

A Path To Hadamard Matrices

P. Embury and A. Rao

School of Mathematics and Geospatial Sciences, RMIT University, Melbourne,
Australia,

s7603100@student.rmit.edu.au, asha@rmit.edu.au

Abstract. There are characteristics of Hadamard matrices that enable an exhaustive search using algorithmic techniques. The search derives primarily from the eigenvalues which are constant after the Hadamard matrix is multiplied by its transpose. Generally this would be a performance concern but there are additional properties that enable the eigenvalues to be predicted. Here an algorithm is given to obtain a Hadamard matrix from a matrix of 1s using optimisation techniques on a row-by-row basis.

Key words: Hadamard Matrices, eigen values, optimization

1 Introduction

Hadamard Matrices are named after Jacques Hadamard (1865-1963) and are square matrices of order $1, 2, 4n, n = 1 \dots \infty$, whose entries are on the unit circle, and whose rows are all orthogonal to each other. That is,

$$H_n H_n^T = nI \quad (1)$$

where n is the order of matrix.

Hadamard Matrices currently are found using construction techniques which have been generally classified [3] as recursion theorems, "plug-in" methods or direct constructions. These are summarised in the following table:

Multiplicative or Recursive Techniques	Any Kronecker product of existing Hadamard matrices is itself a Hadamard matrix.
"Plug-in" methods	Named after James Sylvester, this is the earliest (1863) and simplest construction and is also based on the Kronecker product.
Direct Constructions	The Paley HMs are two constructions found using quadratic residues in a finite field.
Williamson	Also a "plug-in" algorithm where the HM is constructed from an existing HM plus 3 other HMs that satisfy specific conditions.

Full details of the construction methods mentioned above can be found in [3] while other methods can be found in [8] and are not included here. The important

aspect is that using any of the methods above, not all orders of Hadamard matrices have been verified that they can be created since they are constructed from matrices of lower orders. This is the long (since 1893) unsolved Hadamard conjecture. In addition, the nature of equivalence is not fully understood and only orders up to 28 have been fully explored, which has a known 427 equivalence classes.

Hadamard matrices are formed into equivalence classes where they are deemed to be equivalent if the following operations are performed on them to arrive at the same matrix:

1. Any permutation of the existing rows and/or permutation of the columns
2. The multiplication of row(s) or column(s) by -1.

More recently [6] has defined Q-classes to enable more manageable categorisation of the equivalence classes by “extracting” common sub-matrices or substructures of Hadamard matrices and classifying those matrices with these common sub-matrices as Q-equivalent.

2 Background

Instead of relying on construction techniques to construct, and sheer brute force to test for, Hadamard matrices, there are characteristics that enable Hadamard matrices be searched for using optimisation methods [4, 7]. These characteristics also provide clues to the nature of Hadamard equivalence classes.

There are two properties of matrices that offer the opportunity to search for the existence of Hadamard matrices using optimisation techniques: the determinant and the eigenvectors and eigenvalues. These are, of course related measures.

The determinant for a Hadamard matrix is given by

$$\det H = \pm n^{n/2}. \quad (2)$$

But one of the drawbacks in using the determinant is that it has sharp peaks and exhibits properties analogous to the Hamming cliffs encountered when using optimisation methods with binary strings as distinct from gray encoded strings. This primarily arises from the fact that the determinant is 0 when it becomes *singular* which can occur as soon as one row becomes identical to another. Also, there are multiple matrices that have the same determinant with little or no indication as to the direction a search could or should progress.

The eigenvectors of a Hadamard matrix all lie on the unit circle and the absolute value of the eigenvalues are constant and equal to n , where n is the order of the Hadamard matrix. Since the eigenvectors can be complex and half the eigenvalues are negative, it is easier to consider the RHS of (1) and the effects on it when looking at the optimisation path.

The question that arises is: what is the sensitivity of the eigenvalues to errors that would enable its use in optimisation?

It will be shown that the variance of the eigenvalues (EVV) of the RHS of (1) can provide a suitable optimisation path since:

1. The intra-row errors describe an EVV path that is represented by the parabolic equation:

$$8x \left(1 - \frac{x}{N}\right) \tag{3}$$

where x is the number of cells within a row that are inverted and N is the order of the matrix. The variance of the eigenvalues is calculated as:

$$EVV = \sum_{i=1}^n \left(\frac{(\lambda_i - E\lambda)^2}{n} \right) \tag{4}$$

Note: $E\lambda = n$

2. The inter-row errors are described by the recursive equation:

$$EVV_x = \frac{y(y-1)}{2} M \tag{5}$$

where y is the number of non-orthogonal rows and M is the maximum **EVV** for each row which occurs where $x = N/2$ in (3). Eg, for order 8, $M = 16$ and EVV for an entire order 8 matrix of ones = 448.

This information can be used to devise an optimisation path to a Hadamard matrix from a unit matrix since at any stage it is known how many steps a non-Hadamard matrix is from a Hadamard matrix.

It is inescapable that any algorithm that actually uses eigenvalues is not going to be very efficient. Hence there needs to be an improvement to any algorithm that does use the eigenvalues method that enables the eigenvalues to be predicted. Fortunately, for row-by-row optimisation, there are two criteria or tests that can be used to assist. This will be developed in the next section.

3 The Search For A Path

Consider the normalised Hadamard matrix H_4 of order 4 given by

$$H_4 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

and the Hadamard matrix of order 8 H_8 given by:

$$\begin{bmatrix} H_4 & H_4 \\ H_4 & -H_4 \end{bmatrix}$$

As successive errors are introduced to a row, the eigenvalues (of the RHS of (1)) are affected according to the following tables and the variance follows an inverted parabolic path.

Number of cells inverted	0	1	2	3	4
Eigenvalues	4	0.5359	0	0.5359	4
	4	4	4	4	4
	4	4	4	4	4
	4	7.4641	8	7.4641	4
Variance (EVV)	0	6	8	6	0

Number of cells inverted	0	1	2	3	4	5	6	7	8
Eigenvalues	8	2.7085	1.0718	.2540	0	.2540	1.0718	2.7085	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8
Variance (EVV)	0	7	12	15	16	15	12	7	0

By the time all the entries in a row are inverted we have arrived at another (but equivalent) Hadamard matrix. Not only do the eigenvalues follow a predictable path, there is an indication of the number of rows that are not orthogonal to each other.

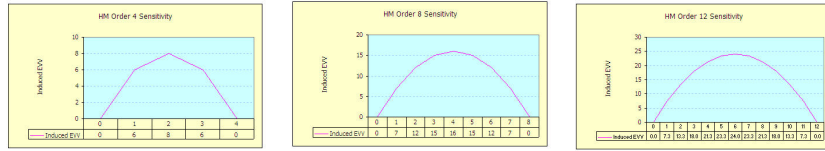


Fig. 1. Orders 4, 8 and 12 EVV as successive cells are inverted

What if there are errors in more than one row? In a simple situation where there is a single error in another row, and the particular column entries of the two rows have the same sign, then the errors aggregate, not only in that column, but also in other columns that have the same sign. When the column entries have opposite signs, they tend to compensate. (Figure 2)

But it is not so simple and with multiple errors, the situation needs to be viewed of sub-matrices of order 2 that include the respective rows. The EVV induced by errors on odd-weighted sub-matrices is double that of even weighted sub-matrices.

What if we overwrite two rows with ones within a Hadamard matrix (besides the first row)? Three rows, four rows....? What is the total EVV that can be found for a matrix consisting only of +1 in a matrix of a given order? The results are in the following table for those of order 8.

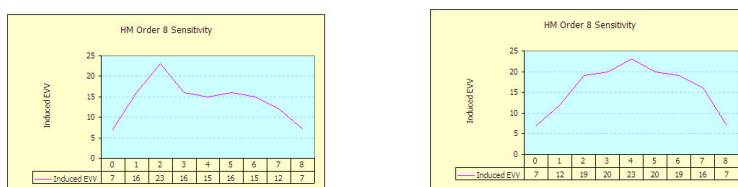


Fig. 2. Order 8 with same sign and opposite sign in another row as successive cells are inverted

Rows Non-orthogonal	EVV
1	0
2	16
3	48
4	96
5	160
6	240
7	336
8	448

Fig. 3. Successive EVV as more rows non-orthogonal

The total EVV of a square matrix of ones, (eg for order 8 is 448) which is the furthest a matrix can digress from Hadamard, can be broken down and it is possible to determine how many rows are not orthogonal to each other. In other words, every Hadamard matrix of any order must comply with (5) since replacing successive rows will automatically induce a known level of EVV until it reaches a maximum.

Putting the aforementioned characteristics into practice can be investigated in two separate ways.

1. The first method is to consider the matrix as a whole and successively test individual cells as to the effect on the EVV. In other words, each cell has a marginal contribution to the total EVV of the matrix. At each iteration, the entire matrix can be searched for the cell that when inverted, gives the greatest reduction in EVV, and then invert that cell. This will not always give a complete path to a Hadamard matrix since there are other criteria elaborated below that also need to be satisfied. Also, given that the eigenvalues are continually being calculated, this method is only suitable for matrices of smaller orders.
2. A second method is to optimise to a Hadamard matrix from a matrix of ones row-by-row. Exactly the opposite to successively substituting rows of a Hadamard matrix by rows of ones. The objective is to find which combination

$$\begin{bmatrix} . & X & X & . & . & . & X & X \\ . & X & . & X & X & . & X & . \\ . & X & . & X & X & . & . & X \\ . & X & . & X & . & X & X & . \\ . & . & X & X & X & . & X & . \\ . & . & X & X & X & . & . & X \\ . & . & X & X & . & X & X & . \\ . & . & X & X & . & X & . & X \end{bmatrix}$$

Fig. 5. Possibilities for row 4

columns that match the -1s already in rows 2 and 3. Only one of these does not include column 4 which stands to reason since there is only one way of matching up exactly two columns with the existing rows 2 & 3. (Also note that this row is the difference between rows 2 and 3. This characteristic may offer a method of taking shortcuts in the algorithm or possibly lead to clues about the nature of equivalence.)

Now compare these rows with rows in figure 6 which will be rejected because they do not have the desired EVV (These are only 6 out of the possible 26.)

Rows	EVV	Row 2 Corr.	Row 3 Corr.
. X X X . . X .	164	1	2
. . . X X X . X	164	2	3
. X X X X . . .	168	1	3
. . X . X X . X	168	3	3
. X X . X X . .	176	4	0
. . . . X X X X	176	0	4

Fig. 6. Rejected Rows

The EVV column displays the resultant EVV of the matrix should that row be used as the new row 4. The last two columns show the difference between the candidate new row and the pre-existing rows 2 and 3 respectively. The last two candidate rows are the same as the rows already inserted and hence should be immediately rejected.

The important thing to note is that all the rows that result in the desired EVV of 160, have exactly $\binom{n}{4}$ matches between itself and all previous rows (not including the first row). In other words, for every possible column permutation for the next particular row, if there aren't $\binom{n}{4}$ matches then the combination can be rejected immediately.

The other noticeable facet concerns the number of all the Xs (see the bottom row of figure 4) or weights of the columns. Each successful combination or candidate row satisfies:

$$\sum Weights = (row - 2) \binom{n}{4} \tag{6}$$

This is another rule that can be used to fine-tune the algorithm. For example, the first successful candidate row tells us to insert Xs in columns 2, 3, 7 and 8. These four columns have weights (from figure (4)) of

$$1 + 1 + 1 + 1 = 4 = (4 - 2) \binom{8}{4} \tag{7}$$

If we assume that the second candidate row is selected (from figure 5) for row 4, what are the circumstances for the next iteration?

Figure 7 has the starting position and figure 8 has the new possible rows for row 5 (there are only 4 possibilities).

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & X & X & X & X \\ \cdot & X & X & \cdot & X & X & \cdot & \cdot \\ \cdot & X & \cdot & X & X & \cdot & X & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & & \vdots & & & \\ \hline 0 & 2 & 1 & 1 & 3 & 2 & 2 & 1 \end{bmatrix}$$

Fig. 7. 4 Completed Rows and new column weights

Rows	EVV	Row 2 Corr.	Row 3 Corr.	Row 4 Corr.
. X X . . . X X	96	2	2	2
. X . X . X . X	96	2	2	2
. . X X X . . X	96	2	2	2
. . X X . X X .	96	2	2	2

Fig. 8. Candidate Rows for Row 5

All these rows have:

- the new desired EVV of 96
- exactly 2 matches with every row preceding it except row 1
- the sum of column weights = 6 which satisfies (5)

In summary, there are 3 criteria to be satisfied for each row:

1. The desired EVV for each successive row needs to satisfy $EVV = \frac{x(x-1)}{2}M$ where M is the maximum **EVV** for each row which occurs where $x = N/2$.
2. Each new row must have $\binom{n}{4}$ matches of X with every preceding row
3. The weights of each column must satisfy $\sum Weights = (row - 2) \binom{n}{4}$

Criteria 1 & 2 are synonymous while criterion 3 forms a type of a classical subset-sum problem [2] where the count of the weights subset is known and there are possible multiple solutions. The solutions are not necessarily known to exist.

3.1 An Incomplete Path

This is an example of when these methods will not allow us to proceed on our search for a Hadamard matrix. The following situation arises at the start of optimising to an *order 12* matrix. After 3 rows we may have the following order 12 matrix:

$$\begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & X & X & X & X & X & X & X & \dots & \dots \\ \dots & X & X & X & \dots & \dots & \dots & \dots & X & X & X & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \vdots & & & & & & & & & & & \end{bmatrix}$$

The following rows are all possible for the fourth row (there are others)...

$$\begin{aligned} & [\dots X X X \dots X X X \dots] \\ & [\dots X X \dots X \dots X X \dots] \\ & [\dots X X \dots X \dots X \dots X \dots] \\ & [\dots X X \dots X \dots X \dots X X \dots] \end{aligned}$$

If the first row was selected then the situation could arise where we have...

$$\begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & X & X & X & X & X & X & X & \dots & \dots \\ \dots & X & X & X & \dots & \dots & \dots & \dots & X & X & X & \dots \\ \dots & X & X & X & \dots & X & X & X & \dots & \dots & \dots & \dots \\ \vdots & & & & & & & & & & & \end{bmatrix}$$

This is a situation which could occur but the problem is there is no further step. The EVV given by (5) is an expected 864 which satisfies criterion 1. Criterion 2 is satisfied since each row matches exactly 3 times against each previous row. However, criterion 3 fails, since the weights all = 2, and the required weights to proceed needs 6 columns to add up to 9 which is an impossibility.

In other words, an additional constraint is needed to ensure that the subsequent steps are valid which could lead to a large recursive tree for matrices of larger orders. This particular problem is easily circumvented by not choosing this particular row for row 4. This is the only exception encountered so far from other simulations on orders 8, 12, 16, 20 and 24 by randomly selecting any of the valid available rows.

4 Conclusions

The methods presented in this paper are only the start of possible alternatives using the eigenvalues and eigenvectors to search for Hadamard matrices. Can they be investigated further?

The Eigenvectors How are they affected by disturbing a Hadamard matrix?

Is there some clue in the direction they shift to that may help find the “way home” and may improve the algorithm?

Equivalence At what stage of the traversal of the optimisation path is it decided whether a matrix belongs to another equivalence class or forms its own equivalence class?

The Determinant Although this was discarded as a measure on its own, this may not necessarily be desirable because it is hard to discuss eigenvalues without acknowledging the determinant because of their direct relationship. Given that the calculation of the determinant is much more efficient to calculate for larger matrices, can it be used?

Although, the optimisation method is discussed as a “Hill-climb”, the final solution forms a series of “Subset-sum” optimisation problems [2]. If these become too unwieldy, maybe alternatives as such “genetic algorithms” [1, 5] could be used.

The one example found that led to an incomplete path described above is interesting because it may offer clues as to the existence of Hadamard matrices (or the non-existence).

References

1. D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. 1st. Addison-Wesley Longman Publishing Co., 1989.
2. M. T. Goodrich and R. Tamassia *Algorithm Design: Foundations, Analysis, and Internet examples*. John Wiley and Sons, New York, 2002.
3. K. J. Horadam, *Hadamard Matrices and Their Applications*. Princeton University Press, Princeton, N.J, 2007.
4. Richard E. Klima and Neil P. Sigmon and Ernest L. Stitzinger *Applications of Abstract Algebra with MapleTM and Matlab[®]* (2nd edition). Chapman & Hall/CRC, Boca Raton, 2006.
5. Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs* (2nd, Extended Ed.). Springer-Verlag New York, 1994.
6. W. P. Orrick, *Switching operations for Hadamard matrices*. <http://www.arxiv.org/abs/math.CO/0507515>, As at 20-Jul-2007.
7. J. A. Snyman, *Practical Mathematical Optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms*. Springer, Cambridge, Massachusetts, 2005.
8. W. D. Wallis and Anne Penfold Street and Jennifer Seberry Wallis, *Combinatorics: Room Squares, Sum-Free Sets, Hadamard Matrices*. Springer-Verlag New York, 1972.