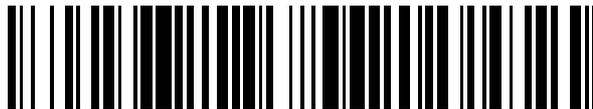


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 370 822**

51 Int. Cl.:
H03M 13/11 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **04715448 .9**
96 Fecha de presentación: **27.02.2004**
97 Número de publicación de la solicitud: **1598942**
97 Fecha de publicación de la solicitud: **23.11.2005**

54 Título: **GENERACIÓN DE MATRIZ DE COMPROBACIÓN PARA CÓDIGOS DE COMPROBACIÓN DE PARIDAD DE BAJA DENSIDAD (LDPC) IRREGULARES CON TASA DE CÓDIGO DETERMINADA.**

30 Prioridad:
28.02.2003 JP 2003053162

45 Fecha de publicación de la mención BOPI:
23.12.2011

45 Fecha de la publicación del folleto de la patente:
23.12.2011

73 Titular/es:
**MITSUBISHI ELECTRIC CORPORATION
7-3, MARUNOUCHI 2-CHOME CHIYODA-KU
TOKYO 100-8310, JP**

72 Inventor/es:
MATSUMOTO, Wataru

74 Agente: **de Elizaburu Márquez, Alberto**

ES 2 370 822 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Generación de matriz de comprobación para códigos de comprobación de paridad de baja densidad (ldpc) irregulares con tasa de código determinada.

ÁMBITO TÉCNICO

La presente invención se refiere a un método y a un aparato para generar una matriz de comprobación para un código de comprobación de paridad de baja densidad (LDPC) que se aplica como un código de corrección de errores y, más particularmente, a un método y a un aparato para la generación de una matriz de comprobación capaz de buscar una matriz de comprobación definida y de característica estabilizada del código de LDPC.

TÉCNICA ANTERIOR

Más adelante se explicará un método convencional de generación de matrices de comprobación para códigos de LDPC. En un sistema convencional de codificación/decodificación de códigos de LDPC, un aparato de comunicación en un lado emisor tiene un codificador y un modulador. Por otro lado, un aparato en el lado receptor tiene un desmodulador y un decodificador. Antes de la explicación del método convencional de generación de matrices de comprobación para códigos de LDPC, se explican primero los flujos de codificación y decodificación utilizando códigos de LDPC.

En el lado emisor, el codificador genera una matriz de comprobación H de acuerdo con el método convencional que se describe más adelante. Entonces, se obtiene una matriz generadora G basándose en la siguiente condición.

G : $k \times n$ matriz (k : longitud de la información, n : longitud de la palabra de código)

$$GH^T = 0 \text{ (T denota transposición)}$$

El codificador recibe entonces un mensaje $(m_1 \ m_2 \dots \ m_k)$ de una longitud k de información, y genera una palabra de código C utilizando la matriz generadora G .

$$\begin{aligned} C &= (m_1 \ m_2 \dots \ m_k) G \\ &= (c_1 \ c_2 \dots \ c_n) \text{ (donde } H (c_1 \ c_2 \dots \ c_n)^T = 0) \end{aligned}$$

El modulador somete a la palabra generada C del código a una modulación digital, tal como una modulación por desplazamiento de fase bivalente (BPSK), modulación por desplazamiento de fase en cuadratura (QPSK) y modulación de amplitud en cuadratura (QAM) de varios valores, y envía la señal modulada.

En el lado receptor por otro lado, el desmodulador recibe la señal modulada a través del canal, y la somete a una desmodulación digital, tal como BPSK, QPSK, y QAM de múltiples valores. Entonces, el decodificador realiza una decodificación iterativa mediante un "algoritmo de suma-producto" con respecto al resultado desmodulado que se codifica con LDPC, y envía un resultado estimado (que corresponde al $m_1 \ m_2 \dots \ m_k$ original).

Más adelante se explicará un método convencional de generación de matrices de comprobación para códigos de LDPC. Como matriz de comprobación para códigos de LDPC se propone la siguiente matriz en R.G. Gallager, "Códigos de comprobación de paridad de baja densidad", M.I.T. Press, Cambridge, MA, 1963 (véase la Fig. 16).

La matriz que se muestra en la Fig. 16 es una matriz binaria de "1" y "0", en la que se crea un parte de "1". Otras partes son todas "0". En esta matriz, el número de "1" (unos) en una fila (expresados como peso de fila) es igual a 4, y el número de "1" en una columna (expresados como peso de columna) es igual a tres. Todas las columnas y las filas tienen respectivos pesos uniformes. Por lo tanto, generalmente se llama un "código de LDPC regular". En los códigos de la literatura que no es de patentes 1, la matriz se separa en tres bloques, por ejemplo, y el segundo y tercer bloques son sometidos a una permutación aleatoria, como se muestra en la Fig. 16.

Debido a que la permutación aleatoria no tiene ninguna regla determinada, es necesario ejecutar una búsqueda lenta por ordenador para encontrar los códigos con una mejor característica.

En el documento "Códigos de comprobación de paridad de baja densidad basados en geometrías finitas: un redescubrimiento", de Y. Kou, Lin S., y Fossorier MPC, ISIT 2000, pág. 200, Sorrento, Italia, 25 a 30 de junio de 2000 se propone un método que utiliza códigos de geometría euclidiana como códigos de LDPC que exhiben una característica relativamente estable y satisfactoria y pueden generar definitivamente una matriz sin el uso de la búsqueda por ordenador. Este método explica el "código de LDPC regular" consistente en conjuntos regulares.

La segunda literatura propone un método para generar una matriz de comprobación para códigos de LDPC usando los códigos de geometría euclidiana $EG(2, 2^b)$ como una especie de códigos geométricos finitos. Este método consigue una característica que se encuentra cerca, pero a 1,45 decibelios del límite de Shannon con una tasa de

error de 10^{-4} . La Fig. 17 es un diagrama de una configuración de los códigos de geometría euclidiana EG (2, 2²), que tiene una estructura de "Códigos de LDPC regulares" con pesos de filas y columnas de 4 y 4, respectivamente.

Los códigos de geometría euclidiana EG (m, 2^s) tienen una característica definida de la siguiente manera:

Longitud de código: $n = 2^{2s}-1$

Longitud de bit redundante: $n-k = 3^s-1$

Longitud de la información: $k = 2^{2s}-3^s$

Distancia mínima: $d_{\min} = 2^s + 1$

Densidad: $r = 2^s / (2^{2s}-1)$.

Como se puede observar en la Fig. 17, los códigos de geometría euclidiana tienen una estructura con una ubicación cíclicamente cribada de "1" en cada fila de una fila adyacente. Esta estructura puede configurar característicamente códigos con facilidad y definitivamente.

El método de generación de una matriz de comprobación en la segunda literatura incluye, además, el cambio de los pesos de fila y columna basándose en los códigos de geometría euclidiana para ampliar las filas y columnas, si es necesario. Por ejemplo, cuando un peso en la columna en EG (2, 2²) se separa en dos mitades, en la segunda literatura, cada uno de los otros cuatro pesos situados en una columna se separa en dos grupos. La Fig. 18 es un diagrama de un ejemplo de separación regular del peso de columna desde 4 a 2.

Por otro lado, el documento "Mejores códigos de comprobación de paridad de baja densidad utilizando gráficos irregulares y propagación de creencia" de M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi y D. A. Spielman, Actas del Simposio Internacional IEEE 1998 en la Teoría de la Información, página 171, Cambridge, Massachusetts, agosto del 16 al 21 de 1998, ha señalado que los "códigos de LDPC irregulares" tienen una mejor característica que los "códigos de LDPC regulares". Esto se analiza teóricamente en el documento "La capacidad de los códigos de comprobación de paridad de baja densidad bajo decodificación de paso de mensajes" de T. J. Richardson y R. Urbanke, IEEE Transacciones en teoría de la información, vol. 47, N ° 2, págs. 599-618, febrero de 2001 y el documento "Análisis de la decodificación de suma-producto de códigos de comprobación de paridad de baja densidad utilizando una aproximación Gaussiana" de S. Y. Chung, T. J. Richardson, y R. Urbanke, IEEE Transacciones en teoría de la información, vol. 47, N ° 2, pp 657-670, febrero de 2001". Los "códigos de LDPC irregulares" representan los códigos de LDPC que no tienen uniformidad en uno o ambos de los pesos de fila y de columna.

En particular, en la quinta literatura, se analiza un "algoritmo de suma-producto" para códigos de LDPC en el supuesto de que una relación de similitud (LLR) entre una entrada y una salida en un decodificador iterativo se puede aproximar en una distribución Gaussiana, para derivar un conjunto satisfactorio de pesos de filas y de columnas.

De acuerdo con el método convencional de generación de matrices de comprobación los códigos de LDPC descritos en la quinta literatura, sin embargo, el número de puntos "1" en una fila (que corresponde a una distribución de grado de nodos variables que se describe más adelante) y el número de puntos "1" en una columna (que corresponde a una distribución de grado de nodos de comprobación que se describe más adelante) son empleados como variables para obtener la distribución de grado de los nodos variables y la distribución de grado de los nodos de comprobación que puede maximizar la siguiente ecuación (1) (tasa: tasa de codificación). En otras palabras, en programación lineal se utiliza para buscar un conjunto que minimiza una relación de señal a ruido (SNR).

$$tasa = 1 - \frac{\int_0^1 p(x)}{\int_0^1 \lambda(x)} \quad (1)$$

Por lo tanto, surge el problema de que una matriz de comprobación derivada del máximo de la "tasa" tiene una característica inestable y que varía continuamente. Además, el método convencional de generación de matrices de comprobación para códigos de LDPC ejecuta reiterativamente la derivación de la distribución de grado de los nodos variables y la derivación de la distribución de grado de los nodos de comprobación durante ciertos momentos. Por lo tanto, surge el problema de que, en cierta medida, la búsqueda lleva tiempo.

El documento "Códigos de comprobación de paridad de baja densidad reticular y su aplicación en los sistemas de respuesta parcial" de Vasic B et al., PROC, IEEE Simposio Internacional sobre Teoría de la Información, ISIT 02, Lausana, Suiza, 30 de junio de 2002, página 453, XP010602164 describe una construcción combinatoria de la alta tasa de códigos de comprobación de paridad de baja densidad basándose en entramados de enteros de dos dimensiones. Se obtiene una clase de códigos con un amplio intervalo de longitudes, pesos de columna y distancias

mínimas. Los códigos resultantes son doblemente circulantes, es decir, la matriz de comprobaciones de paridad es una formación de matrices circulantes. Se establecen los límites de la distancia mínima y se investigan las prestaciones de estos códigos en los canales de respuesta parcial.

5 La presente invención se ha logrado tras la consideración de los problemas anteriores, y en consecuencia tiene como un objetivo el proporcionar un método de generación de matrices de comprobación para códigos de LDPC que tienen unas prestaciones satisfactorias capaces de buscar fácilmente una matriz de comprobación definida y de característica estabilizada para códigos de LDPC correspondiente a un conjunto opcional, y a un aparato de generación de matrices de comprobación.

10

El alcance de la invención se define en las reivindicaciones anexas.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

15 La Fig. 1 es un diagrama de flujo de un método para la generación de matrices de comprobación para códigos de LDPC de acuerdo con la presente invención;
 la Fig. 2 es un diagrama de un sistema de codificación/decodificación de LDPC;
 la Fig. 3 es un diagrama de una estructura reticular cuando $1(x, y) = m^x x + y + 1$, $m = 5$ y $k = 3$;
 la Fig. 4 es un diagrama de m clases que utilizan una línea de pendiente s como una unidad;
 20 la Fig. 5 es un diagrama de un algoritmo para diseñar un agregado que establece 8 como un número mínimo de ciclos;
 la Fig. 6 es un diagrama de un resultado de búsqueda cuando el algoritmo que se muestra en la Fig. 5 se ejecuta con $m = 5$ y $k = 3$;
 la Fig. 7 es un diagrama de un ejemplo de una matriz básica;
 la Fig. 8 es un diagrama de un resultado de búsqueda cuando el algoritmo que se muestra en la Fig. 5 se
 25 ejecuta con $m = 353$ y $k = 10$;
 la Fig. 9 es un diagrama de un algoritmo de permutación (reorganización) de una matriz básica;
 la Fig. 10 es un diagrama de un ejemplo de una matriz básica después de una permutación;
 la Fig. 11 es un diagrama de un conjunto final de una función generadora $\lambda(x)$ y una función generadora $\rho(x)$;
 la Fig. 12 es un diagrama de un procedimiento de división de acuerdo a un documento convencional;
 30 la Fig. 13 es un diagrama de una secuencia aleatoria básica $C(i)$ y los patrones de permutación de la secuencia aleatoria básica;
 la Fig. 14 es un diagrama de secuencias de cuadrado latino $L_{iq}(i)$;
 la Fig. 15 es un diagrama de una relación entre una E_b/N_0 y una tasa de error de bites;
 la Fig. 16 es un diagrama de una matriz de comprobación convencional para códigos de LDPC;
 35 la Fig. 17 es un diagrama de una configuración de códigos de geometría euclidiana $EG(2, 2^2)$; y
 la Fig. 18 es un diagrama de un ejemplo de peso de columnas separados regularmente de 4 a 2.

MEJOR MANERA DE REALIZAR LA INVENCION

40 A continuación se explicarán con detalle unos ejemplos de realización de un método y un aparato para generar una matriz de comprobación de acuerdo con la presente invención haciendo referencia a los dibujos adjuntos. Cabe señalar que no se pretende que las realizaciones limiten la invención.

45 La Fig. 1 es un diagrama de flujo de un método para la generación de matrices de comprobación para códigos de LDPC de acuerdo con la presente invención. El método para la generación de matrices de comprobación para códigos de LDPC de acuerdo con la presente realización puede ser ejecutado dentro de un aparato de comunicación de acuerdo con unos parámetros establecidos, o puede ser ejecutado en otra unidad de control (tal como un ordenador) fuera del aparato de comunicación. Cuando el método para la generación de matrices de comprobación para códigos de LDPC de acuerdo con la presente realización se ejecuta en el exterior del aparato de comunicación, las matrices de comprobación generadas para códigos de LDPC se almacenan en el aparato de comunicación. En la
 50 siguiente realización, el método anterior se ejecuta dentro del aparato de comunicación por el bien de la explicación.

55 Antes de la explicación del método de generación de matrices de comprobación para códigos de LDPC de acuerdo con la presente realización, las posiciones de un codificador y un decodificador capaces de lograr el método se explican primero junto con el método convencional de generación de matrices de comprobación para códigos de "LDPC irregular".

60 La Fig. 2 es un diagrama de un sistema de codificación/decodificación de LDPC. En la Fig. 2, un aparato de comunicación en un lado emisor incluye un codificador 101 y un modulador 102, y un aparato de comunicación en un lado receptor incluye un desmodulador 104 y un decodificador 105. A continuación se explicarán los flujos de codificación y decodificación utilizando códigos de LDPC.

65 En el lado emisor, el codificador 101 genera una matriz de comprobación 11 utilizando el método de generación de matrices de comprobación para los códigos de LDPC de acuerdo con la presente realización que se describe más adelante. Entonces, se deriva una matriz generadora G a partir de la siguiente condición.

G: $k \times n$ matriz (k: longitud de la información, n: longitud de la palabra de código)

$$\mathbf{GH}^T = \mathbf{0} \text{ (T denota transposición)}$$

5 El codificador 101 recibe entonces un mensaje ($m_1 m_2 \dots m_k$) de una longitud k de información, y genera una palabra de código C utilizando la matriz generadora G.

$$\mathbf{C} = (\mathbf{m}_1 \mathbf{m}_2 \dots \mathbf{m}_k) \mathbf{G}$$

10
$$= (\mathbf{c}_1 \mathbf{c}_2 \dots \mathbf{c}_n) \text{ (donde } \mathbf{H} (\mathbf{c}_1 \mathbf{c}_2 \dots \mathbf{c}_n)^T = \mathbf{0} \text{)}$$

El modulador 102 somete a la palabra C de código generado a una modulación digital tal como BPSK, QPSK y QAM de varios valores, y envía la señal modulada.

15 En el lado receptor por otro lado, el desmodulador 104 recibe la señal modulada a través del canal 103, y la somete a una desmodulación digital, tal como BPSK, QPSK, y QAM de múltiples valores. El decodificador 105 somete entonces al resultado desmodulado codificado con LDPC a una decodificación iterativa mediante un "algoritmo de suma-producto" y proporciona un resultado estimado (que corresponde al original ($m_1 m_2 \dots m_k$)).

20 El método convencional para generar matrices de comprobación para "códigos de LDPC irregulares" en teoría analizado en la quinta literatura se explica a continuación con detalle. En este caso, se analiza un "algoritmo de suma-producto" para códigos de LDPC en el supuesto de que una relación de similitud (LLR) entre una entrada y una salida en un decodificador iterativo se pueda aproximar en una distribución Gaussiana, para derivar un conjunto satisfactorio de pesos de filas y de columnas.

25 El método para la generación de matrices de comprobación para códigos de LDPC descrito en la quinta literatura, o la aproximación Gaussiana, tiene una premisa que define un punto de "1" en una fila como un nodo variable y un punto de "1" en una columna como un nodo de comprobación en la matriz de comprobación.

30 En primer lugar se analiza la propagación de mensajes LLR de un nodo de comprobación a un nodo variable. La siguiente función (2) se define con la condición de que $0 < s \leq \infty$ and $0 \leq t < \infty$. En este caso, $s = m_{u0}$ denota una media de $u0$; $u0$ un LLR asociado con una señal recibida a través de un canal que contiene un ruido gaussiano de valor distribuido σ_n^2 , y t una media de conjunto de valores de salida LLR en los nodos de comprobación en el momento de la iteración determinada.

35
$$f_i(s,t) = \phi^{-1} \left(1 - \left[1 - \sum_{i=2}^{d_i} \lambda_i \phi(s + (i+1)t) \right]^{i-1} \right)$$

$$f(s,t) = \sum_{j=2}^{d_i} \rho_j f_j(s,t) \tag{2}$$

40 λ_i y ρ_i denotan relaciones de bordes que pertenecen a los nodos variables y los nódulos de comprobación en un grado de i , respectivamente, d_i denota un grado de nodos variables máximos y d_r un grado de nodos de comprobación máximos. $\lambda(x)$ y $\rho(x)$ denotan unas funciones generadoras de distribución de grado asociadas con los nodos variables y los nodos de comprobación, y pueden representarse por las ecuaciones (3) y (4), respectivamente. (Un grado se expresa como el número de "1" (unos) en cada fila de nodos variables o cada columna de nodos de comprobación).

45
$$\lambda(x) = \sum_{i=2}^{d_i} \lambda_i x^{i-1} \tag{3}$$

50
$$\rho(x) = \sum_{i=2}^{d_i} \rho_i x^{i-1} \tag{4}$$

donde, $\phi(x)$ se define como se muestra en la siguiente ecuación (5).

55
$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_{\mathbb{R}} \tanh \frac{u}{2} e^{-\frac{(u-x)^2}{4x}} du & \text{si } x > 0 \\ 1 & \text{si } x \leq 0 \end{cases} \tag{5}$$

La ecuación (2) se puede representar de forma equivalente por la siguiente ecuación (6).

60
$$t \cdot f(s, t_{\cdot 1}) \tag{6}$$

donde t_i denota una media del conjunto de valores de salida LLR en los nodos de comprobación en el momento de la iteración i -ésima.

Una condición necesaria para derivar un límite (umbral) de SNR que proporciona un error con un valor de 0 incluye $t_i(s) \rightarrow \infty$ (expresada como R') cuando $i \rightarrow \infty$. Para satisfacer esta condición, se requiere satisfacer la siguiente ecuación condicional (7).

$$t < f(s, t), \text{ todos } t \in R' \quad (7)$$

A continuación se analiza la propagación de mensajes LLR desde un nodo variable a un nodo de comprobación. La siguiente función (8) se define con la condición de que $0 < r < 1$ y $0 < r \leq 1$. En este caso, r tiene un valor inicial r_0 de $\phi(s)$.

$$h_i(s, r) = \phi \left(s + (i-1) \sum_{j=2}^{d_r} \rho_j \phi^{-1} (1 - (1-r)^{j-1}) \right)$$

$$h(s, r) = \sum_{i=2}^{d_r} \lambda_i h_i(s, r) \quad (8)$$

La ecuación (8) se puede representar de forma equivalente por la siguiente ecuación (9).

$$r = h(s, r_{i-1}) \quad (9)$$

Una condición necesaria para derivar un límite (umbral) de SNR que proporciona un error con un valor de 0 incluye $r_i(s) \rightarrow 0$. Para satisfacer esta condición, se requiere satisfacer la siguiente ecuación condicional (10).

$$r > h(s, r), \text{ todas } r \in (0, \phi(s)) \quad (10)$$

En la quinta literatura, los grados óptimos se buscan para nodos variables y nodos de comprobación utilizando la ecuación anterior en el siguiente procedimiento (Aproximación Gaussiana).

- (1) En el supuesto de que se dé una función generadora $\lambda(x)$ y un ruido Gaussiano σ_n , se utiliza una función generadora $\rho(x)$ como una variable para buscar un punto que maximice la ecuación (1) descrita anteriormente. Una condición de restricción asociada con esta búsqueda incluye la normalización a $\rho(x) = 1$ y la satisfacción de la ecuación (7).
- (2) En el supuesto de que se dé una función generadora $\rho(x)$ y el ruido gaussiano σ_n (como un valor resultante del procedimiento (1), por ejemplo), se utiliza una función generadora $\lambda(x)$ como una variable para buscar un punto que maximice la ecuación (1). Una condición de restricción asociada con esta búsqueda incluye la normalización a $\lambda(x) = 1$ y la satisfacción de la ecuación (10).
- (3) Con el fin de encontrar la máxima "tasa", los procedimientos (1) y (2) se ejecutan iterativamente para buscar un mejor conjunto de la función generadora $\lambda(x)$ y la función generadora $\rho(x)$, con una programación lineal.
- (4) Por último, la potencia de la señal se normaliza a 1 basándose en el ruido Gaussiano σ_n para encontrar un límite (umbral) de SNR (véase la ecuación (11)).

$$\text{umbral}(dB) = -10 * \log_{10}(2 * \sigma_n^2) \quad (11)$$

En la quinta literatura se encuentra el problema de que una matriz de comprobación derivada de la máxima "tasa (tasa de codificación)" cambia continuamente y la tasa se fija en el diseño a medida que varía la especificación. Además, en la quinta literatura, la derivación de la distribución de grado asociada con los nodos variables y la derivación de la distribución de grado asociada con los nodos de comprobación se ejecutan iterativamente en determinados momentos. Por lo tanto, surge el problema de que, en cierta medida, la búsqueda lleva tiempo. Además, surge un problema de que la matriz de comprobación no se puede aplicar fácilmente a un conjunto opcional, una longitud opcional de código y una tasa opcional de codificación.

En la presente realización, se explica un método para buscar fácilmente en poco tiempo una matriz de comprobación definida y de característica estabilizada para "códigos de LDPC irregulares", que corresponde a un conjunto opcional, un código opcional de longitud y una tasa opcional de codificación (véase la Fig. 1). Específicamente, la matriz de comprobación para "códigos de LDPC irregulares" se genera mediante el uso de una matriz básica que se describe más adelante (definición: "los pesos de fila y columna son constantes", y "el número de ciclos es igual o superior a 6").

En el método para generar una matriz de comprobación para códigos de LDPC de acuerdo con la presente realización, primero se determina una tasa de codificación (etapa S1). A modo de ejemplo, la tasa de codificación se establece en 0,5.

5 Se crea una matriz básica basada en una estructura reticular de enteros en el supuesto de la matriz básica (definición: "los pesos de filas y columnas son constantes", y "el número de ciclos es igual o superior a 6") que es necesario para obtener una matriz de comprobación de "códigos de LDPC irregulares" (etapa S2). En la codificación/decodificación utilizando códigos LDPC, por lo general cuando el número de "ciclo de 4" y "ciclo 6" es lo más pequeño posible en un gráfico binario, se pueden obtener características satisfactorias. Por lo tanto, se prefieren los códigos de LDPC que tienen una construcción para restringir la aparición de un pequeño número de "ciclo 4" y "ciclo 6". Por lo tanto, en la presente realización, se crea una matriz básica que utiliza un número mínimo de ciclos de 8. Se puede explicar un procedimiento para crear una matriz básica (número mínimo de ciclos como 8), basado en una estructura reticular de enteros. Se puede utilizar una matriz básica que utiliza un gráfico de Cayley, una matriz básica que utiliza un gráfico de Ramanujan u otra matriz siempre y cuando la matriz básica satisfaga la definición anterior.

Se explicará el procedimiento para la creación de una matriz básica basándose en una estructura reticular de enteros.

20 (1) En primer lugar, se diseña un agregado de líneas que se conectan entre unos puntos (una combinación de puntos) en una estructura reticular de enteros. Por ejemplo, un agregado de una estructura de entramado se establece como $L = (x, y)$. En este caso, x es un número entero de $0 \leq x \leq k-1$, y es un número entero de $0 \leq y \leq m-1$, k es un número entero, y m es un número primo. $l(x, y)$ es una asignación lineal para asignar el agregado L a un agregado V de puntos. La Fig. 3 es un diagrama de una estructura de entramado cuando $l(x, y) = m^x x + y + 1$, $m = 5$ y $k = 3$; En la Fig. 3, una combinación de puntos que satisface una determinada condición se llama una línea (bloque). Por ejemplo, una línea que tiene una pendiente s ($0 \leq s \leq m-1$) consiste en una línea de $l(x, a+sx \text{ mod } m)$ que tiene un punto $(0, a)$ como punto de partida, donde a es $0 \leq a \leq m-1$. Como resultado, se crean m clases que tienen la línea de la pendiente s como una unidad. La Fig. 4 es un diagrama de m clases que tienen la línea de la pendiente s como una unidad.

30 (2) Un agregado que tiene un número mínimo de ciclos como 8 ha sido diseñado basándose en el algoritmo de la Fig. 5. En otras palabras, se busca una serie de columnas, una serie de filas, el peso de las columnas y el peso de las filas. En la Fig. 5, S denota un agregado de s , y $B(s)$ denota un agregado de una clase correspondiente a la pendiente s .

35 La Fig. 6 es un diagrama de un resultado de búsqueda cuando el algoritmo anterior se ejecuta utilizando $m = 5$ y $k = 3$. $N = |B|$ denota una serie de columnas en la matriz básica, $M = |V|$ denota una serie de filas en la matriz básica, d_c denota el peso de las filas en la matriz básica, y d_v denota una serie de columnas de la base matriz. La Fig. 7 es un diagrama de un resultado de la ejecución del algoritmo anterior, es decir, una matriz básica.

40 La matriz básica debe diseñarse con un tamaño relativamente grande con el fin de realizar una división de filas y columnas como se describe más adelante. Cuando el algoritmo anterior se ejecuta utilizando $k = 10$ y $m = 353$, se obtiene un resultado de la búsqueda como se muestra en la Fig. 8. Cuando $d_v = 10$ y $d_c = 7$ como se muestra en la Fig. 8, se puede llevar a cabo un procesamiento de división como se describe más adelante. En la presente realización, se ejecuta el procesamiento de la etapa S2 para afirmar una serie de columnas N ($= 2.471$) en la matriz básica sobre la base de la estructura reticular de enteros y una serie de filas M ($= 3530$) en la matriz básica sobre la base de la estructura reticular de enteros.

45 A continuación, se determina una serie de columnas N' de una matriz de comprobación (una matriz de comprobación de "códigos de LDPC irregulares") obtenida finalmente (etapa S3). En este punto, una serie de filas M' de la matriz de comprobación de "códigos de LDPC irregulares" se determina como $M' = N'^x$ (1-tasa). Por ejemplo, cuando $N' = 6000$ y la tasa $= 0,5$, M' se determina como $M' = 6000^x 0,5 = 3000$.

50 A continuación, la matriz básica creada anteriormente se reorganiza de acuerdo con un algoritmo que se muestra en la Fig. 9 de modo que se colocan varios 1 (unos) en las posiciones más altas en las columnas (etapa S4). La Fig. 9 es un diagrama de un algoritmo de permutación (reorganización) de una matriz básica. La Fig. 10 es un diagrama de un ejemplo de una matriz básica después de una permutación. La matriz básica cuando $k = 3$ y $m = 5$ que se muestra en la Fig. 7 se reorganiza de acuerdo con el algoritmo que se muestra en la Fig. 9. Basándose en esta permutación, las columnas de gran peso se pueden dejar en la medida de lo posible y una variación en el peso dentro de las columnas se pueden reducir en la medida de lo posible, en el procesamiento de eliminación que se describe más adelante.

55 A continuación, un conjunto (distribución de grado) de "códigos de LDPC irregulares", basado en una tasa de codificación solicitada se obtiene provisionalmente utilizando una optimización por aproximación gaussiana (etapa S5). En este caso, γ_i ($i = 1, 2, \dots$, máximo, $2 \leq \gamma_1 < \gamma_2 < \dots < \gamma_{\max}$) denota un peso de columna, μ y $\mu+1$ ($2 \leq \mu \leq d_c-1$) denota pesos de fila, d_c denota un peso de una matriz básica, λ_{γ_i} ($0 \leq \lambda_{\gamma_i} \leq 1$) denota una relación de un borde que

pertenece a los pesos de columna γ_i , $\rho_{-\mu}$ y $\rho_{-(\mu+1)}$ ($0 \leq \rho_{-\mu}, \rho_{-(\mu+1)} \leq 1$) denota relaciones de un borde que pertenece a los pesos de fila μ y $\mu+1$, $b_{-\mu}$ y $b_{-(\mu+1)}$ denota enteros no negativos, $\lambda(x)$ denota una función generadora de una distribución de pesos de columna, $\rho(x)$ denota una función generadora de una distribución de pesos de filas, $n_{-\mu}$ y $n_{-(\mu+1)}$ denotan números de filas de los pesos de filas μ y $\mu+1$, y $n_{-\gamma_i}$ denota un número de columnas del peso de columna γ_i . La $\lambda(x)$ y $\rho(x)$ anteriores se definen por la ecuación (13).

$$\rho(x) = \rho_{-\mu} x^{\mu-1} + (1-\rho_{-\mu}) x^{\mu}$$

$$b_{-\mu} x^{\mu} + b_{-(\mu+1)} x^{\mu+1} = d_c \quad (12)$$

$$\rho_{\mu_i} = \frac{\mu x b_{-\mu}}{d_c}$$

$$\rho_{\mu+1_i} = \frac{(\mu+1) x b_{-(\mu+1)}}{d_c} \quad (13)$$

Se explicará el procedimiento de ejecución de aproximación gaussiana de acuerdo con la presente realización para buscar un conjunto de una función generadora $\lambda(x)$ de una distribución de pesos de columna y la función generadora $\rho(x)$ de una distribución de pesos de fila.

- (1) Se fija una tasa de codificación "tasa" (S1 paso).
- (2)

Una función generadora $\lambda(x)$ y una función generadora $\rho(x)$ se utilizan simultáneamente como variables, y se utiliza una programación lineal para la búsqueda de funciones generadoras óptimas $\lambda(x)$ y $\rho(x)$ que pueden maximizar el ruido gaussiano (véase la siguiente ecuación (14)). Una condición de restricción asociada a esta búsqueda es satisfacer la ecuación (18) que se describe más adelante.

$$h_i = \phi \left(s + (i-1) \sum_{j=2}^{\mu_{max}} \rho_j \phi^{-1} (1 - (1-r)^{j-1}) \right)$$

$$h(s,r) = \sum_{i=2}^{\mu_{max}} \lambda_i h_i(s,r) \quad (14)$$

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_R \tanh \frac{\mu}{2} \cdot e^{-\frac{(\mu-x)}{4x}} d\mu, & \text{si } x > 0 \\ 1, & \text{si } x \leq 0 \end{cases}$$

La s anterior denota un promedio de una relación de similitud (LLR) entre una señal binaria de $\{-1,1\}$ recibida como una señal de transmisión y una señal recibida a través del canal gaussiano, y puede ser derivada de $s = 2/\sigma_n^2$.

Como se explicó anteriormente, de acuerdo con la presente realización, las funciones generadoras $\lambda(x)$ y $\rho(x)$ que cumplen una condición predeterminada se obtienen en la programación lineal temporal. Por lo tanto, es posible crear un conjunto definido y de característica estabilizada con mayor facilidad en menos tiempo de lo que es por el método descrito en la quinta literatura que ejecuta iterativamente la derivación de las funciones generadoras $\lambda(x)$ y $\rho(x)$ para derivar los dos valores óptimos.

El número de filas M' (= 3.000) se obtiene después de ejecutar el procesamiento de dividir las filas que se describe más adelante $b_{-\mu}$, $b_{-(\mu+1)}$, μ y $\mu+1$ y obtenidos en la etapa S5. Se obtiene el número de filas M (= 3530) de la matriz básica. En este caso, las filas de un número que se muestra en la siguiente ecuación (15) se eliminan en orden desde la parte inferior de la matriz básica después de la permutación (etapa S6). En este ejemplo, se eliminan $(3530-3000) / (1+0) = 530$ filas. Una matriz después de la eliminación de las filas tiene un conjunto de pesos de fila como $\{d_1, d_2, \dots, d_v\}$.

$$\begin{bmatrix} M-M' \\ b_{-\mu} + b_{-(\mu+1)} \end{bmatrix} \quad (15)$$

A continuación, un conjunto (distribución de grado) de "códigos de LDPC irregulares", basado en una tasa solicitada de codificación se obtiene utilizando la optimización por aproximación Gaussiana, usando las ecuaciones de restricción siguientes (16), (17), (18) y (19) (etapa S7). Una matriz $\beta = \{2, 3, \dots, d_v\}$ expresada por $(\beta)_{i,j}$ de la ecuación (16) representa una matriz de un entero no negativo $l(\beta)^{x \times L}$, que incluye todos los elementos que satisfacen la ecuación (16). Una matriz que se muestra expresada por $(\beta)_{i,j}$ de la ecuación (17) representa una matriz cuadrada de un número entero no negativo $|x|$.

$$\begin{bmatrix} a(2)_{1,1} & a(2)_{1,2} & \dots & a(2)_{1,max} \\ a(2)_{2,1} & a(2)_{2,2} & \dots & a(2)_{2,max} \\ \vdots & \vdots & \ddots & \vdots \\ a(2)_{I(2),1} & a(2)_{I(2),2} & \dots & a(2)_{I(2),max} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{max} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ \vdots \\ 2 \end{bmatrix}$$

$$\begin{bmatrix} a(3)_{1,1} & a(3)_{1,2} & \dots & a(3)_{1,max} \\ a(3)_{2,1} & a(3)_{2,2} & \dots & a(3)_{2,max} \\ \vdots & \vdots & \ddots & \vdots \\ a(3)_{I(3),1} & a(3)_{I(3),2} & \dots & a(3)_{I(3),max} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{max} \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ \vdots \\ 3 \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} a(dr)_{1,1} & a(dr)_{1,2} & \dots & a(dr)_{1,max} \\ a(dr)_{2,1} & a(dr)_{2,2} & \dots & a(dr)_{2,max} \\ \vdots & \vdots & \ddots & \vdots \\ a(dr)_{I(dr),1} & a(dr)_{I(dr),2} & \dots & a(dr)_{I(dr),max} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{max} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ \vdots \\ 2 \end{bmatrix}$$

$$\begin{bmatrix} A(\beta)_1 & 0 & 0 & \dots & 0 \\ 0 & A(\beta)_2 & 0 & \dots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & A(\beta)_I \end{bmatrix} \tag{17}$$

$$\frac{\int_0^1 \rho(x)}{\int_0^1 \lambda(x)} = 1 - tasa$$

$$\lambda(1) = 1, \rho(1) = 1$$

$r > h(s,r)$, todos $r \in (0, \phi(s))$ (18)

$$\sum_{\beta=2}^{d_v} \sum_{i=1}^l A(\beta)_i = N'$$

$$n_{Y_i} = \sum_{\beta=2}^{d_v} \sum_{i=1}^l A(\beta)_i \cdot a_{i,j}$$

$$\lambda_{Y_i} = \frac{n_i \times n_{Y_i}}{N' \times d_c} \tag{19}$$

$$b_{\mu} \times \mu + b_{(\mu+1)} \times (\mu+1) = d_c$$

$$n_{\mu} = b_{\mu} \times M', n_{(\mu+1)} = b_{(\mu+1)} \times M'$$

$$\rho_{\mu} = \frac{\mu \times n_{\mu}}{M' \times d_c}, \rho_{(\mu+1)} = \frac{(\mu+1) \times n_{(\mu+1)}}{M' \times d_c}$$

La Fig. 11 es un diagrama de un conjunto final de las funciones generadoras $\lambda(x)$ y $\rho(x)$ obtenido en la etapa S7, cuando el conjunto se ajusta en el procedimiento anterior.

Por último, se explicará un procedimiento para dividir una fila o una columna en la matriz básica tras la permutación (etapa S8). La segunda literatura propone un método regular de división relativo al procedimiento de división. La Fig. 12 es un diagrama del procedimiento de división en la literatura. En primer lugar, una matriz se numera como se muestra en la Fig. 12. En este ejemplo, los números de las columnas se dan como 1, 2, 3, y así sucesivamente en orden desde el extremo izquierdo, y números de las filas se dan como 1, 2, 3, y así sucesivamente en orden desde la parte superior. Para la división de 32 puntos \times una columna en 8 puntos \times 4 columnas, por ejemplo, esto se divide regularmente de acuerdo a la siguiente ecuación (20).

$$\mathbf{S}_m(\mathbf{n}) = \mathbf{B}_i(\mathbf{m} + 4^* \mathbf{n}) \tag{20}$$

En la ecuación anterior, $m = 1, 2, 3$ y 4 , y $n = 0, 1, 2, 3, 4, 5, 6$ y 7 ,

y I denota un número de columna de EG $(2,2^5)$. $B_i(x)$ denota una posición de "1" en una columna i -ésima de EG $(2,2^5)$. $S_m(n)$ denota una posición de "1" en una columna m -ésima de la matriz después de la división.

Específicamente, un número de fila que muestra una posición de "1" en la columna l-ésima de EG (2,2⁵) es la siguiente. B_l(x)={1 32 114 136 149 223 260 382 402 438 467 507 574 579 588 622 634 637 638 676 717 728 790 851 861 879 947 954 971 977 979 998}. Como resultado, un número de fila que muestra una posición de "1" en las columnas primera a cuarta de la matriz después de la división es el siguiente, basado en una extracción regular del número de "1" de B_l(x).

$$S_1(n) = \{1 \ 149 \ 402 \ 574 \ 634 \ 717 \ 861 \ 971\}$$

$$S_2(n) = \{32 \ 223 \ 438 \ 579 \ 637 \ 728 \ 879 \ 977\}$$

$$S_3(n) = \{114 \ 260 \ 467 \ 588 \ 638 \ 790 \ 947 \ 979\}$$

$$S_4(n) = \{136 \ 382 \ 507 \ 622 \ 676 \ 851 \ 954 \ 998\}$$

En otras palabras, 32 puntos ^x una columna se divide en 8 puntos ^x 4 columnas.

Por otro lado, en el procesamiento de división de la matriz básica después de la permutación de acuerdo con la presente invención, no se ejecuta una división regular como el procesamiento anterior, pero el número de "1" se extrae aleatoriamente de B_l(x) (véase un ejemplo detallado de una división aleatoria que se describe más adelante). Se puede utilizar cualquier método de procesamiento de la extracción siempre y cuando se mantenga la aleatoriedad.

Cuando una posición de "1" en la columna m-ésima de la matriz después de la división es R_m(n), por ejemplo, R_m(n) se convierte de la siguiente manera.

$$R_1(n) = \{1 \ 114 \ 574 \ 637 \ 851 \ 879 \ 977 \ 979\}$$

$$R_2(n) = \{32 \ 136 \ 402 \ 467 \ 588 \ 728 \ 861 \ 971\}$$

$$R_3(n) = \{149 \ 260 \ 382 \ 438 \ 579 \ 638 \ 717 \ 998\}$$

$$R_4(n) = \{223 \ 507 \ 622 \ 634 \ 676 \ 790 \ 947 \ 954\}$$

Se explicará con detalle un ejemplo de una división aleatoria, es decir, "un método de división utilizando un cuadrado latino de secuencias aleatorias". Una secuencia aleatoria para llevar a cabo la división aleatoria se crea con facilidad y definitivamente. Una ventaja de este método es que la misma secuencia aleatoria se puede crear en el lado emisor y el lado receptor. Esto es extremadamente importante en un sistema real. Este método también tiene la ventaja de que una condición de la característica de código puede prescribirse con precisión.

(1) Creación de secuencias aleatorias básicas:

Un ejemplo de creación de secuencia aleatoria se describe a continuación utilizando códigos de geometría euclidiana EG (2,2⁵) por comodidad de la explicación. En códigos de geometría euclidiana EG (2,2⁵), el número de "1" (unos) presente en una fila es igual a 2⁵ = 32.

Cuando P se utiliza para el primer número mínimo que satisface P >= d_v = 2^s, por ejemplo, P = 37 en el caso de d_v = 2⁵. Una secuencia aleatoria básica C(i) con una longitud de secuencia, P-5 = 32, se crea de acuerdo con la ecuación (21). d_v denota un peso máximo de una columna. Por lo tanto, cuando se seleccionan unos códigos distintos a los de geometría euclidiana como una matriz básica, el uso de d_v de esta matriz básica hace posible la aplicación de esta división.

$$C(1) = 1$$

$$C(i+1) = G_0^x C(i) \text{ mod } P \quad (21)$$

donde, 1=0,1, ...P-2; y G₀ denota una fuente original de campo de Galois GF (P). Como resultado, C(i) se representa por la siguiente ecuación:

$$C(i) = \{1 \ 2 \ 4 \ 8 \ 16 \ 32 \ 27 \ 17 \ 34 \ 31 \ 25 \ 13 \ 26 \ 15 \ 30 \ 23 \ 9 \ 18 \ 36 \ 35 \ 33 \ 29 \ 21 \ 5 \ 10 \ 20 \ 3 \ 6 \ 12 \ 24 \ 11 \ 22 \ 7 \ 14 \ 28 \ 19\}.$$

(2) Se eliminan números de mayores de 32 con el fin de obtener la longitud de la secuencia d_v = 2⁵ = 32.

$$C(i) = \{1 \ 2 \ 4 \ 8 \ 16 \ 32 \ 27 \ 17 \ 31 \ 25 \ 13 \ 26 \ 15 \ 30 \ 23 \ 9 \ 18 \ 29 \ 21 \ 5 \ 10 \ 20 \ 3 \ 6 \ 12 \ 24 \ 11 \ 22 \ 7 \ 14 \ 28 \ 19\}$$

(3) Se crea un patrón de permutación LB_j(i) utilizando la siguiente ecuación (22).

$$\begin{aligned} j &= 1, 2, \dots, d_v \\ i &= 1, 2, \dots, P-1 \end{aligned} \quad (22)$$

Se elimina un mayor número que $LB_j(i)$. La Fig. 13 es un diagrama de una secuencia aleatoria básica $C(i)$ y los patrones de permutación $LB_j(i)$ de la secuencia aleatoria básica;

(4) Una matriz de cuadrado latino j -ésima $L_{jp}(i)$ en la columna q y la fila i se calcula a partir de la siguiente ecuación (23) para ejecutar una división. Cuando el peso de la columna d_β es $d_\beta < d_v$, números mayores que d_β son rebajados a partir de los elementos de $L_{jq}(i)$, basándose en el procesamiento de eliminación en la etapa S6.

$$L_{jp}(i) = C(LB_j(((q+i-2) \bmod d_v) + 1))$$

$$\begin{aligned} j &= 1, 2, \dots, d_v \\ i &= 1, 2, \dots, d_v \\ q &= 1, 2, \dots, P-1 \end{aligned} \quad (23)$$

La Fig. 14 es un diagrama de secuencias de cuadrado latino $L_{jq}(i)$. Las secuencias de cuadrado latino $L_{jq}(i)$ se utilizan para determinar un patrón de división de la columna $j \times 32 + q$ -ésima de la matriz que se va a ampliar. Por ejemplo, la columna 670-ésima $g_{670}(1)$ de EG $(2, 2^5)$ que será recortada por eliminación se determina como sigue:

$$g_{670}(1) = \{28 \ 48 \ 84 \ 113 \ 153 \ 220 \ 225 \ 234 \ 268 \ 280 \ 283 \ 284 \ 322 \ 363 \ 374 \ 436 \ 497 \ 507 \ 525 \ 593 \ 600 \ 617 \ 623 \ 025 \ 644 \ 670 \ 701 \ 783 \ 805 \ 818 \ 892 \ 929\}.$$

Esto se divide en cinco columnas de peso 6 y una columna de peso 2.

El correspondiente cuadrado latino $L_{jq}(i)$ es $20 \times 32 + 30 = 670$. Por lo tanto, se obtiene el siguiente cuadrado latino:

$$L_{21,30}(1) = \{13 \ 19 \ 9 \ 10 \ 16 \ 24 \ 25 \ 28 \ 23 \ 5 \ 8 \ 12 \ 31 \ 14 \ 30 \ 21 \ 4 \ 6 \ 17 \ 7 \ 15 \ 29 \ 2 \ 3 \ 27 \ 22 \ 26 \ 18 \ 1 \ 20 \ 32 \ 11\}.$$

Como resultado, el patrón de división se convierte en lo que sigue:

$$\begin{aligned} g_{670,1}(1) &= g_{670}(L_{21,30}(1)) = \{322 \ 525 \ 268 \ 280 \ 436 \ 625\} \quad i=1, 2, \dots, 6 \\ g_{670,2}(1) &= g_{670}(L_{21,30}(1)) = \{644 \ 783 \ 623 \ 153 \ 234 \ 284\} \quad i=7, 8, \dots, 12 \\ g_{670,3}(1) &= g_{670}(L_{21,30}(1)) = \{892 \ 363 \ 818 \ 600 \ 113 \ 220\} \quad i=13, 14, \dots, 16 \\ g_{670,4}(1) &= g_{670}(L_{21,30}(1)) = \{497 \ 225 \ 374 \ 805 \ 48 \ 84\} \quad i=17, 18, \dots, 24 \\ g_{670,5}(1) &= g_{670}(L_{21,30}(1)) = \{701 \ 617 \ 670 \ 507 \ 28 \ 593\} \quad i=25, 26, \dots, 30 \\ g_{670,6}(1) &= g_{670}(L_{21,30}(1)) = \{929 \ 283\} \quad i=31, 32. \end{aligned}$$

En general, los elementos $L_{jq}(i)$ del cuadrado latino para $g_{c,e}(l)$ se determinan basándose en la siguiente ecuación (24).

$$\begin{aligned} j &= \lfloor c/d_v \rfloor \\ g &= ((c-1) \bmod d_v) + 1 \end{aligned} \quad (24)$$

Las características de los códigos de LDPC anteriores se comparan a continuación. La Fig. 15 es un diagrama de una relación entre una E_b/N_0 (una relación de potencia de señal a potencia de ruido por un bit de información) y una tasa de errores de bit (BER). Un método de decodificación es un "algoritmo suma-producto". Esta característica utiliza el conjunto que se muestra en la Fig. 11. La Fig. 15 es una comparación de características entre la ejecución de la división regular como se describe en la segunda literatura y la ejecución de la división de acuerdo a un cuadrado latino de secuencias aleatorias.

Como se desprende de la Fig. 15, de acuerdo con la división regular como se describe en la segunda literatura, no se puede esperar una gran mejora, incluso con "códigos de LDPC irregulares". Por el contrario, la división aleatoria

de la presente realización puede proporcionar un rendimiento notablemente mejorado cuando se implementa porque la probabilidad de aparición de un bucle se reduce sustancialmente.

5 Como se explicó anteriormente, de acuerdo con la presente realización, en primer lugar, se determina una tasa de codificación. A continuación, se crea una "matriz básica basada en una estructura reticular de enteros" que tiene pesos constantes de filas y columnas y un número mínimo de ciclos de 8. La matriz básica creada es sustituida sobre la base de una ecuación de relación específica. Un conjunto de códigos de LDPC irregulares se ha buscado provisionalmente por la aproximación de Gauss sobre la base de una condición antes de la eliminación de filas. Se eliminan las filas en orden desde la parte inferior de la matriz básica después de la permutación, considerando el
 10 número de filas después de una división. Un conjunto óptimo de códigos de LDPC irregulares se busca por aproximación Gaussiana sobre la base de una condición después de la eliminación de filas. Por último, el peso de la matriz básica después de la eliminación de filas se divide de forma aleatoria de acuerdo con un procedimiento predeterminado sobre la base de este conjunto. Con esta disposición, se puede generar fácilmente una matriz de comprobación definida y de característica estabilizada para "códigos de LDPC irregulares" en un corto período de
 15 tiempo correspondiente a un conjunto opcional, una longitud opcional de código y una tasa opcional de codificación.

Como se explicó anteriormente, de acuerdo con la presente realización, en primer lugar, se determina una tasa de codificación. A continuación, se crea una matriz básica que tiene pesos constantes de filas y columnas y un número de ciclos igual o mayor que 6. La matriz básica creada es sustituida sobre la base de una ecuación de relación específica. Un conjunto de códigos de LDPC irregulares se ha buscado provisionalmente por la aproximación de Gauss sobre la base de una condición antes de la eliminación de filas. Se eliminan las filas en orden desde la parte inferior de la matriz básica después de la permutación, considerando el número de filas después de una división. Un conjunto óptimo de códigos de LDPC irregulares se busca por aproximación Gaussiana sobre la base de una
 20 condición después de la eliminación de filas. Por último, el peso de la matriz básica después de la eliminación de filas se divide de forma aleatoria de acuerdo con un procedimiento predeterminado sobre la base de este conjunto óptimo. Con esta disposición, existe el efecto de que se puede generar fácilmente una matriz de comprobación definida y de característica estabilizada para "códigos de LDPC irregulares" en un corto período de tiempo correspondiente a un conjunto opcional, una longitud opcional de código y una tasa opcional de codificación.

30 **APLICABILIDAD INDUSTRIAL**

Como se explicó anteriormente, el método para la generación de matrices de comprobación para códigos de LDPC y el aparato de generación de matrices de comprobación de acuerdo con la presente invención son útiles para el sistema de comunicación que utiliza los códigos de LDPC como códigos de corrección de errores. En particular, el método y el aparato son adecuados para un aparato de comunicación que busca los códigos de LDPC definidos y
 35 de características estabilizadas.

REIVINDICACIONES

1. Un método de comunicación, comprendiendo el método las etapas de:

5 generar una matriz de comprobación para un código de comprobación de paridad de baja densidad irregular; utilizar la matriz de comprobación generada para la codificación de un mensaje generando con ello una palabra de código; y modular la palabra de código generada;

10 en el que la etapa del método de generación de una matriz de comprobación comprende las siguientes etapas:

determinar (S1) una tasa de codificación; y **caracterizado porque** comprende además:
 generar (S2) una matriz básica que satisface las condiciones de que "los pesos de las filas y columnas son constantes" y "la longitud del ciclo más pequeño es igual o superior a seis";
 15 determinar (S3) el número de columnas y el número de filas de la matriz de comprobación que finalmente se obtendrá;
 reorganizar (S4) filas de la matriz básica generada, sobre la base de una ecuación de relación específica;
 obtener (S5) provisionalmente un conjunto de pesos de fila y pesos de columna del código de comprobación de paridad de baja densidad mediante la ejecución de una optimización por aproximación Gaussiana sobre la
 20 base de una condición predeterminada antes de la eliminación de filas;
 eliminar (S6) filas de la matriz básica reorganizada en orden desde la parte inferior teniendo en cuenta el número de filas después de una división, es decir, el número de filas de la matriz de comprobación que se obtendrá finalmente;
 obtener (S7) un conjunto óptimo de pesos de fila y de pesos de columna del código de comprobación de
 25 paridad de baja densidad mediante la ejecución de la optimización por la aproximación Gaussiana sobre la base de una condición predeterminada después de la eliminación de filas; y
 dividir aleatoriamente las filas y las columnas de la matriz básica después de la eliminación de filas basándose en el conjunto óptimo, en el que:
 se genera una matriz básica sobre la base de una estructura reticular de enteros que satisface las
 30 condiciones de que "los pesos de las filas y las columnas son constantes" y "la longitud del ciclo más pequeño es ocho", ya que la matriz básica que satisface las condiciones de que "los pesos de las filas y las columnas son constantes" y "la longitud del ciclo más pequeño es igual o superior a seis".

35 2. El método según la reivindicación 1, en el que la ecuación relacional específica utilizada en la etapa de reorganización del método es una ecuación que puede reorganizar las filas de la matriz básica generada de modo que se sitúan en posiciones más altas en las columnas.

40 3. El método según la reivindicación 1, en el que, en la aproximación Gaussiana, se obtiene un conjunto de pesos óptimos de fila y un conjunto de pesos óptimos de columna utilizando una sola programación lineal para maximizar el ruido Gaussiano con tasa fija de codificación.

45 4. El método según cualquiera de las reivindicaciones 1 a 3, en el que, en la etapa de división del método, se genera un cuadrado latino de una secuencia aleatoria básica, y se extrae un "1" de cada fila y cada columna en la matriz básica después de la eliminación de filas, dividiendo con ello cada columna y cada fila aleatoriamente basándose en el cuadrado latino.

5. Un aparato que comprende:

50 unos medios para generar una matriz de comprobación para un código de comprobación de paridad de baja densidad irregular;
 un codificador adaptado para utilizar la matriz de comprobación generada para la codificación de un mensaje adaptado con ello para generar una palabra de código; y
 un modulador para modular la palabra de código generada;
 los medios para generar una matriz de comprobación para un código de comprobación de paridad de baja
 55 densidad comprenden:
 una unidad de determinación de tasa de codificación configurada para determinar una tasa de codificación;

caracterizada porque comprende además:

60 una unidad de generación de matriz básica configurada para generar una matriz básica que satisface las condiciones de que "los pesos de las filas y columnas son constantes" y "la longitud del ciclo más pequeño es igual o superior a seis";
 una unidad de reorganización configurada para reorganizar las filas de la matriz básica generada, basándose en una ecuación de relación específica,

- una primera unidad de obtención de pesos configurada para obtener provisionalmente un conjunto de pesos de fila y pesos de columna del código de comprobación de paridad de baja densidad mediante la ejecución de una optimización por aproximación Gaussiana sobre la base de una condición predeterminada antes de la eliminación de filas;
- 5 una unidad de eliminación de filas configurada para eliminar filas de la matriz básica reorganizada en orden desde la parte inferior teniendo en cuenta el número de filas después de una división, es decir, el número de filas de la matriz de comprobación que se obtendrá finalmente;
- una segunda unidad de obtención de pesos configurada para obtener un conjunto óptimo de pesos de fila y pesos de columna del código de comprobación de paridad de baja densidad mediante la ejecución de una optimización por aproximación Gaussiana sobre la base de una condición predeterminada antes de la eliminación de filas; y
- 10 una unidad de división configurada para dividir aleatoriamente las filas y las columnas de la matriz básica después de la eliminación de filas basada en el conjunto óptimo; en el que:
- 15 dicha unidad de generación de matriz básica se configura para generar una matriz básica sobre la base de una estructura reticular de enteros que satisface las condiciones de que "los pesos de las filas y las columnas son constantes" y "la longitud del ciclo más pequeño es ocho", ya que la matriz básica que satisface las condiciones de que "los pesos de las filas y las columnas son constantes" y "la longitud del ciclo más pequeño es igual o superior a seis".

FIG.1

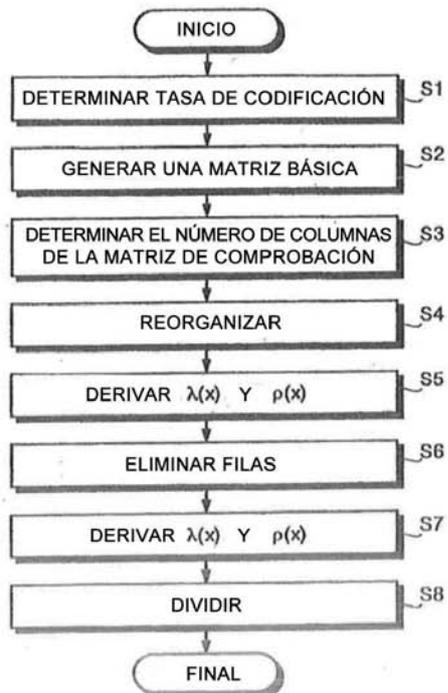


FIG.2

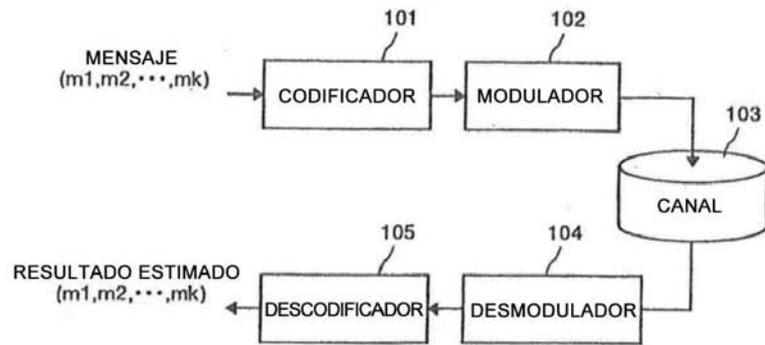


FIG.3

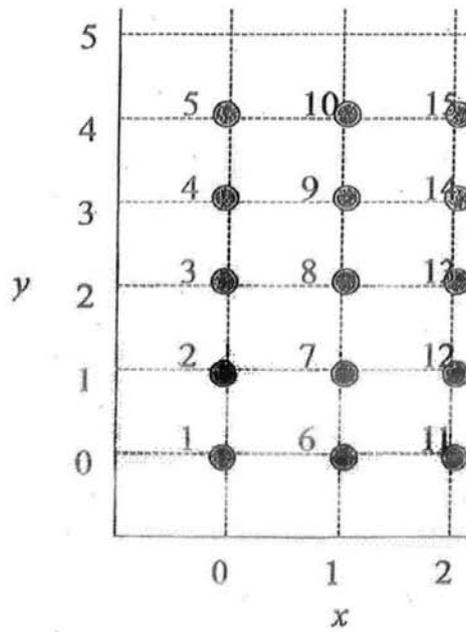


FIG.4

	s=0	s=1	s=2	s=3	s=4
CLASE	{1,6,11}	{1,7,13}	{1,8,15}	{1,9,12}	{1,10,14}
	{2,7,12}	{2,8,14}	{2,9,11}	{2,10,13}	{2,6,15}
	{3,8,13}	{3,9,15}	{3,10,12}	{3,6,14}	{3,7,11}
	{4,9,14}	{4,10,11}	{4,6,13}	{4,7,15}	{4,8,12}
BLOQUE	{5,10,15}	{5,6,12}	{5,7,14}	{5,8,11}	{5,9,13}

FIG.5

```

s=0, S={s}, B=B(s), S'={1,2,...,m-1}.
While S'≠emptyset
s=s+1
if g(V,B'∪B(s))=8
  S=SU{s}
  S'=S'\{s}
  B'=B'∪B(s)
else
  S'=S'\{s}
end
end
    
```

FIG.6

k	m	S	N= B	M= V	(dv,dc)
3	5	0,1	10	15	(3,2)

FIG.7

$$\begin{array}{l}
 p_1 \\
 p_2 \\
 p_3 \\
 p_4 \\
 p_5 \\
 p_6 \\
 p_7 \\
 p_8 \\
 p_9 \\
 p_{10} \\
 p_{11} \\
 p_{12} \\
 p_{13} \\
 p_{14} \\
 p_{15}
 \end{array}
 \begin{pmatrix}
 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0
 \end{pmatrix}$$

FIG.8

k	m	S	$N= B $	$M= V $	(dv,dc)
10	353	0,1,10,11,23,24,224	2471	3530	(10,7)

FIG.9

```

for  $i = 1$  to  $|V|$ 
 $R_{k^{*((i-1) \bmod m) + [(i-1)/m] + 1}} = R_i$ 
end
    
```

FIG.10

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

FIG.11

Tasa	0,5		
	x	λ_x	N°
	2	0,28647619	3008
	3	0,264571429	1852
	4	0,001142857	6
	8	0,138285714	484
	10	0,30952381	650
	x	ρ_x	N°
	7	1	3000
σ_{GA}	0,940358043		

FIG.12

	1	2	3	4	5	6	7	8	9	10	11	12	13
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													

FIG.13

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
C(1)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
C(2)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
C(3)	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	14	17	1	3	5	7	9	11	13	15	17	19	21	23	25	27	
C(4)	3	6	9	12	15	18	21	24	27	30	7	11	2	5	8	11	31	16	20	23	26	29	32	22	1	4	7	10	13	16	19	22	
C(5)	4	8	12	16	20	24	28	32	8	3	18	23	15	19	23	11	15	2	6	10	14	18	9	28	30	24	1	5	9	13	17		
C(6)	5	10	15	20	25	30	5	3	17	13	28	10	28	10	1	6	28	14	21	26	31	21	4	20	14	18	14	28	26	2	7	12	
C(7)	6	12	18	24	30	5	12	11	26	23	3	22	4	24	16	22	8	32	3	9	15	6	27	7	2	8	4	20	18	32	1	7	
C(8)	7	14	21	28	3	11	19	7	6	14	9	17	1	31	1	25	13	22	29	20	28	13	27	31	11	10	25	32	2	8	10	18	32
C(9)	8	16	24	32	8	17	25	16	16	25	21	30	15	9	17	5	31	4	12	4	13	22	18	15	12	21	2	2	18	26	28		
C(10)	9	18	27	3	12	20	3	6	25	26	10	8	6	28	24	12	22	12	23	32	25	20	8	5	3	1	11	30	31	11	20	24	
C(11)	10	20	30	7	18	28	10	14	9	21	20	19	6	2	28	2	30	5	15	9	5	31	29	28	27	1	21	23	4	14	18		
C(12)	11	22	2	11	23	4	17	22	15	18	32	32	20	17	7	19	11	24	18	30	27	17	16	16	16	12	15	27	8	14			
C(13)	12	24	5	15	28	10	24	30	24	28	6	7	8	11	32	23	16	29	6	1	14	12	3	3	4	5	18	3	7	20	2	9	
C(14)	13	26	8	18	1	16	31	1	5	2	17	19	21	25	10	2	13	10	25	21	19	19	20	27	29	31	8	31	28	13	27	4	
C(15)	14	28	11	23	6	22	1	9	14	12	26	31	10	2	25	18	30	28	7	4	3	4	12	14	17	20	25	22	20	6	21	31	
C(16)	15	30	14	27	11	22	8	17	23	22	2	6	23	15	3	13	10	9	26	24	24	26	21	1	5	8	15	13	12	28	15	25	
C(17)	16	32	17	31	18	3	15	25	32	32	13	18	12	30	18	28	27	7	8	11	7	25	30	24	5	4	4	22	9	21			
C(18)	17	1	20	2	21	9	22	4	5	24	30	25	7	11	8	7	8	27	27	28	18	30	12	18	13	32	32	25	15	3	16		
C(19)	18	3	23	6	26	15	23	12	13	15	9	5	1	21	26	24	26	9	10	13	3	16	23	8	2	22	23	17	8	28	11		
C(20)	19	5	28	10	31	21	8	20	22	25	20	17	14	12	4	3	4	7	28	30	18	25	2	10	31	28	12	14	9	1	22	6	
C(21)	20	7	32	18	8	2	20	7	3	18	5	4	3	12	14	1	6	29	16	23	32	11	8	7	6	28	24	30	24	10	28		
C(22)	21	9	32	18	8	2	20	7	3	18	5	4	3	12	14	1	6	29	16	23	32	11	8	7	6	28	24	30	24	10	28		
C(23)	22	11	1	22	14	8	27	15	12	28	16	16	17	27	30	18	24	11	19	7	17	20	32	32	18	15	22	17	4	23			
C(24)	23	13	4	28	19	14	4	23	21	1	27	28	29	31	5	9	15	5	30	2	28	2	6	18	20	21	9	6	14	10	29	18	
C(25)	24	15	7	30	24	20	11	31	30	11	1	3	5	20	25	32	23	12	22	12	24	23	6	8	10	26	25	6	3	23	13		
C(26)	25	17	10	1	28	20	18	2	2	21	12	15	18	22	13	4	12	4	31	5	17	9	15	30	21	25	16	16	27	20	17	8	
C(27)	26	19	13	5	2	32	25	10	11	31	23	27	31	13	28	20	22	13	25	1	31	1	17	9	14	6	7	18	19	11	3		
C(28)	27	21	16	9	7	1	32	18	20	4	8	2	7	27	6	15	9	3	32	8	22	16	24	4	22	3	23	26	11	12	5	30	
C(29)	28	23	19	13	12	7	2	28	28	14	19	14	20	4	21	31	26	21	14	28	6	1	10	28	10	28	13	17	3	5	30	25	
C(30)	29	25	22	17	13	9	5	1	24	30	28	8	18	14	10	6	2	15	11	27	18	15	23	18	3	8	32	28	24	20			
C(31)	30	27	25	21	22	18	16	13	10	7	4	1	22	32	28	26	23	20	16	31	11	8	5	2	11	7	30	27	24	21	18	15	
C(32)	31	29	25	27	25	21	19	17	15	13	11	9	7	5	3	17	14	32	30	28	26	24	22	20	18	16	14	12	10				
C(33)	32	31	31	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	

Secuencia aleatoria básica

patrón de permutación de la secuencia aleatoria básica

FIG.14

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
L ₄ (1)	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9
L ₄ (2)	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10
L ₄ (3)	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16
L ₄ (4)	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24
L ₄ (5)	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25
L ₄ (6)	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28
L ₄ (7)	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23
L ₄ (8)	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5
L ₄ (9)	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8
L ₄ (10)	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12
L ₄ (11)	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31
L ₄ (12)	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14
L ₄ (13)	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30
L ₄ (14)	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21
L ₄ (15)	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4
L ₄ (16)	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6
L ₄ (17)	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17
L ₄ (18)	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7
L ₄ (19)	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15
L ₄ (20)	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28
L ₄ (21)	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2
L ₄ (22)	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3
L ₄ (23)	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27
L ₄ (24)	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22
L ₄ (25)	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26
L ₄ (26)	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18
L ₄ (27)	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1
L ₄ (28)	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20
L ₄ (29)	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32
L ₄ (30)	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11
L ₄ (31)	18	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13
L ₄ (32)	8	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19

FIG.15

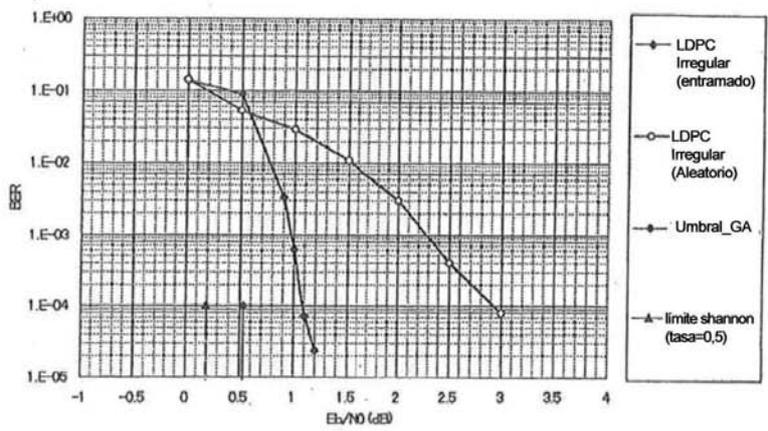


FIG.16

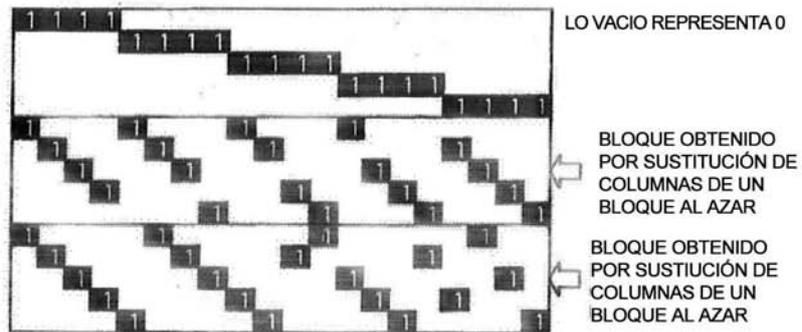


FIG.17

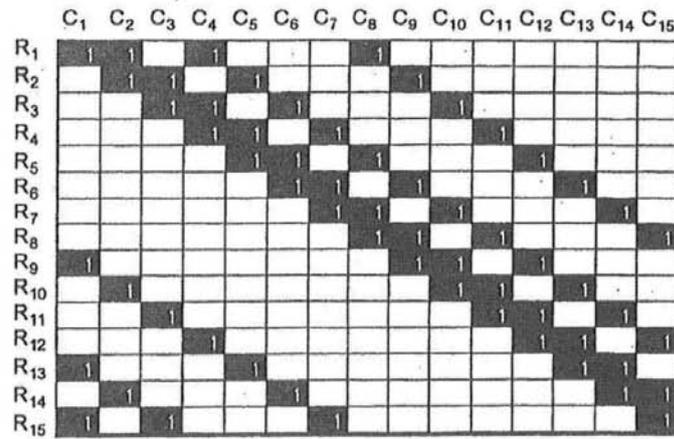


FIG.18

