Facultad
de
Ciencias

Matrices de Hadamard

(Hadamard matrices)

Autora: Paula Tamayo Saiz
Director: Calos Beltrán Álvarez
Junio-2024

Agradecimientos

En primer lugar he de dar las gracias a mi tutor, Carlos Beltrán Álvarez, por su preocupación, dedicación y esfuerzo, ya que sin su ayuda no me hubiera sido posible la realización del presente trabajo.

A todos los docentes que me han enseñado a lo largo de mis años de carrera la gran parte de mis actuales conocimientos.

También me gustaría dar las gracias a mis amigos, tanto a los de Burgos como a los que he conocido en Santander. Gracias por su apoyo, que ha sido fundamental para conseguir llegar hasta aquí.

Y por último a mi familia, a mis padres por enseñarme que la constancia y el trabajo, aunque no lo parezca, tienen su recompensa, por animarme y siempre conseguir que fuese un poco más fácil. A mi hermano, por acompañarme en todo momento. Y a mis tios y abuelos por estar siempre pendientes. Gracias.

Resumen

A lo largo de este trabajo se definirán las matrices de Hadamard y se hablará sobre algunas de sus características y de sus posibles órdenes. Se verán las diferentes maneras de construirlas, añadiendo ejemplos y haciendo uso de programas creados en Matlab, donde nos apoyaremos para la construcción de estas. Se desarrollarán tres métodos: Construcción de Sylvester, Construcción de Paley y Producto de Kronecker. Con ellos, se abarcarán muchas de las matrices de Hadamard de orden múltiplo de 4 inferior a 1000 que cumplan las características necesarias para serlo. También se hablará sobre el valor del determinante, definiendo la Desigualdad de Hadamard, para ver que el valor del determinante de estas matrices es maximal. La última sección de este trabajo será dedicada a una de sus principales aplicaciones, la teoría de códigos, donde se verá la importancia de conocer estas matrices. Definiendo conceptos clave e introduciéndola para una posterior comparación de ejemplos donde determinaremos la importancia de las matrices a escoger.

Palabras clave: **Matrices Hadamard, residuos cuadráticos, Matrices de Jacobsthal, Función χ de Legendre, determinante y códigos.**

Abstract

Throughout this work, Hadamard matrices will be defined, and some of their characteristics and possible orders will be discussed. Different ways to construct them will be examined, including examples and the use of programs created in Matlab, which will support their construction. Three methods will be developed: Sylvester's Construction, Paley's Construction, and the Kronecker Product. These methods will cover many Hadamard matrices of order multiple of 4 less than 1000 that meet the necessary characteristics. The value of the determinant will also be discussed, defining Hadamard's Inequality, to show that the determinant value of these matrices is maximal. The last section of this work will be dedicated to one of their main applications, coding theory, where the importance of constructing these matrices will be highlighted. Key concepts will be defined, and it will be introduced for a subsequent comparison of examples where the importance of the matrices to be chosen will be determining.

Key words: **Hadamard matrix, quadratic residues, Jacobsthal matrix, Legendre χ function, determinant and codes.**

Índice

Introducción.

A lo largo de este trabajo estudiaremos las propiedades y características de las matrices de Hadamard para después desarrollar una aplicación de estas.

Una matriz de Hadamard es un tipo de matriz cuadrada que fue usada por primera vez por Sylvester en 1867. Pasados 26 años, fue Jacques Salomon Hadamard, en 1893, quien consideró que era importante estudiarlas, véase [3].

Jacques Salomon Hadamard fue un matemático francés que contribuyó en la teoría de los números, el análisis complejo, la geometría diferencial ylas ecuaciones diferenciales parciales. En concreto en este artículo, nos interesa su interés y dedicación a las matrices de Hadamard.

Una de las principales preocupaciones del francés fue si para cualquier orden existía esta matriz en particular, es decir, cuáles eran las hipótesis a cumplir para que existiera una matriz de un determinado orden. Esto lo mencionaremos más específicamente en la primera sección del documento y se conoce como: *Conjetura de Hadamard*.

Jacques Hadamard también se interesó sobre la importancia de los determinantes de las matrices cuadradas, construyendo una desigualdad, *La Desigualdad de Hadamard*, sobre la que hablaremos y que demostraremos en la tercera sección del artículo. El matemático hizo ver que esta alcanzaría la igualdad en el caso de que la matriz sea de Hadamard, lo cual trataremos de probar en este trabajo.

Hablaremos con detalle sobre estas matrices y veremos cómo conseguir matrices equivalentes a una dada y matrices normalizadas.

Uno de los factores más importantes a tratar en este trabajo, es la construcción de matrices de Hadamard. El objetivo será conseguir un buen número de los órdenes posibles con existencia de estas matrices, menores de 1000. En este documento detallaremos únicamente tres métodos para construirlas, que serán los principales y con los que más matrices conseguiremos:

1. Construcción de Sylvester.
2. Producto de Kronecker.
3. Construcción de Paley.

Otros métodos que no se especificarán en este documento, pero mediante los cuales podremos conseguir matrices de este tipo son:
Williamson, Goethals-Seidel, Baumert-Hall, Enlich, Miyamoto, Scarpis, Wallis, Conjuntos Diferencia Suplementarios y Cooper-Wallis. En **Anexo 1** se muestra qué metodo será usado para construir la matriz en cada orden en específico, véase [9].

Hay órdenes para los que es posible aplicar más de un método para construir la matriz y otros para los que aún se desconoce la existencia de una matriz de Hadamard.

Una vez que conozcamos qué es una matriz de Hadamard, estudiemos la existencia de una en un determinado orden y sepamos cómo construirla, pasaremos a ver sus posibles aplicaciones.

Algunas de las principales aplicaciones de las matrices de Hadamard son:

1. Teoría de códigos.
2. Diseño de pesadas.
3. Geometría de los espacios de Banach.

En este artículo, detallaremos una de estas tres aplicaciones, la teoría de códigos. Lo haremos en la última sección y para ello, explicaremos de manera teórica la teoría de códigos y después mencionaremos algunos ejemplos donde aplicaremos lo explicado y otros para comparar y ver la importancia a la hora de elegir una matriz u otra. También comprobaremos si las matrices de Hadamard son una buena opción para codificar un código de una manera óptima, con el mayor número de palabras, una menor longitud y una mayor distancia mínima entre ellas o no.

Con esto veremos la importancia del estudio sobre las matrices de Hadamard y de conseguir encontrar todas las matrices de orden menor que 1000, ya que sus aplicaciones son útiles en diversos ámbitos.

1. Matrices de Hadamard: definición y ejemplos.

Para la realización de este trabajo, hemos tomado de base el documento [8], añadiendo demostraciones, contraejemplos, ejemplos y teoría necesaria para conseguir una redacción lo más completa posible de estas matrices y de una de sus aplicaciones.

Comenzaremos este apartado definiendo qué es una matriz de Hadamard, mostrando cuáles son sus principales propiedades.

Definición 1:
Una matriz de Hadamard H de orden n es una matriz n×n, formada por $1's$ *y* $-1's$ *cuyas filas son ortogonales (el producto escalar de dos filas distintas cualesquiera, es igual a cero), es decir,* $HH^t = nI$.

Observando dos columnas o filas, la mitad de las entradas tendrán signos coincidentes mientras que en la otra mitad ocurrirá lo contrario. Con lo cual, cuando n ≠ 1, habrá tantos 1's como $-1's$. Por consecuencia, n debe ser par, véase [5].

Ejemplo 1:
Dada esta primera definición, las siguientes matrices serán unos ejemplos con dichas características:

$$H_1 = (1), \quad H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad H_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Corolario 1:
Si H es matriz de Hadamard, H^t*, también lo es.*
Demostración:
Como H es matriz de Hadamard, cumple $HH^t = nI$.
Veamos que sucede con H^t.

$$HH^t = nI \implies \left(\frac{H}{\sqrt{n}}\right)\left(\frac{H^t}{\sqrt{n}}\right) = I.$$

Sabemos por un teorema de álgebra que si $AB{=}I$ entonces $BA{=}I$.
De esto se puede ver que $\frac{H^t}{\sqrt{n}} = \left(\frac{H}{\sqrt{n}}\right)^{-1}$ y por ello se cumplirá:

$$\left(\frac{H^t}{\sqrt{n}}\right)\left(\frac{H}{\sqrt{n}}\right) = I \implies H^t H = nI.$$

□

Definición 2:
Al multiplicar cualquier fila o columna por -1 en una matriz de Hadamard, se obtiene otra matriz de Hadamard. Estas matrices serán llamadas equivalentes.
Intercambiando filas o columnas, también se obtienen matrices equivalentes, véase [5].

Ejemplo 2:
Considerando la matriz H_4 del ejemplo anterior

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Multiplicando la segunda fila por -1, obtengo una matriz cuyas filas son ortogonales:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}.$$

Además, por definición de matrices de Hadamard:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$

Vamos a demostrarlo de manera genérica:
Demostración:
Sea A una matriz de Hadamard $n \times n$, por lo que verifica $AA^t = nI$.
Tomamos v el vector de la fila k-ésima de la matriz A.
Aplicando la **Definición 3** tenemos que (siendo A_i la fila i-ésima de A):

$$A_i A_j^t = \begin{cases} n & \text{si } i = j \\ \\ 0 & \text{si } i \neq j \text{ (Por ortogonalidad)} \end{cases}$$

Definimos una matriz B de dimensiones $n \times n$, tal que sea como A pero modificando el vector fila v por $-v$. Así queda B definida como: $[A_1; A_2; \ldots; \text{-}v; \ldots; A_n]$.
Ahora probaremos que la nueva matriz construida es de Hadamard:

- En las filas distintas de la k-ésima, por ser A una matriz de Hadamard:

$$B_i B_j^t = A_i A_j^t = \begin{cases} n & \text{si } i = j \\ \\ 0 & \text{si } i \neq j \text{ (Por ortogonalidad)} \end{cases}$$

- La fila del vector $-v$, que es la fila k de A cambiada de signo, $-v(-v^t) = vv^t = ||v||_2^2 = ||A_k||_2^2 = n$, por lo que tenemos, al igual que en el caso anterior, lo mismo que con la matriz A.

- Por último falta ver el caso de B_i y $(-v)$, con i $\neq k$. Sabiendo que las columnas son ortogonales, se da que $B_i(-v) = A_i(-A_k) = 0$.

Por lo visto arriba, los cambios de signo no tendrán repercusión a la hora de demostrar que la multiplicación matricial BB^t es n veces la identidad.

Si en lugar de multiplicar una fila, multiplico una columna, por simetría de la matriz, una demostración similar es fácil de producir.

Aportando una demostración diferente, veremos que cambiando el signo a la columna k-ésima, la nueva matriz B será: $A \cdot J$, donde A es la matriz de Hadamard $n \times n$ de partida, y J la identidad con la entrada k-ésima cambiada de signo. De una forma más rápida que la anterior, veremos que ocurre lo mismo que en el caso de las filas:

Sabiendo que $AA^t = nI$, obtendremos:

$$BB^t = (AJ)(AJ)^t = A(JJ^t)A^t = nI \tag{1}$$

$$\Longrightarrow BB^t = nI.$$

□

Definición 3:
Podemos cambiar la primera fila y la primera columna para que estén enteramente formadas solo por 1's. Las matrices de este tipo se denominan normalizadas.

Por lo explicado hasta el momento sobre estas matrices, surgen varias cuestiones. La primera de ellas es: ¿existen matrices de Hadamard de cualquier orden?. La respuesta irá de la mano de varios teoremas:

Teorema 1: (Hadamard)
Sea H una matriz de Hadamard de orden n, n será 1, 2 o múltiplo de 4 cuando $n > 2$.
Demostración:
Para esta demostración me he basado en las indicaciones de [5] modificando estas y añadiendo por mi parte lo necesario para conseguir una demostración completa.

Definimos una matriz de Hadamard cualquiera H_n con $n > 2$, y tenemos que probar que n es múltiplo de 4.
Supongamos que H_n es normalizada, por lo que al cambiar las columnas de orden a nuestra conveniencia, podemos considerar que las tres primeras filas de H, con c_i $i \in \{1, \ldots, 4\}$ el número de columnas en cada conjunto, son:

$$\begin{array}{cccc} 11\ldots1 & 11\ldots1 & 11\ldots1 & 11\ldots1 \\ 11\ldots1 & 11\ldots1 & -1-1\ldots-1 & -1-1\ldots-1 \\ \underbrace{11\ldots1}_{c_1} & \underbrace{-1-1\ldots-1}_{c_2} & \underbrace{11\ldots1}_{c_3} & \underbrace{-1-1\ldots-1}_{c_4} \end{array}$$

Como la matriz es de orden n, las columnas sumarán n y por ser matriz de Hadamard, las filas son ortogonales (**Definición 1**), por lo que se tienen las siguientes ecuaciones:

$$\begin{aligned} c_1 + c_2 + c_3 + c_4 &= n, \quad (1) \\ c_1 - c_2 - c_3 + c_4 &= 0, \quad (2) \\ c_1 + c_2 - c_3 - c_4 &= 0, \quad (3) \\ c_1 - c_2 + c_3 - c_4 &= 0, \quad (4) \end{aligned}$$

Sumando (1) + (2) + (3) + (4) obtendremos que $4c_1 = n$, por lo que n es múltiplo de 4. □

La siguiente cuestión interesante es: ¿existen matrices de Hadamard de cualquier orden múltiplo de 4?. La respuesta afirmativa a esta pregunta se denomina **'*Conjetura de Hadamard*'**, que hace referencia al recíproco del **Teorema 1**.

Una vez visto que las matrices de Hadamard tienen que tener orden múltiplo de 4, veremos que existen varios métodos de generación de estas matrices, entre los que cabe destacar algunos de los que hablaremos más específicamente a lo largo del documento (véase [9] para una explicación completa):

- Construcción de Sylvester.
- Construcción de Paley.
- Producto de Kronecker

Empezaremos con la Construcción de Sylvester. Antes introduzcamos un concepto necesario para su generalización.

Definición 4: Producto de Kronecker
Sean A una matriz de tamaño $n \times n$ con entradas a_{ij} y B otra matriz cualquiera. Multiplicando cada entrada de A por la matriz B, obtenemos:

$$\begin{pmatrix} a_{11}B & a_{12}B & \dots & a_{1n}B \\ a_{21}B & a_{22}B & \dots & a_{2n}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1}B & a_{n2}B & \dots & a_{nn}B \end{pmatrix}$$

Esto se conoce como Producto de Kronecker de las matrices A y B y se denota $A \otimes B$, véase [5].

Ejemplo 3:
Veamos un ejemplo:

$$A = \begin{pmatrix} 2 & 4 \\ 6 & 8 \end{pmatrix}; \quad B = \begin{pmatrix} 10 & 20 \\ 30 & 40 \end{pmatrix}$$

$$A \otimes B = \begin{pmatrix} 2\cdot 10 & 2\cdot 20 & 4\cdot 10 & 4\cdot 20 \\ 2\cdot 30 & 2\cdot 40 & 4\cdot 30 & 4\cdot 40 \\ 6\cdot 10 & 6\cdot 20 & 8\cdot 10 & 8\cdot 20 \\ 6\cdot 30 & 6\cdot 40 & 8\cdot 30 & 8\cdot 40 \end{pmatrix} = \begin{pmatrix} 20 & 40 & 40 & 80 \\ 60 & 80 & 120 & 160 \\ 60 & 120 & 80 & 160 \\ 180 & 240 & 240 & 320 \end{pmatrix}$$

Volviendo al **Ejemplo 1**, en H_4 podremos observar cierta casualidad en su construcción, esto es, su procedencia de la matriz anterior, H_2, producto de Kronecker con ella misma:

$$H_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} = \begin{pmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{pmatrix} = H_2 \otimes H_2$$

Observado esto podremos generalizar esta construcción para las matrices de órdenes potencia de dos: 1, 2, 4, 8, ..., 2^n, que son las matrices de Sylvester, construidas a través del producto de Kronecker iterado, de n copias de la matriz de Hadamard H_2 de orden 2, ver [10].

Proposición 1:
Las matrices obtenidas mediante la siguiente fórmula, llamadas matrices de Sylvester, son de Hadamard:

$$H_{2n} = \begin{pmatrix} H_n & H_n \\ H_n & -H_n \end{pmatrix}.$$

Demostración:
Vamos a demostrarlo por inducción:
Conociendo el caso n=2, ya visto, por hipótesis de inducción suponemos cierto para el caso $H_n H_n^t = nI$.
Veamos que se verifica para H_{2n}:

$$\begin{pmatrix} H_n & H_n \\ H_n & -H_n \end{pmatrix} \begin{pmatrix} H_n^t & H_n^t \\ H_n^t & -H_n^t \end{pmatrix} = \begin{pmatrix} H_n H_n^t + H_n H_n^t & 0 \\ 0 & H_n H_n^t + H_n H_n^t \end{pmatrix} =$$

$$= \begin{pmatrix} 2nI & 0 \\ 0 & 2nI \end{pmatrix} = 2nI$$

Concluyendo así que es matriz de Hadamard. □

Observación:
De esto podemos confirmar que si conocemos una matriz de Hadamard, de orden cualquiera k, haciendo producto de Kronecker con H_2 mencionada en el **Ejemplo 1**, encontraremos una matriz de orden 2k que también será de Hadamard.

Hasta ahora conocemos un método de contrucción para calcular estas matrices, con el cual solo conseguimos matrices de orden potencia de 2, pero no sabemos construir matrices de otros órdenes, como pueden ser 12, 20 o 28.

2. Residuos cuadráticos.

En este segundo apartado, definiremos los conceptos necesarios para en el siguiente apartado, explicar con detalle en que consiste el método de Construcción de Paley para conseguir matrices de Hadamard.

Definición 5: Residuos cuadráticos.
Llamamos residuos cuadráticos módulo p, a aquellos enteros positivos $1^2, 2^2, 3^2, \ldots, (p-1)^2$ *módulo p, con p un primo mayor que 2.*

Para poder calcular de un modo más rápido los residuos cuadráticos del primo impar que queramos, he creado en Matlab una función:

```
function[r,n] = cuadratico(p)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculará los residuos cuadráticos módulo p del primo p
%ENTRADA:
%p= primo impar para las congruencias modulo p.
%SALIDA:
%r= lista de residuos cuadráticos del primo p impar.
%n= número de residuos
%Uso de la funcion de Matlab:
%b = mod(a,m) devuelve el resto después de dividir a por m, a dividendo y m divisor
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
r = [];
n=0;
for i = 1:p-1
    aux=mod(i^2,p);
    if aux==0
        r(i)=1;   %para que no me saque el cero
    else
        r(i) = aux;
    end
end
r = unique(r);
n=size(r,2);
end
```

Ejemplo 4:
Hagamos un par de ejemplos manualmente.

Sea p=3:

$$1^2 \pmod{p} \equiv 1 \quad \text{y} \quad 2^2 \pmod{p} \equiv 1$$

Por lo que para p=3, el único residuo cuadrático módulo 3 es 1.

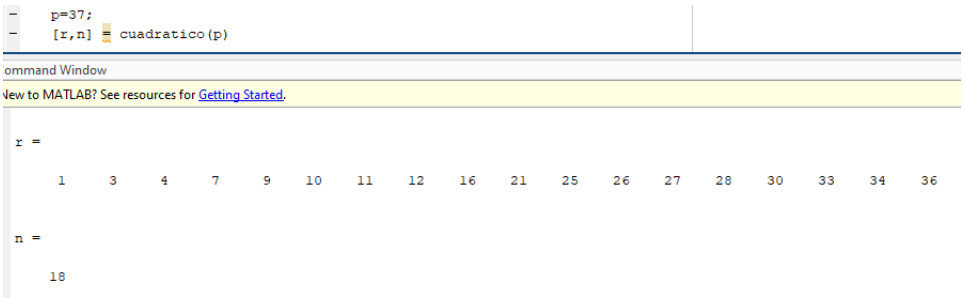
Sea p=7:

$$1^2 \pmod{p} \equiv 1 \quad \text{y} \quad 2^2 \pmod{p} \equiv 4$$
$$3^2 \pmod{p} \equiv 2 \quad \text{y} \quad 4^2 \pmod{p} \equiv 2$$

$$5^2 \pmod{p} \equiv 4 \quad \text{y} \quad 6^2 \pmod{p} \equiv 1$$

En este caso los residuos cuadráticos módulo 7 son 1, 2 y 4.

Usando la función de Matlab, vemos que los residuos de p=37, son :
{1, 3, 4, 7, 9, 10, 11, 12, 16, 21, 25, 26, 27, 28, 30, 33, 34, 36}.



Lema 1:
Sean i,j enteros positivos. Si $1 \leq i, j \leq p-1$*, entonces tienen el mismo residuo si y solo si* $i = p - j$ *o* $i = j$*.*
Demostración:
Para realizar esta demostración, he seguido la idea de [6], modificándolo y adaptándolo a mis notaciones y necesidades para esta demostración. También me he guiado por las indicaciones generales que aparecen en [8].
La demostración sería la siguiente:
$\Longrightarrow$
Siendo $1 \leq i, j \leq p-1$, si tienen el mismo residuo $\Longrightarrow$ $i = p - j$ o $i = j$.

Sean i, j dos enteros entre 1 y p - 1 tales que $i^2 \equiv j^2 \pmod{p}$, por definición, p será divisor de $i^2 - j^2$.
Desarrollándolo tenemos que $i^2 - j^2 = (i + j)(i - j)$, por lo que p, por ser primo, dividirá a uno de los dos términos.
Como tanto i como j están entre 1 y p - 1, obtenemos:

$$2 \leq i + j \leq 2p - 2 \quad \text{y} \quad 2 - p \leq i - j \leq p - 2$$

Distinguiremos dos casos:

- Si p divide a i + j, tenemos que solo se dará en el caso: i + j = p. Consiguiendo así i = p - j.
- Si p divide a i - j, tenemos que solo será posible si i - j = 0, ya que p - 2 es menor que p. De este caso conseguimos i=j.

Vemos que en el primer caso, $p - j = i$ y j, sus residuos son congruentes módulo p, al igual que ocurre con $i = j$ y j en el segundo caso.
$\Longleftarrow$
Con $1 \leq i, j \leq p-1$, si $i = p-j$ o $i = j \Longrightarrow$ de i y j se obtiene el mismo residuo.

Usando las propiedades de los módulos, darán como solución un mismo residuo:

$$(p-j)^2 = p^2 + j^2 - 2jp$$

Si hacemos congruencias módulo p,

$$p^2 + j^2 - 2jp = p(p-2j) + j^2 \implies$$

$$(p-j)^2 = p(p-2j) + j^2 \equiv j^2 \pmod{p}$$

Tanto p - j (es decir i) como j, darán lugar al mismo residuo cuadrático.
Quedando así demostrada la doble implicación. □

Corolario 2:
El número de enteros residuos cuadráticos módulo p primo, es (p - 1)/2.
Demostración:
Para encontrar todos los residuos cuadráticos módulo p entre los enteros positivos menores de p (buscamos cuadrados perfectos módulo p), consideramos las p - 1 congruencias de los cuadrados de $1, 2, \ldots, (p-1)$ y dado que cada congruencia no tiene solución o bien tiene dos soluciones congruentes (visto antes), debe haber exactamente (p - 1)/2 residuos cuadráticos de p entre 1,2,...,p - 1. Los restantes les llamaremos no residuos, y serán también (p - 1)/2. El 0 no será ni residuo ni no residuo. □

Ejemplo 5:
Por lo que acabamos de ver en el **Corolario 2**, en los casos destacados en el **Ejemplo 4** anterior, tendrá que haber (p - 1)/2 residuos y (p - 1)/2 no residuos en cada caso. Veamos si esto se cumple:

Para p=3, $(3-1)/2 = 1$ residuo cuadrático y otro no residuo.
Hemos visto que 1 es residuo cuadrático y que 2 no es residuo por lo que se verifica.

Por otro lado, para p=7, los residuos cuadráticos serán 1,2 y 4, y los no residuos, 3,5 y 6, es decir $(7-1)/2 = 3$ residuos y 3 no residuos.

En la función de Matlab, adjuntada anteriormente, el programa además de devolver los residuos, devuelve el número de estos. En el caso de p=37 serán 18, que son exactamente $(37-1)/2$.

```
p=37;
[r,n] = cuadratico(p)
```

```
ommand Window
lew to MATLAB? See resources for Getting Started.

r =

     1     3     4     7     9    10    11    12    16    21    25    26    27    28    30    33    34    36

n =

    18
```

3. Construcción de Paley.

En este apartado veremos el tercer y último método de construcción de matrices de Hadamard, utilizado para poder construir aquellas que no sea posible con el método de Sylvester o el producto de Kronecker, ya explicados en la sección 1. Sería el siguiente:

Definición 6: Función χ de Legendre.
Definimos la función χ de Legendre sobre los enteros, como:

$$\chi(i) = 0, \qquad si\ i \equiv 0 \pmod{p}$$

$$\chi(i) = 1, \qquad si\ i \pmod{p}\ es\ residuo.$$

$$\chi(i) = -1, \qquad si\ i \pmod{p}\ no\ es\ residuo.$$

Definición 7: Matrices de Jacobsthal.
Sea p un primo mayor que 2, definimos la matriz cuadrada Q de orden p tal que $Q=(q_{ij})$, con $q_{ij}=\chi(j-i)$. Dichas matrices se conocen como matrices de Jacobsthal.

Al igual que hice con los residuos cuadráticos, para poder calcular matrices de Jacobsthal de una forma rápida, he creado una función en Matlab para construirlas.:

```
Q=[];
[r,n] = cuadratico(p);
for i=1:p
    for j=1:p
        valor=j-i;
        if valor < 0
            valor = valor +p;
            if ismember(valor,r)==1
                Q(i,j)=1;
            else
                Q(i,j)=-1;
            end
        else
            if ismember(valor,r)==1
                Q(i,j)=1;
            else
                Q(i,j)=-1;
            end
            if i==j
                Q(i,j)=0;
            end
        end
    end
    Q;
end
```

Ejemplo 6:
Si tomamos p=7, por lo que $i, j \in [1, 7]$, vamos a construir paso a paso la matriz de Jacobsthal, lo haremos por diagonales.
Sabiendo por el **Ejemplo 4** que 1,2 y 4 son los residuos cuadráticos del 7.

- Primero miro los elementos diagonales:
  $q_{11} = \chi(1-1)$, $q_{22} = \chi(2-2)$, ..., $q_{ii} = \chi(i-i)$ por lo que todos serán "0"por ser $0 \equiv 0 \pmod 7$.

- Para los elementos de las diagonales superiores:
  En la primera diagonal superior, $q_{12} = \chi(2-1) = q_{23} = \chi(3-2) = \ldots = q_{i(i+1)} = \chi((i+1)-i) = \chi(1) = 1$, todos serán "1"por ser 1 residuo cuadrático de 7 (visto en el **Ejemplo 4**).
  Ocurrirá lo mismo con aquellos elementos cuya resta de coordenadas sea 2, la siguiente diagonal, $q_{13} = \chi(3-1) = q_{24} = \chi(4-2) = \ldots = q_{i(i+2)} = \chi((i+2)-i) = \chi(2) = 1$.
  Y con los de la cuarta diagonal superior, $q_{15} = \chi(5-1) = q_{26} = \chi(6-2) = q_{37} = \chi(7-3) = \chi(4) = 1$.

  Sin embargo, en las tres diagonales superiores restantes, la diferencia será un número no residuo cuadrático de 7, $q_{i(i+3)} = \chi(3)$, $q_{i(i+5)} = \chi(5)$ y $q_{i(i+6)} = \chi(6)$. Tendrán valor " -1" por la **Definición 6**.
  Así ya tendremos:

$$\begin{pmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 \\ & 0 & 1 & 1 & -1 & 1 & -1 \\ & & 0 & 1 & 1 & -1 & 1 \\ & & & 0 & 1 & 1 & -1 \\ & & & & 0 & 1 & 1 \\ & & & & & 0 & 1 \\ & & & & & & 0 \end{pmatrix}$$

- Para rellenar, la parte inferior, se seguirá el mismo patrón. En este caso la diferencia será un entero negativo, con lo cual le sumaremos 7 (p) para conseguir un entero positivo y poder así calcular si es residuo o no. Por lo que resultará la siguiente matriz:

$$\begin{pmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 0 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 0 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 0 \end{pmatrix} \tag{2}$$

y vemos que con la función de Matlab obtenemos el mismo resultado, como era de esperar:

```
p=7;
[Q] = jacobsthal(p)
```

```
ommand Window
lew to MATLAB? See resources for Getting Started.

Q =

     0     1     1    -1     1    -1    -1
    -1     0     1     1    -1     1    -1
    -1    -1     0     1     1    -1     1
     1    -1    -1     0     1     1    -1
    -1     1    -1    -1     0     1     1
     1    -1     1    -1    -1     0     1
     1     1    -1     1    -1    -1     0
```

Una vez definidas la función χ de Legendre y las matrices de Jacobsthal construidas a partir de esta, el artículo [8] que estoy usando de base sigue con las siguientes líneas:

"*Si ponemos ahora*

$$\begin{pmatrix} 1 & \mathbf{1} \\ \mathbf{1} & Q - I \end{pmatrix}$$

Se puede probar que se obtiene una matriz de Hadamard de orden $p+1$."

Leyendo esto, se puede pensar que es algo que se verifica para todo p, incluso que podría ser el enunciado de un teorema. Lo cierto que es que esto es incorrecto, y que el artículo podría crear confusión haciendo creer lo contrario.

Este enunciado es cierto solo para algunos primos p, por ejemplo para p=7 se obtiene una matriz de Hadamard de orden p+1 = 8. Veamoslo:

Partiendo de la matriz Q de Jacobsthal del **Ejemplo 6** (2) construiremos:

$$\begin{pmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 0 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 0 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 0 \end{pmatrix} - \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 \end{pmatrix}.$$

Y así conseguiremos la matriz de orden 8 siguiente:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \end{pmatrix}. \tag{3}$$

Al multiplicarla por su traspuesta, por la **Definición 1**, quedará demostrado que es de Hadamard.:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

$$= 8 \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} = 8 \cdot I.$$

Tal y como habíamos dicho.
Esto lo he comprobado con la siguiente función de Matlab de manera computacional:

```
function [ifallo,M]=matriz(p)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculará Matrices, cuadradas de orden p+1
%ENTRADA:
%p= primo impar
%SALIDA:
%M= Matriz
%ifallo= 1, si la matriz es de Hadamard
%        0, si la matriz no es de Hadamard
%Uso de la funcion [Q] = jacobsthal(p), que me devuelve un matriz Q de Jacobsthal.
%Y [ifallo]=hadamard(H), que me dirá si es de Hadamard la matriz M resultante
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
M=[];
[Q] = jacobsthal(p);
M1=Q-eye(p);
uno=ones(p,1);
M=[1 ones(1,p);
    uno M1];

[ifallo]=hadamard(M);
end
```

Que nos devolverá lo que ya sabíamos:

```
p=7;
[ifallo,M]=matriz(p)
```

```
ommand Window
lew to MATLAB? See resources for Getting Started.

La matriz es de Hadamard
ifallo =

     1


M =

     1     1     1     1     1     1     1     1
     1    -1     1     1    -1     1    -1    -1
     1    -1    -1     1     1    -1     1    -1
     1    -1    -1    -1     1     1    -1     1
     1     1    -1    -1    -1     1     1    -1
     1    -1     1    -1    -1    -1     1     1
     1     1    -1     1    -1    -1    -1     1
     1     1     1    -1     1    -1    -1    -1
```

Sin embargo, no siempre se verifica, en caso de p=5, p+1=6 no es múltiplo de 4, por lo que no obtendríamos una matriz de Hadamard de este orden, ya que entraría en contradicción con el **Teorema 1**.
Lo comprobaremos con el programa de Matlab:

```
p=5;
[ifallo,M]=matriz(p)
```

```
ommand Window
lew to MATLAB? See resources for Getting Started.

La matriz no es de Hadamard
ifallo =

     0


M =

     1     1     1     1     1     1
     1    -1     1    -1    -1     1
     1     1    -1     1    -1    -1
     1    -1     1    -1     1    -1
     1    -1    -1     1    -1     1
     1     1    -1    -1     1    -1
```

Una vez aclarado esto, observamos que el artículo [8] sigue con:
" *Todo esto es también cierto para cualquier potencia de p siendo p un número primo.*".

Esta frase, al igual que la anterior, también puede confundir al lector, haciendo creer que esto es cierto siempre, pero de nuevo, no es así.

Enunciaremos un teorema que indicará los casos en los que la afirmación será cierta, véase [10].

Teorema 2:
Sea p^k una potencia entera positiva de un primo p, se tiene:

1. Si $p^k + 1 \equiv 0 \pmod 4$, *hay una matriz de Hadamard de orden* $p^k + 1$.

2. Si $p^k + 1 \equiv 2 \pmod 4$, *hay una matriz de Hadamard de orden* $2(p^k + 1)$.

Demostración:
Me he basado en la demostración de [10], pero la he completado, ya que

simplemente eran algunas indicaciones.

1. Partiendo de p^k, una potencia entera positiva de un primo p, definimos una matriz Q, de Jacobsthal de orden p^k, cuya entrada (i,j), con i,j entre 1 y p^k, corresponderá con $\chi(i-j)$, siendo χ similar a la de la **Definición 6**: $\chi(k) =$

$$\chi(i) = 0, \qquad si\ i \equiv 0 \pmod{p^k}$$
$$\chi(i) = 1, \qquad si\ i \pmod{p^k}\ es\ residuo.$$
$$\chi(i) = -1, \qquad si\ i \pmod{p^k}\ no\ es\ residuo.$$

Una vez definida Q, podremos construir una matriz H de Hadamard de la siguiente manera **H=I + S**, siendo S:

$$\begin{pmatrix} 0 & 1 \\ -1 & Q \end{pmatrix}.$$

Vamos a ver que la matriz obtenida realmente es de Hadamard aplicando la **Definición 1**:

$$HH' = \left(I + \begin{pmatrix} 0 & 1 \\ -1 & Q \end{pmatrix}\right) \cdot \left(I + \begin{pmatrix} 0 & 1 \\ -1 & Q \end{pmatrix}\right)'.$$

Llamaremos **1** al vector fila de dimensión p^k de unos y U a la matriz cuadrada de dimensión p^k llena de unos.

$$HH' = \begin{pmatrix} 1 & \mathbf{1} \\ -\mathbf{1}' & Q+I \end{pmatrix} \cdot \begin{pmatrix} 1 & -\mathbf{1} \\ \mathbf{1}' & Q'+I \end{pmatrix} =$$

$$= \begin{pmatrix} p^k+1 & -\mathbf{1}+\mathbf{1}(I+Q') \\ -\mathbf{1}'+(I+Q)\mathbf{1}' & U+(I+Q)(I+Q') \end{pmatrix}.$$

Para demostrar que H es de Hadamard, faltaría ver que:

- $-\mathbf{1}+\mathbf{1}(I+Q')$ es cero:
  Para ello, basta con demostrar $Q \cdot \mathbf{1}' = 0$.
  Construyendo Q con la **Definición 7**, aplicando la definición de χ dada en esta demostración, sabemos que debido al **Corolario 2**, tendrá el mismo número de residuos cuadráticos que de no residuos en cada fila/columna, es decir, tiene tantos 1's como -1's. Por lo que al multiplicar cada fila por el vector **1**, quedará un sumatorio con 1's y -1's, que se anularán unos con otros quedando el sumatorio igual a cero. La diagonal queda llena de ceros.

- $U + (I + Q)(I + Q')$ es $p^k + 1$ veces la identidad:

$$U + (I + Q)(I + Q') = U + I + Q + Q' + QQ' = (p^k + 1) \cdot I.$$

Para ello habrá que ver que:
a) $Q + Q' = 0$, lo cual queda demostrado por la construcción de Q, recordada en las líneas anteriores.

b) Y que

$$QQ' = \begin{pmatrix} p^k - 1 & -1 & -1 & \dots & -1 \\ -1 & p^k - 1 & -1 & \dots & -1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -1 & \dots & -1 & -1 & p^k - 1 \end{pmatrix}.$$

Para demostrarlo, me apoyaré de [5].
Sea la función χ ya definida, una función multiplicativa, es decir que cumple las propiedades: $\chi(1) = 1$ y $\chi(st) = \chi(s)\chi(t)$ con s y t pertenecientes al cuerpo de los enteros módulo p^k,p primo impar y k entero positivo, $\mathbb{F}_{p^k}$.
Tomaremos $b, c \in \mathbb{F}_{p^k}$ con $c \neq 0$.
Vemos que para los elementos de la diagonal se verifica:

$$\sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(b)\chi(b) = \sum_{b \in \mathbb{F}_{p^k}, b \neq 0} 1 = p^k - 1.$$

Para los elementos distintos de la diagonal:

$$\sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(b)\chi(b+1) = \sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(b)\chi(b)\chi(1 + \frac{1}{b}) =$$

$$= \sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(b)^2\chi(1 + \frac{1}{b}) = \sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(1 + \frac{1}{b}).$$

Por ser $\mathbb{F}_{p^k}$ un cuerpo finito de orden p^k, todo elemento tendrá inverso y será único.
Por otro lado tenemos que:

$$\frac{1}{b} = \left\{ b \in \left\{ \mathbb{F}_{p^k}, b \neq 0 \right\} \right\} \equiv \mathbb{F}_{p^k} \setminus \{0\}$$

$$1 + \frac{1}{b} = \left\{ 1 + x, \text{ tal que }, x \in \mathbb{F}_{p^k} \setminus \{0\} \right\} \equiv \left\{ \mathbb{F}_{p^k} : x \neq 1 \right\}.$$

Aplicando esto veremos que:

$$\sum_{b \in \mathbb{F}_{p^k}, b \neq 0} \chi(1 + \frac{1}{b}) = \sum_{a \in \mathbb{F}_{p^k}} \chi(a) - \chi(1) = -1.$$

Por lo que tenemos ya $U+QQ' = p^k I$ y con ello conseguimos lo buscado:

$$U + (I + Q)(I + Q') = (p^k + 1)I.$$

Demostrado esto, habremos llegado a lo que queríamos:

$$HH' = \begin{pmatrix} p^k + 1 & -\mathbf{1} + (I + Q')\mathbf{1} \\ -\mathbf{1}' + (I + Q)\mathbf{1}' & U + (I + Q)(I + Q') \end{pmatrix} = \begin{pmatrix} p^k + 1 & 0 \\ 0 & p^k + 1 \end{pmatrix} = (p^k+1)\cdot I.$$

Obteniendo así que la matriz es de Hadamard.

2. Construyendo la matriz Q de la misma manera, definiremos S, con $\mathbf{1}$ un vector fila de unos de dimensión p^k:

$$\begin{pmatrix} 0 & \mathbf{1} \\ \mathbf{1}' & Q \end{pmatrix}.$$

En este caso la matriz H de Hadamard será definida cómo:

$$\mathrm{H{=}S} \otimes H_2 + \mathrm{I} \otimes - H_2.$$

Siendo H_2 la definida en el **Ejemplo 1**.
Para ello, primero desarrollaremos los productos de Kronecker:

$$\begin{pmatrix} 0 & 0 & 1 & -1 & 1 & -1 & \dots & 1 & -1 \\ 0 & 0 & 1 & 1 & 1 & 1 & \dots & 1 & 1 \\ 1 & -1 & 0 & 0 & & & & & \\ 1 & 1 & 0 & 0 & & & \pm H_2 & & \\ 1 & -1 & & & \ddots & & & & \\ 1 & 1 & & & & \ddots & & & \\ \vdots & \vdots & & & & & & & \\ \vdots & \vdots & & & & & & & \\ 1 & -1 & & \pm H_2 & & & & 0 & 0 \\ 1 & 1 & & & & & & 0 & 0 \end{pmatrix} + \begin{pmatrix} -H_2 & 0 & \mathbb{F}_{p^k} & & & & 0 \\ 0 & -H_2 & & & & & 0 \\ & & & \ddots & & & \\ & & & & \ddots & & \\ & & & & & \ddots & \\ & & & & & & \ddots \\ & & & & & & \\ 0 & 0 & & & & & -H_2 \end{pmatrix} =$$

$$= \begin{pmatrix} -H_2 & H_2 & \dots & H_2 & H_2 \\ H_2 & -H_2 & & \pm H_2 & \\ \vdots & & \ddots & & \\ H_2 & \pm H_2 & & -H_2 & \\ H_2 & & & & -H_2 \end{pmatrix}.$$

Ahora hay que ver que al multiplicarlo por su traspuesta es $p^k + 1$ veces la

identidad como en el apartado anterior:

$$\begin{pmatrix} -H_2 & H_2 & \dots & H_2 & H_2 \\ H_2 & -H_2 & & \pm H_2 & \\ \vdots & & \ddots & & \\ H_2 & \pm H_2 & & -H_2 & \\ H_2 & & & & -H_2 \end{pmatrix} \cdot \begin{pmatrix} -H_2 & H_2 & \dots & H_2 & H_2 \\ H_2 & -H_2 & & \pm H_2 & \\ \vdots & & \ddots & & \\ H_2 & \pm H_2 & & -H_2 & \\ H_2 & & & & -H_2 \end{pmatrix}' = (p^k + 1)I$$

Tenemos que, por ser H_2 matriz de Hadamard, cumple la **Definición 1**, $H_2 H_2' = 2I$. Además, por construcción de Q, las filas y columnas estarán formadas por tantas matrices H_2 como $-H_2$.

Una vez demostrado esto, podremos decir que los elementos distintos de la diagonal serán nulos por anularse unas identidades con sus opuestas en un sumatorio con el mismo número de cada una de ellas.
Por otro lado, en los elementos diagonales, al multiplicar la matriz por su traspuesta, quedará $p^k + 1$ veces la identidad.

Con estas anotaciones habremos demostrado que la multiplicación de H por su traspuesta será $p^k + 1$ veces la identidad, por lo que efectivamente, será de Hadamard como queríamos demostrar. □

Volviendo al hilo de lo visto anteriormente, con los métodos ya explicados podremos conseguir matrices de los siguientes órdenes [9]:

- Sylvester: 1, 2, 4, 8, 16, 32, 64, ..., 2^n.
- Paley: 12, 20, 28, 44, 60, ..., $(p^k + 1)$, para algunos primos p y algunos $k \in \mathbb{N}$.
- Kronecker: 24, 40, 48, 56, 72, ..., $A \otimes B$, en algunas ocasiones.

Ver **Anexo 1** para una tabla completa incluyendo estos y otros casos.

El documento en el que me he basado para hacer este trabajo [8] menciona que se desconocen matrices de Hadamard de orden 268, pero actualmente se confirma su existencia mediante el Método de Williamson (método de construcción de matrices de Hadamard que no se explicará aquí). Hasta el año 2005 la lista de matrices de Hadamard de orden múltiplo de 4 menores de 1000 pendientes por descubrir era {428, 668, 716, 764 y 892 }. Fue en ese año cuando Khagharani y Tayfeh-Rezaie hallaron una matriz de dichas características de orden 428. Lo consiguieron utilizando el Teorema de Goethals-Seidel y secuencias tipo Turyn (de las que no se hablará en este trabajo).
En el año 2008 Đokovic redujo aún más esta lista descubriendo una matriz de orden 764, haciendo uso también del Teorema de Goethals-Seidel, pero

esta vez utilizando el método de conjuntos diferencia suplementarios (tampoco se especificará), véase [9].

En resumen, la lista de órdenes múltiplos de 4 menores de 1000 para los cuales aún no se conoce una matriz de Hadamard del orden correspondiente es:

$$\{668, 716 \text{ y } 892\}.$$

Siguiendo con las cuestiones, mencionadas a lo largo del trabajo, nos surge un nuevo problema interesante, ¿cuántas matrices de orden n pueden ser construidas?.

Por la **Definición 2** sabemos que dos matrices son equivalentes si se puede pasar de una a otra por intercambio de filas o columnas, o multiplicando filas y columnas por -1. [1]

Se puede probar que solo hay una clase de equivalencia para orden 1, 2, 4, 8 y 12. Las matrices de esos órdenes son únicas (salvo isomorfismos). Sin embargo, hay cinco clases de orden 16 y tres de orden 20 y aún no se conoce el número de clases de equivalencia de orden 24.

4. Determinantes maximales.

Una motivación para estudiar las matrices de Hadamard, surge de dar respuesta al siguiente problema inicial:

Cuestión 1:
Sea $A = (a_{ij})$ *una matriz de orden n con entradas reales tales que* $|a_{ij}| \leq 1$.
¿Cuál es el valor máximo posible de $\det(A)$*?*.
O lo que es lo mismo, tratar de resolver el siguiente problema de optimización:

$$\text{Calcular} \max_{|a_{ij}| \leq 1} |\det(A)|.$$

Primero veamos que la restricción $|a_{ij}| \leq 1$ de hecho se puede restringir por $|a_{ij}| = 1$:
Definiendo A una matriz con entradas [-1,1] tal que:

$$|\det(A)| = \max_{b_{ij} \in [-1,1]} |\det(B)|$$

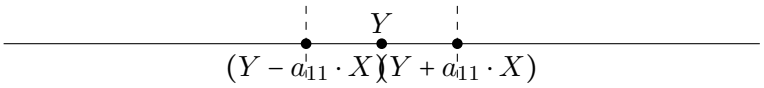
Por reducción al absurdo, suponemos que $|a_{11}| < 1$, es decir, $a_{11} \in (-1, 1)$.
Calculando el valor del determinante de A, desarrollándolo por la primera fila, obtendremos:

$$|\det(A)| = |a_{11} \cdot \det(A^1) + a_{21} \cdot \det(A^2) + \ldots + a_{n1} \cdot \det(A^n)|.$$

Siendo A^r menores de A con $r \in \{1, \ldots, n\}$. Lo podremos expresar de la siguiente forma:

$$|\det(A)| = |a_{11} \cdot X + Y|.$$

El término "Y", que hace referencia a $a_{21} \cdot \det(A^2) + \ldots + a_{n1} \cdot \det(A^n)$, no depende de a_{11}, por lo que al sumarle "$a_{11} \cdot X$", se desplazará sobre la recta real hacia la izquierda si es negativo o la derecha si es positivo:



No conseguiremos maximizar este valor, ya que solo se llegará a alcanzar en el caso de que:

$$a_{11} = \begin{cases} 1 \\ -1. \end{cases}$$

Habiendo demostrado esto para el caso a_{11}, será igual para cualquier a_{ij}.

Una vez justificado el motivo de la restricción, enunciaremos un resultado que podría resultarnos útil, véase [7]

Teorema 3: Desigualdad de Hadamard.
Si $A \in \mathbb{R}^{n\times n}$ *entonces* $\det(A)$ *es menor o igual que el producto de las normas euclídeas de sus columnas o de sus filas:*

$$\det(A) \leq \prod_{j=1}^{n} \sum_{i=1}^{n} a_{ij}^2$$

Demostración:
Apoyándome en los apuntes de Álgebra conmutativa [4], demostraré el teorema.
Primero recordaremos algunos conceptos para poder realizar correctamente la demostración:

- El Algoritmo de ortogonalización de Gram-Schmidt, que consiste en: Sean $\{A_1, \ldots, A_n\}$ las columnas de la matriz A (base ordenada de $\mathbb{R}^{n\times n}$), se definen los vectores ortogonales de la siguiente manera:

$$\hat{A}_1 = A_1$$
$$\vdots$$
$$\hat{A}_i = A_i - \textstyle\sum_{j=1}^{i-1} \mu_{i,j} A_j \quad \text{con} \quad \mu_{i,j} = \frac{\langle \hat{A}_i, \hat{A}_j \rangle}{\langle \hat{A}_j, \hat{A}_j \rangle}.$$

- Teorema de Gram-Schimdt, que, usando la notación anterior, garantiza que se cumple:

$$i) \|\hat{A}_i\| \leq \|A_i\|$$
$$ii) \langle \hat{A}_i, \hat{A}_j \rangle = 0 \quad \text{para todo i} \neq \text{j.}$$

Una vez recordado esto, procedemos con la demostración de la Desigualdad de Hadamard:

Sea A una matriz cualquiera en $\mathbb{R}^{n\times n}$ con determinante no nulo, existe una matriz P triangular superior con 1's en la diagonal principal, tal que $AP = \hat{A}$, cuyas columnas son la base ortogonal calculada mediante el Algoritmo de Gram-Schmidt: $\{\hat{A}_1, \ldots, \hat{A}_n\}$.

Tomando A^t como la traspuesta y usando $\det(P) = 1$, tenemos que:

$$|\det(A)|^2 = \det(AA^t) = \det(A)\det(A^t)\det(P)\det(P^t) =$$
$$= \det(APP^tA^t) = \det(\hat{A}\hat{A}^t)$$

Como conocemos las columnas de $\hat{A}$, trabajaremos con ellas para que nos resulten más sencillos los cálculos:

$$|\det(\hat{A}\hat{A}^t)| = \det \begin{pmatrix} \langle \hat{A}_1, \hat{A}_1 \rangle & \ldots & \langle \hat{A}_1, \hat{A}_n \rangle \\ \vdots & \ddots & \vdots \\ \langle \hat{A}_n, \hat{A}_1 \rangle & \ldots & \langle \hat{A}_n, \hat{A}_n \rangle \end{pmatrix}.$$

Por las propiedades del Algortimo de Gram-Schmidt se tiene que:
$\langle \hat{A}_i, \hat{A}_j \rangle = 0 \quad$ *para todo* $i \neq j$ y que $\|\hat{A}_i\| \leq \|A_i\|$, por lo que obtendremos:

$$|\det(A)|^2 = |\det(\hat{A})|^2 = \prod_{i=1}^{n} \|\hat{A}_i\|^2 \leq \prod_{i=1}^{n} \|A_i\|^2.$$

Como queríamos demostrar. □

Respuesta Cuestión 1:
Sean $A_1, A_2, \ldots, A_n$ las columnas de A. Por **Teorema 3** sabemos que:

$$\det(A) \leq \prod_{j=1}^{n} \|A_j\|$$

Desarrollando esa definición con $\|A_j\| = \sqrt{\sum_{i=1}^{n} a_{ij}^2}$ obtenemos:

$$\det(A) \leq \prod_{j=1}^{n} \|A_j\| = \prod_{j=1}^{n} \sqrt{\sum_{i=1}^{n} a_{ij}^2} = \sqrt{\prod_{j=1}^{n} \sum_{i=1}^{n} a_{ij}^2}.$$

Como $|a_{ij}| \leq 1 \implies a_{ij}^2 \leq 1$, entonces:

$$\det(A) \leq \sqrt{\prod_{j=1}^{n} \sum_{i=1}^{n} 1} = \sqrt{\prod_{j=1}^{n} n} = \sqrt{n^n} \implies \det(A) \leq n^{n/2}. \tag{4}$$

Por todo ello, el máximo valor que puede llegar a tomar el determinante con estas restricciones será $n^{n/2}$. □

Pero, ¿realmente se alcanza ese valor?. Veamoslo en la siguiente proposición.

Proposición 2:
En la desigualdad (4):

$$\det(A) \leq n^{n/2},$$

se da la igualdad si y solo si A es una matriz de Hadamard.
Demostración:
$\Longleftarrow$
Sabemos que:

1. Si A es matriz de Hadamard, verifica la **Definición 1**: $AA^t = nI$.
2. Por propiedades de los determinantes tenemos: $\det(A) = \det(A^t)$.

Usando esto y lo visto en la **Cuestión 1** obtenemos:

$$AA^t = nI \implies \det(A)^2 = \det(A) \cdot \det(A^t) = \det(AA^t) = \det(nI) = n^n$$

De este modo vemos que el determinante de una matriz de Hadamard es maximal.
$\Longrightarrow$
Si partimos de la igualdad en la Desigualdad de Hadamard, siguiendo los pasos de la demostración de la **respuesta de la cuestión 1**, vemos que todas las desigualdades han de ser igualdades. Por lo tanto ha de cumplirse:

$$\det(A) = \prod_{j=1}^{n} \|A_j\|.$$

Esto es, a la hora de aplicar Gram-Schmidt los vectores resultantes han de ser los propios vectores dados. Esto ocurre únicamente si son ortogonales. Así obtenemos la primera propiedad de las matrices de Hadamard.
Una vez vista esta propiedad, en la **respuesta de la cuestión 1** encontramos otra desigualdad, la (4). Para que esta cumpla la igualdad, se tendrá que dar: $a_{ij}^2 = 1$, que es la única forma en que se puede cumplir $\|A_j\| = n$. Así habremos obtenido la segunda propiedad de las matrices de Hadamard, lo que termina la demostración. □

Podemos concluir que si no existe matriz de Hadamard de un determinado orden n, el mayor valor del determinante será estrictamente menor que $n^{n/2}$ (aun no se conoce la fórmula general de estos casos).

Ejemplo 7:
Sabiendo que los primeros órdenes para los que conocemos matrices de Hadamard son $\{1, 2, 4, 8, \dots\}$ (Construcción de Sylvester), muestro un ejemplo del máximo valor de los determinantes de matrices de dichos órdenes, teniendo en cuenta lo que acabamos de demostrar, es decir, que en caso de existir matriz de Hadamard de dicho órden, el valor máximo de esos determinantes será el de la matriz de Hadamard, $n^{n/2}$:

n=1	n=2	n=3	n=4	n=5	n=6
$1^{1/2} = 1$	$2^1 = 2$	$3^{3/2} < 5{,}19$	$4^{4/2} = 16$	$5^{5/2} < 55{,}90$	$6^{6/2} < 216$

n=7	n=8	n=9
$7^{7/2} < 907{,}49$	$8^{8/2} = 4096$	$9^{9/2} < 19683$

Órdenes:	1	2	3	4	5	6	7	8	9	...
Valor det.:	1	2	4	16	48	160	576	4096	14336	...

5. Teoría de códigos.

Las matrices Hadamard tienen diversas aplicaciones y una de las más interesantes es a la teoría de códigos.

Esta rama surgió a principios del siglo pasado, teniendo su origen en problemas de Ingeniería y se ha desarrollado usando métodos matemáticos cada vez más sofisticados.

Hagamos primero una introducción en el tema:

La información se transmite a través de un canal, por lo que debe ser codificada, es decir, debe ser escrita en un cierto alfabeto (un número finito de símbolos concretos). En nuestro caso usaremos codificación binaria, con {0,1}.
El mensaje es enviado por el emisor y recibido por el receptor. Pero antes de poder leerlo, el mensaje tendrá que haber sido decodificado para que llegue en su forma original y el receptor pueda entenderlo.

La teoría de códigos se inventó para poder corregir los posibles errores en canales de comunicación ocasionados por el ruido.
Cuando enviamos mensajes, puede que estos lleguen correctamente o que en lugar de recibir un 0 cuando me envian 0, o un 1 cuando se envía un 1, debido a factores externos el 0 se reciba como 1 o viceversa.
El receptor ha de estar advertido de la posible existencia de estos errores para poder corregirlos y recibir el mensaje correcto.
Se dice que un código detecta un error si al transmitir una palabra del código, la palabra recibida no pertenece al código, aunque puede ocurrir que se produzcan varios errores y desencadene en una palabra del código, véase [2].
Una estrategia frecuente es 'alargar' el mensaje introduciendo signos de control para poder identificar y corregir los errores recibidos.

Una vez explicado esto, procedamos con la teoría. Definamos qué es un código:

Definición 8: Código.
Se llama código C a un conjunto finito de M palabras de tamaño n, en un alfabeto finito (en nuestro caso usaremos $\{0, 1\}$*).*

Definición 9:([2])
Un código es un buen código, si permite corregir la mayor cantidad de errores posibles, introduciendo la menor cantidad de símbolos redundantes.

Ejemplo 8:
El ejemplo más usual para entender esto es nuestro DNI, formado por 8 números y una letra, como variable de control al final de ellos.
Lo que se busca con la letra es que si al introducir los números se produce un error, la letra indique que hay un error entre ellos.
Esta letra se calculará sumando los números y haciendo al valor obtenido módulo 23. El valor del resto obtenido indicará la letra correspondiente a este y será la que se encuentre al final de los números.
En el caso de que haya dos o más errores, lo cual es altamente improbable, será más complicado detectar el error, ya que puede existir casualmente un DNI con esa misma serie de números.

Los códigos más sencillos son los lineales.

Para ello primero desarrollaremos un ejemplo del documento [8].
Ejemplo 9:
Supongamos que queremos mandar una palabra digital de tres letras, u_1, u_2, u_3, con $u_i = \{0,1\}$.
Tomando la siguiente matriz para codificar las palabras:

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix},$$

alargaremos el código para facilitar la busqueda de errores de la siguiente forma:

$$(u_1, u_2, u_3) \to \begin{pmatrix} I \\ A \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \quad (\text{mód } 2).$$

Tendremos que $x_i \forall i \in \{1,2,3,4,5,6\}$ es 0 o 1 y $x_i \forall i \in \{1,2,3\}, x_i = u_i$ por ser I la identidad de orden 3. Obtendremos así la correpondencia entre códigos que queremos enviar y los alargados con tres cifras de control:

$$
\begin{aligned}
000 &\to 000000,\\
100 &\to 100011,\\
010 &\to 010101,\\
001 &\to 001110,\\
110 &\to 110110,\\
101 &\to 101101,\\
011 &\to 011011,\\
111 &\to 111000.
\end{aligned}
$$

Ahora vamos a desarrollar otro ejemplo diferente donde veremos si la elección de la matriz A es determinante para conseguir codificar bien un código.

Ejemplo 10:
Igual que en el **Ejemplo 9**, crearemos un código con 0's y 1's y mandaremos palabras de longitud 3, v_1, v_2, v_3.
Tomaremos la matriz $\hat{A}$ para codificar la palabras:

$$
\hat{A} = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}
$$

Quedando las palabras codificadas de tamaño 6:

$$
(v_1, v_2, v_3) \to \begin{pmatrix} I \\ \hat{A} \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} \quad (\text{mód } 2).
$$

Tendremos, al igual que antes, que $y_i \forall i \in \{1,2,3,4,5,6\}$ es 0 o 1 y $y_i \forall i \in \{1,2,3\}, y_i = v_i$ por ser I la identidad de orden 3. Las palabras alargadas con tres cifras de control quedarán así:

$$
\begin{aligned}
000 &\to 000000,\\
100 &\to 100110,\\
010 &\to 010011,\\
001 &\to 001111,\\
110 &\to 110101,\\
101 &\to 011100,\\
011 &\to 101001,\\
111 &\to 111010.
\end{aligned}
$$

Observemos lo obtenido en el **Ejemplo 9** comparándolo con el **Ejemplo 10**: vemos que los tres últimos dígitos en ambas ocasiones serán las cifras de control. En el primero de los ejemplos, estas se repetirán en el caso de que las tres primeras cifras sean todas diferentes. Con esto podremos detectar si hay uno o dos errores, ya que si varía cualquiera de las tres primeras cifras, la secuencia final no coincidirá.

Sin embargo, en el **Ejemplo 10**, las cifras de control son diferentes en todas las palabras resultantes al alargarlas. Con esto podríamos pensar que será más fácil detectar errores, ya que si hay un error entre las tres primeras, ninguna de las opciones de las cifras de control coincidirán con ese error. Podríamos detectar hasta dos o tres errores. Si el error estuviera entre las cifras de control, pasaría lo mismo. En caso de que este esté en ambas partes, sería posible detectar dos errores, pero tres sería complicado, ya que podría coincidir con otra palabra del código.

Una vez detectada la existencia de un error, hay que corregirlo para poder regenerar la palabra original del mensaje.

En el **Ejemplo 9**, en el caso de que recibiésemos la palabra 101000, por ejemplo, seríamos capaces de detectar que hay un error ya que las únicas opciones con los tres dígitos de control 000 son: 000000 y 111000, que no son coincidentes con la palabra recibida.

En el **Ejemplo 10**, si recibiésemos esa misma palabra, 101000, también seríamos capaces de detectar la existencia del error. Las cifras de control 000 solo se encuentran en la palabra 000000, por lo que barajando la opción de que el error esté entre estas, la única palabra del código que tiene las tres primeras cifras coincidentes es 101001.

Para decidir qué código es en realidad mejor, necesitamos sistematizar la forma de resolver de estos ejemplos. Para ello definiremos el siguiente concepto:

Definición 10: Distancia de Hamming. [2]
Sean $x,y \in \{0,1\}^n$, $x = (x_1, x_2, \ldots, x_n)$ *e* $y = (y_1, y_2, \ldots, y_n)$. *Se llama distancia de Hamming entre* x *e* y *a la cantidad:*

$$d(x,y) = \#\{i : 1 \leq i \leq n, x_i \neq y_i\}.$$

Me indicará el número de cifras diferentes entre dos palabras del código.

Mediante esta distancia, podremos calcular la distancia mínima entre dos palabras. El procedimiento consiste en que dada una palabra, perteneciente al código o no, la palabra a menor distancia de esta será nuestra candidata

elegida para la palabra enviada.

Definición 11: ([2])
Sea C un código, se llama distancia mínima de C a:

$$d = d(C) = \text{mín}\{d(x, y) : x, y \in C, x \neq y\}.$$

Observación:

1. Este procedimiento falla cuando la distancia mínima no es única, es decir, hay más de una palabra en el código con la distancia mínima a la recibida.
2. Para la reconstrucción de palabras es importante que *la distancia de Hamming* (**Definición 10**) sea la mayor posible.

En general se tiene que para cada código, (**Definición 8**) hay tres factores que lo definen, lo denominaremos un (n,M,d) código.

- n será la longitud del código, o lo que es lo mismo, dimensión de las palabras.
- M es el número de palabras que forman el código.
- d la distancia mínima entre dos vectores cualesquiera.

Lo más eficiente sería que n tome un valor pequeño para que el código ocupe lo menos posible y M y d, el mayor valor posible.

En el caso del **Ejemplo 9**, sería un código (6,8,3).
En el **Ejemplo 10**, también será un código (6,8,3).

Teorema 4:([2])
Sea C un código con distancia mínima d, entonces:

- *Detecta s errores si y solo si* $s < d$.
- *Puede corregir* $[\frac{d-1}{2}]$ *errores, es decir, corrige t errores si y solo si* $2t < d$. *(siendo* $[\frac{d-1}{2}]$ *la parte entera de la fracción).*

Demostración:
Siguiendo las líneas de la demostración de [2], partiremos de una palabra del código, $c \in C$, siendo $\mathbf{y}$ la palabra que se recibe:

- Si $\mathbf{y}$ tiene s errores con $0 < s < d$, entonces $d(\mathbf{y}, c) = s$. Como la distancia mínima es d, tendremos que $\mathbf{y}$ no pertenece al código ya que cualquier palabra de C estará al menos a distancia d de c. Por lo que se podrán detectar s errores siempre que $s < d$.

- Si **y** tiene t errores con $2t < d$, entonces $d(\mathbf{y}, c) < d(\mathbf{y}, x) \forall x \in C$. Esto quiere decir que, si tomamos bolas con centro en una palabra del código, para que estas sean disjuntas y las palabras estén a una distancia mayor o igual que d, el radio de estas deberá de ser $\frac{d-1}{2}$. Y tendremos que como $2t < d \Leftrightarrow t < d/2$, **y** solo podrá pertenecer a una de las bolas, por lo que tendrá una única palabra del código a distancia mínima, que será el centro de dicha bola. Concluyendo así, que el código C corrige t errores si y solo si $2t < d$. □

Si volvemos al **Ejemplo 9**, podremos ver que se podrá corregir $\frac{3-1}{2} = 1$ error. Ya que la distancia mínima entre las palabras es 3.
Lo mismo ocurre en el **Ejemplo 10**, por ser su distancia mínima también 3.
Concluimos que ninguno de estos dos códigos es ni mejor ni peor que el otro.

Daremos una cota que se ajustará lo mejor posible a lo que buscamos:

Teorema 5: Cota de Plotkin.
Sea (n,M,d) un código C, para el cual $n < 2d$, se tiene:

$$M \leq 2\left[\frac{d}{2d-n}\right].$$

Demostración:
Primero detallaremos la demostración en el caso de M ser par (como se indica en [8]).
Calcularemos $S = \sum_{u,v \in C} d(u,v)$ de dos maneras:

a) Como M es el número de palabras del código, hay M(M-1) parejas y la distancia entre ellas sabemos que es al menos d. Con esto obtendremos que:

$$S \geq M(M-1)d.$$

b) Si ponemos las M palabras en filas que denotaremos f_i, construiremos una matriz $M \times n$, en la cual habrá x_i 0's y $(M - x_i)$ 1's en la columna i-ésima.
Partiendo de esto obtendremos:

$$\sum_{i,j} d(f_i, f_j) = \sum_{k=1,2,\ldots,n(\text{las columnas})} \sum_{i,j} d(f_i^k, f_j^k).$$

Denotamos v al vector k-ésimo de esos vectores columna y nos queda:

$$\sum_{i,j} d(v_i, v_j) = \sum_{i,j} \begin{cases} 1 \text{ si } v_i \neq v_j \\ 0 \text{ si } v_i = v_j \end{cases}$$

Como el orden en los vectores no influye en el sumatorio, podemos ordenar sus términos poniendo primero todos los 1's y luego todos los 0's. Sabiendo que hay x_i 0's y $(M - x_i)$ 1's en la columna i-ésima, así resultará la siguiente ecuación:

$$\sum_{i,j} d(v_i, v_j) = \sum_{i=1}^{M-x_k} \sum_j d(1, v_j) + \sum_{i=M-x_k+1}^{M} \sum_j d(0, v_j).$$

La desarrollamos sabiendo:

$$\sum_j d(1, v_j) = x_k \quad \text{y} \quad \sum_j d(0, v_j) = M - x_k$$

y así obtendremos:

$$\sum_{i,j} d(v_i, v_j) = (M - x_k) \sum_j d(1, v_j) + (x_k) \sum_j d(0, v_j) = (M - x_k)(x_k) + x_k(M - x_k).$$

Por lo tanto, en la columna i-ésima quedará:

$$S = \sum_{i=1}^{n} 2x_i(M - x_i).$$

Para poder acotarla superiormente, calcularemos el máximo. Para ello calculamos la derivada y la igualamos a cero, por lo que queda: $2M = 4x_i \implies x_i = M/2$ el valor donde se alcanza el máximo de la función. Asi que quedaría acotado superiormente de la siguiente forma:

$$S \leq \frac{nM^2}{2}.$$

Dadas estas dos desigualdades, obtendremos:

$$M(M-1)d \leq \frac{nM^2}{2} \implies M^2 d - Md - \frac{M^2 n}{2} \leq 0$$

lo que implica:

$$M = 0 \text{ ó } M \leq \frac{2d}{2d-n}.$$

Tomando $M \leq \frac{2d}{2d-n}$:

$$M \leq \frac{2d}{2d-n} \implies M \leq 2\left(\frac{d}{2d-n}\right)$$

como queríamos demostrar.

Ahora veremos el caso de que M tome un valor impar:

Al igual que para el caso de M par, calcularemos de dos formas diferentes S, que es el sumatorio de las distancias de las palabras del código, como se ha indicado arriba.
La opción a) será similar en este caso ya que el número de parejas seguirá siendo M(M-1). Obtendremos así:

$$S \geq M(M-1)d.$$

En el caso b), veremos como afecta este cambio.
De la misma forma que en el caso anterior, tendremos:

$$S = \sum_{i=1}^{n} 2x_i(M - x_i)$$

al construir la matriz de dimensiones $M \times n$, con x_i 0's y $(M - x_i)$ 1's en la columna i-ésima, siendo las filas de esta, las M palabras.
Para calcular su valor máximo para acotar S superiormente, calcularemos la derivada y lo igualaremos a cero como antes. Así resultará: $x_i = M/2$.
En este caso M es impar, por lo que M/2, y por ello x_i, no será un número entero positivo, y esto no puede ocurrir porque x_i es el número de 0's que hay en la matriz, por lo que tiene que ser un valor entero.
Para corregir esto cogeremos la parte entera del valor M/2, resultando así:

$$S \leq [\frac{nM^2}{2}],$$

siendo en este caso $[\frac{nM^2}{2}]$ la parte entera de la división.
Desarrollando al desigualdad $M(M-1)d \leq [\frac{nM^2}{2}]$, quedará:

$$M(M-1)d \leq [\frac{nM^2}{2}] \implies [M^2d - Md - \frac{M^2n}{2}] \leq 0$$

lo que implica:

$$M = 0 \text{ ó } M \leq \frac{2d}{2d-n}.$$

Tomando $M \leq \frac{2d}{2d-n}$:

$$M \leq \frac{2d}{2d-n} \implies M \leq 2\left[\frac{d}{2d-n}\right].$$

A diferencia del caso M par, donde la cota quedaba:

$$M \leq \frac{2d}{2d-n} \implies M \leq 2\left(\frac{d}{2d-n}\right),$$

siendo ya un número entero, por lo que si cogemos la parte entera de la fracción, el resultado es el mismo.

Habremos demostrado así que la cota para todo M se puede generalizar de la siguiente forma:

$$M \leq \frac{2d}{2d-n} \implies M \leq 2\left[\frac{d}{2d-n}\right].$$

□

Veamos un ejemplo de lo explicado:

Ejemplo 11:
Si tenemos que la longitud de las palabras es 6 y queremos tener una distancia mínima entre ellas de 4, aplicando la formula del teorema anterior vemos que el número máximo de palabras es a lo sumo:

$$M \leq 2\left[\frac{4}{2\cdot 4-6}\right] = 4.$$

Sin embargo, en el **Ejemplo 9**, donde las palabras son de longitud 6 y la distancia mínima entre ellas es 3, el número máximo de palabras en el código no estará acotado superiormente:

$$M \leq 2\left[\frac{3}{2\cdot 3-6}\right] = \infty.$$

Luego *la Cota de Plotkin* en este caso no nos dará información útil.

Dada *la Cota de Plotkin*, es un problema interesante lograr determinar códigos que lleguen a alcanzarla. Con n y d fijados, intentaremos conseguir un código con el máximo número de palabras posibles. Para ello, las desigualdades anteriores deben cumplir la igualdad:

a) Todas las palabras tienen que estar a distancia d.
b) En cada columna debe haber el mismo número de 0's que de 1's.

Una herramienta fundamental para conseguirlo son las matrices de Hadamard, explicadas a lo largo de todo este trabajo.

Teorema 6: Levenshtein.
Si n y d son pares y existen matrices de Hadamard de orden $4[\frac{d}{2d-n}]$ *y* $4[\frac{d}{2d-n}]+4$*, las cotas de Plotkin son alcanzadas.*

Desarrollaremos un ejemplo para ver lo explicado de manera aplicada:

Ejemplo 12: Definamos un código C con palabras de longitud 12 a distancia mínima 8. Con la Cota de Plotkin (**Teorema 5**) podremos calcular el número máximo de palabras para definir un buen código, **Definición 9**.
Este será: $M \leq 2\lfloor \frac{8}{16-12} \rfloor = 4$, por lo que el código sería: (12,4,8).
Para ello consideramos la matriz de Hadamard orden 8 vista anteriormente, en (3):

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \end{pmatrix}.$$

Modificaremos la matriz cambiando los -1's por 1's y los 1's por 0's:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Una vez tengamos esta matriz, para conseguir el código deseado tendremos que seguir los siguientes pasos:
1. Eliminar la primera columna.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

2. Eliminar las filas resultantes que empiecen por uno.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

3. Volver a eliminar la primera columna.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

4. Pegar la matriz resultante consigo misma.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

Así resultará el código (12,4,8), que podrá corregir hasta tres errores:

$$\left[\tfrac{8-1}{2}\right] = 3.$$

No hemos explicado por qué el ejemplo anterior funciona, ni si la construcción puede ser sistematizada. Estas explicaciones podrían ser objeto de una ampliación de este trabajo, aunque nos detendremos aquí. Pero al menos a modo de ejemplo ilustrativo, ahora intentaremos conseguir el mismo código cogiendo una matriz de Hadamard de orden 8, equivalente a la que hemos escogido primeramente y comprobaremos así si esto nos dará el mismo código u otro diferente.

Sea la matriz de Hadamard anterior de orden 8, cambiando:
1) La tercera fila la multiplico por -1.
2) La segunda columna la sustituyo por la cuarta.
3) La septima fila la cambio de signo.
Que por **Definición 2** es equivalente a la anterior:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \end{pmatrix}.$$

Modificamos los 1's por 0's y los -1's por 1's:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Seguiremos el mismo proceso que antes:
1. Eliminamos la primera columna.
2. Eliminamos las filas que empiezan por 1.
3. Eliminamos la primera columna de nuevo.
4. Pegamos la matriz resultante consigo misma.
Resultará la matriz de 2 filas con 6 columnas tras hacer los tres primeros pasos:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Al pegarlo consigo misma en el cuarto paso, obtendremos:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Como podemos observar, el código conseguido es diferente al anterior, por lo que al contrario de lo que vimos en los **Ejempos 9 y 10**, la matriz elegida en este caso es muy importante, ya que resultan códigos distintos según cual se escoja.

De este ejemplo podremos concluir que la mejor matriz a usar es la construida con el método de Paley, usando las matrices de Jacobsthal y no una equivalente a esta. De cada matriz resultará un código diferente.
En este caso hemos obtenido un código (12,2,8), diferente al (12,4,8) que teníamos de referencia y claramente inferior.

Conclusiones

En esta última sección, haremos una conclusión detallando lo que hemos conseguido a lo largo de todo el trabajo.

El objetivo de este trabajo era ver qué son las matrices de Hadamard, sus diferentes propiedades, así como su búsqueda y construcción, para después poder describir de forma somera una de sus aplicaciones.

Primeramente las hemos definido a través de su propiedad principal: $HH' = nI$. Tras aplicar esta definición sobre una matriz equivalente o normalizada, hemos comprobado que realmente siguen cumpliendo la definición y efectivamente son matrices de Hadamard.
Otro factor a destacar que también se ha demostrado, es que si una matriz es de Hadamard, su traspuesta también lo será.

Una vez hemos localizado cuándo una matriz es o no de Hadamard, hemos buscado para qué órdenes existen matrices de este tipo y las hemos construido aplicando tres métodos.
Con el **Teorema de Hadamard**, se observó que todas ellas serán de orden 1,2 o múltiplo de 4 en caso de ser mayor que 2. Pero nos surgió una cuestión: ¿para cualquier múltiplo de 4 existe matriz de Hadamard?, que hemos visto que se denomina *Conjetura de Hadamard*.

Para poder discutir esto hemos construido estas matrices con los siguientes métodos: el producto de Kronecker, la construcción de Sylvester y la construcción de Paley. Con el primero de ellos hemos visto que aplicado a la matriz de Hadamard H_2 siempre resultará otra matriz de Hadamard. Habrá ocasiones donde al operar dos matrices de Hadamard con este producto, resulte otra. En el segundo de los métodos hemos aplicado el producto de Kronecker para construir estas matrices si su orden es potencia de 2. El último de ellos es el más complejo. En este hemos tenido que definir qué son las matrices de Jacobsthal, hemos construido matrices de orden p+1 con p un número primo mayor que dos. Como ayuda para poder crear estas matrices de una forma más eficiente, he creado un programa en Matlab.

A pesar de que en todo el trabajo he tomado el documento [8] como referencia, ha habido dos ocasiones donde este podría crear confusión, por lo que he buscado un contraejemplo donde se contradice la afirmación del documento y así poder completar de manera correcta esta información. He buscado y demostrado con ayuda de distintos documentos cuáles son los teoremas y corolarios que indican en qué casos son ciertas dichas líneas confusas.
Con esto he visto que los únicos órdenes de la forma $p^k + 1$, con p primo im-

par, para los que existen matrices de Hadamard, son los que son congruentes con 0 módulo 4, surgiendo de aquí una matriz de orden $p^k + 1$ y en el caso de ser $p^k + 1$ congruente con 2 módulo 4, que indicarán la existencia de una matriz de Hadamard de orden $2(p^k + 1)$.

Otra propiedad a destacar de estas matrices, que hemos demostrado y desarrollado con detalle, es que su determinante es maximal, es decir, son las únicas en las que se da la igualdad en la *Desigualdad de Hadamard:* $\det(A) \leq \prod_{j=1}^{n} \sum_{i=1}^{n} a_{ij}^2$, siendo a_{ij} la posición i de la columna j.
Hemos desarrollado ejemplos para ver que ocurre con los primeros 9 enteros positivos.

Tras haber encontrado estas matrices y desarrollado todas sus características y particularidades, hemos visto una de sus aplicaciones prácticas, la teoría de códigos, que se ha explicado en la cuarta sección del trabajo.
En esta hemos visto una introducción teórica sobre la teoría de códigos y después hemos desarrollado un ejemplo sobre cómo aplicar aquí las matrices de Hadamard.

La teoría de códigos consiste en codificar mensajes y decodificarlos corrigiendo errores en caso de existir, para que llegue el mismo mensaje al receptor que el que envió el emisor.
Hemos probado que si hacemos pequeñas modificaciones a matrices de Hadamard de orden la distancia mínima entre las palabras del código, estas son una buena opción para construir un código que sea buen código, es decir, si el código es (n,M,d), n tome el menor valor posible y M y d el mayor.
Para ver esto hemos desarrollado diferentes ejemplos, donde hemos visto la importancia de la elección de la matriz.

Así hemos conseguido nuestro objetivo, conocer qué son las matrices de Hadamard e iniciaremos en su utilidad en la teoría de códigos.

Anexos

Anexo 1

Para completar la lista de posibles órdenes de matrices de Hadamard y el modo por el cual se consiguen, en el documento [9] aparece la siguiente lista con esta información que resulta interesante de conocer.
Con lo visto durante este trabajo, hemos justificado la existencia de las matrices de orden t, siempre múltiplo de 4:

- Potencia de dos, por la construcción de Sylvester. Ocurrirá en los órdenes:
  {4, 8, 16, 32, 64, 128, 256, 512 }
- Mediante la construcción de Paley, matrices de orden p+1 para algún p primo. Ocurrirá para los órdenes:
  { 12, 20, 44, 60, 68, 84, 108, 132, 140, 164, 180, 192, 200, 212, 228, 252, 284, 308, 332, 348, 380, 420, 442, 448, 492, 500, 524, 548, 564, 572, 588, 620, 644, 660, 684, 740, 812, 828, 860, 884, 908, 948, 972 }
- Si conocemos una matriz de Hadamard, haciendo producto de Kronecker con H_2 del **Ejemplo 1**, obtendremos otra matriz de Hadamard. Esto ocurre para los órdenes:
  { 24, 40, 56, 72, 88, 104, 120, 136, 152, 168, 184, 232, 248, 264, 280, 296, 344, 376, 392, 632, 648, 664, 680, 696, 712, 728, 744, 760, 776, 792, 808, 824, 840, 856, 872, 888, 904, 920, 936, 952, 968, 984, 1000 }
- Producto de Kronecker:
  { 48, 80, 96, 112, 144, 160, 176, 208, 224, 240, 272, 288, 304, 320, 336, 352, 368, 384, 400, 416, 432, 448,464, 480, 496, 528, 544, 560, 576, 592, 608, 624, 640, 656, 672, 688, 704, 720, 736, 752, 768, 784, 800, 816, 832, 848, 864, 880, 896, 912, 928, 944, 960, 976, 992}
- Matrices con órdenes de la forma $p^k + 1$:
  {28, 244 }.
- Matrices con órdenes de la forma $2(p + 1)$:
  {36, 76, 124, 148, 196, 204, 220, 276, 300, 316, 364, 388, 396, 676, 700, 708, 748, 780, 796, 804, 820, 844, 900, 916, 924}.
- Matrices con órdenes de la forma $2(p^k + 1)$:
  {52, 340, 580 }.

Hasta aquí son los órdenes conseguidos con los métodos de construcción explicados y desarrollados en este documento.
Para el resto de órdenes existen otros métodos que no hemos desarrollado.

- Método de Williamson:
  {92, 100, 116, 156, 172, 268, 292, 460, 484, 516, 540, 556, 628, 636, 732}.
- Método de Goethals-Seidel:
  {188, 236, 260, 356, 404, 428, 764, 956, 980 }.
- Método de Baumert-Hall:
  {372, 612}.
- Método de Enlich:
  {324}.
- Método de Miyamoto:
  {436, 452, 596, 692, 772, 788, 932, 964}.
- Método de Scarpis:
  {756}.
- Método de Wallis:
  {836, 996}.
- Método de Conjuntos Diferencia Suplementarios:
  {412, 508, 604, 652, 724, 852, 868, 876, 940, 988}.
- Método de Cooper-Wallis:
  {476, 532}.

Con esto, habremos resumido los métodos de construcción para todos los posibles órdenes menores de 1000 para los que existen matrices de Hadamard. Como hemos explicado en las secciones 1 y 2, aún hay órdenes múltiplos de 4 menores de 1000 para los que no se conoce matriz de Hadamard.
Estos son: {668, 716, 892}.

$4 = 2^2$	$204 = 2(101+1)$	$404 =$ Goe-Sei	$604 =$ Supl.Di.Sets	$804 = 2(401+1)$
$8 = 2^3$	$208 = 4 \otimes 52$	$408 = 2 \otimes 204$	$608 = 8 \otimes 76$	$808 = 2 \otimes 404$
$12 = 11+1$	$212 = 211+1$	$412 =$ Supl.Di.Sets	$612 =$ Baumert-Hall	$812 = 811+1$
$16 = 2^4$	$216 = 2 \otimes 108$	$416 = 8 \otimes 52$	$616 = 2 \otimes 308$	$816 = 4 \otimes 204$
$20 = 19+1$	$220 = 2(109+1)$	$420 = 419+1$	$620 = 619+1$	$820 = 2(409+1)$
$24 = 2 \otimes 12$	$224 = 8 \otimes 28$	$424 = 2 \otimes 212$	$624 = 4 \otimes 156$	$824 = 2 \otimes 412$
$28 = 3^3+1$	$228 = 227+1$	$428 =$ Goe-Sei	$628 =$ Williamson	$828 = 827+1$
$32 = 2^5$	$232 = 2 \otimes 116$	$432 = 4 \otimes 108$	$632 = 2 \otimes 316$	$832 = 8 \otimes 104$
$36 = 2(17+1)$	$236 =$ Goe-Sei	$436 =$ Miyamoto	$636 =$ Williamson	$836 =$ Wallis
$40 = 2 \otimes 20$	$240 = 4 \otimes 60$	$440 = 2 \otimes 220$	$640 = 20 \otimes 32$	$840 = 2 \otimes 420$
$44 = 43+1$	$244 = 3^5+1$	$444 = 443+1$	$644 = 643+1$	$844 = 2(421+1)$
$48 = 4 \otimes 12$	$248 = 2 \otimes 124$	$448 = 16 \otimes 28$	$648 = 2 \otimes 324$	$848 = 4 \otimes 212$
$52 = 2(5^2+1)$	$252 = 251+1$	$452 =$ Miyamoto	$652 =$ Supl.Di.Sets	$852 =$ Supl.Di.Sets
$56 = 2 \otimes 28$	$256 = 2^8$	$456 = 2 \otimes 228$	$656 = 4 \otimes 164$	$856 = 2 \otimes 428$
$60 = 59+1$	$260 =$ Goe-Sei	$460 =$ Williamson	$660 = 659+1$	$860 = 859+1$
$64 = 2^6$	$264 = 2 \otimes 132$	$464 = 4 \otimes 116$	$664 = 2 \otimes 332$	$864 = 8 \otimes 108$
$68 = 67+1$	$268 =$ Williamson	$468 = 467+1$	**668** = desconocido	$868 =$ Supl.Di.Sets
$72 = 2 \otimes 36$	$272 = 4 \otimes 68$	$472 = 2 \otimes 236$	$672 = 8 \otimes 84$	$872 = 2 \otimes 436$
$76 = 2(37+1)$	$276 = 2(137+1)$	$476 =$ Cooper-Wallis	$676 = 2(337+1)$	$876 =$ Supl.Di.Sets
$80 = 4 \otimes 20$	$280 = 2 \otimes 140$	$480 = 8 \otimes 60$	$680 = 2 \otimes 340$	$880 = 4 \otimes 220$
$84 = 83+1$	$284 = 283+1$	$484 =$ Williamson	$684 = 683+1$	$884 = 883+1$
$88 = 2 \otimes 44$	$288 = 8 \otimes 36$	$488 = 2 \otimes 244$	$688 = 4 \otimes 172$	$888 = 2 \otimes 444$
$92 =$ Williamson	$292 =$ Williamson	$492 = 491+1$	$692 =$ Miyamoto	**892** = desconocido
$96 = 8 \otimes 12$	$296 = 2 \otimes 148$	$496 = 4 \otimes 124$	$696 = 2 \otimes 348$	$896 = 28 \otimes 32$
$100 =$ Williamson	$300 = 2(149+1)$	$500 = 499+1$	$700 = 2(349+1)$	$900 = 2(449+1)$
$104 = 2 \otimes 52$	$304 = 4 \otimes 76$	$504 = 2 \otimes 252$	$704 = 16 \otimes 44$	$904 = 2 \otimes 452$
$108 = 107+1$	$308 = 307+1$	$508 =$ Supl.Di.Sets	$708 = 2(353+1)$	$908 = 907+1$
$112 = 4 \otimes 28$	$312 = 2 \otimes 156$	$512 = 2^9$	$712 = 2 \otimes 356$	$912 = 4 \otimes 228$
$116 =$ Williamson	$316 = 2(157+1)$	$516 =$ Williamson	**716** = desconocido	$916 = 2(457+1)$
$120 = 2 \otimes 60$	$320 = 16 \otimes 20$	$520 = 2 \otimes 260$	$720 = 4 \otimes 180$	$920 = 2 \otimes 460$
$124 = 2(61+1)$	$324 =$ Enlich	$524 = 523+1$	$724 =$ Supl.Di.Sets	$924 = 2(461+1)$
$128 = 2^7$	$328 = 2 \otimes 164$	$528 = 4 \otimes 132$	$728 = 2 \otimes 364$	$928 = 8 \otimes 116$
$132 = 131+1$	$332 = 331+1$	$532 =$ Cooper-Wallis	$732 =$ Williamson	$932 =$ Miyamoto
$136 = 2 \otimes 68$	$336 = 4 \otimes 84$	$536 = 2 \otimes 268$	$736 = 8 \otimes 92$	$936 = 2 \otimes 468$
$140 = 139+1$	$340 = 2(13^2+1)$	$540 =$ Williamson	$740 = 739+1$	$940 =$ Supl.Di.Sets
$144 = 4 \otimes 36$	$344 = 2 \otimes 172$	$544 = 8 \otimes 68$	$744 = 2 \otimes 186$	$944 = 4 \otimes 236$
$148 = 2(73+1)$	$348 = 347+1$	$548 = 547+1$	$748 = 2(373+1)$	$948 = 947+1$
$152 = 2 \otimes 76$	$352 = 8 \otimes 44$	$552 = 2 \otimes 276$	$752 = 4 \otimes 188$	$952 = 2 \otimes 476$
$156 =$ Williamson	$356 =$ Goe-Sei	$556 =$ Williamson	$756 =$ Scarpis	$956 =$ Goe-Sei
$160 = 8 \otimes 20$	$360 = 2 \otimes 180$	$560 = 4 \otimes 140$	$760 = 2 \otimes 380$	$960 = 16 \otimes 60$
$164 = 163+1$	$364 = 2(181+1)$	$564 = 563+1$	$764 =$ Goe-Sei	$964 =$ Miyamoto
$168 = 2 \otimes 84$	$368 = 4 \otimes 92$	$568 = 2 \otimes 284$	$768 = 24 \otimes 32$	$968 = 2 \otimes 484$
$172 =$ Williamson	$372 =$ Baumert-Hall	$572 = 571+1$	$772 =$ Miyamoto	$972 = 971+1$
$176 = 4 \otimes 44$	$376 = 2 \otimes 188$	$576 = 16 \otimes 36$	$776 = 2 \otimes 388$	$976 = 4 \otimes 244$
$180 = 179+1$	$380 = 379+1$	$580 = 2(17^2+1)$	$780 = 2(389+1)$	$980 =$ Goe-Sei
$184 = 2 \otimes 92$	$384 = 12 \otimes 32$	$584 = 2 \otimes 292$	$784 = 4 \otimes 196$	$984 = 2 \otimes 492$
$188 =$ Goe-Sei	$388 = 2(193+1)$	$588 = 587+1$	$788 =$ Miyamoto	$988 =$ Supl.Di.Sets
$192 = 191+1$	$392 = 2 \otimes 196$	$592 = 4 \otimes 148$	$792 = 2 \otimes 396$	$992 = 8 \otimes 124$
$196 = 2(97+1)$	$396 = 2(197+1)$	$596 =$ Miyamoto	$796 = 2(397+1)$	$996 =$ Wallis
$200 = 199+1$	$400 = 4 \otimes 100$	$600 = 2 \otimes 300$	$800 = 8 \otimes 100$	$1000 = 2 \otimes 500$

Figura 1: Listado de matrices de Hadamard con órdenes menores de 1000, obtenido de [9].

Referencias

[1] Alicia M. Delgado de Brandao, Yanina del Carmen Rodríguez Reyes, Ubaldino Sandoval Moreno, and Temístocles Zeballos Mitre. Determinantes: historia y resultados especiales, 2024. Universidad de Panamá. Último acceso: 2024-03-23.

[2] Mario Fioravanti. Matemática discreta, segunda parte: Teoría de códigos. *Universidad de Cantabria.*, pages 2–9, 2021.

[3] Wolfram MathWorld. Hadamard matrix, 2024. Último acceso: 2024-06-05.

[4] Luis Miguel Pardo. Algunas notas para un curso elemental de Álgebra conmutativa, parte i. *Universidad de Cantabria.*, pages 262–263, 2023.

[5] Eduardo Pisa. Búsqueda de matrices de hadamard a través de secuencias de turyn. *Dialnet. REVISTA DE MATEMÁTICA: TEORÍA Y APLICACIONES.*, 2011.

[6] Wissam Raji. Introducción a los residuos cuadráticos y no residuos, 2024. Último acceso: 2024-04-30.

[7] Fernando Revilla. Desigualdad de hadamard, 2015. Último acceso: 2024-04-06.

[8] Tomás Rodriguez. Matrices hadamard. *Universidad de Sevilla*, pages 1–7, 2003.

[9] Esteban Segura Ugalde. Secuencias de turyn. *Universidad de Costa Rica.REVISTA DE MATEMÁTICA: TEORÍA Y APLICACIONES 2019.*, pages 4–7, 2018.

[10] Guillermo Sosa Gómez. Utilización de la transformada y matrices de hadamard en las funciones booleanas y en el criptoanálisis. *Universidad Central, Marta Abreu de las Villas.*, pages 26–29, 2010.