Int.J.Curr.Microbiol.App.Sci (2019) *8*(7): 2255-2261

EXCELLENT PUBLISHERS

International Journal of Current Microbiology and Applied Sciences
***ISSN: 2319-7706* Volume 8 Number 07 (2019)**
Journal homepage: http://www.ijcmas.com

Original Research Article https://doi.org/10.20546/ijcmas.2019.807.275

Construction of Hadamard Matrices in R

B.B. Revanasiddesha[1*], A. Dhandapani[2], N.K. Choure[3] and M.L. Lakhera[1]

[1]Department of Agricultural Statistics, Indira Gandhi Krishi Vishwavidyalaya, Raipur, Chhattisgarh, India
[2]ICAR-National Academy of Agricultural Research Management, Hyderabad, Telangana, India
[3]Barrister Thakur Chhedilal, College of Agriculture & Research Station, IGKV, Bilaspur, Chhattisgarh, India

**Corresponding author*

Keywords

Hadamard matrices, R functions, Function logic, Hadamard matrices

Article Info

Accepted:
17 June 2019
Available Online:
10 July 2019

A B S T R A C T

Hadamard Matrices are useful in many diverge fields such as Telecommunications, Statistics, etc. In this article, we discuss R functions developed for construction for Hadamard matrices. The construction methods implemented are Williamson (1944), Baumart and Hall (1965) and Goethals and Seidel (1967). These methods use base sequences, Turyn sequences and T-Sequences as initial rows. For each of these methods, a brief explanation of the method, function logic and examples of Hadamard Matrices generated using these methods are given.

Introduction

An Hadamard matrix of order n is a square matrix H with entries ± 1 and satisfies $HH' = H'H = nI_n$ where H'denotes the transpose of H and I_n is the identity matrix of order nThe order of H is necessarily 1, 2, or is divisible by 4. It is conjectured that Hadamard matrices of order n always exist when n is divisible by 4.

Hadamard matrices are useful in different subjects ranging from communications, cryptography. In statistics, Hadamard matrices are used to construct Balanced Incomplete Block Designs, Orthogonal Arrays of Strength 2, and Self-Weighing Designs etc. Hadamard matrices are also useful to construct balanced half-samples to estimate variance of non-linear functions from Complex Survey data. Excellent review articles such as Hedayat and Wallis (1978), Seberry (2007) and Sebrry and Yamada (1992) provide information about construction methods, researchable issues and applications of Hadamard Matrices. A list of

Hadamard matrices is available in Slone (2019) and Gupta *et al.,* (2007). The purpose of this article is to show implementation of construction methods of Hadamard Matrices in R. This would help in those who wish to use Hadamard matrices in their computational needs. The construction methods implemented are explained in section 2 and the functions developed are discussed in section 3. The conclusion section provides directions for future work.

Construction of Hadamard matrices

Several construction methods of Hadamard Matrices are available (Hall, 1988). The construction matrices implemented in this study are related to the construction method originally proposed by Williamson (1944, 1947).

Williamson method

Consider the array H

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

If A, B, C and D are some numbers, then HH' = $I_4 \otimes (A^2+B^2+C^2+D^2)$. The array H would be an Hadamard matrix of order 4 if $A^2+B^2+C^2+D^2 = 4$. The same array can be used by choosing A,B,C and D as matrices of same order. If A, B, C and D are matrices of order n with entries ± 1 and satisfy AA' +BB' + CC' + DD' = 4nI then H is an Hadamard Matrix of order 4n. Thus the method of construction depends only on availability of the matrices A, B, C and D. The array H is also called as Williamson array.

Baumert-Hall method

A generalized form the Williamson array in which every matrix could appear more than once in the same row was given by Baumert and Hall (1965).

An example of size 12 (Hall, 1986) is given as

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

and H is an Hadamard matrix of order 12n where A, B, C and D are matrices of order n with entries ± 1 and satisfy AA' +BB' + CC' + DD' = 4nI.

Goethals-Seidel method

Goethals and Seidel (1967) proposed a construction method which is similar to Williamson array but with weaker conditions on matrices A,B, C and D.

The Goethals-Seidel array is of the form,

$$H = \begin{bmatrix} A & BR & CR & DR \\ -BR & A & D'R & -C'R \\ -CR & -D'R & A & B'R \\ -DR & C'R & -B'R & A \end{bmatrix}$$

Where A, B, C and D are circulant matrices of order n with entries ± 1 and satisfies AA' +BB' + CC' + DD' = $4nI_n$ and R is the back-diagonal identity matrices of order n. Then H is Hadamard matrix of order 4n.

A matrix is said to be circulant matrix if (i+1, j+1)th entry is equal to the (i, j)th entry [row and column numbers are reduced mod(n), when necessary].

Examples of circulant matrix (A) and back diagonal matrix (R) of order 3 is:

$$A = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix},$$

$$R = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Note that all the three methods of construction depend on finding 4 matrices A,B,C and D and they in turn depends on the initial rows, if we restrict only circulant matrices. Thus, much of the attention in construction of Hadamard matrices were aimed at finding the initial rows using computerized algorithm.

The initial rows of the 4 matrices are related to T-Sequences. Before introducing T-sequences, we define non-periodic autocorrelation function, N_A of any a sequence as

Given sequence A = (a_0, a_1… a_{n-1}) of length n, the non-periodic autocorrelation function, N_A is defined as

$N_A(s) = \sum a_i a_{i+s}$, for s = 0,1, ….., n-1

T-sequences

Four (**0, ±1**) sequence A, B, C and D of length **n** are called T-sequences, if

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

Besides T-Sequences, two other sequences namely base sequences and Turyn sequences are available.

Base sequences

Four (**±1**) sequence A, B, C and D of length **n+p, n+p, n, n** are called base sequences, if

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

Note that base sequences contain ±1 as entries and not of same length as compared to T-sequences which contain (0, ±1) as entries and all of them are of same order.

Turyn sequences

Four (**1, -1**) sequences X, Y, Z and W of length **n, n, n and n-1** respectively are called Turyn type sequences, if

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

Changing base and Turyn sequences to T-Sequences

It is possible to change base sequences and Turyn sequences to T-Sequences.

If A, B, C and D are 4 base sequences of lengths n+p, n+p, n and n respectively then the sequences

$$\left(\frac{1}{2}(A+B),\ \ 0_n\right), \left(\frac{1}{2}(A-B),\ \ 0_n\right), \left(0_{n+p}\ \ \frac{1}{2}(C+D)\right), \left(0_{n+p}\ \ \frac{1}{2}(C-D)\right)$$

where 0_n denotes a sequence of zeros of length n, form T-sequences of order 2n+p, 2n+p, 2n+p and 2n+p respectively.

If X, Y, Z and W are Turyn sequences of length n, n, n and n-1 respectively and construct A, B, C and D as

$$A = (Z \quad W), B = (Z \quad -W), C = (X), D = (Y)$$

Then A, B, C and D are base sequences of order 2n-1, 2n-1, n and n respectively

Thus, when Turyn type or base sequences are available, T-Sequences can be constructed.

Construction Hadamard matrix using T-sequences

The standard procedure of constructing an Hadamard matrix from T-sequences, T_1, T_2, T_3 and T_4 of length n is given by Seberry and Yamada (1992).

Step 1: Construct four circulant matrices, P_1, P_2, P_3, P_4 from the T-sequences T_1, T_2, T_3 and T_4 of order n.

Step 2: Construct

$A = P_1 + P_2 + P_3 + P_4$
$B = -P_1 + P_2 + P_3 - P_4$
$C = -P_1 - P_2 + P_3 + P_4$
$D = -P_1 + P_2 - P_3 + P_4$

Step 3: Use A, B, C, and D matrices in Goethals-Seidel array to get an Hadamard matrix of order 4n.

Availability of base sequences, Turyn and T-sequences

The list of base sequences, Turyn and T-sequences currently known are available in Koukouvinos (2007). The same list also contains initial rows of Williamson matrices. These sequences were used by the functions created in R.

R-Functions implemented

Function: had_williamson(x)

This function provides Hadamard matrix of order x where x is an Hadamard number and Williamson sequences of order x/4 are available. The pseudo code of the function is as follows:

(i) From x, get the value of order, order = x/4
(ii) Using seq_williamson(order) function, get specific Williamson sequences. The function seq_williamson reads the initial rows saved as R dataset.
(iii) Using the function circulat_mat() with the 4 initial sequences, construct A, B, C and D matrices
(iv) Use A,B,C and D matrices in the Williamson array to get Hadamard Matrix.

In case, if the Williamson sequence is not available or x is not an Hadamard number, the function would return NULL value.

Example

```
Console ~/

>
>    had_williamson(12)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
 [1,]    1    1    1    1   -1   -1    1   -1   -1     1    -1    -1
 [2,]    1    1    1   -1    1   -1   -1    1   -1    -1     1    -1
 [3,]    1    1    1   -1   -1    1   -1   -1    1    -1    -1     1
 [4,]   -1    1    1    1    1    1   -1    1    1     1    -1    -1
 [5,]    1   -1    1    1    1    1    1   -1    1    -1     1    -1
 [6,]    1    1   -1    1    1    1    1    1   -1    -1    -1     1
 [7,]   -1    1    1    1   -1   -1    1    1    1    -1     1     1
 [8,]    1   -1    1   -1    1   -1    1    1    1     1    -1     1
 [9,]    1    1   -1   -1   -1    1    1    1    1     1     1    -1
[10,]   -1    1    1   -1    1    1    1   -1   -1     1     1     1
[11,]    1   -1    1    1   -1    1   -1    1   -1     1     1     1
[12,]    1    1   -1    1    1   -1   -1   -1    1     1     1     1
```

Fig.1 Example of Hadamard matrix of order 12 from Williamson sequences of length 3 from Williamson method

From had_williamson( ) function we can generate Hadamard matrix of orders are 12, 20, 44, 52, 68, 76, 92, 100, 116, 124, 156, 172, 180, 204, 244 and 252.

Function: had_baumert(x)

This function can be used to construct Hadamard Matrices using Baumert Hall array. The input x to the function is an Hadamard number and Williamson sequences of order x/12 is available. The step for this method is as follows:

(i) From x, get the value of order; order = x/12.
(ii) Using seq_williamson(order) function, get specific Williamson sequences.
(iii) By using circulant_mat(), create the A, B, C and D circulant matrices.
(iv) Use Baumert Hall array with A, B, C and D matrices to generate Hadamard matrices of order x.

Example

```
> had_baumert(12)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
 [1,]    1    1    1    1   -1    1   -1   -1    1     1    -1    -1
 [2,]    1   -1    1   -1   -1   -1    1   -1   -1    -1    -1    -1
 [3,]    1   -1   -1    1   -1    1   -1    1   -1    -1     1    -1
 [4,]    1    1   -1   -1    1    1    1    1    1    -1    -1    -1
 [5,]    1   -1    1    1    1    1    1    1   -1     1    -1     1
 [6,]    1    1   -1    1    1   -1    1   -1   -1     1     1    -1
 [7,]    1   -1    1   -1    1   -1   -1    1    1     1     1    -1
 [8,]   -1   -1   -1   -1    1    1   -1   -1   -1     1    -1    -1
 [9,]    1   -1   -1   -1   -1    1    1   -1    1     1     1     1
[10,]   -1   -1    1    1    1    1    1   -1    1    -1     1    -1
[11,]    1   -1   -1    1    1   -1   -1   -1    1    -1    -1     1
[12,]   -1   -1   -1    1   -1   -1    1    1    1     1    -1    -1
```

Fig.2 example of Hadamard matrix of order 12 from Williamson sequences of length 1 from Baumert-Hall method

From had_baumert( ) function we can generate Hadamard matrix of orders are 12, 36, 60, 132, 156, 204, 228, 276, 300, 348, 372, 468, 516, 540, 612, 732 and 756.

Function: had_goethals_base(x)

This function can be used to generate Hadamard matrix of order x from Base sequence of order n, where

$$n = \frac{\left(\frac{x}{4} - 1\right)}{2}$$

Note that many base sequences are available in the form n+1, n+1, n and n. From these, one can generate T-sequences of order 2n+1, which in turn can be used to generate Hadamard matrix of order 4(2n+1). The steps implemented are as follows:

(i) From x, get the value of order; torder = x/4; order = (torder-1)/2
(ii) Getbase sequences using the function baseseq( )
(iii) Convert base sequences to T-sequences using the function base_to_T().
(iv) Use 4 T sequences as the first row to generate the matrices A, B, C and D.
(v) Use the matrices ingoethals_seidel_array() to generate Hadamard matrix of order x.

If the given order of the matrix, sequence is not available in the dataset or the order is not Hadamard number, the function would return NULL.

Example

```
> had_goethals_base(28)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15]
 [1,]    1    1   -1    1    1    1    1    1   -1     1     1     1    -1    -1     1
 [2,]    1    1    1   -1    1    1    1   -1    1     1     1    -1    -1     1     1
 [3,]    1    1    1    1   -1    1    1    1    1     1    -1    -1     1    -1     1
 [4,]    1    1    1    1    1   -1    1    1    1    -1    -1     1    -1     1    -1
 [5,]    1    1    1    1    1    1   -1    1   -1    -1     1    -1     1     1     1
 [6,]   -1    1    1    1    1    1    1   -1   -1     1    -1     1     1     1    -1
 [7,]    1   -1    1    1    1    1    1   -1    1    -1     1     1     1    -1    -1
 [8,]   -1    1   -1   -1   -1    1    1    1    1    -1     1     1     1     1    -1
 [9,]    1   -1   -1   -1    1    1   -1    1    1     1    -1     1     1     1     1
[10,]   -1   -1   -1    1    1   -1    1    1    1     1     1    -1     1     1     1
[11,]   -1   -1    1    1   -1    1   -1    1    1     1     1     1    -1     1    -1
[12,]   -1    1    1   -1    1   -1   -1    1    1     1     1     1     1    -1     1
[13,]    1    1   -1    1   -1   -1   -1   -1    1     1     1     1     1     1    -1
[14,]    1   -1    1   -1   -1   -1    1    1   -1     1     1     1     1     1    -1
[15,]   -1   -1   -1    1   -1    1    1    1   -1    -1     1    -1     1     1     1
[16,]   -1   -1    1   -1    1    1   -1   -1   -1     1    -1     1     1     1     1
[17,]   -1    1   -1    1    1   -1   -1   -1    1    -1     1     1     1    -1     1
[18,]    1   -1    1    1   -1   -1   -1    1   -1     1     1     1    -1    -1     1
[19,]   -1    1    1   -1   -1   -1    1   -1    1     1     1    -1    -1     1     1
[20,]    1    1   -1   -1   -1    1   -1    1    1     1    -1    -1     1    -1    -1
[21,]    1   -1   -1   -1    1   -1    1    1    1    -1    -1     1    -1     1     1
[22,]    1   -1    1   -1   -1    1    1   -1    1    -1     1     1     1    -1     1
```

Fig.3 Partial rows and columns of Hadamard matrix of order 28 from Base sequences of length 28 from Goethals-Seidel method

The function, had_goethals_base( ) function we can generate Hadamard matrix of orders are 28, 44, 76, 188, 196, 228, 236, 260, 268 and 284.

Function: had_goethals_Turyn(x)

In case, if Turyn sequences are available, Hadamard matrices can be generated by converting the Turyn type sequence to base sequence and then convert them into T-Sequences and construct using Goethals-Seidal array. The steps implemented in the function are follows:

(i) From x, get the value of order; torder = x/4; order = (torder+1)/3.

(ii) Using order in Tseq() function, get Turyn sequences of length order, order, order and order-1.

(iii) Using these 4 sequences, call function T_to_base() to generate base sequences of length (2*order)+p, (2*order)+p, (2*order)+p and (2*order)+p. where p = order-1.

(iv) Using these Base sequences, create the A, B, C and D circulant matrices by circulant_mat( )

(v) Callgoethals_seidel_array( ) to get Hadamard matrix

As in the previous case, in case, for the given order of the matrix, sequence is not available the function will return NULL.

Example

```
Console ~/
>
>     had_goethals_Turyn(356)
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16]
[1,]    1   -1   -1   -1    1   -1    1    1    1    -1    -1     1     1    -1     1    -1
[2,]   -1    1   -1   -1   -1    1   -1    1    1     1    -1    -1     1     1    -1     1
     [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31]
[1,]     1     1     1     1     1     1    -1     1    -1     1    -1    -1     1     1    -1
[2,]    -1     1     1     1     1     1     1    -1     1    -1     1    -1    -1     1     1
     [,32] [,33] [,34] [,35] [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46]
[1,]     1     1    -1     1     1    -1    -1     1    -1     1    -1    -1     1     1     1
[2,]    -1     1     1    -1     1     1    -1    -1     1    -1     1    -1    -1     1     1
     [,47] [,48] [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61]
[1,]     1     1     1     1     1     1    -1     1     1    -1    -1    -1     1     1     1
[2,]     1     1     1     1     1     1     1    -1     1     1    -1    -1    -1     1     1
     [,62] [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73] [,74] [,75] [,76]
[1,]    -1    -1     1    -1     1     1     1    -1    -1    -1    -1    -1     1     1     1
[2,]     1    -1    -1     1    -1     1     1     1    -1    -1    -1    -1    -1     1     1
     [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85] [,86] [,87] [,88] [,89] [,90] [,91]
[1,]     1    -1    -1    -1     1     1    -1     1    -1     1     1     1    -1    -1     1
[2,]     1     1    -1    -1    -1     1     1    -1     1    -1     1     1     1     1    -1
     [,92] [,93] [,94] [,95] [,96] [,97] [,98] [,99] [,100] [,101] [,102] [,103] [,104] [,105]
[1,]    -1    -1     1     1     1     1     1     1     -1      1     -1      1      1     -1
[2,]    -1     1     1     1     1     1     1    -1      1     -1      1      1     -1      1
     [,106] [,107] [,108] [,109] [,110] [,111] [,112] [,113] [,114] [,115] [,116] [,117]
[1,]      1     -1     -1      1     -1     -1     -1      1      1      1     -1     -1
[2,]     -1     -1      1     -1     -1     -1      1      1      1     -1     -1     -1
     [,118] [,119] [,120] [,121] [,122] [,123] [,124] [,125] [,126] [,127] [,128] [,129]
[1,]     -1      1      1     -1     -1     -1      1      1     -1      1      1      1
[2,]      1      1     -1     -1     -1      1      1     -1      1      1      1      1
```

Fig.4 Partial rows of Hadamard matrix of order 356 from Turyn sequences of length 28 from Goethals-Seidel method

From had_goethals_Turyn() function, we can generate Hadamard matrix of orders are 356, 404 and 428.

Conclusions of the study are: hadamard matrices can be easily generated when the initial rows of the matrices to be used in Williamson, Baumart-Hall or Goethals-Seidal methods are available. The functions implemented would be useful for anyone wishes to generate Hadamard matrices. Moreover, most of the new methods of construction of Hadamard matrices are using these methods. For example, the Hadamard Matrix of order 448 which was obtained by Kharaghani and Tayfeh-Rezaie (2004) was using Turyn Sequences. The generated matrices were tested for the all the orders mentioned above and found correct. The functions are part of the package which is being developed to implement different construction methods of Hadamard matrices. Once the package is ready, the same would be submitted to CRAN. Till then, anyone interested can contact the Authors for the codes of the functions mentioned in this article.

Acknowledgements

The author wishes to thank Head, Department of Agricultural Statistics, Indira Gandhi Krishi Vishwavidyalaya, Raipur, Chhattisgarh and Director, ICAR-National Academy of Agricultural Research Management, Hyderabad for providing necessary facilities and help in carrying out this work. The first author thank to ICAR for providing NTS fellowship.

References

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

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

Gupta, V.K., Dhandapani, A., Parsad, R (2007). Hadamard Matrices. Indian Agricultural Statistics Research Institute, New Delhi.

Hall, M. Jr. (1986). Combinatorial Theory, 2ndEedition. John Wiley & Sons, New York.

Hedayat, A. and Wallis, (1978). W. D. Hadamard Matrices and Their Applications. *Ann. Stat.* 6, 1184-1238.

Kharaghani, H. and Tayfeh-Rezaie, B. (2005).An Hadamard matrix of order 428. *J. Combin. Designs*, 13, 435-440.

Koukouvinos, C. (2007). http://www.math.ntua.gr/~ckoukouv/(last accessed on 5-7-2019)

Seberry, J. (2007). A Life's Work on Hadamard Matrices, Statistical Designs, Bent Functions and their Application to Computer and Information Security in Networks and Telecommunications. (Available at https://documents.uow.edu.au/~jennie/lifework.html, last accessed on 5-7-2019).

Seberry, J. and Yamada, M. (1992). Hadamard matrices, sequences and block designs. Chapter 11 in contemporary Design Theory; A collection of surveys (Ed. J. H. Dinitz and D. R. Stinson). John Wiley, New York, 431-560.

Slone, N.J.A. (2019). A Library of Hadamard Matrices. http://neilsloane.com/hadamard/ (last accessed on 5-7-2019)

Williamson, J. (1994). Hadamard's determinant theorem and the sum of four square. *Duke. Math. J.*, 11, 65-81.

Williamson, J. (1947). Note on Hadamard's determinant theorem. *Bull. Amer. Math. Soc.*, 53, 608-613.

How to cite this article:

Revanasiddesha, A. Dhandapani, N.K. Choure and Lakhera, M.L. 2019. Construction of Hadamard Matrices in R. *Int.J.Curr.Microbiol.App.Sci.* 8(07): 2255-2261.
doi: https://doi.org/10.20546/ijcmas.2019.807.275