Flip Graphs for Matrix Multiplication

Manuel Kauers*
Jakob Moosbauer†
manuel.kauers@jku.at
jakob.moosbauer@jku.at
Institute for Algebra
Johannes Kepler University, A4040 Linz, Austria

ABSTRACT

We introduce a new method for discovering matrix multiplication schemes based on random walks in a certain graph, which we call the flip graph. Using this method, we were able to reduce the number of multiplications for the matrix formats $(4, 4, 5)$ and $(5, 5, 5)$, both in characteristic two and for arbitrary ground fields.

CCS CONCEPTS

• **Computing methodologies → Algebraic algorithms**.

KEYWORDS

Bilinear complexity; Strassen's algorithm; Tensor rank

ACM Reference Format:
Manuel Kauers* and Jakob Moosbauer†. 2023. Flip Graphs for Matrix Multiplication. In *International Symposium on Symbolic and Algebraic Computation 2023 (ISSAC 2023), July 24–27, 2023, Tromsø, Norway.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3597066.3597120

1 INTRODUCTION

Nobody knows the computational cost of computing the product of two matrices. Strassen's discovery [18] that two 2×2-matrices can be multiplied with only 7 multiplications in the ground field launched intensive research on the complexity of matrix multiplication during the past decades. One branch of this research aims at finding upper (or possibly also lower) bounds on the matrix multiplication exponent ω. The current world record $\omega < 2.37188$ is held by Duan, Wu and Zhou [7] and only slightly improves the previous record $\omega < 2.37286$ by Alman and Williams [1]. These results concern asymptotically large matrix sizes.

Another branch of research on matrix multiplication algorithms concerns specific small matrix sizes. For 2×2 matrices, it is known that there is no way to do the job with only 6 multiplications [20], and that Strassen's algorithm is essentially the only way to do it with 7 [6]. Also for multiplying a 2×2 matrix with a $2 \times p$ matrix and for multiplying a 2×3 matrix with a 3×3 matrix, optimal algorithms are known [11]. For all other formats, the known upper and lower bounds do not match. For example, for the case 3×3 times 3×3, the best known upper bound is 23 [14] and the best known lower bound is 19 [2] unless we impose restrictions on the ground domain such as commutativity.

An upper bound for a specific matrix format can be obtained by stating an explicit matrix multiplication scheme with as few multiplications as possible. Such schemes can be discovered by various techniques, including hand calculation [14, 18], numerical methods [16, 17], SAT solving [5, 9, 10], or machine learning [8]. The latter approach, due to Fawzi et al., has received a lot of attention, even in the general public, because it led to an unexpected improvement of the upper bound for multiplying two 4×4 matrices from 49 to 47 multiplications in characteristic two. They also reduced the bound for multiplying two 5×5 matrices from 98 to 96 in characteristic two, and found improvements for some formats involving rectangular matrices.

In our quick response [12] to the paper of Fawzi et al., we announced that we can find further schemes for 4×4 matrices using 47 multiplications in characteristic two, and that we can reduce the number of multiplications required for 5×5 matrices in characteristic two to 95. In the present paper, we introduce the method by which we found these schemes.

We define a graph whose vertices are correct matrix multiplication schemes and where there is an edge from one scheme to another if the second can be obtained from the first by some kind of transformation. We consider two transformations. One is called a flip and turns a given scheme to a different one with the same number of multiplications, and the other is called a reduction and turns a given scheme to one with a smaller number of multiplications. The precise construction of this flip graph is given in Sect. 3. In Sect. 4, we illustrate (parts of) the flip graph for 2×2 and 3×3 matrices.

In order to find better upper bounds for a specific format, we start from a known scheme, e.g., the standard algorithm, and perform a random walk in the flip graph. Although reduction edges are much more rare than flip edges, it turned out that there are enough of them to reach interesting schemes with a reasonable amount of computation time. In particular, we were able to match the best known algorithms for all multiplication formats $n \times m$ times $m \times p$ with $n, m, p \leq 5$, and found better bounds in four cases. These results are reported in Sect. 5.

2 MATRIX MULTIPLICATION

Let K be a field, let R be a K-Algebra and let $\mathbf{A} \in R^{n \times m}$, $\mathbf{B} \in R^{m \times p}$. Recall that the computation of the matrix product $\mathbf{C} = \mathbf{AB}$ by a

* M. Kauers was supported by the Austrian FWF grants P31571-N32 and I6130-N.
† J. Moosbauer was supported by the Land Oberösterreich through the LIT-AI Lab.

ISSAC 2023, July 24–27, 2023, Tromsø, Norway
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0039-2/23/07…$15.00
https://doi.org/10.1145/3597066.3597120

Strassen-like algorithm proceeds in two stages. In the first stage we compute certain products $m_1, \dots, m_r$ of linear combinations of entries of $\mathbf{A}$ and linear combinations of entries of $\mathbf{B}$. In the second stage the entries of $\mathbf{C}$ are obtained as linear combinations of the m_i. An algorithm of this form is called a bilinear algorithm [4].

For example, in the case $n = m = p = 2$, if we let

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{pmatrix} \quad \text{and} \quad \mathbf{C} = \begin{pmatrix} c_{1,1} & c_{1,2} \\ c_{2,1} & c_{2,2} \end{pmatrix},$$

then Strassen's algorithm computes $\mathbf{C}$ in the following way:

$$\begin{array}{l|l} m_1 = (a_{1,1} + a_{2,2})(b_{1,1} + b_{2,2}) & c_{1,1} = m_1 + m_4 - m_5 + m_7 \\ m_2 = (a_{2,1} + a_{2,2})(b_{1,1}) & c_{1,2} = m_3 + m_5 \\ m_3 = (a_{1,1})(b_{1,2} - b_{2,2}) & c_{2,1} = m_2 + m_4 \\ m_4 = (a_{2,2})(b_{2,1} - b_{1,1}) & c_{2,2} = m_1 - m_2 + m_3 + m_6. \\ m_5 = (a_{1,1} + a_{1,2})(b_{2,2}) & \\ m_6 = (a_{2,1} - a_{1,1})(b_{1,1} + b_{1,2}) & \\ m_7 = (a_{1,2} - a_{2,2})(b_{2,1} + b_{2,2}) & \end{array}$$

It is common and convenient to express matrix multiplication in the language of tensors. For 2×2 matrices, the matrix multiplication tensor is

$$\mathcal{M}_{2,2,2} = \sum_{i,j,k=1}^{2} E_{i,j} \otimes E_{j,k} \otimes E_{k,i} \in K^{2\times2} \otimes K^{2\times2} \otimes K^{2\times2},$$

where $E_{u,v}$ refers to the matrix having a 1 at position (u, v) and zeros elsewhere. Strassen's algorithm is based on the observation that this tensor can also be written as the sum of only seven tensors of the form $A \otimes B \otimes \Gamma$. Indeed, we have

$$\begin{aligned} \mathcal{M}_{2,2,2} = {} & (a_{1,1} + a_{2,2}) \otimes (b_{1,1} + b_{2,2}) \otimes (c_{1,1} + c_{2,2}) \\ & + (a_{2,1} + a_{2,2}) \otimes (b_{1,1}) \otimes (c_{1,2} - c_{2,2}) \\ & + (a_{1,1}) \otimes (b_{1,2} - b_{2,2}) \otimes (c_{2,1} + c_{2,2}) \\ & + (a_{2,2}) \otimes (b_{2,1} - b_{1,1}) \otimes (c_{1,1} + c_{1,2}) \\ & + (a_{1,1} + a_{1,2}) \otimes (b_{2,2}) \otimes (c_{2,1} - c_{1,1}) \\ & + (a_{2,1} - a_{1,1}) \otimes (b_{1,1} + b_{1,2}) \otimes (c_{2,2}) \\ & + (a_{1,2} - a_{2,2}) \otimes (b_{2,1} + b_{2,2}) \otimes (c_{1,1}), \end{aligned}$$

where for better readability we write $a_{i,j}, b_{i,j}, c_{i,j}$ instead of $E_{i,j}$. Readers not comfortable with tensor products are welcome to understand $a_{i,j}, b_{i,j}, c_{i,j}$ as polynomial variables and $\otimes$ as multiplication. Note that each row in the expression above corresponds to one of the m_i's from before. The first two factors encode their definitions and the third how they enter into the $c_{j,i}$. Note also that the indices in the third factor are swapped in order to be consistent with the matrix multiplication literature, where this version is preferred because it makes certain equations more symmetric.

The general definition is as follows.

Definition 1. *Let $n, m, p \in \mathbb{N}$. The matrix multiplication tensor of the format (n, m, p) is defined as*

$$\mathcal{M}_{n,m,p} = \sum_{i,j,k=1}^{n,m,p} a_{i,j} \otimes b_{j,k} \otimes c_{k,i} \in K^{n\times m} \otimes K^{m\times p} \otimes K^{p\times n},$$

where $a_{u,v}, b_{u,v}, c_{u,v}$ refer to matrices of the respective format having a 1 at position (u, v) and zeros in all other positions.

Rank-one tensors are non-zero tensors that can be written as $A \otimes B \otimes \Gamma$ for certain matrices $A \in K^{n\times m}$, $B \in K^{m\times p}$, $\Gamma \in K^{p\times n}$. The rank of a tensor $\mathcal{T}$ is the smallest number r such that $\mathcal{T}$ can be written as a sum of r rank-one tensors.

An (n, m, p)-matrix multiplication scheme is a finite set S of rank-one tensors whose sum is $\mathcal{M}_{n,m,p}$. We call $|S|$ the rank of the scheme.

The sum of k linearly independent rank-one tensors need not necessarily have tensor rank k. Strassen's discovery amounts precisely to the observation that $\mathcal{M}_{2,2,2}$, which is defined as a sum of 8 linearly independent rank-one tensors, has only rank 7. A more simple example is the equation

$$\begin{aligned} & a_{1,1} \otimes b_{1,1} \otimes c_{1,1} + a_{1,1} \otimes b_{1,2} \otimes c_{3,1} + a_{1,1} \otimes (b_{1,1} + b_{1,2}) \otimes c_{2,2} \\ & = a_{1,1} \otimes b_{1,1} \otimes (c_{1,1} + c_{2,2}) + a_{1,1} \otimes b_{1,2} \otimes (c_{3,1} + c_{2,2}). \end{aligned}$$

If a matrix multiplication scheme contains the three rank-one tensors in the first line, we can replace them by the two rank-one tensors in the second line and thereby reduce the rank by one. This observation leads to the following generalization of Pan's concept of uniting terms [15].

Definition 2. *Let $n, m, p, r \in \mathbb{N}$ and let $S = \{A^{(i)} \otimes B^{(i)} \otimes \Gamma^{(i)} \mid i \in \{1, \dots, r\}\}$ be an (n, m, p)-matrix multiplication scheme. We call S* reducible *if there is a nonempty set $I \subseteq \{1, \dots, r\}$ such that*

(1) $\dim_K \langle A^{(i)} \rangle_{i\in I} = 1$ *and*
(2) $\{B^{(i)} \mid i \in I\}$ *is linearly dependent over K,*

or analogously with B, A or A, Γ or Γ, A or B, Γ or Γ, B in place of A, B.

Proposition 3. *Let $n, m, p, r \in \mathbb{N}$ and let S be a reducible (n, m, p)-matrix multiplication scheme of rank r. Then there exists an (n, m, p)-matrix multiplication scheme of rank $r - 1$.*

Proof. We prove the statement in the case that the conditions in Def. 2 hold for A and B. In the other cases the proof works analogously. Since $\{B^{(i)} \mid i \in I\}$ is linearly dependent, there is a $t \in I$ such that $B^{(t)} = \sum_{i\in I\setminus\{t\}} \beta_i B^{(i)}$ for some $\beta_i \in K$. Moreover, there are $\alpha_i \in K$ such that $A^{(t)} = \alpha_i A^{(i)}$. Hence, we have

$$\begin{aligned} & \sum_{i\in I} A^{(i)} \otimes B^{(i)} \otimes \Gamma^{(i)} \\ & = \sum_{i\in I\setminus\{t\}} A^{(i)} \otimes B^{(i)} \otimes \Gamma^{(i)} + A^{(t)} \otimes \sum_{i\in I\setminus\{t\}} \beta_i B^{(i)} \otimes \Gamma^{(t)} \\ & = \sum_{i\in I\setminus\{t\}} A^{(i)} \otimes B^{(i)} \otimes \Gamma^{(i)} + \sum_{i\in I\setminus\{t\}} A^{(i)} \otimes B^{(i)} \otimes \alpha_i\beta_i\Gamma^{(t)} \\ & = \sum_{i\in I\setminus\{t\}} A^{(i)} \otimes B^{(i)} \otimes (\Gamma^{(i)} + \alpha_i\beta_i\Gamma^{(t)}) \end{aligned}$$

Therefore,

$$\begin{aligned} S' = {} & \{A^{(i)} \otimes B^{(i)} \otimes \Gamma^{(i)} \mid i \in \{1, \dots, r\} \setminus I\} \\ & \cup \{A^{(i)} \otimes B^{(i)} \otimes (\Gamma^{(i)} + \alpha_i\beta_i\Gamma^{(t)}) \mid i \in I \setminus \{t\}\} \end{aligned}$$

is a multiplication scheme with rank $r - 1$. □

We call a scheme S' constructed as above a *reduction* of S.

Symmetries of the matrix multiplication tensor $\mathcal{M}_{n,m,p}$ give rise to various ways to transform a given matrix multiplication scheme into another one. For example, because of the identity

$(\mathbf{AB})^\top = \mathbf{B}^\top\mathbf{A}^\top$, exchanging each rank-one tensor $A \otimes B \otimes \Gamma$ by $B^\top \otimes A^\top \otimes \Gamma^\top$ maps a correct scheme to another correct scheme. A correct scheme is also obtained if we replace every rank-one tensor $A \otimes B \otimes \Gamma$ by $B \otimes \Gamma \otimes A$. Finally, if $U \in K^{m\times m}$ is invertible, then the identity $\mathbf{AB} = \mathbf{A}UU^{-1}\mathbf{B}$ implies that also replacing every rank-one tensor $A \otimes B \otimes \Gamma$ by $AU \otimes U^{-1}B \otimes \Gamma$ maps a correct scheme to another one. These transformations generate the symmetry group of $\mathcal{M}_{n,m,p}$. For more details on this group see [6, 13]. We can let the symmetry group act on the set of matrix multiplication schemes and we call two schemes equivalent if they belong to the same orbit.

Def. 2 is compatible with the action of the symmetry group. For permutations, this is because the definition explicitly allows any two factors to take the roles of A and B, and linear dependence is preserved under transposition. Likewise, if some matrices $B^{(i)}$ ($i \in I$) are linearly dependent, then so are the matrices $VB^{(i)}W^{-1}$ ($i \in I$) for any invertible matrices V, W. Therefore, we can extend Def. 2 from individual matrix multiplication schemes to equivalence classes.

3 THE FLIP GRAPH

As reducibility is preserved by the application of a symmetry, the only chance to get from an irreducible to a reducible scheme is by a transformation that, at least in general, does not preserve reducibility. Flips have this feature. The idea is to subtract something from one rank-one tensor and add it to one of the others. This can be done for any two rank-one tensors that share a common factor. For example, we have

$$\begin{aligned} &A \otimes B \otimes \Gamma \ + \ A \otimes B' \otimes \Gamma' \\ &= A \otimes (B + B') \otimes \Gamma \ + \ A \otimes B' \otimes (\Gamma' - \Gamma). \end{aligned}$$

In contrast to a symmetry transformation, this transformation only affects two elements of the multiplication scheme instead of all. Therefore, we can in general expect the resulting scheme to be non-equivalent to the original one.

Definition 4. *Let $n, m, p, r \in \mathbb{N}$ and let S, S' be (n, m, p)-matrix multiplication schemes of rank r. We call S' a* flip *of S if there are*

(1) $T_1 = A \otimes B \otimes \Gamma \in S$,
(2) $T_2 = A \otimes B' \otimes \Gamma' \in S$ *and*
(3) $T \in \{A \otimes B \otimes \Gamma', A \otimes B' \otimes \Gamma\}$

such that $(S \setminus \{T_1, T_2\}) \cup \{T_1 + T, T_2 - T\} = S'$.

The definition is meant to apply analogously for any permutation of A, B and Γ.

The union is always disjoint since we require S and S' to be of the same rank. Cases where S' would have a smaller rank than S need not be included because in these cases S' is a reduction of S. Note that flips are reversible because if $S' = (S \setminus \{T_1, T_2\}) \cup \{T_1 + T, T_2 - T\}$, then $S = (S' \setminus \{T_2 - T, T_1 + T\}) \cup \{(T_2 - T) + T, (T_1 + T) - T\}$.

Flips are well-defined for equivalence classes of matrix multiplication schemes in the following sense: If S' is a flip of S and g is an element of the symmetry group, then $g(S)$ is a flip of $g(S')$. This is the case since Def. 4 applies for arbitrary permutations of A, B, Γ and because the symmetry transformations act linearly on the individual matrices.

Def. 4 introduces flips as rather specific ways of replacing two rank-one tensors in a given scheme by some other rank-one tensors. However, it turns out that if we only want to replace two rank-one tensors at a time, then there is not much more we can do. Thm. 7 makes this more precise.

Lemma 5. *Let U, V and W be vector spaces. For $i = 1, \ldots, 4$, let $A^{(i)} \in U, B^{(i)} \in V, \Gamma^{(i)} \in W$ be such that*

$$\begin{aligned} &A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)} + A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)} \\ &= A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)} + A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)} \end{aligned} \tag{1}$$

and

$$\dim\langle A^{(1)}, A^{(2)}\rangle = \dim\langle B^{(1)}, B^{(2)}\rangle = \dim\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = 2.$$

Then

$$\begin{aligned} &\left\{A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)},\ A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)}\right\} \\ &= \left\{A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)},\ A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}\right\}. \end{aligned}$$

Proof. Let U', V' and W' be such that $U = \langle A^{(1)}, A^{(2)}\rangle \oplus U'$, $V = \langle B^{(1)}, B^{(2)}\rangle \oplus V'$ and $W = \langle \Gamma^{(1)}, \Gamma^{(2)}\rangle \oplus W'$. Since $A^{(1)}$ and $A^{(2)}$, $B^{(1)}$ and $B^{(2)}$, and $\Gamma^{(1)}$ and $\Gamma^{(2)}$ are linearly independent, there are

$$\begin{aligned} \alpha_1, \alpha_2, \alpha_1', \alpha_2', \beta_1, \beta_2, \beta_1', \beta_2', \gamma_1, \gamma_2, \gamma_1', \gamma_2' &\in K, \\ A, A' &\in U', \\ B, B' &\in V', \\ \Gamma, \Gamma' &\in W' \end{aligned}$$

such that

$$\begin{aligned} A^{(3)} &= \alpha_1 A^{(1)} + \alpha_2 A^{(2)} + A, & A^{(4)} &= \alpha_1' A^{(1)} + \alpha_2' A^{(2)} + A', \\ B^{(3)} &= \beta_1 B^{(1)} + \beta_2 B^{(2)} + B, & B^{(4)} &= \beta_1' B^{(1)} + \beta_2' B^{(2)} + B', \\ \Gamma^{(3)} &= \gamma_1 \Gamma^{(1)} + \gamma_2 \Gamma^{(2)} + \Gamma, & \Gamma^{(4)} &= \gamma_1' \Gamma^{(1)} + \gamma_2' \Gamma^{(2)} + \Gamma'. \end{aligned}$$

After plugging these in (1) and equating coefficients we obtain the following system of equations:

$$\begin{aligned} \alpha_1\beta_1\gamma_1 + \alpha_1'\beta_1'\gamma_1' &= 1, & \alpha_1\beta_1\gamma_2 + \alpha_1'\beta_1'\gamma_2' &= 0, \\ \alpha_1\beta_2\gamma_1 + \alpha_1'\beta_2'\gamma_1' &= 0, & \alpha_1\beta_2\gamma_2 + \alpha_1'\beta_2'\gamma_2' &= 0, \\ \alpha_2\beta_1\gamma_1 + \alpha_2'\beta_1'\gamma_1' &= 0, & \alpha_2\beta_1\gamma_2 + \alpha_2'\beta_1'\gamma_2' &= 0, \\ \alpha_2\beta_2\gamma_1 + \alpha_2'\beta_2'\gamma_1' &= 0, & \alpha_2\beta_2\gamma_2 + \alpha_2'\beta_2'\gamma_2' &= 1. \end{aligned}$$

A straightforward computation confirms that these equations imply that either $\alpha_1\beta_1\gamma_1 = \alpha_2'\beta_2'\gamma_2' = 1$ and all other variables are 0 or $\alpha_2\beta_2\gamma_2 = \alpha_1'\beta_1'\gamma_1' = 1$ and all other variables are 0. It follows further that A, A', B, B', C, C' need to be zero, which completes the proof. □

Lemma 6. *Let U, V be vector spaces. Let $B^{(1)}, B^{(2)}, B^{(3)}, B^{(4)} \in U$ and $\Gamma^{(1)}, \Gamma^{(2)}, \Gamma^{(3)}, \Gamma^{(4)} \in V$ be such that*

$$B^{(1)} \otimes \Gamma^{(1)} + B^{(2)} \otimes \Gamma^{(2)} = B^{(3)} \otimes \Gamma^{(3)} + B^{(4)} \otimes \Gamma^{(4)},$$

$\dim\langle B^{(1)}, B^{(2)}\rangle = \dim\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = 2$, *and* $\Gamma^{(3)}, \Gamma^{(4)} \neq 0$.

Then $\langle B^{(1)}, B^{(2)}\rangle = \langle B^{(3)}, B^{(4)}\rangle$ and $\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = \langle \Gamma^{(3)}, \Gamma^{(4)}\rangle$.

Proof. Since $\Gamma^{(1)}, \Gamma^{(2)}, \Gamma^{(3)}, \Gamma^{(4)} \neq 0$ the equality implies that $B^{(1)}, B^{(2)}, B^{(3)}, B^{(4)}$ are linearly dependent. Therefore, there exist $\mu, \nu, \lambda \in K$ such that $B^{(1)} = \mu B^{(2)} + \nu B^{(3)} + \lambda B^{(4)}$. It follows

$$B^{(2)} \otimes (\Gamma^{(2)} + \mu\Gamma^{(1)}) = B^{(3)} \otimes (\Gamma^{(3)} - \nu\Gamma^{(1)}) + B^{(4)} \otimes (\Gamma^{(4)} - \lambda\Gamma^{(1)}).$$

Since $\Gamma^{(1)}$ and $\Gamma^{(2)}$ are linearly independent this implies $B^{(2)} \in \langle B^{(3)}, B^{(4)}\rangle$. For the same reason we have $B^{(1)} \in \langle B^{(3)}, B^{(4)}\rangle$ and since $B^{(1)}$ and $B^{(2)}$ are linearly independent it follows $\langle B^{(1)}, B^{(2)}\rangle = \langle B^{(3)}, B^{(4)}\rangle$. The equality $\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = \langle \Gamma^{(3)}, \Gamma^{(4)}\rangle$ is shown by the same argument. □

Theorem 7. *Let U, V and W be vector spaces. For $i = 1, \dots, 4$, let $A^{(i)} \in U \setminus \{0\}, B^{(i)} \in V \setminus \{0\}, \Gamma^{(i)} \in W \setminus \{0\}$ be such that*

$$\begin{aligned} &A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)} + A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)} \\ &= A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)} + A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}, \end{aligned} \tag{2}$$

$\dim\langle B^{(1)}, B^{(2)}\rangle = \dim\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = 2$, *and*

$$\begin{aligned} &\{A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)},\ A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)}\} \\ &\neq \{A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)},\ A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}\}. \end{aligned}$$

Then the following hold:

(1) $\dim\langle A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}\rangle = 1$
(2) $\langle B^{(1)}, B^{(2)}\rangle = \langle B^{(3)}, B^{(4)}\rangle$
(3) $\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = \langle \Gamma^{(3)}, \Gamma^{(4)}\rangle$

Proof. We start by showing the first claim. If $A^{(1)}$ and $A^{(2)}$ are linearly independent, then it follows from Lemma 5 that

$$\begin{aligned} &\{A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)},\ A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)}\} \\ &= \{A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)},\ A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}\}. \end{aligned}$$

Thus, $A^{(1)}$ and $A^{(2)}$ must be linearly dependent. So there exists $\alpha \in K$ such that $A^{(2)} = \alpha A^{(1)}$. It follows

$$\begin{aligned} &A^{(1)} \otimes (B^{(1)} \otimes \Gamma^{(1)} + \alpha(B^{(2)} \otimes \Gamma^{(2)})) \\ &= A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)} + A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}. \end{aligned}$$

This can only be the case if $A^{(3)}$ and $A^{(4)}$ or $B^{(3)} \otimes \Gamma^{(3)}$ and $B^{(4)} \otimes \Gamma^{(4)}$ are linearly dependent. If $B^{(3)} \otimes \Gamma^{(3)}$ and $B^{(4)} \otimes \Gamma^{(4)}$ are linearly dependent, then the right side of (2) has rank one, so the left side also has rank one. This would imply that either $\dim\langle B^{(1)}, B^{(2)}\rangle = 1$ or $\dim\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = 1$. Therefore $A^{(3)}$ and $A^{(4)}$ are linearly dependent and so $\dim\langle A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}\rangle = 1$.

Using that the $A^{(i)}$ are constant multiples of each other and that $(\lambda A) \otimes B \otimes \Gamma = A \otimes (\lambda B) \otimes \Gamma$ for all $\lambda \in K$, we may as well assume $A^{(1)} = A^{(2)} = A^{(3)} = A^{(4)}$. The equality (2) then reduces to

$$B^{(1)} \otimes \Gamma^{(1)} + B^{(2)} \otimes \Gamma^{(2)} = B^{(3)} \otimes \Gamma^{(3)} + B^{(4)} \otimes \Gamma^{(4)}.$$

The remaining claims follow directly from Lemma 6. □

Informally, Thm. 7 says that if we want to replace two rank-one tensors nontrivially by two others, then they must agree in one of the factors and this factor cannot change. Additionally, the vector spaces generated by the other factors must stay the same.

We now introduce the main concept of this paper.

Definition 8. *Let $n, m, p \in \mathbb{N}$ and let V be the set of all orbits of (n, m, p)-matrix multiplication schemes under the symmetry group and define*

$$\begin{aligned} E_1 &= \{(S, S') \mid S' \text{ is a flip of } S\}, \\ E_2 &= \{(S, S') \mid S' \text{ is a reduction of } S\}. \end{aligned}$$

(1) *The graph $G = (V, E_1 \cup E_2)$ is called the (n, m, p)-*flip graph*. The edges in E_1 are called* flips *and the edges in E_2 are called* reductions.
(2) *For a given $r \in \mathbb{N}$, the subgraph of G consisting of all vertices of rank at most r is called the (m, n, p)-*flip graph *of rank at most r.*
(3) *For a given $r \in \mathbb{N}$, the set $\{S \in V : \operatorname{rank}(S) = r\}$ is called the rth* level *of G.*

Flips always connect vertices belonging to the same level and a reduction always leads to a vertex belonging to a lower level. Also keep in mind that if there is a flip from S to S', then there is one from S' to S. The flip graph may have loops, these correspond to flips that accidentally turn a certain scheme into an equivalent one.

Since we are interested in schemes of low rank, we are interested in paths containing reductions, because these lead us into lower levels. We have not introduced any edges leading to higher levels, although that would be an easy thing to do. For example, given a scheme S containing a rank-one tensor $A \otimes B \otimes \Gamma$, we can replace this tensor by the two tensors $A \otimes B \otimes (\Gamma - \Gamma')$ and $A \otimes B \otimes \Gamma'$, for arbitrary $\Gamma' \in K^{p \times n}$. The result is a scheme that admits a reduction to S. We could call this step a split of Γ. A split produces a correct matrix multiplication scheme as long as the original scheme does not already contain any of the newly added elements. With this observation, we can show that the flip graph is connected.

Theorem 9. *For every $n, m, p \in \mathbb{N}$, the (n, m, p)-flip graph is weakly connected, i.e., the undirected graph obtained from it by replacing every reduction by a bidirectional edge, is connected.*

Proof. For any given scheme S_0, we construct a path to the standard algorithm in the undirected graph. The first part of the path consists of reductions to an irreducible scheme S_1. Then any two elements $A \otimes B \otimes \Gamma, A' \otimes B' \otimes \Gamma' \in S_1$ are such that $A \otimes B$ and $A' \otimes B'$ are linearly independent. This ensures that splits of Γ lead to pairwise distinct rank-one tensors. Next we repeatedly split Γ for every element to construct a scheme S_2 such that every element of S_2 can be written as $A \otimes B \otimes c_{i,j}$ for some $i, j \in \mathbb{N}$.

For any two elements $A \otimes B \otimes \Gamma, A' \otimes B' \otimes \Gamma' \in S_2$ where A and A' are linearly dependent and Γ and Γ' are linearly dependent there is a reduction that combines these two elements. We follow such reductions until we get a scheme S_3 where any two elements $A \otimes B \otimes \Gamma, A' \otimes B' \otimes \Gamma' \in S_3$ have the property that $A \otimes \Gamma$ and $A' \otimes \Gamma'$ are linearly independent. Then we can repeatedly split B for every element to construct a scheme S_4 where every element has the form $A \otimes b_{i,j} \otimes c_{k,l}$. We then use reductions to combine all elements of S_4 with matching $b_{i,j}$ and $c_{k,l}$. This way we get matrix multiplication scheme S_5 which for all $i, j, k, l \in \mathbb{N}$ contains at most one element of the form $A \otimes b_{i,j} \otimes c_{k,l}$. Therefore, S_5 must be the standard algorithm. □

We are interested in paths in the flip graph that lead to schemes of low rank. Such paths are more likely to exist if there are many flips. Therefore, we select a ground field K for which we can expect the number of flips to be large. As specified in Def. 4 and justified in Thm. 7, a flip is possible whenever a scheme contains two rank-one tensors sharing a common factor. The chances for a common factor are higher if the field K is small, because the smaller the field, the smaller the number of possible factors. For this reason, we consider the ground field $K = \mathbb{Z}_2$ in the following experiments. For this ground field Thm. 7 implies that the only way to replace two rank-one tensors in a scheme is a flip.

Corollary 10. *Let $r \in \mathbb{N}$ and let S and S' be two irreducible matrix multiplication schemes of rank r over $\mathbb{Z}_2$ that differ in exactly two elements. Then S' is a flip of S.*

Proof. Let $A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)}, A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)} \in S$ and $A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)}, A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)} \in S'$ be those elements. Then we have

$$\begin{aligned} &A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)} + A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)} \\ &= A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)} + A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)} \end{aligned}$$

and

$$\begin{aligned} &\{A^{(1)} \otimes B^{(1)} \otimes \Gamma^{(1)},\ A^{(2)} \otimes B^{(2)} \otimes \Gamma^{(2)}\} \\ &\neq \{A^{(3)} \otimes B^{(3)} \otimes \Gamma^{(3)},\ A^{(4)} \otimes B^{(4)} \otimes \Gamma^{(4)}\}. \end{aligned}$$

Since S is not reducible at least two of $\langle A^{(1)}, A^{(2)}\rangle$,$\langle B^{(1)}, B^{(2)}\rangle$ and $\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle$ have dimension two, say

$$\dim\langle B^{(1)}, B^{(2)}\rangle = \dim\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = 2.$$

Then from Theorem 7 follows that $\dim\langle A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}\rangle = 1$ and therefore we also have $A^{(1)} = A^{(2)} = A^{(3)} = A^{(4)}$. Moreover, $\langle B^{(1)}, B^{(2)}\rangle = \langle B^{(3)}, B^{(4)}\rangle$ and $\langle \Gamma^{(1)}, \Gamma^{(2)}\rangle = \langle \Gamma^{(3)}, \Gamma^{(4)}\rangle$. Because of $K = \mathbb{Z}_2$, this can only be if $B^{(3)} = B^{(1)} + B^{(2)}$ or $\Gamma^{(3)} = \Gamma^{(1)} + \Gamma^{(2)}$ and likewise for $B^{(4)}$ and $\Gamma^{(4)}$. Thus, S' is a flip of S. □

4 2×2 MATRICES AND 3×3 MATRICES

The $(2, 2, 2)$-flip graph of rank at most 8 for $K = \mathbb{Z}_2$ is not too big. Fig. 1 shows the connected component containing the standard algorithm. It has 272 vertices, each representing the orbit of one scheme, and 1183 edges, 7 of which are reductions (shown by green arrows). The component also contains Strassen's algorithm. The distance between the standard algorithm and Strassen's algorithm is 8, a path is highlighted in the figure. Although the standard algorithm allows many flips, it only has one neighbor, because any two schemes obtained by a flip from the standard algorithm are equivalent. The diameter of the component is 12.

Using SAT solvers as in [9, 10], we have tried to find out whether there are other matrix multiplication schemes of rank at most 8. For 2×2 matrices, modern SAT solvers have no trouble generating many multiplication schemes in a short time. Strangely enough, while we found many solutions belonging to the connected component shown in Fig. 1, we only found exactly one solution (up to symmetries) that does not belong to this component:

$$\begin{aligned} \mathcal{M}_{2,2,2} = {} & (a_{1,1} + a_{2,2}) \otimes (b_{1,1} + b_{2,2}) \otimes (c_{2,1}) \\ & + (a_{2,2}) \otimes (b_{1,1} + b_{2,1}) \otimes (c_{1,2} + c_{2,1} + c_{2,2}) \\ & + (a_{2,1} + a_{2,2}) \otimes (b_{1,1}) \otimes (c_{1,2} + c_{2,2}) \\ & + (a_{1,1}) \otimes (b_{1,2} + b_{2,2}) \otimes (c_{1,1}) \\ & + (a_{1,2}) \otimes (b_{1,1} + b_{1,2} + b_{2,1} + b_{2,2}) \otimes (c_{1,1} + c_{2,1} + c_{2,2}) \\ & + (a_{1,2} + a_{2,1}) \otimes (b_{1,1} + b_{1,2}) \otimes (c_{2,2}) \\ & + (a_{1,2} + a_{2,2}) \otimes (b_{2,1} + b_{2,2}) \otimes (c_{2,1} + c_{2,2}) \\ & + (a_{1,1} + a_{1,2}) \otimes (b_{1,1} + b_{1,2} + b_{2,2}) \otimes (c_{1,1} + c_{2,1}). \end{aligned}$$

This scheme has no neighbors and thus forms a connected component of its own. We do not know whether the $(2, 2, 2)$-flip graph of rank at most 8 has any further components.

For 3×3-matrices and $K = \mathbb{Z}_2$, the flip graph is so large that it is no longer possible to determine the entire component of the standard algorithm in the $(3, 3, 3)$-flip graph of rank at most 27. Again, and for the same reason as before, the standard algorithm itself has only one neighbor. At distance 2, we found 600 vertices, at distance 3 there are about 20000, and at distance 4 nearly 600000. None of them is reducible. Computing the whole neighborhood of distance 5 is infeasible.

With long random walks however, it is quite likely to encounter reducible vertices. We employ the following simple procedure to search for reductions.

Algorithm 1. *Input: A matrix multiplication scheme S and a limit ℓ for the path length.*
Output: A matrix multiplication scheme with rank decreased by one or $\perp$.

```
if $S$ has no neighbors, return $\perp$
for $i = 1, \ldots, \ell$ do:
    if $S$ is reducible, then return a reduction of $S$.
    if one of the neighbors of $S$ is reducible, then return a reduction of it.
    Set $S$ to a randomly selected neighbor of $S$.
return $\perp$
```

An implementation of this procedure in C can explore paths of lengths 10^8 within minutes and needs almost no memory.

Starting from the standard algorithm we easily find schemes of rank 23, matching the record set by Laderman in 1976 [14], but we found no scheme of rank 22. Restricting the lengths of the random walks to 10^7, more than 95% of the walks reach a scheme of rank 23, and almost all a scheme of rank 24. Recall that along a random walk, the rank can only decrease but not increase. In Fig. 2, we show for 10000 random walks after how many steps they reach a scheme of a specific rank. Specifically searching for short paths, we found a path of length 13 from the standard algorithm to a scheme of rank 23. Such short paths are very rare, longer ones are much easier to find.

If we now focus on the flip graph of rank at most 23, it is feasible to determine for a given vertex the entire connected component to which it belongs. The schemes we reached by random walks from the standard algorithm turned out to belong to 584 different connected components with altogether 64061 vertices. The components are quite diverse with respect to size and symmetry; Fig. 3 shows three examples. The component on the top has 681 vertices and has no automorphisms. Some connected components are isomorphic to each other, typically such components enjoy nontrivial symmetries. For example, the component shown at the bottom right of Fig. 3 has an automorphism group of order 576 and appears 32 times.

The small component shown on the left appears 39 times and has an automorphism group of order 8. The triangular structure in this component appears often in the flip graph. It originates from the two possibilities to choose T for a flip. Two flips of a scheme that use the same rank-one tensors in the flip always are adjacent. The square structure in the middle appears whenever there is an element

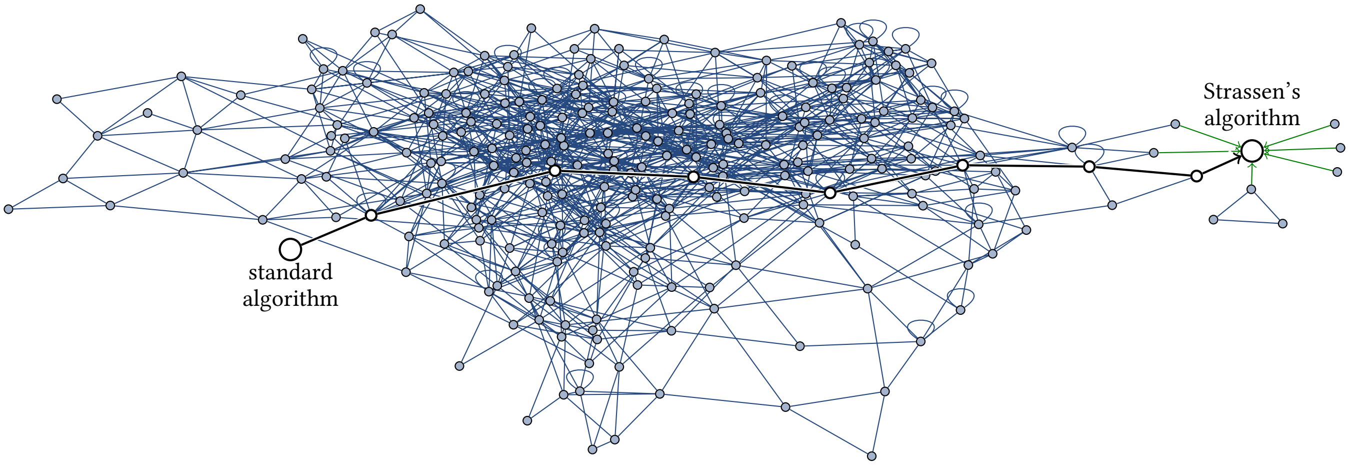

Figure 1: In the $(2,2,2)$-flip graph of rank at most 8, this figure shows the component containing the standard algorithm. Flips are depicted by undirected edges and reductions by directed edges.

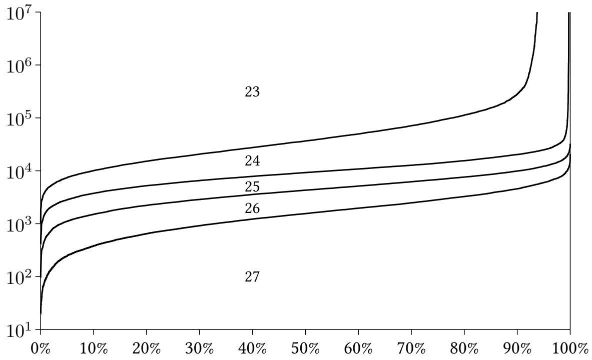

Figure 2: Sparsity of reduction steps. If a point (x, y) belongs to a region labeled r, then for x random paths starting from the standard algorithms, the yth vertex has rank r.

that shares a different factor with each of two other elements of a scheme.

There are also components consisting of a single vertex. Laderman's scheme is such an example.

It seems that not all schemes of rank 23 can be reached from the standard algorithm, because the 64000 solutions we were able to obtain by random paths starting from the standard algorithm do not include all the 17000 solutions found in [10] using SAT solving. Of course, given the size of the graph and the lengths of the paths, it is virtually impossible to check whether there really is no path or we were just not lucky enough to find it.

5 OTHER FORMATS

For larger matrix formats, it becomes harder to find random paths starting from the standard algorithm that go all the way down to a scheme of low rank. An adjusted search strategy that simultaneously considers many partial random paths was found to work more efficiently. In this variant, we maintain a pool of schemes of a certain rank r from which we randomly choose starting points, then do random walks starting from there until either a length limit or a scheme of rank $r - 1$ is encountered. In the latter case, the new scheme is saved. The procedure is repeated until a prescribed number of schemes of rank $r - 1$ is reached. Then these schemes form the new pool of starting points and the method is repeated until the desired target rank is reached.

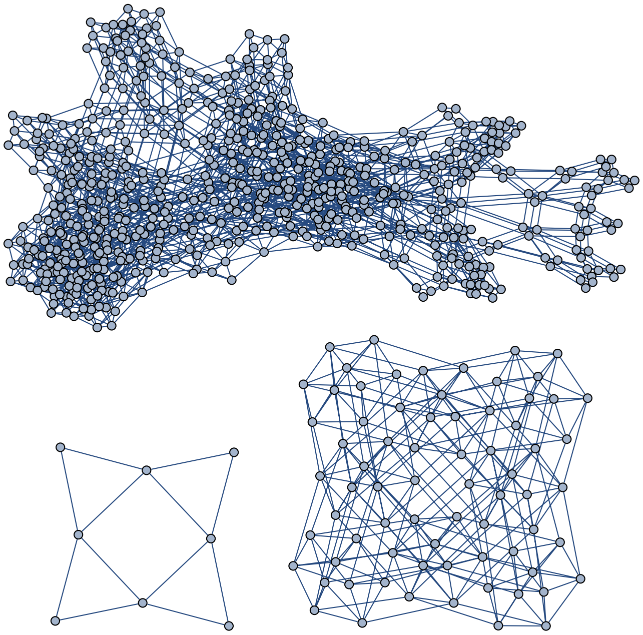

Figure 3: Three example components of the $(3,3,3)$-flip graph of rank at most 23.

Algorithm 2. *Input: A set P of schemes of a certain rank, a path length limit ℓ, a pool size limit s, and a target rank r*
Output: A set Q of s schemes of rank r

```
if $P$ consists of schemes of rank $r$, return $P$.
$Q = \emptyset$
while $|Q| < s$ do:
    apply Alg. 1 to a random element of $P$ and $\ell$.
    if Alg. 1 returns a scheme, add it to $Q$.
call the algorithm recursively with $Q$ in place of $P$.
```

For our experiments, we used as P the set containing only the standard algorithm, $\ell = 10^6$ and $s = 20000$. With these settings, we were able to find schemes matching the best known rank bounds for all (n, m, p) with $2 \leq n, m, p \leq 5$, except for $(n, m, p) = (5, 5, 5)$. For the latter case, starting from the standard algorithm we only get down to rank 97 while Fawzi et al. [8] discovered a scheme of rank 96 (valid mod 2). However, taking their scheme as starting point of a random walk, we discovered schemes of rank 95 within seconds. The typical length of such a random walk is between 10^5 and 10^6 steps. By specifically searching for a short path we found one of length 9, but finding such a short path takes much more time. One of our schemes of rank 95 we announced in [12]. For $(n, m, p) = (4, 4, 5)$, Fawzi et al. give a scheme of rank 63, improving the previous record by one, while we were able to find a scheme of rank 60 starting from the standard algorithm. Also this scheme is only valid mod 2.

One of the remarkable outcomes of the recent work of Fawzi et al. [8] is an apparent discrepancy of the rank depending on the characteristic of the ground field. Their scheme for $(n, m, p) = (4, 4, 4)$ of rank 47 as well as their scheme for $(n, m, p) = (5, 5, 5)$ of rank 96 are only valid in characteristic two and can be shown not to be the homomorphic image of a scheme for $K = \mathbb{Q}$. As our search in the flip graph uses $K = \mathbb{Z}_2$, the question is whether our schemes are also restricted to ground fields of characteristic two. To answer this question, we have applied Hensel lifting [19] to the schemes we discovered.

A set $S = \{((\alpha_{i,j}^{(\ell)})) \otimes ((\beta_{j,k}^{(\ell)})) \otimes ((\gamma_{k,i}^{(\ell)})) : \ell = 1, \ldots, r\}$ of rank-one tensors is a matrix multiplication scheme if and only if the cubic equations

$$\sum_{\ell=1}^{r} \alpha_{i_1,i_2}^{(\ell)} \beta_{j_1,j_2}^{(\ell)} \gamma_{k_1,k_2}^{(\ell)} = \delta_{i_2,j_1} \delta_{j_2,k_1} \delta_{k_2,i_1}$$

are satisfied for all $i_1, i_2, j_1, j_2, k_1, k_2$, where δ is the Kronecker delta function. These equations are known as the Brent equations [3], and finding a matrix multiplication scheme is equivalent to solving these equations.

Knowing a solution valid mod 2^s for some $s \in \mathbb{N}$, we can view it as an approximation to order s of a solution valid in the 2-adic integers, make an ansatz with undetermined coefficients for a refinement of this approximation to order $s + 1$, plug this ansatz into the Brent equations, reduce mod 2^{s+1} and divide by 2^s. This leads to a linear system over $\mathbb{Z}_2$ for the undetermined coefficients in the ansatz, which can be solved with linear algebra. If it has no solution, this proves that the approximation does not admit any refinement to order $s + 1$. If it does have a solution, we pick one and proceed to refine. Once a decent approximation order is reached (we generously used $s = 100$ although much less would have been sufficient in most cases), we can apply rational reconstruction [19] to find a candidate solution with coefficients in $\mathbb{Q}$ or even $\mathbb{Z}$. Whether the reconstruction was successful, i.e., whether the candidate solution over $\mathbb{Q}$ or $\mathbb{Z}$ is indeed a solution, can be checked easily by plugging it into the Brent equations.

(n, m, p)	K	previously best known rank	our rank
$(2, 2, 2)$	any	7 [18]	7
$(2, 2, 3)$	any	11 [11]	11
$(2, 2, 4)$	any	14 [11]	14
$(2, 3, 3)$	any	15 [11]	15
$(2, 2, 5)$	any	18 [11]	18
$(2, 3, 4)$	any	20 [11]	20
$(3, 3, 3)$	any	23 [14]	23
$(2, 3, 5)$	any	25 [11]	25
$(2, 4, 4)$	any	26 [11]	26
$(3, 3, 4)$	any	29 [17]	29
$(2, 4, 5)$	any	33 [11]	33
$(3, 3, 5)$	any	36 [17]	36
$(3, 4, 4)$	any	38 [17]	38
$(2, 5, 5)$	any	40 [11]	40
$(3, 4, 5)$	any	47 [8]	47
$(4, 4, 4)$	$\mathbb{Z}_2$	47 [8]	47
$(4, 4, 4)$	any	49 [18]	49
$(3, 5, 5)$	any	58 [16]	58
$(4, 4, 5)$	$\mathbb{Z}_2$	63 [8]	**60**
$(4, 4, 5)$	any	63 [8]	**62**
$(4, 5, 5)$	any	76 [8]	76
$(5, 5, 5)$	$\mathbb{Z}_2$	96 [8]	**95**
$(5, 5, 5)$	any	98 [16]	**97**

Table 1: Comparison between best known rank and the rank we found

Proceeding as described above, we can confirm the mismatch for $(n, m, p) = (4, 4, 4)$ and $(n, m, p) = (5, 5, 5)$ observed by Fawzi et al. [8]: none of our more than 100000 schemes of rank 47 for $(n, m, p) = (4, 4, 4)$ and none of our more than 30000 schemes of rank 95 for $(n, m, p) = (5, 5, 5)$ can be lifted from $\mathbb{Z}_2$ to $\mathbb{Z}_4$. However, we were able to lift a scheme of rank 97 for $(n, m, p) = (5, 5, 5)$ from $\mathbb{Z}_2$ to $\mathbb{Z}$, thereby breaking the record set by Smirnov and Sedoglavic [16] for this size. Moreover, while none of our schemes of rank 60 for $(n, m, p) = (4, 4, 5)$ could be lifted, we were able to lift some of the schemes of rank 62, thereby breaking the record set by Fawzi et al. [8] for this size. For all other formats, we found no improvements but were able to match the best known bounds on the rank. An overview over the current state of affairs is given in Table 1.

One scheme for each format and the implementation of the search procedure are available at

https://github.com/jakobmoosbauer/flips.git.

The other schemes are available upon request.

6 OPEN QUESTIONS

The flip graph provides a new explanation for the existence of Strassen's algorithm and has led us to improved matrix multiplication schemes for some formats. We believe that the flip graph is interesting in its own right and deserves to be better understood.

For example, is not clear how well-suited the standard algorithm is as starting point for the search procedure. Thm. 9 states that if we allow to use reductions backwards, then every algorithm is reachable from the standard algorithm. However, we found a $(2, 2, 2)$-matrix multiplication scheme of rank 8 that is not connected to the standard algorithm by a path that uses only vertices of rank 8.

Question 1. *For $n, m, p \in \mathbb{N}$, is there a rank r such that all vertices in level r of the (n, m, p)-flip graph are reachable from the standard algorithm?*

If there are schemes of low rank which can not be reached from the standard algorithm, they might become reachable if we add additional edges to the graph. In Corollary 10 we show that, at least over $\mathbb{Z}_2$, the only way to replace exactly two rank-one tensors in one step is a flip.

Question 2. *Under which condition can we replace more than two rows of a matrix multiplication scheme at the same time, such that the resulting scheme is not necessarily reachable by a sequence of flips?*

Another way to add edges to the graph would be to add for every reduction also the reverse edge. However, this would create a lot of additional edges to higher levels. It is not clear at which points in the search procedure one should go to a higher level and which of these edges to use.

Question 3. *How can we utilize edges leading to a higher level in the search procedure?*

The search procedure would also benefit if we could determine whether two vertices belong to the same connected component in the current level. This would allow us to restrict the pool of schemes in Algorithm 2 such that there are not too many vertices in the same component and thus the search potentially covers a larger part of the graph.

Question 4. *Given two matrix multiplication schemes of the same format and rank, is there an efficient way to determine whether they are connected within one level of the flip graph?*

More generally, in order to search for matrix multiplication schemes of low rank, there might be a better way than following random paths in the graph, considering that there seem to be paths much shorter than those we typically find.

Question 5. *Given a matrix multiplication scheme S, is there a systematic way to find reduction edges that can be reached from S?*

Finally, we observed that many of the connected components in the $(3, 3, 3)$-flip graph of rank at most 23 are highly symmetric. Understanding these symmetries would help understanding the structure of the flip graph and might be useful in the search procedure.

Question 6. *What is the significance of the high symmetry in some of the large components in the $(3, 3, 3)$-flip graph of rank at most 23?*

Acknowledgement. We thank Martina Seidl for offering some of her computing power for conducting the experiments reported in this paper and her student Max Heisinger for valuable technical support with their system.

REFERENCES

[1] Josh Alman and Virginia Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proc. SODA'21*, pages 522–539, 2021.

[2] Markus Bläser. On the complexity of the multiplication of matrices of small formats. *J. Complexity*, 19(1):43–60, 2003.

[3] Richard P. Brent. Algorithms for matrix multiplication. Technical report, Department of Computer Science, Stanford, 1970.

[4] Peter Bürgisser, Michael Clausen, and Mohammad A Shokrollahi. *Algebraic complexity theory*, volume 315. Springer Science & Business Media, 2013.

[5] Nicolas T. Courtois, Gregory V. Bard, and Daniel Hulme. A new general-purpose method to multiply 3×3 matrices using only 23 multiplications. Technical Report 1108.2830, ArXiv, 2011.

[6] Hans F. de Groote. On varieties of optimal algorithms for the computation of bilinear mappings ii. optimal algorithms for 2×2-matrix multiplication. *Theoretical Computer Science*, 7(2):127–148, 1978.

[7] Ran Duan, Hongxun Wu, and Renfei Zhou. Faster matrix multiplication via asymmetric hashing. Technical Report 2210.10173, arXiv, 2022.

[8] Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J. R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, David Silver, Demis Hassabis, and Pushmeet Kohli. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

[9] Marijn J.H. Heule, Manuel Kauers, and Martina Seidl. Local search for fast matrix multiplication. In *Proc. SAT'19*, pages 155–163, 2019.

[10] Marijn J.H. Heule, Manuel Kauers, and Martina Seidl. New ways to multiply 3×3-matrices. *J. Symbolic Comput.*, 104:899–916, 2021.

[11] J. E. Hopcroft and L. R. Kerr. On minimizing the number of multiplications necessary for matrix multiplication. *SIAM Journal on Applied Mathematics*, 20(1):30–36, 1971.

[12] Manuel Kauers and Jakob Moosbauer. The FBHHRBNRSSSHK-algorithm for multiplication in $\mathbb{Z}_2^{5\times5}$ is still not the end of the story. Technical Report 2210.04045, arXiv, 2022.

[13] Manuel Kauers and Jakob Moosbauer. A normal form for matrix multiplication schemes. In Dimitrios Poulakis and George Rahonis, editors, *Proc. CAI'20*, pages 149–160, 2022.

[14] Julian D. Laderman. A noncommutative algorithm for multiplying 3×3 matrices using 23 multiplications. *Bull. Amer. Math. Soc.*, 82(1):126–128, 1976.

[15] V. Ya. Pan. New fast algorithms for matrix operations. *SIAM Journal on Computing*, 9(2):321–342, 1980.

[16] Alexandre Sedoglavic and Alexey V. Smirnov. The tensor rank of 5×5 matrices multiplication is bounded by 98 and its border rank by 89. In *Proc. ISSAC'21*, pages 345–351, 2021.

[17] Alexey V. Smirnov. The bilinear complexity and practical algorithms for matrix multiplication. *Zh. Vychisl. Mat. Mat. Fiz.*, 53(12):1970–1984, 2013.

[18] Volker Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[19] Joachim von Zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge Univ. Press, 3 edition, 2013.

[20] Shmuel Winograd. On multiplication of 2×2 matrices. *Linear Algebra and its Applications*, 4(4):381–388, 1971.