On the Complexity of Matrix Multiplication

Andrew James Stothers

Doctor of Philosophy
University of Edinburgh
2010

In principio erat Verbum,
et Verbum erat apud Deum,
et Deus erat Verbum.
Hoc erat in principio apud Deum.
Omnia per ipsum facta sunt,
et sine ipso factum est nihil, quod factum est;
in ipso vita erat,
et vita erat lux hominum,
et lux in tenebris lucet
et tenebrae eam non comprehenderunt.

Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Andrew James Stothers*)

This thesis is dedicated to my parents, Jim and Lynda.

Abstract

The evaluation of the product of two matrices can be very computationally expensive. The multiplication of two $n \times n$ matrices, using the "default" algorithm can take $O(n^3)$ field operations in the underlying field k. It is therefore desirable to find algorithms to reduce the "cost" of multiplying two matrices together. If multiplication of two $n \times n$ matrices can be obtained in $O(n^\alpha)$ operations, the least upper bound for α is called the *exponent of matrix multiplication* and is denoted by ω.

A bound for $\omega < 3$ was found in 1968 by Strassen in his algorithm. He found that multiplication of two 2×2 matrices could be obtained in 7 multiplications in the underlying field k, as opposed to the 8 required to do the same multiplication previously. Using recursion, we are able to show that $\omega \leq \log_2 7 < 2.8074$, which is better than the value of 3 we had previously.

In chapter 1, we look at various techniques that have been found for reducing ω. These include Pan's Trilinear Aggregation, Bini's Border Rank and Schönhage's Asymptotic Sum inequality.

In chapter 2, we look in detail at the current best estimate of ω found by Coppersmith and Winograd. We also propose a different method of evaluating the "value" of trilinear forms.

Chapters 3 and 4 build on the work of Coppersmith and Winograd and examine how cubing and raising to the fourth power of Coppersmith and Winograd's "complicated" algorithm affect the value of ω, if at all.

Finally, in chapter 5, we look at the Group-Theoretic context proposed by Cohn and Umans, and see how we can derive some of Coppersmith and Winograd's values using this method, as well as showing how working in this context can perhaps be more conducive to showing $\omega = 2$.

Acknowledgements

The most gratitude goes to my supervisor, Sandy Davie, who not only introduced me to the topic, but also helped me understand it and encouraged me when it almost became too much. He comes highly recommended as a supervisor. I would also like to mention Istvan Gyongy, my second supervisor for his encouragement and Jim Wright for encouraging me to take this on after my Undergraduate degree.

I would also like to thank the secretarial staff, who were a pleasure to work with.

I am indebted to the Engineering and Physical Science Research Council and to the School of Mathematics for the generous financial support.

I would like to say thanks to my PG colleagues for the irreverent (and often irrelevant) discussions at lunchtime and elsewhere. I feel blessed that I am able to work with such great people.

Outside the department, I would like to thank James Whiteford and John Walker for the encouraging chat (and occasional Pool) at lunchtimes.

My heartfelt thanks go to the people of St Catherine's Argyle Church of Scotland for supporting me prayerfully (and often for feeding me!), especially to those in the 20s and 30s Group.

God has blessed me with such stong Christian friends, and I am eternally thankful to Him for this.

I would like to name and thank my flatmates (current and old) James, Edward, Alasdair, Jaideep, Andrew, William, Luke, Douglas, and Mark, and my good friends Emma, David, Justin, Steve, Laura, Catherine, Tim, AnnaLauren, Colin, Kirsty and Philip for praying, listening to me, and putting up with my mood swings.

Contents

Chapter 1

Introduction and Background

1.1 Introduction

This thesis aims to look at the concept of *Matrix Multiplication.* We consider that the number of operations over a field k required to multiply two $n \times n$ matrices is $O(n^{\omega})$. We look at the ways in which ω may be reduced, leading to faster algorithms to multiply two matrices of this type together.

In the first chapter, we look at how this problem has been approached historically: we look at the techniques that have been used and how bounds for ω have been affected by them.

In the second chapter, we look in detail at the current upper bound for ω found by Coppersmith and Winograd [12], and how it was obtained.

Chapters 3 and 4 take the algorithm with which Coppersmith and Winograd get their record bound and raise it to the third and fourth powers respectively. We explain why these algorithms are more complex and investigate how they change ω.

In chapter 5, we look at Cohn and Umans new Group-Theoretic framework for matrix multiplication, introduced in [9], placing some of Coppersmith and Winograd's discoveries in this context and explaining how proving some combinatorical conjectures can show that $\omega = 2$.

1.2 History of the problem

in 1968, Winograd [28] made the discovery that, by using a different method of calculating the inner product, one could find the product of two $n \times n$ matrices, which, while using a similar number of overall operations, shifted the emphasis more on addition than on multiplication. This was important as addition was computationally less demanding than multiplication.

The same year, Strassen [24] provided an explicit algorithm which could multiply two $2^n \times 2^n$ matrices in less than 6.7^n operations, where using Winograd or the trivial algorithm, we would have had approximately 8^n operations. Using this, it is shown that $\omega \leq \log_2(7) < 2.81$.

In 1978, Pan [18] (also in [19],[20]) found explicit algorithms to further reduce ω by means of the technique of *trilinear aggregation.* This technique uses the fact that computing the trace of the product of three $n \times n$ matrices is equivalent to the problem of multiplying two $n \times n$ matrices (in terms of the total number of multiplications). By defining a function on the indices of the entries in the matrices A, B and C to be multiplied, we may define an *aggregate* to do all the required multiplications, plus some extra terms. We *unite* terms to remove these extra terms in as few calculations as possible. Using this technique, Pan shows that we can multiply two 70×70 matrix multiplications in 143640 operations. This gives $\omega \leq \log_{70} 143640 < 2.79512$, and further, we can perform a 46×46 matrix multiplication in 41952 operations, giving $\omega \leq 2.78017$.

In 1980, Bini et al. [3] showed that the number of operations required to perform a matrix multiplication could be reduced by considering *approximate algorithms*. If we change our underlying field k to be the field of polynomials of λ, a variable which, if k is $\mathbb{R}$ can be assumed to be just a small number (allowing negative powers of λ) with coefficients in k, we may obtain, using fewer operations, an approximation of the required matrix multiplication (in the sense that each entry will be "out" by a power of λ). Using this method (which is similar to trilinear aggregation), they obtain $\omega \leq 2.7799$.

In 1981, Schönhage [23] showed that an algorithm which could approximately compute multiple independent matrix multiplications could be used to further reduce ω. This is the result of his *asymptotic sum inequality*- using it, he shows that $\omega \leq 2.5479$.

Using similar techniques, Coppersmith and Winograd [11] showed that one can take an algorithm (of a certain type) that can perform multiple disjoint matrix multiplications and square it. The resulting algorithm will be capable of multiplying larger matrices than expected. This method gives $\omega \leq 2.4955480$.

In 1986 Strassen [25],[26] showed that one could start with an algorithm that was not a matrix product: we have a series of blocks, where the blocks can themselves be seen as elements of a matrix multiplication, and the blocks themselves are matrix multiplications. Raising this original algorithm to a large power, we may set some blocks to zero to obtain a large number of independent matrix products: we then use Schönhage to find a value for ω. This method yields $\omega \leq 2.4785$.

In 1987, [12]Coppersmith and Winograd used this method to great effect to provide the current record for ω. They start with an algorithm, raise it to the Nth power and show that setting certain variables as being zero will lead to the algorithm calculating a large number of independent matrix products of a certain size: using Schönhage, we get that $\omega \leq 2.376$.

In 2005, Cohn and Umans [9],[10] placed the matrix multiplication in a group theoretic context: while they were unable to find new bounds for ω, the group-theoretic context provides new conjectures, which, if proved, will show that $\omega = 2$.

A related problem is determining the *rank* of Matrix Multiplication. The rank is the total number of non-scalar multiplications required to evaluate a Matrix product (including scalar multiplications this becomes the *Multiplicative Complexity*).

1.3 The main problem

Matrices have long been the subject of much study by many Mathematicians. However, the rise of computers in the late 20th century has led to new problems, the main one being the problem of *Matrix Multiplication*.

Computers are required to do many Matrix Multiplications at a time, and hence it is desirable to find algorithms to reduce the number of steps required to multiply two matrices together. Until 1968, we had only the *Trivial Algorithm* to multiply matrices together. This is as follows:

Algorithm 1. *If we have two $n \times n$ matrices, $n \in \mathbb{N}$, $\mathbf{A}$ and $\mathbf{B}$, with entries in a field*

k, such that

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots \\ a_{2,1} & a_{2,2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} \text{ and } \mathbf{B} = \begin{pmatrix} b_{1,1} & b_{1,2} & \cdots \\ b_{2,1} & b_{2,2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix},$$

then

$$[AB]_{p,q} = \sum_{i=1}^{n} a_{p,i} b_{i,q}$$

where multiplication is defined as in the field k.

We see that this algorithm requires $2n^3 - n^2$ operations in k to multiply two $n \times n$ matrices, of which n^3 are multiplications and $n^3 - n^2$ are additions.

However in 1968, Winograd [28] showed that one could take the inner product of two vectors using fewer multiplications, but with more additions.

We consider finding the inner product of two vectors $(x_1, .., x_n)$ and $(y_1, .., y_n)$. Set

$$\begin{aligned} \xi &= \sum_{j=1}^{\lfloor n/2 \rfloor} x_{2j-1} x_{2j} \\ \eta &= \sum_{i=1}^{\lfloor n/2 \rfloor} y_{2j-1} y_{2j}. \end{aligned}$$

Then the inner product is given, for even n, by

$$\sum_{j=1}^{\lfloor n/2 \rfloor} (x_{2j-1} + y_{2j})(x_{2j} + y_{2j-1}) - \xi - \eta$$

and for odd n by

$$\sum_{j=1}^{\lfloor n/2 \rfloor} (x_{2j-1} + y_{2j})(x_{2j} + y_{2j-1}) - \xi - \eta + x_n y_n.$$

Hence the total number of multiplications required is

$$2\lfloor n/2 \rfloor + \lfloor (n+1)/2 \rfloor$$

and the number of additions required is

$$2(\lfloor n/2 \rfloor - 1) + (n + \lfloor n/2 \rfloor + 1).$$

Since matrix multiplication can be regarded as taking multiple inner products, we get that the total number of multiplications required to multiply an $n \times n$ matrix by another matrix of the same size is about $n^3/2$ and the number of additions required is about $(3/2)n^3$, improving slightly on the trivial algorithm. Denote by $M_k(n)$ the total number of operations required by a bilinear algorithm to multiply two $n \times n$ matrices over k.

Definition 1. *The* exponent of matrix multiplication over a field k *is defined as*

$$\omega(k) := \inf\{\tau \in \mathbb{R} | M_k(n) = O(n^\tau)\}.$$

We see from the trivial algorithm that $\omega(k)$ has an upper limit of 3. Since there must be an output of n^2 entries, there cannot be any fewer operations than this in total matrix multiplication, so hence $\omega(k) \in [2,3]$. We see that the total number of operations in Winograd's algorithm implies that the least upper bound for ω is 3. However, in the same year, Strassen managed to find an improvement for ω.

1.4 Strassen's Algorithm

In [24], Strassen demonstrated via a recursive method that $\omega \leq \log_2 7 = 2.807....$ The improvements arise because it was found to be possible to multiply two 2×2 matrices in just 7 multiplications as opposed to 8 using the trivial algorithm.

We suppose we want to multiply two 2×2 matrices over a field k, $\mathbf{A}$ and $\mathbf{B}$. The algorithm then works as follows:

Algorithm 2. *First we write*

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{pmatrix}, \mathbf{AB} = \begin{pmatrix} c_{1,1} & c_{1,2} \\ c_{2,1} & c_{2,2} \end{pmatrix}.$$

Compute

- $I = (a_{1,1} + a_{2,2})(b_{1,1} + b_{2,2}),$
- $II = (a_{2,1} + a_{2,2})b_{1,1},$
- $III = a_{1,1}(b_{1,2} - b_{2,2}),$
- $IV = a_{2,2}(-b_{1,1} + b_{2,1}),$
- $V = (a_{1,1} + a_{1,2})b_{2,2}$
- $VI = (-a_{1,1} + a_{2,1})(b_{1,1} + b_{1,2})$
- $VII = (a_{1,2} - a_{2,2})(b_{2,1} + b_{2,2})$

Then we have

- $c_{1,1} = I + IV - V + VII$
- $c_{2,1} = II + IV$
- $c_{1,2} = III + V$
- $c_{2,2} = I + III - II + VI.$

It is easy to check that the expressions obtained here match the ones one would have obtained using the trivial algorithm. Now, this algorithm becomes more powerful when we make use of *recursion.* We will show later on how we reduce the matrix exponent using this algorithm, but first some important concepts need to be defined.

1.5 The rank of a Bilinear Map

We now turn to the more general field of bilinear maps to define the rank of matrix multiplication.

Definition 2. *Let U, V, W be vector spaces over a field k. A* Bilinear map *is a map $\phi : U \times V \to W$ satisfying*

$$\phi(\lambda_{11}u_1 + \lambda_{12}u_2, \lambda_{21}v_1 + \lambda_{22}v_2) = \sum_{i,j \leq 2} \lambda_{i,j}\phi(u_i, v_j)$$

for all $\lambda_{ij} \in K, u_i \in U, v_j \in V$.

From this definition of bilinear maps, it is easy to see that Matrix Multiplication is a bilinear map.

Definition 3. *Let $\phi : U \times V \to W$ be a bilinear map over a field k. For $i \in [1, .., r]$ let $f_i \in U^*$, $g_i \in V^*$, $w_i \in W$ be such that*

$$\phi(u, v) = \sum_{i=1}^{r} f_i(u)g_i(v)w_i$$

for all $u \in U$, $v \in V$. Then $(f_1, g_1, w_1; ...; f_r, g_r, w_r)$ is called a bilinear computation (algorithm) of length r for ϕ.

Definition 4. *The length of a shortest bilinear computation for ϕ is called the* bilinear complexity *or the* rank *of ϕ and is denoted by $R(\phi)$.*

We can re-write Strassen's algorithm in the terms of Definition 3.
Set

- $f_1 = A_{11} + A_{22}, g_1 = B_{11} + B_{22}$
- $f_2 = A_{21} + A_{22}, g_2 = B_{11}$
- $f_3 = A_{11}, g_3 = B_{12} - B_{22}$
- $f_4 = A_{22}, g_4 = -B_{11} + B_{21}$
- $f_5 = A_{11} + A_{12}, g_5 = B_{22}$
- $f_6 = -A_{11} + A_{21}, g_6 = B_{11} + B_{12}$
- $f_7 = A_{12} - A_{22}, g_7 = B_{21} + B_{22}$

$$\mathbf{w_1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{w_2} = \begin{pmatrix} 0 & 0 \\ 0 & -1 \end{pmatrix}, \mathbf{w_3} = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}, \mathbf{w_4} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix},$$
$$\mathbf{w_5} = \begin{pmatrix} -1 & 0 \\ -1 & 0 \end{pmatrix}, \mathbf{w_6} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{w_7} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

We thus see that the rank of 2 × 2 by 2 × 2 matrix multiplication is at most 7. Were we to do the same thing for the trivial algorithm, we would see that the rank was 8. We denote the problem of multiplying a $m \times n$ matrix by a $n \times p$ matrix by $\langle m, n, p \rangle$, and we can thus say that

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

Waksman [27] also showed this rank was possible by modifying Winograd's algorithm slightly.
This concept of rank is important as it is a measure of how efficient an algorithm is. We explain some of its more important properties.

1.5.1 Properties of the Rank

First, we show that the rank of a concise bilinear map is greater than $\max(dim(U), dim(V), dim(W))$.

Definition 5. *A bilinear map is* concise *if and only if the left kernel* $\{u \in u | \phi(u,v) = 0 \; \forall \; v \in V\} = 0$ *and the right kernel* $\{v \in V | \phi(u,v) = 0 \; \forall \; u \in U\} = 0$ *and if the span of* $\phi(U,V) = W$.

From this definition it is easy to show that matrix products are concise.

Lemma 1. *The rank of a concise bilinear map is greater than or equal to* $\max(dim(U), dim(V), dim(W))$.

Proof. If the rank of a map is less than the dimension of U then the f_i do not form a basis for U^*. Hence, one can always find a non-zero $u \in U$ such that $f_i(u) = 0$ for all f_i, and hence ϕ will have a non-zero kernel, contradiciting conciseness.
An analogous argument holds for V.
If the rank of the bilinear map is less than the dimension of W, then the dimension of the image of $\phi(U,V)$ will be less than the dimension of the space W, contradicting conciseness.
Therefore, the assertion must hold. □

We work with rank (rather than the total number of operations) as it is better behaved than the total number of operations. From proposition 15.1 of [8], we may actually define ω in terms of the rank:

Proposition 1. *For every field k we have*

$$\omega(k) = \inf\{\tau \in \mathbb{R} | R(\langle h,h,h \rangle) = O(h^\tau)\}.$$

Proof. (sketch) Using a bilinear algorithm, it may be shown that the total number of operations $M_k(\langle m^{i+1}, m^{i+1}, m^{i+1} \rangle)$ required to multiply two $m^{i+1} \times m^{i+1}$ is

$$rM_k(\langle m^i, m^i, m^i \rangle) + cm^{2i}$$

for a c which depends on m and r. Solving the recurrence relation yields

$$M_k(\langle m^i, m^i, m^i \rangle) \leq \alpha r^i + \beta m^{2i}$$

where $\alpha = M(1) + m^2 c/(r - m^2)$ and $\beta = -m^2 c/(r - m^2)$
which yields $M_k(\langle m^i, m^i, m^i \rangle) = O(r^i)$. Using manipulation of logs and the definition of ω, the statement follows. □

The above proposition ensures that one can find a bound for the total number of operations by considering the rank alone.
Other important properties are:
The rank is invariant when permuting the sizes of the matrices, i.e.

$$R(\langle e,h,l \rangle) = R(\langle h,l,e \rangle) = R(\langle l,e,h \rangle). \tag{1.1}$$

The proof of this requires us to introduce a different notation.

Proposition 2. *If U, V, W are vector spaces over a field k, there exists a unique isomorphism $U^* \otimes V^* \otimes W \to Bil(U, V; W)$ which sends $f \otimes g \otimes w$ to the bilinear map $(u, v) \mapsto f(u)g(v)w$.*

In other words, instead of an explicit map, we have an equivalent tensor in $U^* \otimes V^* \otimes W$.

Proof. There exists a unique homomorphism $\sigma : U^* \otimes V^* \otimes W \to Bil(U, V; W)$ which sends (f, g, w) to the map $(u, v) \mapsto f(u)g(v)w$. Surjectivity is shown by considering the bases of U, V, W. □

This unique tensor is the *structural tensor* of ϕ.
The *rank* of the structural tensor is the minimum number of triads $u_i \otimes v_i \otimes w_i$ such that t can be represented as

$$t = \sum_{i=1}^{r} u_i \otimes v_i \otimes w_i.$$

Proof. of 1.4. From the tensorial notation, it is clear that the rank of t is invariant under permutation of the coordinates, hence the rank of ϕ is also invariant under permutation. □

Thus the sum of the triads of $\langle 2, 2, 2\rangle$ is

$$(a_{1,1}b_{1,1} + a_{1,2}b_{2,1}, a_{1,1}b_{1,2} + a_{1,2}b_{2,2}, a_{2,1}b_{1,1} + a_{2,2}b_{2,1}, a_{2,1}b_{1,2} + a_{2,2}b_{2,2})$$

with triads

$$\begin{aligned}
(1,0,0,1) \otimes (1,0,0,1) \otimes (1,0,0,1)\\
(0,0,1,1) \otimes (1,0,0,0) \otimes (0,0,0,-1)\\
(1,0,0,0) \otimes (0,1,0,-1) \otimes (0,0,1,1)\\
(0,0,0,1) \otimes (-1,0,1,0) \otimes (1,1,0,0)\\
(1,1,0,0) \otimes (0,0,0,1) \otimes (-1,-1,0,0)\\
(-1,0,1,0) \otimes (1,1,0,0) \otimes (0,0,0,1)\\
(0,1,0,-1) \otimes (0,0,1,1) \otimes (1,0,0,0).
\end{aligned}$$

If U has a basis $\{u_i\}$, V has a basis $\{v_j\}$ and W a basis $\{e_l\}$, then there exist $t_{ijl} \in k$ such that for all i and j we have

$$\phi(u_i, v_j) = \sum_{l=1}^{dim(W)} t_{ijl} e_l.$$

The entries t_{ijk} are called the *coordinate tensor* of ϕ. We will revisit this notion in the next chapter.

$$R(\phi_1 \oplus \phi_2) \leq R(\phi_1) + R(\phi_2). \tag{1.2}$$

We prove this for general tensors t_1, t_2 of bilinear maps ϕ_1, ϕ_2.

Proof. Let $\phi_1 : U_1 \times V_1 \to W_1$ and $\phi_2 : U_2 \times V_2 \to W_2$ be bilinear maps.
We consider their associated tensors

$$t_1 = \sum_{i=1}^{r} u_{1i} \otimes v_{1i} \otimes w_{1i}$$

and

$$t_2 = \sum_{i=1}^{r'} u_{2i} \otimes v_{2i} \otimes w_{2i}.$$

We consider the tensor $t_1 \oplus t_2$. The triads of this tensor will be in a space isomorphic to

$$(U_1 \oplus U_2) \otimes (V_1 \oplus V_2) \otimes (W_1 \oplus W_2).$$

Let 0_{U_1} be the zero element of the space U_1 (and analogously for other spaces). We then consider the sum

$$\sum_{i=i}^{r} u_{1i} \oplus 0_{U_2} \otimes v_{1i} \oplus 0_{V_2} \otimes w_{1i} \oplus 0_{W_2} + \sum_{j=i}^{r'} 0_{U_1} \oplus u_{2j} \otimes 0_{V_1} \oplus v_{2j} \otimes 0_{W_1} \oplus w_{2j}.$$

This is isomorphic to $t_1 \oplus t_2$ and hence the rank of $\phi_1 \oplus \phi_2$ is less than or equal to $r + r'$. □

$$R(\phi_1 \otimes \phi_2) \leq R(\phi_1)R(\phi_2). \tag{1.3}$$

Proof. We take t_1 and t_2 as previously. We consider the tensor product of

$$\sum_{i=1}^{r} u_{1i} \otimes v_{1i} \otimes w_{1i} \bigotimes \sum_{j=1}^{r'} u_{2j} \otimes v_{2j} \otimes w_{2j}$$

$$\simeq \sum_{i=1}^{r} \sum_{j=1}^{r'} u_{1i} \otimes u_{2j} \otimes v_{1i} \otimes v_{2j} \otimes w_{1i} \otimes w_{2j}.$$

This shows that the rank of $t_1 \otimes t_2$ is at most $r \times r'$.

□

If $e \leq e'$, $h \leq h'$,and $l \leq l'$ are all positive integers, then

$$R(\langle e, h, l\rangle) \leq R(\langle e', h', l'\rangle). \tag{1.4}$$

Proof. Since $e \leq e'$, $h \leq h'$,and $l \leq l'$, we may embed the smaller bilinear map into the larger one by padding the functions with zeroes. □

Using all these facts, we can now finally show that Strassen's algorithm is indeed asymptotically faster than the trivial one.
Using equation 1.4 we have that

$$R(\langle h, h, h\rangle) \leq R(\langle 2^{\lceil \log_2 h \rceil}, 2^{\lceil \log_2 h \rceil}, 2^{\lceil \log_2 h \rceil}\rangle).$$

By equation 1.3 we get

$$R(\langle 2^{\lceil \log_2 h \rceil}, 2^{\lceil \log_2 h \rceil}, 2^{\lceil \log_2 h \rceil} \rangle) \leq R(\langle 2,2,2 \rangle)^{\lceil \log_2 h \rceil},$$

and since we already stated that $R(\langle 2,2,2 \rangle) \leq 7$ we have that

$$R(\langle h,h,h \rangle) \leq 7^{\lceil \log_2 h \rceil}.$$

Some manipulation of logs yields that

$$R(\langle h,h,h \rangle) \leq 7.h^{\log_2 7}$$

and hence, by proposition 1, $\omega \leq log_2 7 < 2.81$ as required.
Implicit in this proof is the concept of *recursion.*
We take the matrices to be multiplied and pad them out with zeroes to make them of the form $2^k \times 2^k$ for an appropriately sized $k \in \mathbb{N}$. We then divide the matrix up as follows:

$$\left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right)$$

where $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are $2^{k-1} \times 2^{k-1}$ matrices.We then treat the resulting matrices as 2×2 matrices, with their elements being themselves $2^{k-1} \times 2^{k-1}$ matrices. We then apply the algorithm: we can multiply 2 2×2 matrices using 7 multiplications. Each of these consists of multiplication of a $2^{k-1} \times 2^{k-1}$ by $2^{k-1} \times 2^{k-1}$ matrix, which can then be done in 7 multiplications.
Using an inductive argument, it can by shown that multiplication of two $2^k \times 2^k$ matrices can be done using 7^k multiplications. Thus the rank of multiplying two $2^k \times 2^k$ matrices together is $\leq 7^k$. Using these, we still have that $\omega \leq 2.81$ for all h.
By considering properties of the rank, we may show that Strassen's algorithm is, in fact, an optimal bilinear computation in the sense that there is no algorithm of rank less than 7 which can compute $\langle 2,2,2 \rangle$.
It was shown by Brocket and Dobkin [7] that

$$R(\langle n,n,n \rangle) \geq 2n^2 - 1$$

therefore, as shown by Hopcroft and Kerr [14], and Winograd [29], the rank of $\langle 2,2,2 \rangle \geq 7$ and so we see that Strassen's algorithm is optimal.
In fact, Lafon and Winograd [16] went further by showing that the multiplicative complexity (including scalar multiplications) of $\langle m,n,p \rangle$ matrix multiplications was greater than or equal to

$$(m+p)n + p - n - 1.$$

A corollary for this is that multiplicative complexity for symmetric matrix multiplications is equal to the rank. The current lower bound for the rank of $\langle n,n,n \rangle$ matrix multiplication over arbitrary fields, as shown by Bläser [4],[5], is

$$\frac{5}{2}n^2 - 3n.$$

Bläser [6] later proves a lower bound of $2mn + 2n - m - 2$ for the rank of $\langle n,m,n \rangle$ matrix multiplication, for $m,n \geq 3$. This means that the lowest rank we can achieve for $\langle 3,3,3 \rangle$ is 19; however no algorithm for exact rank less than 23 has been found.
It was shown by de Groote [13] that Strassen's algorithm is unique in the sense that

every “optimal” algorithm for $\langle 2,2,2\rangle$ can be shown to be equivalent to it.
We also obtain for the above arguments for general matrix products of size $\langle e,h,l\rangle$, $\langle h,l,e\rangle$ and $\langle l,e,h\rangle$ that, if the rank of all these is equal to r:

$$R(\langle ehl, ehl, ehl\rangle) = R(\langle e,h,l\rangle \otimes \langle h,l,e\rangle \otimes \langle l,e,h\rangle) \leq r^3$$

which implies that

$$(ehl)^{\omega/3} \leq r. \tag{1.5}$$

1.6 Trilinear Aggregation

Until Pan in [18], [19] came upon this new method , no-one was able to improve on Strassen's algorithm. He found that complexity of matrix multiplication could be reduced further by remarking that the problem of matrix multiplication was equivalent (in the sense that they share the same rank) to finding the trace of the product of three matrices A,B, C of sizes $m \times n$, $n \times p$ and $p \times m$ respectively (section 3 of [18]).

Proof. If A is an $m \times n$, B a $n \times p$ and C a $p \times m$ matrix respectively, then it is shown that the trace of the product ABC is given by

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{p} A_{ij}B_{jk}C_{ki}.$$

If we set $C_{ki} = 1$ and the other entries of C to 0, we find $\sum_{j=1}^{n} A_{ij}B_{jk}$, that is AB_{ik}. Hence the problem of multiplication of two matrices is contained in this problem of finding the trace of the product of three matrices: hence the number of multiplications required to find the matrix product is less than or equal to the number of multiplications required to find this trace.
To prove this the other way, if the rank of matrix multiplication of two matrices A and B is r, we have that the product ABC may be written, for some $f_i \in k^{m\times n*}, g_i \in k^{m\times p*}, w_i \in k^{p\times m}$:

$$\sum_{i=1}^{r} f_i(A)g_i(B)w_iC.$$

We seek to find the trace of this product. However

$$\text{Trace}\sum_{i=1}^{r} f_i(A)g_i(B)w_iC = \sum_{i=1}^{r} f_i(A)g_i(B)\text{Trace}(w_iC).$$

The function Trace(w_iC) is in the dual of $k^{p\times m}$, so the overall problem of finding the trace of the product of three matrices is a trilinear map of the form

$$\sum_{i=1}^{r} f_i(A)g_i(B)h_i(C)$$

where the h_i is found by taking the trace of each product $w_i C$, that is

$$h_i = \sum_{u=1}^{m} \sum_{v=1}^{p} w_{iuv} C_{vu}.$$

Thus the rank of the two problems is equal.

□

The trace of this matrix is found by computing

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} a_{ij} b_{jk} c_{ki}. \tag{1.6}$$

This is a *Trilinear Map* $\phi : U \times V \times W \to K$. The rank r is the minimal number such that, for $f_i \in U^*$, $g_i \in V^*$, $h_i \in W^*$:

$$\phi(u, v, w) = \sum_{i=0}^{r} f_i(A) g_i(B) h_i(C).$$

We use this to present the concept of *Trilinear Aggregation.* This method is not used for the main results of this thesis: however it is included because of its influence on later methods (notably the use of approximation algorithms).
We begin by recalling the trivial algorithm for finding the trace of the product of three matrices, that is

$$\sum_{i,j,k} a_{i,j} b_{j,k} c_{k,i}.$$

We call each term in this sum *desirable.* Now, we consider the product

$$(a_{i,j} + a_{k_1,i_1})(b_{j,k} + b_{i_1,j_1})(c_{k,i} + c_{j_1,k_1})$$

which we find equals

$$\begin{array}{rcl} a_{i,j} b_{j,k} c_{k,i} & + & a_{k_1,i_1} b_{i_1,j_1} c_{j_1,k_1} + \\ +a_{k_1,i_1} b_{i_1,j_1} c_{k,i} & + & a_{k_1,i_1} b_{j,k} c_{k,i} + \\ +a_{i,j} b_{i_1,j_1} c_{j_1,k_1} & + & a_{i,j} b_{i_1,j_1} c_{k,i} + \\ +a_{k_1,i_1} b_{j,k} c_{j_1,k_1} & + & a_{i,j} b_{j,k} c_{j_1,k_1}. \end{array}$$

Now, the i_1, j_1, k_1 are all functions of i, j, k respectively: we may choose i, j, k and i_1, j_1, k_1 such that all possible combinations of i, j and k in the required range are obtained.
A simple example, as shown in algorithm 1 of [18], is to set our initial i, j, k as being all possible combinations of i, j, k such that $i + j + k$ is even, and to set $i_1 = i + 1, j_1 = j + 1, k_1 = k + 1$.
The following table shows that

$$\sum_{i+j+k \text{ is even}} (a_{i,j} b_{j,k} c_{k,i} + a_{k_1,i_1} b_{i_1,j_1} c_{j_1,k_1}) = \sum_{i,j,k} a_{i,j} b_{j,k} c_{k,i} :$$

- i is even, j is even, k is even → i_1 is odd, j_1, is odd, k_1 is odd
- i is even, j is odd, k is odd → i_1 is odd, j_1, is even, k_1 is even

- i is odd, j is odd, k is even → i_1 is even, j_1, is even, k_1 is odd
- i is odd, j is even, k is odd → i_1 is even, j_1, is odd, k_1 is even.

Thus, the desired result can be obtained using $n^3/2$ scalar multiplications: it remains to find a way of removing the "undesirable" terms in as few operations as possible. We find that we can acheive this by grouping similar "undesirable" terms together. We notice that in our above expansion, we have that both entries in the second row share their a and c terms. We can therefore rewrite

$$a_{k_1,i_1}b_{i_1,j_1}c_{k,i} + a_{k_1,i_1}b_{j,k}c_{k,i} = a_{k_1,i_1}(b_{i_1,j_1} + b_{j,k})c_{k,i}.$$

Thus, if we were to sum this expression over all i, j, k (as we will do shortly), we get

$$\sum_{i,j,k}(a_{k_1,i_1}b_{i_1,j_1}c_{k,i} + a_{k_1,i_1}b_{j,k}c_{k,i}) = \sum_{i,k} a_{k_1,i_1}c_{k,i}\sum_{j}(b_{i_1,j_1} + b_{j,k}).$$

We see that this requires only n^2 multiplications over the underlying field.
We obtain similar identities for the third and fourth rows- thus we can finally obtain our algorithm.

$$\begin{aligned}\sum_{i,j,k:i+j+k=0 \mod 2} & (a_{i,j} + a_{k_1,i_1})(b_{j,k} + b_{i_1,j_1})(c_{k,i} + c_{j_1,k_1}) & - \\ & \sum_{i,k} a_{k_1,i_1} \sum_{j:i+j+k=0 \mod 2} (b_{i_1,j_1} + b_{j,k})c_{k,i} & - \\ & \sum_{i,j} a_{i,j}b_{i_1,j_1} \sum_{k:i+j+k=0 \mod 2} (c_{k,i} + c_{j_1,k_1}) & - \\ & \sum_{k,j} \sum_{i:i+j+k=0 \mod 2} (a_{i,j} + a_{k_1,i_1})b_{j,k}c_{j_1,k_1} & \\ & = \sum_{i,j,k} a_{i,j}b_{j,k}c_{k,i}. & \end{aligned}$$

The total number of field multiplications is $n^3/2 + 3n^2$, if we were to find the trace of the product of three $n \times n$ matrices. Using the previous theorem, we get that, on setting $n = 34$, this gives $\omega \leq 2.84953$. Clearly, this algorithm does not beat Strassen's, but the techniques used here are used by Pan later on to derive more significant improvements.
We call terms such as $(a_{i,j} + a_{k_1,i_1})(b_{j,k} + b_{i_1,j_1})(c_{k,i} + c_{j_1,k_1})$ *aggregates* and the action of collecting together "undesirable" terms *uniting.*
We need not sum over all i, j, k such that $i + j + k$ is even: indeed Pan uses a more complex subset of $\{0, .., n - 1\}^3$ to obtain his best result using this method. The algorithm he finds is too long to reproduce here: it can be found in the appendices. Pan assumes that we intend to multiply to $n \times n$ matrices, where $n = 2s$ is even. Summing over the set

$$S^1 = \{(i, j, k), 0 \leq 1 \leq j < k \leq s - 1\} \cup \{(i, j, k), 0 \leq k < j \leq i \leq s - 1\}$$

we can obtain the trace of the product of three matrices ABC in $\frac{n^3-4n}{3} + 6n^2$ operations. Setting $n = 70$, we get that $\omega \leq \log_{70} 143640 = 2.79512$, less than the Strassen algorithm.

Related to this trilinear aggregating is the idea of *degeneration*: though it mainly relies on less complex algorithms and does not give exact answers. Pan ([21] , section 10) speculates that it would take a different approach as to the construction of algorithms to find an exact algorithm (that is, one that does not rely on an approximation) which will yield $\omega << 2.5$.

1.7 Border Rank and Degeneration

In 1979, Bini et al. [3] discovered that one could obtain algorithms which required fewer scalar multiplications to compute, at the cost of being only an (arbitrarily close) approximation of the "correct" result. Thus, the concept of *Border Rank* was introduced.
We start with the trilinear form

$$\sum_{i=1}^{r} f_{i\lambda}(A)g_{i\lambda}(B)h_{i\lambda}(C) :$$

however, instead of the constants in the functions f_i, g_i, h_i being in the field k, we set them as being in the extension $k[\lambda]$.
Here, $k[\lambda]$ is the set of all polynomials in λ (but we may also add negative powers of λ). The choice of λ will depend on what field we work in, but if $k = \mathbb{R}$, we may simply take λ as being a very small number. We judiciously choose our functions $f_{i\lambda}, g_{i\lambda}, h_{i\lambda}$ so that

$$\sum_{i=1}^{r} f_{i\lambda}(A)g_{i\lambda}(B)h_{i\lambda}(C) = \text{Trace}(ABC) + \lambda G(\lambda)$$

where $G(\lambda)$ is a polynomial in λ. As an example of this, we look at example 2.3 of [23]: we look at the trilinear version (as opposed to the bilinear one contained therein). This provides a method of finding an approximation of multiplication of two 3×3 matrices. It is known that there exists an exact algorithm which has rank 23 (see [15]), but this method can find an approximate one in 21 multiplications:

$$\begin{aligned}
F_1(\lambda) &= (a_{11}+\lambda^2 a_{12})(\lambda^2 b_{11}+b_{21})c_{11}\\
&+ (a_{21}+\lambda^2 a_{22})(\lambda^2 b_{12}+b_{22})c_{22}\\
&+ (a_{31}+\lambda^2 a_{32})(\lambda^2 b_{13}+b_{23})c_{33}\\
&- a_{11}(b_{21}+b_{31})(c_{11}+c_{12}+c_{13})\\
&- a_{21}(b_{22}+b_{32})(c_{21}+c_{22}+c_{23})\\
&- a_{31}(b_{23}+b_{33})(c_{31}+c_{32}+c_{33})\\
&+ (a_{11}+\lambda^2 a_{22})(b_{21}-\lambda b_{12})c_{12}\\
&+ (a_{21}+\lambda^2 a_{12})(b_{22}-\lambda b_{11})c_{21}\\
&+ (a_{11}+\lambda^2 a_{32})(b_{21}-\lambda b_{13})c_{13}\\
&+ (a_{31}+\lambda^2 a_{12})(b_{23}-\lambda b_{11})c_{31}\\
&+ (a_{21}+\lambda^2 a_{32})(b_{22}-\lambda b_{13})c_{23}\\
&+ (a_{31}+\lambda^2 a_{22})(b_{23}-\lambda b_{12})c_{32}\\
&+ (a_{11}+\lambda^2 a_{23})(b_{31}+\lambda b_{12})(c_{12}+\lambda c_{21})\\
&+ (a_{21}+\lambda^2 a_{13})(b_{32}+\lambda b_{11})(c_{21}+\lambda c_{12})\\
&+ (a_{11}+\lambda^2 a_{33})(b_{31}+\lambda b_{13})(c_{13}+\lambda c_{31})\\
&+ (a_{31}+\lambda^2 a_{13})(b_{33}+\lambda b_{12})(c_{31}+\lambda c_{13})\\
&+ (a_{21}+\lambda^2 a_{33})(b_{32}+\lambda b_{13})(c_{23}+\lambda c_{32})\\
&+ (a_{31}+\lambda^2 a_{23})(b_{33}+\lambda b_{12})(c_{32}+\lambda c_{23})\\
&+ (a_{11}+\lambda^2 a_{13})b_{31}(c_{11}-\lambda c_{31}-\lambda c_{21})\\
&+ (a_{21}+\lambda^2 a_{23})b_{32}(c_{22}-\lambda c_{32}-\lambda c_{12})\\
&+ (a_{31}+\lambda^2 a_{33})b_{33}(c_{33}-\lambda c_{13}-\lambda c_{23})\\
&= \lambda^2(\mathrm{Trace}(ABC)+\lambda G(\lambda)).
\end{aligned}$$

Hence it is possible to find an approximation of this matrix multiplication in fewer multiplications than it takes to find the exact answer.
We can still associate this with a tensor t: however, each summand $u_i \otimes v_i \otimes w_i$ is contained in $u_i(\lambda) \otimes v_i(\lambda) \otimes w_i(\lambda)$ where each $u_i \in U^{k[\lambda]} = k[\lambda]^{dim(U)}$, $v_i \in V^{k[\lambda]} =\in k[\lambda]^{dim(V)}$, $w_i \in W^{k[\lambda]} =\in k[\lambda]^{dim(W)}$.

Definition 6. *The set $\langle r\rangle$ is the equivalence class of bilinear maps whose rank is equal to r.*

Definition 7. *We say that t is a* degeneration of order q of $\langle r\rangle$ *iff there exist vectors $u_i(\lambda) \in k[\lambda]^{dim(U)}, v_i(\lambda) \in k[\lambda]^{dim(V)}, w_i(\lambda) \in k[\lambda]^{dim(W)}$ for $1 \le i \le r$ such that*

$$\lambda^{q-1}t + \lambda^q t'(\lambda) = \sum_{i=1}^{r} u_i(\lambda) \otimes v_i(\lambda) \otimes w_i(\lambda)$$

for some $t'(\lambda)$, a tensor of aa polynomial in λ. We write this as $t \trianglelefteq_q \langle r\rangle$.The least number r over all q is called the Border Rank *of t (or of ϕ, if we work in the non-tensorial notation).*

If we consider explicit maps, we say that ϕ is a degeneration of order q of ϕ' iff there is a $\psi : U^{k[\lambda]} \times V^{k[\lambda]} \to W^{k[\lambda]}$ such that

$$\lambda^{q-1}\phi + \lambda^q \psi = \phi'.$$

Here, ϕ' is an algorithm whose entries are polynomials in λ.
We see from above that $Trace(ABC) \trianglelefteq_3 \langle 21 \rangle$.
How does Border Rank relate to the Exact Rank? This was shown by Bini [2]: we rewrite

$$u_i(\lambda) = \sum_{\mu} u_i^{(\mu)} \lambda^{\mu}, v_i(\lambda) = \sum_{\nu} v_i^{(\nu)} \lambda^{\nu}, w_i(\lambda) = \sum_{\lambda} w_i^{(\lambda)} \lambda^{\lambda}.$$

By multiplying out, we obtain that the coefficient of λ^{q-1} is equal to

$$t = \sum_{i=1}^{r} \sum_{\mu,\nu,\lambda} u_i^{(\mu)} \otimes v_i^{(\nu)} \otimes w_i^{(\lambda)}$$

with $\mu + \nu + \lambda = q - 1$ and $\mu, \nu, \lambda > 0$. There are $q(q+1)/2$ such values of μ, ν, λ, so we have that

$$t \trianglelefteq_q \langle r \rangle \Rightarrow R(t) \leq (q(q+1)/2)r \leq q^2 r. \tag{1.7}$$

Since $q \geq 1$ (although if $q = 1$, the extra calculations are somewhat redundant), we find that the Border Rank is always less than or equal to the Exact Rank. For our example, the border rank of 21 and the fact that $q = 3$ implies that the rank is less than or equal to 189. However, we know that it is at most 23: the gains come through the use of recursion.

Theorem 2. *If $t_1 \trianglelefteq_q \langle r_1 \rangle$ and $t_2 \trianglelefteq_{q'} \langle r' \rangle$ then*

$$t_1 \otimes t_2 \trianglelefteq_{q+q'-1} \langle rr' \rangle.$$

We may use this fact to show how it is affected when one raises the tensor t_1 to the Nth power for some $N \in \mathbb{N}$

$$\otimes_{i=1}^{N} t_1 \trianglelefteq_{(q-1)N+1} \langle r^N \rangle.$$

Proof. Taking the products, on the left hand side we have

$$(\lambda^{q-1})(\lambda^{q'-1})t_1 t_2 + \lambda^{q-1}\lambda^{q'}(t_1 t_2'(\lambda)) + \lambda^{q}\lambda^{q'-1}(t_1'(\lambda)t_2) + \lambda^{q+q'} t_1'(\lambda) t_2'(\lambda)$$

which can be written as

$$\lambda^{q+q'-2} t_1 t_2 + \lambda^{q+q'-1}((t_1 t_2'(\lambda) + t_1'(\lambda)t_2)) + \lambda t_1'(\lambda) t_2'(\lambda)).$$

as desired. Using our previous assertion that the rank of the product of two bilinear maps is less than or equal to the product of the ranks, we have that the right hand side has rank less than or equal to $r_1 r_2$, hence the statement follows.
The second statement will follow directly. □

We can now show how our algorithm helps reduce the upper bound of ω. Recall that if $R\langle e, h, l \rangle \leq r$, then $(ehl)^{\omega/3} \leq r$. Using this and the fact that $\langle 3, 3, 3 \rangle \leq 23$, we obtain $\omega \leq log_3(23) = 2.854...$
However, we have that $\langle 3, 3, 3 \rangle \trianglelefteq_3 \langle 21 \rangle$. We raise this expression to the Nth power, to obtain

$$\langle 3^N, 3^N, 3^N \rangle \trianglelefteq_{2N+1} \langle 21^N \rangle.$$

Then, using our estimate of the rank from the border rank, we obtain

$$R(\langle 3^N, 3^N, 3^N \rangle) \leq (2N+1)^2 21^N.$$

Using equation 1.5, we obtain

$$3^{\omega N} \leq (2N+1)^2 21^N$$

which, on taking logs, implies

$$\omega \leq \frac{2\log(2N+1) + N\log(21)}{N\log(3)}.$$

On letting N go to infinity, we get

$$\omega \leq \frac{\log(21)}{\log(3)} = 2.7712...$$

Thus Border Rank does indeed yield improvements on the exact rank, if only asymptotically. Bini et al. [3] showed (though not explicity, [8] provides a more explicit reasoning) that it was the case that one could replace the exact rank in equation 1.5 with the Border Rank:

Theorem 3. *If $\underline{R}(\langle e, h, l \rangle) \leq r$ for positive integers e, h, l, r, then we have $(ehl)^{\omega/3} \leq r$.*

Proof. Since we may symmetrize, we show this for $e = h = l = n$. By definition we have that

$$\langle n, n, n \rangle \trianglelefteq_q \langle r \rangle$$

for some q. We raise this expression to the Nth power. By theorem 2, we have

$$\langle n^N, n^N, n^N \rangle \trianglelefteq_{(q-1)N+1} \langle r^N \rangle.$$

By 1.7 we have

$$R(\langle n^N, n^N, n^N \rangle) \leq ((q-1)N+1)^2 r^N$$

By theorem 1.5

$$n^{\omega N} \leq ((q-1)N+1)^2 r^N,$$

from which letting N grow and taking Nth roots yields the desired result. □

In fact, a lower bound the Border Rank can immediately be obtained by looking at the dimensions of the spaces U, V and W.

Theorem 4. *Let the matrix product $\phi : U \times V \to W$ be a degeneration of order q of ϕ' with border rank $\underline{R}$. Then*

$$\underline{R} \geq \max\{dim(U), dim(V), dim(W)\}.$$

Proof. We have that $\phi \trianglelefteq_q \phi'$, and the border rank of ϕ is $\underline{R}$. Raising ϕ to the Nth power, we find the border rank of ϕ^N is an integer $\underline{R}^N$. Using Theorem 2 and the estimate for obtaining exact rank from border rank, we get that the exact rank of ϕ^N is at most $((q-1)N+1)^2\underline{R}^N$. Since, from Lemma 1, the rank of a bilinear map is greater than the maximum of the sizes of all dimensions, we obtain

$$\max\{dim(U)^N, dim(V)^N, dim(W)^N\} \leq ((q-1)N+1)^2\underline{R}^N.$$

We let N go to infinity and take Nth roots to obtain the desired statement.

□

The Border Rank was shown in [23] to be non-additive (that is $\underline{R}(\phi_1 + \phi_2) \neq \underline{R}(\phi_1) + \underline{R}(\phi_2)$) To demonstrate, we consider the two matrix products $\langle e, 1, l\rangle$ and $\langle 1, h, 1\rangle$ where $h = (e-1)(l-1)$. These matrix products are *disjoint*, that is, they share no variables. We see that we can compute these two matrix products simultaneously in the (trilinear) algorithm

$$\phi = \sum_{i=1}^{e}\sum_{j=1}^{l} a_i b_j c_{j,i} + \sum_{i=1}^{h} X_i Y_i Z$$

where the $a_i, b_i, c_i, X_i, Y_i, Z$ are all indeterminates over k.
From the previous theorem, we see that the border rank of $\langle e, 1, l\rangle$ is at least el and the border rank of $\langle 1, h, 1\rangle$ is at least $(e-1)(l-1)$. We show that there exists an algorithm such that we can compute both of these products simultaneously with a border rank of $el + 1$.

Theorem 5. *Border rank is not additive.*

Proof. Consider the simultaneous computation of the two matrix products above. We rewrite this as

$$\phi = \sum_{i=1}^{e}\sum_{j=1}^{l} a_i b_j c_{j,i} + \sum_{i=1}^{e-1}\sum_{j=1}^{l-1} X_{i,j} Y_{i,j} Z$$

(which will make our algorithm easier notationally- we are still performing the same calculations).
We define

$$\begin{aligned} X_{i,l} &= 0 \\ X_{e,j} &= -\sum_{i=1}^{e-1} X_{i,j} \\ Y_{i,l} &= -\sum_{j=1}^{l-1} Y_{i,j} \\ Y_{e,j} &= 0; \end{aligned}$$

doing so makes notation easier and also enables cancellation. Finally, consider the algorithm

$$\begin{aligned} F' &= \sum_{i=1}^{e}\sum_{j=1}^{l}(a_i + \epsilon X_{i,j})(b_j + \epsilon Y_{i,j})(\epsilon^2 c_{j,i} + Z) \\ &- (\sum_{i=1}^{e} a_i)(\sum_{j=1}^{l} b_j)Z \\ &= \epsilon^2\phi + \epsilon^3 G \end{aligned}$$

for some G. We therefore have that

$$\langle e, 1, l\rangle \oplus \langle 1, h, 1\rangle \trianglelefteq_3 \langle el + 1\rangle.$$

Hence the border rank is not additive. □

We will use this example to derive a bound for ω. We have the *Additivity Conjecture for the Exact Rank* which states that

$$R(\bigoplus_{i=1}^{n}\langle a_i, b_i, c_i\rangle) = \sum_{i=1}^{n} R(\langle a_i, b_i, c_i\rangle)$$

which remains unproven, but which Schönhage (Theorem 6.3 of [23]) uses to derive a bound for ω (which will be later shown to be obtainable without resorting to conjecture).

Theorem 6. *The exponent of matrix multiplication* $\omega \leq 2.548$*, if the additivity conjecture holds.*

Proof. We start by noting that raising $\langle e, 1, l\rangle \oplus \langle 1, h, 1\rangle$ to the Nth power yields

$$\bigoplus_{s=0}^{N}\binom{N}{s} \odot \langle e^s, h^{N-s}, l^s\rangle$$

where the $\odot$ means that we have multiple instances of $\oplus$: that is all the different products that share the same value of s are gathered together. Using properties of border rank as defined above, we have

$$R(\bigoplus_{s=0}^{N}\binom{N}{s} \odot \langle e^s, h^{N-s}, l^s\rangle) \leq (1 + 2N)^2(el + 1)^N.$$

At this point we bring in the additivity conjecture to obtain

$$\sum_{s=0}^{N}\binom{N}{s} R(\langle e^s, h^{N-s}, l^s\rangle) \leq (1 + 2N)^2(el + 1)^N.$$

Using equation 1.5, we obtain

$$\sum_{s=0}^{N}\binom{N}{s} (el)^{s\omega/3}h^{(N-s)\omega/3} \leq (1 + 2N)^2(el + 1)^N,$$

which, on taking sth roots and letting $s \to \infty$, leaves us with

$$el^{\omega/3} + h^{\omega/3} \leq kn + 1.$$

On taking $e = l = 4$ (and hence $h = 9$) we get the desired result. □

However, as previously stated, Schönhage [23] found it possible to obtain the same result without resorting to conjecture: his *Asymptotic Sum Inequality* proved that it was sufficient to know the border rank and the matrix products being simulated in order to derive a value for ω. We state and prove this theorem below.

Theorem 7. *(Schönhage's Asymptotic Sum Inequality) Suppose we have s matrix products $\langle e_i, h_i, l_i\rangle$, and we can evaluate them all simultaneously using an algorithm ϕ whose border rank is r. Then*

$$\sum_{i=1}^{s}(e_i h_i l_i)^{\omega/3} \leq r.$$

Proof. We start by noting that

$$\bigoplus_{i=1}^{s}\langle e_i, h_i, l_i\rangle \trianglelefteq_q \langle r\rangle$$

for some $q \in \mathbb{N}$. We raise this to the Nth power, obtaining

$$(\bigoplus_{i=1}^{s}\langle e_i, h_i, l_i\rangle)^N \trianglelefteq_{(q-1)N+1} \langle r^N\rangle$$

and equation 1.7 implies that

$$R((\bigoplus_{i=1}^{s}\langle e_i, h_i, l_i\rangle)^N)) \leq ((q-1)N+1)^2 r^N$$

Performing the expansion, we get that, where $\mu = (\mu_1, .., \mu_s)$ is a vector such that $\sum_i \mu_i = N$,

$$R(\bigoplus_{\mu}(\binom{N}{\mu} \odot \prod_{i=1}^{s}\langle e_i, h_i, l_i\rangle^{\mu_i})) \leq ((q-1)N+1)^2 r^N$$

from which we can conclude that

$$R(\binom{N}{\mu} \odot \prod_{i=1}^{s}\langle e_i, h_i, l_i\rangle^{\mu_i}) \leq ((q-1)N+1)^2 r^N.$$

Recall that there exists for every $\epsilon > 0$ a constant $c_\epsilon \in \mathbb{N}$ such that for all n

$$R(\langle n, n, n\rangle) \leq c_\epsilon n^{\omega+\epsilon}$$

Therefore we can set a number $P = (\binom{N}{\mu})^{\frac{1}{\omega+\epsilon}}$ so that we get

$$R(\langle P, P, P\rangle) \leq (c_\epsilon \binom{N}{\mu})$$

We use this definition to show that we can perform the multiplication

$$\langle P\prod_{i=1}^{s} e_i^{\mu_i}, P\prod_{i=1}^{s} h_i^{\mu_i}, P\prod_{i=1}^{s} l_i^{\mu_i}\rangle$$

in $\leq c_\epsilon((q-1)N+1)^2 r^N$ operations in k.
For the "U" matrix, we regard the elements as being $\prod_{i=1}^{s} e_i^{\mu_i} \times \prod_{i=1}^{s} e_i^{\mu_i}$ matrices.
For the "V" matrix, we regard the elements as being $\prod_{i=1}^{s} h_i^{\mu_i} \times \prod_{i=1}^{s} h_i^{\mu_i}$ matrices.
Finally, the elements of the resulting "W" matrix will be $\prod_{i=1}^{s} l_i^{\mu_i} \times \prod_{i=1}^{s} l_i^{\mu_i}$ matrices.
Since we showed that multiplication of $P \times P$ matrices requires $c_\epsilon\binom{N}{\mu}$ multiplications in the underlying field, we see that this number of $\prod_{i=1}^{s}\langle e_i, h_i, l_i\rangle^{\mu_i}$ matrix products is

required. Since this number of multiplications of this type can be done in $((q-1)N+1)^2r^N$ operations in k, we have that

$$R(\langle P\prod_{i=1}^{s}e_i^{\mu_i}, P\prod_{i=1}^{s}h_i^{\mu_i}, P\prod_{i=1}^{s}l_i^{\mu_i}\rangle) \leq c_\epsilon((q-1)N+1)^2r^N.$$

Using equation 1.5, we see that

$$(P^3(\prod_{i=1}^{s}e_i^{\mu_i})(\prod_{i=1}^{s}h_i^{\mu_i})(\prod_{i=1}^{s}l_i^{\mu_i}))^{\omega/3} \leq c_\epsilon((q-1)N+1)^2r^N.$$

We multiply both sides by $\binom{N}{\mu}^{\frac{\epsilon}{\omega+\epsilon}}$ and use the facts that $\binom{N}{\mu} \leq s^N$ and that $a/2 \leq \lfloor a \rfloor$ to obtain

$$\binom{N}{\mu}\prod_{i=1}^{s}(e_ih_il_i)^{\omega/3} \leq 2^\omega s^{\frac{N\epsilon}{\omega+\epsilon}}c_\epsilon((q-1)N+1)^2r^N.$$

If we sum all possible distributions of μ, we get

$$(\sum_{i=1}^{s}(e_ih_il_i)^{\omega/3})^N \leq \binom{N+s-1}{s-1}2^\omega s^{\frac{N\epsilon}{\omega+\epsilon}}c_\epsilon((q-1)N+1)^2r^N.$$

Finally, we take Nth roots and let N go to infinity. This gives us the result

$$\sum_{i=1}^{s}(e_ih_il_i)^{\omega/3} \leq s^{\frac{\epsilon}{\omega+\epsilon}}r,$$

which, on letting $\epsilon \to 0$, gives us our desired result. □

So we see that the additivity conjecture need not be proved. Using Schönhage's example:

$$\underline{R}(\langle e,1,l\rangle \oplus \langle 1,h,1\rangle) \leq el+1$$

implies

$$(el)^{\omega/3} + ((e-1)(l-1))^{\omega/3} \leq el+1,$$

which, on setting $e=l=4$, yields $\omega \leq 2.5479$, as required.
This "Asymptotic sum inequality" is used to great advantage by Coppersmith and Winograd [12] to obtain further reductions in ω. We will outline how this was done in the next chapter.
Prior to this, the same authors [11] discovered that they could make greater use of Schönhage's example:

Definition 8. *Let $\mathcal{B}$ be a direct sum of matrix multiplications, and let $\dashv = (f_1,..,f_r,g_1,...,g_r,w_1,..,w_r)$ be an algorithm which can approximately compute (that is, compute the matrix product, with some error λ) $\langle 1,R,1\rangle \oplus \mathcal{B}$. Then $\langle 1,R,1\rangle$ is* isolated *relative to $\dashv$ if*

$$\langle 1,R,1\rangle = \sum_{i=1}^{R}u_iv_i = \sum_{j=1}^{r}c_jf_j(u)g_j(v)$$

for some $\{c_j\}$ in $k(\lambda)$. If all the $c_j \neq 0$ then we say that $\langle 1,R,1\rangle$ is full *relative to $\dashv$*

We then have the following theorem (proof omitted):

Theorem 8. *If $\dashv$ is an algorithm of rank r which can approximately compute $\langle 1, R, 1\rangle \oplus \mathcal{B}$, and $\langle 1, R, 1\rangle$ is full and isolated relative to $\dashv$ then there exists and algorithm $\dashv'$ which can approximately compute $\langle 1, R^*, 1\rangle \oplus \mathcal{B}$ where the rank of $\dashv'$ is equal to the rank of $\dashv$ and $R^* = r - dim(U) - dim(V)$ where U and V are the domain of $\mathcal{B}$.*

The usefulness of this theorem is borne out when we raise the algorithms to the nth tensor power: each time we do so, there will always be a $\langle 1, R^n, 1\rangle$ term which will be full and isolated relative to $\dashv$. We do not uses this theorem for the main results- it is included to show the notion that rFaising algorithms to higher powers may be used to reduce ω. To demonstrate, we start with Schönhage's construction, setting $e = l = 3$. We get that the product

$$\langle 1, 4, 1\rangle \oplus \langle 3, 1, 3\rangle$$

has border rank 10. Hence, using theorem 7, we obtain

$$\omega \leq 2.5938833$$

If we square this construction, we get that the product

$$\langle 1, 16, 1\rangle \oplus 2 \odot \langle 3, 4, 3\rangle \oplus \langle 9, 1, 9\rangle$$

has border rank 100. However, applying theorem 8, we get that the U part of $\mathcal{B}$ has dimension

$$2 \times (3 \times 4) + 9 \times 1 = 33$$

which is the same as the V part.

This means, by theorem 8, that there exists another algorithm of border rank 100 which can compute

$$\langle 1, 100 - 33 - 33 = 34, 1\rangle \oplus 2 \odot \langle 3, 4, 3\rangle \oplus \langle 9, 1, 9\rangle.$$

Using theorem 7, we obtain $\omega \leq 2.5198543$, which is an improvement. We may continue to do this and perform small modifications of this construction to ultimately obtain $\omega \leq 2.4966271$.

Though the methodology is different, Coppersmith and Winograd [12] use this method of creating new algorithms from old to obtain the current best result. We will look at this in the next chapter.

Chapter 2

Coppersmith and Winograd's Algorithms

In this chapter, we introduce a new construction which Strassen [25], [26] uses to improve on his previous algorithm, and introduce the idea of Salem-Spencer sets which Coppersmith and Winograd use to find asymptotically quicker algorithms.

2.1 Direct Sum Decomposition

When considering a bilinear map ϕ (with associated structural tensor t), it will be convenient (in the case where ϕ works out multiple matrix products) to consider the spaces U, V and W themselves as direct sums of smaller spaces. For example, the space U can be written as

$$U = \bigoplus_{i \in I} U_i$$

where $I = \{1, .., m\}$ for some m. We let D represent the decompostion of U, V and W, that is

$$D : U = \bigoplus_{i \in I} U_i, V = \bigoplus_{j \in J} V_j, W = \bigoplus_{l \in L} W_l$$

Now, we have that the *support of* t $\mathsf{supp_D(t)}$ is the set (i, j, l) such that $t_{ijl} \neq 0$, where the $t_{i,j,l}$ are the *components of* t *with respect to* D.
We may choose I such that $|I| \neq dim(U)$ (and analogously for J, K). This has the effect that we may construct an object which has the structure of a matrix product, but whose "multiplications" are in fact, themselves matrix products. We discuss this idea further in the next section.

2.2 $\mathcal{C}$-tensors

In his 1986 paper [25] (put into more detail in [26]), Strassen discovered that a trilinear form needed not be a matrix product in order to be able to compute a fast matrix multiplication algorithm. We acheive this by taking an algorithm for multiplying a small matrix product (for example, the scalar product of a 1 × 2 vector by a 2 × 1 vector, which we shall see below), raising it to the Nth power (in a tensor multiplication sense) and "choosing" a set of independent matrix multiplication

tensors from this (in the sense that there is not a great deal of choice) and applying theorem 7 to it.
Central to this is the idea of $\mathcal{C}$-tensors, which are informally defined as follows:
If b is a matrix multiplication of the form $\langle e,h,l\rangle$, and $\mathcal{C}$ is the set of all matrix products satisfying a particular property (for example, those multiplications $\langle m,n,p\rangle$ with $mnp = q^N$ for some q, N), we have that t is a $\mathcal{C}$-tensor over $\langle e,h,l\rangle$ iff t possesses a direct sum decompostion D such that $\mathsf{supp}_D t \simeq \mathsf{supp}\langle e,h,l\rangle$ and such that all $D-$ *components* of t are isomorphic to matrix tensors in $\mathcal{C}$. From this definition, we obtain the following result:

Proposition 3. *Let $e,h,l,m,n,p,q \in \mathbb{N}$, and let $\mathcal{C}$ be the set of matrix products $\langle m,n,p\rangle$ with the property that $mnp = q$. If the tensor t is a $\mathcal{C}$-tensor over $\langle e,h,l\rangle$ then $(ehl)^2 q^\omega \leq \underline{R}(t)^3$.*

The proof requires the following lemma:

Lemma 9. *Let t be an $N \times N$ matrix product. t can approximately (in the λ-sense) of calculate $\lceil 3N^2/4\rceil$ independent scalar products.*

Proof. Represent $\langle N,N,N\rangle$ as

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} x_{i,j}y_{j,k}z_{k,i}.$$

Les $g \ni \mathbb{N}$ and multiply each variable by an appropriate integer power of λ to get

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}(x_{i,j}\lambda^{i^2+2ij})(y_{j,k}\lambda^{j^2+2j(k-g)})(z_{k,i}\lambda^{(k-g)^2+2(k-g)i})$$
$$= \sum_{i+j+k=g} x_{i,j}y_{j,k}z_{k,i} + O(\lambda)$$

since the exponent of λ

$$i^2+2ij+j^2+2j(k-g)+(k-g)^2+2(k-g)i = (i+j+k-g)^2$$

is zero when $i+j+k=g$ and positive otherwise. We have that any two indices (i,j) uniquely determine the third k, each variable $x_{i,j}$ is involved in at most one product. There are about $\lceil 3N^2/4\rceil$ triples (i,j,k) with $1 \leq i,j,k \leq N$ with $i+j+k=g$, thus proving the assertion. □

We denote the set of independent scalar products provided by a matrix product as Δ, and the set of independent matrix products obtained from raising it to the Nth power as Δ_N.

Proof. Let π denote the cyclic permutation (123), and let $N \in \mathbb{N}$. The tensor $t_N := (t \otimes \pi t \otimes \pi^2 t)^{\otimes N}$ inherits the direct sum decompostion $D_N := (D \otimes \pi D \otimes \pi^2 D)^{\otimes N}$ from D. We have that

$$\begin{aligned}\mathsf{supp}_{D_N} t_N &\simeq (\mathsf{supp}_D t \otimes \mathsf{supp}_{\pi D}\pi t \otimes \mathsf{supp}_{\pi^2 D}\pi^2 t)^N \\ &\simeq \mathsf{supp}\langle (ehl)^N, (ehl)^N, (ehl)^N\rangle\end{aligned}$$

We know that the D-components of t_N are isomorphic to matrix tensors $\langle m,n,p\rangle$ satisfying $mnp = q^{3N}$ The lemma 9 shows that there is a set of independent matrix products of size $\lceil 3(ehl)^{2N}/4\rceil$ which is a degeneration of $\mathsf{supp}_{D_N} t_N$. We use Proposition 15.30 of [8] which implies

$$\bigoplus_{i,j,l\in\Delta_N} t_N(i,j,l) \trianglelefteq t_N$$

and so by the asymptotic sum inequality (theorem 7)we have

$$|\Delta_N| q^{N\omega} \leq \underline{R}(t_N) \leq \underline{R}(t)^{3N}$$

thus by taking Nth roots and letting $N \to \infty$ we get the desired statement.

□

2.3 Strassen's Construction

To demonstrate the statements of the previous section, consider the algorithm from [25]:

$$\begin{aligned}\sum_{i=1}^{q}(x_0^{[2]} + \lambda x_i^{[1]})(y_0^{[1]} + \lambda y_i^{[2]})(z_i\lambda^{-1}) \quad &+ \quad (x_0^{[2]})(y_0^{[1]})(-\sum_{i=1}^{q} z_i\lambda^{-1}) \\ = \sum_{i=1}^{q}(x_i^{[1]} y_0^{[1]} z_i + x_0^{[2]} y_i^{[2]} z_i) + O(\lambda) \qquad &. \end{aligned} \tag{2.1}$$

We see that this is a $\mathcal{C}$-tensor over $\langle 1,2,1\rangle$, where $\mathcal{C}$ is the set of matrix products $\langle m,n,p\rangle$ with $mnp = q$ (the first block is a $\langle q,1,1\rangle$ and the second is a $\langle 1,1,q\rangle$), and that all this can be acheived in $q+1$ non-scalar multiplications. Hence by theorem 7:

$$e = 1, h = 2, l = 1, mnp = q \Rightarrow 2^2 q^\omega \leq (q+1)^3$$

which, when $q = 5$ yields $\omega \simeq 2.4785$.
This is worked out as follows: if we tensor multiply the original algorithm with its cyclic permutations i.e. multiply $\langle 1,1,2\rangle$, $\langle 1,2,1\rangle$ and $\langle 2,1,1\rangle$, we get, in $(q+1)^3$ multiplications, a $\langle 2,2,2\rangle$ matrix product

$$\begin{aligned}\sum_{i,j,k=1}^{q}(x_{i,j,0}^{[1,1]} y_{0,j,k}^{[1,1]} z_{i,0,k}^{[1,1]} \quad &+ \quad x_{i,j,k}^{[2,1]} y_{0,j,k}^{[1,1]} z_{i,0,0}^{[2,1]} \\ +x_{i,j,0}^{[1,1]} y_{0,0,k}^{[1,2]} z_{i,j,k}^{[2,1]} \quad &+ \quad x_{i,j,k}^{[2,1]} y_{0,0,k}^{[1,2]} z_{i,j,0}^{[2,2]} \\ +x_{0,j,0}^{[1,2]} y_{i,j,k}^{[2,1]} z_{i,0,k}^{[1,1]} \quad &+ \quad x_{0,j,k}^{[2,2]} y_{i,j,k}^{[2,1]} z_{i,0,0}^{[1,2]} \\ +x_{0,j,0}^{[1,2]} y_{i,0,k}^{[2,2]} z_{i,j,k}^{[2,1]} \quad &+ \quad x_{0,j,k}^{[2,2]} y_{i,0,k}^{[2,2]} z_{i,j,0}^{[2,2]}) \\ &+ \quad O(\epsilon)\end{aligned}$$

where each block is a smaller matrix product of size $\langle m,n,p\rangle$ where $mnp = q^{3N}$. For example, the sixth block is of size $\langle 1,q^2,q\rangle$:

$$\sum_{i,j,k=1}^{q} x_{0,j,k} y_{i,j,k} z_{i,0,0}$$

where 0 is the I-index, (j,k) is the J-index and i is the K-index.
Take the Nth tensor power of this algorithm. We obtain a $2^N \times 2^N$ matrix product, whose blocks are also matrix products $\langle m,n,p\rangle$ where $mnp = q^{3N}$. Using lemma 9, we find that within these blocks we have $3/4(2^N)^2$ independent matrix products, of size $\langle m,n,p\rangle$ where $mnp = q^{3N}$. Using theorem 7 we get

$$(q+1)^{3N} = (3/4)2^{2N}(q^{3N})^{\tau}$$

which becomes, if we take Nth roots and let N grow,

$$(q+1)^3 = 2^2 q^{3\tau}$$

which, for $q = 5$, yields $\tau \simeq 0.82616667$ which gives $\omega \simeq 2.4785$, which is what we obtained before. Thus, when we encounter a $\mathcal{C}$-tensor, using the proposition will have the same effect as tensoring permutations of the original tensor together.

2.4 Coppersmith and Winograd's Algorithms

The algorithms described by Coppersmith and Winograd [12] use a further relaxation of ground rules: they start with an algorithm that is not a matrix product or even a $\mathcal{C}$-tensor (and hence Strassen's theorem above does not apply). We tensor this algorithm with itself N times for some large N, which yields a number of blocks. From these blocks we can choose a set Δ of independent matrix products, and therefore can use theorem 7 to find an estimate of τ. We use a theorem of Salem and Spencer [22] to prove the existence of such sets.

2.4.1 Salem-Spencer sets

In [22] *Salem-Spencer set* is defined as follows:
A set of integers B is Salem-Spencer if and only if we have that for all $a,b,c \in M$,

$$a + b = 2c \Rightarrow a = b = c.$$

We will find the following theorem of Salem and Spencer (proof found in [22]) useful:

Theorem 10. *Given $\epsilon > 0$, there exists $M_\epsilon \simeq 2^{c/\epsilon^2}$ such that for all $M > M_\epsilon$ there is a Salem-Spencer set B of $M' > M^{1-\epsilon}$ distinct integers $\{b_1, .., b_{M'}\}$ with*

$$0 < b_1 < b_2 < ... < b_{M'} < M/2.$$

Behrend [1] gives a better (i.e.higher) estimate than [22] does for the size of a progression-free set: however, the Salem-Spencer bound is sufficient for our needs.
In this case, we consider only the ring $\mathbb{Z}_M$ of integers mod M, and since elements in the set B satisfy $0 < b_i < M/2$, it still follows that no three form an arithmetic progression:

$$\text{for } b_i, b_j, b_k \in B,\ b_i + b_j = 2b_k \mod M \text{ iff } b_i = b_j = b_k$$

An application

We use this theorem to prove the following lemma.

Lemma 11. *Suppose we have a set S where $|S| = 3N$ and we wish to divide the elements of S into three sets A, B, C by the following rules:*

- $|A| = |B| = |C| = N$
- $A \cap B = A \cap C = B \cap C = \emptyset$.

Let S_{part} be the set of all possible partitions of S satisfying the above conditions. Then, given ϵ, one can find an N_0 such that there is a subset $\Delta = \{\{A_1, B_1, C_1\}, .., \{A_{|\Delta|}, B_{|\Delta|}, C_{|\Delta|}\}\} \subset S_{\text{part}}$ such that, for all partitions $\{A, B, C\} \in \Delta$

- *if partitions $\{A, B, C\}, \{A', B', C\} \in \Delta$, then $A = A'$ and $B = B'$, similarly, if two partitions share a A or B block, then they must be the same partition.*
- *for all $\{A_i, B_i, C_i\} \in \Delta$, if $A_i \cup B_j \cup C_k = S$, then $i = j = k$.*

such that $|\Delta| \geq (\frac{27}{4})^{N(1-\epsilon)}$ for all $N > N_0$.

Properties of Uniform Random Variables

In order to show these (and following) estimates, it is necessary to assert some properties of uniform random variables w_i over $\{0, .., M-1\} \mod M$. By nature of the uniform random variable, $\mathbb{P}(w_i = k) = \frac{1}{M}$ for all $k \in \{0, .., M-1\}$.

Lemma 12. *If $1 < \mu < M$ and $\mu \in \mathbb{N}$ is coprime to M, then $\mu w_i \mod M$ is also a uniform random variable.*

Proof. We have that the probability that $\mu w_i = k \mod M$ is the same as the probability that $w_i = \frac{k}{\mu} \mod M$. Since μ is coprime to M, division by μ is well-defined, and so the probability that $\mu w_i = k \mod M$ is $\frac{1}{M}$ for all k, hence uniformity. □

Lemma 13. *The sum $\mod M$ of N independent uniform random variables over $\{0, .., M-1\}$ is also a uniform random variable over $\{0, .., M-1\}$.*

Proof. First, we show that if w_1 is a random variable (not necessarily uniform) over $\{0, ..M-1\}$, and w_2 is a uniform random variable over $\{0, .., M-1\}$, then $w_1 + w_2 \mod M$ is also a uniform random variable over $\{0, .., M-1\}$. Then

$$\begin{aligned} &\mathbb{P}(w_1 + w_2 = m \in \{0, .., M-1\} \mod M) = \\ &\qquad \sum_k \mathbb{P}(w_1 = k)\mathbb{P}(w_2 = m - k \mod M) \\ &\qquad\qquad = \frac{1}{M} \sum_k \mathbb{P}(w_1 = k) \\ &\qquad\qquad = \frac{1}{M}. \end{aligned}$$

This holds for any m and hence $w_1 + w_2$ is a uniform distribution over $\{0, .., M-1\}$. Further, the random variables w_1 and $w_1 + w_2$ are independent:

Choose a k and an $m \in \{0, .., M-1\}$.

$$\begin{aligned}
\mathbb{P}(w_1 = k \wedge w_1 + w_2 = m \mod M) &= \mathbb{P}(w_1 = k \wedge w_2 = m - k \mod M) \\
&= \mathbb{P}(w_1 = k)\mathbb{P}(w_2 = m - k) \\
&= \mathbb{P}(w_1 = k) \times \frac{1}{M} \\
&= \mathbb{P}(w_1 = k)\mathbb{P}(w_1 + w_2 = m \mod M)
\end{aligned}$$

and hence independence follows.
Since the sum of two independent uniform random variables is also a uniform random variable, we can rewrite

$$\begin{aligned}
& w_1 + w_2 + ... + w_N \mod M \\
= & (w_1 + w_2) + ... + w_N \mod M \\
= & ((w_1 + w_2) + w_3) + ... + w_N \mod M
\end{aligned}$$

and thus the sum of N independent uniform random variables is also a uniform random variable over $\{0, .., M-1\} \mod M$. □

From the two above lemmas, we can thus assert that, if $1 \leq \mu_i \in \mathbb{N} < M$ is coprime to M, and w_i are independent uniform random variables, then

$$\mu_1 w_1 + \mu_2 w_2 + ... + \mu_n w_n$$

is also a uniform random variable over over $\{0, .., M-1\} \mod M$.

Lemma 14. *Let $A = \{a_1, .., a_n\}$ and $B = \{b_1, .., b_n\}$ be vectors of size n, where the entries are either 0,1 or are coprime to M, and there exist i, k with $a_i \neq 0$ and $a_i b_k - a_k b_i$ non-zero and coprime to M. Then the two random variables*

$$\begin{aligned}
b_A &= \sum_{i=1}^{n} a_i w_i \mod M \textit{ and} \\
b_B &= \sum_{j=1}^{n} b_j w_j \mod M
\end{aligned}$$

are independent.

Proof. We wish to assert independence: that is we wish to show that

$$\mathbb{P}(b_A = r \mod M \wedge b_B = s \mod M) = \mathbb{P}(b_A = r)\mathbb{P}(b_B = s).$$

We define a map $T : \mathbb{Z}_M^n \to \mathbb{Z}_M^2$ by $T(w) = (b_A, b_B)$. Then we have that b_A and b_B are independent if T is surjective. Thus, for any pair (r, s), we have a w' such that $T(w') = (r, s)$. Thus, the probability that $T(w) = (r, s)$ is the probability that $T(w - w') = (0, 0)$, which is a problem independent of w' (see earlier assertions). Thus we need to show that for any (r, s), there is a w such that $T(w) = (r, s)$.
Let a_i be an a_i such that $a_i b_k - a_k b_i$ is non-zero and coprime to M for some k. Define

$$
\begin{aligned}
U &= a_i w_i + \sum_{j \neq i} a_j w_j \mod M (= b_A) \\
V &= \sum_{j \neq i} (a_i b_j - a_j b_i) w_j \mod M.
\end{aligned}
$$

We see that $b_A = r$ and $b_B = s$ when $U = r$ and $V = a_i s - b_i r$. Since there is a k such that $(a_i b_k - a_k b_i)$ is not 0 and coprime to M, we may choose w_i and w_k to make U and V equal to our desired values, irrespective of the other values of w_j. Therefore, $b(A)$ and $b(B)$ are independent.

□

Lemma 15. *Let $A = \{a_1, .., a_n\}, B = \{b_1, .., b_n\}, C = \{c_1, .., c_n\}$ be linearly independent vectors of size n, where the entries are either 0,1 or are coprime to M and $\sum_i a_i \neq 0 \mod M$, $\sum_i b_i \mod M = \sum_i c_i \mod M$. We choose M so that possible values of $a_i b_k - b_i a_k$ are coprime to M. Then the three random variables*

$$
\begin{aligned}
b_A &= \sum_{i=1}^{n} a_i w_i \mod M, \\
b_B &= w_0 + \sum_{j=1}^{n} b_j w_j \mod M \textit{ and} \\
b_C &= w_0 + \sum_{k=1}^{n} c_k w_k \mod M
\end{aligned}
$$

are independent.

Proof. We have that the three variables are independent if the function $T(w) = (b_A, b_B, b_C)$ is surjective. If M is prime, then $\mathbb{Z}_M$ is a field, and since the three vectors are linearly independent, the image of T will span $\mathbb{Z}_M^3$, so we have independence.
Since we know that these three vectors are pairwise independent, we show that they are all independent by considering whether b_A is independent of the *pair* b_B, b_C. If we set $\mu_i = B_i - C_i \mod M$, it is enough to show that there is a w with $\sum_i a_i w_i = r \mod M$ and $\sum_i \mu_i w_i = s \mod M$. We have $\sum_i a_i \neq 0 \mod M$ and $\sum_i \mu_i = 0 \mod M$. This means that we may find i, k with $a_i \mu_k - a_k \mu_i \neq 0$, which, from previous assertions, we know to mean that b_A and $b_B - b_C$ are independent, since it will be coprime to M.

□

We use the theorem of Salem and Spencer to prove our assertion.

Proof. (of Lemma 11) We begin by noting that there are $\binom{3N}{N}\binom{2N}{N}$ elements in S_{part}. Choose $M = 2\binom{2N}{N} + 1$. Using the Salem-Spencer theorem, we know that, given $\epsilon > 0$ if we choose large enough N (and hence, by definition, M) there exists a Salem-Spencer set H such that $|H| > M^{1-\epsilon}$. We thus choose such a H. We select $|S| + 1$ random integers $w_j, 0 \leq j \leq |S|$ such that they are selected uniformly at random from $\{0, .., M - 1\}$.

If we enumerate the elements of S as $\{s_1, .., s_{|S|}\}$, we can define the following functions on the subsets A, B, C of S:

$$b_A(A) = \sum_{s_i \in A}^{3N} w_i \mod M$$

$$b_B(B) = w_0 + \sum_{s_i \in B}^{3N} w_i \mod M$$

$$b_C(C) = \frac{1}{2}(w_0 + \sum_{s_i \in A \cup B} w_i) \mod M.$$

It is immediately clear that for all partitions, $b_A(A) + b_B(B) = 2b_C(C)$. However, we desire only a set of partitions that satisfy the criteria stated in the lemma. It is here that we use the theorem of Salem and Spencer. We take $S = 3N$ with N large, and define M in terms of N. So we take N large enough such that $M > M_\epsilon$. According to theorem 10, there exists a Salem-Spencer set H with size $|H| > M^{1-\epsilon}$. If a block A,B or C does not map into an element $h \in H$ then we remove all partitions containing that block.

Since $b_A(A), b_B(B), b_C(C) \in H$ in all partitions we have left, and $b_A(A) + b_B(B) = 2b_C(C)$ for all partitions, it must follow from the Salem-Spencer theorem that for all remaining partitions $b_A(A) = b_B(B) = b_C(C)$.

We wish to apply the above criteria to all the remaining sets. We note that if two remaining blocks share an A, B or C block, then they will have the same value of $b_A(A)$. It will also follow that if one can choose A, B, C from three remaining partitions to form a different partition, then this different partition will also remain, and the three original partitions will all share a common value of $h \in H$. Hence we draw up $|H| > M^{1-\epsilon}$ lists, each of which contains triples which map to $h \in H$. We then eliminate an appropriate number of triples in each list in order to satisfy the criteria. Finally, we add up the number of remaining triples in order to get our final answer.

We recall the assertions previously made about uniform random variables. Since $b_A(A)$ and $b_B(B)$ are linear sums of independent random variables, they themselves are random variables over the same space. They are also independent: each has associated with it a vector of length N, but due to no two blocks in the same partition having the same vector, we must have that, due to the third lemma, these two random variables are indeed independent. Thus we have that for a given $h \in H$, the probability that $b_A(A) = b_B(B) = h$ is $\frac{1}{M^2}$.

Hence the expected number of partitions such that $b_A(A) = b_B(B) = h$ is

$$\frac{1}{M^2}\binom{3N}{N}\binom{2N}{N}.$$

We obtain $|H| > M^{1-\epsilon}$ lists (one for each element of B), each of which we expect to contain $\frac{1}{M^2}\binom{3N}{N}\binom{2N}{N}$ different partitions. Removing partitions with blocks that do not map into H ensures that any remaining blocks will satisfy this criterion.

Now, we must ensure that no two partitions share A, B or C. We know that there are

$$\frac{1}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)$$

unordered pairs sharing A. We also know that $b_A(A), b_B(B)$ are independent within a partition. It remains to show that if two blocks share an A-block, then $b_A(A), b_B(B)$ and $b_B(B')$ are independent: again this arises because they will all have different blocks associated with them, and the fourth lemma states that $b_A(A)$, $b_B(B)$ and $b_B(B')$ will therefore be independent. Therefore we have that the expected number of pairs such that $b_A A = b_B B = b_B B' = h$ (where B' is the B-block from a different partition sharing A) is

$$\frac{1}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)M^{-3},$$

similarly for those sharing B. The expected number of pairs sharing C is worked out slightly differently, since b_C is worked out by what is not in C rather than what is in it: a block C will be included if it is in a partition with an A and a B such that $b_A(A) = b_B = b \in B$. Two triples will share C if $A \cup B$ is equal in both. Hence,a block C will be shared by a pair if $A \cup B = A' \cup B'$ and $b_A(A) = b_B(B) = b_{A'} = b_{B'}$. There are $\binom{2N}{N}$ triples containing a particular block C, and since $b_A(A)$and $b_B(B)$ are independent, we have the probability that this triple makes it into one of our lists is $\frac{1}{M^2}$. We have that $A \cup B = A' \cup B'$, so we must have that $b_A(A') + b_B(B') = 2h \mod M \in H$. There are $\binom{2N}{N} - 1$ other ways of arranging A' and B'. By the above lemmas, B' is independent of both A and B, so therefore the probability that $\{A', B', C\}$ is in our new list as well as $\{A, B, C\}$ is $\frac{1}{M}$. Therefore, we get the same estimate as above.

If two triples $\{A, B, C\}, \{A, B', C'\}$ share an A-block, we remove all those triples containing B. This not only eliminates the pair $\{A, B, C\}, \{A, B', C'\}$, but also any pairs containing this block B. Since if we remove L blocks containing B, we remove at least $\binom{L}{2} + 1$ pairs containing B. Since $\binom{L}{2} + 1 \geq L$, we have that if we remove at least $\frac{1}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)M^{-3}$ blocks (and hence at least as many sets), we will remove at least as many pairs of triples.

Vitally, using this method, if we have $\{A, B, C\}$,$\{A', B', C'\}$, $\{A'', B'', C''\}$, $\{A, B', C''\}$ left in our list (the latter is guaranteed to be in the list if the first three are), then eliminating all blocks containing B, for example, will ensure that the last criterion is fulfilled.

Eliminating the required blocks yields that the number of triples remaining is at least

$$\begin{aligned}\frac{1}{M^2}\binom{3N}{N}\binom{2N}{N} - \frac{3}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)M^{-3} \\ \geq \frac{1}{4}\binom{3N}{N}\binom{2N}{N}M^{-2}\end{aligned}$$

and hence the expected total number of partitions remaining, by the Salem-Spencer theorem, is at least

$$\frac{1}{4}M^{1-\epsilon}\binom{3N}{N}\binom{2N}{N}M^{-2}.$$

This means that there is a choice of w_i such that more than this number of triples remain: fix this choice of w_i and we get that for each value of $h \in H$ (of which, for large enough M given ϵ, there are more than $M^{1-\epsilon}$ by the Salem-Spencer theorem), there are at least

$$\frac{1}{4}\binom{3N}{N}\binom{2N}{N}M^{-2}$$

triples.

Since then we have that, for increasingly large N, the above term becomes

$$\frac{M^{-\epsilon}}{4M}\binom{3N}{N}\binom{2N}{N}$$

It follows from Stirling's formula that

$$\frac{M^{-\epsilon}}{4M}\binom{3N}{N}\binom{2N}{N} \simeq \frac{4^{-N\epsilon}}{4}(\frac{27}{4})^{N}.$$

We find that, for $N > \frac{\log 4}{\epsilon(log(27)-2\log(4)}$ that this is greater than $(\frac{27}{4})^{N(1-\epsilon)}$. □

2.4.2 Coppersmith and Winograd's "Easy" algorithm

An immediate consequence of the previous theorem is that we can use a starting algorithm that is not a matrix product or a $\mathcal{C}$-tensor to obtain an estimate for ω. We start by using $q+2$ multiplications to obtain Coppersmith and Winograd's [12] "Easy" algorithms:

$$\begin{aligned}
&\sum_{i=i}^{q} \lambda^{-2}(x_0^{[0]} + \lambda x_i^{[1]})(y_0^{[0]} + \lambda y_i^{[1]})(z_0^{[0]} + \lambda z_i^{[1]}) \\
-\lambda^{-3}(x_0^{[0]} + \lambda^2 \sum x_i^{[1]})&(y_0^{[0]} + \lambda^2 \sum y_i^{[1]})(z_0^{[0]} + \lambda^2 \sum z_i^{[1]}) \\
&+(\lambda^{-3} - q\lambda^{-2})(x_0^{[0]})(y_0^{[0]})(z_0^{[0]}) \\
&= \sum_{i=1}^{q}(x_0^{[0]} y_i^{[1]} z_i^{[1]} + x_i^{[1]} y_0^{[0]} z_i^{[1]} + x_i^{[1]} y_i^{[1]} z_0^{[0]}) + O(\lambda). \qquad (2.2)
\end{aligned}$$

We see that this does not represent a matrix multiplication or a $\mathcal{C}$-tensor, and therefore Strassen's theorem does not apply. However, each block does denote a matrix product, so we may raise this algorithm to the $3N$th power and set certain blocks to zero, leaving a number of independent matrix products remaining. We will show that this number is large enough to reduce the value of ω.

To use the language of the start of this chapter, we have that

$$t := \sum_{i=1}^{q}(e_{0ii} + e_{i0i} + e_{ii0})$$

The direct sum decomposition is

$$U_0 = V_0 = W_0 = k(1, 0, .., 0)$$
$$U_1 = V_1 = W_1 = \{\xi \in k^{q+1} : \xi_0 = 0\}$$

and the support of t is

$$\mathsf{supp}_D t = \{(0,1,1),(1,0,1),(1,1,0)\}$$

the individual elements of the support correspond to matrix multiplication as follows:

- $t(0,1,1) \simeq \langle 1,1,q\rangle$
- $t(1,0,1) \simeq \langle q,1,1\rangle$
- $t(1,1,0) \simeq \langle 1,q,1\rangle$

Finally, we have that $\underline{R} = q+2$.

Theorem 16. *The exponent of matrix multiplication* $\omega \leq 2.40364$.

Proof. Raise the expression (2.2) to the $3N$th tensor power. We have a number of blocks of the form

$$X_{x_1,x_2,x_3,..,x_{3N}} Y_{y_1,y_2,y_3..,y_{3N}} Z_{z_1,z_2,z_3..,z_{3N}}.$$

Each of these blocks can be represented by a triple of vectors I, J, K, each in $\{0,1\}^{3N}$ where $I_i = 0$ when $x_i = 0$ and $I_i = 1$ when $x_i = \{1,..,q\}$, and analogously for J (with the Y subscripts) and K (with the Z-subscripts). Thus, given I, J, K, we can work out what kind of matrix product we have. After raising the algorithm to the $3N$th power, we are left with a number of blocks, each of which have associated with them a triple of vectors I, J, K. We wish to retain only those matrix products of the form $\langle q^N, q^N, q^N\rangle$. This happens when each of the I, J, K blocks contains exactly N zeroes, and when each of $I_i + J_i + K_i = 2$. We retain these triples and set the others to zero: that is if we have an X-block with a vector I that does not have the desired form, we simply set all the x_i it contains to being zero, and similarly with Y-blocks and Z-blocks.

We are left with $\binom{3N}{N}\binom{2N}{N}$ blocks, all of which are matrix multiplications of the form $\langle q^N, q^N, q^N\rangle$. We now choose a subset Δ of these blocks which has the following conditions

- if we have in Δ two blocks represented by vectors I, J, K and I, J', K', then it must follow that $J = J'$ and $K = K'$, and likewise if two distinct blocks share J or K
- if I, J, K, I', J', K' and I'', J'', K'' are all elements of Δ and I, J', K'' is also a partition then $I = I' = I''$, $J = J' = J''$ and $K = K' = K''$.

This problem is equivalent to the problem described in the Lemma above: instead of dividing objects into three sets the task is to determine which points in the vectors have zeroes in the I, J and K positions respectively.

We set $M = 2\binom{2N}{N} + 1$, construct a Salem-Spencer set B of size greater than $M^{1-\epsilon}$ and choose uniform random variables $w_0, .., w_{3N}$ over $\{0, ..M-1\}$. We now define three functions on I, J, K:

$$b_X(I) = \sum_{i=1}^{3N} I_i w_i \mod M$$

$$b_Y(J) = w_0 + \sum_{i=1}^{3N} J_i w_i \mod M$$

$$b_Z(K) = \frac{1}{2}(w_0 + \sum_{i=1}^{3N}(2 - K_i) w_i).$$

For each triple, the probability that $b_X(I) = b_Y(J) = b \in B$ is $\frac{1}{M^2}$ due to the independence arguments above. Those blocks that do not map into any $b \in B$, we set to zero. We expect to have about

$$\binom{3N}{N}\binom{2N}{N}\frac{1}{M^2}$$

triples remaining for each $b \in B$.
The expected number of pairs that share an X-block is

$$\frac{1}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)M^{-3}$$

and similarly for Y-blocks. The argument for Z-blocks is analogous to that for C-blocks in the lemma above. If two triples share a block, we set to zero one of the other blocks in one of the triples. Again this means that we eliminate at least as many pairs as blocks: we are left with

$$\binom{3N}{N}\binom{2N}{N}\frac{1}{M^2} - \frac{3}{2}\binom{3N}{N}\binom{2N}{N}(\binom{2N}{N} - 1)M^{-3}$$

Thus, the total number of triples remaining is approximately

$$\frac{M^{-\epsilon}}{4M}\binom{3N}{N}\binom{2N}{N}.$$

This "setting to zero" method means that the set Δ of remaining matrix products is in fact a direct sum: none of the matrix products obtained will share any variables (as ensured by the first bullet point criterion) and hence the overall sum will be isomoprhic to a direct sum of matrix multiplications.
Lemma 11 shows that,for large enough N, $|\Delta| \geq (\frac{27}{4})^{N(1-\epsilon)}$, and hence using theorem 7 we have that

$$(q + 2)^{3N} \geq (\frac{27}{4})^{N(1-\epsilon)}q^{3N\tau}.$$

Letting N grow, we get that $|\Delta|$ becomes arbitrarilly close to $\frac{27}{4}^N$ and taking Nth roots we obtain

$$(q + 2)^3 \geq \frac{27}{4}q^{3\tau}$$

and setting $q = 8$ yields $\omega \leq 2.40364$.

□

2.5 More Complicated Algorithms

Having established that it is possible to start with algorithms that are not in themselves matrix products, Coppersmith and Winograd [12] move on to more complicated starting algorithms. If we start with the algorithm

$$\begin{aligned}
&\sum_{i=i}^{q}\mu^{-2}(x_0^{[0]}+\mu x_i^{[1]})(y_0^{[0]}+\mu y_i^{[1]})(z_0^{[0]}+\mu z_i^{[1]})\\
&-\mu^{-3}(x_0^{[0]}+\mu^2\sum x_i^{[1]})(y_0^{[0]}+\mu^2\sum y_i^{[1]})(z_0^{[0]}+\mu^2\sum z_i^{[1]})\\
&+(\mu^{-3}-q\mu^{-2})(x_0^{[0]}+\mu^3 x_{q+1}^{[2]})(y_0^{[0]}+\mu^3 y_{q+1}^{[2]})(z_0^{[0]}+\mu^3 z_{q+1}^{[2]})\\
&=\sum_{i=1}^{q}(x_0^{[0]}y_i^{[1]}z_i^{[1]}+x_i^{[1]}y_0^{[0]}z_i^{[1]}+x_i^{[1]}y_i^{[1]}z_0^{[0]})\\
&+x_0^{[0]}y_0^{[0]}z_{q+1}^{[2]}+x_0^{[0]}y_{q+1}^{[2]}z_0^{[0]}+x_{q+1}^{[2]}y_0^{[0]}z_0^{[0]}+O(\mu)
\end{aligned}$$

Using the language of the start of this chapter, we have

$$t := \sum_{i=1}^{q}(e_{0ii}+e_{i0i}+e_{ii0})+e_{0,0,q+1}+e_{0,q+1,0}+e_{q+1,0,0}$$

The direct sum decomposition is

$$\begin{aligned}
U_0=V_0=W_0 &= k(1,0,..,0)\\
U_1=V_1=W_1 &= \{\xi\in k^{q+2}:\xi_0=0,\xi_{q+2}=0\}\\
U_2=V_2=W_2 &= k(0,..,0,1)
\end{aligned}$$

and the support of t is

$$\mathsf{supp}_D t=\{(0,1,1),(1,0,1),(1,1,0),(0,0,2),(0,2,0),(2,0,0)\}$$

the individual elements of the support correspond to matrix multiplication as follows:

- $t(0,1,1)\simeq\langle 1,1,q\rangle$
- $t(1,0,1)\simeq\langle q,1,1\rangle$
- $t(1,1,0)\simeq\langle 1,q,1\rangle$
- $t(0,0,2)\simeq\langle 1,1,1\rangle$
- $t(0,2,0)\simeq\langle 1,1,1\rangle$
- $t(2,0,0)\simeq\langle 1,1,1\rangle$

As before, we raise the original algorithm to the $3N$th power. Each resulting block can be uniquely identified by a triple of vectors $I,J,K\in\{0,1,2\}^{3N}$. If $I_i=0$ then $x_i=0$, if $I_i=1$ then $x_i\in\{1,..q\}$ and if $I_i=2$ then $x_i=q+1$, and analogously for J (with Y blocks) and K (with Z blocks).

As before, we wish to have the following criteria in our final set of triples. This will ensure the independence of all variables, and hence that the sum of the matrix products is in fact isomoprhic to a direct sum.

- if we have in Δ two blocks represented by vectors I,J,K and I,J',K', then it must follow that $J=J'$ and $K=K'$, and likewise if two distinct blocks share J or K

- if I, J, K, I', J', K' and I'', J'', K'' are all elements of Δ and I, J', K'' is also a partition then $I = I' = I''$, $J = J' = J''$ and $K = K' = K''$.

We let $L = \beta N$ for some $\beta \in [0, 1]$ to be determined later. We retain only those blocks with $N + L$ indices of 0, $2N + 2L$ indices of 1 and L indices of 2, setting others to zero. Thus, the number of triples remaining is

$$\binom{3N}{L, N+L, 2N-2L}\binom{N+L}{L, L, N-L}\binom{2N-2L}{N-L, N-L}.$$

These are different (but not, as yet, independent) matrix products of size $\langle q^{N-L}, q^{N-L}, q^{N-L}\rangle$. Now set

$$M = 6\binom{N+L}{L, L, N-L}\binom{2N-2L}{N-L, N-L} + 1,$$

choose a Salem-Spencer set B and define random variables $w_0, .., w_{3N}$ as before. We need to alter the random arguments slightly: we define

$$b_X(I) = \sum_{i=1}^{3N} I_i w_i \mod M$$
$$b_Y(J) = w_0 + \sum_{j=1}^{3N} J_i w_i \mod M$$
$$b_Z(K) = \frac{1}{2}(w_0 + \sum_{i=1}^{3N}(2 - K_i)w_i) \mod M.$$

We need to draw on the independence statements from earlier : within a partition $b_X(I)$ and $b_Y(J)$ are independent and if two partitions share an X-block, then $b_X(I)$, $b_Y(J)$ and $b_Y(J')$ are independent.

We proceed as before: the expected number of triples that satisfy $b_X(I) = b_Y(J) = b \in B$ is

$$\frac{1}{M^2}\binom{3N}{L, N+L, 2N-2L}\binom{N+L}{L, L, N-L}\binom{2N-2L}{N-L, N-L}.$$

We set any blocks that do not map onto a $b \in B$ to zero.

The expected number of pairs sharing a block i.e. $b_X(I) = b_Y(J) = b_Y(J') = b \in B$ for different J,J' is

$$\frac{1}{M^3}\binom{3N}{L, N+L, 2N-2L}(\binom{N+L}{L, L, N-L} \times$$
$$\times\binom{2N-2L}{N-L, N-L})(\binom{N+L}{L, L, N-L}\binom{2N-2L}{N-L, N-L} - 1)$$

and similarly for those sharing J. The K case is slightly different, but the same probabilities arise (as above) and we get the same result. As before, eliminating this number of blocks will eliminate at least this number of pairs.

We thus have that the expected number of blocks remaining in each list is at least

$$\frac{1}{36M^2}\binom{3N}{L, N+L, 2N-2L}\binom{N+L}{L, L, N-L}\binom{2N-2L}{N-L, N-L}.$$

Since for large M, $|B| > M^{1-\epsilon}$ we have that the total expected number of remaining blocks is at least

$$\frac{M^{-\epsilon}}{36}\binom{3N}{L, N+L, 2N-2L}$$

possible blocks. Hence, the remaining blocks form a direct sum of matrix products. Thus, theorem 7 states that

$$(q+2)^{3N} \geq \frac{M^{-\epsilon}}{36}\binom{3N}{L, N+L, 2N-2L}q^{3(N-L)\tau}.$$

Using Stirling's Formula,letting ϵ go to zero, taking Nth roots and letting N grow, we get

$$(q+2)^3 \geq \frac{27}{\beta^{\beta}(1+\beta)^{(1+\beta)}(2-2\beta)^{(2-2\beta)}}q^{3(1-\beta)\tau}.$$

Finally, setting $q = 6, \beta = 0.048$ we find that $\omega \leq 2.38719$.

2.6 Coupling the w_i

In this section, we use the same starting algorithm as before, but this time, for all the w_i, we have that $w_{2j-1} = w_{2j}$. The consequence of this is that we can square the starting algorithm and regard that as our starting algorithm instead. Here, we need to introduce the concept of "value" ,as is done in [12] and what it achieves.
We suppose that we have a trilinear form A. The "value" of A is obtained as follows: if A is a matrix product $\langle m, n, p\rangle$then the "value" is simply $(mnp)^{\tau}$;
or else if A is not a matrix product we must use a more complex method. If π is the cyclic permutation of variables x, y, z in A, then we tensor the permutations of x, y, z together to obtain

$$(A \otimes \pi A \otimes \pi^2 A).$$

We then raise this expression to the Nth power, which yields a number of (not necessarily independent) matrix products. As before, we set individual blocks to zero (this eliminating a number of these matrix products) so that we are left with a number of independent matrix products $\langle m_h, n_h, p_h\rangle$.
The value $V_{\tau}(A)$ of A is the limit as $N \to \infty$ of the supremum of

$$(\sum_h (m_h n_h p_h)^{\tau})^{1/3N}$$

over all possible values of m_h, n_h, p_h.
If we let $V_{\tau,N}$ be the supremum of the above function at N (rather than letting $N \to \infty$), then a later theorem will show that

$$V_{\tau,kN} \geq V_{\tau,N}$$

(since $V_{\tau,kN}$ relates to raising the algorithm previously raised to the Nth power to the kNth power).
If $M \geq N$ then $V_{\tau,M}^{M} \geq V_{N}^{N}$. Setting $M = kN + r$ for some $k, r \in \mathbb{N}$, we get

$$V_{\tau,M}^{M} \geq V_{\tau,kN}^{kN} \geq V_{\tau,N}^{kN}$$

which implies

$$V_{\tau,M} \geq V_{\tau,N}^{kN/(kN+r)}.$$

Thus

$$V_{\tau,N} \leq \lim_{M\to\infty} \inf V_{\tau,M}.$$

Since $V_{\tau,M}$ is bounded above, this shows that $V_\tau(A)$ is a limit point.
In the calculations in this and subsequent chapters, the "value" is a two-fold expression which highlights both the *size* of Matrix Multiplications and the *number* thereof.

Properties of Value

Value is supermultiplicative:

$$V_\tau(A \otimes B) \geq V_\tau(A) \times V_\tau(B)$$

Proof. We multiply $A \otimes B \otimes \pi(A \otimes B) \otimes \pi^2(A \otimes B)$ and take the Nth tensor power. We may view the resulting trilinear form as being one similar to $(A \otimes \pi A \otimes \pi^2 A)^N$ whose entries are trilinear forms of type $(B \otimes \pi B \otimes \pi^2 B)^N$.
We suppose that A, when raised to the Nth power, is capable of producing matrix products $\{\langle m_h, n_h, p_h\rangle\}$ and that B, raised to the Nth power is capable of producing matrix products $\{\langle m'_{h'}, n'_{h'}, p'_{h'}\rangle\}$
We then have that we have independent matrix multiplications of the form

$$\langle m_i m'_j, n_i n'_j, p_i p'_j\rangle$$

where $i = 1, .., h$, $j = 1, .., h'$.
The expression

$$\sum_h (m''_h n''_h p''_h)^\tau$$

can be written as

$$\sum_{h,h'} (m_i m'_k n_i n'_k p_i p'_k)^\tau$$

which is equal to

$$(\sum_h \sum_{h'} (m_h n_h p_h)^\tau (m_{h'} n_{h'} p_{h'})^\tau).$$

We get

$$\sum_h (m_h n_h p_h)^\tau \times \sum_{h'} (m_{h'} n_{h'} p_{h'})^\tau.$$

Thus $(V_{\tau,N}(A \otimes B))^{3N} \geq (V_{\tau,N}(A))^{3N} \times (V_{\tau,N}(B))^{3N}$
We need to take $3N$th roots, let $N \to \infty$ and find the supremum. Since the supremum of the product of two sequences is greater than or equal to the product of the two suprema, we may take the suprema of both expressions to obtain

$$V_\tau(A \otimes B) \geq V_\tau(A) \times V_\tau(B)$$

□

A corollary of this is that $V_\tau(A) = V_\tau(\pi(A)) = V_\tau(\pi^2(A))$.
The value is also super-additive:

$$V_\tau(A \oplus B) \geq V_\tau(A) + V_\tau(B).$$

Proof. We start by considering the expression

$$(A \oplus B) \otimes \pi(A \oplus B) \otimes \pi^2(A \oplus B). \tag{2.3}$$

We raise this expression to the Nth power. We obtain expressions of the form

$$A^{k_1} \otimes \pi(A)^{k_2} \otimes \pi^2(A)^{k_3} \otimes B^{k_4} \otimes \pi(B)^{k_5} \otimes \pi^2(B)^{k_6}$$

where $k_1, .., k_6 \geq 0$ and $\sum_i k_i = 3N$. We consider only the terms with
$k_1 = k_2 = k_3 = k$ and $k_4 = k_5 = k_6 = N - k$. For a particular k, there are

$$(\binom{N}{k})^3$$

terms.
These will be isomorphic to $A^k \otimes \pi(A)^k \otimes \pi^2(A)^k \otimes B^{N-k} \otimes \pi(B)^{N-k} \otimes \pi^2(B)^{N-K}$.
If $A^k \otimes \pi(A)^k \otimes \pi^2(A)^k$ is capable of producing a direct sum of matrix products $\{\langle m_h, n_h, p_h \rangle\}$ and $B^{N-k} \otimes \pi(B)^{N-k} \otimes \pi^2(B)^{Nk}$ is capable of producing a direct sum of matrix products $\{\langle m'_{h'}, n'_{h'}, p'_{h'} \rangle\}$, then we obtain that the total for
$A^k \otimes \pi(A)^k \otimes \pi^2(A)^k \otimes B^{N-k} \otimes \pi(B)^{N-k} \otimes \pi^2(B)^{N-K}$ is at least

$$(\sum_h (m_h n_h p_h)^\tau)(\sum_{h'} (m'_{h'} n'_{h'} p'_{h'})^\tau).$$

Thus we have, for a given k, that the value is

$$(\binom{N}{k})^3 (\sum_h (m_h n_h p_h)^\tau)(\sum_{h'} (m'_{h'} n'_{h'} p'_{h'})^\tau).$$

$$= (\binom{N}{k})^3 V_{\tau,k}(A)^{3k} V_{\tau,N-k}(B)^{3(N-k)}$$

We choose k to maximise this expression. We find that

$$k = \frac{V_{\tau,k}(A)N}{V_{\tau,k}(A) + V_{\tau,N-k}(B)}$$

will maximise it at approximately

$$(V_{\tau,k}(A) + V_{\tau,N-k}(B))^{3N}$$

Hence, taking $3N$th roots and letting N (and hence k, $N - k$) go to infinity, we may obtain

$$V_\tau(A \oplus B) \geq V_\tau(A) + V_\tau(B)$$

as required. □

Now, we take the tensor square of the original construction, and we relabel the superscripts. We get that, in $(q+2)^2$ multiplications, we obtain the algorithm

$$\begin{aligned}
&\sum_{i,k=1}^{q} (x^{[0]}_{0,0}y^{[2]}_{i,k}z^{[2]}_{i,k} + x^{[1]}_{0,k}y^{[1]}_{i,0}z^{[2]}_{i,k} + x^{[1]}_{0,k}y^{[2]}_{i,k}z^{[1]}_{i,0} \\
&\quad +x^{[1]}_{i,0}y^{[1]}_{0,k}z^{[2]}_{i,k} + x^{[2]}_{i,k}y^{[0]}_{0,0}z^{[2]}_{i,k} + x^{[2]}_{i,k}y^{[1]}_{0,k}z^{[1]}_{i,0} \\
&\quad +x^{[1]}_{i,0}y^{[2]}_{i,k}z^{[1]}_{0,k} + x^{[2]}_{i,k}y^{[1]}_{i,0}z^{[1]}_{0,k} + x^{[2]}_{i,k}y^{[2]}_{i,k}z^{[0]}_{0,0}) \\
&+\sum_{i=1}^{q}(x^{[2]}_{0,q+1}y^{[1]}_{i,0}z^{[1]}_{i,0} + x^{[0]}_{0,0}y^{[3]}_{i,q+1}z^{[1]}_{i,0} + x^{[0]}_{0,0}y^{[1]}_{i,0}z^{[3]}_{i,q+1} \\
&\quad +x^{[3]}_{i,q+1}y^{[0]}_{0,0}z^{[1]}_{i,0} + x^{[1]}_{i,0}y^{[2]}_{0,q+1}z^{[1]}_{i,0} + x^{[1]}_{i,0}y^{[0]}_{0,0}z^{[3]}_{i,q+1} \\
&\quad +x^{[3]}_{i,q+1}y^{[1]}_{i,0}z^{[0]}_{0,0} + x^{[1]}_{i,0}y^{[3]}_{i,q+1}z^{[0]}_{0,0} + x^{[1]}_{i,0}y^{[1]}_{i,0}z^{[2]}_{0,q+1}) \\
&+\sum_{k=1}^{q}(x^{[2]}_{q+1,0}y^{[1]}_{0,k}z^{[1]}_{0,k} + x^{[3]}_{q+1,k}y^{[0]}_{0,0}z^{[1]}_{0,k} + x^{[3]}_{q+1,k}y^{[1]}_{0,k}z^{[0]}_{0,0} \\
&\quad +x^{[0]}_{0,0}y^{[3]}_{q+1,k}z^{[1]}_{0,k} + x^{[1]}_{0,k}y^{[2]}_{q+1,0}z^{[1]}_{0,k} + x^{[1]}_{0,k}y^{[3]}_{q+1,k}z^{[0]}_{0,0} \\
&\quad +x^{[0]}_{0,0}y^{[1]}_{0,k}z^{[3]}_{q+1,k} + x^{[1]}_{0,k}y^{[0]}_{0,0}z^{[3]}_{q+1,k} + x^{[1]}_{0,k}y^{[1]}_{0,k}z^{[2]}_{q+1,0}) \\
&+(x^{[4]}_{q+1,q+1}y^{[0]}_{0,0}z^{[0]}_{0,0} + x^{[2]}_{q+1,0}y^{[2]}_{0,q+1}z^{[0]}_{0,0} + x^{[2]}_{q+1,0}y^{[0]}_{0,0}z^{[2]}_{0,q+1} + x^{[2]}_{0,q+1}y^{[2]}_{q+1,0}z^{[0]}_{0,0} \\
&+x^{[0]}_{0,0}y^{[4]}_{q+1,q+1}z^{[0]}_{0,0} + x^{[0]}_{0,0}y^{[2]}_{q+1,0}z^{[2]}_{0,q+1} + x^{[2]}_{0,q+1}y^{[0]}_{0,0}z^{[2]}_{q+1,0} + x^{[0]}_{0,0}y^{[0]}_{0,0}z^{[4]}_{q+1,q+1}).
\end{aligned}$$

Again, this is not a matrix product, but we can use theorem 7 to use this algorithm to find a value of ω.
As stated, the variables divide into five blocks:

$$\begin{aligned}
X^{[0]} &= \{x_{0,0}\}, \text{ a scalar} \\
X^{[1]} &= \{x_{i,0}, x_{0,k}\}, \text{ a vector of length } 2q \\
X^{[2]} &= \{x_{q+1,0}, x_{i,k}, x_{0,q+1}\}, \text{ a vector of length } q^2+2 \\
X^{[3]} &= \{x_{q+1,k}, x_{i,q+1}\}, \text{ a vector of length } 2q \\
X^{[4]} &= \{x_{q+1,q+1}\}, \text{ a scalar.}
\end{aligned}$$

We note that this algorithm splits into four different types of trilinear form. Three of these are matrix products: the fourth is not, and we use the notion of "value" to estimate the size of matrix products it can simulate as N gets larger.
The first is simply a scalar multiplication:

$$X^{[0]}Y^{[0]}Z^{[4]} = x^{[0]}_{0,0}y^{[0]}_{0,0}z^{[4]}_{q+1,q+1}$$

Scalar multiplications are matrix products of size $\langle 1,1,1\rangle$ and hence the "value" of this trilinear form is 1. There are three such trilinear forms:

$$X^{[0]}Y^{[0]}Z^{[4]}, X^{[0]}Y^{[4]}Z^{[0]}, X^{[4]}Y^{[0]}Z^{[0]}$$

The second is a multiplication of a vector by a scalar:

$$X^{[0]}Y^{[1]}Z^{[3]} = \sum_{i=1}^{q} x_{0,0}^{[0]} y_{i,0}^{[1]} z_{i,q+1}^{[3]} + \sum_{k=1}^{q} x_{0,0}^{[0]} y_{0,k}^{[1]} z_{q+1,k}^{[3]}$$

that is, a scalar $x_{0,0}$ times a vector $\langle y_{i,0}, y_{0,k} \rangle$. It has size $\langle 1, 1, 2q \rangle$, and hence value $(2q)^{\tau}$. There are six such trilinear forms:

$$X^{[0]}Y^{[1]}Z^{[3]}, X^{[0]}Y^{[3]}Z^{[1]}, X^{[1]}Y^{[0]}Z^{[3]}, X^{[1]}Y^{[3]}Z^{[0]}, X^{[3]}Y^{[1]}Z^{[0]}, X^{[3]}Y^{[0]}Z^{[1]}.$$

The third is a multiplication of a vector of a different size by a scalar:

$$X^{[0]}Y^{[2]}Z^{[2]} = x_{0,0}^{[0]} y_{q+1,0}^{[2]} z_{0,q+1}^{[2]} + x_{0,0}^{[0]} y_{0,q+1}^{[2]} z_{q+1,0}^{[2]} + \sum_{i,k=1}^{q} x_{0,0}^{[0]} y_{i,k}^{[2]} z_{i,k}^{[2]}$$

that is, a scalar $x_{0,0}$ times a vector $\langle y_{q+1,0}, y_{0,q+1} y_{i,k} \rangle$. It has size $\langle 1, 1, (q^2 + 2) \rangle$ and hence has value $(q^2 + 2)^{\tau}$.
There are three such trilinear forms:

$$X^{[0]}Y^{[2]}Z^{[2]}, X^{[2]}Y^{[0]}Z^{[2]}, X^{[0]}Y^{[2]}Z^{[2]}.$$

The final trilinear form

$$\begin{aligned} X^{[1]}Y^{[1]}Z^{[2]} = & \sum_{i=1}^{q} x_{i,0}^{[1]} y_{i,0}^{[1]} z_{0,q+1}^{[2]} + \sum_{k=1}^{q} x_{0,k}^{[1]} y_{0,k}^{[1]} z_{q+1,0}^{[2]} \\ & + \sum_{i,k=1}^{q} x_{i,0}^{[1]} y_{0,k}^{[1]} z_{i,k}^{[2]} + \sum_{i,k=1}^{q} x_{0,k}^{[1]} y_{i,0}^{[1]} z_{i,k}^{[2]} \end{aligned}$$

is not a matrix product at all, and needs to be dealt with differently. This is where the notion of "value" is most useful, and this will be demonstrated in a lemma in the next section, which will show its "value" for $q > 3$ is at least

$$2^{2/3} q^{\tau} (q^{3\tau} + 2)^{1/3}.$$

As discussed above, this number represents both the size and number of independent matrix products that the trilinear form is capable of producing when it is raised to a large power.
We now take the Nth tensor power of the square of the algorithm, with N divisible by 3. Let $\alpha_l, 0 \leq l \leq 4$ be be positive real numbers such that

$$\sum_{l=0}^{4} \alpha_l = 1, \sum_{l=0}^{4} l\alpha_l = 4/3$$

and let A_l be integer approximations to $\alpha_l N$ such that

$$\sum_{l=0}^{4} A_l = N, \sum_{l=0}^{4} lA_l = 4N/3.$$

and retain only those blocks of variables $(X^{[I]}Y^{[J]}Z^{[K]})$ such that

$$\begin{aligned}\#\{i|1 \leq j \leq N, I_i = l\} &= A_l\\ \#\{j|1 \leq j \leq N, J_j = l\} &= A_l\\ \#\{k|1 \leq j \leq N, K_k = l\} &= A_l.\end{aligned}$$

Informally, when we set the α_l we say that we will retain only those X-blocks which contain α_0 entries of 0, α_1 entries of 1 and so on. The additional restriction

$$\sum_{l=0}^{4} l\alpha_l = 4/3$$

arises from the fact that $I_i + J_i + K_i = 4$ for all i, and hence $\sum_{i=1}^{N} I_i + J_i + K_i = 4N$. By definition of the A_l we have that $\sum_{i=1}^{N} I_i = \sum_{l=0}^{4} A_l$, and if we set the same possible partitioning on J and K, it follows that $3\sum_{l=0}^{4} lA_l = 4N$. We discard any blocks that do not have the above restriction.
A particular set $\{A_0, A_1, A_2, A_3, A_4\}$ means that there are

$$\binom{N}{A_0}\binom{N-A_0}{A_1}\binom{N-A_0-A_1}{A_2}\binom{N-A_0-A_1-A_2}{A_3}$$

possible X-blocks remaining. We will represent this (and similar expressions) as

$$\binom{N}{A_0, A_1, A_2, A_3, A_4}$$

for convenience.
We suppose that for a given block $X^{[I]}Y^{[J]}Z^{[K]}$ that $\eta_{l,m,n}$ is the number of times that $(I_i, J_j, K_k) = (l, m, n)$. Some restrictions on the $\eta_{l,m,n}$ immediately arise:

$$\begin{aligned}\sum_{m,n} \eta_{l,m,n} &= A_l\\ \sum_{l,n} \eta_{l,m,n} &= A_m\\ \sum_{l,m} \eta_{l,m,n} &= A_n\\ \sum_{l,m,n} \eta_{l,m,n} &= 1.\end{aligned}$$

It should be noted that these blocks are not in themselves matrix products. Rather, they represent the sum of several independent matrix products. We have that the each block has a "value": this is represented by

$$(2q)^{(\eta_{1,0,3}+\eta_{1,3,0}+\eta_{3,0,1}+\eta_{3,1,0}+\eta_{0,1,3}+\eta_{0,3,1})\tau}(q^2+2)^{(\eta_{2,2,0}+\eta_{2,0,2}+\eta_{0,2,2})\tau} \times \\ \times (2^{2/3}q^{\tau}(q^{3\tau}+2)^{1/3})^{(\eta_{2,1,1}+\eta_{1,2,1}+\eta_{1,1,2})\tau}.$$

This value is the sum $\sum_h (m_h n_h p_h)^{\tau}$ of all the independent matrix products generated by this combination of the $\eta_{l,m,n}$.
For a given partition $\{\eta_{l,m,n}\}$, the number of nonzero triples containing a given block

$X^{[I]}$ is

$$\binom{A_0}{\eta_{0,m,n}}\binom{A_1}{\eta_{1,m,n}}\binom{A_2}{\eta_{2,m,n}}\binom{A_3}{\eta_{3,m,n}}\binom{A_4}{\eta_{4,0,0}}$$

where $\eta_{0,m,n} = \{\eta_{0,0,4}, \eta_{0,4,0}, \eta_{0,1,3}, \eta_{0,3,1}, \eta_{0,2,2}\}$ and analogously for other $\eta_{l,m,n}$. This is equal to

$$\frac{\prod_{0\le l\le 4} A_l!}{\prod_{l+m+n=4} \eta_{l,m,n}!}$$

hence the total number of nonzero triples containing a given X-block is

$$M'' = \sum_{\eta_{l,m,n}} \frac{\prod_{0\le l\le 4} A_l!}{\prod_{l+m+n=4} \eta_{l,m,n}!}.$$

and we say that this summand is maximized at

$$\begin{aligned}
\eta_{l,m,n} &= \gamma_{l,m,n}\\
\gamma_{0,0,4} = \gamma_{0,4,0} = \gamma_{4,0,0} &= \hat{a}\\
\gamma_{0,1,3} = \gamma_{0,3,1} = \gamma_{1,0,3} = \gamma_{1,3,0} = \gamma_{3,0,1} = \gamma_{3,1,0} &= \hat{b}\\
\gamma_{0,2,2} = \gamma_{2,0,2} = \gamma_{2,2,0} &= \hat{c}\\
\gamma_{1,1,2} = \gamma_{1,2,1} = \gamma_{2,1,1} &= \hat{d}\\
A_0 &= 2\hat{a} + 2\hat{b} + \hat{c}\\
A_1 &= 2\hat{b} + 2\hat{d}\\
A_2 &= 2\hat{c} + \hat{d}\\
A_3 &= 2\hat{b}\\
A_4 &= \hat{a}.
\end{aligned}$$

The symmetry arises from the fact that, if we fix k,

$$A_k = \sum_{m,n} \eta_{k,m,n} = \sum_{l,m} \eta_{l,m,k} = \sum_{l,n} \eta_{l,k,n}.$$

If we set $k = 4$ we automatically get $\gamma_{0,0,4} = \gamma_{0,4,0} = \gamma_{4,0,0} = \hat{a}$. For $k = 3$, we also get that $\eta_{3,1,0} = A_3 - \eta_{3,0,1}, \eta_{1,3,0} = A_3 - \eta_{0,3,1}, \eta_{1,0,3} = A_3 - \eta_{0,1,3}$. Using the fact that $\eta_{0,2,2} = A_0 - 2A_6 - \eta_{0,3,1} - \eta_{0,1,3}$ (and similar expressions for $\eta_{2,0,2}$ and $\eta_{2,2,0}$), and $\eta_{2,1,1} = A_2 - \eta_{2,2,0} - \eta_{2,0,2}$ (and similar expressions for $\eta_{1,2,1}$ and $\eta_{1,1,2}$), we find that, given $A_0, .., A_6$, that setting $\eta_{3,0,1}, \eta_{0,3,1}, \eta_{0,1,3}$ will force all the other values of $\eta_{l,m,n}$. If we rewrite all the terms of

$$F = \frac{\prod_{0\le l\le 4} A_l!}{\prod_{l+m+n=4} \eta_{l,m,n}!}$$

in terms of these three variables, approximate using Stirling's formula, and take logs, we find that

$$\begin{aligned}\frac{\partial F}{\partial \eta_{1,0,3}} &= \log(\eta_{1,0,3}) - \log(\eta_{0,1,3}) + \log(\eta_{0,2,2}) - \log(\eta_{2,0,2}) + \\ &+ \log(\eta_{2,1,1}) - \log(\eta_{1,2,1}) \\ \frac{\partial F}{\partial \eta_{0,3,1}} &= \log(\eta_{1,3,0}) - \log(\eta_{0,3,1}) - \log(\eta_{0,2,2}) + \log(\eta_{2,0,2}) - \\ &- \log(\eta_{2,1,1}) + \log(\eta_{1,1,2}) \\ \frac{\partial F}{\partial \eta_{3,0,1}} &= \log(\eta_{3,0,1}) - \log(\eta_{3,1,0}) + \log(\eta_{2,2,0}) - \log(\eta_{2,0,2}) + \\ &+ \log(\eta_{1,1,2}) - \log(\eta_{1,2,1}).\end{aligned}$$

We find that setting $\eta_{3,1,0} = \eta_{3,0,1} = \eta_{1,3,0} = \eta_{0,3,1} = \eta_{1,0,3} = \eta_{0,1,3} = \frac{A_3}{2}$ will cause all the $\eta_{l,m,n}$ of the same type (that is, the ones of which the l, m, n are cyclic permutations of each other) to be equal, and this will set all derivatives to zero as required. The convexity of $\log(F)$ (due to the convexity of $x\log(x)$) will show that this is a maximum.

We proceed via the Salem-Spencer method as before: Set $M = 300030M'' + 1$, construct a Salem-Spencer set B of size greater than $M^{1-\epsilon}$ (as we can, for large enough N) and choose random weights $w_j, 0 \leq j \leq N$ from the set $\{0, .., M-1\}$. We define our functions as before with a slight modification:

$$b_X(I) = \sum_{i=1}^{N} I_i w_i \mod M$$

$$b_Y(J) = w_0 + \sum_{i=1}^{N} J_i w_i \mod M$$

$$b_Z(K) = \frac{1}{2}(w_0 + \sum_{i=1}^{N}(4 - K_i)w_i) \mod M.$$

Since M is odd and coprime to 300030 (and hence division $\mod M$ by all integers up to 15 is well defined), we have that all $I_j w_j$ are independent uniform random variables, and thus by arguments as above, we have that $b_X(I)$ are independent of all possible $b_Y(J)$ and that all $b_Y(J)$ are independent of each other.

We proceed as before: we choose an element from B and compute the expected number of triples such that $b_X(I) = b_Y(J) = b_Z(K) = b \in B$. If any block I or J does not map into any $b \in B$, we set it to zero.

This number is

$$\frac{M''}{M^2}\binom{N}{A_0, A_1, A_2, A_3, A_4}$$

We then find the expected number of pairs with $b_X(I) = b_Y(J) = b_Y(J') = b \in B$: if we find that two triples share an X-block (for example) then we set one of the Y-blocks to zero. This means we subtract about

$$\frac{3(M'')(M''-1)}{M^3}\binom{N}{A_0, A_1, A_2, A_3, A_4}$$

triples. Thus we have more than

$$\frac{M^{1-\epsilon}}{cM}\binom{N}{A_0, A_1, A_2, A_3, A_4}$$

triples remaining, where $c > 0$ is a constant.
The Salem-Spencer method has only accounted for the number of X-blocks containing 0,1 etc. and not individual distributions of $\eta_{l,m,n}$: we must choose values of $\eta_{l,m,n}$ such that we maximise the right hand side of theorem 7.
Recall that the value of each triple of blocks is about

$$\begin{aligned} V_{tot} &= (2q)^{(\eta_{1,0,3}+\eta_{1,3,0}+\eta_{3,0,1}+\eta_{3,1,0}+\eta_{0,1,3}+\eta_{0,3,1})\tau} \times \\ &\times \ (q^2+2)^{(\eta_{2,2,0}+\eta_{2,0,2}+\eta_{0,2,2})\tau}(2^{2/3}q^\tau(q^{3\tau}+2)^{1/3})^{(\eta_{2,1,1}+\eta_{1,2,1}+\eta_{1,1,2})\tau}. \end{aligned}$$

There are $\binom{N}{A_0,A_1,A_2,A_3,A_4}$ blocks remaining. We wish to determine how many of these contain a particular set of $\{\eta_{l,m,n}\}$. If we let $M_{\eta_{l,m,n}}$ be the number of Y-blocks that, given a block X, cause $\{\eta_{l,m,n}\}$ to arise, we have that, since there are

$$\binom{N}{A_0, A_1, A_2, A_3, A_4}$$

blocks in total,that we must have about

$$\binom{N}{A_0, A_1, A_2, A_3, A_4}\frac{M_{\eta_{l,m,n}}}{M''}$$

of a particular type.
We approximate M'' by its largest term M''_{max}(which was shown earlier to be attained when $\gamma_{l,m,n}$ were symmetric) times a polynomial N^3. This arises because, given $A_0, .., A_6$, M'' has three degrees of freedom: these can take values between 0 and N, so $N^3 M''_{max}$ is therefore an approximation of M''. This means that, since the "value" of each block represents the sum of independent matrix products, by theorem 7, we have

$$(q+2)^2 \geq \frac{1}{N^p}\binom{N}{A_0, A_1, A_2, A_3, A_4}\frac{M_{\eta_{l,m,n}}}{M''_{max}}V_{tot}.$$

Since M''_{max} is determined by the values of $A_0, .., A_4$, we must find the maximal value of the numerator given $A_0, .., A_4$. Recalling the symmetry that is forced upon the $\eta_{l,m,n}$, we discover that only the $\eta_{1,0,3}$ terms (and permutations thereof) are not forced by the $A_0, .., A_4$. A similar calculation shows that the numerator is maximised when all these terms are equal: thus the $M_{\eta_{l,m,n}}/M''_{max}$ term becomes 1, and the "value" of each block becomes

$$(2q)^{6\tau\hat{b}}(q^2+2)^{3\tau\hat{c}}[4q^{3\tau}(q^{3\tau}+2)]^{\hat{d}}$$

and the auxiliary equation is therefore

$$(q+2)^{2N} \geq N^{-p}\binom{N}{A_0, A_1, A_2, A_3, A_4}(2q)^{6\tau\hat{b}}(q^2+2)^{3\tau\hat{c}}[4q^{3\tau}(q^{3\tau}+2)]^{\hat{d}}.$$

We choose $\hat{a}, \hat{b}, \hat{c}, \hat{d}$ to maximize the right hand side. If we let $\frac{\hat{a}}{N} = \overline{a}$ and so on, we get, using Striling's Formula, letting N grow and taking Nth roots, that

$$(q+2)^2 = \frac{(2q)^{6\tau\overline{b}}(q^2+2)^{3\tau\overline{c}}[4q^{3\tau}(q^{3\tau}+2)]^{\overline{d}}}{(2\overline{a}+2\overline{b}+\overline{c})^{(2\overline{a}+2\overline{b}+\overline{c})}(2\overline{b}+2\overline{d})^{(2\overline{b}+2\overline{d})}(2\overline{c}-\overline{d})^{(2\overline{c}-\overline{d})}(2\overline{b})^{(2\overline{b})}(\overline{a})^{(\overline{a})}}$$

we find that for

$$\begin{aligned} \overline{a} &= 0.000233 \\ \overline{b} &= 0.012506 \\ \overline{c} &= 0.102546 \\ \overline{d} &= 0.205542 \\ q &= 6 \end{aligned}$$

$\omega < 2.375477$.

2.7 Values and $\mathcal{C}$-tensors

In his paper [26], Strassen describes the notion of $\mathcal{C}$-tensors,and Coppersmith and Winograd [12] make use of them to estimate the "value" of the $X^{[1]}Y^{[1]}Z^{[2]}$ trilinear form. However, we can avoid their use and still get the same "value" by using the fact that the notion of value is symmetrized. We will demonstrate this in the following lemma:

Lemma 17. *The "value" of the trilinear form*

$$\begin{aligned} X^{[1]}Y^{[1]}Z^{[2]} = \sum_{i=1}^{q} x_{i,0}^{[1,0]} y_{i,0}^{[1,0]} z_{0,q+1}^{[0,2]} + \sum_{k=1}^{q} x_{0,k}^{[0,1]} y_{0,k}^{[0,1]} z_{q+1,0}^{[2,0]} \\ + \sum_{i,k=1}^{q} x_{i,0}^{[1,0]} y_{0,k}^{[0,1]} z_{i,k}^{[1,1]} + \sum_{i,k=1}^{q} x_{0,k}^{[0,1]} y_{i,0}^{[1,0]} z_{i,k}^{[1,1]} \end{aligned}$$

is at least

$$(2^{2/3})q^{\tau}(q^{3\tau}+2)^{1/3}.$$

Proof. We have, from the definition of "value" that the value of this trilinear form is at least

$$V_\tau(X^{[1]}Y^{[1]}Z^{[2]}) \geq (\sum_h (m_h n_h p_h)^\tau)^{1/3N}.$$

where the $\langle m_h, n_h, p_h\rangle$ are the independent matrix products generated by raising each of ($X^{[1]}Y^{[1]}Z^{[2]}$, $X^{[1]}Y^{[2]}Z^{[1]}$, $X^{[2]}Y^{[1]}Z^{[1]}$) to the Nth power, and tensor multiplying the resulting blocks together.
We raise each of the cyclic permutations of $X^{[1]}Y^{[1]}Z^{[2]}$ to the $2N$th power, and tensor the resulting trilinear forms together. We have that each block has X, Y and Z blocks which have superscripts in the set

$$\{[1,0],[0,1],[1,1],[0,2],[2,0]\}^{6N}.$$

In the $X^{[1]}Y^{[1]}Z^{[2]}$ and $X^{[1]}Y^{[2]}Z^{[1]}$ sections of the vectors, we retain only those X-blocks that have N indices of $(0,1)$ and N indices of $(1,0)$. In the $X^{[2]}Y^{[1]}Z^{[1]}$

section, we retain those blocks that have L indices of $(2,0)$, L indices of $(0,2)$ and $2G$ indices of $(1,1)$.
In the $X^{[1]}Y^{[1]}Z^{[2]}$ and $X^{[2]}Y^{[1]}Z^{[1]}$ sections of the vectors, we retain only those Y-blocks that have N indices of $(0,1)$ and N indices of $(1,0)$. In the $X^{[1]}Y^{[2]}Z^{[1]}$ section, we retain those blocks that have L indices of $(2,0)$, L indices of $(0,2)$ and $2G$ indices of $(1,1)$.
Finally, in the $X^{[2]}Y^{[1]}Z^{[1]}$ and $X^{[1]}Y^{[2]}Z^{[1]}$ sections of the vectors, we retain only those Z-blocks that have N indices of $(0,1)$ and N indices of $(1,0)$. In the $X^{[1]}Y^{[1]}Z^{[2]}$ section, we retain those blocks that have L indices of $(2,0)$, L indices of $(0,2)$ and $2G$ indices of $(1,1)$.
The number of possible X-blocks is

$$\binom{2N}{N}^2 \binom{2N}{L,L,2G}$$

and the number of blocks containing a given X-block is

$$\binom{N}{L}^4 \binom{2G}{G}$$

which is the same for Y and Z-blocks.
Set $M = 6\binom{2G}{G,G}\binom{N}{G,L}^4 + 1$, and random weights $\{w_{01}, w_{02}, .., w_{6N1}, w_{6N2}\}$. We then define our Salem-Spencer functions:

$$b_X(I) = \sum_{i=1}^{6N}(I_{i1}w_{i1} + I_{i2}w_{i2})$$
$$b_Y(J) = w_0 + \sum_{i=1}^{6N}(J_{i1}w_{i1} + J_{i2}w_{i2})$$
$$b_Z(K) = \frac{1}{2}(\sum_{6N}(4 - K_{i1} - K_{i2})w_i).$$

The expected number of triples such that $b_X(I) = b_Y(J) = b \in B$ is

$$\binom{2N}{N}^2 \binom{2N}{L,L,2G}\binom{N}{L}^4\binom{2G}{G}\frac{1}{M^2}.$$

We set to zero any blocks that do not map into any $b \in B$.
The expected number of pairs sharing an X-block is

$$\binom{2N}{N}^2 \binom{2N}{L,L,2G}\binom{N}{L}^4\binom{2G}{G}(\binom{N}{L}^4\binom{2G}{G} - 1)\frac{1}{M^3}.$$

and similarly for Y and Z-blocks. We set appropriate blocks to zero to eliminate these pairs.
We are left with more than

$$\frac{M^{1-\epsilon}}{M}\binom{2N}{N}^2 \binom{2N}{L,L,2G}$$

triples remaining. Each block represents matrix products of size $\langle m,n,p\rangle$ with $mnp = q^{24G+12L}$. Hence the auxiliary equation is

$$
\begin{aligned}
V_\tau^{6N} &\simeq \binom{2N}{N}^2 \binom{2N}{L,L,2G} q^{(12G+6L)\tau} \\
&\simeq \frac{(2N)^{2N}(2N)^{2N}(2N)^{2N}}{(N)^N(N)^N(N)^N(N)^N(L)^L(L)^L(2G)^{2G}} q^{(12G+6L)\tau} \\
&\simeq \frac{2^{6N}N^{2N}}{(L)^{2L}2^{2G}G^{2G}} q^{(12G+6L)\tau} \\
&\simeq 2^{4N+2L}\binom{N}{L}^2 q^{(12G+6L)\tau} \\
&\simeq 2^{4N}q^{6N\tau}\binom{N}{L}^2 2^{2L}(q^{3\tau})^{2G}.
\end{aligned}
$$

We find that this expression is maximised at

$$L = \frac{2N}{q^{3\tau}+2}, G = \frac{q^{3\tau}N}{q^{3\tau}+2}.$$

which yields the maximum of $2^{4N}q^{6N\tau}(q^{3\tau}+2)^{2N}$. We let N grow and take $6N$th roots and we thus obtain

$$V_\tau \geq 2^{2/3}q^\tau(q^{3\tau}+2)^{1/3}$$

which is the same as the value obtained in [12]

□

The paper mentions two other possibilities: they suggest setting

$$w_{3j-2} = w_{3j-1} = w_{3j},$$

which we will investigate in the next chapter. The other alternative

$$w_{2j-1} = -2w_{2j}$$

yields different trilinear forms, but does not seem to provide a better estimate for ω.

Chapter 3

Extending Coppersmith and Winograd to the Third Tensor Power

Further to Coppersmith and Winograd's "coupling the weights"' ([12],section 8) we now investigate the dependence $w_{3j} = w_{3j-1} = w_{3j-2}$. We use the same notion of "value" V_τ as before.
Now, we start with the tensor cube of Construction (10) in the Coppersmith and Winograd paper [12] (which is too long to produce here). This is neither a matrix product, nor a $\mathcal{C}$-tensor, but we proceed as before, albeit with some differences regarding the final optimization.
Now, we divide the $(q+2)^3$ x-variables into seven blocks:

$X^{[0]} = \{x^{[0]}_{0,0,0}\}$
$X^{[1]} = \{x^{[1]}_{i,0,0}, x^{[1]}_{0,j,0}, x^{[1]}_{0,0,k}\}$
$X^{[2]} = \{x^{[2]}_{i,j,0}, x^{[2]}_{i,0,k}, x^{[2]}_{0,j,k}, x^{[2]}_{q+1,0,0}, x^{[2]}_{0,q+1,0}, x^{[2]}_{0,0,q+1}\}$
$X^{[3]} = \{x^{[3]}_{i,j,k}, x^{[3]}_{q+1,j,0}, x^{[3]}_{q+1,0,k}, x^{[3]}_{i,q+1,0}, x^{[3]}_{i,0,q+1}, x^{[3]}_{0,q+1,k}, x^{[3]}_{0,j,q+1}\}$
$X^{[4]} = \{x^{[4]}_{i,j,q+1}, x^{[4]}_{i,q+1,k}, x^{[4]}_{q+1,j,k}, x^{[4]}_{q+1,q+1,0}, x^{[4]}_{q+1,0,q+1}, x^{[4]}_{0,q+1,q+1}\}$
$X^{[5]} = \{x^{[5]}_{q+1,q+1,k}, x^{[5]}_{q+1,j,q+1}, x^{[5]}_{i,q+1,q+1}\}$
$X^{[6]} = \{x^{[6]}_{q+1,q+1,q+1}\}$

We note that if $X^{[I]}Y^{[J]}Z^{[K]}$ appears in a trilinear form, then $I + J + K = 6$, and hence the trilinear form can be written in block form as

$$\sum_{I+J+K=6} X^{[I]}Y^{[J]}Z^{[K]}.$$

3.1 Trilinear Forms

There are seven types of terms in this trilinear form. The first

$$X^{[0]}Y^{[0]}Z^{[6]} = x^{[0]}_{0,0,0}y^{[0]}_{0,0,0}z^{[6]}_{q+1,q+1,q+1}.$$

is a matrix product of size $\langle 1, 1, 1\rangle$, that is, a scalar $x_{0,0,0}$ times another scalar $y_{0,0,0}$, whose "value" is 1. There are three such terms:

$$X^{[0]}Y^{[0]}Z^{[6]}, X^{[0]}Y^{[6]}Z^{[0]}, X^{[6]}Y^{[0]}Z^{[0]}.$$

The second term

$$\begin{aligned} X^{[0]}Y^{[1]}Z^{[5]} &= \sum_{i=1}^{q} x^{[0]}_{0,0,0}y^{[1]}_{i,0,0}z^{[5]}_{i,q+1,q+1} + \\ &+ \sum_{j=1}^{q} x^{[0]}_{0,0,0}y^{[1]}_{0,j,0}z^{[5]}_{q+1,j,q+1} + \sum_{k=1}^{q} x^{[0]}_{0,0,0}y^{[1]}_{0,0,k}z^{[5]}_{q+1,q+1,k}. \end{aligned}$$

is a matrix product of size $\langle 1, 1, 3q \rangle$, a scalar $x_{0,0,0}$ times a vector

$$\langle y_{i,0,0}, y_{0,j,0}, y_{0,0,k} \rangle$$

whose "value" is $(3q)^\tau$. There are six such terms:

$$\begin{array}{lll} X^{[0]}Y^{[1]}Z^{[5]} & X^{[0]}Y^{[5]}Z^{[1]} & X^{[1]}Y^{[0]}Z^{[5]} \\ X^{[1]}Y^{[5]}Z^{[0]} & X^{[5]}Y^{[1]}Z^{[0]} & X^{[5]}Y^{[0]}Z^{[1]}. \end{array}$$

The third term

$$\begin{aligned} X^{[0]}Y^{[2]}Z^{[4]} &= \sum_{i,j=1}^{q} x^{[0]}_{0,0,0}y^{[2]}_{i,j,0}z^{[4]}_{i,j,q+1} + \sum_{j,k=1}^{q} x^{[0]}_{0,0,0}y^{[2]}_{0,j,k}z^{[4]}_{q+1,j,k} + \\ &+ \sum_{i,k=1}^{q} x^{[0]}_{0,0,0}y^{[2]}_{i,0,k}z^{[4]}_{i,q+1,k} + \\ &+ x^{[0]}_{0,0,0}y^{[2]}_{q+1,0,0}z^{[4]}_{0,q+1,q+1} + x^{[0]}_{0,0,0}y^{[2]}_{0,q+1,0}z^{[4]}_{q+1,0,q+1} + \\ &+ x^{[0]}_{0,0,0}y^{[2]}_{0,0,q+1}z^{[4]}_{q+1,q+1,0} \end{aligned}$$

This is a matrix product of size $\left\langle 1, 1, 3q^2 + 3 \right\rangle$, a scalar $x_{0,0,0}$ times a vector

$$\langle y_{i,j,0}, y_{0,j,k}, y_{i,0,k}, y_{q+1,0,0}, y_{0,q+1,0}, y_{0,0,q+1} \rangle$$

, with "value" $(3q^2 + 3)^\tau$. There are six such terms:

$$\begin{array}{lll} X^{[0]}Y^{[2]}Z^{[4]} & X^{[0]}Y^{[4]}Z^{[2]} & X^{[2]}Y^{[0]}Z^{[4]} \\ X^{[2]}Y^{[4]}Z^{[0]} & X^{[4]}Y^{[2]}Z^{[0]} & X^{[4]}Y^{[0]}Z^{[2]}. \end{array}$$

The fourth term

$$\begin{aligned} X^{[0]}Y^{[3]}Z^{[3]} &= \sum_{i,j,k=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{i,j,k}z^{[3]}_{i,j,k} + \sum_{i=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{i,q+1,0}z^{[3]}_{i,0,q+1} + \\ &+ \sum_{i=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{i,0,q+1}z^{[3]}_{i,q+1,0} + + \sum_{j=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{q+1,j,0}z^{[3]}_{i,q+1,0} + \\ &+ \sum_{j=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{0,j,q+1}z^{[3]}_{q+1,j,0} + \sum_{k=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{0,q+1,k}z^{[3]}_{q+1,0,k} + \\ &+ \sum_{k=1}^{q} x^{[0]}_{0,0,0}y^{[3]}_{q+1,0,k}z^{[3]}_{0,q+1,k} \end{aligned}$$

This is a scalar $x_{0,0,0}$ times a vector of length $q^3 + 6q$

$$\langle y_{i,j,k}, y_{i,q+1,0}, y_{i,0,q+1}, y_{q+1,j,0}, y_{0,j,q+1}, y_{0,q+1,k}, y_{q+1,0,k} \rangle$$

that is a matrix product of size $\langle 1, 1, q^3 + 6q \rangle$, with "value" $(q^3 + 6q)^{\tau}$. There are three such terms:

$$X^{[0]}Y^{[3]}Z^{[3]} \quad X^{[3]}Y^{[0]}Z^{[3]} \quad X^{[3]}Y^{[3]}Z^{[0]}.$$

The remaining trilinear forms are neither matrix products nor are they $\mathcal{C}$-tensors. We deal with them in a similar way to how we dealt with the $X^{[1]}Y^{[1]}Z^{[2]}$ form in the previous chapter: we raise them to a large power and set appropriate blocks to zero in the resulting trilinear form, and we use both the size and number of the remaining triples to come up with an estimate for "value" of these trilinear forms, which will be realised when N is large.
The $X^{[1]}Y^{[1]}Z^{[4]}$ term contains nine blocks:

$$\begin{array}{lll} \displaystyle\sum_{i,j=1}^{q} x^{[1]}_{i,0,0}y^{1}_{0,j,0}z^{[4]}_{i,j,q+1} & +\sum_{i,j=1}^{q} x^{[1]}_{0,j,0}y^{1}_{i,0,0}z^{[4]}_{i,j,q+1} & + \displaystyle\sum_{j,k=1}^{q} x^{[1]}_{0,j,0}y^{1}_{0,0,k}z^{[4]}_{q+1,j,k} + \\ \displaystyle\sum_{j,k=1}^{q} x^{[1]}_{0,0,k}y^{1}_{0,j,0}z^{[4]}_{q+1,j,k} & +\sum_{i,k=1}^{q} x^{[1]}_{i,0,0}y^{1}_{0,0,k}z^{[4]}_{i,q+1,k} & + \displaystyle\sum_{i,k=1}^{q} x^{[1]}_{0,0,k}y^{1}_{i,0,0}z^{[4]}_{i,q+1,k} + \\ \displaystyle\sum_{i=1}^{q} x^{[1]}_{i,0,0}y^{1}_{i,0,0}z^{[4]}_{0,q+1,q+1} & +\sum_{j=1}^{q} x^{[1]}_{0,j,0}y^{1}_{0,j,0}z^{[4]}_{q+1,0,q+1} & + \displaystyle\sum_{k=1}^{q} x^{[1]}_{0,0,k}y^{1}_{0,0,k}z^{[4]}_{q+1,q+1,0} \end{array}$$

and it will be shown in lemma 18 that the "value" is at least $3q^{\tau}(1 + q^{3\tau})^{1/3}$. There are three such terms:

$$X^{[1]}Y^{[1]}Z^{[4]} \quad X^{[4]}Y^{[1]}Z^{[1]} \quad X^{[1]}Y^{[4]}Z^{[1]}.$$

The $X^{[1]}Y^{[2]}Z^{[3]}$ term contains fifteen blocks:

$$\begin{array}{lll}
\sum_{i,j,k=1}^{q} x^{[1]}_{i,0,0}y^{[2]}_{0,j,k}z^{[3]}_{i,j,k} & +\textstyle\sum_{i,j,k=1}^{q} x^{[1]}_{0,j,0}y^{[2]}_{i,0,k}z^{[3]}_{i,j,k} & +\sum_{i,j,k=1}^{q} x^{[1]}_{0,0,k}y^{[2]}_{i,j,0}z^{[3]}_{i,j,k} + \\
\sum_{i,j=1}^{q} x^{[1]}_{0,j,0}y^{[2]}_{i,j,0}z^{[3]}_{i,0,q+1} & +\textstyle\sum_{i,j=1}^{q} x^{[1]}_{i,0,0}y^{[2]}_{i,j,0}z^{[3]}_{i,0,q+1} & +\sum_{j,k=1}^{q} x^{[1]}_{0,j,0}y^{[2]}_{0,j,k}z^{[3]}_{q+1,0,k} + \\
\sum_{j,k=1}^{q} x^{[1]}_{0,0,k}y^{[2]}_{0,j,k}z^{[3]}_{q+1,j,0} & +\textstyle\sum_{i,k=1}^{q} x^{[1]}_{0,0,k}y^{[2]}_{i,0,k}z^{[3]}_{i,q+1,0} & +\sum_{i,k=1}^{q} x^{[1]}_{i,0,0}y^{[2]}_{i,0,k}z^{[3]}_{0,q+1,k} + \\
\sum_{i=1}^{q} x^{[1]}_{i,0,0}y^{[2]}_{0,q+1,0}z^{[3]}_{i,0,q+1} & +\textstyle\sum_{i=1}^{q} x^{[1]}_{i,0,0}y^{[2]}_{0,0,q+1}z^{[3]}_{i,q+1,0} & +\sum_{j=1}^{q} x^{[1]}_{0,j,0}y^{[2]}_{q+1,0,0}z^{[3]}_{0,j,q+1} + \\
\sum_{j=1}^{q} x^{[1]}_{0,j,0}y^{[2]}_{0,0,q+1}z^{[3]}_{q+1,j,0} & +\textstyle\sum_{k=1}^{q} x^{[1]}_{0,0,k}y^{[2]}_{0,q+1,0}z^{[3]}_{q+1,0,k} & +\sum_{k=1}^{q} x^{[1]}_{0,0,k}y^{[2]}_{q+1,0,0}z^{[3]}_{0,q+1,k}.
\end{array}$$

and there are six such terms:

$$\begin{array}{ccc}
X^{[1]}Y^{[2]}Z^{[3]} & X^{[1]}Y^{[3]}Z^{[2]} & X^{[2]}Y^{[1]}Z^{[3]} \\
X^{[2]}Y^{[3]}Z^{[1]} & X^{[3]}Y^{[1]}Z^{[2]} & X^{[3]}Y^{[2]}Z^{[1]}.
\end{array}$$

We will show in lemma 19 that the "value“ of this trilinear form is at least $3^{2/3}q^{\tau}(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3}$.

The final $X^{[2]}Y^{[2]}Z^{[2]}$ term contains twenty-one blocks, and there is only one such term (since the only superscript is 2 in this case, superscripts are omitted):

$$\begin{array}{c}
\sum_{i,j,k=1}^{q} x_{i,j,0}y_{0,j,k}z_{i,0,k} + \sum_{i,j,k=1}^{q} x_{i,j,0}y_{i,0,k}z_{0,j,k} + \sum_{i,j,k=1}^{q} x_{0,j,k}y_{i,j,0}z_{i,0,k} + \\
\sum_{i,j,k=1}^{q} x_{i,0,k}y_{i,j,0}z_{0,j,k} + \sum_{i,j,k=1}^{q} x_{0,j,k}y_{i,0,k}z_{i,j,0} + \sum_{i,j,k=1}^{q} x_{i,0,k}y_{0,j,k}z_{i,j,0} + \\
\sum_{i,j=1}^{q} x_{i,j,0}y_{i,j,0}z_{0,0,q+1} + \sum_{i,j=1}^{q} x_{i,j,0}y_{0,0,q+1}z_{i,j,0} + \sum_{i,j=1}^{q} x_{0,0,q+1}y_{i,j,0}z_{i,j,0} + \\
\sum_{i,k=1}^{q} x_{0,q+1,0}y_{i,0,k}z_{i,0,k} + \sum_{i,k=1}^{q} x_{i,0,k}y_{0,q+1,0}z_{i,0,k} + \sum_{i,k=1}^{q} x_{i,0,k}y_{i,0,k}z_{0,q+1,0} + \\
\sum_{j,k=1}^{q} x_{q+1,0,0}y_{0,j,k}z_{0,j,k} + \sum_{j,k=1}^{q} x_{0,j,k}y_{q+1,0,0}z_{0,j,k} + \sum_{j,k=1}^{q} x_{0,j,k}y_{0,j,k}z_{q+1,0,0} + \\
x_{q+1,0,0}y_{0,q+1,0}z_{0,0,q+1} + x_{q+1,0,0}y_{0,0,q+1}z_{0,q+1,0} + x_{0,q+1,0}y_{q+1,0,0}z_{0,0,q+1} + \\
x_{0,q+1,0}y_{0,0,q+1}z_{q+1,0,0} + x_{0,0,q+1}y_{q+1,0,0}z_{0,q+1,0} + x_{0,0,q+1}y_{0,q+1,0}z_{q+1,0,0}.
\end{array}$$

We will show in lemma 20 that the "value“ is at least $3(1+q^{3\tau})$.

3.2 Raising the Algorithm to the Third Tensor Power

Take the Nth tensor power of the cube of construction 10 in Coppermsith and Winograd, where N is divisible by 3. Let α_l, $0 \leq l \leq 6$, be positive real numbers such that

$$\sum_{l=0}^{6} \alpha_l = 1, \sum_{l=0}^{6} l\alpha_l = 2.$$

Let A_l be integer approximations to $\alpha_l N$ such that

$$\sum_{l=0}^{6} A_l = N, \sum_{l=0}^{6} lA_l = 2N.$$

Retain only those blocks of variables such that

$$\#\{j|1 \leq j \leq N, I_j = l\} = A_l,$$

setting the others to zero, where as before I_j picks out the jth index position.
Let M'' be the number of nonzero triples $(X^{[I]}, Y^{[J]}, Z^{[K]})$ containing a given block $X^{[I]}$. We have

$$M'' = \sum_{\{\eta_{l,m,n}\}} \frac{\prod_{0\leq l\leq 6} A_l!}{\prod_{l+m+n=6} \eta_{l,m,n}!}$$

where $\{\eta_{l,m,n}\}$ range over partitions of N such that

$$\sum_{m,n} \eta_{l,m,n} = A_l, \sum_{l,n} \eta_{l,m,n} = A_m, \sum_{l,m} \eta_{l,m,n} = A_n$$

and the only nonzero values of $\eta_{l,m,n}$ occur with $l + m + n = 6$, $0 \leq l, m, n \leq 6$. We wish to approximate M'' with its largest term times a polynomial N^p for some p. We first demonstrate that at the maximal term, all the $\eta_{l,m,n}$ are symmetric, i.e. that $\eta_{l,m,n} = \eta_{l,n,m} = \eta_{m,l,n} = \eta_{m,n,l} = \eta_{n,m,l} = \eta_{n,l,m}$ for all l, m, n.
We write M'' as

$$M'' = \binom{A_0}{\eta_{0,0,6}, \eta_{0,6,0}, \eta_{0,1,5}, \eta_{0,5,1}, \eta_{0,4,2}, \eta_{0,2,4}, \eta_{0,3,3}} \binom{A_1}{\eta_{1,0,4}, \dots} \times \dots$$

then we consider that the number of triples containing a given Y block is

$$M''_Y = \binom{A_0}{\eta_{0,0,6}, \eta_{6,0,0}, \eta_{1,0,5}, \eta_{5,0,1}, \eta_{4,0,2}, \eta_{2,0,4}, \eta_{3,0,3}} \binom{A_1}{\eta_{0,1,5}, \dots} \times \dots$$

and the number of triples containing a Z-block is

$$M''_Z = \binom{A_0}{\eta_{0,6,0}, \eta_{6,0,0}, \eta_{1,5,0}, \eta_{5,1,0}, \eta_{4,2,0}, \eta_{2,4,0}, \eta_{3,3,0}} \binom{A_1}{\eta_{4,0,1}, \dots} \times \dots$$

Now, we suppose that we relabel the variables in M'' as $x_1, ..., x_n$ where n is the number of variables in M''. We discover that each variable in $\{x_1, .., x_n\}$ can be relabelled in such a way so as to create M'', M''_Y and M''_Z. Hence a stationary point in any of these will also be a stationary point for the other two. We wish for the $\eta_{l,m,n}$ terms in M'', M''_Y and M''_Z to be equal. Since it appears in three different places in the three functions, it must follow that the x_i terms corresponding to these places must all be equal.It will then follow that all the permutations of $\eta_{l,m,n}$ will be equal and so symmetry is proved.
We say that the summand is maximised at

$$\eta_{l,m,n} = \gamma_{l,m,n}$$

$\gamma_{0,0,6} = \gamma_{0,6,0} = \gamma_{6,0,0} = \hat{a}$
$\gamma_{0,1,5} = \gamma_{0,5,1} = \gamma_{1,0,5} = \gamma_{1,5,0} = \gamma_{5,1,0} = \gamma_{5,0,1} = \hat{b}$
$\gamma_{0,2,4} = \gamma_{0,4,2} = \gamma_{2,0,4} = \gamma_{2,4,0} = \gamma_{4,2,0} = \gamma_{4,0,2} = \hat{c}$
$\gamma_{0,3,3} = \gamma_{3,3,0} = \gamma_{3,0,3} = \hat{d}$
$\gamma_{1,1,4} = \gamma_{1,4,1} = \gamma_{4,1,1} = \hat{e}$
$\gamma_{1,2,3} = \gamma_{1,3,2} = \gamma_{2,1,3} = \gamma_{2,3,1} = \gamma_{3,2,1} = \gamma_{3,1,2} = \hat{f}$
$\gamma_{2,2,2} = \hat{g}$
$A_0 = 2\hat{a} + 2\hat{b} + 2\hat{c} + \hat{d}$
$A_1 = 2\hat{b} + 2\hat{e} + 2\hat{f}$
$A_2 = 2\hat{c} + 2\hat{f} + \hat{g}$
$A_3 = 2\hat{d} + 2\hat{f}$
$A_4 = 2\hat{c} + \hat{e}$
$A_5 = 2\hat{b}$
$A_6 = \hat{a}.$

M'' will be approximated by its largest term, times a polynomial N^p. Set $M = cM'' + 1$. The constant c will be suitably chosen in order to enable the independence arguments of chapter 2 to hold. Construct the salem-spencer set B. Choose random weights w_j, $0 \le j \le N$. The c in the definition of M ensures that the independence arguments from the previous chapter follow (it will ensure that M is coprime to any values that the variables may take). We define functions as follows:

$$b_X(I) \equiv \sum_{j=1}^{N} I_j w_j \mod M$$

$$b_Y(J) \equiv w_0 + \sum_{j=1}^{N} J_j w_j \mod M$$

$$b_Z(K) \equiv \left[w_0 + \sum_{j=1}^{N} (6 - K_j) w_j \right] /2 \mod M.$$

We have (due to independence) that the expected number of triples such that $b_X(I) = b_Y(J) = b \in B$ is

$$\frac{1}{M^2} \binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6} M'' \\ \simeq \frac{1}{cM} \binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}$$

and the expected number of pairs of triples sharing an X-block such that $b_X(I) = b_Y(J) = b_Y(J') = b \in B$ is

$$\frac{1}{M^3} \binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6} M''(M'' - 1) \\ \le \frac{1}{c^2 M} \binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}.$$

Thus the expected number of compatible triples remaining such that $b_X(I) = b_Y(J) = b \in B$ and independence is not comprimised is at least, for a constant $c' > 0$

$$\frac{1}{c'M}\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}.$$

The Salem-Spencer theorem states that for large enough N, the size of the Salem-Spencer set B is greater than $M^{1-\epsilon}$. Therefore, the total number of remaining triples is greater than

$$\frac{3M^{1-\epsilon}}{100}\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}.$$

These, however, include all possible values of $\eta_{l,m,n}$. In order to maximise the auxiliary equation at the end, it is necessary to pick out an appropriate distribution of the $\eta_{l,m,n}$. Previously, due to the 1:1 relationship with the A_l and the $\gamma_{l,m,n}$, this was not necessary, but now choosing the A_l does not force all of the $\eta_{l,m,n}$- only $\hat{b}$ and $\hat{a}$ are forced, with the remainder being subject to a single degree of freedom.
Thus, since we have approximately

$$\frac{M^{1-\epsilon}}{M}\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}$$

a fraction of these will have a particular distribution $\{\eta_{l,m,n}\}$- we wish to find the distribution of $\{\eta_{l,m,n}\}$ which maximizes the overall auxiliary equation. By similar arguments to before, it can be shown that, for all $\eta_{l,m,n}$ with $l + m + n = 6$, the overall auxiliary equation is maximised at $\{\gamma'_{l,m,n}\}$ with

$$\gamma'_{0,2,4} = \gamma'_{2,4,0} = \gamma'_{4,0,2} = \gamma'_{0,4,2} = \gamma'_{2,0,4} = \gamma'_{4,2,0} = \hat{c}'$$

and so on (we remember that the $\hat{a}$ and $\hat{b}$ values are forced by the choice of A_l so we need not redefine them here). The proportion of remaining triples that has this distribution will be

$$\frac{1}{M''}\binom{A_0}{\hat{a}', \hat{a}', \hat{b}', \hat{b}', \hat{c}', \hat{c}', \hat{d}'} \times ... = \frac{M'}{M''}$$

Therefore, we have more than

$$\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}\frac{M'}{M''}$$

blocks with “value”

$$\begin{aligned} (3q)^{6\tau\hat{b}}(3q^2+3)^{6\tau\hat{c}'}(q^3+6q)^{3\tau\hat{d}'} \times \\ \times 3q^{\tau}(1+q^{3\tau})^{\hat{e}'}(3^{2/3}q^{\tau}(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3})^{6\hat{f}'}3(1+q^{3\tau})^{\hat{g}'}. \end{aligned}$$

remaining and our auxiliary equation is therefore

$$\begin{aligned}(q+2)^{3N} &\geq \frac{M^{1-\epsilon}}{M}\binom{N}{A_0,A_1,A_2,A_3,A_4,A_5,A_6}\frac{M'}{M''}\times\\ &\times (3q)^{6\tau\hat{b}}(3q^2+3)^{6\tau\hat{c}'}(q^3+6q)^{3\tau\hat{d}'}\times\\ &\times 3q^{\tau}(1+q^{3\tau})^{\hat{e}'}(3^{2/3}q^{\tau}(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3})^{6\hat{f}'}3(1+q^{3\tau})^{\hat{g}'}.\end{aligned}$$

We find that if we allow $\hat{c}$ to vary, the maximal term of M'', given $A_0, ..A_6$ occurs when $\hat{c} = \frac{\hat{d}\hat{e}\hat{g}}{\hat{f}^2}$. Thus, we choose values $\hat{b}, \hat{d}, \hat{e}, \hat{f}, \hat{g}$, forcing the value of $\hat{a}$ (due to the restriction that $3\hat{a}+6\hat{b}+6\hat{c}+3\hat{d}+3\hat{e}+6\hat{f}+\hat{g}=N$) , $\hat{c}$ and all the A_l.Multiplying this by a polynomial N^p gives us an approximation for M''. We call this choice M_{max}. The auxiliary equation becomes

$$\begin{aligned}(q+2)^{3N} &\geq N^{-p}\binom{N}{A_0,A_1,A_2,A_3,A_4,A_5,A_6}\frac{M'}{M_{\mathsf{max}}}\times\\ &\times (3q)^{6\tau\hat{b}}(3q^2+3)^{6\tau\hat{c}'}(q^3+6q)^{3\tau\hat{d}'}\times\\ &\times 3q^{\tau}(1+q^{3\tau})^{\hat{e}'}(3^{2/3}q^{\tau}(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3})^{6\hat{f}'}3(1+q^{3\tau})^{\hat{g}'}.\end{aligned}$$

Using these values for A_l, we then allow $\hat{c}'$ to vary in the numerator, allowing us to choose a $\hat{c}'$ which maximises the numerator. Thus, the right hand side is a function of six variables. We set $\overline{a} = \hat{a}/N$ and similarly for $\hat{b}, \hat{c}$, etc.. We also set $\overline{c}' = \hat{c}'/N$ and similarly for $\hat{d}', \hat{e}'$ etc. Taking Nth roots and letting N grow, we find that the right hand side is maximised at

$$\begin{aligned}\overline{a} &= 0.0000515\\ \overline{b} &= 0.000319\\ \overline{c} &= 0.006855\\ \overline{d} &= 0.03722085\\ \overline{e} &= 0.009132\\ \overline{f} &= 0.101608\\ \overline{g} &= 0.208233618\\ \overline{c}' &= 0.0069335\\ \overline{d}' &= 0.03705988500\\ \overline{e}' &= 0.0089710\\ \overline{f}' &= 0.101769\\ \overline{g}' &= 0.2077550\\ q &= 6.\end{aligned}$$

setting $\omega = 2.375477$ gives the right hand side a value of 507.85, so this algorithm will not provide an improvement for the value of ω.
An alternative way of doing this is to regard the blocks as a product of the original algorithm times its square- this takes advantage of the improvements already gained from squaring the algorithm, and increases the right hand side slightly, but not enough to reduce the value of ω.
As an example, we take the $X^{[1]}Y^{[1]}Z^{[4]}$ term. We may regard this as

$$X^0Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[1]}Z^{[0]} \oplus X^1Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[0]}Z^{[2]} \oplus X^0Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[0]}Z^{[1]}.$$

The first block has "value" q^τ, the second has "value" $2^{2/3}q^\tau(1+q^{3\tau})^{1/3}$, and the third has "value" $(2q^2)^\tau$.
We use a similar technique as below to evaluate the overall "values" of these trilinear forms. On setting $q = 6$, $\tau = 2.375477/3$, we obtain the numerical value of this trilinear form as being greater than 54.57, while the numerical value of the one shown below is greater than 51.46. We find that all the trilinear forms with non-zero terms have values greater than their equivalents in the method we show below: however, as stated, this does not affect the value of ω. However, this method will become useful in the next chapter when we investigate the dependency

$$w_{4j-3} = w_{4j-2} = w_{4j-1} = w_{4j}.$$

Why does this algorithm not provide a better estimate for ω? It is probable that the gains from squaring Coppersmith and Winograd's original algorithm are negated by introducing the weaker original algorithm again. Thus, taking the square of the square (i.e. the fourth tensor power) of the original algorithm may be more condusive to obtaining a lower estimate of ω since we retain the gains from squaring the original algorithm, and square them again, thus possibly making further gains. We investigate this in the next chapter.

3.3 Finding the Values of the Trilinear Forms

Lemma 18. *The value of the trilinear form (rewritten to demonstrate component parts)*

$$\sum_{i,j=1}^{q} x_{i,0,0}^{[100]}y_{0,j,0}^{[010]}z_{i,j,q+1}^{[112]} + \sum_{i,j=1}^{q} x_{0,j,0}^{[010]}y_{i,0,0}^{[010]}z_{i,j,q+1}^{[112]} + \sum_{i,k=1}^{q} x_{i,0,0}^{[100]}y_{0,0,k}^{[001]}z_{i,q+1,k}^{[121]} +$$
$$+\sum_{i,k=1}^{q} x_{0,0,k}^{[001]}y_{i,0,0}^{[100]}z_{i,q+1,k}^{[121]} + \sum_{j,k=1}^{q} x_{0,j,0}^{[010]}y_{0,0,k}^{[001]}z_{q+1,j,k}^{[211]} + \sum_{j,k=1}^{q} x_{0,0,k}^{[001]}y_{0,j,0}^{[010]}z_{q+1,j,k}^{[211]} +$$
$$+\sum_{i=1}^{q} x_{i,0,0}^{[100]}y_{i,0,0}^{[100]}z_{0,q+1,q+1}^{[022]} + \sum_{j=1}^{q} x_{0,j,0}^{[010]}y_{0,j,0}^{[010]}z_{q+1,0,q+1}^{[202]} + \sum_{k=1}^{q} x_{0,0,k}^{[001]}y_{0,0,k}^{[001]}z_{q+1,q+1,0}^{[220]}$$

is at least $3q^\tau(1+q^{3\tau})^{\frac{1}{3}}$.

Proof. Take the $3N$th tensor power. Retain only those X-blocks with exactly N indices of [100], N of [010] and N of [001]. Similarly for Y blocks. Retain those Z-blocks with exactly $2L$ indices of $[112], [211]$ and $[121]$ and G of $[022], [202]$ and $[220]$. Now, we do analogous things to 141 and 411 and tensor these together with the 114 case.
Hence the total number of X-blocks (and also Y and Z-blocks) is

$$\binom{3N}{N,N,N}^2\binom{3N}{2L,2L,2L,G,G,G}.$$

The number of nonzero triples containing a given X-block is

$$H = \binom{N}{L, L, G}^6 \binom{2L}{L}^3$$

We set $M = 2H + 1$. We set $9N + 1$ random weights

$$\{w_0, (w_1)_1, (w_1)_2, (w_1)_3, ..(w_{3N})_1, (w_{3N})_2, (w_{3N})_3\},$$

selected uniformly from $\{0, ..M - 1\}$. Each block will have assigned to it three vectors in $\{[100], [010], [001], [211], [121], [112], [022], [202], [220]\}^{3N}$.
Define three hash functions:

$$\begin{aligned} b_X(I) &= \sum_{i=1}^{3N}\sum_{j=1}^{3}(I_i)_j(w_i)_j \mod M \\ b_Y(J) &= w_0 + \sum_{i=1}^{3N}\sum_{j=1}^{3}(J_i)_j(w_i)_j \mod M \\ b_Z(K) &= (w_0 + \sum_{i=1}^{3N}\sum_{j=1}^{3}(2 - (K_i)_j)(w_i)_j)/2 \mod M \end{aligned}$$

where $(I_i)_j$ denotes the jth entry in the ith entry of I, which contains values in $\{0, 1, 2\}$. Due to independence, the expected number of blocks such that $b_X(I) = b_Y(J) = b \in B$ is equal to

$$\binom{3N}{N, N, N}^2 \binom{3N}{2L, 2L, 2L, G, G, G} \binom{N}{L, L, G}^6 \binom{2L}{L}^3 \times \frac{1}{M^2}$$

which is about

$$\binom{3N}{N, N, N}^2 \binom{3N}{2L, 2L, 2L, G, G, G} \times \frac{1}{210M}$$

The expected number of pairs of triples that share an X-block, and that $b_X(I) = b_Y(J) = b_Y(J') = b \in B$ is

$$\binom{3N}{N, N, N}^2 \binom{3N}{2L, 2L, 2L, G, G, G} (\binom{N}{L, L, G}^6 \binom{2L}{L}^3) \times \\ \times (\binom{N}{L, L, G}^6 \binom{2L}{L}^3 - 1) \times \frac{1}{M^3}$$

which is less than

$$\binom{3N}{N, N, N}^2 \binom{3N}{2L, 2L, 2L, G, G, G} \times \frac{1}{210^2 M}.$$

We have that the number remaining is approximately equal to

$$\frac{23}{4900} \binom{3N}{N, N, N}^2 \binom{3N}{2L, 2L, 2L, G, G, G}.$$

Thus the auxiliary equation is

$$\begin{aligned} V_\tau^{9N} &\approx \frac{23}{4900}\binom{3N}{N,N,N}^2\binom{3N}{2L,2L,2L,G,G,G}q^{(36L+9G)\tau} \\ &\approx 3^{9N}\binom{N}{2L}^3 q^{(36L+9G)\tau} \approx 3^{9N}q^{9N\tau}[\binom{N}{2L}q^{6L\tau}]^3 \end{aligned}$$

Setting $\frac{G}{N}=\frac{q^{3\tau}}{1+q^{3\tau}}$ and $\frac{2L}{N}=\frac{1}{1+q^{3\tau}}$ yields

$$V_\tau^{9N} \approx 3^{9N}q^{9N\tau}(1+q^{3\tau})^{3N}$$

and hence taking $9N$th roots yields the required value. □

Lemma 19. *The value of the trilinear form*

$$\begin{aligned} &\sum_{i,j,k=1}^{q} x^{[100]}_{i,0,0}y^{[011]}_{0,j,k}z^{[111]}_{i,j,k} + \sum_{i,j,k=1}^{q} x^{[010]}_{0,j,0}y^{[101]}_{i,0,k}z^{[111]}_{i,j,k} + \sum_{i,j,k=1}^{q} x^{[001]}_{0,0,k}y^{[110]}_{i,j,0}z^{[111]}_{i,j,k} + \\ &+\sum_{i,j=1}^{q} x^{[010]}_{0,j,0}y^{[110]}_{i,j,0}z^{[102]}_{i,0,q+1} + \sum_{i,j=1}^{q} x^{[100]}_{i,0,0}y^{[110]}_{i,j,0}z^{[012]}_{0,j,q+1} + \sum_{j,k=1}^{q} x^{[010]}_{0,j,0}y^{[011]}_{0,j,k}z^{[201]}_{q+1,0,k} + \\ &+\sum_{j,k=1}^{q} x^{[001]}_{0,0,k}y^{[011]}_{0,j,k}z^{[210]}_{q+1,j,0} + \sum_{i,k=1}^{q} x^{[001]}_{0,0,k}y^{[101]}_{i,0,k}z^{[120]}_{i,q+1,0} + \sum_{i,k=1}^{q} x^{[100]}_{i,0,0}y^{[101]}_{i,0,k}z^{[021]}_{0,q+1,k} + \\ &+\sum_{i=1}^{q} x^{[100]}_{i,0,0}y^{[020]}_{0,q+1,0}z^{[102]}_{i,0,q+1} + \sum_{i=1}^{q} x^{[100]}_{i,0,0}y^{[002]}_{0,0,q+1}z^{[120]}_{i,q+1,0} + \sum_{j=1}^{q} x^{[010]}_{0,j,0}y^{[200]}_{q+1,0,0}z^{[012]}_{0,j,q+1} + \\ &+\sum_{j=1}^{q} x^{[010]}_{0,j,0}y^{[002]}_{0,0,q+1}z^{[210]}_{q+1,j,0} + \sum_{k=1}^{q} x^{[001]}_{0,0,k}y^{[020]}_{0,q+1,0}z^{[201]}_{q+1,0,k} + \sum_{k=1}^{q} x^{[001]}_{0,0,k}y^{[200]}_{q+1,0,0}z^{[021]}_{0,q+1,k} \end{aligned}$$

is at least $3^{2/3}q^\tau(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3}$.

Proof. We proceed with a similar method to before. We take the $X^{[1]}Y^{[2]}Z^{[3]}$, $X^{[3]}Y^{[1]}Z^{[2]}$ and $X^{[2]}Y^{[3]}Z^{[1]}$ and raise each to the $6N$th tensor power. For The expression $X^{[1]}Y^{[2]}Z^{[3]}$, we retain only those triples which contain $2N$ instances of each of [100], [010] and [001]. For $X^{[3]}Y^{[1]}Z^{[2]}$, we retain only those triples whose X-blocks contain $6K$ instances of [111] and $G+L$ instances each of [120], [102],[210],[201],[012] and [012] respectively. Finally, for $X^{[2]}Y^{[3]}Z^{[1]}$, we retain only those triples whose X-blocks contain $2L+2K$ instances of each of [101],[110] and [011] and $2G$ instances of each of [200],[020] and [002].
We thus have that $N=G+L+K$, and that, if we choose the number of Y-blocks in each appropriately, we can find that the number and size of matrix products (that is, the product mnp) to be simulated will be the same for all three permutations.
Tensoring these three permutations together, we find that the number of X-blocks in this new algorithm is

$$\begin{aligned} &\binom{6N}{2N,2N,2N}\binom{6N}{2L+2K,2L+2K,2L+2K,2G,2G,2G}\times \\ &\quad\times\binom{6N}{6K,G+L,G+L,G+L,G+L,G+L,G+L}. \end{aligned}$$

and the number of Y-blocks containing a given X-block is

$$M_Y = \binom{2N}{2K, L, L, G, G}^3 \binom{2L+2K}{L, L, 2K}^3 \binom{2G}{G, G}^3 \binom{6K}{2K, 2K, 2K} \binom{G+L}{G, L}^6.$$

We then set $M = 2M_Y + 1$, which ensures that all the independence arguments follow. We then choose uniform random variables $w_0, ..w_{18N}$ over $\{0, .., M-1\}$, construct a Salem-Spencer set B of size close to M and define our three functions on the blocks as above:

$$\begin{aligned} b_X(I) &= \sum_{i=1}^{3N} \sum_{j=1}^{3} (I_i)_j (w_i)_j \mod M \\ b_Y(J) &= w_0 + \sum_{i=1}^{3N} \sum_{j=1}^{3} (J_i)_j (w_i)_j \mod M \\ b_Z(K) &= (w_0 + \sum_{i=1}^{3N} \sum_{j=1}^{3} (2 - (K_i)_j)(w_i)_j)/2 \mod M \end{aligned}$$

After finding the expected number of triples such that $b_X(I) = b_Y(J) = b \in B$ and subtracting expected number of pairs of triples such that share an X-block, Y-block or Z-block which both map into $b \in B$, and collecting the lists for each $b \in B$ together, we find we are left with approximately

$$\binom{6N}{2N, 2N, 2N} \binom{6N}{2L+2K, 2L+2K, 2L+2K, 2G, 2G, 2G} \times \\ \times \binom{6N}{6K, G+L, G+L, G+L, G+L, G+L, G+L}$$

triples remaining. Each of these triples represents a matrix product of size $\langle m, n, p \rangle$ with

$$mnp = (q^3)^{18K} (q^2)^{18L} (q)^{18G} = q^{56K+26L+18G}.$$

Therefore, the "value" V_τ is such that

$$V_\tau^{18} = \binom{6N}{2N, 2N, 2N} \binom{6N}{2L+2K, 2L+2K, 2L+2K, 2G, 2G, 2G} \times \\ \times \binom{6N}{6K, G+L, G+L, G+L, G+L, G+L, G+L} q^{56K+26L+18G}.$$

Using Striling's Formula, which approximates $a!$ by a^a, this is approximately equal to

$$3^{12N} q^{18N} (\binom{N}{L+K, G} q^{3(K+L)\tau})^6 (\binom{N}{G+L, K} q^{3K\tau} 6^{G+L})^6$$

We set $G = \frac{N}{1+q^{3\tau}}$, $L + K = \frac{q^{3\tau}N}{1+q^{3\tau}}$, $K = \frac{6N}{6+q^{3\tau}}$ and $G + L = \frac{q^{3\tau}N}{6+q^{3\tau}}$, and in doing so we obtain that the above expression is approximately equal to

$$3^{12N} q^{18N} (1 + q^{3\tau})^{6N} (6 + q^{3\tau})^{6N}$$

and taking $18N$th roots, we obtain the desired expression.

□

Lemma 20. *Finally, the value of the trilinear form*

$$\begin{array}{lll}
\displaystyle\sum_{i,j,k=1} x_{i,j,0}y_{0,j,k}z_{i,0,k} & +\sum_{i,j,k=1} x_{i,j,0}y_{i,0,k}z_{0,j,k} & +\displaystyle\sum_{i,j,k=1} x_{0,j,k}y_{i,j,0}z_{i,0,k} + \\
\displaystyle\sum_{i,j,k=1} x_{i,0,k}y_{i,j,0}z_{0,j,k} & +\sum_{i,j,k=1} x_{0,j,k}y_{i,0,k}z_{i,j,0} & +\displaystyle\sum_{i,j,k=1} x_{i,0,k}y_{0,j,k}z_{i,j,0} + \\
\displaystyle\sum_{i,j=1} x_{i,j,0}y_{i,j,0}z_{0,0,q+1} & +\sum_{i,j=1} x_{i,j,0}y_{0,0,q+1}z_{i,j,0} & +\displaystyle\sum_{i,j=1} x_{0,0,q+1}y_{i,j,0}z_{i,j,0} + \\
\displaystyle\sum_{i,k=1} x_{0,q+1,0}y_{i,0,k}z_{i,0,k} & +\sum_{i,k=1} x_{i,0,k}y_{0,q+1,0}z_{i,0,k} & +\displaystyle\sum_{i,k=1} x_{i,0,k}y_{i,0,k}z_{0,q+1,0} + \\
\displaystyle\sum_{j,k=1} x_{q+1,0,0}y_{0,j,k}z_{0,j,k} & +\sum_{j,k=1} x_{0,j,k}y_{q+1,0,0}z_{0,j,k} & +\displaystyle\sum_{j,k=1} x_{0,j,k}y_{0,j,k}z_{q+1,0,0} + \\
x_{q+1,0,0}y_{0,q+1,0}z_{0,0,q+1} & +x_{q+1,0,0}y_{0,0,q+1}z_{0,q+1,0} & +x_{0,q+1,0}y_{q+1,0,0}z_{0,0,q+1} + \\
x_{0,q+1,0}y_{0,0,q+1}z_{q+1,0,0} & +x_{0,0,q+1}y_{q+1,0,0}z_{0,q+1,0} & +x_{0,0,q+1}y_{0,q+1,0}z_{q+1,0,0}.
\end{array}$$

is at least $3(1 + q^{3\tau})$.

Proof. Since $X^2Y^2Z^2$ is already a symmetric trilinear form, there is no need to symmetrize. We raise it to the $3N$th tensor power. Each resulting block will have associated with it three vectors in I, J, K in

$$\{[1,1,0],[1,0,1],[0,1,1],[0,0,2],[0,2,0],[2,0,0]\}^{3N}.$$

We set to zero all those X-blocks which do not have α occurences of each of $[1,1,0]$, $[1,0,1]$ and $[0,1,1]$ and β occurences each of $[0,0,2]$, $[0,2,0]$ and $[2,0,0]$. We do the same for Y and Z-blocks.The number of X-blocks is thus

$$\binom{3N}{\alpha,\alpha,\alpha,\beta,\beta,\beta}$$

We then have that the number of blocks containing a given X, Y or Z block is

$$M'' = \sum_{L,G,K} \binom{\alpha}{L,L,G,G}^3 \binom{\beta}{G,K,K}$$

where we sum over all possible distributions of L, G, K. We thus have that
$N = 2L + 3G + 2K$. We set $M = 2M'' + 1$, and define random variables $w_0, .., w_{3N}$ over $\{0, .., M-1\}$. We define our hash functions

$$\begin{aligned}
b_X(I) &= \sum_{i=1}^{3N}\sum_{j=1}^{3}(I_i)_j(w_i)_j \mod M \\
b_Y(J) &= w_0 + \sum_{i=1}^{3N}\sum_{j=1}^{3}(J_i)_j(w_i)_j \mod M \\
b_Z(K) &= (w_0 + \sum_{i=1}^{3N}\sum_{j=1}^{3}(2-(K_i)_j)(w_i)_j)/2 \mod M
\end{aligned}$$

Given M, we construct a Salem-Spencer set B of size greater than $M^{1-\epsilon}$. We set to zero any blocks which do not map onto a $b \in B$. If two triples share an X-block, we set to zero one of the Y-blocks contained in one of these triples (and we perform analogous actions for triples sharing a Y or Z block). We then have about

$$\frac{M^{1-\epsilon}}{M}\binom{3N}{\alpha,\alpha,\alpha,\beta,\beta,\beta}$$

We have that these triples are matrix products of size $\langle m,n,p\rangle$ where $mnp = q^{18L+18G} = q^{9\alpha}$.If we choose a distribution L, G, K, we have that the overall value of the resulting trilinear form is

$$\frac{M^{1-\epsilon}}{M}\binom{3N}{\alpha,\alpha,\alpha,\beta,\beta,\beta}\frac{M_{L,G,K}}{M''}q^{9\alpha\tau}.$$

Since the size of the matrix product is affected only by the values of α and β, we choose L, G, K such that it matches the maximal term of M'', which we also approximate by its largest term, times N. This makes the $\frac{M_{L,G,K}}{M''}$ term approximately equal to $\frac{1}{N}$. If we let N grow and ϵ go to zero, we get that the value is approximately equal to

$$\frac{1}{N}3^{3N}[\binom{N}{\alpha,\beta}q^{3\alpha\tau}]^3.$$

Setting $\alpha = \frac{q^{3\tau}N}{1+q^{3\tau}}$ and $\beta = \frac{N}{1+q^{3\tau}}$ makes this expression approximately equal to

$$\frac{1}{N}3^{3N}[1+q^{3\tau}]^{3N}.$$

Letting N grow and taking $3N$th roots, we obtain the desired estimate. □

Chapter 4

Extending Coppersmith and Winograd to the Fourth Tensor Power

Raising the original Coppermsith and Winograd algorithm [12] to the third power did not yield a reduction in the upper bound for the value of ω. However, it has provided the framework for the method we will now use to derive an improvement.

4.1 Trilinear forms

We proceed by raising the original algorithm to the fourth tensor power, yielding 1296 different blocks. These split into ten different kinds of trilinear forms. In order to reduce the overall number of blocks and to take advantage of the improvements obtained by squaring the original algorithm , we regard this algorithm as the square of the square for particular trilinear forms (as in the $X^{[1]}Y^{[1]}Z^{[2]}$ case).
These trilinear forms are as follows:

$$X^{[0]}Y^{[0]}Z^{[8]} = x^{[0]}_{0,0,0,0}y^{[0]}_{0,0,0,0}z^{[8]}_{q+1,q+1,q+1,q+1}.$$

is a multiplication of two scalars, a matrix product of size $\langle 1, 1, 1\rangle$ whose value is 1.
There are three such terms:

$$X^{[0]}Y^{[0]}Z^{[8]}, X^{[0]}Y^{[8]}Z^{[0]}, X^{[8]}Y^{[0]}Z^{[0]}$$

The second:

$$X^{[0]}Y^{[1]}Z^{[7]} = x^{[0]}_{0,0,0,0}y^{[1]}_{i,0,0,0}z^{[7]}_{i,q+1,q+1,q+1} + x^{[0]}_{0,0,0,0}y^{[1]}_{0,j,0,0}z^{[7]}_{q+1,j,q+1,q+1}$$

$$+x^{[0]}_{0,0,0,0}y^{[1]}_{0,0,k,0}z^{[7]}_{q+1,q+1,k,q+1} + x^{[0]}_{0,0,0,0}y^{[1]}_{0,0,0,l}z^{[7]}_{q+1,q+1,q+1,l}.$$

is a matrix product of size $\langle 1, 1, 4q\rangle$, that is a scalar times a vector

$$\langle y_{i,0,0,0}, y_{0,j,0,0}, y_{0,0,k,0}, y_{0,0,0,l}\rangle$$

Its "value" is $(4q)^\tau$. There are six such terms:

$$X^{[0]}Y^{[1]}Z^{[7]}, X^{[0]}Y^{[7]}Z^{[1]}, X^{[1]}Y^{[0]}Z^{[7]}, X^{[1]}Y^{[7]}Z^{[0]}, X^{[7]}Y^{[1]}Z^{[0]}, X^{[7]}Y^{[0]}Z^{[1]}$$

The trilinear form

$$X^{[0]}Y^{[2]}Z^{[6]}$$

has two parts. The first is of the form

$$x_{0,0,0,0}y_{q+1,0,0,0}z_{0,q+1,q+1,q+1} + x_{0,0,0,0}y_{0,q+1,0,0}z_{q+1,0,q+1,q+1}$$

$$+x_{0,0,0,0}y_{0,0,q+1,0}z_{q+1,q+1,0,q+1} + x_{0,0,0,0}y_{0,0,0,q+1}z_{q+1,q+1,q+1,0}$$

and the second is of the form

$$\sum_{i=1}^{q}\sum_{j=1}^{q} x_{0,0,0,0}y_{i,j,0,0}z_{i,j,q+1,q+1} + \sum_{i=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{i,0,k,0}z_{i,q+1,k,q+1}$$

$$+\sum_{i=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,0,0,l}z_{i,q+1,q+1,l} + \sum_{j=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{0,j,k,0}z_{q+1,j,k,q+1}$$

$$+\sum_{j=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{0,j,0,l}z_{q+1,j,q+1,l} + \sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{0,0,k,l}z_{q+1,q+1,k,l}.$$

Combining these yields a trilinear form which represents a scalar $x_{0,0,0,0}$ times a vector

$$\begin{aligned}\langle \quad & y_{q+1,0,0,0}, y_{0,q+1,0,0}, y_{0,0,q+1,0}, y_{0,0,0,q+1}, \\ & y_{i,j,0,0}, y_{i,0,k,0}, y_{i,0,0,l}, y_{0,j,k,0}, y_{0,j,0,l}, y_{0,0,k,l} \quad \rangle\end{aligned}$$

which is a matrix product of size $\langle 1, 1, (4 + 6q^2)\rangle$ and hence the overall value of this trilinear form is $(4 + 6q^2)^\tau$.

We have six permutations of the blocks in the trilinear form:

$$X^{[0]}Y^{[2]}Z^{[6]}, X^{[0]}Y^{[6]}Z^{[2]}, X^{[2]}Y^{[0]}Z^{[6]}, X^{[2]}Y^{[6]}Z^{[0]}, X^{[6]}Y^{[2]}Z^{[0]}, X^{[6]}Y^{[0]}Z^{[2]}.$$

The trilinear form

$$X^{[0]}Y^{[3]}Z^{[5]}$$

again has two parts.

We have

$$\sum_{i=1}^{q}\sum_{j=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{i,j,k,0}z_{i,j,k,q+1} + \sum_{i=1}^{q}\sum_{j=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,j,0,l}z_{i,j,q+1,l}$$

$$+\sum_{i=1}^{q}\sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,0,k,l}z_{i,q+1,k,l} + \sum_{j=1}^{q}\sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{0,j,k,l}z_{q+1,j,k,l}$$

and

$$\sum_{i=1}^{q} x_{0,0,0,0}y_{i,q+1,0,0}z_{i,0,q+1,q+1} + \sum_{i=1}^{q} x_{0,0,0,0}y_{i,0,q+1,0}z_{i,q+1,0,q+1}$$

$$+\sum_{i=1}^{q} x_{0,0,0,0}y_{i,0,0,q+1}z_{i,q+1,q+1,0} + \sum_{j=1}^{q} x_{0,0,0,0}y_{q+1,j,0,0}z_{0,j,q+1,q+1}$$

$$+\sum_{j=1}^{q} x_{0,0,0,0}y_{0,j,q+1,0}z_{q+1,j,0,q+1} + \sum_{j=1}^{q} x_{0,0,0,0}y_{0,j,0,q+1}z_{q+1,j,q+1,0}$$
$$+\sum_{k=1}^{q} x_{0,0,0,0}y_{q+1,0,k,0}z_{0,q+1,k,q+1} + \sum_{k=1}^{q} x_{0,0,0,0}y_{0,q+1,k,0}z_{q+1,0,k,q+1}$$
$$+\sum_{k=1}^{q} x_{0,0,0,0}y_{0,0,k,q+1}z_{q+1,q+1,k,0} + \sum_{l=1}^{q} x_{0,0,0,0}y_{q+1,0,0,l}z_{0,q+1,q+1,l}$$
$$+\sum_{l=1}^{q} x_{0,0,0,0}y_{0,q+1,0,l}z_{q+1,0,q+1,l} + \sum_{k=1}^{q} x_{0,0,0,0}y_{0,0,q+1,l}z_{q+1,q+1,0,l}.$$

This represents a scalar x_{0000} times a vector

$$\langle y_{i,j,k,0}, y_{i,j,0,l}, y_{i,0,k,l}, y_{0,j,k,l}, y_{i,q+1,0,0}, y_{i,0,q+1,0}, y_{i,0,0,q+1}, y_{q+1,j,0,0}\ ,$$
$$y_{0,j,q+1,0}, y_{0,j,0,q+1}, y_{q+1,0,k,0}, y_{0,q+1,k,0}, y_{0,0,k,q+1}, y_{q+1,0,0,l}, y_{0,q+1,0,l}\ ,\ y_{0,0,q+1,l}\rangle$$

which is a matrix product of size $\langle 1, 1, 4q^3 + 12q\rangle$, and has value $(4q^3 + 12q)^\tau$.
The trilinear form

$$X^{[0]}Y^{[4]}Z^{[4]}$$

has three parts:

$$\sum_{i=1}^{q}\sum_{j=1}^{q}\sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,j,k,l}z_{i,j,k,l}$$

is unique.
The second part is

$$\begin{aligned} x_{0,0,0,0}y_{q+1,q+1,0,0}z_{0,0,q+1,q+1} &+ x_{0,0,0,0}y_{q+1,0,q+1,0}z_{0,q+1,0,q+1} + \\ +x_{0,0,0,0}y_{q+1,0,0,q+1}z_{0,q+1,q+1,0} &+ x_{0,0,0,0}y_{0,q+1,q+1,0}z_{q+1,0,0,q+1} + \\ +x_{0,0,0,0}y_{0,q+1,0,q+1}z_{q+1,0,q+1,0} &+ x_{0,0,0,0}y_{0,0,q+1,q+1}z_{q+1,q+1,0,0}. \end{aligned}$$

Finally

$$\sum_{i=1}^{q}\sum_{j=1}^{q} x_{0,0,0,0}y_{i,j,q+1,0}z_{i,j,0,q+1} + \sum_{i=1}^{q}\sum_{j=1}^{q} x_{0,0,0,0}y_{i,j,0,q+1}z_{i,j,q+1,0}$$
$$+\sum_{i=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{i,q+1,k,0}z_{i,0,k,q+1} + \sum_{i=1}^{q}\sum_{j=1}^{q} x_{0,0,0,0}y_{i,0,k,q+1}z_{i,q+1,k,0}$$
$$+\sum_{i=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,q+1,0,l}z_{i,0,q+1,l} + \sum_{i=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{i,0,q+1,l}z_{i,q+1,0,l}$$
$$+\sum_{j=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{q+1,j,k,0}z_{0,j,k,q+1} + \sum_{j=1}^{q}\sum_{k=1}^{q} x_{0,0,0,0}y_{0,j,k,q+1}z_{q+1,j,k,0}$$
$$+\sum_{j=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{q+1,j,0,l}z_{0,j,q+1,l} + \sum_{j=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{0,j,q+1,l}z_{q+1,j,0,l}$$

$$+\sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{q+1,0,k,l}z_{0,q+1,k,l} + \sum_{k=1}^{q}\sum_{l=1}^{q} x_{0,0,0,0}y_{0,q+1,k,l}z_{q+1,0,k,l}.$$

Overall, this trilinear form represents a scalar x_{0000} times a vector

$$\begin{aligned}\langle y_{i,j,k,l}, y_{q+1,q+1,0,0}, y_{q+1,0,q+1,0}, y_{q+1,0,0,q+1}, y_{0,q+1,q+1,0},\\ , y_{0,q+1,0,q+1}, y_{0,0,q+1,q+1}, y_{i,j,q+1,0}, y_{i,j,0,q+1}, y_{i,q+1,k,0},\\ , y_{i,0,k,q+1}, y_{i,q+1,0,l}, y_{i,0,q+1,l}, , y_{q+1,j,k,0}, y_{0,j,k,q+1},\\ , y_{q+1,j,0,l}, y_{0,j,q+1,l}, y_{q+1,0,k,l}, y_{0,q+1,k,l}\rangle.\end{aligned}$$

or a matrix product of type $\langle 1, 1, (q^4 + 12q^2 + 6)\rangle$ which has value $(q^4 + 12q^2 + 6)^{\tau}$. There are three such forms:

$$X^{[0]}Y^{[4]}Z^{[4]}, X^{[4]}Y^{[0]}Z^{[4]}, X^{[4]}Y^{[4]}Z^{[0]}.$$

The remaining trilinear forms are more complex and do not represent matrix multiplications, and hence we will need to use the methods from previous sections to find suitable "values" for them.
The $X^{[1]}Y^{[1]}Z^{[6]}$ trilinear form has four parts:

$$\sum_{i=1}^{q} x_{i,0,0,0}y_{i,0,0,0}z_{0,q+1,q+1,q+1} + \sum_{j=1}^{q} x_{0,j,0,0}y_{0,j,0,0}z_{q+1,0,q+1,q+1}$$

$$+\sum_{i,j=1}^{q} x_{i,0,0,0}y_{0,j,0,0}z_{i,j,q+1,q+1} + \sum_{i,j=1}^{q} x_{0,j,0,0}y_{i,0,0,0}z_{i,j,q+1,q+1}$$

which is equal to $X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[0]}Z^{[4]}$;

$$\sum_{k=1}^{q} x_{0,0,k,0}y_{0,0,k,0}z_{q+1,q+1,0,q+1} + \sum_{l=1}^{q} x_{0,0,0,l}y_{0,0,0,l}z_{q+1,q+1,q+1,0}$$

$$+\sum_{k,l=1}^{q} x_{0,0,k,0}y_{0,0,0,l}z_{q+1,q+1,k,l} + \sum_{i,j=1}^{q} x_{0,0,0,l}y_{0,0,k,0}z_{q+1,q+1,k,l}$$

which is equal to $X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[1]}Z^{[2]}$;

$$\sum_{i,k=1}^{q} x_{0,0,k,0}y_{i,0,0,0}z_{i,q+1,k,q+1} + \sum_{l=1}^{q} x_{0,0,k,0}y_{0,j,0,0}z_{q+1,j,k,q+1}$$

$$\sum_{i,l=1}^{q} x_{0,0,0,l}y_{i,0,0,0}z_{i,q+1,q+1,l} + \sum_{k,l=1}^{q} x_{0,0,0,l}y_{0,0,k,0}z_{q+1,q+1,k,l}$$

which is equal to $X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[0]}Z^{[3]}$;
and

$$\sum_{i,k=1}^{q} x_{i,0,0,0}y_{0,0,k,0}z_{i,q+1,k,q+1} + \sum_{j,k=1}^{q} x_{0,j,0,0}y_{0,0,k,0}z_{q+1,j,k,q+1}$$

$$\sum_{i,l=1}^{q} x_{i,0,0,0}y_{0,0,0,l}z_{i,q+1,q+1,l} + \sum_{j,l=1}^{q} x_{0,j,0,0}y_{0,0,0,l}z_{q+1,j,q+1,l}$$

which is equal to $X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[1]}Z^{[3]}$.
Using the fact that $V_\tau(A \otimes B) \geq V_\tau(A) \times V_\tau(B)$, we see that the value of the indivual blocks is greater than

$$2^{2/3}q^\tau(q^{3\tau}+2)^{1/3} \times 1$$

for the first two blocks, and

$$(2q)^\tau \times (2q)^\tau = (4q^2)^\tau$$

for the last two.
Lemma 21 will show that the value of this term (that is, the sum of the four preceding parts), is at least

$$2^{2/3}(8q^{3\tau}(q^{3\tau}+2)+(2q)^{6\tau})^{1/3}.$$

We will denote this number as V_{116}.
Using a similar method, we find that

$$X^{[2]}Y^{[2]}Z^{[4]} = (X^{[2]}Y^{[2]}Z^{[0]} \otimes X^{[0]}Y^{[0]}Z^{[4]}) \oplus (X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[2]}Y^{[2]}Z^{[0]})$$

$$\oplus (X^{[2]}Y^{[0]}Z^{[2]} \otimes X^{[0]}Y^{[2]}Z^{[2]}) \oplus (X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[2]}Y^{[0]}Z^{[4]})$$

$$\oplus (X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[1]}Y^{[1]}Z^{[2]})$$

$$\oplus (X^{[1]}Y^{[2]}Z^{[1]} \otimes X^{[1]}Y^{[0]}Z^{[3]}) \oplus (X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[1]}Y^{[2]}Z^{[1]})$$

$$\oplus (X^{[2]}Y^{[1]}Z^{[1]} \otimes X^{[0]}Y^{[1]}Z^{[3]}) \oplus (X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[2]}Y^{[1]}Z^{[1]}).$$

The blocks in the first row will have value greater than $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times 1$.
The blocks in the second row have value greater than $(q^2+2)^\tau \times (q^2+2)^\tau$.
The block in the third row has value greater than
$2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times 2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)}$. The blocks in the fourth and fifth rows have value greater than $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times (2q)^\tau$.
It will be shown in Lemma 25 that the overall value of this trilinear form exceeds

$$V_{224} = \frac{(4q^{3\tau}(q^{3\tau}+2)+2(q^2+2)^{3\tau})^{2/3}(2(2q)^{3\tau}+(q^2+2)^{3\tau}+2)^{1/3}}{(q^2+2)^\tau}.$$

Similarly,

$$X^{[1]}Y^{[3]}Z^{[4]} = (X^{[1]}Y^{[2]}Z^{[1]} \otimes X^{[0]}Y^{[1]}Z^{[3]}) \oplus (X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[2]}Z^{[1]})$$

$$\oplus (X^{[1]}Y^{[3]}Z^{[0]} \otimes X^{[0]}Y^{[0]}Z^{[4]}) \oplus (X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[3]}Z^{[0]})$$

$$\oplus(X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[3]}Z^{[1]}) \oplus (X^{[0]}Y^{[3]}Z^{[1]} \otimes X^{[1]}Y^{[0]}Z^{[3]})$$

$$\oplus(X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[2]}Z^{[2]}) \oplus (X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[1]}Y^{[1]}Z^{[2]})$$

contains four kinds of block. The blocks in the first row contribute $(2q)^\tau \times 2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)}$, the second row blocks contribute $(2q)^\tau \times 1$, the third row $(2q)^\tau \times (2q)^\tau$ and the final row $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times (q^2+2)^\tau$.
It will be proven in Lemma 23 that the overall value of this trilinear form exceeds

$$V_{134} = 2^{2/3}(4q^{3\tau}(q^{3\tau}+2) + (2q)^{3\tau})^{1/3}(2(2q)^{3\tau} + (q^2+2)^{3\tau} + 2)^{1/3}$$

Five kinds of block are present in

$$X^{[2]}Y^{[3]}Z^{[3]} = (X^{[2]}Y^{[1]}Z^{[1]} \otimes X^{[0]}Y^{[2]}Z^{[2]}) \oplus (X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[2]}Y^{[1]}Z^{[1]})$$

$$\oplus(X^{[2]}Y^{[0]}Z^{[2]} \otimes X^{[0]}Y^{[3]}Z^{[1]}) \oplus (X^{[0]}Y^{[3]}Z^{[1]} \otimes X^{[2]}Y^{[0]}Z^{[2]})$$

$$\oplus(X^{[2]}Y^{[2]}Z^{[0]} \otimes X^{[0]}Y^{[1]}Z^{[3]}) \oplus (X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[2]}Y^{[2]}Z^{[0]})$$

$$\oplus(X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[1]}Y^{[2]}Z^{[1]}) \oplus (X^{[1]}Y^{[2]}Z^{[1]} \otimes X^{[1]}Y^{[1]}Z^{[2]})$$

$$\oplus(X^{[1]}Y^{[3]}Z^{[0]} \otimes X^{[1]}Y^{[0]}Z^{[3]}) \oplus (X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[1]}Y^{[3]}Z^{[0]}).$$

The blocks in the first row contribute $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times (q^2+2)^\tau$, the blocks in the second and third rows $(q^2+2)^\tau \times (2q)^\tau$, the fourth row $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times 2^{2/3}q^\tau(q^{3\tau}+2)n^{(1/3)}$ and the fifth row $(2q)^\tau \times (2q)^\tau$.
It will be shown in Lemma 24 that the overall value of this trilinear form is

$$V_{233} = \frac{(2(q^2+2)^{3\tau} + 4q^{3\tau}q^{3\tau}(q^{3\tau}+2))^{1/3}(4(q^{3\tau}+2) + (2q)^{3\tau})^{2/3}}{q^\tau(q^{3\tau}+2)^{1/3}}.$$

Finally,

$$X^{[1]}Y^{[2]}Z^{[5]} = (X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[1]}Z^{[3]}) \oplus (X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[1]}Z^{[2]})$$

$$\oplus(X^{[1]}Y^{[2]}Z^{[1]} \otimes X^{[0]}Y^{[0]}Z^{[4]}) \oplus (X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[2]}Z^{[1]})$$

$$\oplus(X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[2]}Z^{[2]}) \oplus (X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[1]}Y^{[0]}Z^{[3]}).$$

The blocks in the first row have value $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times (2q)^\tau$, the second row blocks have value $2^{2/3}q^\tau(q^{3\tau}+2)^{(1/3)} \times 1$ and the last blocks have value $(2q)^\tau \times (q^2+2)^\tau$.
It will be shown in Lemma 22 that the overall value of this trilinear form is at least

$$V_{125} = 2^{2/3}(4q^{3\tau}(q^{3\tau}+2) + 2(q^2+2)^{3\tau})^{1/3}(\frac{4q^{3\tau}(q^{3\tau}+2)}{(q^2+2)^{3\tau}} + (2q)^{3\tau})^{1/3}$$

4.2 Raising the Algorithm to the Fourth Tensor Power

Take the Nth tensor power of the original algorithm raised to the fourth power, where N is divisible by 3. Let α_l, $0 \leq l \leq 8$ be positive real numbers snuch that

$$\sum_{l=0}^{8} \alpha_l = 1$$

and

$$\sum_{l=0}^{8} l\alpha_l = 8/3,$$

and let A_l be integer approximations to $\alpha_l N$ such that

$$\sum_{l=0}^{8} A_l = N$$

and

$$\sum_{l=0}^{8} lA_l = 8N/3$$

Retain only those blocks of variables such that

$$|\{j|1 \leq j \leq N, I_j = l\}| = A_l,$$

setting the others to zero, where I_j picks out the jth index position.
Let M'' be the number of nonzero triples containing a given block $X^{[I]}$. We have that

$$M'' = \sum_{\eta_{l,m,n}} \frac{\prod_{0\leq l\leq 8} A_l!}{\prod_{l+m+n=8} \eta_{l,m,n}!}.$$

We say that M'' is maximized at

$$\eta_{l,m,n} = \gamma_{l,m,n}$$
$$\gamma_{008} = \gamma_{080} = \gamma_{800} = \hat{a}$$
$$\gamma_{017} = \gamma_{071} = \gamma_{107} = \gamma_{170} = \gamma_{701} = \gamma_{710} = \hat{b}$$
$$\gamma_{026} = \gamma_{062} = \gamma_{206} = \gamma_{260} = \gamma_{602} = \gamma_{620} = \hat{c}$$
$$\gamma_{035} = \gamma_{053} = \gamma_{503} = \gamma_{305} = \gamma_{530} = \gamma_{350} = \hat{d}$$
$$\gamma_{044} = \gamma_{440} = \gamma_{404} = \hat{e}$$
$$\gamma_{116} = \gamma_{161} = \gamma_{611} = \hat{f}$$
$$\gamma_{125} = \gamma_{152} = \gamma_{215} = \gamma_{251} = \gamma_{512} = \gamma_{521} = \hat{g}$$
$$\gamma_{134} = \gamma_{143} = \gamma_{341} = \gamma_{314} = \gamma_{413} = \gamma_{431} = \hat{h}$$
$$\gamma_{233} = \gamma_{323} = \gamma_{332} = \hat{i}$$
$$\gamma_{224} = \gamma_{242} = \gamma_{422} = \hat{j}.$$

This symmetry arises from the fact that the number of triples containing a given X block is M'', the number of triples containing a given Y-block is also M'' (via a relabelling of variables) and similarly for the number of triples containing a Z block. Now, we suppose we wish to find the value of $\gamma_{l,m,n}$ where the maximal term of M'' arises. Finding $\frac{\partial M''}{\partial \eta_{l,m,n}}$, $\frac{\partial M''}{\partial \eta_{n,l,m}}$ and $\frac{\partial M''}{\partial \eta_{m,n,l}}$ for each of these three functions yields three sets of three equations, each of which is identical except for relabelling- we find that we should set $\eta_{l,m,n} = \eta_{m,n,l} = \eta_{n,l,m}$ which will find a zero common to all three functions. This follows for all the η and so symmetry is found.

$$A_0 = 2\hat{a} + 2\hat{b} + 2\hat{c} + 2\hat{d} + \hat{e}$$

$$A_1 = 2\hat{b} + 2\hat{f} + 2\hat{g} + 2\hat{h}$$

$$A_2 = 2\hat{c} + 2\hat{g} + \hat{i} + 2\hat{j}$$

$$A_3 = 2\hat{d} + 2\hat{h} + 2\hat{i}$$

$$A_4 = 2\hat{e} + 2\hat{h} + \hat{j}$$

$$A_5 = 2\hat{d} + 2\hat{g}$$

$$A_6 = 2\hat{c} + \hat{f}$$

$$A_7 = 2\hat{b}$$

$$A_8 = \hat{a}.$$

In later calculations, we will approximate M'' by its largest term, times a polynomial N^p. We will call this maximal term M_{max}, and it occurs at $\hat{a}, \hat{b}, ...\hat{j}$. We suppose that the $A_0, ..., A_8$ are fixed. This will fix the value of $\hat{a}(= A_8)$ and $\hat{b}(= A_7/2)$. Fixing the $A_0, ..., A_8$ will not fix the other values, however. Since there were two conditions imposed on the nine $A_0...A_8$ variables and only one on the ten $\hat{a}, .., \hat{j}$ variables, we have two degrees of freedom among the $\hat{a}, .., \hat{j}$ given $A_0, ..., A_8$.

As in the previous chapter, this affects how we do our calculations. The usual Salem-Spencer pruning will only remove those blocks that share an X, Y or Z-block- it will not take into account distributions of trilinear forms. We therefore need to augment our method.

Performing the usual Salem-Spencer pruning will give us a set Δ of blocks which do not share X, Y or Z blocks. These blocks all represent independent matrix products, but of all distributions of $\hat{a}, .., \hat{j}$ given $A_0, .., A_8$. Therefore, if we seek how much of a *particular* distribution of $\hat{a}, ..\hat{j}$ remains after this pruning, we must consider only a proportion of Δ. If M' is the term which contains a particular distribution $\hat{a}', .., \hat{j}'$, the proportion remaining will be $|\Delta|\frac{M'}{M''}$.

As stated before, we will find it easier in future calculations to approximate M'' by N^2 times its largest term M_{max}. Since there are two degrees of freedom among the $\hat{a}, .., \hat{j}$, we select $\hat{c}$ and $\hat{d}$, and derive their relationships with the remaining $\hat{a}, .., \hat{j}$ at M_{max}. We obtain

$$\begin{aligned}
\hat{e} &= A_0 - 2\hat{a} - 2\hat{b} - 2\hat{c} - 2\hat{d} \\
\hat{f} &= A_6 - 2\hat{c} \\
\hat{g} &= \frac{1}{2}(A_5 - 2\hat{g}) \\
\hat{h} &= \frac{1}{2}(A_1 - 2\hat{b} - 2\hat{f} - 2\hat{g}) \\
\hat{i} &= \frac{1}{2}(A_3 - 2\hat{d} - 2\hat{h}) \\
\hat{j} &= A_4 - 2\hat{e} - 2\hat{h}.
\end{aligned}$$

Placing these into

$$F = \frac{\prod_l A_l!}{(\hat{a}!)^3(\hat{b}!)^6(\hat{c}!)^6(\hat{d}!)^6(\hat{e}!)^3(\hat{f}!)^3(\hat{g}!)^6(\hat{h}!)^6(\hat{i}!)^3(\hat{j}!)^3}.$$

We approximate the factorials using Stirling's formula, take logs and differentiate with respect to $\hat{c}$ and $\hat{d}$ to obtain

$$\begin{aligned}
\frac{\partial F}{\partial \hat{c}} &= 6\log(\hat{c}) - 6\log(\hat{e}) - 6\log(\hat{f}) + 12\log(\hat{h}) - 6\log(\hat{i}) \\
\frac{\partial F}{\partial \hat{d}} &= 6\log(\hat{d}) - 6\log(\hat{e}) - 6\log(\hat{g}) + 6\log(\hat{h}) - 6\log(\hat{i}) - 6\log(\hat{j}).
\end{aligned}$$

We therefore have that both of these equal zero when

$$\hat{c} = \frac{\hat{e}\hat{f}\hat{i}}{\hat{h}^2}, \hat{d} = \frac{\hat{e}\hat{g}\hat{i}}{\hat{h}\hat{j}}.$$

This will give a global maximum, since the function $x\log(x)$ is convex for $x > 0$ means that the function F is also convex for $\hat{c}, \hat{d}, \hat{e}, ..., \hat{j} > 0$.
Given this, if we choose values of $\hat{a}, \hat{b}, \hat{e}, \hat{f}, \hat{g}, \hat{h}, \hat{i}$ and $\hat{j}$, we may put these values into our $\hat{c}$ and $\hat{d}$ formulae, which will give us values of $A_0, .., A_8$ (with its maximal term already known).
If we choose a particular distribution $\hat{a}', .., \hat{j}'$ to keep, we get our auxiliary equation as being

$$\begin{aligned}
|\Delta| \quad & \frac{M'}{N^2 M_{max}}[1]^{3\hat{a}}[V_{017}]^{6\hat{b}}[V_{026}]^{6\hat{c}'} \times \\
\times \quad & [V_{035}]^{6\hat{d}'}[V_{044}]^{3\hat{e}'}[V_{116}]^{3\hat{f}'}[V_{125}]^{6\hat{g}'}[V_{134}]^{6\hat{h}'}[V_{233}]^{3\hat{i}'}[V_{224}]^{6\hat{j}'}.
\end{aligned}$$

That is, we have $|\Delta|\frac{M'}{N^2 M_{max}}$ independent matrix products of value

$$[1]^{3\hat{a}}[V_{017}]^{6\hat{b}}[V_{026}]^{6\hat{c}'}[V_{035}]^{6\hat{d}'}[V_{044}]^{3\hat{e}'}[V_{116}]^{3\hat{f}'}[V_{125}]^{6\hat{g}'}[V_{134}]^{6\hat{h}'}[V_{233}]^{3\hat{i}'}[V_{224}]^{6\hat{j}'}.$$

Our task, therefore is to find values of $A_0, .., A_8$ and $\hat{c}', .., \hat{j}'$ to satisfy maximise this overall equation (and hence find as low a value of τ as possible). We acheive this by choosing values of $\hat{a}, \hat{b}, \hat{d}, .., \hat{j}$, and using them to find values of $A_0, .., A_8$ as described above. We then allow $\hat{c}'$ and $\hat{d}'$ to vary. Overall, the auxiliary equation is a function of nine variables. The values obtained below were obtained using a Newton Raphson argument in Maple.

As before, we set our $M = cM'' + 1$ where c is a constant which ensures that all the independence arguments from chapter 2 hold. Construct the Salem-Spencer set of size $M^{1-\epsilon}$. Choose random weights $w_j, 0 \leq j \leq N$. Define the three hash functions:

$$\begin{aligned} b_X(I) &= \sum_{i=1}^{N} I_i w_i \mod M \\ b_Y(J) &= w_0 + \sum_{i=1^N} J_i w_i \mod M \\ b_Z(K) &= \frac{1}{2}(w_0 + \sum_{i=1}^{N}(8 - K_j)) \mod M. \end{aligned}$$

We set to zero any blocks that do not hash into a $b \in B$. We also judiciously set to zero any blocks that appear in more than one triple, and we find that the number of blocks remaining is proportional to

$$\frac{M^{1-\epsilon}}{M}\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8}.$$

Now, for a portion of these triples, which is at least

$$\frac{M_{\eta_{l,m,n}}}{M_{\max} N^2}$$

of the total, the N indices $j = 1, 2, .., N$ will contain about $\eta_{l,m,n}$ instances of $X^{[l]}Y^{[m]}Z^{[N]}$. We say that the *overall auxiliary equation* is maximized when

$$\begin{gathered} \eta_{l,m,n} = \gamma'_{l,m,n} \\ \gamma'_{008} = \gamma'_{080} = \gamma'_{800} = \hat{a} \\ \gamma'_{017} = \gamma'_{071} = \gamma'_{107} = \gamma'_{170} = \gamma'_{701} = \gamma'_{710} = \hat{b} \\ \gamma'_{026} = \gamma'_{062} = \gamma'_{206} = \gamma'_{260} = \gamma'_{602} = \gamma'_{620} = \hat{c}' \\ \gamma'_{035} = \gamma'_{053} = \gamma'_{503} = \gamma'_{305} = \gamma'_{530} = \gamma'_{350} = \hat{d}' \\ \gamma'_{044} = \gamma'_{440} = \gamma'_{404} = \hat{e}' \\ \gamma'_{116} = \gamma'_{161} = \gamma'_{611} = \hat{f}' \\ \gamma'_{125} = \gamma'_{152} = \gamma'_{215} = \gamma'_{251} = \gamma'_{512} = \gamma'_{521} = \hat{g}' \\ \gamma'_{134} = \gamma'_{143} = \gamma'_{341} = \gamma'_{314} = \gamma'_{413} = \gamma'_{431} = \hat{h}' \\ \gamma'_{233} = \gamma'_{323} = \gamma'_{332} = \hat{i}' \\ \gamma'_{224} = \gamma'_{242} = \gamma'_{422} = \hat{j}'. \end{gathered}$$

The relationship of these values with the A_n is entirely analogous to the $\hat{c}, .., \hat{j}$ values. The symmetry is shown using similar arguments to above. The value of each triple of blocks is therfore about

$$\begin{aligned} &[1]^{3\hat{a}}[V_{017}]^{6\hat{b}}[V_{026}]^{6\hat{c}'}[V_{035}]^{6\hat{d}'}[V_{044}]^{3\hat{e}'}[V_{116}]^{3\hat{f}'}[V_{125}]^{6\hat{g}'}[V_{134}]^{6\hat{h}'}[V_{233}]^{3\hat{i}'}[V_{224}]^{6\hat{j}'} \\ &= V_{\text{overall}}. \end{aligned}$$

The auxiliary equation is thus

$$(q+2)^{4N} \geq \binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8} \frac{M_{\gamma'_{l,m,n}}}{M_{\max} N^2} V_{\text{overall}}.$$

We want to choose $\hat{a}, \hat{b}, \hat{c}', \hat{d}', \hat{e}', \hat{f}', \hat{g}', \hat{h}', \hat{\imath}', \hat{\jmath}'$ to maximize the RHS, subject to

$$3\hat{a} + 6\hat{b} + 6\hat{c}' + 6\hat{d}' + 3\hat{e}' + 3\hat{f}' + 6\hat{g}' + 6\hat{h}' + 3\hat{\imath}' + 3\hat{\jmath}' = N.$$

Setting $\hat{a} = \overline{a}N$ and so on, letting N grow and taking Nth roots, we get

$$\begin{aligned}(q+2)^4 &= \frac{V_{017}^{6\overline{b}} V_{026}^{6\overline{c}'} V_{035}^{6\overline{d}'} V_{044}^{3\overline{e}'} V_{116}^{3\overline{f}'} V_{125}^{6\overline{g}'} V_{134}^{6\overline{h}'} V_{233}^{3\overline{\imath}'} V_{224}^{6\overline{\jmath}'}}{\overline{A_0}^{\overline{A_0}} \overline{A_1}^{\overline{A_1}} \overline{A_2}^{\overline{A_2}} \overline{A_3}^{\overline{A_3}} \overline{A_4}^{\overline{A_4}} \overline{A_5}^{\overline{A_5}} \overline{A_6}^{\overline{A_6}} \overline{A_7}^{\overline{A_7}} \overline{A_8}^{\overline{A_8}}} \times \\ &\times \frac{\overline{c}^{6\overline{c}} \overline{d}^{6\overline{d}} \overline{e}^{3\overline{e}} \overline{f}^{3\overline{f}} \overline{g}^{6\overline{g}} \overline{h}^{6\overline{h}} \overline{\imath}^{3\overline{\imath}} \overline{\jmath}^{3\overline{\jmath}}}{\overline{c'}^{6\overline{c'}} \overline{d'}^{6\overline{d'}} \overline{e'}^{3\overline{e'}} \overline{f'}^{3\overline{f'}} \overline{g'}^{6\overline{g'}} \overline{h'}^{6\overline{h'}} \overline{\imath'}^{3\overline{\imath'}} \overline{\jmath'}^{3\overline{\jmath'}}}.\end{aligned}$$

We find that

$$\overline{a} = 0.00000003659$$

$$\overline{b} = 0.00000630812$$

$$\overline{c}' = 0.00026561642$$

$$\overline{d}' = 0.00355378473$$

$$\overline{e}' = 0.01227154212$$

$$\overline{f}' = 0.00035081817$$

$$\overline{g}' = 0.00726671323$$

$$\overline{h}' = 0.04705061728$$

$$\overline{\imath}' = 0.1317211657$$

$$\overline{\jmath}' = 0.07270369094$$

$$\overline{c} = 0.00026306415$$

$$\overline{d} = 0.00350153973$$

$$\overline{e} = 0.01238113686$$

$$\overline{f} = 0.00035592270$$

$$\overline{g} = 0.00731895819$$

$$\overline{h} = 0.04699326750$$

$$\overline{\imath} = 0.13183076142$$

$$\overline{\jmath} = 0.07259920020$$

$$q = 6$$

gives an exponent of

$$\omega \leq 3\tau < 2.373689703.$$

These figures were originally calculated to 200 decimal points, but have been truncated to 11 for the purposes of this thesis.

4.3 Finding the Values of the Trilinear Forms

Lemma 21. *The value of the trilinear form*

$$X^{[1]}Y^{[1]}Z^{[6]} = X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[0]}Z^{[4]} \oplus X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[1]}Z^{[2]}$$

$$\oplus X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[0]}Z^{[3]} \oplus X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[1]}Z^{[3]}.$$

is at least

$$2^{2/3}(8q^{3\tau}(q^{3\tau}+2)+(2q)^{6\tau})^{1/3}.$$

Proof. Take the $2N$th tensor power. Let α, β be such that $\alpha+\beta = N, 0 \leq \alpha, \beta \leq N$ Retain only those X-blocks with exactly N indices of $[1,0]$ and N of $[0,1]$, and do the same for the Y-blocks. Retain those Z-blocks that have α indices of $[2,4]$ and $[4,2]$, and 2β of $[3,3]$.
Hence, here, α represents the number of $X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[0]}Y^{[0]}Z^{[4]}$ and $X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[1]}Z^{[2]}$ type blocks, and β the number of $X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[0]}Z^{[3]}$ and $X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[1]}Z^{[3]}$ type blocks.
Do analogous operations for $X^{[1]}Y^{[6]}Z^{[1]}$ and $X^{[6]}Y^{[1]}Z^{[1]}$, and tensor multiply them all together.
Hence, the remaining number of different X, Y and Z blocks is the same at

$$\binom{2N}{N,N}\binom{2N}{N,N}\binom{2N}{2\beta,\alpha,\alpha}.$$

The number of non-zero triples containing a given X-block is

$$M'' = \binom{N}{\alpha}^4\binom{2\beta}{\beta}.$$

Similarly for Y and Z blocks. We set $M = 6M''+1$, and construct a Salem-Spencer set B of size $M^{1-\epsilon}$ (which is possible from theorem 10). Set random variables $\{w_0, w_{11}, w_{12}, .., w_{2N1}, w_{2N2}\}$ and define the functions

$$\begin{aligned} b_X(I) &= \sum_i I_{i1}w_{i1} + I_{i2}w_{i2} \mod M \\ b_Y(J) &= w_0 + \sum_i J_{i1}w_{i1} + J_{i2}w_{i2} \mod M \\ b_Z(K) &= \frac{1}{2}(w_0 + \sum_i (I_{i1}w_{i1} + I_{i2}w_{i2} + J_{i1}w_{i1} + J_{i2}w_{i2})) \mod M. \end{aligned}$$

The probability that a triple I, J, K is such that $b_X(I) = b_Y(J) = b \in B$ is $\frac{1}{M^2}$, and the expected number of triples that have this property (for a particular b) is

$$\frac{1}{M^2}\binom{2N}{N,N}\binom{2N}{N,N}\binom{2N}{2\beta,\alpha,\alpha}M''.$$

The expected number of pairs sharing an X-block with $b_X(I) = b_Y(J) = b_Y J' = b \in B$ is

$$\frac{1}{M^3}\binom{2N}{N,N}\binom{2N}{N,N}\binom{2N}{2\beta,\alpha,\alpha}M''(M''-1).$$

Subtracting this expression from the previous one and adding up all possible values of b yields that the number of remaining triples is greater than

$$\frac{cM^{1-\epsilon}}{M}\binom{2N}{N,N}\binom{2N}{N,N}\binom{2N}{2\beta,\alpha,\alpha}$$

for some constant c. Letting N become large and ϵ go to zero we have that

$$V_{116}^{6N} \approx \binom{2N}{N,N}\binom{2N}{N,N}\binom{2N}{2\beta,\alpha,\alpha}(V_{112}V_{004})^{6\alpha}(V_{103}^2)^{6\beta}$$

$$\approx \frac{(2N)^{2N}(2N)^{2N}(2N)^{2N}}{N^N N^N N^N N^N (2\beta N)^{2\beta}(\alpha)^\alpha(\alpha)^\alpha}(V_{112}V_{004})^{6\alpha}(V_{103}^2)^{6\beta}$$

$$= \frac{2^{4N}2^{2\alpha}N^{2N}}{(\beta)^{2\beta}(\alpha)^{2\alpha}}(V_{112}V_{004})^{6\alpha}(V_{103}^2)^{6\beta}$$

$$\approx 2^{4N}\left[\binom{N}{\alpha}(2V_{112}^3V_{004}^3)^\alpha(V_{103}^6)^\beta\right]^2$$

Setting

$$\alpha = \frac{2(V_{112}V_{004N})^3}{2(V_{112}V_{004})^3 + V_{103}^6} \text{ and } \beta = \frac{V_{103}^6 N}{2(V_{112}V_{004})^3 + V_{103}^6}$$

maximizes the equation and letting N grow and taking the $6N$th root yields

$$V_{116} = 2^{2/3}(2V_{112}^3 + V_{103}^6)^{1/3} = 2^{2/3}(8q^{3\tau}(q^{3\tau}+2) + (2q)^{6\tau})^{1/3}$$

as required.
It is clear, since all the values are greater than 0, that the α and β found here satisfy the above constraints. □

Lemma 22. *The value of the trilinear form*

$$X^{[1]}Y^{[2]}Z^{[5]}$$

is at least

$$2^{2/3}(4q^{3\tau}(q^{3\tau}+2) + 2(q^2+2)^{3\tau})^{1/3}(\frac{4q^{3\tau}(q^{3\tau}+2)}{(q^2+2)^{3\tau}} + (2q)^{3\tau})^{1/3}.$$

Proof. As above, we regard this trilinear form as the sum of six different blocks obtained by multiplying two trilinear forms from the square of the original algorithm. Again, this gives the advantage of any improvements in value we obtained in that algorithm, as well as making our calculations simpler. We raise this trilinear form to the $2N$th tensor power, and do the same with $X^{[5]}Y^{[1]}Z^{[2]}$ and $X^{[2]}Y^{[5]}Z^{[1]}$, tensor multiplying all three together. We have α, β, γ such that $\alpha + \beta + \gamma = N$ and $0 \le \alpha, \beta, \gamma \le N$. Here, α represents the number of $X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[1]}Z^{[2]}$ type blocks (and permutations thereof), β the number of $X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[2]}Z^{[1]}$ type

blocks and γ the number of $X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[2]}Z^{[2]}$ type blocks.
Each resulting block will have three vectors
$I, J, K \in \{[1,0],[0,1],[1,1],[2,0],[0,2],[2,3],[3,2],[1,4],[4,1]\}^{6N}$ associated with it.
We set to zero, in the portion belonging to $X^{[1]}Y^{[2]}Z^{[5]}$ any X-blocks whose I vector does not contain N instances of each of $[1,0]$ and $[0,1]$ respectively. In the $X^{[5]}Y^{[1]}Z^{[2]}$ portion, we set to zero any X-blocks that do not contain $\alpha+\gamma$ instances each of $[2,3]$ and $[3,2]$ and β instances each of $[1,4]$ and $[4,1]$. Finally, in the $X^{[2]}Y^{[5]}Z^{[1]}$ portion, we retain only those blocks that contain 2α instances of $[1,1]$ and $\beta+\gamma$ instances each of $[2,0]$ and $[0,2]$.
We do similar operations on Y and Z-blocks: we find that the number of blocks containing a given X, Y or Z block is

$$\binom{N}{\alpha,\beta,\gamma}^2 \binom{\beta+\gamma}{\beta,\gamma}^2 \binom{2\alpha}{\alpha,\alpha} \binom{\alpha+\gamma}{\alpha,\gamma}^2.$$

Using similar probabilistic arguments as before, selcecting an M roughly similar to the number of triples containing a given X-block, we find that, for a constant c, the number of independent triples remaining is greater than

$$\frac{cM^{1-\epsilon}}{M}\binom{2N}{N,N}\binom{2N}{\beta+\gamma,\beta+\gamma,2\alpha}\binom{2N}{\alpha+\gamma,\alpha+\gamma,\beta,\beta}.$$

Letting N grow and letting ϵ go to zero, we get that this expression is proportional to the number of available X-blocks. Since the choice of X-blocks automatically determines the number of Y-blocks of a given type, all these blocks (and not just a portion, as in previous examples) match the desired criteria.
Let $V_{a,b,c}$ denote the "value" of the trilinear form $X^{[a]}Y^{[b]}Z^{[c]}$ from Coppersmith and Winograd's algorithm (we note that several different trilinear forms will have this value: we just pick one for the sake of notation). Then the overall value becomes

$$\binom{2N}{N,N}\binom{2N}{\beta+\gamma,\beta+\gamma,2\alpha}\binom{2N}{\alpha+\gamma,\alpha+\gamma,\beta,\beta}\times \\ \times(V_{112}V_{013})^{6\alpha}(V_{112}V_{004})^{6\beta}(V_{022}V_{013})^{6\gamma}.$$

Approximating this using Stirling's formula, using algebraic manipulation and taking the square root, this reduces to

$$2^{2N}\binom{N}{\alpha,\beta+\gamma}(V_{112}^3)^{\alpha}(2V_{022}^3)^{\beta+\gamma}\binom{\beta}{\alpha+\gamma}(\frac{V_{112}^3}{V_{022}^3})^{\beta}(V_{013}^3)^{\alpha+\gamma}$$

We set

$$\alpha = \frac{V_{112}^3 N}{V_{112}^3 + 2V_{022}^3}, \beta+\gamma = \frac{2V_{022}^3 N}{V_{112}^3 + 2V_{022}^3}$$

and

$$\beta = \frac{\frac{V_{112}^3}{V_{022}^3}N}{\frac{V_{112}^3}{V_{022}^3} + V_{013}^3}, \alpha+\gamma = \frac{V_{013}^3 N}{\frac{V_{112}^3}{V_{022}^3} + V_{013}^3}$$

which renders the value as being approximately equal to

$$2^{2N}(V_{112}^3 + 2V_{022}^3)^N(\frac{V_{112}^3}{V_{022}^3} + V_{013}^3)^N.$$

We let N grow and take $3N$th roots. Substituting the values previously obtained for the appropriate V terms we obtain the desired result.
It remains to show that α, β and γ satisfy the constraints. If we take $q = 6$ and $\tau = V/3$. These values give $\alpha = 0.64019N$, $\beta = 0.00983N$ and $\gamma = 0.3461507N$. Since these values all lie between 0 and 1, and their sum is N, we have that all the constraints are satisfied.

□

Lemma 23. *The value of the trilinear form*

$$X^{[1]}Y^{[3]}Z^{[4]}$$

is at least

$$V_{134} = 2^{2/3}(4q^{3\tau}(q^{3\tau} + 2) + (2q)^{3\tau})^{1/3}(2(2q)^{3\tau} + (q^2 + 2)^{3\tau} + 2)^{1/3}$$

Proof. We have $\alpha + \beta + \gamma + \delta = N$ and $0 \le \alpha, \beta, \gamma, \delta \le N$. We raise $X^{[1]}Y^{[3]}Z^{[4]}$, $X^{[4]}Y^{[1]}Z^{[3]}$ and $X^{[3]}Y^{[4]}Z^{[1]}$ to the $2N$th tensor power and tensor multiply all three together.
We let α be the number of $X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[1]}Y^{[2]}Z^{[1]}$ blocks, β the number of $X^{[0]}Y^{[0]}Z^{[4]} \otimes X^{[1]}Y^{[3]}Z^{[0]}$ blocks, γ the number of $X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[0]}Y^{[3]}Z^{[1]}$ blocks and δ the number of $X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[1]}Y^{[1]}Z^{[2]}$ blocks (with the same proportions holding for the appropriate permutations). Retain only those X blocks (respectively , Y, Z blocks) which contain N instances of $[1, 0]$ and N instances of $[0, 1]$. Retain only those Y-blocks (respectively, Z,X blocks) that contain $\alpha + \delta$ instances of $[1, 2]$ and $[2, 1]$ and $\beta + \gamma$ instances of $[0, 3]$ and $[3, 0]$. Finally, retain only those Z-blocks (respectively X and Y blocks) that contain $\alpha + \gamma$ instances of $[3, 1]$ and $[1, 3]$, β instances of $[0, 4]$ and $[4, 0]$ and 2δ instances of $[2, 2]$.
As before we define M, a Salem-Spencer set B and appropriate hash functions, and set to zero blocks which do not map into B, and set to zero appropriate blocks such that no X, Y or Z blocks are shared.
We are left with about

$$\binom{2N}{N, N}\binom{2N}{\alpha + \delta, \alpha + \delta, \beta + \gamma, \beta + \gamma}\binom{2N}{\alpha + \gamma, \alpha + \gamma, \beta, \beta, 2\delta}$$

independent blocks (for N greater than some $\epsilon > 0$). The "value“ of each block is greater than

$$(V_{112}V_{013})^{6\alpha}(V_{013}V_{014})^{6\beta}(V_{013}^2)^{6\gamma}(V_{112}V_{022})^{6\delta}.$$

If we multiply these last two expressions together, approximate using Stirling's formula and take the square root, we obtain

$$2^{2N}\binom{N}{\alpha + \delta, \beta + \gamma}(V_{112}^3)^{\alpha+\delta}(V_{013}^3)^{\beta+\gamma}\binom{N}{\alpha + \gamma, \beta, \delta}(2V_{013}^3)^{\alpha+\gamma}(2V_{004}^3)^{\beta}(V_{022}^3)^{\delta}.$$

Setting

$$\beta = \frac{2N}{2V_{013}^3 + V_{022}^3 + 2}$$
$$\delta = \frac{V_{022}^3 N}{2V_{013}^3 + V_{022}^3 + 2}$$
$$\alpha + \gamma = \frac{2V_{013}^3 N}{2V_{013}^3 + V_{022}^3 + 2}$$
$$\alpha + \delta = \frac{V_{112}^3 N}{V_{112}^3 + V_{013}^3}$$
$$\beta + \gamma = \frac{V_{013}^3 N}{V_{112}^3 + V_{013}^3}$$

and taking $3N$th roots, this becomes approximately

$$2^{2/3}(V_{112}^3 + V_{013}^3)^{1/3}(2V_{013}^3 + V_{022}^3 + 2)^{1/3}.$$

Substituting the appropriate V values in, we get the desired result.
We must show that we can get values of α, β, γ and δ to satisfy the constraints. With $q = 6$ and $\tau = 2.373691893/3$, we obtain $\alpha = 0.0974439531N$, $\beta = 0.0003546866N$, $\gamma = 0.017245N$ and $\delta = 0.884956N$, which do indeed satisfy the constraints. □

Lemma 24. *The value of the trilinear form*

$$X^{[2]}Y^{[3]}Z^{[3]}$$

is at least

$$\frac{1}{q^\tau(q^{3\tau}+2)^{1/3}}(2(q^2+2)^{3\tau} + 4q^{3\tau}(q^{3\tau}+2))^{1/3}(4q^{3\tau}(q^{3\tau}+2) + (2q)^{3\tau})^{2/3}.$$

Proof. We have $\alpha, \beta, \gamma, \delta, \epsilon$ such that $\alpha + \beta + \gamma + \delta + \epsilon = N$ and $0 \leq \alpha, \beta, \gamma, \delta, \epsilon \leq N$. Let α be the number of instances of $X^{[0]}Y^{[2]}Z^{[2]} \otimes X^{[2]}Y^{[1]}Z^{[1]}$, β the number of instances of $X^{[2]}Y^{[0]}Z^{[2]} \otimes X^{[0]}Y^{[3]}Z^{[1]}$, γ the number of instances of $X^{[0]}Y^{[1]}Z^{[3]} \otimes X^{[2]}Y^{[2]}Z^{[0]}$, δ the number of instances of $X^{[1]}Y^{[1]}Z^{[2]} \otimes X^{[1]}Y^{[2]}Z^{[1]}$ and ϵ the number of instances of $X^{[1]}Y^{[0]}Z^{[3]} \otimes X^{[1]}Y^{[3]}Z^{[0]}$. The same proportions hold for appropriate permutations. As before, we raise the three permutations of X, Y and Z to the $2N$th power. Set to zero those X-blocks which do not contain $\alpha + \beta + \gamma$ instances of $[2, 0]$ and $[0, 2]$, and $2(\delta + \epsilon)$ instances of $[1, 1]$. Set to zero those Y-blocks that do not contain $\alpha + \gamma + \delta$ instances of $[1, 2]$ and $[2, 1]$ and $\beta + \epsilon$ instances of $[3, 0]$ and $[0, 3]$. Finally, retain only those Z-blocks which have $\alpha + \beta + \delta$ instances of $[1, 2]$ and $[2, 1]$ and $\gamma + \epsilon$ instances of $[3, 0]$ and $[0, 3]$.
The number of blocks remaining is

$$\binom{2N}{\alpha+\beta+\gamma, \alpha+\beta+\gamma, 2(\delta+\epsilon)}\binom{2N}{\alpha+\gamma+\delta, \alpha+\gamma+\delta, \beta+\epsilon, \beta+\epsilon} \times \\ \times \binom{2N}{\alpha+\beta+\delta, \alpha+\beta+\delta, \gamma+\epsilon, \gamma+\epsilon}.$$

After defining the usual Salem-Spencer pruning, we are left with approcimately the same number of blocks. Of these, a proportion will have value equal to

$$(V_{112}V_{022})^{6\alpha}(V_{022}V_{013})^{6(\beta+\gamma)}(V_{112}^2)^{6\delta}(V_{013}^2)^{6\epsilon}.$$

However, it is possible to re-arrange this expression so that the overall size of the matrix products created does not depend on the individual choices of $\alpha..\epsilon$. If we approximate M'' by its largest term, we can set the $\alpha..\epsilon$ to match these, and so we are left with a fraction $\frac{1}{N}$ of the number of boxes remaining.
Thus, the overall auxiliary equation is roughly equal to the product of the previous two terms, and, after applying Stirling's formula and taking the square root, can be rewritten as

$$\begin{aligned}\frac{2^{2N}}{V_{112}^{3N}} &\times \binom{N}{\alpha+\beta+\gamma,\delta+\epsilon}(2V_{022}^3)^{\alpha+\beta+\gamma}(V_{112}^3)^{\delta+\epsilon}\\ &\times \binom{N}{\alpha+\beta+\delta,\beta+\epsilon}(V_{112}^3)^{\alpha+\gamma+\delta}(V_{013}^3)^{\beta+\epsilon}\\ &\times \binom{N}{\alpha+\beta+\delta,\gamma+\epsilon}(V_{013}^3)^{\gamma+\epsilon}(V_{112}^3)^{\alpha+\beta+\delta}.\end{aligned}$$

We then set

$$\begin{aligned}\alpha+\beta+\gamma &= \frac{2V_{022}^3N}{2V_{022}^3+V_{112}^3}\\ \beta+\epsilon &= \frac{V_{112}^3N}{2V_{022}^3+V_{112}^3}\\ \alpha+\gamma+\delta &= \alpha+\beta+\delta=\frac{V_{112}^3N}{V_{112}^3+V_{013}^3}\\ \gamma+\epsilon &= \beta+\epsilon=\frac{V_{013}^3N}{V_{112}^3+V_{013}^3}\end{aligned}$$

and take $3N$th roots to get

$$\frac{2^{2/3}}{V_{112}}(2V_{022}^3+V_{112}^3)^{1/3}(V_{112}^3+V_{013}^3)^{2/3}$$

which, upon substituting the required values back, gives us the desired result.
To show that $\alpha,\beta,\gamma,\delta,\epsilon$ satisfy the constraints, we set $q=6$ and $\tau=2.373691893/3$. We obtain the following equations:

$$\begin{aligned}\alpha+\beta+\gamma &= 0.3559809N\\ \delta+\epsilon &= 0.6440190993N\\ \alpha+\beta+\delta &= 0.9824002N\\ \gamma+\epsilon &= 0.017599787N.\end{aligned}$$

This system of equations does not yield a unique solution for $\alpha,\beta,\gamma,\delta,\epsilon$. However, it is clear that a solution in our required region can exist: set

$$\begin{aligned}\alpha &= 0.033598N\\ \beta &= 0.01N\\ \gamma &= 0.01N\\ \delta &= 0.6364193N\\ \epsilon &= 0.007599N.\end{aligned}$$

We see all the constraints are satisfied. □

Lemma 25. *The trilinear form*

$$X^{[2]}Y^{[2]}Z^{[4]}$$

has value at least

$$\frac{1}{(q^2+2)^\tau}(4q^{3\tau}(q^{3\tau}+2)+2(q^2+2)^{3\tau})^{2/3}(2(2q)^{3\tau}+(q^2+2)^{3\tau}+2)^{1/3}$$

Proof. We work as before, taking $2N$th tensor powers of each of the three permutations and tensor them together. We have $\alpha,\beta,\gamma,\delta,\epsilon$ such that $\alpha+\beta+\gamma+\delta+\epsilon=N$ and $0\le\alpha,\beta,\gamma,\delta,\epsilon\le N$.
Here, α represents the number of instances of $X^{[0]}Y^{[0]}Z^{[4]}\otimes X^{[2]}Y^{[2]}Z^{[0]}$, β the number of instances of $X^{[0]}Y^{[2]}Z^{[2]}\otimes X^{[2]}Y^{[0]}Z^{[2]}$, γ the number of instances of $X^{[1]}Y^{[1]}Z^{[2]}\otimes X^{[1]}Y^{[1]}Z^{[2]}$, δ the number of instances of $X^{[1]}Y^{[0]}Z^{[3]}\otimes X^{[1]}Y^{[2]}Z^{[1]}$ and ϵ the number of instances of $X^{[0]}Y^{[1]}Z^{[3]}\otimes X^{[2]}Y^{[1]}Z^{[1]}$ (with the same numbers holding for appropriate permutations).
We retain only those X-blocks with $\alpha+\beta+\epsilon$ instances of $[2,0]$ and $[0,2]$ and $2(\gamma+\delta)$ of $[1,1]$. Retain only those Y-blocks with $\alpha+\beta+\delta$ instances of $[2,0]$ and $[0,2]$ and $2(\gamma+\epsilon)$ of $[1,1]$. Finally, retain only those Z-blocks which have α instances each of $[4,0]$ and $[0,4]$, $2(\beta+\gamma)$ instances of $[2,2]$ and $\delta+\epsilon$ of $[3,1]$ and $[1,3]$.
The number of X-blocks remaining is thus

$$\binom{2N}{\alpha+\beta+\epsilon,\alpha+\beta+\epsilon,2(\gamma+\delta)}\times\\ \binom{2N}{\alpha+\beta+\delta,\alpha+\beta+\delta,2(\gamma+\epsilon)}\times\\ \binom{2N}{\alpha,\alpha,2(\beta+\gamma),\delta+\epsilon,\delta+\epsilon}.$$

Each of the remaining blocks will have a value of

$$(V_{220}V_{004})^{6\alpha}(V_{022}^2)^{6\beta}(V_{112}^2)^{6\gamma}(V_{112}V_{013})^{6\delta}(V_{112}V_{013})^{6\epsilon}$$

Agani, the overall value can be manipulated in such a way that it does not depend on the individual values of $\alpha,..\epsilon$. Hence, we approximate M'' by its largest term and set $\alpha..\epsilon$ as being the same as in this term. The overall value is thus a fraction $\frac{1}{N}$ times the product of the two previous statements.
Using Stirling's formula, rearranging and taking the square root, we obtain approximatelyWe have $\alpha,\beta,\gamma,\delta,\epsilon$ such that $\alpha+\beta+\gamma+\delta+\epsilon=N$ and $0\le\alpha,\beta,\gamma,\delta,\epsilon\le 1$.

$$2^{3N}\binom{N}{\alpha+\beta+\epsilon,\gamma+\delta}(\frac{V_{112}^3}{2})^{\gamma+\delta}(V_{022}^3)^{\alpha+\beta+\epsilon}\times$$
$$\binom{N}{\alpha+\beta+\delta,\gamma+\epsilon}(\frac{V_{112}^3}{2})^{\gamma+\epsilon}(V_{022}^3)^{\alpha+\beta+\delta}\times$$
$$\binom{N}{\alpha,\beta+\gamma,\delta+\epsilon}(V_{013}^3)^{\delta+\epsilon}(V_{004})^{\alpha}(\frac{V_{022}^3}{2})^{\beta+\gamma}\times$$
$$\frac{1}{V_{022}^{3N}}$$

Setting

$$\begin{aligned}
\alpha &= \frac{V_{004}N}{V_{103}^3+V_{022}^3/2+1}\\
\delta+\epsilon &= \frac{V_{013}^3N}{V_{103}^3+V_{022}^3/2+1}\\
\beta+\gamma &= \frac{V_{022}^3N}{2V_{103}^3+V_{022}^3+2}\\
\gamma+\delta &= \gamma+\epsilon=\frac{V_{112}^3N}{2V_{022}^3+V_{112}^3}\\
\alpha+\beta+\epsilon &= \alpha+\beta+\delta=\frac{V_{022}^3N}{V_{022}^3+V_{112}^3/2}
\end{aligned}$$

makes the expressions approximately equal to

$$(2(\frac{V_{112}^3}{2}+V_{022}^3)^{2/3}(V_{013}^3+\frac{V_{022}^3}{2}+V_{004})^{1/3})^{3N}\times\frac{1}{V_{022}}^{3N}.$$

Taking $3N$th roots and substituting in the appropriate variables, we get the desired result.
Finally, proving that $\alpha,\beta,\gamma,\delta,\epsilon$ satisfy the constraints, we get, on setting $q=6$ and $\tau=2.373691893/3$, that

$$\begin{aligned}
\alpha &= 0.0003147913N\\
\beta &= 0.2983016300N\\
\gamma &= 0.5866546200N\\
\delta &= 0.05736447930N\\
\epsilon &= 0.05736447930N
\end{aligned}$$

which satisfy the constraints, as required. □

Chapter 5

Group-Theoretic Methods for Determining ω

In [9], Cohn and Umans devised a new way of representing Matrix Multiplication, that is via a Group-theoretic means. They show that is is possible to represent a $m \times n$ by $n \times p$ matrix multiplication by taking subsets of size m, n, p satisfying a particular property, and, by multiplying elements of the group algebra $\mathbb{C}G$ together (via a Discrete Fourier Transform), showing that one can read off the appropriate values of the matrix product from this result.

This group-theoretic framework has, so far, not provided any algorithms to reduce ω, but importantly it has produced several conjectures that will lead to $\omega = 2$ if any of them are proven. In this chapter we will explain how groups can realize matrix multiplications, show some examples of this, and show how we can derive Coppersmith and Winograd's algorithms in this context. Finally, we explain some conjectures which would imply $\omega = 2$.

Some of the proofs require a knowledge of basic representation theory, so we will explore some background on that first.

5.1 Background to Representation Theory

This section uses definitions and theorems found James and Liebeck [17] : for a more in-depth introduction to Representation Theory, please consult this book.

Representation Theory is the study of writing groups as a set of matrices.

Definition 9. *The* Group Algebra $\mathbb{C}G$ *(or equivalently* $\mathbb{R}G$*) of a group* $G = \{1, g_1, g_2..\}$ *is the set*

$$\{a_0.1 + a_1.g_1 + a_2g_2 + ..\}$$

where the a_i *can take any values in* $\mathbb{C}$ *(or* $\mathbb{R}$*).*

Addition in $\mathbb{C}G$ *(*$\mathbb{R}G$*) follows the rule:*

$$\sum_i a_ig_i + \sum_i b_ig_i = \sum_i (a_i + b_i)g_i$$

and multiplication follows the rule: where the a_i *can take any values in* $\mathbb{C}$ *(or* $\mathbb{R}$*).*

Addition in $\mathbb{C}G$ *(*$\mathbb{R}G$*) follows the rule:*

$$\sum_i a_i g_i \times \sum_i b_i g_i = \sum_i (\sum_{j,k:g_j g_k = g_i} a_j b_k) g_i .$$

Definition 10. *A* Representation of a group G over a field F*is a homomorphism* ρ *from* G *to* $GL(n, F)$ *(the set of* $n \times n$ *matrices with non-zero determinant and with entries in* F*) for some* n*. The degree of* ρ *is the integer* n*.*

We write the action of ρ on $g \in G$ as $g\rho$. Since ρ is a homomorphism it must follow that

$$(gh)\rho = (g\rho)(h\rho)$$

for all $g, h \in G$ and

$$1\rho = I_n$$

where I_n is the $n \times n$ identity matrix.

Definition 11. *The* Kernel of a Representation *is the set* $\{g \in G | g\rho = I_n\}$*.*

Now, we let $V = F^n$ be the set of row vectors of length n with entries in F. We then have that the product $v(g\rho)$ is also a vector of length N. We say that V is an FG-module if a multiplication vg is defined, satisfying the following conditions for all $u, v \in V, \mu \in F$ and $g, h \in G$.

- $vg \in V$
- $v(gh) = (vg)h$
- $v1 = v$
- $(\mu v)g = \mu(vg)$
- $(u + v)g = ug + vg$

We now define submodules and irreducibility.

Definition 12. *Let* V *be an* FG*-module. A subset* W *of* V *is an* FG-submodule *of* V *if* W *is a subspace and* $wg \in W$ *for all* $w \in W$ *and all* $g \in G$*.*

In this case W is also an FG-module.

Definition 13. *An* FG*-module* V *is said to be* irreducible *if its only* FG*-submodules are* $\{0\}$ *and* V*. If it has other submodules, then it is* reducible*. We say that the representation* ρ *is (ir)reducible if* V *is (ir)reducible.*

We need to show that some representations are distinct from others. In order to do this, we define what it means for two representations to be isomorphic.

Definition 14. *Let* V *and* W *be* FG*-modules. A function* $\mathcal{G} : V \to W$ *is said to be an* FG*-homomorphism if* $\mathcal{G}$ *is a linear transformation and*

$$(vg)\mathcal{G} = (v\mathcal{G})g$$

for all $v \in V, g \in G$*.*

Definition 15. *Let V and W be FG-modules. We call a function $\mathcal{G} : V \to W$ and FG-isomorphism if $\mathcal{G}$ is an FG-homomorphism and invertible. If such an FG-isomorphism exists, then V and W are isomorphic.*

The number of (non-isomorphic) irreducible representations is related to the size of the group. In order to show this, we first state Maschke's Theorem.

Theorem 26. *(Maschke) If G is a finite group, and V is a $\mathbb{R}G$ or $\mathbb{C}G$-module with $\mathbb{R}G$ or $\mathbb{C}G$ submodule U then there is a $\mathbb{R}G$ or $\mathbb{C}G$ submodule W such that*

$$V = U \oplus W.$$

The proof can be found in Chapter 8 of James and Liebeck [17].
An iterative argument shows that every FG-module V (for $F = \mathbb{C}$ or $\mathbb{R}$) may be written as the sum of irreducible FG-submodules.
We now show that for abelian G, the dimension of every irreducible FG-module is equal to 1.

Lemma 27. *(Schur) (part 2) If, for an irreducible $\mathbb{C}G$-module V, $\mathcal{G} : V \to V$ is a $\mathbb{C}G$-isomorphism, then $\mathcal{G}$ is a scalar multiple of the identity endomorphism 1_V.*

Proposition 4. *For abelian G, the dimension of every irreducible $\mathbb{C}G$-module V is equal to 1.*

Proof. Let $g, h \in G$. Since G is abelian we have

$$vgh = vhg$$

for all $h, g \in G$. Hence $v \to vh$ is a $\mathbb{C}G$-homomorphism. By Lemma 27, this means that

$$vh = \mu_h v$$

for all $v \in V$, and some $\mu_h \in \mathbb{C}$. This implies that every subspace of V is a $\mathbb{C}G$-submodule. Since V is irreducible, it must be the case that $dim(V) = 1$. □

Finally, we use two more theorems to show that

$$|G| = \sum_{i=1}^{k} dim(V_k)^2$$

where the V_k are the complete set of non-isomorphic irreducible $\mathbb{C}G$-modules.

Theorem 28. *If we have $\mathbb{C}G = \bigoplus_i^r U_i$, a direct sum of irreducible $\mathbb{C}G$-submodules, and U any irreducible $\mathbb{C}G$-module, then the number of $\mathbb{C}G$-modules U_i with $U_i \simeq U$ is equal to $dim(U)$.*

Proof in James and Liebeck [17] (chapter 11).

Theorem 29. *Let V_k be the complete set of non-isomorphic irreducible $\mathbb{C}G$-modules. Then*

$$|G| = \sum_{i=1}^{k} V_k^2$$

Proof. We start by noting from Maschke that we may rewrite $\mathbb{C}G$ in terms of $\mathbb{C}G$-submodules

$$\mathbb{C}G = \bigoplus_{i=1}^{r} U_i$$

where the U_i are the irreducible submodules. By the previous theorem, the number of irreducible $\mathbb{C}G$-modules $U_j \simeq V_i$ is $dim(V_i)$.
Then we have, taking the dimension of both sides, that

$$dim(\mathbb{C}G) = \sum_{i=1}^{k} (dim(V_i))^2$$

and since $dim(\mathbb{C}G) = |G|$ the result follows. □

Corollary 30. *Multiplication of two elements in the group algebra is isomorphic to* $\oplus_i \langle d_i, d_i, d_i \rangle$.

Proof. Use of the Discrete Fourier Transform to multiply two elements of $\mathbb{C}G$ together will show that the two are isomorphic. □

5.2 The Triple Product Property

In [9], it was shown that if three subsets of a group G satisfy the *Triple Product Property*,then the group is capable of simulating matrix multiplications. We let $Q(S)$ be the right quotient set of S, a subset of a group G, that is

$$Q(S) = \{s_1 s_2^{-1} : s_1, s_2 \in S\}.$$

Of course, if S is a subgroup of G, we see that $Q(S) = S$. For now, though, we only consider arbitrary subsets.

Definition 16. *We say that three subsets S_1, S_2, S_3 of a group satisfy the* Triple Product Property *if, for $q_i \in Q(S_i)$,*

$$q_1 q_2 q_3 = 1 \Rightarrow q_1 = q_2 = q_3 = 1.$$

Why is this condition important? It arises because, as previously said, the matrix multiplication is embedded in multiplication of the group algebra $\mathbb{C}[G]$. We choose three subsets of G, S, T and U, all of which satisfy the Triple Product Property. We then consider the product

$$(\sum_{s \in S, t \in T} A_{st} s^{-1} t)(\sum_{t' \in T, u \in U} B_{t'u} t'^{-1} u)$$

We have that

$$ss'^{-1} tt'^{-1} uu'^{-1} = 1$$

if and only if

$$ss'^{-1} = tt'^{-1} = uu'^{-1} = e.$$

It must therefore follow that $s = s'$, $t = t'$ $u = u'$, and performing appropriate operations means that

$$s^{-1}tt^{-1}u = s^{-1}u.$$

This means that, for a given s, u the coefficient of $s^{-1}u$ in our product as described above is

$$\sum_{t\in T} A_{s,t}B_{t,u}$$

which is indeed a matrix product. It follows from the above arguments that the number of operations required to multiply to elements of $\mathbb{C}[G]$ is more than or equal to the number of operations required to multiply an $|S| \times |T|$ matrix with a $|T| \times |U|$ one.

We say that the group G *realizes* $\langle |S|, |T|, |U|\rangle$. It is easily shown that if G realises $\langle m, n, p\rangle$, then G will also realise any permutation of m, n, p.

In order to put this into context, we need to show how a group G realizing $\langle m, n, p\rangle$ relates to ω. To do this, we need to use theorem 29:

$$\mathbb{C}[G] \simeq \oplus_i \langle d_i, d_i, d_i\rangle$$

where the $\{d_i\}$ are the dimensions of the $\mathbb{C}G$-submodules of G.

Theorem 31. *(Theorem 4.1 of [9]) If a group G with character degrees $\{d_i\}$ realizes $\langle m, n, p\rangle$ then*

$$(mnp)^{\omega/3} \leq \sum_i d_i^{\omega}$$

Proof. We assume that G realizes $\langle m, n, p\rangle$. We know from previously that the number of operations required to perform $\langle m, n, p\rangle$ is at most the number of operations required to multiply two elements of $\mathbb{C}[G]$, which we know to be isomorphic to $\oplus_i \langle d_i, d_i, d_i\rangle$.

Hence

$$\langle m, n, p\rangle \leq \oplus_i \langle d_i, d_i, d_i\rangle.$$

That is to say, performing $\oplus_i \langle d_i, d_i, d_i\rangle$ is sufficient to obtain the product $\langle m, n, p\rangle$. We raise this expression to the lth tensor power. This yields

$$\langle m^l, n^l, p^l\rangle \leq \oplus_{i_l} \langle d_{i_1}...d_{i_l}, d_{i_1}...d_{i_l}, d_{i_1}...d_{i_l}\rangle,$$

that is, there are i^l different possible matrix multiplications (all square) to be obtained from raising it to the lth power. From previous results the rank of the right hand side is less than or equal to

$$C\sum_{i_l}(d_{i_1}...d_{i_l})^{\omega+\epsilon} = C(\sum_i d_i^{\omega+\epsilon})^l.$$

We also have that $(nmp)^{l\omega/3} \leq R(\langle m^l, n^l, p^l\rangle)$, so taking the rank of both sides of the above expression yields

$$(nmp)^{l\omega/3} \leq C(\sum_i d_i^{\omega+\epsilon})^l.$$

If we let l grow, and take lth roots of both sides, we obtain the desired result. □

We see that this does not provide non-trivial estimates for ω unless $nmp > \sum_i d_i^3$. In [10], a group that satisfies this condition is constructed.

The Triple Product Property can be extended so that, for larger groups, we can have many matrix multiplications simulated by a single multiplication in the group algebra. We call this the *Simultaneous Triple Product Property*

Definition 17. *We say that n triples of subsets A_i, B_i, C_i of a group G satisfy the* Simultaneous Triple Product Property *if*

- *for each i the three subsets A_i, B_i, C_i satisfy the triple product property, and*
- *for all i, j, k,*
$$a_i(a'_j)^{-1}b_j(b'_k)^{-1}c_k(c'_i)^{-1} = 1 \Rightarrow i = j = k$$
for $a_i \in A_i, a'_j \in A_j, b_j \in B_j, b'_k \in B_k, c_k \in C_k, c'_i \in C_i$.

We see from this property that if

$$(a'_j)^{-1}b_j(b'_k)^{-1}c_k = a_i^{-1}c'_i$$

then $i = j = k$ so

$$(a'_i)^{-1}b_i(b'_i)^{-1}c_i = a_i^{-1}c'_i.$$

and since the A_i, B_i, C_i satisfy the triple product property, we get that $a = a', b = b', c = c'$. Thus, if we multiply two elements in the group algebra, we get that the coefficient of $a_i^{-1}c'_i$ is

$$\sum_{b_i \in B_i} A_{a_i^{-1}, b_i} B_{b_i^{-1}c'_i}$$

as the matrix product.

The following lemma shows that the Simultaneous Triple Product Property also applies to group products (this lemma is found in [10], but there, no proof is provided):

Lemma 32. *If n triples of subsets $A_i, B_i, C_i \subseteq H$ and n' triples of subsets $A'_j, B'_j, C'_j \subseteq H'$ all satsify the Simultaneous Triple Product Property, then so do the nn' triples of subsets*

$$A_i \times A'_j, B_i \times B'_j, C_i \times C'_j \subseteq H \times H'.$$

Proof. We first show that three sets $A_i \times A'_j, B_i \times B'_j, C_i \times C'_j \subseteq H \times H'$ satisfy the triple product property. Consider $(a_1, a'_1), (a_2, a'_2) \in A_i \times A'_j$, $(b_1, b'_1), (b_2, b'_2) \in B_i \times B'_j$ and $(c_1, c'_1), (c_2, c'_2) \in C_i \times C_j$. Looking at the equation

$$(a_1, a'_1)(a_2, a'_2)^{-1}(b_1, b'_1)(b_2, b'_2)^{-1}(c_1, c'_1)(c_2, c'_2)^{-1} = 1 \in H \times H'$$

we see that

$$(a_1(a_2)^{-1}b_1(b_2)^{-1}c_1(c_2^{-1}), a'_1(a'_2)^{-1}b'_1(b'_2)^{-1}c'_1(c_2'^{-1})) = (1, 1) \in H \times H'.$$

We have two sepearate equations to solve. However, since A_i, B_i, C_i and A'_j, B'_j, C'_j, it follows that $a_1(a_2)^{-1}b_1(b_2)^{-1}c_1(c_2^{-1}) = 1$ only if $a_1 = a_2, b_1 = b_2, c_1 = c_2$ and $a'_1(a'_2)^{-1}b_{1'}(b'_2)^{-1}c'_1(c_2'^{-1}) = 1$ only if $a'_1 = a'_2, b'_1 = b'_2, c'_1 = c'_2$. Hence $(a_1, a'_1) = (a_2, a'_2)$ and so on, and so the triple product property is preserved.

To prove the second statement, consider the following subsets of $H \times H'$:

$$\begin{array}{cc} A_{ii'} = A_i \times A'_{i'} & A_{jj'} = A_j \times A'_{j'} \\ B_{jj'} = B_j \times B'_{j'} & B_{kk'} = B_k \times B'_{k'} \\ C_{kk'} = C_k \times C'_{k'} & C_{ii'} = C_i \times C'_{i'}. \end{array}$$

Let $a_{ii'} \in A_{ii'}, a_{jj'} \in A_{jj'}, b_{jj'} \in B_{jj'}, b_{kk'} \in B_{kk'}, c_{kk'} \in C_{kk'}, c_{ii'} \in C_{ii'}$. We wish to show that

$$a_{ii'}(a_{jj'})^{-1}b_{jj'}(b_{kk'})^{-1}c_{kk'}(c_{ii'})^{-1} = 1 \in H \times H' \Rightarrow ii' = jj' = kk'.$$

Considering each coordinate seperately we have

$$\begin{array}{rcl} a_i(a_j)^{-1}b_j(b_k)^{-1}c_k(c_i)^{-1} & = & 1 \in H \\ a'_{i'}(a'_{j'})^{-1}b'_{j'}(b'_{k'})^{-1}c'_{k'}(c'_{i'})^{-1} & = & 1 \in H' \end{array}$$

Since all triples A_i, B_i, C_i and A'_j, B'_j, C'_j satisfy the Simultaneous Triple Product Property, it follows that $i = j = k$ and $i' = j' = k'$ and so $ii' = jj' = kk'$, and hence these groups satisfy the Simultaneous Triple Product Property. □

The usefulness of the simultaneous triple product property is borne out in the following theorem (compare with Schönhage's asymptotic sum inequality (theorem 7)):

Theorem 33. *(Theorem 5.5 in [10]) If a group H simultaneously realizes $\langle a_1, b_1, c_1\rangle, ..., \langle a_n, b_n, c_n\rangle$, and has character degrees $\{d_k\}$ then*

$$\sum_{i=1}^{n}(a_i b_i c_i)^{\omega/3} \leq \sum_k d_k^{\omega}.$$

We note that if H is abelian then the right hand side of this equals $|H|$- and hence we could regard $|H|$ as being the "rank" of the Matrix Multiplication, making this equation analogous to Schönhage's asymptotic sum inequality. In order to prove this, we need the following lemmas.

Lemma 34. *(lemma 1.1 of [10]) If we have non-negative real numbers $s_1, .., s_n$, and also that*

$$\binom{N}{\mu}\prod_{i=1}^{n} s_i^{\mu_i} \leq C^N$$

for all $N \in \mathbb{N}$ and all μ where μ is a vector of non-negative integers with $\sum_{i=1}^{n}\mu_i = N$ and $C > 0$, then

$$\sum_{i=1}^{n} s_i \leq C.$$

Proof. Fix N. We have for all μ with $\mu = \{\mu_1, .., \mu_n\}$ and $\sum_i \mu_i = N$ that

$$\binom{N}{\mu}\prod_{i=1}^{n} s_i^{\mu_i} \leq C^N$$

we sum over all possible values of μ (with $\sum_i \mu_i = N$), of which there are $\binom{N+n-1}{n-1}$:

$$\sum_{\mu} \binom{N}{\mu} \prod_{i=1}^{n} s_i^{\mu_i} \leq \binom{N+n-1}{n-1} C^N$$

which equals

$$(\sum_{i=1}^{n} s_i)^N \leq \binom{N+n-1}{n-1} C^N.$$

Letting N grow and taking Nth roots we get the desired result. □

Lemma 35. *(Theorem 7.1 of [10]) We suppose n triples of subsets $A_i, B_i, C_i \subseteq H$ satisfy the simultaneous triple product property, then the subsets $H_1, H_2, H_3 \subset G = H^n \rtimes Sym_n$ satisfy the triple product property:*

$$\begin{aligned} H_1 &= \{h\pi : \pi \in Sym_n, h_i \in A_i \forall i\} \\ H_2 &= \{h\pi : \pi \in Sym_n, h_i \in B_i \forall i\} \\ H_3 &= \{h\pi : \pi \in Sym_n, h_i \in C_i \forall i\} \end{aligned}$$

Proof. Let $h_i\pi_i, h_i'\pi_i' \in H_i$, and consider the triple product

$$h_1\pi_1(\pi_1')^{-1}(h_1')^{-1}h_2\pi_2(\pi_2')^{-1}(h_2')^{-1}h_3\pi_3(\pi_3')^{-1}(h_3')^{-1} = 1. \tag{5.1}$$

We must have

$$\pi_1(\pi_1')^{-1}\pi_2(\pi_2')^{-1}\pi_3(\pi_3')^{-1} = 1. \tag{5.2}$$

We then say that

$$\pi_1(\pi_1')^{-1} = \pi, \pi_1(\pi_1')^{-1}\pi_2(\pi_2')^{-1} = \rho,$$

which makes the above equivalent to

$$(h_3')^{-1}h_1((h_1')^{-1}h_2)^{\pi}((h_2')^{-1}h_3)^{\rho} = 1$$

where the superscripts denote that the actions of performing π and ρ have been performed on the group elements.
Thus, for each co-ordinate i,

$$[(h_3')^{-1}]_i[h_1]_i[(h_1')^{-1}]_{\pi(i)}[h_2]_{\pi(i)}[(h_2')^{-1}]_{\rho(i)}[h_3]_{\rho(i)} = 1$$

Since $A_i, A_{\pi(1)}, B_{\pi(i)}, B_{\rho(i)}, C_{\rho_i}, C_i$ are all parts of triples that satisfy the simultaneous triple product property, it must follow that $\pi(i) = \rho(i) = i$, meaning that $\pi = \rho = 1$. Since A_i, B_i, C_i satisfy the triple product property, it then follows that

$$h_1(h_1')^{-1}h_2(h_2')^{-1}h_3(h_3')^{-1} = 1$$

implies that $h_1 = h_1', h_2 = h_2', h_3 = h_3'$. Thus the three sets described above satisfy the Triple Product Property. □

Lemma 36. *If $\{d_k\}$ are the character degrees of a finite group H and $\{c_j\}$ are the character degrees of $Sym_n \ltimes H^n$, then*

$$\sum_j c_j^\omega \leq (n!)^{\omega-1}(\sum_k d_k^\omega)^n$$

Proof. If H is abelian, we use the fact that

$$\sum_j c_j^2 = n!|H|^n.$$

Since $c_j \leq n!$, we multiply both sides by $(n!)^{\omega-2}$. Since

$$\sum_j c_j^\omega \leq (n!)^{\omega-2}\sum_j c_j^2$$

the lemma holds for abelian H.
The proof for non-abelian H can be found in [10], where it is lemma 1.2. We do not go into detail here as it relies on more advanced representation theory. □

Lemma 37. *(7.2 in [10]). If H is a finite group with character degrees $\{d_k\}$ and n triples of subsets $A_i, B_i, C_i \subseteq H$ satisfying the simultaneous triple product property, then*

$$n(\prod_i(|A_i||B_i||C_i|)^{\omega/3})^{1/n} \leq \sum_k d_k^\omega$$

Proof. From lemma 35, we may build subsets of G of size $n!\prod_i |A_i|$, $n!\prod_i |B_i|$ and $n!\prod_i |C_i|$. We have from theorem 31 that

$$((n!)^3 \prod_i |A_i||B_i||C_i|)^{\omega/3} \leq \sum_j c_j^\omega$$

where the c_j are the character degrees of G. Lemma 36 tells us that the right hand side is at most

$$(n!)^{\omega-1}(\sum_k d_k^\omega)^n.$$

We divide both sides by $(n!)^\omega$ to obtain

$$(\prod_i |A_i||B_i||C_i|)^{\omega/3} \leq \frac{1}{n!}(\sum_k d_k^\omega)^n$$

To obtain the desired inequality, we replace H with H^t, and similarly n with n^t. This gives

$$(\prod_i |A_i||B_i||C_i|)^{tn^{t-1}\omega/3} \leq \frac{1}{n^t!}(\sum_k d_k^\omega)^n,$$

we then take tn^tth roots and let $t \to \infty$ to get the desired statement (via approximation of the factorial term using Stirling's formula). □

We can now prove theorem 33.

Proof. We raise H to the Nth power, and we take subsets of H^N $A'_j, B'_j, C'_j \subseteq H^N$. To create these subsets we choose a vector in $\mathbb{Z}^n$, $\mu = \mu_1, .., \mu_n$, $\mu \geq 0, \sum_{i=1}^n \mu_i = N$. We set $A'_j = \prod_{i=1}^n A_i^{\mu_i}, B'_j = \prod_{i=1}^n B_i^{\mu_i}, C'_j = \prod_{i=1}^n C_i^{\mu_i}$. There are $\binom{N}{\mu}$ such triples, and $|A'_j||B'_j||C'_j| = \prod_{i=1}^n (a_i b_i c_i)^{\mu_i}$.
We apply lemma 37 to these triples to obtain

$$\binom{N}{\mu}(\prod_{i=1}^n (a_i b_i c_i)^{\mu_i})^{\omega/3} \leq (\sum_k d_k)^N.$$

We apply lemma 34 to obtain the desired inequality. □

We seek groups, and subsets of these groups that satisfy the Simultaneous Triple Product Property. Using objects called *Uniquely Solvable Puzzles*, introduced in [10], it is indeed possible to generate such groups.

Definition 18. *A* Uniquely Solvable Puzzle *of width k is a subset $U \subseteq \{1,2,3\}^k$ satisfying the following property:*
For all bijections $\pi_1, \pi_2, \pi_3 \in Sym(U)$, either $\pi_1 = \pi_2 = \pi_3$ or else there exist $u \in U$ and $i \in \{1, ..k\}$ such that at least two of $(\pi_1(u))_i = 1, (\pi_2(u))_i = 2$ or $(\pi_3(u))_i = 3$ hold.

That is, if we are given any three bijections from the symmetric group $Sym(U)$, our task is to find a unique u such that this condition holds.
In chapter 2, we showed that the number of ways in which one could divide $3k$ objects into three subsets of size k, (such that no two divisions shared the same first, second or third subset and that choosing the first, second and third subsets from three different divisions could form a new division) was more than $(\frac{27}{4})^{k(1-\epsilon)}$ for large k and $\epsilon > 0$. We show now that a set U of such partitions u is a uniquely solvable puzzle.

Proof. Suppose we have a set of $3k$ objects. We wish to divide them into three sets of size k: we do so by labelling k objects with 1, k objects with 2 and k objects with 3. Let $\pi_1, \pi_2, \pi_3 \in Sym(U)$. We look at the sets
$I_1 = \{i : (\pi_1(u))_i = 1\}, I_2 = \{i : (\pi_2(u))_i = 2\}, I_3 = \{i : (\pi_3(u))_i = 3\}$ for each $u \in U$. Each of these sets will have size k. If there is a $u \in U$ such that
$I_1 \cup I_2 \cup I_3 = \{1, .., 3k\}$, then the three sets I_1, I_2, I_3 are disjoint and so can themselves form an element of U: however since this violates one of our constraints, there is no such element in U and thus $\pi_1 = \pi_2 = \pi_3$. Therefore U is a USP. □

5.2.1 Using USPs to Generate Subsets

We now show how we use Uniquely Solvable Puzzles to generate subsets that satisfy the Simultaneous Triple Product Property.

Definition 19. *Let H be an abelian group of finite order. An $\mathcal{H}$-chart $\mathcal{C} = (\Gamma, A, B, C)$ consists of a finite set of symbols Γ, together with three mappings $A, B, C : \Gamma \to 2^H$ such that for each $x \in \Gamma$, the sets $A(x), B(x), C(x)$ satisfy the Triple Product Property. Let $\mathcal{H}(\mathcal{C}) \subseteq \Gamma^3$ denote the set of ordered triples (x, y, z) such that*

$$0 \notin A(x) - A(y) + B(y) - B(z) + C(z) - C(x)$$

where 0 denotes the identity element of the group H.

A local $\mathcal{C}$-USP of width k *is a subset* $U \subseteq \Gamma^k$ *such that for each ordered triple* $(u, v, w) \in U^3$ *with* u, v, w *not all equal, there exists* $i \in \{1, .., k\}$ *such that* $(u_i, v_i, w_i) \in \mathcal{H}(\mathcal{C})$.

Informally, an $\mathcal{H}$-chart, together with a local $\mathcal{C}$-USP, is a means of generating groups that satisfy the simultaneous triple product property.
The condition that

$$0 \notin A(x) - A(y) + B(y) - B(z) + C(z) - C(x)$$

is equivalent to the triple product property statement (but we use additive notation here as we are dealing with Abelian groups).
It is easy to check that Local USPs are also USPs, and that they can achieve the same size as USPs.
One such $\mathcal{H}$ chart is given below, it is equivalent to Coppersmith and Winograd's "easy" algorithm in [12]. Here, $\hat{H} = Cyc_l \backslash \{0, 1\}$ and $\mathcal{C} = (\{1, 2, 3\}, A, B, C)$ with A, B, C defined as follows

$$\begin{array}{rcc} A(1) = \{0\} & B(1) = -\hat{H} & C(1) = \{0\} \\ A(2) = \{1\} & B(2) = \{0\} & C(2) = \hat{H} \\ A(3) = \hat{H} & B(3) = \{0\} & C(3) = \{0\} \end{array}$$

Finally, it remains to show that the sets generated by this method satisfy the simultaneous triple product property:
we define

$$A_u = \prod_{i=1}^{k} A(u_i) \quad B_u = \textstyle\prod_{i=1}^{k} B(u_i) \quad C_u = \displaystyle\prod_{i=1}^{k} C(u_i).$$

We thus have $|U|$ sets of three subsets of H^k.

Theorem 38. *(Theorem 6.6 in [10], the proof there is omitted) These subsets of* H^k *satsify the simultaneous triple product property.*

Proof. First, we show that, for a fixed u, the three sets A_u, B_u, C_u satisfy the triple product property. Let $a_1, a_2 \in A_u, b_1, b_2 \in B_u, c_1, c_2 \in C(u)$. Then, we wish to show that

$$a_1 - a_2 + b_1 - b_2 + c_1 - c_2 = 0$$

only if $a_1 = a_2, b_1 = b_2, c_1 = c_2$. We consider each u_i individually. If $u_i = 1$ then it is automatic that $(a_1)_i = (a_2)_i$ and $(c_1)_i = (c_2)_i$. So $(a_1 - a_2 + b_1 - b_2 + c_1 - c_2)_i = 0$ only if $(b_1)_i = (b_2)_i \in -\hat{H}$. Analogous arguments follow if $u_i = 2$ or 3. It follows that $(a_1)_i = (a_2)_i, (b_1)_i = (b_2)_i, (c_1)_i = (c_2)_i$ for all i and hence $a_1 = a_2, b_1 = b_2, c_1 = c_2$.
It remains to show the second criterion. This follows from the definition of an H-chart: there is an i such that for every triple $(u, v, w) \in U^3$, $(u_i, v_i, w_i) \in \mathcal{H}(\mathcal{C})$ where $\mathcal{H}(\mathcal{C})$ is the set of ordered triples with

$$0 \notin A(x) - A(y) + B(y) - B(z) + C(z) - C(x).$$

It follows that one cannot create the identity of H^k from the subsets generated by distinct u, v, w, and hence the simultaneous triple product property applies here. □

Finally, we use these groups to get a suitable estimate for ω. Since H is abelian, theorem 33 reduces to

$$\sum_{i=1}^{k}(a_i b_i c_i)^{\omega/3} = |H|.$$

The sizes of A_u, B_u and C_u are $(l-2)^k$ and the size of H is l^{3k}. Since there are more than $(\frac{27}{4})^{k(1-\epsilon)}$ possible groups to be created, we obtain

$$(\frac{27}{4})^k (l-2)^{\omega k} = l^{3k}$$

and letting k grow, taking kth roots and setting $l = 10$, we obtain $\omega \leq 2.403...$
We can obtain the same estimates that Coppersmith and Winograd [12] obtain using similar methods.
To obtain $\omega < 2.38719$, we set $H = Cyc_l$ and $\widehat{H} = H \backslash \{0, 1\}$ and we define our $\mathcal{H}$-chart as follows:

$$\begin{array}{rll} A(1) = \{0\} & B(1) = -\widehat{H} & C(1) = \{0\} \\ A(2) = \{1\} & B(2) = \{0\} & C(2) = \widehat{H} \\ A(3) = \widehat{H} & B(3) = \{0\} & C(3) = \{0\} \\ A(4) = \{1\} & B(4) = \{0\} & C(4) = \{0\} \\ A(5) = \{1\} & B(5) = \{1\} & C(5) = \{1\} \\ A(6) = \{1\} & B(6) = \{0\} & C(6) = \{1\}. \end{array}$$

We thus need a different definition of our local Uniquely Solvable Puzzle. We first need to define our $\mathcal{H}$. Our local USP U is a subset of $\{1, 2, 3, 4, 5, 6\}^k$ such that, for every ordered $(u, v, w) \in U^3$, there is an i such that $(u_i, v_i, w_i) \in \mathcal{H}$.
We define three functions $\alpha, \beta, \gamma : \{1, 2, 3, 4, 5, 6\} \to \{0, 1, 2\}$:

$$\alpha(2) = \alpha(3) = 1, \alpha(4) = 2, \alpha(1) = \alpha(5) = \alpha(6) = 0$$

$$\beta(1) = \beta(3) = 1, \beta(5) = 2, \beta(2) = \beta(4) = \beta(6) = 0$$

$$\gamma(1) = \gamma(2) = 1, \gamma(6) = 2, \gamma(3) = \gamma(4) = \gamma(5) = 0$$

Then, $\mathcal{H}$ is the set of all triples $(u, v, w) \in \{1, 2, 3, 4, 5, 6\}^3$ such that

$$\alpha(u) + \beta(v) + \gamma(w) < 2.$$

There are 81 such elements, for example $(2, 2, 4) \in \mathcal{H}$.
From Coppersmith and Winograd [12], it can be shown that the maximum size of U with $k = 3N$ is roughly

$$\binom{3N}{L, N+L, 2N-2L}$$

if, in each $u \in U$ we allow $N - L$ occurences each of 1,2 and 3 and L occurences each of 4,5, and 6.

The subsets

$$A_u = \prod_{i=1}^{k} A(u_i) \quad B_u = \textstyle\prod_{i=1}^{k} B(u_i) \quad C_u = \displaystyle\prod_{i=1}^{k} C(u_i)$$

thus all satisfy the simultaneous triple product property. Using theorem 7, setting $l = 8$ and $L = 0.048N$, we get the desired upper bound of ω.
This algorithm is equivalent to Coppersmith and Winograd's "Complicated" algorithm (Section 7 of [12]); the following correspondence applies:

$$\begin{aligned}
1 &\rightarrow X^{[0]}Y^{[1]}Z^{[1]} \\
2 &\rightarrow X^{[1]}Y^{[0]}Z^{[1]} \\
3 &\rightarrow X^{[1]}Y^{[1]}Z^{[0]} \\
4 &\rightarrow X^{[2]}Y^{[0]}Z^{[0]} \\
5 &\rightarrow X^{[0]}Y^{[2]}Z^{[0]} \\
6 &\rightarrow X^{[0]}Y^{[0]}Z^{[2]}
\end{aligned}$$

The USP condition ensures that all the matrix products are independent. Here, l is equivalent to $q + 2$ in Coppersmith and Winograd's version.

5.3 Using Group Theory to show $\omega = 2$

We have objects called *Strong Uniquely Solvable Puzzles*, defined in [10], which are similar to USPs with one important exception:

Definition 20. *A* Strong Uniquely Solvable Puzzle *of width k is a subset $U \subseteq \{1, 2, 3\}^k$ satisfying the following property:
For all bijections $\pi_1, \pi_2, \pi_3 \in Sym(U)$, either $\pi_1 = \pi_2 = \pi_3$ or else there exist $u \in U$ and $i \in \{1, ..k\}$ such that exactly two of $(\pi_1(u))_i = 1, (\pi_2(u))_i = 2$ or $(\pi_3(u))_i = 3$ hold.*

In our comparison with dividing $3k$ objects into 3 sets of size k, we have that this is equivalent what we had before, with the additional restriction that each object can only be placed in a maximum of two sets, throughout the USP.
It is conjectured in [10] that, given ϵ, one can find a k large enough such that the number of elements in the strong USP is greater than $\binom{27}{4}^{k(1-\epsilon)}$. We explain how [10] shows how this would show $\omega = 2$.
We let H be the abelian group of functions from $U \times \{1, .., k\}$ to the cyclic group Cyc_m. The symmetric group $Sym(U)$ acts on H in the following manner, for $\pi \in Sym(U), h \in H, u \in U, i \in \{1, .., k\}$:

$$\pi h(u, i) = h(\pi^{-1}(u), i)$$

Let G be the semidirect product $H \rtimes Sym(U)$. We then define three subsets $S_1, S_2, S_3 \subseteq G$.
Set

$$S_i = h\pi : h(u, j) \neq 0 \text{ iff } u_j = i$$

for all $u \in U$ and $j \in \{1, .., k\}$.
It is shown in [10] that these three subsets satisfy the triple product property. Each of these subsets has size $|U!|(m-1)^{|U|k}$.

Using corollary 3.6 in [10], we have that

$$\omega \leq \frac{3\log(m)}{\log(m-1)} - \frac{3\log(|U|!)}{|U|k\log(m-1)}.$$

If the maximum size of the USP is greater than C^k for some C, then this becomes

$$\omega \leq 3\frac{\log(m) - \log(C)}{\log(m-1)}.$$

We thus have that if the size of strong USPs is the same as that for regular USPs, $C = \frac{3}{2^{2/3}}$, that, on setting $m = 3$, that $\omega = 2$. However, it is not known if such a strong USP exists.

In Proposition 3.8 of [10], it is shown that C is at least $2^{2/3}$. For regular USPs, C was roughly equal (using Stirling's Formula and Salem-Spencer sets) to the number of possible combinations of blocks that included the symbol 1. We have the additional restriction that within the USP, no entry can contain any more than two symbols. Unfortunately, the Salem-Spencer set method (as it is described in chapter 2 and in [12]) cannot detect and remove triples which violate this restriction: it can only deal with triples which share an X, Y or Z block. However, there is no known reason that we cannot remove an appropriate number of triples and be left with a number of blocks proportional to the number of X-blocks, thereby getting the required value of C.

Finally, another property, similar to the Simultaneous Triple Product Property, the *Simultaneous Double Product Property*, together with appropriately defined subgroups, can be used to how that $\omega = 2$, if certain, unproven conditions can be met. This is also defined and described in [10].

Two subsets S_1, S_2 of a group H satisfy the *Double Product Property* if

$$q_1 q_2 = 1 \Rightarrow q_1 = q_2 = 1$$

for $q_i \in Q(S_i)$.

Definition 21. *Then we say that n pairs of subsets A_i, B_i satisfy the* Simultaneous Double Product Property *if*

- *for all i, the pair A_i, B_i satisfies the double product property and*
- *for all i, j, k*

$$a_i(a'_j)^{-1}b_j(b'_k)^{-1} = 1 \Rightarrow i = k$$

where $a_i \in A_i, a'_j \in A_j, b_j \in B_j$ and $b'_k \in B_k$.

It is possible to define subsets of a group G which satisfy the triple product property using subsets which define the double product property. If

$$\Delta_n = \{(a, b, c) \in \mathbb{Z}^3 : a + b + c = n - 1, a, b, c \geq 0\}$$

and n pairs of subsets $A_i, B_i \subseteq H$ that satisfy the Simultaneous Double Product Property, then the following subsets of H^3 satsify the triple product property. Here, $v = \{v_1, v_2, v_3\} \in \Delta_n$:

$$\begin{aligned}\widehat{A}_v &= A_{v_1} \times \{1\} \times B_{v_3}\\ \widehat{B}_v &= B_{v_1} \times A_{v_2} \times \{1\}\\ \widehat{C}_v &= \{1\} \times B_{v_2} \times A_{v_3}.\end{aligned}$$

Lemma 39. *The above sets satsify the triple product property.*

Proof. Let $a, a' \in \widehat{A}_v$, $b, b' \in \widehat{B}_v$ and $c, c' \in \widehat{C}_v$. We then suppose that

$$a(a')^{-1}b(b')^{-1}c(c')^{-1} = 1 \in H^3.$$

We consider the first coordinate. Let $a_{v_1}, a'_{v_1} \in A_{v_1}$ and $b_{v_1}, b'_{v_1} \in B_{v_1}$. Then the product obtained in the first coordinate is

$$a_{v_1}(a'_{v_1})^{-1}b_{v_1}(b'_{v_1})^{-1}1(1)^{-1} = 1.$$

Now, since the pair A_{v_1}, B_{v_1} satisfied the Simulatenous Double Product Property, it must follow that $a_{v_1} = a'_{v_1}$ and $b_{v_1} = b'_{v_1}$. Doing this for all three coordinates will show that $a = a'$, $b = b'$ and $c = c'$, and hence these subsets satisfy the triple product property. □

This gives rise to a lemma, which we will use to prove an analogous theorem to Theorem 33. In order to prove this lemma, we must use a theorem from [9] (Theorem 3.3).

Theorem 40. *Let Δ_n be the set of triples (v_1, v_2, v_3) with $v_1, v_2, v_3 \geq 0 \in \mathbb{Z}$ and $v_1 + v_2 + v_3 = n - 1$. Then the sets*

$$\begin{aligned}H_1 &= \{\pi \in Sym(\Delta_n) : [\pi(v_1, v_2, v_3)]_1 = v_1 \forall v \in Sym(\Delta_n)\}\\ H_2 &= \{\pi \in Sym(\Delta_n) : [\pi(v_1, v_2, v_3)]_2 = v_2 \forall v \in Sym(\Delta_n)\}\\ H_3 &= \{\pi \in Sym(\Delta_n) : [\pi(v_1, v_2, v_3)]_3 = v_3 \forall v \in Sym(\Delta_n)\}\end{aligned}$$

satisfy the triple product property.

Proof. For $\pi_1, \pi'_1 \in H_1, \pi_2, \pi'_2 \in H_2, \pi_3, \pi'_3 \in H_3$, consider the triple product

$$\pi_1(\pi'_1)^{-1}\pi_2(\pi'_2)^{-1}\pi_3(\pi'_3)^{-1} = 1.$$

Set

$$\pi = \pi_1(\pi'_1)^{-1}, \rho = \pi_2(\pi'_2)^{-1}, \theta = \pi_3(\pi'_3)^{-1}$$

to obtain

$$\pi\rho\theta = 1.$$

It follows that $\pi \in H_1, \rho \in H_2, \theta \in H_3$.
Consider the element $(0, 0, n-1) \in \Delta_n$. Then $\theta((0, 0, n-1)) = (0, 0, n-1)$ since θ preserves the third co-ordinate. We then apply ρ to $(0, 0, n-1)$ to get $(i, 0, n-1-i)$ for some $i \geq 0$. Finally, we apply π. Since π preserves the first co-ordinate, and since $\pi\rho\theta = 1$, we must have that $i = 0$ and that $\pi = \rho = \theta = 1$. Similar work will show that $(n-1, 0, 0)$ and $(0, n-1, 0)$ are also "'fixed points"'.
Now, for any other point (v_1, v_2, v_3), we have that after performing θ and ρ, that the resulting point will be $(v_1 + i + j, v_2 - i, v_3 - j)$ for some $i \geq 0, j \in \mathbb{Z}$. Now, π fixes the

first coordinate, which we wish to equal v_1, so $i+j=0$, making $i=-j$. Therefore our point is $\pi(v_1, v_2-i, v_3+i)$. Now, we rank the triples in order, setting $(n-1,0,0)$ as the largest and $(0,0,n-1)$ as the smallest (so therefore $(0,1,n-2)$ is larger than $(0,0,n-1)$ in our ranking). Since $(0,0,n-1)$ was a fixed point, nothing can map to this. We suppose that all points smaller than (v_1,v_2,v_3) in our ranking are fixed, and so we cannot map to them. We therefore have that since h_3 sends (v_1,v_2,v_3) to (v_1+i, v_2-i, v_3). The i must be greater than or equal to 0, since an $i<0$ would result in h_3 mapping to a smaller, fixed point, which is not possible. So our point (v_1, v_2-i, v_3+i) must have $i=0$ as $i>0$ will map to a fixed point, thus $i=0$ and $\pi=\rho=\theta=1$. Therefore the Triple Product Property holds. □

Lemma 41. *(Theorem 4.3 of [10]. The proof is omitted in the paper) If n pairs of subsets $A_i, B_i \subseteq H$ satisfy the simultaneous double product property, then the following subsets of $G=(H^3)^{\Delta_n} \rtimes Sym(\Delta_n)$ satisfy the triple product property:*

$$\begin{aligned} S_1 &= \{\hat{a}\pi : \pi \in Sym(\Delta_n), \hat{a}_v \in \widehat{A}_v \ \forall \ v\} \\ S_2 &= \{\hat{b}\pi : \pi \in Sym(\Delta_n), \hat{b}_v \in \widehat{B}_v \ \forall \ v\} \\ S_3 &= \{\hat{c}\pi : \pi \in Sym(\Delta_n), \hat{c}_v \in \widehat{C}_v \ \forall \ v\} \end{aligned}$$

Proof. Consider the triple product

$$a_1\pi_{a_1}(\pi_{a_2})^{-1}(a_2)^{-1}b_1\pi_{b_1}(\pi_{b_2})^{-1}(b_2)^{-1}c_1\pi_{c_1}(\pi_{c_2})^{-1}c_2^{-1}=1$$

where $a_1\pi_{a_1}, a_2\pi_{a_2} \in S_1$, $b_1\pi_{b_1}, b_2\pi_{b_2} \in S_2$, $c_1\pi_{c_1}, c_2\pi_{c_2} \in S_3$.
We must have

$$\pi_{a_1}(\pi_{a_2})^{-1}\pi_{b_1}(\pi_{b_2})^{-1}\pi_{c_1}(\pi_{c_2})^{-1}=1,$$

so setting

$$\begin{aligned} \pi &= \pi_{a_1}(\pi_{a_2})^{-1} \\ \rho &= \pi_{a_1}(\pi_{a_2})^{-1}\pi_{b_1}(\pi_{b_2})^{-1} \end{aligned}$$

means that the original triple product is equivalent to

$$c_2^{-1}a_1(a_2^{-1}b_1)^{\pi}(b_2^{-1}c_1)^{\rho}=1.$$

Hence for each v

$$(c_2^{-1})_v(a_1)_v(a_2^{-1})_{\pi(v)}(b_1)_{\pi(v)}(b_2^{-1})_{\rho(v)}(c_1)_{\rho(v)}=1$$

If we consider only the first coordinate of v, we must have that

$$1 \in A_{v_1}(A_{(\pi(v))_1})^{-1}B_{(\pi(v))_1}(B_{(\rho(v))_1})^{-1}.$$

However, due to the fact all sets A_i, B_i satisfy the Simultaneous Double Product Property, the above can only hold if $v_1=(\rho(v))_1$. The map ρ must therefore be in the set of maps in $Sym(\Delta_n)$ which preserve the first coordinate.

If we consider the second coordinate of v, we must have that

$$1 \in A_{(\pi(v))_2}(A_{(\rho(v))_2})^{-1}B_{(\rho(v))_2}(B_{v_2})^{-1}.$$

Again, due to the fact all sets A_i, B_i satisfy the Simultaneous Double Product Property, the above can only hold if $v_2 = (\pi(v))_2$. Therefore the map π can only be in the set of maps in $Sym(\Delta_n)$ which preserve the second coordinate.
Finally, considering the third coordinate of v, we must have that

$$1 \in A_{(\rho(v))_3}(A_{v_3})^{-1}B_{v_3}(B_{(\pi(v))_3})^{-1}$$

which holds if and only if $(\rho(v))_3 = (\pi(v))_3$, that is if the two maps $\pi^{-1}\rho$ and $\rho^{-1}\pi \in Sym(\Delta_n)$ both preserve the third coordinate of v.
Recall that the subsets H_i of $Sym(\Delta_n)$ which preserve the ith coordinate satsify the triple product property. Thus, for $h_i \in H_i$, if $h_1h_2 = h_3$, we have that $h_1h_2(h_3)^{-1} = 1$, and since $h_i, h_i^{-1} \in Q(H_i)$, we must have that $h_1 = h_2 = h_3 = 1$, so $\pi = \rho = 1$.
The triple product becomes

$$a_1(a_2)^{-1}b_1(b_2)^{-1}c_1(c_2)^{-1} = 1$$

We look at the first entry in the vth coordinate:

$$1 \in a_{v_1}(a'_{v_1})^{-1}b_{v_1}(b'_{v_1})^{-1}$$

where

$$a_{v_1}, a'_{v_1} \in A_{v_1}, b_{v_1}, b'_{v_1} \in B_{v_1}.$$

Since all A_i, B_i satisfy the Double Product property, $a_{v_1} = a'_{v_1}$ and $b_{v_1} = b'_{v_1}$.
Analogous analysis on the second and third coordinates will yield that $(a_1)_{v_i} = (a_2)_{v_i}, (b_1)_{v_i} = (b_2)_{v_i}$ and $(c_1)_{v_i} = (c_2)_{v_i}$ for all $v \in \Delta_n, i \in \{1, 2, 3\}$. Hence the triple product property holds.

□

Theorem 42. *(Theorem 4.4 of [10]) If H is a finite group with character degrees $\{d_k\}$, and n pairs of subsets $A_i, B_i \subseteq H$ satisfy the Simultaneous Double Product Property, then*

$$\sum_{i=1}^{n}(|A_i||B_i|)^{\omega/2} \leq (\sum_k d_k^\omega)^{3/2}.$$

Proof. For an integer N set $A'_i = A_i^n$ and $B'_i = B_i^N$, and μ be a vector in $\{1, .., N\}^n$, with $\sum_{i=1}^n \mu_i = N$. There are $M = \binom{N}{\mu}$ pairs A'_i, B'_i such that $|A'_i||B'_i| = \prod_{i=1}^n (|A_i||B_i|)^{\mu_i} = L$. We then set $P = |\Delta_M|$, so that $P = \frac{M(M+1)}{2}$. We now use lemma 41, where the three subsets have size $P!L^P$, and a previous theorem to show that

$$(P!L^P)^\omega \leq (P!)^{\omega-1}(\sum_k d_k^\omega)^{3NP}.$$

We take $2P$th roots and let $k \to \infty$ to obtain

$$\binom{N}{\mu}(\prod_{i=1}^{n}(|A_i||B_i|)^{\mu_i})^{\omega/2} \leq (\sum_k d_k^\omega)^{3N/2}$$

The original statement then follows from lemma 34. □

We conjecture, as do the authors of [10], that there exist n subsets A_i, B_i of H, where H is an Abelian group of size $n^{2+o(1)}$such that $|A_i||B_i| \geq n^{2-o(1)}$. Placing these figures into the previous theorem will show that $\omega = 2$.

5.4 Relationship between ω and Group Algebra multiplication

Implicit in this work is the relationship between matrix multiplication and the multiplication of two elements of the group algebra $\mathbb{C}G$ with $|G| \leq n$. We show the relationship between the two problems.

Definition 22. *The* rank $r(G)$ of multiplication in the group algebra *is the minimum number of multiplications in* $\mathbb{C}$ *which are required to multiply two elements of* $\mathbb{C}G$*.*

Definition 23. *The* exponent of multiplication of the group algebra $\mathbb{C}G$ *is the minimum number* α *such that*

$$\max\{r(G) : |G| \leq n\} = O(n^{\alpha+\epsilon})$$

for any $\epsilon > 0$*.*

Theorem 43. *The exponent of matrix multiplication* $\omega = 2\alpha$*.*

Proof. Let $\{d_i\}$ be the character degrees of G. The group algebra is a direct sum of $d_i \times d_i$ matrix products. It follows that

$$r(G) < K\sum_i d_i^{\beta}$$

with $\beta > \omega$ and K a constant depending on β. Then we have

$$r(G) \leq K|G|^{\beta/2}$$

by using the fact that $|G| = \sum_i d_i^2$. Thus we have that $\alpha \leq \omega/2$.
Conversely, Schönhage's theorem states that

$$\sum_i d_i^{\omega} \leq r(G).$$

using Hölder's inequality, we obtain

$$|G| \leq (\sum_i d_i^{\omega})^{2/\omega} c(G)^{\omega/(\omega-2)}$$

Where $c(G)$ is the class number of G. Raising both sides to the power of $\omega/2$ we obtain

$$|G|^{\omega/2} \leq r(G)c(G)^{\omega/2-1}.$$

Then $\omega/2 \leq \alpha$ if we choose G such that $\log(G) >> \log(c(G))$ (e.g. Sym_n). □

Thus, ω can be shown to be equal to 2 if α can be shown equal to 1 (and vice versa).

Chapter 6

Conclusions and Further Work

We finish with a summary of the conclusions of this thesis, and identify areas of possible future work.

6.1 Conclusions

It was demonstrated in chapter 2 that we were able to compute the “value” as described in [12] of a trilinear form in a different fashion. We *symmetrized:* that is if we have a trilinear form $X^{[a]}Y^{[b]}Z^{[c]}$, then we could tensor multiply it together with $X^{[c]}Y^{[a]}Z^{[b]}$ and $X^{[b]}Y^{[c]}Z^{[a]}$, thus symmetrizing the numbers of available X, Y and Z-blocks. Then, there is no need to regard anything as a $\mathcal{C}$-tensor, and we may regard this new, symmetrised, trilinear form as a trilinear form in its own right. Taking its value yielded the same value as was obtained in [12].
Using this method, we were able to take higher powers of Coppersmith and Winograd’s algorithm, and find values for the associated trilinear forms. It was found that raising the algorithm to the third power did not yield an algorithm capable of reducing ω, but that raising it to the fourth power did.
However, the improvement gained in raising it to the fourth power was not as stark as the improvement gained from raising it from the first to the second power. It is therefore probable that the upper bound of ω generated by this algorithm approaches a limit larger than 2 as we increase the algorithm by higher powers. It is also probable that the bound for ω may be reduced only when we raise it to a power 2^M for some M, due to other powers not making full use of gains obtained in previous algorithms.
We also showed that the exponent of matrix multiplication ω and the rank α of multiplication in $\mathbb{C}(G)$ are linked in that $\omega = 2\alpha$. Therefore, investigation of multiplication in the group algebra, could yield results for ω too (and vice-versa).

6.2 Possible Further Work

As conjectured above, it may be possible to raise Coppersmith and Winograd’s algorithm to the eighth (or higher) power to obtain a reduction in ω. However, it is likely that any gains obtained in doing this will be very small.
Chapter 5 provided us with two conjectures: we had that if we could show that the strong USP capacity was the same as the capacity for regular USPs, then $\omega = 2$. We also showed that if we could find a group H with n subsets A_i, B_i satisfying the double product property and $|H| = n^{2+o(1)}$ and $|A_i||B_i| \geq n^{2-o(1)}$, then $\omega = 2$.

It was stated that Salem-Spencer was not adequate for proving the former conjecture-however there remains scope for it to be modified in such a way as to make it possible, or for there to be another possible method of showing it. As stated, the problem is equivalent to finding the number of ways of sorting objects $3N$ into 3 sets of size N, with the criteria:

- No two partitions may share an identical first, second or third set;.
- We should not be able to choose a first set from one partition, a second set from another and a third set from another still to make a new partition; and
- no objects may appear in all three sets throughout the partition.

The other conjecture may be investigated by starting with concrete examples of the Double Product Property. The paper [10] provides only a trivial example of a set that satisfies it.

Finally, it was also mentioned in Chapter 5 that rank α of multiplication in the group algebra $\mathbb{C}G$ is half of ω. Therefore, it is also desirable to know if $\alpha = 1$. Investigation of this group algebra multiplication could yield improvements for ω.

Appendix A

Pan's Trilinear Aggregation Algorithm

The following is the trilinear aggregration algorithm as described by Pan in [18],[19]. First we define:

$$\begin{aligned} n &= 2s \\ S^1(s) &= S_1^1(s) \cup S_2^1(s) \\ S_1^1(s) &= \{(i,j,k), 0 \le i \le j < k \le s-1\} \\ S_2^1(s) &= \{(i,j,k), 0 \le k < j \le i \le s-1\} \\ \bar{i} &= i + s \\ \bar{j} &= j + s \\ \bar{k} &= k + s. \end{aligned}$$

We also have Kronecker's δ:

$$\delta_{pq} = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$$

Finally, we have the symbol

$$\sum_{k=0}^{s-1}{}^{*}$$

which is the same as

$$\sum_{k=0}^{s-1}$$

when $i \neq j$ and

$$\sum_{k=0,k\neq i}^{s-1}$$

when $i = j$.

We collect all the aggregates together for the first part of the algorithm.

$$\begin{aligned}
T^0 &= \sum_{(i,j,k)\in S^1(s)} \big[(a_{ij}+a_{jk}+a_{ki})(b_{jk}+b_{ki}+b_{ij})(c_{ki}+c_{ij}+c_{jk})\\
&- (a_{ij}-a_{\bar{j}k}+a_{k\bar{i}})(b_{j\bar{k}}+b_{ki}-b_{\bar{i}j})(-c_{\bar{k}i}+c_{i\bar{j}}+c_{jk})\\
&- (-a_{\bar{i}j}+a_{j\bar{k}}+a_{ki})(b_{jk}-b_{\bar{k}i}+b_{i\bar{j}})(c_{k\bar{i}}+c_{ij}-c_{j\bar{k}})\\
&- (a_{i\bar{j}}+a_{jk}-a_{\bar{k}i})(-b_{\bar{j}k}+b_{k\bar{i}}+b_{ij})(c_{ki}-c_{\bar{i}j}+c_{j\bar{k}})\\
&- (a_{\bar{i}j}+a_{\bar{j}\bar{k}}-a_{k\bar{i}})(-b_{j\bar{k}}+b_{\bar{k}i}+b_{\bar{i}\bar{j}})(c_{\bar{k}\bar{i}}-c_{i\bar{j}}+c_{\bar{j}k})\\
&- (-a_{i\bar{j}}+a_{\bar{j}k}+a_{\bar{k}\bar{i}})(b_{\bar{j}\bar{k}}-b_{k\bar{i}}+b_{\bar{i}j})(c_{\bar{k}i}+c_{\bar{i}\bar{j}}-c_{j\bar{k}})\\
&- (a_{\bar{i}\bar{j}}-a_{j\bar{k}}+a_{\bar{k}i})(b_{\bar{j}k}+b_{\bar{k}\bar{i}}-b_{i\bar{j}})(-c_{k\bar{i}}+c_{\bar{i}j}+c_{\bar{j}\bar{k}})\\
&+ (a_{\bar{i}\bar{j}}+a_{\bar{j}\bar{k}}+a_{\bar{k}\bar{i}})(b_{\bar{j}\bar{k}}+b_{\bar{k}\bar{i}}+b_{\bar{i}\bar{j}})(c_{\bar{k}\bar{i}}+c_{\bar{i}\bar{j}}+c_{\bar{j}\bar{k}})\big].
\end{aligned}$$

In the second part, we gather together those undesirable terms which share a and b coordinates.

$$\begin{aligned}
T^1 &= \sum_{i,j=0}^{s-1} \Big\{a_{ij}b_{ij}\big[(s-2\delta_{ij})c_{ij}+\sum_{k=0}^{s-1} {*}(c_{ki}+c_{jk})\big]\\
&+ a_{ij}b_{\bar{i}j}\big[(s-\delta_{ij})c_{i\bar{j}}+\sum_{k=0}^{s-1} {*}(-c_{\bar{k}i}+c_{jk})\big]\\
&+ a_{\bar{i}j}b_{i\bar{j}}\big[(s-\delta_{ij})c_{ij}-\delta_{ji}c_{\bar{j}\bar{i}}+\sum_{k=0}^{s-1} {*}(c_{k\bar{i}}-c_{\bar{j}k})\big]\\
&+ a_{i\bar{j}}b_{ij}\big[(s-\delta_{ij})c_{\bar{i}j}-\sum_{k=0}^{s-1} {*}(c_{ki}+c_{j\bar{k}})\big]\\
&+ a_{\bar{i}j}b_{\bar{i}\bar{j}}\big[(s-\delta_{ij})c_{i\bar{j}}-\sum_{k=0}^{s-1} {*}(c_{\bar{k}\bar{i}}+c_{\bar{j}k})\big]\\
&+ a_{i\bar{j}}b_{\bar{i}j}\big[(s-\delta_{ij})c_{\bar{i}\bar{j}}-\delta_{\bar{j}\bar{i}}c_{ji}+\sum_{k=0}^{s-1} {*}(c_{\bar{k}i}-c_{j\bar{k}})\big]\\
&+ a_{\bar{i}\bar{j}}b_{i\bar{j}}\big[(s-\delta_{ij})c_{\bar{i}j}+\sum_{k=0}^{s-1} {*}(-c_{k\bar{i}}+c_{\bar{j}\bar{k}})\big]\\
&+ a_{\bar{i}\bar{j}}b_{\bar{i}\bar{j}}\big[(s-2\delta_{ij})c_{\bar{i}\bar{j}}+\sum_{k=0}^{s-1} {*}(c_{\bar{k}\bar{i}}+c_{\bar{j}\bar{k}})\big]\Big\}.
\end{aligned}$$

In the third part, we gather together those undesirable terms which share a and c coordinates (which we have not yet already gathered).

$$
\begin{aligned}
T^2 &= \sum_{i,j=0}^{s-1} \{ a_{ij} \sum_{k=0}^{s-1} *(b_{ki} + b_{jk}) c_{ij} - a_{ij} \sum_{k=0}^{s-1} *(b_{ki} + b_{j\bar{k}}) c_{i\bar{j}} \\
&+ a_{\bar{i}j} \sum_{k=0}^{s-1} *(b_{jk} - b_{\bar{k}i}) c_{ij} - a_{i\bar{j}} [\sum_{k=0}^{s-1} *(b_{k\bar{i}} - b_{\bar{j}k}) - \delta_{ji} b_{\overline{ji}}] c_{\overline{ij}} \\
&+ a_{\bar{i}j} [\sum_{k=0}^{s-1} *(b_{\overline{ki}} - b_{j\bar{k})} - \delta_{\overline{ji}} b_{ji}] c_{ij} + a_{i\bar{j}} \sum_{k=0}^{s-1} *(b_{\overline{jk}} - b_{k\bar{i}}) c_{\overline{ij}} \\
&- a_{\overline{ij}} \sum_{k=0}^{s-1} *(b_{\overline{ki}} + b_{\bar{j}k}) c_{\bar{i}j} - a_{\overline{ij}} \sum_{k=0}^{s-1} *(b_{\overline{ki}} + b_{\overline{jk}}) c_{\overline{ij}} \}.
\end{aligned}
$$

Finally, we gather together those ungathered undesirable terms which share b and c coordinates.

$$
\begin{aligned}
T^3 &= \sum_{i,j=0}^{s-1} \{ \sum_{k=0}^{s-1} *(a_{ki} + a_{jk}) b_{ij} c_{ij} + [\sum_{k=0}^{s-1} *(a_{k\bar{i}} - a_{\bar{j}k}) - \delta_{ji} a_{\overline{ji}}] b_{\bar{i}j} c_{i\bar{j}} \\
&- \sum_{k=0}^{s-1} *(a_{ki} + a_{j\bar{k}}) b_{i\bar{j}} c_{ij} + \sum_{k=0}^{s-1} *(a_{jk} - a_{\bar{k}i}) b_{ij} c_{\bar{i}j} \\
&+ \sum_{k=0}^{s-1} *(a_{\overline{jk}} - a_{k\bar{i}}) b_{\overline{ij}} c_{i\bar{j}} - \sum_{k=0}^{s-1} *(a_{\overline{ki}} + a_{\bar{j}k}) b_{\bar{i}j} c_{\overline{ij}} \\
&+ [\sum_{k=0}^{s-1} *(a_{\bar{k}i} - a_{j\bar{k}}) - \delta_{\overline{ji}} a_{ji}] b_{i\bar{j}} c_{\bar{i}j} + \sum_{k=0}^{s-1} *(a_{\overline{ki}} + a_{\overline{jk}}) b_{\overline{ij}} c_{\overline{ij}} \}.
\end{aligned}
$$

Finally, we have

$$
\sum_{i,j,k=0}^{2s} = T^0 - T^1 - T^2 - T^3
$$

which we achieve in $8(s^3 - s)/3 + 24s^2$ multiplications, which, on setting $n = 2s$ is $(n^3 - 4n)/3 + 6 * n^2$ operations.

Appendix B

Optimisation Methods in Chapters 3 and 4

B.1 Optimisation for Chapter 3

In order to determine the optimal values of $\hat{a}, .., \hat{g}, \hat{c}', .., \hat{g}'$, we use a Newton Raphson procedure. We start with the function

$$\begin{aligned} & N^{-p}\binom{N}{A_0, A_1, A_2, A_3, A_4, A_5, A_6}\frac{M'}{M_{\mathsf{max}}} \times \\ \times \quad & (3q)^{6\tau\hat{b}}(3q^2+3)^{6\tau\hat{c}'}(q^3+6q)^{3\tau\hat{d}'} \times \\ \times \quad & 3q^{\tau}(1+q^{3\tau})^{\hat{e}'}(3^{2/3}q^{\tau}(1+q^{3\tau})^{1/3}(6+q^{3\tau})^{1/3})^{6\hat{f}'}3(1+q^{3\tau})^{\hat{g}'}. \end{aligned}$$

Approximating the function using Stirling's formula, taking logs and allowing $N \to \infty$, we get our function F.

We use Coppersmith and Winograd's value of τ here, and use various values of q: if the overall value of the function at our new point exceeds $(q+2)^3$, then we may reduce τ in order to make the equality exact (further, this shows that we are tending towards a maximum).

We start with initial values for $\hat{b}, \hat{d}, \hat{e}, \hat{f}, \hat{g}$. From calculations in chapter 3, we know that M_{max} occurs when $\hat{c} = \frac{\hat{d}\hat{e}\hat{g}}{\hat{f}^2}$, so we set $\hat{c}$ as this. This also forces $\hat{a}$ (due to the restriction that $3\hat{a} + ... + \hat{g} = N$. Hence we also have our values of $A_0, .., A_6$. We use these same $A_0, ..., A_6$ in M'. We note that given $A_0, .., A_6$, there is one degree of freedom among the $\hat{c}', .., \hat{g}'$ so we allow $\hat{c}'$ to vary. This will force $\hat{d}', \hat{e}', \hat{f}', \hat{g}'$.

Thus, this is a function of $\hat{b}, \hat{c}, \hat{d}, \hat{e}, \hat{f}, \hat{g}, \hat{c}'$. We use a Newton-Raphson procedure in several dimensions. Given a function F over n variables $x_1, .., x_n$, and an initial vector $a = a_1, .., a_n$, the procedure is defined as follows:

We start by setting a column vector v where the entries are

$$v_i = \frac{\partial F}{\partial x_i}$$

evaluated at $x_i = a_i$.

We then set a matrix G where

$$G_{i,j} = \frac{\partial^2 F}{\partial x_i \partial x_j}$$

evaluated at $x_i = a_i$.
Then, we set our new "initial" vector as being

$$a - G^{-1}v.$$

The initial values were chosen by considering the proportions of the blocks $X^{[0]}Y^{[1]}Z^{[1]}$, $X^{[1]}Y^{[0]}Z^{[1]}$,$X^{[1]}Y^{[1]}Z^{[0]}$, $X^{[0]}Y^{[0]}Z^{[2]}$,$X^{[0]}Y^{[2]}Z^{[0]}$, and $X^{[2]}Y^{[0]}Z^{[0]}$. There, after raising the original algorithm to the Nth power, each final block contained $0.317N$ of each of the first three and $0.016N$ of each of the last three. Therefore, our initial value for $\hat{b}$ (which represented $X^{[0]}Y^{[1]}Z^{[5]}$ was

$$3 \times (0.317 \times 0.016 \times 0.016) = 0.000243456.$$

We find all our other initial values in a similar manner. It is reasonable to assume that the values that give the maximum will be obtain in this region, since previous maxima were also obtained around these points.
Finally, we use Maple to perform 100+ iterations of the Newton-Raphson method to get a very close approximation to the maximum.

B.2 Optimisation for Chapter 4

The methods for finding the maximum values in Chapter four were similar to those in Chapter 3, but with some important differences.
We start by noting that we have ten variables $\hat{a}, \hat{b}, \hat{c}, \hat{d}, \hat{e}, \hat{f}, \hat{g}, \hat{h}, \hat{\imath}, \hat{\jmath}$. As shown in chapter 4, we know that M_{max} occurs when

$$\hat{c} = \frac{\hat{e}\hat{f}\hat{\imath}}{\hat{h}^2} \hat{d} = \frac{\hat{e}\hat{g}\hat{\imath}}{\hat{h}\hat{\jmath}}$$

if $A_0, .., A_8$ are fixed.
Therefore, we set values for $\hat{b}, \hat{e}, \hat{f}, \hat{g}, \hat{h}, \hat{\imath}, \hat{\jmath}$ and set $\hat{c}$,$\hat{d}$ to be defined as above (thus forcing $\hat{a}$).
As a result, we will have our $A_0, .., A_8$. Given these, we have two degrees of freedom, so we allow $\hat{c}'$ and $\hat{d}'$ (in M') to vary, and define our $\hat{e}', \hat{f}', \hat{g}', \hat{h}', \hat{\imath}', \hat{\jmath}'$ appropriately (in terms of $A_0, .., A_8$ and $\hat{c}'$ and $\hat{d}'$).
Here, our $F =$

$$|\Delta| \frac{M'}{N^2 M_{max}} [V_{017}]^{6\hat{b}} [V_{026}]^{6\hat{c}'} [V_{035}]^{6\hat{d}'} [V_{044}]^{3\hat{e}'} [V_{116}]^{3\hat{f}'} [V_{125}]^{6\hat{g}'} [V_{134}]^{6\hat{h}'} [V_{233}]^{3\hat{\imath}'} [V_{224}]^{6\hat{\jmath}'}$$

with Δ defined as in chapter 4 (it is a function of $\hat{b}, \hat{e}, .., \hat{\jmath}$). Again, we approximate the factorials using Stirling, allowing $N \to \infty$ and take logs to obtain a function F.
We perform the Newton-Raphson interation as before, where F is a function over $\hat{b}, \hat{e}, \hat{f}, \hat{g}, \hat{h}, \hat{\imath}, \hat{\jmath}, \hat{c}', \hat{d}'$, trying several values of q, and using Coppersmith and Winograd's τ.
The initial values are found similarly to above, except here we may use the values of $\hat{a}, .., \hat{d}$ in Coppersmith and Winograd, as we may regard the algorithm as the product of two squares.
For example, we may take our initial value for $\hat{b}$ as being

$$2 \times 0.000233 \times 0.012506 = 0.00005827796.$$

We find our other initial values similarly and use 100 iterations of the Newton-Raphson procedure as before to obtain our desired values.

Bibliography

[1] F.A. Behrend *On Sets of Integers which Contain no Three Terms in Arithmetical Progression* Proc. Nat. Acad. Sci. USA Volume 32 (1946), pp 331-332.

[2] D. Bini *Relations Between Exact and Approximate Bilinear Algorithms. Applications.* Calcolo 17 (1980) pp 87-97

[3] D. Bini, M. Capovani, G. Lotti, F. Romani. $O(n^{2.7799})$ *Complexity for Matrix Multiplication.* Inf. Proc. letters 8 (1979) pp 234-235

[4] M. Bläser. *A* $\frac{5}{2}n^2$ *Lower Bound for the Rank of* $n \times n$ *Matrix Multiplication over Arbitrary Fields.* Proceedings of the 40th Annual Symposium on Foundations of Computer Science (1999) page 45.

[5] M. Bläser. *Beyond the Alder Strassen Bound.* Theoretical Computer Science, Volume 331, Issue 1, Automata, languages and programming , (2005) pp 3-21.

[6] M. Bläser.*On the complexity of the multiplication of matrices of small formats* Journal of Complexity Volume 19 , Issue 1 (2003)pp 43 - 60

[7] R.W. Brockett and D. Dobkin. *On the optimal evaluation of a set of bilinear forms.*Proceedings of the fifth annual ACM symposium on Theory of computing (1973) pp. 88-95

[8] P. Bürgisser, M. Clausen, M.A. Shokrollahi *Algebraic Complexity Theory*, Volume 315, Grundlehren der Mathematischen Wissenschaften, Springer-Verlag 1997

[9] H. Cohn and C. Umans. *A Group Theoretic Approach to Fast Matrix Multiplication.* Proceedings of the 44th Annual Symposium on Foundations of Computer Science, 11-14th October 2003, Cambridge, MA, IEEE Computer Society, pp.438-449, arXiv:math.GR/0307321.

[10] H. Cohn, R. Kleinberg, B. Szegedy and C. Umans. *Group-theoretic Algorithms for Matrix Multiplication.* Proceedings of the 46th Annual Symposium on Foundations of Computer Science, 9-12 July 2006, Genova, Italy, IEEE Computer Society, pp.379-388, arXiv: math.GR/0511460v1.

[11] D. Coppersmith and S. Winograd. *On the Asymptotic Complexity of Matrix Multiplication.* SIAM Journal of Computation, Volume 11, No. 3 (1982), pp 472-492

[12] D. Coppersmith and S. Winograd. *Matrix Multiplication via Arithmetic Progressions.* Journal of Symbolic Computation, Volume 9:pp 251-280, 1990.

[13] H. 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 (1978) pp 127-148.http://www.facebook.com/?ref=home

[14] J.E. Hopcroft and L.R. Kerr *On Minimizing the Number of Multiplications Necessary for Matrix Multiplication* SIAM Journal of Applied Mathematics, Volume 20 (1971) pp 30-36.

[15] J. Laderman *A Noncommutative Algorithm for multiplying* 3×3 *matrices using 23 multiplications.* Bull. Amer. Math. Soc. Volume 82 (1976) pp 180-182.

[16] J. C. Lafon and S. Winograd *A Lower Bound for the Multiplicative Complexity of the Product of Two Matrices.* Centre de Calcul de L'Esplanade, U.E.R. de Mathematique, Univ. Louis Pasteur, Strasbourg (1978)

[17] G. James and M. Liebeck *Representations and Characters of Groups.* Cambridge University Press, Cambridge, Second Edition, 2001.

[18] V. Pan *Strassen's Algorithm is not Optimal, Trilinear Technique of Aggregating, Uniting and Canceling for Constructing Fast Algorithms for Matrix Operations.* Proceedings of the 19th Annual Symposium on Foundations of Computer Science (1978), pp 166-176.

[19] V. Pan *Field Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplications*, Proc. 20th IEEE Symposium on Foundations of Computer Science, 1979.

[20] V. Pan *New Fast Algorithms for Matrix Operations*, SIAM Journal of Computation. Volume 9, No.2 (1980), pp 321-342.

[21] V. Pan *How Can We Speed Up Matrix Multiplication?*, SIAM Review Volume 26, No. 3. (1984) pp 393-415

[22] R. Salem, and D.C. Spencer *On Sets of Integers which Contain no Three Terms in Arithmetical Progression* Proc. Nat. Acad. Sci. USA Volume 28 (1942), pp 561-563.

[23] A. Schönhage *Partial and Total Matrix Multiplication*, SIAM Journal of Computation. Volume 10 (1981), pp 434-455.

[24] V. Strassen *Gaussian Elimination is not Optimal*, Numer. Math., 13 (1969), pp. 354-356

[25] V. Strassen *The Asymptotic Spectrum of Tensors and the Exponent of Matrix Multiplication* Proceedings of the annual IEEE Symposium on Foundations of Computer Science (1986), pp49-54.

[26] V. Strassen *Relative Bilinear Complexity and Matrix Multiplication* J. Reine Angew. Mathe. 375-376 (1987) pp. 406-443.

[27] A. Waksman *On Winograd's Algorithm for Inner Products*, IEEE Transactions on Computers. Volume 19, Issue 4 (1970), pp. 360-361.

[28] S. Winograd. *A New Algorithm for Inner Product* IEEE Transactions on Computers, Volume 17, Issue 7 (1968) pp 693-694.

[29] S. Winograd: *On Multiplication of* 2×2 *matrices.*Lin. Alg. Appl. 4 (1971) pp 381-388