# Constructing New Turyn Type Sequences, T-Sequences and Hadamard Matrices

BY

STEPHEN LONDON
B.S., North Carolina State University, 1995
M.S., University of Illinois at Chicago, 2006

THESIS

Submitted in partial fulfillment of the requirements
For the degree of Doctor of Philosophy in Mathematics
in the graduate college of
University of Illinois at Chicago 2013

Chicago, Illinois

Defense Committee:

Vera Pless, Chair and Advisor
Gyorgy Turan
Samad Hedayat
Robert Sloan, Computer Science
William Huffman, Loyola University Chicago

# TABLE OF CONTENTS

LIST OF TABLES

# SUMMARY

This study investigates computational methods for the construction and enumeration of Turyn Type sequences, which can be used to construct T-Sequences and Hadamard matrices. Extending and optimizing an approach originally introduced by Hadi Kharaghani, we have performed the most comprehensive search for Turyn Type sequences to date. We have checked the enumeration of Turyn Type sequences recently reported by Best et al [3] and found it to be correct for lengths up to 30, but identified 66 additional sequences that were missed for length 32. We then randomly selected and evaluated approximately $\frac{1}{32}$ of the search space to estimate the total number of Turyn Type sequences of lengths 34 and 36. In order to identify longer sequences, the search algorithm was further optimized by restricting the values of some parameters. This approach allowed the identification of 119 new Turyn Type sequences of lengths 34 and 101 new Turyn Type Sequences of length 36. We verified that each sequence leads to a unique Hadamard matrix. This calculation has allowed the identification of 101 new Hadamard matrices of order 428; only one had been identified prior to this analysis. We have also found 10 Turyn Type sequences of length 38. This directly leads to 10 of the 11 known T-sequences of length 113, with the other being discovered by Best et al [3]. Finally, we discovered the first three Turyn Type sequences of length 40 which are the longest known Turyn Type sequences.

# CHAPTER 1

# INTRODUCTION

## 1.1 Basic Definitions

A Hadamard matrix $H$ is an $n \times n$ matrix of 1s and -1s such that $HH^T = nI$ or equivalently an $n \times n$ matrix of 1s and -1s with every two rows orthogonal and every two columns orthogonal. All Hadamard matrices except the trivial ones with orders 1 and 2 must have an order which is a multiple of 4. The French mathematician Jacques Hadamard conjectured in 1893 that a Hadamard matrix exists for every order which is a multiple of 4. The Hadamard Conjecture has been proven for all orders less than 668, but remains an open question to this day [6]. Two Hadamard matrices are considered equivalent if one can be obtained from the other by a combination of permutations and negations of its rows and columns. We call these Hadamard operations. The number of distinct inequivalent Hadamard matrices is only known for orders 32 and below [19].

## 1.2 Progress on the Hadamard Conjecture

James Joseph Sylvester [27] first discovered these matrices in 1867. Sylvester discovered a simple easy way to construct a Hadamard matrix for every order which is a power of 2. French mathematician Jacques Hadamard [11] discovered matrices of order 12 and 20 in 1893.

Raymond Paley [25] made the first major discovery in 1933. He constructed Hadamard matrices for all orders of the form $q+1$ where $q \equiv 3 (mod\ 4)$ is a prime power and of the form $2(q+1)$ where $q \equiv 1 (mod\ 4)$ is a prime power. With this construction, a Hadamard of every multiple of 4 order up to 100 except 92 can be constructed [29]. However, since primes get less common this construction works for smaller and smaller percentages of numbers as the orders get larger. John Williamson [31] made the next major breakthrough in 1944. He found a way to construct Hadamards by putting together a $4 \times 4$ array of circulant matrices. He discovered a number of new matrices using this method. Leonard Baumert, Solomon Golomb and Marshall Hall finally discovered the first known order 92 Hadamard matrix in 1961 [1]. Numerous other orders have since been discovered with this method, often with the aid of computers. Baumert and Hall generalized this method in 1965 to $4t \times 4t$ arrays [2]. Goethals and Seidel discovered another construction around the late 1960s [10]. The Goethals-Seidel method in conjunction with computer technology has been used in many of the more recent discoveries, including those discovered here.

A Hadamard matrix of every order which is a multiple of 4 less than 428 except for 268 was known at the beginning of 1985. That year, Kazue Sawade [26] found, using the Goethals-Seidel method and a computer search, the first Hadamard of order 268. Masahiko Miyamoto [24] proved in 1991 that if you have a Hadamard of order $q - 1$ where $q$ is the power of an odd prime, then there exists a Hadamard of order $4q$. It was not until 2004 when a computer search by Hadi Kharaghani and Behruz Tayfeh-Rezaie yielded the first known Hadamard of order 428 [17]. This discovery was made by finding the first Turyn Type sequences of length 36. With the above methods and some variations on them, this left 668, 716, 764 and 892 as the only orders under 1000 for which there was no known Hadamard matrix [6]. In 2007, Dragomir Djokovic discovered the first known Hadamard of order 764 using a method called symmetric difference sets [6]. A complete list of how to construct a Hadamard matrix for any known order up to 1000 is in the appendix.

## 1.3 Progress on Hadamard Enumeration

A much more difficult problem is to count the number of inequivalent Hadamard matrices up to equivalence for each order. There are $(2^n n!)^2$ ways to permute and negate the rows and columns of a Hadamard matrix. This number is huge for $n \geq 16$ and consequently being able to tell when two Hardmard matrices are equivalent can be very difficult.

It has long been known that the Hadamard matrices of orders 1, 2, 4, 8 and 12 are unique. Marshall Hall discovered that there were exactly 5 of order 16 in 1961 [12] and there are exactly 3 unique Hadamards of order 20 in 1965 [13]. Ito and Leon [16] discovered that there were exactly 60 of order 24 in the late 1980s and Kimura [20] counted the 487 Hadamards of order 28 in 1994. Kharaghani and Tayfeh-Rezaie [18,19] have found that there are exactly 13,710,027 Hadamard matrices of order 32 in 2012.

TABLE I

NUMBER OF INEQUIVALENT HADAMARDS FOR EACH ORDER

| Order | Number |
|---:|---:|
| 1 | 1 |
| 2 | 1 |
| 4 | 1 |
| 8 | 1 |
| 12 | 1 |
| 16 | 5 |
| 20 | 3 |
| 24 | 60 |
| 28 | 487 |
| 32 | 13710027 |

## 1.4 Turyn Type Sequences

The main topic of this paper is Turyn Type sequences, which are sets of four sequences of lengths $n, n, n, n-1$, where $n$ is even and often denoted by $TT(n)$. They lead directly to T-sequences of order $3n-1$ and to Hadamard matrices of order $12n-4$. These sequences were first discovered by Richard Turyn in 1974 [28] and he discovered them for $n = 2, 4, 6, 8$. In 1994, examples were found for all orders up to $n = 24$ [22] and further extended up to $n = 34$ in 2001 [21]. Several years later, Kharaghani and Tayfeh-Reazie found the first example for $n = 36$ which directly led to the first known Hadamard matrix of order 428. In 2012 Best et. al [3] found an example for $n = 38$. We have independently found 10 additional examples for $n = 38$ and also the first three examples for $n = 40$. The details of these results will be presented in chapters 3-5.

If a $TT(n)$ can be found for $n = 56$ and $n = 60$, it would lead to the first Hadamard matrices of orders 668 and 716, respectively.

# CHAPTER 2

# HADAMARD MATRIX CONSTRUCTION METHODS

## 2.1 Sylvester and Kronecker Product Constructions

Sylvester showed in 1867 that if $H$ is a Hadamard matrix of order $n$ then the matrix below is a Hadamard matrix of order $2n$ [27].

$$\begin{bmatrix} H & H \\ H & -H \end{bmatrix}$$

This can be repeated to construct a Hadamard matrix of order $2^n$ for any positive integer $n$.

If a Hadamard matrix $A$ of order $m$ and a Hadamard matrix $B$ of order $n$ are Hadamard matrices, then their Kronecker or tensor product gives a Hadamard matrix of order $mn$.

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1m}B \\ a_{21}B & a_{22}B & \dots & a_{2m}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \dots & a_{mm}B \end{bmatrix}$$

## 2.2 Paley Construction

There are two infinite classes of Hadamard matrices of Paley type [29]. A type I Paley can be constructed for any order $q + 1$ where $q \equiv 3 (mod\ 4)$ is a prime power and a type II Paley can be constructed for any order $2(q + 1)$ where $q \equiv 1 (mod\ 4)$ is a prime power.

We define a conference matrix $C$ to be an $n \times n$ matrix of 1s and -1s except 0s along the main diagonal such that $CC^T = (n - 1)I_n$.

We can construct such an antisymmetric conference matrix $C$ of any order $n$ of the form $q + 1$ where $q \equiv 3 (mod\ 4)$ is a prime power as follows. We create a $q \times q$ matrix with rows and columns numbered from 0 to $q-1$. We put 0s along the main diagonal, 1s in the $(i,j)$ entries with $i \neq j$ where $i-j$ is a perfect square in the field $F_q$ and -1s everywhere else. We then append a row of 1s on the top, a row of -1s on the left and a 0 in the upper left corner. This gives a conference matrix which is antisymmetric. A Hadamard matrix of order $n$ is

$$H = I + C$$

We can construct such a symmetric conference matrix $C$ of any order $n$ of the form $2(q+1)$ where $q \equiv 1(mod\ 4)$ is a prime power as follows. We create a $q \times q$ matrix with rows and columns numbered from 0 to $q-1$. We put 0s along the main diagonal, 1s in the $(i,j)$ entries with $i \neq j$ where $i-j$ is a perfect square in the field $F_q$ and -1s everywhere else. We then append a row of 1s on the top, a row of 1s on the left and a 0 in the upper left corner. This gives a conference matrix which is symmetric. $A$ Hadamard matrix of order $n$ is

$$H = \begin{bmatrix} I+C & -I+C \\ -I+C & -I-C \end{bmatrix}$$

2.3 <u>Williamson Construction</u>

If we have four symmetric commutative matrices $A, B, C, D$ with entries 1 and -1 of order n with n odd with the below condition, then we can form a Hadamard matrix $H$ of order $4n$ [29].

$$A^2 + B^2 + C^2 + D^2 = 4nI_n$$

$$H = \begin{bmatrix} A & B & C & D \\ -B & A & -D & C \\ -C & D & A & -B \\ -D & -C & B & A \end{bmatrix}$$

If a Williamson matrix existed for all odd $n$, it would prove the Hadamard conjecture. However, it was shown that there exist no Williamson matrices for orders 35, 47, 53 and 59 [14].

Baumert and Hall [2] discovered a way to generalize the Williamson construction by replacing the $4 \times 4$ matrix used in the Williamson construction with a $4t \times 4t$ matrix called a Baumert Hall array. The below condition must be met.

$$t(A^2 + B^2 + C^2 + D^2) = 4nI_n$$

An example of a $12x12$ Baumert Hall array is given below [15].

$$H = \begin{bmatrix} A & A & A & B & -B & C & -C & -D & B & C & -D & -D \\ A & -A & B & -A & -B & -D & D & -C & -B & -D & -C & -C \\ A & -B & -A & A & -D & D & -B & B & -C & -D & C & -C \\ B & A & -A & -A & D & D & D & C & C & -B & -B & -C \\ B & -D & D & D & A & A & A & C & -C & B & -C & B \\ B & C & -D & D & A & -A & C & -A & -D & C & B & -B \\ D & -C & B & -B & A & -C & -A & A & B & C & D & -D \\ -C & -D & -C & -D & C & A & -A & -A & -D & B & -B & -B \\ D & C & -B & -B & -B & C & C & -D & A & A & A & D \\ -D & B & C & C & C & B & B & -D & A & -A & D & -A \\ C & -B & -C & C & D & -B & -D & -B & A & -D & -A & A \\ -C & -D & -D & C & -C & -B & B & B & D & A & -A & -A \end{bmatrix}$$

2.4 <u>Goethals-Seidel Construction</u>

Goethals and Seidel discovered a construction similar to, but more sophisticated than the Williamson in 1967 [10].

A set of four $q \times q$ circulant $0, \pm 1$ matrices $T_1, T_2, T_3, T_4$ with $T_1 T_1^T + T_2 T_2^T + T_3 T_3^T + T_4 T_4^T = qI_q$ are called T-matrices. Let $W_1, W_2, W_3, W_4$ be a set of four $r \times r$ Williamson matrices and $S$ be a back circulant $q \times q$ matrix with 0s except 1s along the back diagonal. We build a Hadamard matrix of order $4qr$.

$$R = S \otimes I_r$$

$$A_1 = T_1 \otimes W_1 + T_2 \otimes W_2 + T_3 \otimes W_3 + T_4 \otimes W_4$$

$$A_2 = -T_1 \otimes W_2 + T_2 \otimes W_1 + T_3 \otimes W_4 - T_4 \otimes W_3$$

$$A_3 = -T_1 \otimes W_3 - T_2 \otimes W_4 + T_3 \otimes W_1 + T_4 \otimes W_2$$

$$A_4 = -T_1 \otimes W_4 + T_2 \otimes W_3 - T_3 \otimes W_2 + T_4 \otimes W_1$$

$$H = \begin{bmatrix} A_1 & A_2R & A_3R & A_4R \\ -A_2R & A_1 & A_4^tR & -A_3^tR \\ -A_3R & -A_4^tR & A_1 & A_2^tR \\ -A_4R & A_3^tR & -A_2^tR & A_1 \end{bmatrix}$$

If a Goethals-Seidel construction existed for all odd $n$, it would prove the Hadamard conjecture. In Best et al. [3] and here, we have found the first T-Matrices of order 113. The smallest order for which none are known is 131 [4].

2.5 Symmetric Difference Sets Construction

Djokovic [6, 7, 8] has used a symmetric difference set construction to come up with many new orders of Hadamard matrices in recent years. We construct a matrix of order 4n, by first finding a cyclic group of order $n-1$. Call the cyclic group H. We find a subgroup of some prime order, p. We create the $c = \frac{n-1}{p}$ cosets of this group. We number the cosets $\propto_0, \propto_1, \ldots, \propto_{c-1}$.

We next define four index sets $J_1, J_2, J_3, J_4$. We create these sets where each contains some subset of the numbers between 0 and $c-1$. Each number in these index sets corresponds to a coset of our group. We have a search space of size $2^{4c}$ to find suitable index sets. We now form 4 sets $S_1, S_2, S_3, S_4$ which are the union of members from the index sets defined by $S_i = \bigcup_{i \epsilon j_k} \propto_i, k = 1,2,3,4.$

We next define the sets $a_1, a_2, a_3, a_4$ to be n element sets with all members 1 and -1. For $a_k$ let the ith element be 1 if $i \in S_k$ and -1 otherwise. Let each of these sets form the top row of an $nxn$ circulant matrix. We also denote these 4 matrices $A_1, A_2, A_3, A_4$. If these matrices satisfy $\sum_{k=1}^{4} A_k A_k^T = 4nI_n$ we can build a Hadamard matrix of order 4n. We do this by plugging the matrices $A_1, A_2, A_3, A_4$ into the Goethals-Seidel array of section 2.4. The result is a Hadamard matrix of order $4n$.

2.6 Miyamoto Construction

Let $q \equiv 1(mod\ 4)$ and suppose we have a Hadamard matrix $K$ of order $q-1$. Then there is a Hadamard matrix $H$ of order $4q$ [24]. Let $K$ be the Hadamard of order $q-1$ where $q$ the power of an odd prime. We break $K$ up into 4 equal parts as shown below.

$$K = \begin{bmatrix} K_1 & K_2 \\ -K_3 & K_4 \end{bmatrix}$$

We construct a conference matrix of order $q+1$ as in the Paley II construction. $D$ is symmetric in this construction. Note that $e$ represents a row of 1s and $e^T$ represents a column of 1s.

$$C = \begin{bmatrix} 0 & e \\ e^T & D \end{bmatrix}$$

We can permute rows and columns to get the below symmetric matrix. Since the matrix is symmetric, $C_1, C_4$ are also symmetric.

$$E = \begin{bmatrix} 0 & 1 & e & e \\ 1 & 0 & e & -e \\ e^T & e^T & -C_1 & C_2 \\ e^T & -e^T & C_2^T & C_4 \end{bmatrix}$$

We next create the following matrices, which we then break into 16 equal parts each.

$$U = [U_{ij}] = \begin{bmatrix} C_1 & C_2 & 0 & 0 \\ -C_2^T & C_4 & 0 & 0 \\ 0 & 0 & C_1 & C_2 \\ 0 & 0 & -C_2^T & C_4 \end{bmatrix}$$

$$V = [V_{ij}] = \begin{bmatrix} I & 0 & K_1 & K_2 \\ 0 & I & K_3 & -K_4 \\ -K_1^T & -K_3^T & I & 0 \\ -K_2^T & K_4^T & 0 & I \end{bmatrix}$$

$$T_{ij} = U_{ij} \otimes \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} + V_{ij} \otimes \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Finally, we construct a Hadamard matrix of order $4q$.

$$H = \begin{bmatrix} 1 & -e & 1 & e & 1 & e & 1 & e \\ -e^T & T_{11} & e^T & T_{12} & e^T & T_{13} & e^T & T_{14} \\ -1 & -e & 1 & -e & 1 & e & -1 & -e \\ -e^T & T_{21} & -e^T & T_{22} & e^T & T_{23} & -e^T & T_{24} \\ -1 & -e & -1 & -e & 1 & -e & 1 & e \\ -e^T & T_{31} & -e^T & T_{32} & -e^T & T_{33} & e^T & T_{34} \\ -1 & -e & 1 & e & -1 & -e & 1 & -e \\ -e^T & T_{41} & e^T & T_{42} & -e^T & T_{43} & -e^T & T_{44} \end{bmatrix}$$

## 2.7 Hadamard Profiles

One method that is commonly used to tell that two Hadamard matrices are inequivalent is a k-profile where k is a multiple of 4 [5]. The method for computing a 4-profile is below. Larger profiles can be computed analogously.

Let $\pi(t)$ be the number of 4-sets $\{i, j, k, l\}$ such that $P_{ijkl} = t$, where

$$P_{ijkl} = \left| \sum_{\alpha} h_{i\alpha} h_{j\alpha} h_{k\alpha} h_{l\alpha} \right|$$

We note $0 \leq t \leq n$. Hadamard operations preserve the values for $\pi(t)$. Thus two Hadamard matrices must be inequivalent if $\pi(t)$ is different for any values of $t$.

2.8 Symmetric Hamming Distance Distribution

The Symmetric Hamming distance distribution [30] is more efficient than the profile method for differentiating Hadamard matrices. We now summarize the algorithm.

We have an $nxn$ matrix. We begin with $k = 3$ and repeat until $k = \frac{n}{2}$ if necessary. If k gets to be too large, this algorithm becomes extremely inefficient, thus it is only good for telling us that two Hadamard matrices are different.

1) Let $H$ be a Hadamard matrix of order $n$.
2) Take each of the $\binom{n}{k}$ different k-column subsets of the sets of all columns to form $n \times k$ submatrices of $H$.
3) For each of the $\binom{n}{2}$ pairs of rows of the submatrix, we define the Hamming distance of the rows to be the number of places in which the rows differ.
4) Create the symmetric Hamming distance distribution polynomial based on these Hamming distances where $a_i$ is the frequency of a Hamming distance of $i$ among the $\binom{n}{2}$ pairs. We define the Symmetric Hamming polynomial as $SW_k(x) = \sum_{i=0}^{i=k} a_i \, x^{\min(i,k-i)}$.
5) We have a set of $\binom{n}{k}$ polynomials for the Hadamard matrix. This set of polynomials does not change when we do Hadamard operations to the matrix.

We take two Hadamard matrices and start with $k = 3$. If there is any difference in the set of Symmetric Hamming polynomials, they must be different. Otherwise, we're not sure and we increment $k$ and repeat. If we get to $k = \frac{n}{2}$ and the polynomials are the same, the Hadamards are equivalent. In practice, this can be an efficient way to tell when Hadamards are inequivalent.

# CHAPTER 3

# TURYN TYPE SEQUENCES

## 3.1 Definitions

We begin by presenting some useful definitions. Let $A = (a_0, a_1, \dots, a_{n-1})$ be a sequence of length $n$. We define the nonperiodic autocorrelation function $N_A$ as

$$N_A(s) = \sum_{i=0}^{n-1-s} a_i a_{i+s} \; for\; 0 \leq s \leq n-1, N_A(s) = 0 \; for\; s \geq n.$$

The Hall polynomial for sequence $A$ is defined as

$$h_A(t) = \sum_{i=0}^{n-1} a_i t^i$$

We define the nonnegative real value function $f_A$ as

$$f_A(\theta) = \left|h_A(e^{i\theta})\right|^2 = N_A(0) + 2\sum_{j=1}^{n-1} N_A(j)\cos(j\theta).$$

Four $(-1, 1)$ sequences $X, Y, Z, W$ of lengths $n, n, n, n-1$ are of Turyn Type if

$$(N_X + N_Y + 2N_Z + 2N_W)(s) = 0 \; for\; s \geq 1.$$

We denote these as $TT(n)$.

We define $x, y, z, w$ as the sum of the elements in $X, Y, Z, W$ respectively.

Define $X_r, Y_r, Z_r, W_r$ be the $X, Y, Z, W$ sequences written in reverse order.

For sequences $A, B$ define $A \geq B$ if for the smallest $i$ where $a_i \neq b_i, a_i > b_i$.

Four $(-1,1)$ sequences $A, B, C, D$ of lengths $n+p, n+p, n, n$ are called base sequences if

$$(N_A + N_B + N_C + N_D)(s) = 0 \; for \; s \geq 1.$$

We denote these $BS(n+p, n)$.

Four $(0,-1,1)$ sequences $T1, T2, T3, T4$ each of lengths $n$ are called T-sequences if there is exactly one nonzero entry in each position of the sequences and

$$(N_{T1} + N_{T2} + N_{T3} + N_{T4})(s) = 0 \; for \; s \geq 1.$$

We denote these $T(n)$.

Let $X, Y, Z, W$ be Turyn Type sequences of lengths $n, n, n, n-1$. Turyn discovered in 1974 [28] that $A = (Z, W), B = (Z, -W), C = X, D = Y$ are base sequences of lengths $2n-1, 2n-1, n, n$ and that the sequences $T1 = \left(\frac{1}{2}(A+B), 0_n\right), T2 = \left(\frac{1}{2}(A-B), 0_n\right), T3 = \left(0_{2n-1}, \frac{1}{2}(C+D)\right), T4 = \left(0_{2n-1}, \frac{1}{2}(C-D)\right)$ are T-sequences of order $3n-1$. We now create four circulant matrices from our T-sequences and denote them $T_1, T_2, T_3, T_4$. These are T-matrices as defined in section 2.4. We plug these into the Goethals-Seidel construction with $W_1 = W_2 = W_3 = W_4 = [1]$ to create a Hadamard matrix of order $4(3n-1) = 12n-4$.

A $TT(n)$ can be written in a more compact format as a hexadecimal number of length $n$. Let $H_i = 4(1-X_i) + 2(1-Y_i) + (1-Z_i) + \frac{1-W_i}{2}, i \in \{0,1,2,\ldots,n-2\}$ and $H_{n-1} = 2(1-X_{n-1}) + (1-Y_{n-1}) + \frac{1-Z_{n-1}}{2}$. A simple example for $n = 10$ follows.

A $TT(10)$ is now given.

```
X ++--++-++-
Y +-++-+-++-
```

```
Z ++++++---+
W ++-+-+++-
```

The hex form for the $TT(10)$ is 049850E236.

Which in turn gives a $BS(19, 10)$.

```
A ++++++---+++-+-+++-
B ++++++---+--+-+----+
C ++--++-++-
D +-++-+-++-
```

Finally we get a $T(29)$.

```
T1 ++++++---+0000000000000000000
T2 0000000000++-+-+++-0000000000
T3 0000000000000000000+0000+-++-
T4 00000000000000000000+--+00000
```

This $T(29)$ plugged into the Goethals-Seidel construction gives a Hadamard matrix of order 116.

3.2 Properties of Turyn Type Sequences

Assume we have a Turyn Type sequence consisting of $X, Y, Z, W$. We observe that the following four operations will preserve a Turyn Type sequence.

1) Negate any of $X, Y, Z, W$.
2) Reverse any of $X, Y, Z, W$.
3) Swap $X$ and $Y$.
4) Negate each element with odd indices of all of $X, Y, Z, W$.

For a $TT(n)$ we have the two important identities

$$x^2 + y^2 + 2z^2 + 2w^2 = 6n - 2$$

$$(f_X + f_Y + 2f_Z + 2f_W)(\theta) = 6n - 2$$

Since $f_X(\theta) \geq 0$ and $f_Y(\theta) \geq 0$ we have $f_X(\theta) + f_Y(\theta) \leq 3n - 1$.

Turyn's Theorem [28] states that if $n$ is even $x_i x_{n-1-i} + y_i y_{n-1-i} = 0, i = 1,2, \dots, n-2$.

When we create a Turyn Type sequence, we can use the above properties to put it in a normalized form. We let $x_0 = y_0 = z_0 = w_0 = 1$ by property 1. We simply negate any of the four sequences which begins with $-1$. We can let $x_n = y_n = -1$ and $z_n = 1$ by property 4 and that fact that $\left(N_X + N_y + 2N_z + 2N_w\right)(n-1) = 0.$ We can further let $x_1 = x_{n-2} = 1$ by a combination of properties 1, 3 and Turyn's theorem. Also, as a consequence of this theorem, exactly one of $y_1\ y_{n-2} = -1$.

A normalized $TT(n)$ satisfies the following four relations.

1) $x_0 = y_0 = z_0 = w_0 = 1$
2) $x_{n-1} = y_{n-1} = -1,\ z_{n-1} = 1$
3) $X \geq \pm X_r,\ Y \geq \pm Y_r,\ Z \geq \pm Z_r, W \geq \pm W_r$
4) $x_1 = x_{n-2} = 1, y_1 y_{n-2} = -1$

We only need to construct a $TT(n)$ in normalized form, since we can always use some combination of the four operations above to put them in normalized form. If two $TT(n)$ can be put into the same normalized form, they are considered equivalent. The number of $TT(n)$ for a particular order is the number of distinct normalized forms for that order.

We note that $n$ must be even. If $n$ were odd, $(N_X + N_Y + 2N_Z + 2N_W)(1) \equiv 2\ (mod\ 4)$ which means we do not have a $TT(n)$.

### 3.3 Algorithm for Constructing Turyn Type Sequences

We now give an efficient algorithm to construct Turyn Type sequences. We want to construct $X, Y, Z, W$ in normalized form. The algorithm can be broken into six steps and is an improved version of the approach used by Kharaghani [17].

STEP 1 – Find all possibilities for the first and last elements of $X, Y, Z, W$.

We begin by finding all the possibilities for the first and last 6 elements of $X, Y, Z$, and the first 6 and last 5 elements of $W$. Note that the lengths of $X, Y, Z$ are $n$ and the length of $W$ is $n-1$. Call these strings $X^*, Y^*, Z^*, W^*$ as shown below.

$$X^* = (x_0, \dots, x_5, *, \dots, *, x_{n-6}, \dots, x_{n-1})$$

$$Y^* = (x_0, \dots, x_5, *, \dots, *, y_{n-6}, \dots, y_{n-1})$$

$$Z^* = (x_0, \dots, x_5, *, \dots, *, z_{n-6}, \dots, z_{n-1})$$

$$W^* = (x_0, \dots, x_5, *, \dots, *, w_{n-6}, \dots, w_{n-2})$$

Since $X, Y, Z, W$ will be in normalized form, we can let $x_0 = y_0 = z_0 = w_0 = 1, x_{n-1} = -1, y_{n-1} = -1, z_n = 1, x_1 = x_{n-2} = 1$. Also, by Turyn's theorem we have $y_1 y_{n-2} = -1$. We also use the facts that $\geq \pm X_r,\ Y \geq \pm Y_r, Z \geq \pm Z_r, W \geq \pm W_r$. The conditions $\left(N_X + N_y + 2N_z + 2N_w\right)(s) = 0\ for\ s \geq n-6$ must also be satisfied.

There are 1911620 such possibilities for $X^*, Y^*, Z^*, W^*$ that meet the above requirements. We experimented with using more or less than the first and last 6, but it smaller values slowed the program down and larger values used much more memory and did not speed up the program.

Let an $(x, y, z, w)$ quadruple be a solution in integers of $x^2 + y^2 + 2z^2 + 2w^2 = 6n - 2$. Let a $(\zeta, \omega)$ pair correspond to all quadruples such that $z = \pm\zeta$ and $w = \pm\omega$. For example, if $n = 24$, the possible $(\zeta, \omega)$ pairs are $(6, 5), (6, 3), (6, 1), (2, 7), (2, 3)$. When we run the program, we enter the value for $n$ followed by the $(\zeta, \omega)$ pair. If we enter $n = 24$ and $(\zeta, \omega) = (2, 7)$, the program will find the $TT(24)$ with $z = \pm 2$ and $w = \pm 7$. Based on the $(\zeta, \omega)$ we have chosen, we compute all possible values for $(x, y)$. In this case they will be $(x, y) = (0, \pm 6), (\pm 6, 0)$. If we want to find all $TT(24)$ we would need to run the program five times, one for each $(\zeta, \omega)$ pair.

STEP 2 – Find the possible strings for Z

We find all $\pm 1$ strings $Z$ of length $n$ which satisfy the following.

1) Sum of the elements of $Z$ is $\pm z$
2) $f_Z(\theta) \leq 3n - 1, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k \epsilon \mathbb{Z}$
3) $z_0 = 1, z_{n-1} = 1$
4) $Z \geq Z_r$

STEP 3 – Find the possible strings for W

We find all $\pm 1$ strings $W$ of length $n - 1$ which satisfy the following.

1) Sum of the elements of $W$ is $\pm w$
2) $f_W(\theta) \leq 3n - 1, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k \epsilon \mathbb{Z}$

3) $w_0 = 1$

4) If $w_{n-1} = 1,\ W \geq W_r$.   If $w_{n-1} = -1, W \geq -W_r$.

STEP 4 – Find the possible $Z, W$ pairs.

Find the set of possible $Z, W$ pairs is limited by the following restrictions.

1) $f_Z(\theta) + f_W(\theta) \leq 3n - 1, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$

2) $Z, W$ match one of the $Z^*, W^*$ pairs.

We temporarily store for each $Z$ string found the integer part of $f_z(\theta), \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100$, $k\epsilon\mathbb{Z}$ and for each $W$ string found the integer part of $f_W(\theta), \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100,\ k\epsilon\mathbb{Z}$. This both saves memory and more importantly speeds up computation enormously when computing the $Z, W$ pairs. The reason for this is that it takes many trillions of additions, and adding one or two byte integers is much faster than floating point additions.

STEP 5 – Find the first and last 6 elements of $X, Y$

For each $Z, W$ pair that we have found, we now find all the possibilities for the first and last 6 elements of $X, Y$ by comparing the looking at the $X^*, Y^*, Z^*, W^*$ found previously. We take each possibility for the elements of $X, Y$. One example for $n = 36$ is below.

| | 0-5 | 6-29 | 30+ |
|---|---|---|---|
| X | +++++- | | -++++- |
| Y | ++---+ | | -+++-- |
| Z | +++-+- | +--+++-+-++++--+-++++---- | -+--++ |
| W | +++++- | --++---+---+--+---+-++-+ | +-+-- |

STEP 6 – Complete sequences $X, Y$.

We next build the rest of $X, Y$ from the outside in using the following process.

We begin with $i = 6$.

Let $x_i$, $x_{n-1-i}, y_i$ each be $\pm 1$. We go through all 8 possibilities.

Use Tuyrn's Theorem to find $y_{n-1-i}$ for each of the above possibilities.

We now check two things.

1) $(N_X + N_y + 2N_z + 2N_w)(n - i - 1) = 0$.
2) It is possible to get to an $(x, y)$ pairs which satisfies $x^2 + y^2 + 2z^2 + 2w^2 = 6n - 2$.

If both of these are true and $i < \frac{n}{2} - 1$, increase $i$ by 1 and repeat. If we have gone through all 8 possibilities for $x_i$, $x_{n-1-i}, y_i$ we backtrack and decrement $i$.

If the above two conditions are met and $i = \frac{n}{2} - 1$ we've completed $X, Y$. We now check that $(N_X + N_y + 2N_z + 2N_w)(s) = 0, 1 \leq s < \frac{i}{2}$. If it is, we have a $TT(n)$. There is a slight possibility that we end up with $Y < -Y_r$. If this is the case, we simply throw it out since it is not in standard form.

3.4 Computational Optimization

Several techniques have greatly improved the performance of the algorithm that we explain below. The algorithm was programmed in Microsoft Visual C++ and run on an Intel i7 3930K running at 3.2 GHz with 6 CPUs was used for the computations. All running times used in this paper are in are expressed in CPU hours. The approximate running time for enumerating $TT(n)$ is $10^{\frac{n}{2}-12}$ processor hours, so it takes exponential time. This made a full enumeration in a reasonable amount of time feasible for all lengths up to $n = 32$. In chapter 4 we discuss the details of the enumerations and in chapter 5 we explain how it was used to find examples of $TT(n)$ up to $n = 40$. We list some of those techniques here.

1) We create huge arrays to store the values of $f_Z(\theta)$ and $f_W(\theta)$. These values are computed as 8 byte double precision numbers. However, storing them as 1 or 2 byte integers was found to be very helpful. Not only do they require less memory, but STEP 4 of the algorithm requires an immense number of additions. Addition of integers is significantly quicker than addition of double floating point numbers. For enumerating all of $TT(n)$ we store them as 2 byte integers as $\lfloor 256 f_Z(\theta) \rfloor$ and $\lfloor 256 f_W(\theta) \rfloor$. We lose a small amount of precision and have potential roundoff errors so we check for $f_Z(\theta) \leq 3n - 1 + \frac{1}{256}$ and $f_W(\theta) \leq 3n - 1 + \frac{1}{256}$. On the other hand, when we are just looking for one $TT(n)$ for a particular order, we store them as 1 byte integers as $\lfloor f_Z(\theta) \rfloor$ and $\lfloor f_W(\theta) \rfloor$. Since we are only searching part of the search space, we can afford to lose a bit more precision. Also, adding 1 byte integers is even quicker than adding 2 byte integers.
2) We store as much data as possible in arrays to improve efficiency. We needed to compute cosine of a function trillions of times, but the function could have at most 10,000 different values, so we stored these 10,000 values in an array. We cycle through eight different values in order for each of three variables $x_i$, $x_{n-1-i}, y_i$ in STEP 6. We created an array which stored the difference between the kth value and the k+1st value, which saved a significant number of clock cycles.
3) When first running the program, we input a $(\zeta, \omega)$ pair rather than values for $(x, y, z, w)$. This allows us to look at multiple $(x, y, z, w)$ quadruples in a single run of the program. We note that for a $(\zeta, \omega)$ pair $z^2 + w^2$ is constant. This implies that $x^2 + y^2$ is also a

constant. We find all possible $(x, y)$ solutions. We then create a $2x2$ array in which the entries are the distance to the nearest $(x, y)$ solution. The $(x_1, y_1)$ element of the array is $\max(|x_1 - x|, |y_1 - y|)$ where the $(x, y)$ value is chosen to minimize this quantity. This array is used extensively in STEP 6 of our algorithm. As an example, suppose the $(x, y)$ solutions are $(6, 0), (-6, 0), (0, 6), (0, -6)$. Further suppose that our current partial sums for $X$ and $Y$ are 14 and -4 and we still have 6 entries left to fill. The $(14, -4)$ element in our array is 8 since the nearest solution to $(14, -4)$ is $(6, 0)$. Since $8 > 6$ there is no way that we can form a $TT(n)$ regardless of how the remainder of $X, Y$ are filled in. We note this and backtrack immediately.

4) Throughout the program we have used 100 equally spaced values for the variable $\theta$. Specifically, $\theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$. If we increase the number of equally spaced $\theta$ values, we have fewer $Z$ strings and $W$ strings and $Z, W$ pairs, which results in a quicker execution of STEP 6. However, STEPS 3, 4 and 5 generally run slower since there are more strings to check. A value around 100 generally minimized the total running time. When enumerating the number of $TT(n)$ for orders above $n = 32$ it would be wise to take some time to optimize this number further for each $(\zeta, \omega)$ pair.

5) As critical as what to include in the algorithm is what not to include. Every operation utilizes processor cycles. For example computing $f_X(\theta), f_Y(\theta)$ and using that information slowed down the algorithm so we did not compute them.

6) We checked equivalence of Hadamard matrices of orders 404, 428, 452 and 476 in chapter 5. We attempted to use 4-profiles which is a relatively simple computation. The number of operations needed to compute 4-profiles for a Hadamard matrix is $n\binom{n}{4}$ which is of the order of $n^5$. This was feasible and took around 10 minutes per Hadamard

matrix. Fortunately, they all turned out to be inequivalent. Had any two 4-profiles been equivalent, the test would have been inconclusive for those matrices and we would have had to use another method. 8-profiles are not feasible for orders this high. Computing symmetric hamming distribution polynomials is significantly more complex than using profiles, but would have been feasible for lower values of $k$.

# CHAPTER 4

# ENUMERATING TURYN TYPE SEQUENCES

## 4.1 Count of Turyn Type Sequences of Lengths up to 32

Best et. al [3] have enumerated all $TT(n)$ up to $n = 32$. The results that obtained were identical for all $n$ up to 30. However, for $n = 32$ we found 66 additional classes in addition to the 6226 found in [3] for a total of 6292. 6226 of the 6292 sequences we found were identical to the 6226 found in [3], but the other 66 were inequivalent. We verified this inequivalence, taking each one of their sequences and converting it from the format they used to the format used here by the use of the four operations.

The running time to count all $TT(n)$ classes increases by approximately an order of magnitude for each increase in $n$ by 2. There are typically more $(\zeta, \omega)$ pairs for lengths in which $n \equiv 2\ (mod\ 4)$ than lengths n$\equiv 0\ (mod\ 4)$, which is reflected in the running times. The run time for $n = 32$ was approximately 8200 hours, which is an improvement from the approximately 50000 hours reported by Best et al [3]. The approximate running time and the number of $(\zeta, \omega)$ pairs is also shown in the table.

TABLE II

TURYN TYPE SEQUENCES OF LENGTHS UP TO 32

| Length | Number | Hours | $(\zeta, \omega)$ |
|---|---|---|---|
| 2 | 1 | | 2 |
| 4 | 1 | | 2 |
| 6 | 4 | | 4 |
| 8 | 6 | | 2 |
| 10 | 43 | | 5 |
| 12 | 127 | | 5 |
| 14 | 186 | | 7 |
| 16 | 739 | | 4 |
| 18 | 675 | | 7 |
| 20 | 913 | | 6 |
| 22 | 3105 | | 10 |
| 24 | 3523 | 1 | 5 |
| 26 | 3753 | 9 | 10 |
| 28 | 4161 | 80 | 9 |
| 30 | 4500 | 1100 | 12 |
| 32 | 6292 | 8200 | 7 |

The computing time consistently increases by approximately one order of magnitude. The number of $TT(n)$ never decreases, except slightly between $TT(16)$ and $TT(18)$. The 66 additional classes of $TT(32)$ not found in [3] are shown in table III.

TABLE III

NEW TURYN TYPE SEQUENCES OF LENGTH 32

| Num | Hex |
|---|---|
| 1 | 070825988484FB35F816EECC1EF4E926 |
| 2 | 070F9C19C715CE204559F6D73A93B526 |
| 3 | 079E1024357445F1953CCC0BF8B75326 |
| 4 | 01CBFEA06764660E493CD23C4EA93966 |
| 5 | 0781429C8F4EAC8FBD5C626C9282B326 |
| 6 | 07CE999D2E950FB3BEAACEB852C1AB26 |
| 7 | 0780450A707CA63344EC9CA2093D9D26 |
| 8 | 074012D49D7D9314359EAE5209877326 |
| 9 | 00C0856A7DADE5536FC48FBCDD1E8B66 |
| 10 | 018B4C5146BD6DFD567C7C024DB33166 |
| 11 | 07CBB5263D0E809803808AB6D5D01B26 |
| 12 | 01857332AED3023C66A4B7A17790F166 |
| 13 | 000F00AB45A1D3571F49803E3155A666 |
| 14 | 0741AE8EA4FAAD0C485CF621BD80B326 |
| 15 | 078A8421730486D6041EC99D6A3EE326 |
| 16 | 078B637373C09A115802B9A04F50D326 |
| 17 | 0781184CAE2644A4D13EC8719E356D26 |
| 18 | 061D23C39FB3E0C16666AE508BA46B26 |
| 19 | 0742C273173BAC95EF9C3B359E57A326 |
| 20 | 078BA9FE7029AA2638EE085C57EC3326 |
| 21 | 07880F30BFF5B87D9DDCE6A1945ADD26 |
| 22 | 07008970CA53D81047F6630B51F39926 |
| 23 | 06AF55E90A85813B0B6E22DAFA21B326 |
| 24 | 0182659FA2C670B433733741ACF2A166 |
| 25 | 07819679BA0DBC3376AC7BFFDD11AD26 |
| 26 | 00E0DF004188B658DEFCE263A958B966 |
| 27 | 0781EE969662BEC2A44C6333D388B326 |
| 28 | 079E0621655F588F7EFE7F3254EB5326 |
| 29 | 078077B6A224EB9431FB3A9D1221AD26 |
| 30 | 01582E57FF15C9D7F5392A7B0C78F166 |
| 31 | 079E8E91D455FFBC5E9B212A53BD9326 |
| 32 | 019436C9F395001AD4A6328617273166 |
| 33 | 075E87ADFC61BFB731B15E66702D9326 |
| 34 | 07863D1CF5C59939F4123AE49B753326 |
| 35 | 078EE990EDE4EC0426B609798C1AB326 |
| 36 | 018BF18480B1D7317437829FEC79D166 |
| 37 | 07839680316ED6AE73159E759F336326 |
| 38 | 078005F6DD09BDDEA8B2399B14E9AD26 |
| 39 | 06A155DA8BBEA6F82BCF93E30921B326 |
| 40 | 07014EB857F5E8843DFA373D135DB526 |

TABLE III (continued)

NEW TURYN TYPE SEQUENCES OF LENGTH 32

| Num | Hex |
|---|---|
| 41 | 078EA0F08AC81D4F5EAA062F7B4CB326 |
| 42 | 079EFE363DCE519BD19C4A9B25B59326 |
| 43 | 00006341D714A3882F9F0A25B172A566 |
| 44 | 0084B80AE7057CEB254207C7D9CD1E66 |
| 45 | 07C1991F998744317B2CDEDEBB0C5B26 |
| 46 | 0742AEE60A0F04F735C8A4FB16B29326 |
| 47 | 070085B4C0DEAF15DB92AA8BCC3E9926 |
| 48 | 079C240E74018D09D47D7A22541A7326 |
| 49 | 0758F643E090D1590C6B904A420B3326 |
| 50 | 01815A79083C475C40334A263DF1B166 |
| 51 | 0788A0A585F9CE3E88953B3011B2DD26 |
| 52 | 07181BEA8918CE1EBBBA34FF09CBD526 |
| 53 | 0199C2AE48206C8F5C705A3C816B3166 |
| 54 | 0782BFFE099CCE573DB693FB57AD6326 |
| 55 | 01983F185C7897CD98211EA2EB793166 |
| 56 | 078E34144546B7FAF62E12DF2AC57326 |
| 57 | 078E10981CEA8EB9D15E17992285B326 |
| 58 | 004BDFF151AB365F72F5EF8D69A51366 |
| 59 | 070198865F7232C9F695BF5D131EB526 |
| 60 | 0704BCCFC82A9CE719A2E6F664AD3926 |
| 61 | 078E3016DDBD1ED53BB99246D7457326 |
| 62 | 078B6BF222635192B24C4E7847E0D326 |
| 63 | 074BCE38A742A156DF7E5D1D26773326 |
| 64 | 0143212A7FB45B55EEF23F4C2A747166 |
| 65 | 078B952DC402D0AA0095471BB81E3326 |
| 66 | 07084CAEA47BF06A1C6BE2D04C7DD926 |

## 4.2 Estimate of the Count of Turyn Type Sequences of Lengths 34 and 36

For $n = 34$ and higher, the running time that the algorithm would require to enumerate the number of $TT(n)$ is not feasible. We are, however, able to compute a rough estimate of the number of $TT(34)$ and $TT(36)$ by using the algorithm as follows.

We take the list of $Z$ strings found in STEP 2 of the algorithm. We divide them up into 32 groups by taking the indices of the list modulo 32. We follow the same procedure for the list of $W$ strings. We then create a random list of ordered pairs of integers between 0 and 31. The first ordered pair came out to be $(18, 24)$ so we used group 18 of the $Z$ strings and group 24 of the $W$ strings. This reduces the search space by a factor of 1024. We then repeat for other ordered pairs. We chose to divide them into 32 groups because each group was likely to have several matrices, assuming the numbers of $TT(34)$ and $TT(36)$ are similar to those of slightly smaller lengths. Also, dividing them up into 32 groups makes the running time for each reasonable. There are 12 $(\zeta, \omega)$ pairs for $n = 34$ and 10 $(\zeta, \omega)$ pairs for $n = 36$. For $n = 34$ and $n = 36$ we did 16 runs per $(\zeta, \omega)$ pair which is $\frac{1}{64}$ of the total search space.

For $n = 34$, the program found 6, 5, 7, 3, 4, 4, 4, 4, 9, 4, 4, 1, 6, 6, 5, 7 sequences for the 16 ordered pairs or a total of 79. This leads to an estimated total count of $79 * 64 = 5056$ for $TT(34)$. Each ordered pair took about 120 CPU hours. The total running time for the 16 ordered pairs took about 1920 hours, so it took about 24 CPU hours on average to find each $TT(34)$.

For $n = 36$, the program found 2, 3, 4, 7, 6, 5, 4, 3, 2, 3, 0, 6, 2, 5, 2, 4 sequences for the 16 ordered pairs or a total of 58. This suggests the total count of $TT(36)$ is approximately $58 * 64 = 3712$. Each ordered took about 800 CPU hours. The total running time for the 16 ordered pairs took about 12800 hours, so it took about 220 CPU hours on average to find each $TT(36)$.

We ran the same tests for lengths between 24 and 32 and since we know the actual number, we are able to compute the error percentage. These results in table IV show our estimates for $n = 34$ and $n = 36$ are likely to not be too far off.

TABLE IV

ESTIMATES FOR THE NUMBER OF SEQUENCES

| Length | Estimate | Actual | Error |
|---:|---:|---:|---:|
| 24 | 4544 | 3523 | 29% |
| 26 | 3008 | 3753 | -20% |
| 28 | 4608 | 4161 | 11% |
| 30 | 4992 | 4500 | 11% |
| 32 | 5760 | 6292 | -8% |
| 34 | 5056 | | |
| 36 | 3712 | | |

# CHAPTER 5

# LONGER LENGTH TURYN TYPE SEQUENCES

## 5.1 Methods

There were very few known Turyn Type sequences with lengths above 32 before 2012. The search spaces increase exponentially in size for lengths of 34 and above. Based on our results from section 4.2, it would take about 120000 CPU hours to go through the entire search space for $n = 34$ and 800000 CPU hours for $n = 36$, so we need a better way to find $TT(n)$ for $n \geq 34$. We won't be able to enumerate all of them, but will be able to find some for lengths up to $n = 36$.

We know that $f_Z(\theta) + f_W(\theta) \leq 3n - 1$. The program we used here is identical to the one used to count the $TT(n)$ for $n \leq 32$ except that we must enter three more numbers to further restrict $f_Z(\theta), f_W(\theta), f_Z(\theta) + f_W(\theta)$ which allows us to shrink the search space to more manageable sizes. The idea is to decrease the running time faster than decreasing the number of Hadamard matrices found to minimize the time per Hadamard matrix found. The trick here is finding appropriate restrictions on these values. We made a good prediction by observing patterns from lengths 32 and below.

We found the least values of $m$ for which the below restrictions yield 1, 10, 100 $TT(n)$ for orders 24, 26, 28, 30 and 32. These are shown in table V.

$$f_Z(\theta) \leq m, f_W(\theta) \leq m, \lfloor f_Z(\theta) \rfloor + \lfloor f_W(\theta) \rfloor \leq 3n - 4, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k \epsilon \mathbb{Z}$$

TABLE V

LEAST VALUES OF M NEEDED FOR 1, 10, 100 TURYN TYPE SEQUENCES

| Length | 1 $TT(n)$ | 10 $TT(n)$ | 100 $TT(n)$ |
|---|---|---|---|
| 24 | 46 | 49 | 53 |
| 26 | 51 | 56 | 60 |
| 28 | 58 | 61 | 64 |
| 30 | 60 | 64 | 69 |
| 32 | 66 | 70 | 73 |

Keeping $\lfloor f_Z(\theta)\rfloor + \lfloor f_W(\theta)\rfloor$ no more than $3n-4$ lowered the running time while minimally decreasing the number of $TT(n)$. The $m$ values increase by approximately 5 each time $n$ increases by 2. We used this as an approximate guide for choosing restrictions for the orders above 32.

5.2 New Turyn Type Sequences for Length 34

For $n = 34$ we can use the fact that $f_Z(\theta) + f_W(\theta) \leq 101$. If we do not need to find every $TT(34)$, we can restrict the search space further. Since we know $f_X(\theta), f_Y(\theta), f_Z(\theta), f_W(\theta) \geq 0$ and $f_Z(\theta) + f_W(\theta) \leq 101$ we try further restricting $f_Z(\theta), f_W(\theta)$. We want to minimize the time it takes to find each $TT(34)$. The below restriction was based on section 5.1 and some trial and error.

$$f_Z(\theta) \leq 80, f_W(\theta) \leq 80, \lfloor f_Z(\theta)\rfloor + \lfloor f_W(\theta)\rfloor \leq 98, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$$

We found 119 new inequivalent Turyn Type sequences in 738 CPU hours or about one every 6 hours. The average time to find a $TT(34)$ is reduced from 24 to 6 hours or by a factor of approximately four.

These sequences give 119 new $T(101)$ and Hadamard matrices of order 404. The computed 4-profile on each of these matrices was different, so that the Hadamard matrices are all inequivalent.

## TABLE VI

## NEW TURYN TYPE SEQUENCES OF LENGTH 34

| Number | Hex Format |
|---:|---:|
| 1 | 048182ABBA195ABE50AE1F3522125F8136 |
| 2 | 000F0E331A2A3989D5E30BD938A9669A66 |
| 3 | 0000EC6139EF03A017A1B698C7ADAAAA56 |
| 4 | 0000F072C2C373B02B4E9C8B9AA255AA56 |
| 5 | 008F8691C27D5F440AF2A26354A1234366 |
| 6 | 0002BEA4373C3A542B212458C5F09F4666 |
| 7 | 05B28DA2B5D1880FD7BB31BBD261A16C36 |
| 8 | 00A2E999A3233EAD61AD4A84B210C79C66 |
| 9 | 060A5E78CCC6E05D726376D49BC17ECB26 |
| 10 | 06FD94CA9E263A13F86A2409430A109B26 |
| 11 | 008CC4638E69F34542FD87E252521AA366 |
| 12 | 015187B0436A69DBAD3B3D31BC4ED3B166 |
| 13 | 0041A038A6EB7A5A6C00FE3610F98CA356 |
| 14 | 04D9FF066CEAAF0F1A45B028A128942A36 |
| 15 | 0405F71B3EEB1669C7FFC7155BCBC53836 |
| 16 | 0011F33FB5996B5A60FFD2C3D27988BA56 |
| 17 | 049751CD3B79B62D399B21ECE987631236 |
| 18 | 000A0CCB5ED9A66E9F7CE2C59E0BA72966 |
| 19 | 0493B9FF5776B4A751F2CDDCE3952DAC36 |
| 20 | 049BB6121E8ED9284034C74E756AC22236 |
| 21 | 008ABE698CA150617D9E606C72C3630366 |
| 22 | 05E845FE701EB6A070A2EE899094230A36 |
| 23 | 00F882A8D14E65F8A1C4CFA3A722923966 |
| 24 | 00B40FB8E6AC637FC460B0A2251C56D256 |
| 25 | 052A23A968467163348F700CCFCE883436 |
| 26 | 0789F18B37C988BCF86FF33B15EE553D26 |
| 27 | 061A1246707F0EA3D46268809CDDAA1726 |
| 28 | 070E74F199819EE117784D7F336923B526 |
| 29 | 06A0B0927998C260245186CE3C7FACB326 |
| 30 | 05BA4E68F2304342167C805449FD712236 |
| 31 | 000004B4B78680EF63A49DE31DC2369A66 |
| 32 | 01947FC02F8FC7A3B27115A075BB9F3166 |
| 33 | 008CAA8F244B56F61A2A1EF0E9BD12AE56 |
| 34 | 041B989DC5B3AB176F15D2AC369FED3836 |
| 35 | 0097268DCB522FF8CAC868AF2A721AC356 |
| 36 | 000A3A1CE950C7DEE456C9603875F62A66 |
| 37 | 00B5C363FFEDD5176DD9FA878A539AD166 |
| 38 | 074031C3055B996AAF3F3D1CD6B9497326 |
| 39 | 06B82D460EA64DFDC8A57C0078FEBBE326 |
| 40 | 0181C05FFF899A3EC879E3D2A953B97166 |

TABLE VI (continued)

NEW TURYN TYPE SEQUENCES OF LENGTH 34

| Number | Hex Format |
|---:|---:|
| 41 | 00808D77C8C315A2B0B0296507306F8E66 |
| 42 | 00889E9824BEF8A4EB03F49134C3AE2D56 |
| 43 | 040C92611FF2EC63EB7C68574672AC7836 |
| 44 | 04131891A550B357A0EE9D41EFA335B436 |
| 45 | 000BD1D667F66F0AE416AF261CC74B2966 |
| 46 | 04566988520501A75F1E9A0AAC0FF02236 |
| 47 | 000B6A3A87A2F64BB10E359CEC26121656 |
| 48 | 00F000421F4A86D2346B123D994D6AAA56 |
| 49 | 00F0A1F2C6BAB1722A638CCDDB68BC9A66 |
| 50 | 04AFDD053FE8BEE66E388ED867C696B036 |
| 51 | 052EC426796D819DC67C8C9EFCFB0A6836 |
| 52 | 052B7023F00D8D33E545428A94A882E436 |
| 53 | 079488AD183220B2F64184A42651EDD326 |
| 54 | 06028FCD8EE105F6D815C447A278635726 |
| 55 | 06E8806077802DB96321459E1E63AE3B26 |
| 56 | 0670FC71E80C18220689085B355E6A9326 |
| 57 | 000C8CE37A9E3ED2ACB69B4E234651BA66 |
| 58 | 050DD6DF5391F44E90B239B243B7D58A36 |
| 59 | 0004A626E5897F6D827C6F13EADA2C3966 |
| 60 | 041FDD699D475CE1FAB6B03F9D1166B836 |
| 61 | 049E7C5F349FDF75E82365BC199152A136 |
| 62 | 000EF3D5C52CC009CA1A6A5A39EA875A66 |
| 63 | 00063F46B654AC2ACBDA8CF30F1DAB2A66 |
| 64 | 00BF10E98C6C2AE65F05E87272F8BAA256 |
| 65 | 04A2816AB6FAF4108E44E5DA100D7DA036 |
| 66 | 057A06D0ECED4B27AFD40DABB4AA282236 |
| 67 | 00620269DAFE7209AC2951A4C5CF48B166 |
| 68 | 04E36D458046941B2A3BCECE9D228F8836 |
| 69 | 00A729091BC8EA72E74EE5FA3824E9D266 |
| 70 | 046224E2A37ACD0D7FA86B038D78EA5036 |
| 71 | 0780DF4AFDD30C8660109447AA02A7AD26 |
| 72 | 0698A9D084561043FB6248D2CEA6F01F26 |
| 73 | 0700E4C0C13BD04871CEA6F66956E9A926 |
| 74 | 06096F5C2866EE524D9CA5332AA1B0EB26 |
| 75 | 070DE67DE3395BC9B3EBDC146891D57926 |
| 76 | 079E51DD5358BFBF8C814F21A19A6F5326 |
| 77 | 075F8748DCAB8BD914241FF197F3E19326 |
| 78 | 07064B942386FD9D427EAA2EBAC0CF2926 |
| 79 | 0613B40EDF5D7BB5BEFC1C817A95D18726 |
| 80 | 078A2FBC34084A644CC1DFF80B6D68D326 |

TABLE VI (continued)

NEW TURYN TYPE SEQUENCES OF LENGTH 34

| Number | Hex Format |
|---|---|
| 81 | 0001078BF22865317245E1D86B1C1BA956 |
| 82 | 01810E2DD5D5F20753E748F6C447B4B166 |
| 83 | 045152A593858C44AEF18123B0F3429236 |
| 84 | 002314AA96375D3938D66329F32F24B866 |
| 85 | 00041702AA1144FEC4BA3CB611A525DA56 |
| 86 | 041A9BB7721A221E6C6AB50AADC302F436 |
| 87 | 04866B3ECA37545160AD2018D4B8031C36 |
| 88 | 05321269737AC7B86EE20803BDE1646836 |
| 89 | 069080CCACFF3EC451F4BA549F655E8F26 |
| 90 | 069BB54DD38317B82720F8AEB763310F26 |
| 91 | 06A94C840141F120BEB594738A00BCF326 |
| 92 | 008848B7606ADB649ECFCB0D53D0DE2E56 |
| 93 | 00041FBCB36ED693F85D076D5C715A3956 |
| 94 | 0040F1399DDE35EC5DA5376781F6949366 |
| 95 | 001C3F28C0EF0414ACC62B57A9DB89A966 |
| 96 | 00912E27DCC86A07060BCE398BF8949E56 |
| 97 | 001B0C1F0995A70FF65A1133F95A86E556 |
| 98 | 040FB74DC585CE10E98A74EEE6831DA836 |
| 99 | 05ACD3C058DC97CFF6A4C2A6C19964A236 |
| 100 | 00B4D751558E1ADD9E87E5413FA317E256 |
| 101 | 0532E1899660A327BEF69E7102ECA46836 |
| 102 | 04ABBF31AABE8563417EE0ACE0B68ED036 |
| 103 | 06888D8A94E5A66FC4BD1036DFD04C3F26 |
| 104 | 069C197DAB3AC5CCD468C609EF6005AF26 |
| 105 | 078027F9C7A9BBD66E05CCF2E6EA34AD26 |
| 106 | 079E02FE6D159C8E0890A2C85E49AB5326 |
| 107 | 07C2CFEE3DE9C89F244C3A084459758B26 |
| 108 | 00040D835ED78C4FF492613B4F4D663956 |
| 109 | 000000FF8073435B1A2C9F834D56A96956 |
| 110 | 0081C65346974440FBADCEF03D53D94366 |
| 111 | 0448C6B6F74DE117C5178691D8FE051236 |
| 112 | 00857458544CB2387FE9604FD1DE3D2356 |
| 113 | 00853DCA651E118CCA8F8BAB12D87B2D56 |
| 114 | 0000F007BC34B82F63451D19AE19956A56 |
| 115 | 052A986373790081C9A1481D7C4F12E436 |
| 116 | 00724AF555DF2B11C565C54A1C0BCF9266 |
| 117 | 0063EE647251C5BE175EF872ACF3B5B266 |
| 118 | 0783C2F74550CA7BA2332BAD32E49B5326 |
| 119 | 06E023998752B614370502BFEFE29B6B26 |

## 5.3 New Turyn Type Sequences for Length 36

For $n = 36$ we can use the fact that $f_Z(\theta) + f_W(\theta) \leq 107$. If we do not need to find every $TT(36)$, we can restrict the search space further. Since we know $f_X(\theta), f_Y(\theta), f_Z(\theta), f_W(\theta) \geq 0$ and $f_Z(\theta) + f_W(\theta) \leq 107$ we try further restricting $f_Z(\theta), f_W(\theta)$. We want to minimize the time it takes to find each $TT(36)$. The below restriction was based on section 5.1 and some trial and error.

$$f_Z(\theta) \leq 84, f_W(\theta) \leq 84, \lfloor f_Z(\theta) \rfloor + \lfloor f_W(\theta) \rfloor \leq 104, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$$

We found 101 new inequivalent Turyn Type sequences in 3600 CPU hours or about one every 36 hours. The average time to find a $TT(36)$ is reduced from 220 to 36 hours, or a factor of approximately six. These sequences give 101 new $T(107)$ and Hadamard matrices of order 428. We computed 4-profiles on each one and on the one found in [17]. They were all different, so the Hadamard matrices are all inequivalent. Thus we have over 100 new different Hadamards of order 428.

TABLE VII

NEW TURYN TYPE SEQUENCES OF LENGTH 36

| Number | Hex Format |
|---:|---:|
| 1 | 00977032312F05C79E4D07FA574644AE2356 |
| 2 | 044462FB76EA5C37870134A1340DF6902136 |
| 3 | 0700A9C5AE9D736BDBCBF39C81823BF29926 |
| 4 | 00E778575CC16201A09E8A4256B2C3133A66 |
| 5 | 04F71F790EB35336ED9714515E462D683836 |
| 6 | 06F9FC6C8AA4C8A6D389E3F431C09DAB2B26 |
| 7 | 0008037BBC0257C5215A0BCE5AACF2683A66 |
| 8 | 00002BA231EE941A852D1ADCB664B1F46A66 |
| 9 | 00AEA04F61A8584E615F8CF3FE7242507166 |
| 10 | 000857A880FE34F181B0A7267462DEC23966 |
| 11 | 00AA0EF50E306C0B0A09165E994527B7E266 |
| 12 | 048A7AD26A77229009FA8E8BCC11B4EE1C36 |
| 13 | 0568FACF009E1591A7204C28A3A467F42236 |
| 14 | 05722A51041682F6B1BED643C8CABC288236 |
| 15 | 00C84EF502E4FF26919E1A4AFAA8F8921B66 |
| 16 | 008DC22CF00696B5A6EDDFD1C6A768BAAD66 |
| 17 | 000F7F457529E34FFEB47D95F81321719A66 |
| 18 | 04E0A7B516CDCCD6A8CF0487BAF4E3F26836 |
| 19 | 078808644FA1956E753C6E10615EF2EAED26 |
| 20 | 00B4546D853DF266605EC375A5C3BFCDD166 |
| 21 | 00819B060B8E8E3DF91987B29DEAD4328366 |
| 22 | 045BB656FF55100DF4D66486CE79C3C3C236 |
| 23 | 00C7B2154445385982A0E3CA3D132B430B66 |
| 24 | 0486323476B8A94594FFD13201F01B5A0136 |
| 25 | 070562C8047D92A908183E53623B358C3926 |
| 26 | 0491940F7159A333A4124793C38CAAC35236 |
| 27 | 0004BF94B8958668E816F3031C2131613666 |
| 28 | 04110CE70D3ECEE5F70B15668B47D5757836 |
| 29 | 00464B53EC9D707D541F617D4D6AB2E12356 |
| 30 | 00EAFEB59D59DA58FD56CE35DEB2CD643966 |
| 31 | 040D9D136A5A029D2ACBAC45EE0E45A2B836 |
| 32 | 01510BF01F9308F7E7D7D52B8F884AF9B166 |
| 33 | 00018A65C7E2665298E24F0179F8C3EC9566 |
| 34 | 048C46A205B8C5983807F2F4E12BAF0C6136 |
| 35 | 008876AA0E5227070F7BCB386E84E3D11E66 |
| 36 | 04982EEEE274D63A95C2C429D1645A58E136 |
| 37 | 04168D75F00FAE8E026A9C809A792E6DD836 |
| 38 | 000A724CCF32E60E281B9415649B4F83C966 |
| 39 | 000A62F8DDCE93AC8E709EB25766FA5E2966 |
| 40 | 00B5DEFD3F06AC01AE92484D38846665E166 |

TABLE VII (continued)

NEW TURYN TYPE SEQUENCES OF LENGTH 36

| Number | Hex Format |
|---:|---:|
| 41 | 05BE8199415473578D63E3AD316DDF90A236 |
| 42 | 00E3F5DDBE4ED707652315C97DB385389966 |
| 43 | 05F66976BCAA258C907E52C8D1502C020A36 |
| 44 | 007BB3C62190C9D0D2B47AC48E660A90E166 |
| 45 | 0013DCDCDB8161AE3D79627D93197AB96A56 |
| 46 | 05603E6AE40418B854CCF0E71B0AD3788236 |
| 47 | 061AD51D3F791BFEC2657BF4FE49782B1726 |
| 48 | 078E8BE8FD062017E6C4364628943B33AD26 |
| 49 | 0697227F3E98ACB795CEE2402CA98E780F26 |
| 50 | 00C847DA4A743B06AAF2E7D9DE0FE6C12B66 |
| 51 | 049BDF4F5B166AFCE73797EDFA1D90A71236 |
| 52 | 00A41D73C2B298F49ACFEBD043A660AAF266 |
| 53 | 00BEACC7D60700A6693F2EA62B18345E9166 |
| 54 | 04C894699F576650928E52F000A3E0FE0A36 |
| 55 | 04116ADA22D565615BECACF2E6B8C6227836 |
| 56 | 0004F6ED48CA5E85C2B4137C342EBA063956 |
| 57 | 0009FEEC9EAB3B45B82E2C2B0E52A746DA56 |
| 58 | 048FABCDE66EFA46DF95220652C7A83E8136 |
| 59 | 00ABA4A05C8F791C5E40443CBD6E72E1D266 |
| 60 | 00B7FBF8CD097C81CFBA91B3C5A717071166 |
| 61 | 00BFFF785A5D57119930A4F14F2C316A9166 |
| 62 | 0508D52FC24D84C6C8FBD7138D6B65D42636 |
| 63 | 00143A65ED542060DC58AE5823B513CB1966 |
| 64 | 00606FA4B353B3466712FF608F8F0DA3A256 |
| 65 | 00EBB4EC342BFD7767DDF09AD4F7A732DA66 |
| 66 | 048D1589F789CCC72AE81A653CDACFE96136 |
| 67 | 00C88452DDEFE5482DA93F0A45BBACF1CB56 |
| 68 | 0491D15F68653046125412BB0CDE5FA99236 |
| 69 | 0408F71A7E42F3676932FD69516226DAE836 |
| 70 | 05A4E7C7957E9B2297ED49F36ECEEBE42236 |
| 71 | 075E9E13A011DF41543C80998BAE85639326 |
| 72 | 07CA7B4571E24B5697FF222EA760CC211B26 |
| 73 | 048840E053B8CD1A307725942DAF99823C36 |
| 74 | 00054AE54C801F5727E51F4680B3C93C3966 |
| 75 | 00710748C5EA995040A182EF24353DE97256 |
| 76 | 00B8D67B74E6256B4B3F11352B3EE1C93266 |
| 77 | 00A97F90E56B326753431E462EE9A38F2256 |
| 78 | 00A7802B59D014C471B3D9F8AB13C96C3166 |
| 79 | 0619BF7B9C6097EE46305A0E5343E2722B26 |
| 80 | 06799B0E7CC8F944A40E1EC5355D450FE326 |

TABLE VII (continued)

NEW TURYN TYPE SEQUENCES OF LENGTH 36

| Number | Hex Format |
|---|---|
| 81 | 008C94F489C798F76E8F15CFE8D2CA329356 |
| 82 | 04500035AC51CD6A70A0EE4B9153E9876136 |
| 83 | 040087BA0C1EC55021775D257A4AED1CB436 |
| 84 | 008DA9AC9FF3AE8C1C87436C4B5C7D1CA356 |
| 85 | 0000B1A70A872D513D96BF98D2C61DA29556 |
| 86 | 044A72172F4A0D45AD7E2E9B32ABF4730236 |
| 87 | 0522196F4D81110C58F0B577BF406CF74836 |
| 88 | 0691CEDFEAB63EFC11777B94F2E5765A8F26 |
| 89 | 008899B4086A6C0F381778BD01E5E213CD66 |
| 90 | 00854794C158373DB4F159E6C2A93DE11366 |
| 91 | 0505E06C51848F445BE1337E208CBCB50A36 |
| 92 | 00B4F4004886B5C337E788ED1CDE65E9D166 |
| 93 | 052A3B12CD02C4DF46E08625459987D61836 |
| 94 | 008889506EF2082B74ED18FA94B053C31E56 |
| 95 | 05011285B45DF54606DA2F1B50DEF3998636 |
| 96 | 04A262DC48D17686D6FA11E3451059AC9036 |
| 97 | 04FB2A8C8E7225B29B1E72E763BC6038D836 |
| 98 | 06B0DA8A1365C674772CFFFAEC6A1C17BE26 |
| 99 | 0000F4ADEFCE908E8D538D83749B9C36A656 |
| 100 | 000EAE07EBFC70BE6B2CB1B384F5CBBDA666 |
| 101 | 06B17BB39738AE9FF6366181D6C392CDA326 |

## 5.4 New Turyn Type Sequences for Length 38

For $n = 38$ we can use the fact that $f_Z(\theta) + f_W(\theta) \leq 113$, but the search space is so large it might take close to 10 million CPU hours. If we let

$$f_Z(\theta) \leq 86, f_W(\theta) \leq 86, \lfloor f_Z(\theta) \rfloor + \lfloor f_W(\theta) \rfloor \leq 110, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$$

the size of the search space becomes manageable. The result was 10 new inequivalent Turyn Type sequences, different from the one found by Best et al [3]. The running time was roughly

7500 CPU hours, so it took approximately 750 CPU hours for each $TT(38)$. One of them is below.

```
X = ++-+-++-++-+++---+---++-+-+----++-+-+-
Y = +-+++-+-+----+++-+-++-+---++-++-+---+-
Z = +-++++++++--+--+--+-+-+++-++---++++--+
W = ++++++++-+-++--+++----++++---+--++--+
```

These sequences give ten new $T(113)$ and Hadamard matrices of order 452. Each of these ten Hadamard matrices are inequivalent since their 4-profiles are all different. The hex form of all ten is in the below table.

TABLE VIII

NEW TURYN TYPE SEQUENCES OF LENGTH 38

| Number | Hex Format | x | y | z | w |
|---|---|---|---|---|---|
| 1 | 05BA09540B012AAE4E6E4293DB86283DE9C236 | 0 | 8 | 0 | 9 |
| 2 | 05AC53E58504D13768AF13D674D4EEE4616236 | 10 | -10 | 2 | 3 |
| 3 | 078758B46714EE174C1AA2D49604E2E3333326 | 12 | 0 | -4 | 5 |
| 4 | 049AEA04DF8C763F1FA096A6EFADAB39CADC36 | -12 | 0 | -4 | 5 |
| 5 | 0000F72B242BAC276991F3DAA31629E9CA5A66 | 2 | 10 | -6 | 5 |
| 6 | 0182774DD71862957840D10DA126C6AEE16166 | 10 | -2 | 6 | 5 |
| 7 | 00840456979017218EFB814934AE11E3E6C356 | 8 | 4 | 8 | 3 |
| 8 | 049105089DA64A8423BF45D0230FA318F4BC36 | 4 | 8 | 8 | 3 |
| 9 | 00032F0723765FCA3E15A728B3DCB9266A8656 | 8 | 4 | -8 | 3 |
| 10 | 0680840C14F643B8E2DB970C4E19FAB50C5F26 | 0 | 0 | 8 | 7 |

5.5 <u>New Turyn Type Sequences for Length 40</u>

For $n = 40$ we can use the fact that $f_Z(\theta) + f_W(\theta) \leq 119$, but the search space is so large it might take close to 100 million CPU hours. If we let

$$f_Z(\theta) \leq 92, f_W(\theta) \leq 92, \lfloor f_Z(\theta) \rfloor + \lfloor f_W(\theta) \rfloor \leq 116, \theta \in \frac{k\pi}{100}, 1 \leq k \leq 100, k\epsilon\mathbb{Z}$$

and further restrict $(\zeta, \omega)$ pairs to $(6,9), (6,7), (10,3)$ the search space becomes more manageable. The result of this search was the identification of the first three known Turyn Type sequences of length 40. The running time was roughly 36000 CPU hours, so the approximate time to find each $TT(40)$ was about 12000 CPU hours. One is given below.

```
X = ++++--+++++-+-+-+-+----++--++-+------++-
Y = +-++-+------++++--++-+++++----+----++-+-
Z = +-+--+-++-++---+++--++--++++++-+++-+---+
W = +++-+--++----+-+++-+++--+-++++++-++-+-+
```

These sequences give three new $T(119)$ and Hadamard matrices of order 476. Each of these three Hadamard matrices are inequivalent since their 4-profiles are all different. The hex form of all three is in the table below.

TABLE IX

NEW TURYN TYPE SEQUENCES OF LENGTH 40

| Number | Hex | x | y | z | w |
|---:|---:|---:|---:|---:|---:|
| 1 | 0603E974475D3A384C3AC8B309C44C2CDCE9A726 | 0 | -2 | 6 | 9 |
| 2 | 00121C3005EC11EF8F8CAD4C5B9B87354AA6A956 | -2 | 4 | 10 | 3 |
| 3 | 070680B1858C9566513C856EFFD0BE11519F2926 | 4 | 2 | 10 | -3 |

5.6 <u>Future Work</u>

It could be conjectured that there is a $TT(n)$ for every even integer. So far, every $TT(n)$ known has been either found by hand or more recently by long computer searches. No one has yet been able to find any patterns to compute a $TT(k)$ from a $TT(n)$ with $k > n$, if they exist. The count of $TT(36)$ is likely less than the count of $TT(34)$ which in turn is likely less than the count of $TT(32)$. The count of $TT(n)$ has generally increased up to $TT(32)$ and represents a reversal of the trend. The time required to find a $TT(n)$ using the further restrictions on $f_Z(\theta), f_W(\theta)$, jumped considerably between 36 and 38 and between 38 and 40 as shown in the table below. This suggests the possibility that the count of $TT(n)$ may continue to decrease for $n = 38$ and $n = 40$, though the sample size is small.

TABLE X

TIME TO COMPUTE EACH SEQUENCE USING EFFICIENT ALGORITHM

| Length | Hours |
|---:|---:|
| 34 | 6 |
| 36 | 36 |
| 38 | 750 |
| 40 | 12000 |

Finding a $TT(44)$ would lead to the first $T(131)$ which is currently the lowest order for which no T-sequences are known. Finding a $TT(56)$ and a $TT(60)$, which would give Hadamard matrices of the lowest two unknown orders, 668 and 716, would be difficult without significant improvements to the search algorithm.

# CITED LITERATURE


1) Baumert, L., Golomb, S. W., and Hall, M., Jr., Discovery of an Hadamard Matrix of Order 92, *Bulletin of the American Mathematical Society 68*:237-238, 1962.

2) Baumert, L.D. and Hall, M. Jr. A new construction method for Hadamard matrices. *Bull. Amer. Math. Soc. 71*, 169-170,1965.

3) Best, D., Djokovic, D. Z., Kharaghani, H., and Ramp, H. Turyn-Type sequences: classification, enumeration, and construction. *J. Combin. Designs 21* 24-35; 2012.

4) Colbourn, C. J.; Dinitz, J. H. Handbook of Combinatorial Designs (2nd Edition eds.), Boca Raton: Chapman & Hall/ CRC; 2007.

5) Cooper, J., Milas, J., Wallis, W.D.: Hadamard equivalence, In: Combinatorial Mathematics, Lecture Notes in Mathematics, vol. 686, pp. 126-135. Springer-Verlag, Berlin, Heidelberg, New York, 1978.

6) Djokovic, D. Z., Hadamard Matrices of Order 764 Exist. Combinatorica 28, 487-489; 2008.

7) Djokovic, D.Z. Five new Hadamard matrices of order skew type. *Austral. J. Combinatorics 10*, 259-264, 1994.

8) Djokovic, D.Z. Two Hadamard matrices of order 956 of Goethals-Seidel type. *Combinatorica, 14*, 375-377, 1994.

9) Djokovic, D.Z., A new Yang number and consequences, *Des. Codes Cryptogr. 54*, 201-204; 2010.

10) Goethals, J.M. and Seidel, J. J. Orthogonal matrices with zero diagonal, *Can. J. Math. 19*, 1001-1010; 1967.

11) Hadamard, J. Resolution d'une question relative aux determinants. *Bull. Sci. Math. 17*, 240-246; 1893.

12) Hall, M., Jr., Hadamard matrices of order 16, *JPL Research Summary No. 36-10, i*, 21-26; 1961.

13) Hall, M., Jr., Hadamard matrices of order 20, *J.P.L. Technical Report, 32*, 761; 1965.

14) Holzmann, W. H., Kharaghani, H. and Tayfeh-Rezaie, B. Williamson Matrices up to Order 59, *Design, Codes and Cryptography 46*, 343-352; 2008.

15) Hedayat, A.S. and Wallis, W.D. Hadamard Matrices and their Applications, *The Annals of Statistics 6*,. 1184-1238; 1978.

16) Ito, N., Leon, J. S., & Longyear, J. Q. Classification of 3-(24,12,5) designs and 24-dimensional Hadamard matrices, *J. Combin. Theory Ser. A31*, 66-93; 1981.

17) Kharaghani, H. and Tayfeh-Rezaie, B. A Hadamard matrix of order 428, *J. Combin. Designs 13*, 435–440; 2005.

18) Kharaghani , H. and Tayfeh-Rezaie, B. On the classification of Hadamard matrices of order 32, *J. Combin. Designs* 18 (2010), 328-336.

19) Kharaghani , H. and Tayfeh-Rezaie, B., Hadamard matrices of order 32, *J.Combin.Designs*, to appear.

20) Kimura, H. Classification of Hadamard matrices of order 28 with Hall sets, *Disc. Math. 128*, 257-268, 1994.

21) Kounias, S. and Sotirakoglou, K. Construction of orthogonal sequences, in: Proceedings of the 14th Greek Statistical Conference, pp. 229–236, 2001, (in Greek).

22) Koukouvinos, C., Kounias, S., Seberry, J., Yang C. H., and Yang, J. On sequences with zero autocorrelation, *Des. Codes Cryptogr. 4*, 327–340; 1994.

23) Koukouvinos, C. E-mail message posted on Nov. 1, 1999 at URL: http://www.uow.edu.au/~jennie/WILLIAMSON/williamson.html

24) Miyamoto, M. A construction of Hadamard matrices. *J. Combin. Theory Ser. A* 57(1), 86108; 1991.

25) Paley, R. E. A. C. On Orthogonal Matrices. *J. Math. Phys. 12*, 311-320, 1933.

26) Sawade, K. A Hadamard matrix of order 268, *Graphs Combin. 1*, no. 2, 185-187; 1985.

27) Sylvester, J. J. "Thoughts on Orthogonal Matrices, Simultaneous Sign-Successions, and Tessellated Pavements in Two or More Colours, with Applications to Newton's Rule, Ornamental Tile-Work, and the Theory of Numbers." Phil. Mag. 34, 461-475, 1867.

28) Turyn, R. J., Hadamard matrices, Baumert-Hall units, four-symbol sequences, pulse compression, and surface wave encodings, *J. Combin. Theory Ser. A 16*, 313–333; 1974.

29) van Lint, J. H., and Wilson, R. M. A Course in Combinatorics, chapter 18, 2002.

30) Wallis, W.D. Designs 2002: Further Computational and Constructive Design Theory, Springer, New York, pp 173-176; 2003.

31) Williamson, J. "Hadamard's Determinant Theorem and the Sum of Four Squares." *Duke. Math. J. 11*, 65-81, 1944.

# APPENDIX

## CHART OF HADAMARD MATRICES

We've created a chart of  how to construct a Hadamard matrix for every known order up to 1000. Some of these constructions use new T-matrices found in this paper.

The Paley and Kronecker constructions are used in expressions involving only numbers and symbols.  Paley I is used when the prime or power of a prime is 3 (mod 4) and Paley II is used when the prime or power of a prime is 1 (mod 4).

Wxx  means use the set of Williamson matrices of order xx.

Txx  means use the set of T-matrices of order xx.

Axx  means use the set of A-matrices of order xx.

M[p-1] means use the Miyamoto construction of order p-1 to create a Hadamard of order 4p where p is the power of an odd prime.

No Hadamard matrices are currently known for orders 668, 716 and 892.

Hadamard Matrices of Orders to 400

| Order | Constr | | Order | Constr | | Order | Constr | | Order | Constr |
|---|---|---|---|---|---|---|---|---|---|---|
| 4 | $2^2$ | | 104 | 103+1 | | 204 | 2(101+1) | | 304 | 2(151+1) |
| 8 | $2^3$ | | 108 | 107+1 | | 208 | 2(103+1) | | 308 | 307+1 |
| 12 | 11+1 | | 112 | $2^2(3^3+1)$ | | 212 | 211+1 | | 312 | 311+1 |
| 16 | $2^4$ | | 116 | W29 | | 216 | 2(107+1) | | 316 | 2(157+1) |
| 20 | 19+1 | | 120 | 2(59+1) | | 220 | 2(109+1) | | 320 | $2^2(79+1)$ |
| 24 | 23+1 | | 124 | 2(61+1) | | 224 | 223+1 | | 324 | T3W27 |
| 28 | $3^3+1$ | | 128 | $2^7$ | | 228 | 227+1 | | 328 | 2(163+1) |
| 32 | $2^5$ | | 132 | 131+1 | | 232 | $W29\otimes 2$ | | 332 | 331+1 |
| 36 | 2(17+1) | | 136 | 2(67+1) | | 236 | T59 | | 336 | 2(167+1) |
| 40 | 2(19+1) | | 140 | 139+1 | | 240 | 239+1 | | 340 | $2(13^2+1)$ |
| 44 | 43+1 | | 144 | 2(71+1) | | 244 | $3^5+1$ | | 344 | $7^3+1$ |
| 48 | 47+1 | | 148 | 2(73+1) | | 248 | $2^2(61+1)$ | | 348 | 347+1 |
| 52 | $2(5^2+1)$ | | 152 | 151+1 | | 252 | 251+1 | | 352 | $2^3(43+1)$ |
| 56 | $2(3^3+1)$ | | 156 | T13W3 | | 256 | $2^8$ | | 356 | M[2(43+1)] |
| 60 | 59+1 | | 160 | 2(79+1) | | 260 | T65 | | 360 | 359+1 |
| 64 | $2^6$ | | 164 | 163+1 | | 264 | 263+1 | | 364 | 2(181+1) |
| 68 | 67+1 | | 168 | 167+1 | | 268 | A67 | | 368 | 367+1 |
| 72 | 71+1 | | 172 | W43 | | 272 | 271+1 | | 372 | T3W31 |
| 76 | 2(37+1) | | 176 | $2^2(43+1)$ | | 276 | 2(137+1) | | 376 | $T47\otimes 2$ |
| 80 | 79+1 | | 180 | 179+1 | | 280 | 2(139+1) | | 380 | 379+1 |
| 84 | 83+1 | | 184 | $W23\otimes 2$ | | 284 | 283+1 | | 384 | 383+1 |
| 88 | 2(43+1) | | 188 | T47 | | 288 | $2^2(71+1)$ | | 388 | 2(193+1) |
| 92 | W23 | | 192 | 191+1 | | 292 | A73 | | 392 | $2^2(97+1)$ |
| 96 | 2(47+1) | | 196 | 2(97+1) | | 296 | $2^2(73+1)$ | | 396 | 2(197+1) |
| 100 | $2(7^2+1)$ | | 200 | 199+1 | | 300 | 2(149+1) | | 400 | 2(199+1) |

Hadamard Matrices of Orders 404 through 700

| Order | Constr | | Order | Constr | | Order | Constr |
|---|---|---|---|---|---|---|---|
| 404 | T101 | | 504 | 503+1 | | 604 | A151 |
| 408 | $2^2$(101+1) | | 508 | A127 | | 608 | 607+1 |
| 412 | A103 | | 512 | $2^9$ | | 612 | T17W9 |
| 416 | $2^2$(103+1) | | 516 | 2(257+1) | | 616 | 2(307+1) |
| 420 | 419+1 | | 520 | T65 $\otimes$ 2 | | 620 | 619+1 |
| 424 | 2(211+1) | | 524 | 523+1 | | 624 | 2(311+1) |
| 428 | T107 | | 528 | 2(263+1) | | 628 | 2(313+1) |
| 432 | 431+1 | | 532 | T7W19 | | 632 | 631+1 |
| 436 | M[107+1] | | 536 | A67 $\otimes$ 2 | | 636 | 2(317+1) |
| 440 | 439+1 | | 540 | 2(269+1) | | 640 | $2^3$(79+1) |
| 444 | 443+1 | | 544 | 2(271+1) | | 644 | 643+1 |
| 448 | 2(223+1) | | 548 | 547+1 | | 648 | 647+1 |
| 452 | T113 | | 552 | $2^2$(137+1) | | 652 | A163 |
| 456 | 2(227+1) | | 556 | 2(277+1) | | 656 | $2^2$(163+1) |
| 460 | 2(229+1) | | 560 | $2^2$(139+1) | | 660 | 659+1 |
| 464 | 463+1 | | 564 | 563+1 | | 664 | 2(331+1) |
| 468 | 467+1 | | 568 | 2(283+1) | | 668 | |
| 472 | T59 $\otimes$ 2 | | 572 | 571+1 | | 672 | $2^2$(167+1) |
| 476 | T119 | | 576 | $2^3$(71+1) | | 676 | 2(337+1) |
| 480 | 479+1 | | 580 | 2($17^2$+1) | | 680 | $2^2$($13^2$+1) |
| 484 | 2(241+1) | | 584 | A73 $\otimes$ 2 | | 684 | 683+1 |
| 488 | 487+1 | | 588 | 587+1 | | 688 | 2($7^3$+1) |
| 492 | 491+1 | | 592 | $2^3$(73+1) | | 692 | 691+1 |
| 496 | $2^3$(61+1) | | 596 | M[2(73+1)] | | 696 | 2(347+1) |
| 500 | 499+1 | | 600 | 599+1 | | 700 | 2(349+1) |

Hadamard Matrices of Orders 704 to 1000

| Order | Constr | | Order | Constr | | Order | Constr |
|---|---|---|---|---|---|---|---|
| 704 | $2^4(43+1)$ | | 804 | 2(401+1) | | 904 | T113⊗2 |
| 708 | 2(353+1) | | 808 | T101⊗2 | | 908 | 907+1 |
| 712 | M[2(43+1)]⊗2 | | 812 | 811+1 | | 912 | 911+1 |
| 716 | | | 816 | $2^3(101+1)$ | | 916 | 2(457+1) |
| 720 | 719+1 | | 820 | 2(409+1) | | 920 | 919+1 |
| 724 | $2(19^2+1)$ | | 824 | 823+1 | | 924 | 2(461+1) |
| 728 | 727+1 | | 828 | 827+1 | | 928 | 2(463+1) |
| 732 | T3W61 | | 832 | $2^3(103+1)$ | | 932 | M[W29⊗2] |
| 736 | 2(367+1) | | 836 | T19W11 | | 936 | 2(467+1) |
| 740 | 739+1 | | 840 | 839+1 | | 940 | T47W5 |
| 744 | 743+1 | | 844 | 2(421+1) | | 944 | $T59\otimes 2^2$ |
| 748 | 2(373+1) | | 848 | $2^2(211+1)$ | | 948 | 947+1 |
| 752 | 751+1 | | 852 | T71W3 | | 952 | T119⊗2 |
| 756 | T7W27 | | 856 | T107⊗2 | | 956 | A239 |
| 760 | 2(379+1) | | 860 | 859+1 | | 960 | 2(479+1) |
| 764 | A191 | | 864 | 863+1 | | 964 | M[239+1] |
| 768 | 2(383+1) | | 868 | 2(433+1) | | 968 | 967+1 |
| 772 | M[191+1] | | 872 | M[107+1] ⊗2 | | 972 | 971+1 |
| 776 | $2^2(193+1)$ | | 876 | T73W3 | | 976 | 2(487+1) |
| 780 | 2(389+1) | | 880 | 2(439+1) | | 980 | T35W7 |
| 784 | $2^3(97+1)$ | | 884 | 883+1 | | 984 | 983+1 |
| 788 | 787+1 | | 888 | 887+1 | | 988 | T13W19 |
| 792 | $2^2(197+1)$ | | 892 | | | 992 | 991+1 |
| 796 | 2(397+1) | | 896 | $2^2(223+1)$ | | 996 | T83W3 |
| 800 | $2^2(199+1)$ | | 900 | 2(449+1) | | 1000 | 2(499+1) |

W1
A +
B +
C +
D +

W3
A +++
B +--
C +--
D +--

W5
A ++--+
B +-++-
C +----
D +----

W7
A +++--++
B +++--++
C ++-++-+
D +-+--+-

W9
A ++++--+++
B ++-+--+-+
C ++-+--+-+
D +---++---

W11
A ++++----+++
B +-+++--+++-
C +-+--++--+-
D +--+----+--

W13
A ++++-+--+-+++
B ++---+--+---+
C ++---+--+---+
D +---+-++-+---

W15
A +++++-+--+-++++
B ++-+-+----+-+-+
C ++--++-++-++--+
D +--+++----+++--

W17
A +++++---++---++++
B +++-++--++--++-++
C ++----+-++-+----+
D +-+-+--+--+--+-+-

W19
A ++++++---++---+++++
B +++---+--++--+---++
C ++---+-+-++-+-+---+
D +-+--+--+--+--+--+-

W21
A +++++--++----++--++++
B +++-+--+-++++-+--+-++
C +++--+-+++--+++-+--++
D +---+-+-++--++-+-+---

W23
A +++++-++---++---++-++++
B +-+++--++-+--+-++--+++-
C +-+--+++++----+++++--+-
D +--++-+-+------+-+-++--

W25
A ++++++---+-+--+-+---+++++
B +++-+--+++--++--+++--+-++
C ++-+--++-+-++++-+-++--+-+
D +-+++--+--++++++--+--+++-

W27
A ++++++--+-++----++-+--+++++
B ++-+---+--+++--+++--+---+-+
C ++--++-+-+-++++++-+-+-++--+
D +--+++++--+-+--+-+--+++++--

W29
A +++++++---+-++--++-+---++++++
B ++++---+++-+--++--+-+++---+++
C ++-+++---+--+-++-+--+---+++-+
D +-+-++---+--+-++-+--+---++-+-

W31
A ++++++-+--+---++++---+--+-+++++
B +--++---+-+-++----++-+-+---++--
C +--++---+-+-++----++-+-+---++--
D +-----+-++-+++----+++-++-+-----

W33
A ++++++-+-+-+++------+++-+-+-+++++
B ++-+-++-+----+++--+++----+-++-+-+
C ++--++-+++-+--+-++-+--+-+++-++--+
D +--++--+++++-++----++-+++++--++--

W37
A +++++-+-----++----++----++-----+-++++
B +--+++-+-----+----++----+-----+-+++--
C +--+-+-+-++---+--++++--+---++-+-+-+--
D +---++-++--+-+-++----++-+-+--++-++---

W39
A +++--+-+-----+--++----++--+-----+-+--++
B +++--++-+---+-+--+----+--+-+---+-++--++
C ++++---+--++----+-+--+-+----++--+---+++
D +---++-+-+-----+++-++-+++-----+-+-++---

W41
A ++++--+-++++-++--++----++--++-++++-+--+++
B ++++--+-++++-++--++----++--++-++++-+--+++
C +++-++-+-+-+-----+++--+++-----+-+-+-++-++
D +--+--+-+-+-+++++---++---+++++-+-+-+--+--

W43
A +++++-++----+-++--++-++-++--++-+----++-++++
B +++---++--+-+-+-++--------++-+-+-+--++---++
C ++-+-++++-+--+--+++--++--+++--+--+-++++-+-+
D +-++-----++++-+-+++-++++-+++-+-++++-----++-

W45
A +++---+--++-+++++-+-+-++-+-+-+++++-++--+---++
B -++---+--++-+++++-+-+-++-+-+-+++++-++--+---++
C ++--+-+--++----+++++-++++-+++++----++--+-+--+
D ++--+-+--++----+++++-++++-+++++----++--+-+--+

W49
A ++++-++-+---++-+++---++-++-++---+++-++---+-++-+++
B ++++-++-+---++-+++---++-++-++---+++-++---+-++-+++
C +----+-++++--+-+++-+-+++--+++-+-+++-+--++++-+----
D +++++-+----++-+---+-+---++---+-+---+-++----+-++++

W51
A +---+++-++-+-+++--+++++--++--+++++--+++-+-++-+++---
B ----+++-++-+-+++--+++++--++--+++++--+++-+-++-+++---
C -+--+----+-+++-+-+++++--+--+--+++++-+-+++-+----+--+
D -+--+----+-+++-+-+++++--+--+--+++++-+-+++-+----+--+

W55
A +-+--+-+-++--+-+++++-+++--++++--+++-+++++-+--++-+-+--+-
B --+--+-+-++--+-+++++-+++--++++--+++-+++++-+--++-+-+--+-
C +++----++-++--++----+-+-++++++++-+-+----++--++-++----++
D +++----++-++--++----+-+-++++++++-+-+----++--++-++----++

```
W57
A +---++-+--++++-+++-++---+-++++++-+---++-+++-++++--+-++---
B ----++-+--++++-+++-++---+-++++++-+---++-+++-++++--+-++---
C --+-+-+++--+--+-++---+++++-++++-+++++---++-+--+--+++-+-+-
D --+-+-+++--+--+-++---+++++-++++-+++++---++-+--+--+++-+-+-

W61 [23]
A ++--+--++--+-+-++++--+-----+------+-----+--++++-+-+--++--+--+
B ++--+--++--+-+-++++--+-----+------+-----+--++++-+-+--++--+--+
C +---+-+-++++---++--+-++-+---++++++---+-++-+--++---++++-+-+---
D ++++-+-+----+++--++-+--+-+++------+++-+--+-++--+++----+-+-+++

W63
A++-+++--++-++--+--+-++-+-+++--------+++-+-++-+--+--++-++--+++-+
B-+-+++--++-++--+--+-++-+-+++--------+++-+-++-+--+--++-++--+++-+
C++++-++-+-++++-+---+---+++---++++++---+++---+---+-++++-+-++-+++
D++++-++-+-++++-+---+---+++---++++++---+++---+---+-++++-+-++-+++

T1 +
T2 0
T3 0
T4 0

T3
T1 +00
T2 0+0
T3 00+
T4 000

T5
T1 ++000
T2 00+00
T3 000+-
T4 00000

T7
T1 ++-0000
T2 000+0+0
T3 0000+00
T4 000000+

T11
T1 +--+0000000
T2 0000+++0000
T3 0000000+0+-
T4 00000000+00
```

T13
T1 +0+0+00000000
T2 0+0+0-+000000
T3 0000000+0000-
T4 00000000++--0

T17
T1 +++-++00000000000
T2 000000++-+-000000
T3 00000000000++0-0-
T4 0000000000000-0+0

T19
T1 ++0+00+0-0000000000
T2 00+0++0-0+000000000
T3 0000000000++00-00-+
T4 000000000000+-0-+00

T23
T1 ++-+--++000000000000000
T2 00000000++---+-00000000
T3 000000000000000++0+0+0-
T4 00000000000000000+0+0+0

T25
T1 +00++0+0+-000000000000000
T2 0++00+0+00--+000000000000
T3 0000000000000++---00+-++-
T4 000000000000000000+-00000

T29
T1 +++---+--+0000000000000000000
T2 0000000000++-+-+++-0000000000
T3 0000000000000000000+00+0+0++-
T4 00000000000000000000++0+0-000

T35
T1 ++-+--+---++00000000000000000000000
T2 000000000000+++---+-++-000000000000
T3 00000000000000000000000++0+++000+0-
T4 0000000000000000000000000+000+-+0+0

T41
T1 +-++--+++----+000000000000000000000000000
T2 00000000000000++++-+---++-+00000000000000
T3 000000000000000000000000000+0++00+0+-00+-
T4 0000000000000000000000000000+00++0+00+-00

T47
T1 +-+-----+++-+--+0000000000000000000000000000000
T2 0000000000000000++++--++---++-+0000000000000000
T3 0000000000000000000000000000000+0+00+++000+-0+-
T4 00000000000000000000000000000000+0++000+-+00-00

T53
T1 +-++-+---+--+++--+00000000000000000000000000000000000
T2 000000000000000000++-+-+++-----++-+000000000000000000
T3 00000000000000000000000000000000000+0+0+0000-+++0-0+-
T4 000000000000000000000000000000000000+0+0++++0000+0-00

T59
T1 +++-----++--+-++-+-+000000000000000000000000000000000000000
T2 00000000000000000000+++---+---++--+--+-00000000000000000000
T3 000000000000000000000000000000000000000+0+0++000+0-+-00+0+-
T4 0000000000000000000000000000000000000000+0+00+++0+000--0+00

T65
T1 +-----++-+-+---+-----+0000000000000000000000000000000000000000000
T2 0000000000000000000000+++-++-++---++----+-+0000000000000000000000
T3 0000000000000000000000000000000000000000000+000-+000++00---00-++-
T4 00000000000000000000000000000000000000000000++-00+-+00+-000--0000

T71
T1 =
++-++-++-----+++-++---++0000000000000000000000000000000000000000000000
0
T2 =
000000000000000000000000++--++++-++++++--+-+-+-00000000000000000000000
0
T3 =
00000000000000000000000000000000000000000000000++0++++--000-+-000000+0
-
T4 =
0000000000000000000000000000000000000000000000000+000000+-+000+-+--+0+
0

T73 [9]
T1
+0+0-0+0-0-0+0-0+0-0+0+0-0-0+0+0-0-00000000000000000000000000000000000
000
T2
0-0+0+0+0+0+0-0+0-0+0+0+0-0-0+0-0-0-+000000000000000000000000000000000
000
T3
0000000000000000000000000000000000000++0+00000--0++0+0+-0+0-+0++00000+
0-+
T4
000000000000000000000000000000000000000-0-+---00-00+0+00+0+00+00+---+0
+00

T83
T1 =
++++++++--+--++-+---++---+-+000000000000000000000000000000000000000000
0000000000000
T2 =
0000000000000000000000000000++++--++-+++---+-++++-++-+-000000000000000
0000000000000
T3 =
0000000000000000000000000000000000000000000000000000000+0+-00+0++00-0+
0--00+0-+00+-
T4 =
00000000000000000000000000000000000000000000000000000000+00-+0+00-+0-0
-00++0+00+-00

T101
T1
+++-+---+----+++++--+-++-+-+--+--+000000000000000000000000000000000000
0000000000000000000000000000000
T2
0000000000000000000000000000000000+++-++---+++--+---+-+----++-++-++000
0000000000000000000000000000000
T3
0000000000000000000000000000000000000000000000000000000000000000000+++
-+-+++0+0+000-0-++0-0+00000000-
T4
0000000000000000000000000000000000000000000000000000000000000000000000
000000-0-0---0+000-0-0---++--+0

T107
T1
+++-+-+--+++-+-++++--+-+++-----+--++0000000000000000000000000000000000
0000000000000000000000000000000000000
T2
000000000000000000000000000000000000+++++---++---+---+--+---+-++-++-+-
-000000000000000000000000000000000000
T3
0000000000000000000000000000000000000000000000000000000000000000000000
0++00000+--0-0+0-00+-0+0-0-000+-+++0-
T4
0000000000000000000000000000000000000000000000000000000000000000000000
000+++-+000-0+0+0++00+0+0+0+--00000+0

T113
T1
+-++++++++--+--+--+-+-+++-++---++++--+00000000000000000000000000000000
0000000000000000000000000000000000000000000
T2
00000000000000000000000000000000000000++++++++-+-++--+++----++++---+--
++--+00000000000000000000000000000000000000
T3
0000000000000000000000000000000000000000000000000000000000000000000000
00000+00+00+-+0-00+00-+-000+-0-+0-000+-0-+-
T4
0000000000000000000000000000000000000000000000000000000000000000000000
000000+-0-+000+0++0--000--+00+00-0--+00+000

T119
T1
+-+--+-++-++---+++--++--++++++-+++-+---+000000000000000000000000000000
0000000000000000000000000000000000000000000000000
T2
0000000000000000000000000000000000000000+++-+--++----+-+++-+++--+-++++
++-++-+-+0000000000000000000000000000000000000000
T3
0000000000000000000000000000000000000000000000000000000000000000000000
000000000+0++-000000-+0+00-+0-00++0-00-+----000+-
T4
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000+000-+++++00-0-+00-0--00-0++000000--+00

A67
A1 -++++++++++--+-----+++-----++-+---+-+--+---+----+-+------+-+-----+-
A2 -+-++++-+-++--+-+-+-++++----++-+---+-+---+---+--+-+-+-+--+------+++--
A3 -+--++---+-+--++-+-+---+++++-+---++-+--------++++-++-++-----++--++-
A4 -++----+-++-++-+--++--++---+-+--++++-+----+--+++---++----++-++-+++-

A103 [7]
A1 ---+++-++--+++++++-+--+-++-+-+-++-+++-+-++---+----++++++++----+-+-
--+--+-+-+-+--+-++-+------++--+----++
A2 --+---+++++--+++++-+++-+-+++++-+--+-+--++--+++--+-+-+-------+++-++--
+--+-+-----+-+---++---++++-+---++-+-----
A3 -+++---+-+++-++--+-+++-+-++-+----++---++-++-++++++++--++++---+-+-++
--------++++-+---+---+--+++++++-++----++
A4 -++++----+-++-+--++++++---++-----+--++-+++---+++-++-++-++-++--++-++++
+++-------+---+-+++-+-+++-+++-+++----+++

A127
A1 -++-+---++-+-+-+++++-++--++---+++-+++++--++-+---+-++-+-------+++-+--
-+++++++-+--+-++-+--++-----+---+++--++--+-----+-+-+--+++-+--
A2 -++-+--++--+-++++--+--+--++++-+++--+--+-----+----+++-+++--++-+++--
+-+++-+--++-+-+------++-+-+-+-++++-++++---++++++-+--++++-++++
A3 -++-++--+++----++++-++-----+--+++++-++--+++----+---+--++----++++++-+
-+--++-+--+-++-+-+--+--+-+-+-+++----+--+---+++++-+-+-+++++++++
A4 -----+-+--++--+----+++++-+-++----+---++-+++-+-++-++--++++-+-+-+---+
+-+---++++++--+-+-+--+++-+++++-++-+++----+++++++-+--+++--++-++

A151 [7]
A1 -+--++--+-++------+++-+-+-----+-+++++++++-++-----+-+------+-+--++-+--+--
-+++--++-+--++-+---+++-++-+--++-+-++++++-+++++-+---+----+-+++++-+-+----++
+++--+-++--++-
A2 --+-+-------+---++-------+----+++-++++-+---++++----++-++----+--+++---
-+-+--+--+--+-+--+++-++-+-+----+-+++--++++++---+---++-+++-+-+-+++++--++-
---+-+-------+-
A3 -+++++++++------+-++++--++-++--+-++-+-++-++++----+-+---++---++-+++++--+++
+-++-+--++-+---++-+-+-+--+--+---+-+++-+++-+++-+++--++-+++-+--+++-+--++-+-
------++++++-++
A4 ---+--+--+++++-------+-+-+++-+++-----++-+-+--++-++++-++-+-++--------
-+++-+++--++++++-+-++++++++-+-+-++++-++-+++++---+-++----++++-+----++----
+++--++-+--+-+

A163
A1 -++++++++++-+-++-++---++--+-++--+-++--++-+--+-+--++---+--+-+-------+-
--+-+-+-
-++--+----++---+-+-+---++--+++-+------++-++-+++-----+-----+-+--+-+++--
-+--+-++--
+-+-++--
A2 -+++++-++++++++-++---------+++--+-+++-++-++++++--+++--++-+-+----+----
--+-+-++
-+---+----++---+++-+-+-++--+++-++-----++-++-+++-----+-++--+----+-+++--
--+-+-++-+
--+-++--
A3 -+++--++-+-+-----+-+++++++++---++-+++----+-+-----+-----+++--+-++----+-++
-+-+---+-
++++++-+-++--+----+-++-+--+--++++-+-+-+++-++-+++-+++-----+-+-+----++----
++-++-+---
+---+--+
A4 -+++--++++++++---++--++--++-+---+++++-+-+++-++-----+--++-+-+--++++++
-+----++
+++-++----++-----++-+-+-+----++-++-----++++++-+++-----+--++-++-++-++++++--
-++-+++--+
+-+++---

```
A191 [6]
A1 -------++-+--+-----+--------++-+++-+--+-+++---+-+-+-----+-+-+++++--
---++-+-++---+-+++--+++--+--++--++-+--+--+-+----++---++--+++----++--+-
+-+-+++-++-+-++++-+--++-+---+-++----+--+--+-+++-+++--+
A2 ---+---+-+-+-----+++------+-+--+---+-----++----+----++--++++--++-++
-++++++--+-----++-+--+-+++-+++-+-----------+++-++-++++-+---++++--+-+-+
+-++-+----++-+-+-+--+--++---+++++++-++++-+--+++-+++--+
A3 -+-+--+-----+-++--+--+--------++--++-+-++-+++-+-+++-+--+----+++--++
-+----+++-----+--++++++-+++-++++-+-+-++-+-+++-+-+-++-+---+-++-+------+
+-+-+----++++++-+++--++--+---+-+-----+-+-++----++++++-
A4 ---+--++---+-++-+-+-+++-----------+---++-+--++++++-++++-+++--+-+-+++
++-+--+++---+++++-+-----+-+-+--+---+-++++---++++-++++--++-+++-+++----+
--++-++-+-+-+--+++--++-----+-+--++-+++-+-+--+-+----+++

A239 [8]
A1 -++++---+-+--+--+-+-+---+-----+--+++--+++--++++-+-----++---+-+++---
++---++-++-+++--++-+++--+--+-+-++++--++++++-++++--------+++-+--+-+-+--
--+-+++-+--+--++++-----++-++-++------+-+-+-+---+------+++-++-+++++---+
--+-+---+---+-+-+-++--+-----++++
A2 -+-----+--+--+++---+-+--+-+-+------+++++-+++++-+----++----+----++++
-+++--+--++++----+++-------+--++-+-+-+++-+--+---++--+-+-++-+-+-++-+-+-
+++++-+--++--+++-+++-++--+-----+--++--+-+---++--+++++++--++-+++-++-++-
++-+++--+-+-+--++---+-+-+--++-++
A3 --++-++----++--++++++-++-+-+-++++++-+--+--++----+++++++++---+++---+
----++-++++---++--+--+--+--+---+-+-+---+-++---+-++-++++--+-+++--++++-
+----+-++-+-+++----+---+--+--++-++----+--+++++-+----+++-+--+---+---+--
--+---++--+++-----++++-+-+++-++-
A4 -++-++----+-++-+++--+++++-++-+-++---+-+++-+++-++++++++--+-++---++++
----+-++-++++-++---+++----++++-+-+-++-+---+--+----+--++----+--++---+--
+-+-+-+-++++++++--+-++++++++-+--++-+--++--+-++---+++--+---+++-+++----+
+--+--+++--++++-+--+-++++++-+-++
```

# VITA

| | |
|---|---|
| NAME: | Stephen London |
| EDUCATION: | B.S., Mathematics, North Carolina State University, Raleigh, North Carolina, 1995 |
| | M.S., Mathematics, University of Illinois at Chicago, Chicago, Illinois, 2006 |
| | Ph.D., Mathematics, University of Illinois at Chicago, Chicago, Illinois, 2013 |
| TEACHING: | Department of Mathematics, University of Illinois at Chicago: Instructor: Finite Mathematics, Calculus III, Mathematical Reasoning, 2007, 2011-2012<br>Summer Enrichment Workshop Instructor: Algebra, 2007-2010<br>Teaching Assistant: Algebra, Precalculus, Calculus, Business Mathematics, Finite Mathematics, Advanced Mathematics for Business, Introduction to Programming, 2004-2010 |
| | Department of Mathematics, Dominican University, River Forest, Illinois: Adjunct Instructor: Beginning Algebra, Contemporary Mathematics, 2010-2011 |
| HONORS: | The UIC graduate student teaching award, Honorable Mention, University of Illinois, Chicago, Illinois, 2008 |
| | 73rd in Nation in William Lowell Putnam Examination, 1993 |
| PRESENTATIONS: | Presented Hadamard Matrices at Workshop in Graph Theory and Combinatorics in Memory of Uri Peled, Chicago, Illinois, 2010 |
| | Presented latest research on Hadamard Matrices at the American Mathematical Society Sectional Meeting, Chicago, Illinois, 2007 |