after all, it’s only seven multiplications and four linear combinations—it is rather un-illuminating. In particular, the verification gives no sense of *why* such a decomposition exists.

In this paper, we give a proof of Strassen’s algorithm that is the most transparent we are aware of. The basic idea is to note that Strassen’s algorithm has a symmetric group of vectors lurking in it, which form what is known as a (unitary) 2-design. Using the representation theory of finite groups, we obtain generalizations to higher dimensions, which suggest further directions to explore in our hunt for efficient algorithms.

1.1 Other explanations of Strassen’s algorithm

Landsberg [Lan08, Section 3.8] points out that Strassen’s algorithm could have been anticipated because the border-rank of any $4 \times 4 \times 4$ tensor is at most seven. Although this may lead one to suspect the existence of an algorithm such as Strassen’s, it does not give an explanation for the fact that the rank (rather than border-rank) of 2×2 matrix multiplication is at most seven, nor does it give an explanation of Strassen’s particular construction.

Several authors have tried to make Strassen’s construction more transparent through various calculations, e. g., [Gas71, Yuv78, Cha86, Ale97, Pat09, GK00, Min15, CILO16]. While these lend some insight, and some provide proofs that are perhaps easier to remember (and teach) than Strassen’s original presentation, each of them either involves some ad hoc constructions or some un-illuminating calculations, which are often left to the reader. We feel that they do not really offer *conceptual* explanations for the fact that the rank of MM_2 is at most 7.

Clausen [Cla88] (see [BCS97, pp. 11–12] for a more widely available explanation in English) showed how one can use group orbits to show that the rank of MM_2 is at most 7. In fact, Clausen’s beautiful construction was one of the starting points of our investigation. However, that construction relies on a seemingly magical property of a certain 4×4 multiplication table. More recently, Ikenmeyer and Lysikov [IL17] gave a beautiful explanation of Clausen’s construction, but ultimately their proof for Strassen’s algorithm still relies on the same magical property of the same 4×4 multiplication table, and it is not immediately obvious how to generalize to all n . In contrast, our result easily generalizes to all n , and more generally to orbits of any irreducible representation of any finite group.

1.2 Related work

This paper is a simplified and self-contained version of Section 5 of [GM16], in which we explored highly symmetric algorithms for multiplying matrices. Recently, there have been several papers analyzing the geometry and symmetries of algebraic algorithms for small matrices [Bur14, Bur15, LR16, LM16, CILO16]. In [GM16], we tried to take this line of research one step further by using symmetries to discover *new* algorithms for multiplying matrices of small size. While those algorithms did not improve the state-of-the-art bounds on the matrix multiplication exponent, they suggested that we can use group symmetries and group orbits to find new algorithms for matrix multiplication. In addition to their potential value for future endeavors, we believe that these

highly symmetric matrix multiplication algorithms are beautiful in their own right, and deserve to be shared simply for their beauty.

Although the method of construction suggested in [GM16], and independently in [CILO16], is more general than this, the constructions we ended up finding in [GM16] were in fact all instances of a single design-based construction yielding $n^3 - n + 1$ multiplications for $n \times n$ matrix multiplication. The proof that this construction works is the simplest and most transparent proof of Strassen’s algorithm that we are aware of.

One may also reasonably wonder whether there is any relationship between our group-based construction and the family of group-based constructions suggested by Cohn and Umans [CU03], including the constructions given in [CKSU05] and generalizations in [CU13]. While there may be a common generalization that captures both methods, at the moment we don’t know of any direct relationship between the two. Indeed, one cannot use the group-theoretic approach of [CU03] to explain Strassen’s result, even though the constructions of [CKSU05] get a better exponent: The only way to use their approach for the 2×2 case is to embed MM_2 into the cyclic group C_7 , but Cohn and Umans showed that one could not beat n^3 using only abelian groups. (Some of their more complicated constructions can beat n^3 in abelian groups, but those involve embedding multiple copies of MM into the same group simultaneously, whereas here we are explicitly talking about embedding a single copy of MM_2 .)

2 Complexity, symmetry, and designs

For general background on algebraic complexity, we refer the reader to the book [BCS97]. Bläser’s survey article [Blä13], in addition to excellent coverage around matrix multiplication, has a nice tutorial on tensors and their basic properties.

In the algebraic setting, since matrix multiplication is a bilinear map, it is known that it can be reformulated as a tensor, and that the algebraic complexity of matrix multiplication is within a factor of 2 of the rank of this tensor. The matrix multiplication tensor for $n \times n$ matrices is

$$\text{MM}_{def}^{abc} = \delta_e^a \delta_f^b \delta_d^c, \quad (1)$$

where the indices range from 1 to n , and where δ is the Kronecker delta, $\delta_b^a = 1$ if $a = b$ and 0 if $a \neq b$. This is also defined by the inner product

$$\langle \text{MM} | A \otimes B \otimes C \rangle = \text{tr } ABC.$$

Given vector spaces V_1, \dots, V_k , a vector $v \in V_1 \otimes \dots \otimes V_k$ is said to have tensor rank one if it is a separable tensor, that is, of the form $v_1 \otimes v_2 \otimes \dots \otimes v_k$ for some $v_i \in V_i$. The *tensor rank* of v is the smallest number of rank-one tensors whose sum is v . In the case of MM_n , we have $k = 3$ and $V_1 \cong V_2 \cong V_3 \cong \mathbb{C}^{n \times n}$.

The matrix multiplication tensor MM is characterized by its symmetries (e.g., [BI11]). That is, up to a constant, it is the unique operator fixed under the following action of $\text{GL}(n)^3$: given $X, Y, Z \in \text{GL}(n)$, we have

$$\text{MM} = (X \otimes Y \otimes Z) \text{MM} (Z^{-1} \otimes X^{-1} \otimes Y^{-1}), \quad (2)$$

where, if the notation isn’t already clear, it will become so in the next equation. To see that MM has this symmetry, note that

$$\text{tr } ABC = \text{tr}(Z^{-1}AX)(X^{-1}BY)(Y^{-1}CZ).$$

The fact that MM is the only such operator up to a constant comes from a simple representation-theoretic argument, which generalizes the fact that the only matrices which are invariant under conjugation are scalar multiples of the identity.

This suggests that a good way to search for matrix multiplication algorithms is to start with sums of separable tensors where the sum has some symmetry built in from the beginning. As we will see, one useful kind of symmetry is the following. We say that a set of n -dimensional vectors S is a *unitary 2-design* if it has the following two properties:

$$\sum_{v \in S} v = 0 \quad \text{and} \quad \frac{1}{|S|} \sum_{v \in S} |v\rangle\langle v| = \frac{1}{n} \mathbf{1}, \quad (3)$$

where $\mathbf{1}$ denotes the identity matrix. Here we use the Dirac notation $|u\rangle\langle v|$ for the outer product of u and v , i.e., the matrix whose i, j entry is $u_i v_j^*$ where $*$ denotes the complex conjugate.

The following theorem shows how 2-designs can be used to construct matrix multiplication algorithms.

Theorem 2.1. *Let $S \subset \mathbb{C}^n$ be a unitary 2-design, and let $s = |S|$. Then the tensor rank of MM_n is at most $s(s-1)(s-2) + 1$.*

Proof. Let $S = \{w_1, \dots, w_s\}$. We will show that the following is a decomposition of MM_n :

$$MM_n = \mathbf{1}^{\otimes 3} + \frac{n^3}{s^3} \sum_{i, j, k \text{ distinct}} |w_i\rangle\langle w_j - w_i| \otimes |w_j\rangle\langle w_k - w_j| \otimes |w_k\rangle\langle w_i - w_k|. \quad (4)$$

Since there are $s(s-1)(s-2)$ distinct ordered triples $i, j, k \in \{1, \dots, s\}$, this decomposition has $s(s-1)(s-2) + 1$ terms.

To prove (4), we use the fact (1) that MM can be written as a kind of twisted tensor product of identity matrices or Kronecker deltas. By the second property in the definition (3) of a 2-design, we have

$$MM_n = \frac{n^3}{s^3} \sum_{i, j, k} |w_i\rangle\langle w_j| \otimes |w_j\rangle\langle w_k| \otimes |w_k\rangle\langle w_i| \quad (5)$$

At the same time, the un-twisted version of this identity is

$$\mathbf{1}^{\otimes 3} = \frac{n^3}{s^3} \sum_{i, j, k} |w_i\rangle\langle w_i| \otimes |w_j\rangle\langle w_j| \otimes |w_k\rangle\langle w_k|. \quad (6)$$

Now we expand (4). We can sum over all s^3 triples i, j, k , since if any of these are equal the

summand is zero. Then

$$\begin{aligned}
& \sum_{i,j,k} |w_i\rangle\langle w_j - w_i| \otimes |w_j\rangle\langle w_k - w_j| \otimes |w_k\rangle\langle w_i - w_k| \\
&= \sum_{i,j,k} |w_i\rangle\langle w_j| \otimes |w_j\rangle\langle w_k| \otimes |w_k\rangle\langle w_i| \\
&- \sum_{i,j,k} \left[|w_i\rangle\langle w_i| \otimes |w_j\rangle\langle w_k| \otimes |w_k\rangle\langle w_i| + |w_i\rangle\langle w_j| \otimes |w_j\rangle\langle w_j| \otimes |w_k\rangle\langle w_i| + |w_i\rangle\langle w_j| \otimes |w_j\rangle\langle w_k| \otimes |w_k\rangle\langle w_k| \right]
\end{aligned} \tag{7}$$

$$\begin{aligned}
&+ \sum_{i,j,k} \left[|w_i\rangle\langle w_j| \otimes |w_j\rangle\langle w_j| \otimes |w_k\rangle\langle w_k| + |w_i\rangle\langle w_i| \otimes |w_j\rangle\langle w_k| \otimes |w_k\rangle\langle w_k| + |w_i\rangle\langle w_i| \otimes |w_j\rangle\langle w_j| \otimes |w_k\rangle\langle w_i| \right]
\end{aligned} \tag{8}$$

$$- \sum_{i,j,k} |w_i\rangle\langle w_i| \otimes |w_j\rangle\langle w_j| \otimes |w_k\rangle\langle w_k|$$

The mixed terms in lines (7) and (8) disappear because each product has an index that appears only once, and the first property in the definition (3) of a 2-design implies that summing over that index gives zero. Combining this with (5) and (6) leaves us with

$$\frac{n^3}{s^3} \sum_{i,j,k} |w_i\rangle\langle w_j - w_i| \otimes |w_j\rangle\langle w_k - w_j| \otimes |w_k\rangle\langle w_i - w_k| = \text{MM}_n - \mathbf{1}^{\otimes 3},$$

which completes the proof. □

Now, in $n = 2$ dimensions the three corners of an equilateral triangle form a 2-design:

$$S = \{(1, 0), (-1/2, \sqrt{3}/2), (-1/2, -\sqrt{3}/2)\}. \tag{9}$$

The outer products of these vectors with themselves are

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 1/4 & -\sqrt{3}/4 \\ -\sqrt{3}/4 & 3/4 \end{pmatrix}, \begin{pmatrix} 1/4 & \sqrt{3}/4 \\ \sqrt{3}/4 & 3/4 \end{pmatrix},$$

and the average of these is $\mathbf{1}/2$. This design (9) has size $s = 3$, in which case Theorem 2.1 shows that MM_2 has a tensor rank of at most 7.

The reader might object that we haven't really re-derived Strassen's algorithm since our algorithm doesn't seem to yield the same equations as Strassen's. However, de Groote [dG78] has shown that all 7-term decompositions of MM_2 are equivalent up to a change of basis, i.e., an instance of the $\text{GL}(n)^3$ action (2). Thus the algorithm based on the triangular design (9) is in fact isomorphic to Strassen's algorithm, and in any case, it gives a conceptual explanation for the fact that MM_2 has tensor rank 7. (For the reader wondering about algorithms for matrix multiplication over rings other than \mathbb{C} , see Section 4.)

3 Generalizations to larger n from group orbits

The triangular design (9) has a pleasing symmetry. In this section we show how to find similar designs in higher dimensions as the orbits of group actions. We assume basic familiarity with finite groups, for which we refer the reader to any standard textbook such as [Art91]. We need a few

facts from representation theory, which we spell out for completeness in the hopes of making the paper self-contained for a larger audience. Everything we do will be over the complex numbers \mathbb{C} , but generalizes to other fields, with some modifications.

A *representation* of a finite group G is a vector space V together with a group homomorphism $\rho: G \rightarrow \text{GL}(V)$, where $\text{GL}(V)$ denotes the general linear group of V , namely, the group of all invertible linear transformations from V to itself. By choosing a basis for V , we identify $V \cong \mathbb{C}^{\dim V}$, and each $\rho(g)$ becomes a $\dim V \times \dim V$ matrix such that $\rho(g)\rho(h) = \rho(gh)$ for all $g, h \in G$.

When the homomorphism ρ is understood from context, we refer to V as a representation of G . In this case, for $g \in G$ and $v \in V$ we write gv instead of $\rho(g)(v)$. The trivial representation is the identity map on V , where $gv = v$ for all $g \in G$.

A representation V of G is called *unitary* if each $\rho(g)$ is a unitary matrix. Any representation of a finite group over \mathbb{C} is equivalent, up to change of basis, to a unitary representation. In this basis we define the inner product $\langle u|v \rangle = \sum_i u_i^* v_i$ and the norm $|v|^2 = \langle v|v \rangle = \sum_i |v_i|^2$ as usual. Note that $\langle u|v \rangle = \langle gu|gv \rangle$ and $|gv|^2 = |v|^2$.

Given two representations $\rho: G \rightarrow \text{GL}(V)$ and $\rho': G \rightarrow \text{GL}(V')$ of the same group G , their *direct sum* is a representation $(\rho \oplus \rho'): G \rightarrow \text{GL}(V \oplus V')$ given by the matrices

$$(\rho \oplus \rho')(g) = \begin{pmatrix} \rho(g) & 0 \\ 0 & \rho'(g) \end{pmatrix}.$$

A representation $\rho: G \rightarrow \text{GL}(V)$ is *irreducible* if the only subspaces $W \subseteq V$ that are sent to themselves by every $g \in G$, i.e., such that $\rho(g)(w) \in W$ for all $g \in G$ and $w \in W$, are the trivial subspaces $W = 0$ or $W = V$. In fields of characteristic zero such as \mathbb{C} or \mathbb{R} , a representation is a direct sum if and only if it is not irreducible.

Given two representations V, W of G , a *homomorphism* of representations is a linear map $\varphi: V \rightarrow W$ that commutes with the action of G , in the sense that $\varphi(gv) = g\varphi(v)$ for all $g \in G, v \in V$.

Lemma (Schur's Lemma). *If V and W are two irreducible representations of a group G , then every nonzero homomorphism $V \rightarrow W$ is invertible. In particular, over an algebraically closed field, every homomorphism $V \rightarrow V$ is a scalar multiple of the identity $\mathbf{1}_V$.*

Schur's Lemma implies that the orbit of any unit-length vector in an irreducible representation is a 2-design in the sense defined above. We include a proof of this classical fact for completeness.

Corollary 3.1. *If V is a nontrivial irreducible representation of G , and $v \in V$ with $|v|^2 = 1$, then the orbit $\{gv \mid g \in G\}$ is a 2-design.*

Proof. First, the vector $\sum_{g \in G} gv$ always spans a 1-dimensional trivial sub-representation $W \subseteq V$, since $h \cdot (\sum gv) = \sum_{g \in G} (hg)v = \sum_{g \in G} gv$. If V is irreducible, then either $W = 0$ or $W = V$, but if V is nontrivial we cannot have $W = V$. Thus $\sum_g gv = 0$.

Second, let $\varphi = \sum_{g \in G} |gv\rangle\langle gv|$. Then for any $h \in G$ we have

$$h\varphi(w) = h \left(\sum_g |gv\rangle\langle gv| \right) w = \sum_g |hgv\rangle\langle gv|w\rangle = \sum_g |gv\rangle\langle h^{-1}gv|w\rangle = \sum_g |gv\rangle\langle gv|hw\rangle = \varphi(hw).$$

Thus φ is a homomorphism of representations, and by Schur's Lemma φ is a multiple of $\mathbf{1}_V$. We obtain the scaling factor by taking traces:

$$\frac{1}{|G|} \text{tr } \varphi = \frac{1}{|G|} \sum_g |gv|^2 = |v|^2 = 1 = \frac{1}{\dim V} \text{tr } \mathbf{1}_V,$$

so $\varphi/|G| = \mathbf{1}_V/\dim V$. □

We can now combine Corollary 3.1 with Theorem 2.1 to produce matrix multiplication algorithms in all dimensions, by considering families of finite groups and their irreducible representations. In particular, we have the following.

Corollary 3.2. *For every $n \geq 1$, the tensor rank of MM_n is at most $n^3 - n + 1$.*

Proof. Let $G = S_{n+1}$ be the symmetric group acting by permuting the coordinates on $V = \mathbb{R}^{n+1}$. As a representation of G , this splits into a direct sum of the trivial representation (spanned by the all-ones vector) and the so-called “standard representation” V of S_{n+1} , of dimension n (consisting of the vectors whose coordinates sum to zero). Let $w \in V \subset \mathbb{R}^{n+1}$ be the unit vector

$$w = \frac{1}{\sqrt{n(n+1)}} (n, -1, \dots, -1)$$

The orbit of w has size $s = n + 1$, and consists of unit vectors pointing to the corners of a simplex. Now apply Theorem 2.1. □

4 Future directions

Highly symmetric algorithms? Since any design must span \mathbb{C}^n , it has size $s \geq n$. Indeed, since its elements sum to zero, they are linearly dependent, so $s \geq n + 1$. Thus the simplex designs of Corollary 3.2 are optimal in this context, and applying Theorem 2.1 to larger n cannot directly improve the matrix multiplication exponent. This leaves open the question of whether there are other families of highly symmetric algorithms. See [GM16, CILO16] for work in this direction, as well as [Bur14, Bur15, LR16, LM16, IL17].

Using t -designs for $t > 2$? The key fact we used was that any orbit in an irreducible representation of a finite group is a unitary 2-design. Similarly, a *unitary t -design* in a vector space V is a set of vectors $S \subseteq V$ such that, for every polynomial f on V of degree at most t , the average of f over S is the same as the average of f over the unit sphere in V . (Over the reals, these are traditionally called “spherical t -designs,” but we are working in complex vector spaces.) Another open question is then

Question 4.1. *Can t -designs for $t > 2$ help us construct efficient matrix multiplication algorithms?*

For example, one might hope for a similar construction to Theorem 2.1, in which one could leverage the t -design property to get even more terms to cancel.

Working over arbitrary rings? One fact which is obvious from Strassen’s original construction, but not from ours, is that the rank of MM_2 is at most 7 over any ring. Strassen’s construction works in all rings since it only uses coefficients in $\{0, \pm 1\}$; ours uses coefficients in $\mathbb{Z}[\sqrt{3}, 1/2, 1/3]$, so it works in any ring where the elements $\sqrt{3}, 1/2, 1/3$ exist. Note that $1/2$ and $1/3$ exist in any ring R of characteristic m coprime to 6, since any such ring contains $\mathbb{Z}/m\mathbb{Z}$ as a subring, in which 2 and 3 are units. If $\sqrt{3}$ doesn’t exist in R , we can formally adjoin it by considering $R' = R[x]/(x^2 - 3)$; by a standard trick (e.g. [BCS97, Section 15.3]), this implies the same exponent over R itself (although it may not actually yield an algorithm for MM_2 over R).

Question 4.2. *Is there a similarly transparent and conceptual proof of Strassen’s result that works over arbitrary rings?*

Acknowledgments

Parts of this project were inspired in 2015 by discussions with Jonah Blasiak, Thomas Church, Henry Cohn, and Chris Umans, via a collaboration funded by the AIM SQuaRE program, with an additional visit hosted by the Santa Fe Institute. J.A.G. was funded by an Omidyar Fellowship from the Santa Fe Institute during this work and by NSF grant DMS-1620484, and C.M. was funded partly by the John Templeton Foundation. C.M. also thanks École Normale Supérieure for providing a visiting position during which some of this work was carried out.

References

- [Ale97] V. B. Alekseyev. Maximal extensions with simple multiplication for the algebra of matrices of the second order. *Disc. Math. Appl.*, 7:89–101, 1997. doi:10.1515/dma.1997.7.1.89.
- [Art91] Michael Artin. *Algebra*. Prentice Hall, Inc., Englewood Cliffs, NJ, 1991.
- [BCS97] Peter Bürgisser, Michael Clausen, and M. Amin Shokrollahi. *Algebraic complexity theory*, volume 315 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1997. With the collaboration of Thomas Lickteig. doi:10.1007/978-3-662-03338-8.
- [BI11] Peter Bürgisser and Christian Ikenmeyer. Geometric complexity theory and tensor rank. In *STOC '11: 43rd Annual ACM Symposium on Theory of Computing*, pages 509–518. ACM, New York, 2011. Preprint of the full version available as arXiv:1011.1350 [cs.CC]. doi:10.1145/1993636.1993704.
- [Blä13] Markus Bläser. *Fast Matrix Multiplication*. Number 5 in Graduate Surveys. Theory of Computing Library, 2013. doi:10.4086/toc.gs.2013.005.
- [Bur14] Vladimir P. Burichenko. On symmetries of the Strassen algorithm. arXiv:1408.6273 [cs.CC], 2014.
- [Bur15] Vladimir P. Burichenko. Symmetries of matrix multiplication algorithms. I. arXiv:1508.01110 [cs.CC], 2015.
- [Cha86] Philippe Chatelin. On transformations of algorithms to multiply 2×2 matrices. *Inform. Process. Lett.*, 22(1):1–5, 1986. doi:10.1016/0020-0190(86)90033-5.
- [CILO16] Luca Chiantini, Christian Ikenmeyer, J.M. Landsberg, and Giorgio Ottaviani. The geometry of rank decompositions of matrix multiplication I: 2×2 matrices. arXiv:1610.08364 [cs.CC], 2016.
- [CKSU05] Henry Cohn, Robert Kleinberg, Balazs Szegedy, and Christopher Umans. Group-theoretic algorithms for matrix multiplication. In *FOCS '05: 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 379–388. IEEE Computer Society, 2005. Preprint available as arXiv:math.GR/0511460. doi:10.1109/SFCS.2005.39.
- [Cla88] M. Clausen. *Beiträge zum Entwurf schneller Spektraltransformationen*. Habilitation, Universität Karlsruhe, 1988.

- [CU03] Henry Cohn and Christopher Umans. A group-theoretic approach to fast matrix multiplication. In *FOCS '03: 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 438–449. IEEE Computer Society, 2003. Preprint available as arXiv:math.GR/0307321. doi:10.1109/SFCS.2003.1238217.
- [CU13] Henry Cohn and Christopher Umans. Fast matrix multiplication using coherent configurations. In *SODA '13: 24th ACM–SIAM Symposium on Discrete Algorithms*, pages 1074–1087, 2013. Preprint available as arXiv:1207.6528 [math.NA]. doi:10.1137/1.9781611973105.77.
- [dG78] Hans F. de Groote. On varieties of optimal algorithms for the computation of bilinear mappings. II. Optimal algorithms for 2×2 -matrix multiplication. *Theoret. Comput. Sci.*, 7(2):127–148, 1978. doi:10.1016/0304-3975(78)90045-2.
- [DS13] A. M. Davie and A. J. Stothers. Improved bound for the complexity of matrix multiplication. *Proc. Roy. Soc. Edinburgh Sect. A*, 143:351–369, 2013. doi:10.1017/S0308210511001648.
- [Gas71] N. Gastinel. Sur le calcul des produits de matrices. *Numer. Math.*, 17:222–229, 1971. doi:10.1007/BF01436378.
- [GK00] Ann Q. Gates and Vladik Kreinovich. Strassen’s algorithm made (somewhat) more natural: A pedagogical remark. Technical Report 502, University of Texas at El Paso, Department of Computer Science, 2000. URL: http://digitalcommons.utep.edu/cs_techrep/502.
- [GM16] Joshua A. Grochow and Cristopher Moore. Matrix multiplication algorithms from group orbits. arXiv:1612.01527 [cs.CC], 2016.
- [IL17] Christian Ikenmeyer and Vladimir Lysikov. Strassen’s 2×2 matrix multiplication algorithm: a conceptual perspective. arXiv:1708.08083v1 [cs.DS], 2017.
- [Lan08] J. M. Landsberg. Geometry and the complexity of matrix multiplication. *Bull. Amer. Math. Soc. (N.S.)*, 45(2):247–284, 2008. doi:10.1090/S0273-0979-08-01176-2.
- [Lan14] J. M. Landsberg. New lower bounds for the rank of matrix multiplication. *SIAM J. Comput.*, 43:144–149, 2014. doi:10.1137/120880276.
- [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. ACM. Full version available as arXiv:1401.7714 [cs.DS]. doi:10.1145/2608628.2608664.
- [LM16] J. M. Landsberg and Mateusz Michałek. On the geometry of border rank algorithms for matrix multiplication and other tensors with symmetry. arXiv:1601.08229 [math.AG], 2016.
- [LR13] Richard J. Lipton and Kenneth W. Regan. Volker Strassen: Amazing results. In *People, Problems, and Proofs: Essays from Gödel’s Lost Letter: 2010*, pages 75–78. Springer Berlin Heidelberg, 2013. Based on blog post available at <https://rjlipton.wordpress.com/2010/03/27/fast-matrix-products-and-other-amazing-results/>. doi:10.1007/978-3-642-41422-0_13.

- [LR16] J. M. Landsberg and Nicholas Ryder. On the geometry of border rank algorithms for $n \times 2$ by 2×2 matrix multiplication. *Exper. Math.*, 2016. In press. Preprint available as arXiv:1509.08323 [cs.NA]. doi:10.1080/10586458.2016.1162230.
- [Min15] Jacob Minz. Derivation of Strassen’s algorithm for the multiplication of 2×2 matrices. Blog post at <http://jacobminz.blogspot.in/2015/05/derivation-of-strassens-algorithm-for.html>, 2015.
- [Pat09] Mike Paterson. Strassen symmetries. Presentation at Leslie Valiant’s 60th birthday celebration, May 2009. URL: <http://www.cis.upenn.edu/~mkearns/valiant/paterson.ppt>.
- [San09] Piotr Sankowski. Maximum weight bipartite matching in matrix multiplication time. *Theoret. Comput. Sci.*, 410(44):4480–4488, 2009. doi:10.1016/j.tcs.2009.07.028.
- [Str69] Volker Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969. doi:10.1007/BF02165411.
- [Wil12] Virginia Vassilevska Williams. Multiplying matrices faster than Coppersmith–Winograd. In *STOC ’12: 44th Annual ACM Symposium on Theory of Computing*, pages 887–898, New York, NY, USA, 2012. ACM. doi:10.1145/2213977.2214056.
- [Yuv78] Gideon Yuval. A simple proof of Strassen’s result. *Inform. Process. Lett.*, 7(6):285–286, 1978. doi:10.1016/0020-0190(78)90018-2.
- [Zwi02] Uri Zwick. All pairs shortest paths using bridging sets and rectangular matrix multiplication. *J. ACM*, 49(3):289–317 (electronic), 2002. doi:10.1145/567112.567114.