(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 383 835**

(51) Int. CI.:
H03M 13/11 (2006.01)
H03M 13/39 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **07870358 .4**
(96) Fecha de presentación: **29.11.2007**
(97) Número de publicación de la solicitud: **2095512**
(97) Fecha de publicación de la solicitud: **02.09.2009**

(54) Título: **Procedimiento y dispositivo de decodificación para códigos LDPC, y aparato de comunicación que comprende dicho dispositivo**

(30) Prioridad:
01.12.2006 FR 0655275

(45) Fecha de publicación de la mención BOPI:
26.06.2012

(45) Fecha de la publicación del folleto de la patente:
26.06.2012

(73) Titular/es:
COMMISSARIAT A L'ÉNERGIE ATOMIQUE ET AUX ÉNERGIES ALTERNATIVES
BÂTIMENT "LE PONANT D" 25, RUE LEBLANC
75015 PARIS, FR

(72) Inventor/es:
SAVIN, Valentin

(74) Agente/Representante:
de Elzaburu Márquez, Alberto

DESCRIPCIÓN

Procedimiento y dispositivo de decodificación para códigos LDPC, y aparato de comunicación que comprende dicho dispositivo.

La presente invención se refiere a un procedimiento y un dispositivo de decodificación, así como a un aparato de comunicación que comprende dicho dispositivo.

La codificación de palabras de información (de una longitud generalmente indicada como K) en palabras de código (de una longitud indicada generalmente como N, con N > K) se utiliza cuando se desea añadir redundancia a las informaciones referidas con el fin de recuperar la totalidad de las informaciones de origen aunque una parte de las mismas sea errónea o se haya perdido, como en el caso de la transmisión de informaciones en un canal con perturbaciones o de su almacenamiento en un soporte sujeto a degradaciones (tales como arañazos en un disco óptico).

La decodificación llevada a cabo en la recepción (o en la lectura en el caso del almacenamiento) con el fin de recuperar las palabras de información de origen comprende en general una primera fase de corrección de errores, que consiste en recuperar de forma precisa la palabra de código emitida (o almacenada) a partir de la palabra recibida (vocabulario usado también para designar la palabra leída en el caso del almacenamiento), a pesar de eventuales errores y gracias a la redundancia introducida, y a continuación una fase de "*demapping*" que consiste en realizar la operación inversa a la efectuada durante la codificación.

En este contexto, se conocen los códigos de matriz de control de paridad con baja densidad, denominados en lo sucesivo códigos LDPC (según la abreviatura de la denominación anglosajona "*Low Density Parity Check*"), tales como los descritos por ejemplo en el artículo "*Low density parity check codes*", de R. Gallager en IEEE Trans. Inform. Theory, vol., IT-8, págs. 21 a 28, 1962.

Estos códigos son particularmente interesantes no solo porque están muy próximos a la capacidad del canal (límite de Shannon) y ofrecen el mejor compromiso posible entre rendimiento y prestaciones (próximos al límite de Gilbert-Varshamov), sino también porque admiten una decodificación iterativa de tipo paso de mensaje.

Hasta el momento se han propuesto dos algoritmos principales de decodificación de los códigos LDPC, ya sean códigos binarios (para los cuales los símbolos que representan las informaciones son 0 ó 1, es decir, adoptados en el campo de Galois ($GF(2)$) o códigos no binarios (para los cuales los símbolos se adoptan en el campo de Galois $GF(q)$ con $q > 2$).

El primero, propuesto en el artículo citado previamente con el nombre de "*probabilistic decoding*" se conoce generalmente (y se denomina en lo sucesivo) como decodificación SPA (de "*Sum – Product Algorithm*") o BP (de "*Belief Propagation*"). Este algoritmo se califica generalmente como *óptimo* puesto que la decodificación SPA converge hacia la máxima verosimilitud a condición de que el grafo bipartito asociado al código LDPC no contenga ciclos. En el caso de los códigos LDPC no binarios, este algoritmo es sin embargo inservible en un sistema real de comunicaciones a causa de su muy amplia dinámica, lo cual acarrea una fuerte inestabilidad de los cálculos efectuados. Además, requiere la realización de un número importante de productos, lo cual lo hace complejo, y depende del conocimiento del ruido térmico.

El segundo algoritmo de decodificación, subóptimo, es conocido principalmente con el nombre de MSA (de "*Min-Sum Algorithm*"). Es menos complejos que el SPA y es independiente del conocimiento del ruido térmico, aunque presenta, en términos de tasa de errores de bit, una pérdida de prestaciones con respecto al SPA que varía generalmente de 0,25 dB a 1 dB para un canal AWGN (del inglés "*Additive White Gaussian Noise*" de ruido blanco gaussiano aditivo), en función del rendimiento, de la irregularidad o de la longitud del código utilizado.

Así, en este contexto, la invención pretende especialmente proponer una solución para la decodificación de los códigos LDPC, en particular no binarios, que combine buenas prestaciones (por ejemplo con respecto al MSA) y una complejidad reducida con respecto al algoritmo óptimo (algoritmo SPA).

Así, la invención propone un procedimiento de decodificación iterativa de una palabra recibida, representada por valores de una señal, según un código de matriz de control de paridad, del tipo de paso de mensajes entre dos nodos de variable y dos nodos de control de un grafo bipartito asociado a dicha matriz, caracterizado porque comprende una etapa de inicialización de por lo menos un mensaje de un nodo de variable, en función de dichos valores, mediante una información representativa de la relación entre la probabilidad de tener el símbolo más probable en la posición correspondiente al nodo de variable y la probabilidad de tener el símbolo actual en dicha posición.

Por otra parte, se puede prever según este procedimiento que el mismo comprenda por lo menos una de las etapas siguientes:

- determinación de por lo menos un mensaje, referente a un símbolo determinado, de un nodo de control hacia un nodo de variable determinado, como el valor mínimo adoptado, entre las secuencias de símbolos que verifican la

ecuación en el nodo de control utilizando dicho símbolo determinado en el nodo de variable determinado, por el valor máximo de los mensajes recibidos en el nodo de control desde nodos de variable que no sean el nodo de variable determinado y referentes, cada uno de ellos, al símbolo asociado a este otro nodo de variable en la secuencia que verifica la ecuación;

- determinación de los mensajes de un nodo de variable hacia un nodo de control referentes al conjunto de los símbolos de tal manera que el valor mínimo de dichos mensajes sea nulo.

La solución así propuesta (denominada en lo sucesivo MMA de "*Min-Max Algorithm*") permite obtener buenas prestaciones para una complejidad reducida, tal como se desprenderá a partir de ejemplos de realización presentados posteriormente.

En la práctica, la etapa de inicialización del mensaje de un nodo de variable comprende, por ejemplo, las siguientes etapas:

- para cada símbolo del alfabeto, determinación de la suma de las relaciones de verosimilitud logarítmica binarias referentes a los bits no nulos del símbolo y en la posición correspondiente al nodo de variable y;

- determinación del mínimo de las sumas determinadas;

- resta, de cada suma determinada, del mínimo determinado.

Por otro lado, se puede prever según este procedimiento una etapa de determinación de una información *a posteriori* referente a un nodo de variable y a un símbolo, como la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable y referentes al símbolo, en cuyo caso el final del procedimiento iterativo se puede determinar de la manera siguiente:

- para cada nodo de variable, determinación del símbolo para el cual la información *a posteriori* es mínima;

- si la secuencia de los símbolos así determinados para el conjunto de los nodos de variable es una palabra de código, utilización de dicha palabra de código como palabra estimada.

Según un modo de realización de la etapa de determinación de los mensajes de un nodo de variable hacia un nodo de control dado previsto más arriba, el mismo comprende las siguientes etapas:

- para cada símbolo, determinación de la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable procedentes de un nodo de control que no sea el nodo de control dado y referentes al símbolo;

- determinación del mínimo de entre las sumas determinadas;

- para cada símbolo, resta del mínimo determinado, de la suma determinada referente al símbolo.

Para la realización práctica de la etapa de determinación de por lo menos un mensaje de un nodo de control hacia un nodo de variable determinado, el mismo puede comprender, por ejemplo, las siguientes etapas:

- determinación, para cada símbolo, de un valor intermedio igual al valor mínimo adoptado por el valor máximo de mensajes recibidos en el nodo de control desde una parte solamente de los nodos de variable asociados al nodo de control;

- determinación del valor mínimo adoptado por el valor máximo de entre los valores intermedios y los mensajes recibidos de un nodo de variable asociado al nodo de control y que no pertenecen a dicha parte.

De forma más precisa, estas dos etapas se pueden describir de la manera siguiente:

- determinación, para cada símbolo, de un valor intermedio igual al valor mínimo adoptado, de entre las secuencias de símbolos asociados a una parte solamente de los nodos de variable conectados a dicho nodo de control y tales que la suma de los símbolos de la secuencia multiplicados por los coeficientes correspondiente en la matriz a dichos nodos de variable sea igual a dicho símbolo determinado, por el valor máximo de los mensajes recibidos en el nodo de control desde los nodos de variable de dicha parte y referentes, cada uno de ellos, al símbolo asociado a este nodo de variable en dicha secuencia;

- determinación de manera recursiva, para cada símbolo, de un valor intermedio *nuevo* igual al valor mínimo adoptado, de entre las parejas de símbolos cuyo primer símbolo está asociado a un valor intermedio calculado *previamente* y correspondiente a una parte de los nodos de variable y cuyo segundo símbolo está asociado a un nodo de variable *nuevo* que no pertenece a dicha parte, y tales que la suma del primer símbolo y del segundo símbolo multiplicado por el coeficiente en la matriz del nodo de variable correspondiente sea igual a dicho símbolo determinado, por el valor máximo entre el valor intermedio calculado previamente referente al primer símbolo de dicha pareja y el mensaje recibido en el nodo de control desde el nodo de variable nuevo y referente al segundo

símbolo de dicha pareja. El valor intermedio nuevo, calculado de esta manera, se corresponde con una *parte nueva* de nodos de variable que comprenden *la parte antigua y el nodo nuevo de variable*.

Se trata de un modo de realización particularmente práctico, denominado en lo sucesivo «forward - backward» (del inglés que significa "*hacia delante, hacia atrás*"), que permite limitar (preferentemente a dos) el número de nodos de variables (es decir de dimensiones) sobre las cuales se debe buscar el mínimo del máximo.

Por otra parte, según un modo de realización propuesto posteriormente ("*implementación selectiva*"), el procedimiento comprende una etapa de determinación del mínimo, entre las parejas de dos símbolos que pertenecen al alfabeto del código, del máximo de dos valores asociados respectivamente a los dos símbolos de cada pareja, que comprende las siguientes subetapas:

- determinación de conjuntos que agrupan, cada uno de ellos, los símbolos a los cuales están asociados valores comprendidos en un intervalo determinado;

- selección de conjuntos, de entre los conjuntos determinados, de tal manera que la reunión de los conjuntos seleccionados contiene por lo menos un número predeterminado de símbolos (siendo suficiente este número predeterminado para el cálculo del mínimo de los máximos considerados aquí y explicado posteriormente con el nombre de "*fórmula min-max*");

- utilización, como máximo entre dos valores, del valor asociado al símbolo comprendido en el conjunto correspondiente al intervalo superior cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a intervalos distintos;

- determinación por comparación del máximo entre dos valores cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a un mismo intervalo.

El agrupamiento de los símbolos por conjuntos según el valor adoptado, en la práctica mediante la utilización de la parte entera de este valor, permite reducir claramente el número de símbolos implicados y simplificar las operaciones de comparación de los valores asociados a los símbolos implicados.

El código es, por ejemplo, un código de alfabeto no binario, en cuyo caso el procedimiento propuesto es particularmente interesante.

La invención propone también un dispositivo de decodificación iterativa de una palabra recibida, representada por valores de una señal, según un código de matriz de control de paridad, del tipo de paso de mensajes entre dos nodos de variable y dos nodos de control de un grafo bipartito asociado a dicha matriz, caracterizado porque comprende:

- medios de inicialización de por lo menos un mensaje de un nodo de variable, en función de dichos valores, mediante una información representativa de la relación entre la probabilidad de tener el símbolo más probable en la posición correspondiente al nodo de variable y la probabilidad de tener el símbolo actual en dicha posición.

Este dispositivo puede constar de características opcionales correspondientes a las previstas anteriormente para el procedimiento de decodificación y las ventajas asociadas.

La invención propone finalmente un aparato de comunicación que comprende dicho dispositivo.

Se pondrán de manifiesto otras características y ventajas de la invención considerando la descripción que se ofrece seguidamente, realizada en referencia a los dibujos adjuntos en los cuales:

- la figura 1 ofrece a título ilustrativo un ejemplo de matriz de paridad y de grafo asociado;

- la figura 2 representa los elementos principales de un sistema de comunicación que comprende un receptor de acuerdo con las enseñanzas de la invención;

- la figura 3 representa esquemáticamente la técnica de paso de mensajes entre los nodos de variable y los nodos de control;

- la figura 4 ilustra el método «forward - backward» utilizado en el primer modo de realización;

- la figura 5 ofrece a título explicativo una ilustración de las técnicas propuestas en el segundo modo de realización;

- la figura 6 representa una comparación de las prestaciones obtenidas al utilizar el decodificador MSA y el correspondiente propuesto por la invención para códigos LDPC binarios y no binarios;

- la figura 7 representa una comparación de las prestaciones obtenidas al utilizar los dos modos de realización presentados posteriormente en este documento;

- la figura 8 representa una comparación de las complejidades de la decodificación MSA y de la decodificación

propuesta por la invención para códigos binarios y no binarios.

En lo sucesivo se presentan dos ejemplos de realización de la invención, con las siguientes notaciones comunes:

- p , un entero estrictamente positivo denominado *dimensión (o grado de extensión) del campo de Galois*;

- $q = 2^p$, denominado *cardinal del campo de Galois*;

- $P(X)$ un polinomio irreducible de grado *p* con coeficientes en $GF(2) = \{0,1\}$;

- $GF(q) = \{0,1,\ldots,q-1\}$, el *campo de Galois* de q *elementos*, denominándose *símbolos* los elementos de $GF(q)$.

Se observa que el campo de Galois $GF(q)$ se define típicamente como el cociente del anillo de polinomios $GF(2)[X]$ por el ideal generado por el polinomio $P(X)$, aunque, en este caso, por simplificar la exposición, el campo de Galois $GF(q)$ se identifica con el conjunto de los enteros $\{0,1,...,q-1\}$ provisto de las operaciones «suma» y «producto» descritas más abajo (aunque cada elemento del campo de Galois se denomina *símbolo* con el fin de diferenciarlo de un «simple» entero).

Recuérdese en este momento que, con el fin de definir las operaciones «suma» y «producto» que dotan a $GF(q)$ de una estructura de campo, cada símbolo $a \in GF(q)$ se identifica con una secuencia binaria $a_0 a_1 \ldots a_{p-1}$ correspondiente a la escritura de a en base 2, dicho de otra manera $a = \sum_{i=0}^{p-1} a_i 2^i, a_i \in \{0,1\}$. Además, al símbolo a se le asocia el polinomio $a(X) = \sum_{i=0}^{p-1} a_i X^i$, con la consecuencia de que $a(2) = \sum_{i=0}^{p-1} a_i 2^i = a$.

Los valores binarios $a_0 a_1 \ldots a_{p-1}$ se denominarán en lo sucesivo *los bits del símbolo* $a \in GF(q)$.

La suma de dos símbolos $a,b \in GF(q)$, indicada $a \oplus b$, se puede definir entonces mediante la suma bit a bit de los símbolos a y b , dicho de otra manera $(a \oplus b)_i = a_i + b_i \mod 2$.

El producto de dos símbolos $a,b \in GF(q)$, indicado $a \otimes b$, se define por su parte mediante el producto de los polinomios $a(X)$ y $b(X)$ módulo $P(X)$, dicho de otra manera $(a \otimes b)(X) = a(X) \cdot b(X) \bmod P(X)$.

Un código LDPC sobre $GF(q)$ es un código lineal definido por una *matriz de paridad* $H \in M_{M,N}(GF(q))$ con baja densidad de elementos no nulos (donde $M_{M,N}(GF(q))$ es el conjunto de las matrices de M filas y N columnas con coeficientes en $GF(q)$). Se pueden introducir entonces las siguientes nociones:

- N , la *longitud de código*, se corresponde con el número de columnas de la matriz H ;

- M , el *número de controles de paridad,* se corresponde con el número de filas de la matriz H ;

- $K = N - M$, la *dimensión del código*, se corresponde con la longitud de la palabra de información;

- $x = (x_1, x_2 \ldots\ldots, x_N)$ una *palabra de código q-aria*, es decir, una secuencia de símbolos $x_n \in GF(q)$, tales que $H \cdot x' = 0$, donde x' es el vector transpuesto de x , es decir, tales que $\underset{n=1,\ldots\ldots,N}{\oplus} h_{m,n} \otimes x_n = 0, \forall m = 1,\ldots,M$, donde $h_{m,n}$ es el elemento de la fila m y la columna n de la matriz H ;

- $(x_{1,0}, x_{1,1}, \ldots, x_{1,p-1}, x_{2,0}, x_{2,1}, \ldots x_{2,p-1}, \ldots\ldots, x_{N,0}, x_{N,1}, \ldots, x_{N,p-1})$, donde $x_{n,0}, x_{n,1}, \ldots, x_{n,p-1}$ son los bits del símbolo $x_n \in GF(q)$, la *palabra de código binario* correspondiente;

- $G \in M_{K,N}(GF(q))$, la *matriz generadora* del código, es decir, una matriz de rango K tal que $H \cdot G' = 0$, donde G' es la matriz transpuesta de G ; a una secuencia $(a_1, a_2, \ldots, a_K) \in GF(q)^K$ se le denomina *palabra de información q-aria* y está codificada en $(x_1, x_2, \ldots\ldots, x_N) = (a_1, a_2, \ldots, a_K) \cdot G$.

Se define como grafo bipartito, o grafo de Tanner, asociado a un código LDPC un grafo que contiene $N + M$ nodos divididos en dos tipos:

- N *nodos de tipo variable*, correspondientes a las columnas de la matriz H ;

- M *nodos de tipo control*, correspondientes a las filas de la matriz H ;

y en el cual un nodo de variable está conectado a un nodo de control si y solamente si el elemento correspondiente de la matriz H que define el código es no nulo.

En lo sucesivo se utilizan las siguientes notaciones:

- los índices m,n designan un nodo de control, respectivamente un nodo de variable;

- $H(m)$ designa el conjunto de los nodos de variable conectados al nodo de control m , denominándose *grado del nodo de variable* m a su cardinal.

- $H(n)$ designa el conjunto de los nodos de control conectados al nodo de variable n , denominándose *grado del nodo de variable* n a su cardinal.

Para obtener más detalles sobre este tipo de grafos, puede hacerse referencia al artículo "*A recursive approach to low complexity codes*" de R. M. Tanner en IEEE Trans. Inform. Theory, vol. 27, n.º 5, págs. 533 a 547, 1981.

A título ilustrativo, la figura 1 representa un ejemplo de matriz de paridad de un código LDPC sobre $GF(8)$ y el grafo bipartito asociado. A cada arista del grafo bipartito le corresponde un coeficiente no nulo de H aunque, para no sobrecargar la representación gráfica, se ha omitido la presentación de los coeficientes de H sobre las aristas del grafo.

La figura 2 describe el esquema de un sistema de comunicación que utiliza un código LDPC no binario y que comprende un aparato de recepción de acuerdo con las enseñanzas de la invención.

En la emisión (es decir, en el nivel de un aparato de emisión de las informaciones a transmitir), en particular en el seno de un módulo de codificación, una *palabra de información binaria* $(a_{1,0},a_{1,1},\ldots a_{1,p-1},\ldots\ldots,a_{K,0},a_{K,1},\ldots,a_{K,p-1})$ se transforma en una *palabra de información q -aria* $(a_1,\ldots,a_K)$, donde a_k es el símbolo determinado por los bits $a_{k,0}a_{k,1}\ldots a_{k,p-1}$, y a continuación la palabra de información q-aria se codifica en la *palabra de código q-aria* $(x_1,x_2,\ldots,x_N)=(a_1,\ldots,a_K)\cdot G$. Por lo tanto, la palabra de código q-aria $(x_1,x_2,\ldots,x_N)$ se transforma en una *palabra de código binario* $(x_{1,0},x_{1,1},\ldots x_{1,p-1},x_{2,0},x_{2,1},\ldots x_{2,p-1},\ldots\ldots,x_{N,0},x_{N,1},\ldots x_{N,p-1})$.

En la práctica, naturalmente se puede pasar de manera directa de la palabra de información binaria a la palabra de código binario utilizando una matriz generadora binaria.

La palabra de código binario generada por el módulo de codificación se transmite al módulo de *modulación*, el cual la transforma en una señal a emitir en el *canal de comunicación*.

El aparato de recepción comprende:

- un módulo de *demodulación*, que determina las relaciones de verosimilitud logarítmica binarias (denominadas en lo sucesivo *LLR binarias,* según la formulación anglosajona "*Logarithmic Likelihood Ratio*"), indicadas $\lambda_{n,i}$, $n\in\{1,\ldots,N\}, i\in\{0,\ldots,p-1\}$, correspondientes a la señal recibida en el canal de comunicación según la fórmula:

$$\lambda_{n,i}=\ln\left(\frac{\Pr(x_{n,i}=0|observación\ del\ canal)}{\Pr(x_{n,i}=1|observación\ del\ canal)}\right);$$

- un módulo de corrección de errores (denominado en lo sucesivo *decodificador*), que determina, por medio del procedimiento de decodificación propuesto por la invención y cuyos dos ejemplos se ofrecen posteriormente, una secuencia q-aria de longitud N (igual a la palabra de código q-aria emitida cuando la decodificación resulta satisfactoria) sobre la base de la secuencia de las LLR binarias recibidas del módulo de demodulación;

- un módulo de *demapping* que transforma la secuencia q-aria a la salida del decodificador en una palabra de información, que se corresponde, en caso de resultado satisfactorio de la decodificación, en la palabra de información emitida $(a_1,\ldots,a_K)$.

A continuación se describe un primer ejemplo de un procedimiento de decodificación propuesto por la invención, llevado a la práctica en el seno del módulo de corrección de errores (o decodificador) y que permite la estimación de una palabra recibida (o secuencia q-aria) sobre la base de las LLR binarias suministradas por el módulo de demodulación.

La decodificación propuesta se realiza de manera iterativa y con paso de mensajes. Así, este procedimiento comprende:

- una etapa de inicialización sobre la base de las LLR binarias suministradas por el módulo de demodulación;

- etapas de iteración durante las cuales se intercambian mensajes entre los nodos de variable y los nodos de control conectados en el grafo bipartito, aplicándose la primera etapa de iteración en los valores emitidos de la etapa de inicialización, mientras que las etapas siguientes se aplican en los valores resultantes de la etapa anterior.

La etapa de inicialización comienza mediante un cálculo de las *informaciones a priori disponibles en los nodos de variable* $n \in \{1,\ldots N\}$ *en relación con los símbolos* $a \in GF(q)$, indicados $\gamma_n(a)$, de acuerdo con el procedimiento de decodificación presentado aquí:

$$\gamma_n(a) = \ln\left(\frac{\Pr(x_n = s_n | observación\ del\ canal)}{\Pr(x_n = a | observación\ del\ canal)}\right)$$

donde s_n indica el símbolo más probable para x_n, teniendo en cuenta la observación del canal.

Observando que:

$$\gamma_n(a) = \ln\left(\frac{\Pr(x_n = 0 | observación\ del\ canal)}{\Pr(x_n = a | observación\ del\ canal)}\right) - \ln\left(\frac{\Pr(x_n = 0 | observación\ del\ canal)}{\Pr(x_n = s_n | observación\ del\ canal)}\right)$$
$$= \ln\left(\frac{\Pr(x_n = 0 | observación\ del\ canal)}{\Pr(x_n = a | observación\ del\ canal)}\right) - \min_{s \in GF(q)} \ln\left(\frac{\Pr(x_n = 0 | observación\ del\ canal)}{\Pr(x_n = s | observación\ del\ canal)}\right)$$

se pueden calcular de la manera siguiente las informaciones *a priori* (en lo siguiente, a_i indica el bit i[ésimo] del símbolo a), y ello en cada nodo de variable n y para todo símbolo a del alfabeto:

$$\gamma_n(a) = \sum_{\substack{i=0,\ldots,p-1 \\ \text{bit } a_i=1}} \lambda_{n,i}$$
$$\gamma_n = \min_{a \in GF(q)} \gamma_n(a)$$
$$\gamma_n(a) = \gamma_n(a) - \gamma_n$$

Cada información *a priori* $\gamma_n(a)$ ofrece por lo tanto una medida de la fiabilidad de símbolo a (para la posición n), en forma de una distancia al símbolo más probable.

Por tanto, la etapa de inicialización puede proseguir (y acabar) con la inicialización de los *mensajes de los nodos de variable hacia los nodos de control* (denominados también *informaciones extrínsecas)*. El *mensaje enviado por el nodo de variable* $\alpha_{m,n}(a)$ *al nodo de control* $m \in H(n)$ *en relación con el símbolo* $a \in GF(q)$ *se indica como* $n \in \{1,\ldots N\}$.

Según el procedimiento de decodificación propuesto aquí, las informaciones extrínsecas se inicializan a partir de las informaciones *a priori* de la manera siguiente:

$$\alpha_{m,n}(a) = \gamma_n(a).$$

Cada etapa de iteración se realiza mediante paso de mensajes entre los nodos del grafo bipartito, tal como se representa esquemáticamente en la figura 3, y comprende por tanto las siguientes subetapas:

- cálculo de los mensajes de los nodos de control hacia los nodos de variable. El mensaje enviado por el nodo de control $m \in \{1,\ldots M\}$ al nodo de variable $n \in H(m)$ en relación con el símbolo $a \in GF(q)$ se indica como $\beta_{m,n}(a)$. Este mensaje se calcula en función de los mensajes $\alpha_{m,n}(a')$ recibidos por el nodo de control m desde los nodos de variable $n' \in H(m) - \{n\}$, en relación con todos los símbolos $a' \in GF(q)$, de acuerdo con el método de decodificación propuesto aquí:

$$\beta_{m,n}(a) = \min_{\substack{a_{n'} \in GF(q):\, n' \in H(m)-\{n\} \\ \left(\bigoplus_{n' \in H(m)-\{n\}} h_{m,n'} \otimes a_{n'}\right) \oplus (h_{m,n} \otimes a) = 0}} \max_{n' \in H(m)-\{n\}} \alpha_{m,n'}(a_{n'}).$$

- cálculo de las *informaciones a posteriori*. La *información a posteriori* calculada *en el nodo de variable* $n \in \{1,\ldots N\}$ en relación con el símbolo $a \in GF(q)$ se indica como $\tilde{\gamma}_n(a)$. Esta información se calcula en función de la información

a priori $\gamma_n(a)$ disponible en el nodo de variable n y de los mensajes $\beta_{m,n}(a)$ recibidos por el nodo de variable n desde los nodos de control $m \in H(n)$, en relación con el símbolo $a \in GF(q)$, de acuerdo con la fórmula a continuación:

$$\tilde{\gamma}_n(a) = \gamma_n(a) + \sum_{m \in H(n)} \beta_{m,n}(a)$$

- cálculo de los *mensajes de los nodos de variable hacia los nodos de control* (o *informaciones extrínsecas*). El mensaje (nuevo) $\alpha_{m,n}(a)$ se calcula en función de la información a priori $\gamma_n(a)$ disponible en el nodo de variable n y de los mensajes $\beta_{m',n}(a)$ recibidos por el nodo de variable n desde los nodos de control $m' \in H(n) - \{m\}$, en relación con el símbolo $a \in GF(q)$, en este caso de acuerdo con:

$$\begin{aligned} \alpha_{m,n}(a) &= \gamma_n(a) + \sum_{m' \in H(n)-\{m\}} \beta_{m',n}(a) = \tilde{\gamma}_n(a) - \beta_{m,n}(a) \\ \alpha_{m,n} &= \min_{a \in GF(q)} \alpha_{m,n}(a) \\ \alpha_{m,n}(a) &= \alpha_{m,n}(a) - \alpha_{m,n} \end{aligned}$$

Así, para el símbolo más probable s_n (es decir, tal que $\alpha_{m,n} = \alpha_{m,n}(s_n)$), se tiene siempre en cada iteración $\alpha_{m,n}(s_n) = 0$

En cada iteración, se puede calcular una *decisión dura* correspondiente a los símbolos más probables a partir de las informaciones *a posteriori* $\tilde{\gamma}_n(a), n \in \{1, \ldots N\}, a \in GF(q)$. Como ya se ha dado a conocer, los símbolos más probables son aquellos que producen el mínimo de informaciones *a posteriori* $\tilde{\gamma}_n(a)$. Si la secuencia de estos símbolos más probables se corresponde con una palabra de código, se considera que la palabra de código encontrada es la palabra de código emitida en el origen: la decodificación se detiene y se pasa al módulo de *demapping* la palabra de código encontrada.

En cambio, si la secuencia de estos símbolos más probables no se corresponde con una palabra de código, la decodificación continúa hasta que se obtenga un número máximo predeterminado de iteraciones, en cuyo caso se considera que la decodificación ha fracasado.

El interés de la fórmula min – max utilizada en el cálculo de los mensajes de los nodos de control $\beta_{m,n}(a)$ se puede explicar de la manera siguiente.

En lo sucesivo, se fijan un nodo de control m, un nodo de variable n conectado al nodo de control m y un símbolo $a \in GF(q)$. Los otros nodos de variable (es decir, diferentes de n) conectados al nodo de control m se indican como $n_1, \ldots, n_d$. Así, con las notaciones anteriores se tiene $H(m) = \{n, n_1, \ldots, n_d\}$.

El mensaje $\beta_{m,n}(a)$ se calcula en función de los mensajes recibidos por el nodo de control m desde los nodos de variable $n_1, \ldots, n_d$ (y ello para cada símbolo de $GF(q)$).

Si se considera una secuencia de símbolos $(a_1, \ldots, a_d)$ que comprende un símbolo para cada nodo de variable $n_1, \ldots, n_d$, se dirá en lo sucesivo que una secuencia de este tipo es admisible si la secuencia $(a, a_1, \ldots, a_d)$ - que comprende un símbolo para cada nodo de variable $n, n_1, \ldots, n_d$ - satisface el control m, es decir, la ecuación de la matriz de paridad en la fila m:

$$\left(\bigoplus_{k=1}^{d} h_{m,n_k} \otimes a_k \right) \oplus \left(h_{m,n} \otimes a \right) = 0$$

Por tanto, una secuencia admisible es susceptible de constituir una parte de una palabra de código.

El cálculo de $\beta_{m,n}(a)$ tiene en cuenta la totalidad de las secuencias admisibles. Así, para cada secuencia admisible $(a_1, \ldots, a_d)$, se puede calcular el máximo de los mensajes recibidos por el nodo de control m desde los nodos de variable $n_1, \ldots, n_d$ en relación con los símbolos de esta secuencia, que ofrece una medida de la fiabilidad de esta secuencia (correspondiéndose un valor bajo, tal como se ha visto anteriormente, con una información fiable). Con mayor precisión, se trata de:

$$\max_{k=1,\ldots,d}\left(\alpha_{m,n_k}(a_k)\right).$$

Si este máximo se indica como $M(a_1,\ldots,a_d)$, entonces el mensaje $\beta_{m,n}(a)$ es igual al mínimo de todos los máximos calculados para la totalidad de las secuencias admisibles, es decir:

$$\beta_{m,n}(a) = \min_{\substack{(a_1,\ldots,a_d)\\ \text{admisible}}} M(a_1,\ldots,a_d)$$
$$= \min_{\substack{(a_1,\ldots,a_d)\\ \text{admisible}}}\left(\max_{k=1,\ldots,d}\left(\alpha_{m,n_k}(a_k)\right)\right).$$

Por tanto, este mínimo apunta a la más probable de las secuencias admisibles y ofrece una medida de su fiabilidad.

Se obtiene así una información sobre la fiabilidad del símbolo a en la posición n, que tiene en cuenta otras informaciones disponibles en el nodo de control m.

A continuación se propone un primer método práctico para llevar a cabo el cálculo de los mensajes $\beta_{m,n}(a)$ sin tener que enumerar cada vez el conjunto de las secuencias admisibles. Este primer modo de realización utiliza un método de tipo «forward - backward» y se puede expresar de la manera siguiente.

Algoritmo Min – Max

Entrada: $\lambda_{1,0},\lambda_{1,1},\ldots,\lambda_{1,p-1},\lambda_{2,0},\lambda_{2,1},\ldots,\lambda_{2,p-1},\ldots,\lambda_{N,0},\lambda_{N,1},\ldots,\lambda_{N,p-1}$ - LLR binarias

Salida: $s_1,s_2,\ldots\ldots,s_N$ - secuencia q-aria

Inicialización

- *Informaciones a priori*

Para cada nodo de variable $n\in\{1,2,\ldots,N\}$ hacer:

$$\left.\begin{aligned}\gamma_n(a) &= \sum_{\substack{i=0,\ldots,p-1\\ \text{bit } a_i=1}}\lambda_{n,i};\\ \gamma_n &= \min_{a\in GF(q)}\gamma_n(a);\\ \gamma_n(a) &= \gamma_n(a)-\gamma_n;\end{aligned}\right\}\ \forall a\in GF(q)$$

- *Inicialización de los mensajes de los nodos de variable*

Para cada nodo de variable $n\in\{1,2,\ldots,N\}$ y cada nodo de control $m\in H(n)$ hacer:

$$\left.\alpha_{m,n}(a)=\gamma_n(a);\right\}\ \forall a\in GF(q)$$

Iteraciones

- *Mensajes de los nodos de control*

Para cada nodo de control $m\in\{1,2,\ldots,M\}$ (se plantea $H(m)=\{n_1,n_2,\ldots,n_d\}$, siendo consecuentemente d el cardinal de $H(m)$) hacer:

// forward

$$\left.\begin{aligned}F_1(a) &= \alpha_{m,n_1}\left(h_{m,n_1}^{-1}\otimes a\right);\\ F_j(a) &= \min_{\substack{a',a''\\ a'\oplus(h_{m,n_j}\otimes a'')=a}}\left(\max\left(F_{j-1}(a'),\alpha_{m,n_j}(a'')\right)\right);\end{aligned}\right\}\begin{aligned}&\forall a\in GF(q)\\ &\forall a\in GF(q),\forall j=2,\ldots,d-1\end{aligned}$$

// backward

$$\left.\begin{aligned} B_d(a) &= \alpha_{m,n_d}\left(h_{m,n_d}^{-1} \otimes a\right); \\ B_j(a) &= \min_{\substack{a',a'' \\ a'\oplus(h_{m,n_j}\otimes a'')=a}} \left(\max\left(B_{j+1}(a'), \alpha_{m,n_j}(a'')\right)\right); \end{aligned}\right| \begin{aligned} &\forall a \in GF(q) \\ &\forall a \in GF(q), \forall j = d-1,\ldots,2 \end{aligned}$$

// mensajes del nodo de control m

$$\left.\begin{aligned} \beta_{m,n_1}(a) &= B_2(a); \\ \beta_{m,n_j}(a) &= \min_{\substack{a',a'' \\ a'\oplus a''=h_{m,n_j}\otimes a}} \left(\max\left(F_{j-1}(a'), B_{j+1}(a'')\right)\right); \\ \beta_{m,n_d}(a) &= F_{d-1}(a); \end{aligned}\right| \begin{aligned} &\forall a \in GF(q) \\ &\forall a \in GF(q), \forall j = 2,\ldots,d-1 \\ &\forall a \in GF(q) \end{aligned}$$

- *Informaciones a posteriori*

Para cada nodo de variable $n \in \{1,2,\ldots,N\}$ hacer:

$$\left.\tilde{\gamma}_n(a) = \gamma_n(a) + \sum_{m\in H(n)} \beta_{m,n}(a);\right| \forall a \in GF(q)$$

- *Decisión dura: elección de los símbolos más probables*

Para cada nodo de variable $n \in \{1,2,\ldots,N\}$ hacer:

$$s_n = \arg\min_{a\in GF(q)} \tilde{\gamma}_n(a);$$

Si $\underset{n\in H(m)}{\oplus}\left(h_{m,n} \otimes s_n\right) = 0, \forall m \in \{1,2,\ldots M\}$ entonces SALIR;

- *Mensajes de los nodos de variable*

Para cada nodo de variable $n \in \{1,2,\ldots,N\}$ y cada nodo de control $m \in H(n)$ hacer:

$$\left.\begin{aligned} \alpha_{m,n}(a) &= \tilde{\gamma}_n(a) - \beta_{m,n}(a); \\ \alpha_{m,n} &= \min_{a\in GF(q)} \alpha_{m,n}(a); \\ \alpha_{m,n}(a) &= \alpha_{m,n}(a) - \alpha_{m,n}; \end{aligned}\right| \begin{aligned} &\forall a \in GF(q) \\ &\\ &\forall a \in GF(q) \end{aligned}$$

El método «forward - backward» utilizado en este modo de realización se ilustra en la figura 4. La notación $\alpha_{m,n}(\,)$ designa el conjunto de los mensajes $\alpha_{m,n}(a), a \in GF(q)$; de manera similar para $F_j(\,), B_j(\,)$ y $\beta_{m,n}(\,)$. La utilización de cantidades (o valores) intermedias $F_j(\,), B_j(\,)$ permite no implicar más que fórmulas de dos símbolos (a', a'') es decir, no tener que determinar cada vez el mínimo de los valores máximos, más que sobre dos dimensiones.

Por otra parte, las fórmulas utilizadas anteriormente en el presente documento para determinar las $F_j(a)$ y $B_j(a)$ en el cálculo de los mensajes de los nodos de control (fórmulas de tipo «min - max») se pueden llevar a la práctica de la manera siguiente.

Para simplificar la explicación, se ignora en este caso el valor de los coeficientes de la matriz H (es decir, se considera que $h_{m,n} = 1, \forall n \in H(m)$), lo cual permite expresar una fórmula de este tipo con la forma:

$$f(a) = \min_{\substack{a',a''\in GF(q) \\ a'\oplus a''=a}} \left(\max\left(f'(a'), f''(a'')\right)\right),$$

y proceder a su cálculo de la manera siguiente:

Para $a \in GF(q)$ inicializar: $f(a) = +\infty$;

Para $a' \in GF(q)$ hacer:

Para $a'' \in GF(q)$ hacer:

$$a = a' \oplus a'';$$

$$f = \max(f'(a'), f''(a''));$$

Si $f < f(a)$ entonces $f(a) = f$;

Se observa que la complejidad de este cálculo es proporcional a q^2.

Se puede señalar por otra parte que, dadas las fórmulas utilizadas, la dinámica de los mensajes intercambiados es baja y consecuentemente no es necesaria ninguna normalización. Además, la solución propuesta es independiente del conocimiento del ruido térmico.

A continuación se describe un segundo modo de realización de la invención en el cual la baja dinámica a la que se ha hecho referencia más arriba se utiliza para reducir la complejidad. Se pretende especialmente reducir el coste de los cálculos min-max de la forma

$$f(a) = \min_{\substack{a',a'' \in GF(q) \\ a' \oplus a'' = a}} (\max(f'(a'), f''(a'')))$$

Se puede constatar en la práctica (y además se demuestra en el anexo posteriormente) que, para este cálculo, es suficiente con considerar solamente los símbolos a' y a'' correspondientes a los $q+1$ valores más bajos de entre las $f'(a')$ y $f''(a'')$ (es decir $2 \cdot q$ valores en total).

El número de símbolos a', respectivamente a'', correspondientes a los $q+1$ valores más bajos de entre las $f'(a')$ y $f''(a'')$, se indica como q', q''. Se tiene entonces $q' + q'' = q+1$ y la complejidad del cálculo anterior se hace proporcional a $q' \cdot q'' \in \left\{ q, q+1, \ldots \frac{q}{2}\left(\frac{q}{2}+1\right) \right\}$.

Se podrá por tanto prever la reducción de la complejidad del cálculo de las $f(a), a \in GF(q)$, a condición de ordenar los valores $f'(a')$ y $f''(a'')$. Ahora bien, la ordenación de los valores $f'(a')$ y $f''(a'')$ añadiría una complejidad no despreciable en términos tanto del número de operaciones como de acceso a memoria.

Con el fin de evitar la ordenación de los valores $f'(a')$ y $f''(a'')$, en este caso los símbolos a' y a'' se almacenan según la parte entera de $f'(a')$ y de $f''(a'')$. Los conjuntos Δ'_k (respectivamente Δ''_k) que contienen los símbolos a' tales que $\lfloor f'(a') \rfloor = k$ (respectivamente los símbolos a'' tales que $\lfloor f''(a'') \rfloor = k$) donde el operador $\lfloor \ \rfloor$ designa la parte entera, se indican entonces.

$$\Delta'_k = \{a' \in GF(q) \mid \lfloor f'(a') \rfloor = k\}$$

$$\Delta''_k = \{a'' \in GF(q) \mid \lfloor f''(a'') \rfloor = k\}$$

Bastaría entonces con determinar el entero más pequeño E tal que el número de símbolos a' y a'' contenidos en los conjuntos $\Delta' = \Delta'_0 \cup \ldots \cup \Delta'_E$ y $\Delta'' = \Delta''_0 \cup \ldots \cup \Delta''_E$ sea superior o igual a $q+1$; y a continuación calcular $f(a)$ utilizando:

$$f(a) = \min_{\substack{a' \in \Delta', a'' \in \Delta'' \\ a' \oplus a'' = a}} (\max(f'(a'), f''(a'')))$$

Evidentemente, utilizando este método, puede que sea necesario utilizar más símbolos que cardinal del campo más uno (> $q+1$), aunque se evita así la ordenación de los valores $f'(a')$ y $f''(a'')$.

La utilización de los conjuntos Δ'_k y Δ''_k tiene un interés doble:

- se reduce el número de símbolos (y por tanto el número de bucles) necesarios para el cálculo de las fórmulas de tipo «min - max»);

- la mayor parte de los cálculos de máximo resultan obsoletos. Así, el máximo $\max(f'(a'), f''(a''))$ se debe calcular solamente si los símbolos a' y a'' se encuentran en conjuntos del mismo rango, es decir, $a' \in \Delta'_k$ y $a'' \in \Delta''_k$, con $k' = k''$. En los otros casos, el máximo se corresponde con el símbolo del conjunto de rango máximo (es decir,

$\max = f'(a')$ si $k' > k''$ y $\max = f''(a'')$ si $k'' > k'$).

Para utilizar este método, se procede de la manera siguiente:

- se define una dinámica predefinida de las informaciones *a priori*, de tal manera que la parte entera de los mensajes intercambiados proporcione un criterio suficientemente ajustado para distinguir entre los símbolos del alfabeto (en este caso el campo de Galois $GF(q)$). Para ello se utiliza una constante AI (de "*Average a priori Information*") y en la etapa de inicialización, después del cálculo de las informaciones *a priori* (véase el primer modo de realización), se calcula:

$$\gamma_{ave} = media\ de\ los \left\{ \gamma_n(a) \middle| n \in \{1,2,\ldots N\}, a \in GF(q) \right\};$$

a continuación se normalizan las informaciones *a priori*:

$$\gamma_n(a) = \frac{AI}{\gamma_{ave}} \cdot \gamma_n(a); \text{ para } n \in \{1,\ldots,N\}, a \in GF(q)$$

- se define una dinámica máxima de los mensajes de los nodos de control, con el fin de tener un número reducido de conjuntos Δ'_k y Δ''_k . Para ello, se utiliza una constante COT (de "*Cut Off Threshold*") y el cálculo de la fórmula «min - max» se realiza tal como se describe a continuación:

// determinar los conjuntos Δ'_k y Δ''_k

para $a \in GF(q)$ hacer

si $\left(k = \lfloor f'(a') \rfloor\right) < COT$ entonces añadir a a Δ'_k

si $\left(k = \lfloor f''(a'') \rfloor\right) < COT$ entonces añadir a a Δ''_k

// determinar el número de conjunto a utilizar (indicado E)

$card = 0;$

para $(E = 0\,;\, E < COT - 1\,;\, E{+}{+})$

si $\left(card{+}{=} \left(card(\Delta'_E) + card(\Delta''_E)\right)\right) \geq q+1$ entonces salir del bucle

Se define así E de tal manera que los conjuntos $\Delta' = \Delta'_0 \cup \ldots \cup \Delta'_E$ y $\Delta'' = \Delta''_0 \cup \ldots \cup \Delta''_E$ agrupan por lo menos $q+1$ elementos (como se ha visto anteriormente), excepto en el caso en el que se alcanza el valor máximo para E (es decir el valor $COT - 1$).

En la salida del bucle, E define por tanto el nombre de conjuntos que se utilizarán. Se observa que es posible que el bucle llegue a su fin, en cuyo caso $E = COT - 1$ (valor máximo de símbolo E logrado) y $card(\Delta' \cup \Delta'')$ puede ser entonces inferior a $q+1$ (con lo que el número de símbolos a' y a'' que se utilizarán es inferior a $q+1$ y, consecuentemente, la complejidad del cálculo «min-max» se reducirá todavía más). Esto se produce en general durante las últimas iteraciones, cuando la decodificación está bastante avanzada y cuando el decodificador no pone en duda más que un número muy bajo de símbolos.

Cuando sea necesario, para poner en evidencia el hecho de que el entero E se determina a partir de los conjuntos Δ'_k y Δ''_k , $0 \leq k \leq COT$, se indicará $E = E(\Delta', \Delta'')$.

Se propone así el siguiente algoritmo:

// cálculo de la fórmula «min - max» implementación selectiva

para $a \in GF(q)$ inicializar: $f(a) = COT$;

para $k' = 0,\ldots,E$ y $a' \in \Delta'_k$ hacer:

para $k'' = 0,\ldots,k'-1$ y $a'' \in \Delta''_k$ hacer:

$a = a' \oplus a''$;

// obsérvese que $f'(a') = \max(f'(a'), f''(a''))$ puesto que $k' > k''$

si $f'(a') < f(a)$ entonces $f(a) = f'(a')$

para $k'' = 0,\ldots,E$ y $a'' \in \Delta_k''$ hacer:

para $k' = 0,\ldots,k''-1$ y $a' \in \Delta_k'$ hacer:

$a = a' \oplus a''$;

// obsérvese que $f''(a'') = \max(f'(a'), f''(a''))$ puesto que $k'' > k'$

si $f''(a'') < f(a)$ entonces $f(a) = f''(a'')$

para $k = 0,\ldots,E$ hacer:

para $a' \in \Delta_k'$ y $a'' \in \Delta_k''$ hacer:

$a = a' \oplus a''$;

$f = \max(f'(a'), f''(a''))$

si $f < f(a)$ entonces $f(a) = f$

A continuación se ofrecerá la descripción completa del segundo modo de realización del algoritmo propuesto por la presente invención.

Se definen tres tipos de conjuntos:

- conjuntos $A_j\Delta_k$: contienen los símbolos $a \in GF(q)$ tales que $\lfloor \alpha_{m,n_j}(a) \rfloor = k$,

- conjuntos $F_j\Delta_k$: contienen los símbolos $a \in GF(q)$ tales que $\lfloor F_j(a) \rfloor = k$,

- conjuntos $B_j\Delta_k$: contienen los símbolos $a \in GF(q)$ tales que $\lfloor B_j(a) \rfloor = k$.

En esta descripción se mantienen las notaciones del primer modo de realización.

Por lo tanto el algoritmo propuesto en este segundo nodo de realización es el siguiente:

Algoritmo Min – Max

Entrada: $\lambda_{1,0}, \lambda_{1,1}, \ldots, \lambda_{1,p-1}, \lambda_{2,0}, \lambda_{2,1}, \ldots, \lambda_{2,p-1}, \ldots\ldots, \lambda_{N,0}, \lambda_{N,1}, \ldots, \lambda_{N,p-1}$ - LLR binarias

Salida: $s_1, s_2, \ldots\ldots, s_N$ - secuencia q-aria

definir AI , por ejemplo: $AI = 4.647$ para $GF(8)$

definir COT , por ejemplo: $COT = 10$ para $GF(8)$

Inicialización

- *Informaciones a priori*

Para cada nodo de variable $n \in \{1,2,\ldots,N\}$ hacer:

$$\left.\begin{array}{l} \gamma_n(a) = \sum_{\substack{i=0,\ldots,p-1 \\ \text{bit } a_i=1}} \lambda_{n,i}; \\ \gamma_n = \min_{a \in GF(q)} \gamma_n(a); \\ \gamma_n(a) = \gamma_n(a) - \gamma_n; \end{array}\right] \quad \forall a \in GF(q)$$

Hacer: $\gamma_{ave} = media\ de\ los \{\gamma_n(a) | n \in \{1,2,\ldots N\}, a \in GF(q)\}$;

Para cada de variable $n \in \{1,2,\ldots,N\}$ hacer:

$$\left.\begin{array}{l} \gamma_n(a) = \dfrac{AI}{\gamma_{ave}} \cdot \gamma_n(a); \end{array}\right] \quad \forall a \in GF(q)$$

- *Inicialización de los mensajes de los nodos de variable*

Para cada nodo de variable $n \in \{1,2,\ldots,N\}$ y cada nodo de control $m \in H(n)$ hacer:

$$\left.\begin{array}{l} \alpha_{m,n}(a) = \gamma_n(a); \end{array}\right] \quad \forall a \in GF(q)$$

Iteraciones

- *Mensajes de los nodos de control*

Para cada nodo de control $m \in \{1,2,\ldots,M\}$ (se plantea $H(m) = \{n_1, n_2, \ldots, n_d\}$) hacer:

// conjuntos $A_j\Delta_k$

$$\left.\begin{array}{l} \end{array}\right] \quad \forall a \in GF(q), \forall j = 1,\ldots,d$$

Determinar los conjuntos $A_j\Delta_k$;

// forward

$$\left.\begin{array}{l} F_1(a) = \alpha_{m,n_1}\left(h_{m,n_1}^{-1} \otimes a\right); \end{array}\right] \quad \forall a \in GF(q)$$

Determinar los conjuntos $F_1\Delta_k$;

$$\left.\begin{array}{l} E = E\left(F_{j-1}\Delta, A_j\Delta\right); \\ F_j(a) = \min\limits_{\substack{a' \in \bigcup_{k=0}^{E} F_{j-1}\Delta_k, a'' \in \bigcup_{k=0}^{E} A_j\Delta_k \\ a' \oplus (h_{m,n_j} \otimes a'') = a}} \left(\max\left(F_{j-1}(a'), \alpha_{m,n_j}(a'')\right)\right); \end{array}\right] \quad \forall a \in GF(q) \quad \left.\begin{array}{l} \\ \\ \end{array}\right] \forall j = 2,\ldots,d-1$$

Determinar los conjuntos $F_j\Delta_k$;

//backward

$$\left.\begin{array}{l} B_d(a) = \alpha_{m,n_d}\left(h_{m,n_d}^{-1} \otimes a\right); \end{array}\right] \quad \forall a \in GF(q)$$

Determinar los conjuntos $B_d\Delta_k$;

$$\left.\begin{array}{l} E = E\left(B_{j+1}\Delta, A_j\Delta\right); \\ B_j(a) = \min\limits_{\substack{a' \in \bigcup_{k=0}^{E} B_{j+1}\Delta_k, a'' \in \bigcup_{k=0}^{E} A_j\Delta_k \\ a' \oplus (h_{m,n_j} \otimes a'') = a}} \left(\max\left(B_{j+1}(a'), \alpha_{m,n_j}(a'')\right)\right); \end{array}\right] \quad \forall a \in GF(q) \quad \left.\begin{array}{l} \\ \\ \end{array}\right] \forall j = d-1,\ldots,2$$

Determinar los conjuntos $B_j\Delta_k$;

// mensajes del nodo de control m

$$\left.\begin{array}{l}\beta_{m,n_1}(a)=B_2(a);\\ E=E(F_{j-1}\Delta,B_{j+1}\Delta);\\ \beta_{m,n_j}(a)=\min_{a'\in\left(\bigcup_{k=0}^{c}F_{j-1}\Delta_k\right),a''\in\left(\bigcup_{k=0}^{c}B_{j+1}\Delta_k\right)\atop a'\oplus a''=h_{m,n_j}\otimes a}(\max(F_{j-1}(a'),B_{j+1}(a'')));\end{array}\right\}\begin{array}{l}\forall a\in GF(q)\\ \\ \forall a\in GF(q)\end{array}\quad\left.\begin{array}{l}\\ \\ \\ \end{array}\right\}\forall j=2,\ldots,d-1$$

$$\left.\begin{array}{l}\beta_{m,n_d}(a)=F_{d-1}(a);\end{array}\right\}\forall a\in GF(q)$$

- *Informaciones a posteriori*

  Para cada nodo de variable $n\in\{1,2,\ldots,N\}$ hacer:

$$\left.\begin{array}{l}\tilde{\gamma}_n(a)=\gamma_n(a)+\sum_{m\in H(n)}\beta_{m,n}(a);\end{array}\right\}\forall a\in GF(q)$$

- *Decisión dura: elección de los símbolos más probables*

  Para cada nodo de variable $n\in\{1,2,\ldots,N\}$ hacer:

$$s_n=\arg\min_{a\in GF(q)}\tilde{\gamma}_n(a);$$

  Si $\oplus_{n\in H(m)}(h_{m,n}\otimes s_n)=0,\quad\forall m\in\{1,2,\ldots,M\}$ entonces SALIR;

- *Mensajes de los nodos de variable*

  Para cada nodo de variable $n\in\{1,2,\ldots,N\}$ y cada nodo de control $m\in H(n)$ hacer:

$$\left.\begin{array}{l}\alpha_{m,n}(a)=\tilde{\gamma}_n(a)-\beta_{m,n}(a);\\ \alpha_{m,n}=\min_{a\in GF(q)}\alpha_{m,n}(a);\\ \alpha_{m,n}(a)=\alpha_{m,n}(a)-\alpha_{m,n};\end{array}\right\}\begin{array}{l}\forall a\in GF(q)\\ \\ \forall a\in GF(q)\end{array}$$

En los cálculos anteriores, para determinar el resultado de cada una de las fórmulas de tipo «min-max» se utiliza el algoritmo presentado más arriba en el presente documento *("// el cálculo de la fórmula «min-max» implementación selectiva"*).

Por otra parte, tal como se ha precisado anteriormente, en la etapa de inicialización, la dinámica de las informaciones *a priori* se modifica de tal manera que la parte entera de los mensajes intercambiados proporciona un criterio suficientemente ajustado para distinguir entre los símbolos del alfabeto (en este caso el campo de Galois $GF(q)$). En la práctica, el cálculo de γ_{ave} se puede realizar solamente de forma periódica, sabiendo que su variación entre dos palabras de código consecutivas es despreciable.

Los parámetros mencionados más arriba en el presente documento se escogen preferentemente tales que:

- después de la multiplicación por $\frac{AI}{\gamma_{ave}}$, los valores de las informaciones *a priori* $\gamma_n(a)$ estén suficientemente separados, de tal modo que los conjuntos de tipo Δ_k contienen pocos símbolos;

- el valor de COT, que se puede determinar por medio de simulaciones, debe ser relativamente bajo puesto que determina el número de conjuntos de tipo Δ_k que se utilizará.

A este respecto se puede observar que los valores de las constantes AI y COT ofrecidas como ejemplo más arriba en el presente documento sobre GF(8) se determinaron por medio de simulaciones y se pueden utilizar para un canal AWGN y para todo código LDPC sobre GF(8), con independencia de las características del código (longitud, irregularidad, rendimiento) o de la modulación utilizada.

La utilización conjunta de los conjuntos $A_j\Delta_k$, $F_j\Delta_k$, $B_j\Delta_k$ y del método «forward - backward» se describe esquemáticamente a título meramente explicativo en la figura 5, para una constante COT igual a 3 y un nodo de control m de grado 4 ($H(m=\{n_1,n_2,n_3,n_4\})$.

En este ejemplo ofrecido sobre la base del campo de Galois, es decir, GF(8), se considera que los conjuntos $F_1\Delta_k, k=0,1,2$, contienen respectivamente 3, 1 y 4 símbolos y que los conjuntos $A_1\Delta_k, k=0,1,2$, contienen respectivamente 4, 3 y 1 símbolos. Por tanto, para el cálculo de los valores $F_2(a), a\in GF(8)$, es suficiente con utilizar los valores $F_1(a')$ y $\alpha_{m,n_2}(a'')$ correspondientes a los símbolos $a'\in F_1\Delta_k$ y $a''\in A_2\Delta_k$ solamente para $k=0$ y $k=1$ (así pues 4 símbolos a' y 7 símbolos a'', es decir, un total de 11 símbolos, lo cual es suficiente puesto que debe haber por lo menos $q+1=9$ símbolos en total).

Una vez que se han calculado los valores $F_2(a), a\in GF(8)$, los símbolos $a\in GF(8)$ se almacenan en los conjuntos $F_2\Delta_k, k=0,1,2$ (según las partes enteras de los valores calculados). Supóngase que los conjuntos $F_2\Delta_k, k=0,1,2$ contienen respectivamente 6, 2 y 0 símbolos. Por lo tanto, para el cálculo de los valores $F_3(a), a\in GF(8)$ basta con utilizar solamente los valores $F_2(a')$ y $\alpha_{m,n_3}(a'')$ correspondientes a los 6 símbolos $a'\in F_2\Delta_0$ y a los 3 símbolos $a''\in A_3\Delta_0$ (es decir, un total de 9 símbolos).

A continuación se presenta el resultado de simulaciones comparativas de los diferentes algoritmos previstos anteriormente en el presente documento y en la técnica anterior, en este caso para códigos LDPC sobre GF(8) o binarios.

El algoritmo objeto de la presente invención se denomina MMA (de "*Min-Max Algorithm*"), "*implementación estándar*" su primer modo de realización más arriba e "*implementación selectiva*" su segundo modo de realización.

Todos los códigos utilizados son códigos LDPC irregulares. El grafo bipartito se construye utilizando el algoritmo "*Progressive Edge Growth*", descrito en "*Regular and irregular progressive edge – growth Tanner graphs*", H. Y. Hu, E. Eleftheriou, D.M. Arnold, IEEE Trans. Inform. Theory, vol. 51, n.º 1, págs. 386 a 398, 2005. En el caso de un código LDPC sobre GF(8), las aristas del grafo bipartito se corresponden con los lugares de los coeficientes no nulos en la matriz de paridad. Estos coeficientes se han escogido de manera aleatoria. En lo sucesivo no se indicará la irregularidad de los códigos utilizados, aunque se indicará el grado medio de los nodos de variable, que se designará como d_n^{ave}.

Las *irregularidades* de los códigos LDPC binarios se han optimizado por "*evolución de densidad*" tal como se describe en "*Design of capacity approaching irregular low density parity check codes*", T. J. Richardson, M.A. Shokrollahi, R.L. Urbanke, IEEE Trans. Inform. Theory, vol. 47, n.º 2, págs. 619 a 637, 2001. Sin embargo, la irregularidad correspondiente a un buen código binario no es adecuada para obtener un buen código q-ario (y a la inversa). En general, cuanto más aumenta el cardinal del campo de Galois, más "huecos" (menos aristas en el grafo bipartito) deben ser los códigos utilizados, y de allí la diferencia entre los grados medios de los nodos de variable para los códigos binarios y para los códigos sobre GF(8) utilizados más abajo.

Las simulaciones se han realizado para un canal AWGN, utilizando una modulación QPSK. Por otra parte, las prestaciones de prestaciones se han realizado para una tasa de errores de bit (BER) de 10^{-5}. El número máximo de iteraciones se fija a 200.

La figura 6 representa prestaciones de los decodificadores MMA y MSA con un código LDPC sobre GF(8), y del decodificador MSA con un código LDPC binario para códigos que tienen las características siguientes:

- 1.008 bits (es decir, 336 símbolos) de información, 2.016 bits (es decir, 672 símbolos) codificados (es decir, un rendimiento de 1/2) por un lado;

- 4.032 bits (es decir, 1.344 símbolos) de información, 8.064 bits (es decir, 2.688 símbolos) codificados (es decir, un rendimiento de 1/2) por otro lado;

- para los códigos binarios: $d_n^{ave}=3,27$;

- para los códigos sobre GF(8): $d_n^{ave}=2,5$.

Se observa que:

- para los códigos LDPC sobre GF(8), el decodificador MMA aventaja al decodificador MSA de 0,15 a 0,22 dB;

- el código sobre GF(8) (decodificación MMA) avanza al código binario en 0,33 dB.

Se puede señalar también que, tras las simulaciones realizadas (no representadas), se ha constatado que para los códigos LDPC sobre GF(8), el decodificador MMA está a solamente 0,04 dB del decodificador SPA.

Por otra parte, se comparan, en referencia a la figura 7, las prestaciones del decodificador MMA, obtenidas utilizando las implementaciones “estándar” y “selectiva”. Se utilizan códigos LDPC sobre GF(8) de longitud binaria igual a 2.016 bits y de rendimientos 1/3, 1/2 y 4/5, cuyos grados medios de los nodos de variables son respectivamente 2,4, 2,5 y 2,8. Por otra parte, las constantes AI y COT utilizadas para la implementación selectiva se han fijado a:

$$AI = 4{,}647\ ,\ COT = 10$$

En la vista de la figura 7 se observa que las prestaciones obtenidas son (casi) idénticas, dicho de otra manera que no hay ninguna pérdida sustancial de prestaciones de la implementación selectiva con respecto a la implementación estándar.

La figura 8 permite la comparación de la complejidad de las decodificaciones MSA y MMA (implementaciones estándar y selectiva) sobre GF(8), y de la decodificación MSA sobre GF(2) (código binario).

El número de operaciones representa el número medio de operaciones en millares por bit codificado (todas las iteraciones acumuladas), efectuadas por los dos decodificadores. Este número depende de la relación señal/ruido (SNR) puesto que el número medio de iteraciones efectuadas por los dos decodificadores disminuye cuando la relación señal/ruido E_b/N_0 aumenta.

Para cada rendimiento (1/3, 1/2 y 4/5), se registra el número medio de operaciones por bit codificado para los valores de E_b/N_0, correspondientes a la región de caída de la BER.

Sobre GF(8), el decodificador MMA – estándar es menos complejo que el decodificador MSA. Esto es debido al hecho de que el decodificador MMA converge más rápidamente que el MSA; dicho de otra manera, el número de iteraciones efectuadas por el MMA es inferior al número de iteraciones efectuadas por el MSA (efectuando los dos decodificadores casi el mismo número de operaciones por iteración).

Sobre GF(8), el decodificador MMA – selectivo es de 2 a 3 veces menos complejo que el decodificador MMA – estándar. El número de iteraciones efectuadas por los dos decodificadores es el mismo aunque, para cada iteración, el MMA – selectivo efectúa menos operaciones que el MMA – estándar.

Finalmente, se observa que el decodificador MMA – selectivo sobre GF(8) es solamente de 1 a 2 veces más complejo que el decodificador MSA binario.

Los modos de realización presentados anteriormente no son más que posibles ejemplos de puesta en práctica de la invención, la cual no se limita a los mismos.

ANEXO

Se consideran dos funciones f' y f'' definidas sobre $GF(q)$ y de valores reales, y f la función definida por la fórmula «min-max»:

$$f(a)=\min_{\substack{a',a''\in GF(q)\\ a'\oplus a''=a}}(\max(f'(a'),f''(a')))$$

Sean Δ' y Δ'' subconjuntos de $GF(q)$ tales que el conjunto de los valores

$$\{f'(a')\,|\,a'\in\Delta'\}\cup\{f''(a'')\,|\,a''\in\Delta''\}$$

contiene los $q+1$ valores más bajos de entre todos los valores de f' y f'', es decir, de entre

$$\{f'(a')\,|\,a'\in GF(q)\}\cup\{f''(a'')\,|\,a''\in GF(q)\}.$$

Se mostrará entonces que $f(a)$ se puede calcular mediante la fórmula

$$f(a)=\min_{\substack{a'\in\Delta',a''\in\Delta''\\ a'\oplus a''=a}}(\max(f'(a'),f''(a')))$$

lo cual implica un número reducido de símbolos a' y a''.

En una primera etapa, se mostrará que todo símbolo $a\in GF(q)$ se puede escribir como la suma de dos símbolos:

$a=\delta'+\delta''$ con $\delta'\in\Delta'$ y $\delta''\in\Delta''$.

Para ello se definen dos funciones $I,T_a:GF(q)\to GF(q)$ con $I(x)=x$ (función identidad) y $T_a(x)=a\oplus x$ (traslación en a). Señalando que las dos funciones son biyectivas, se obtiene

$$\mathrm{card}(I(\Delta'))+\mathrm{card}(T_a(\Delta''))=\mathrm{card}(\Delta')+\mathrm{card}(\Delta'')\geq q+1>\mathrm{card}(GF(q))$$

De aquí resulta que existen $\delta'\in\Delta'$ y $\delta''\in\Delta''$ tales que:

$I(\delta')=T_a(\delta'')$, lo cual equivale a $\delta'=a\oplus\delta''$, de donde $a=\delta'\oplus\delta''$.

En una segunda etapa de la demostración, los dos símbolos tales que $s'\oplus s''=a$ se indican $s',s''\in GF(q)$, logrando el mínimo en la definición de $f(a)$, dicho de otra manera, tales que

$$\max(f'(s'),f''(s''))=\min_{\substack{a',a''\in GF(q)\\ a'\oplus a''=a}}(\max(f'(a'),f''(a')))=f(a)$$

Sean $\delta'\in\Delta'$ y $\delta''\in\Delta''$ tales que $a=\delta'+\delta''$ (después de la primera etapa). Recuérdese que los símbolos de Δ' y Δ'' se corresponden con los $q+1$ valores más bajos de entre el conjunto de los valores de las funciones f' y f''.

Entonces (por reducción al absurdo) si se tenía $s'\notin\Delta'$ o $s''\notin\Delta''$, uno por lo menos de los valores $f'(s')$, $f''(s'')$ estará fuera de los $q+1$ valores más bajos y se obtendría por tanto

$$\max(f'(\delta'),f''(\delta''))<\max(f'(s'),f''(s'')),$$

lo cual contradice el carácter mínimo de la pareja s',s''.

Por tanto, se tiene necesariamente $s'\in\Delta'$ y $s''\in\Delta''$ y consecuentemente

$$f(a)=\max(f'(s'),f''(s''))=\min_{\substack{a'\Delta',a''\in\Delta''\\ a'\oplus a''=a}}(\max(f'(a'),f''(a')))$$

REIVINDICACIONES

1. Procedimiento de decodificación iterativa de una palabra recibida, representada por valores de una señal, según un código de matriz de control de paridad, del tipo de paso de mensajes entre dos nodos de variable y dos nodos de control de un grafo bipartito asociado a dicha matriz, caracterizado una etapa de inicialización de por lo menos un mensaje de un nodo de variable, en función de dichos valores, mediante una información representativa de la relación entre la probabilidad de tener el símbolo más probable en la posición correspondiente al nodo de variable y la probabilidad de tener el símbolo actual en dicha posición.

2. Procedimiento de decodificación según la reivindicación 1, que comprende una etapa de determinación de por lo menos un mensaje, referente a un símbolo determinado, de un nodo de control hacia un nodo de variable determinado, como el valor mínimo adoptado, entre las secuencias de símbolos que verifican la ecuación en el nodo de control utilizando dicho símbolo determinado en el nodo de variable determinado, por el valor máximo de los mensajes recibidos en el nodo de control desde nodos de variable que no sean el nodo de variable determinado y referentes, cada uno de ellos, al símbolo asociado a este otro nodo de variable en la secuencia que verifica la ecuación.

3. Procedimiento de decodificación según la reivindicación 1 ó 2, que comprende una etapa de determinación de los mensajes de un nodo de variable hacia un nodo de control referentes al conjunto de los símbolos de tal manera que el valor mínimo de dichos mensajes sea nulo.

4. Procedimiento de decodificación según una de las reivindicaciones 1 a 3, en el cual la etapa de inicialización del mensaje de un nodo de variable comprende las siguientes etapas:

- para cada símbolo del alfabeto, determinación de la suma de las relaciones de verosimilitud logarítmica binarias referentes a los bits no nulos del símbolo y en la posición correspondiente al nodo de variable y;

- determinación del mínimo de las sumas determinadas;

- resta, de cada suma determinada, del mínimo determinado.

5. Procedimiento de decodificación según una de las reivindicaciones 1 a 4, que comprende una etapa de determinación de una información *a posteriori* referente a un nodo de variable y a un símbolo, como la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable y referentes al símbolo.

6. Procedimiento de decodificación según la reivindicación 5, que comprende las siguientes etapas:

- para cada nodo de variable, determinación del símbolo para el cual la información *a posteriori* es mínima;

- si la secuencia de los símbolos así determinados para el conjunto de los nodos de variable es una palabra de código, utilización de dicha palabra de código como palabra estimada.

7. Procedimiento de decodificación según una de las reivindicaciones 3 a 6, considerándose las reivindicaciones 4 y 5 en dependencia de la reivindicación 3, en el cual la etapa de determinación de los mensajes de un nodo de variable hacia un nodo de control dado comprende las siguientes etapas:

- para cada símbolo, determinación de la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable procedentes de un nodo de control que no sea el nodo de control dado y referentes al símbolo;

- determinación del mínimo de entre las sumas determinadas;

- para cada símbolo, resta, del mínimo determinado, de la suma determinada referente al símbolo.

8. Procedimiento de decodificación según una de las reivindicaciones 2 a 7, considerándose las reivindicaciones 3 a 7 en dependencia de la reivindicación 2, en el cual la etapa de determinación de por lo menos un mensaje de un nodo de control hacia un nodo de variable determinado comprende las siguientes etapas:

- determinación, para cada símbolo, de un valor intermedio igual al valor mínimo adoptado por el valor máximo de mensajes recibidos en el nodo de control desde una parte solamente de los nodos de variable asociados al nodo de control;

- determinación del valor mínimo adoptado por el valor máximo de entre los valores intermedios y los mensajes recibidos de un nodo de variable asociado al nodo de control y que no pertenecen a dicha parte.

9. Procedimiento de decodificación según una de las reivindicaciones 1 a 8, que comprende una etapa de determinación del mínimo, entre las parejas de dos símbolos que pertenecen al alfabeto del código, del máximo de dos valores asociados respectivamente a los dos símbolos de cada pareja, que comprende las siguientes

subetapas:

- determinación de conjuntos que agrupan, cada uno de ellos, los símbolos a los cuales están asociados valores comprendidos en un intervalo determinado;

- selección de conjuntos, de entre los conjuntos determinados, de tal manera que la reunión de los conjuntos seleccionados contiene por lo menos un número predeterminado de símbolos;

- utilización, como máximo entre dos valores, del valor asociado al símbolo comprendido en el conjunto correspondiente al intervalo superior cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a intervalos distintos;

- determinación por comparación del máximo entre dos valores cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a un mismo intervalo.

10. Procedimiento de decodificación según una de las reivindicaciones 1 a 9, en el cual el código es un código de alfabeto no binario.

11. Dispositivo de decodificación iterativa de una palabra recibida, representada por valores de una señal, según un código de matriz de control de paridad, del tipo de paso de mensajes entre dos nodos de variable y dos nodos de control de un grafo bipartito asociado a dicha matriz, caracterizado porque comprende medios de inicialización de por lo menos un mensaje de un nodo de variable, en función de dichos valores, mediante una información representativa de la relación entre la probabilidad de tener el símbolo más probable en la posición correspondiente al nodo de variable y la probabilidad de tener el símbolo actual en dicha posición.

12. Dispositivo de decodificación según la reivindicación 11, que comprende medios de determinación de por lo menos un mensaje, referente a un símbolo determinado, de un nodo de control hacia un nodo de variable determinado, como el valor mínimo adoptado, entre las secuencias de símbolos que verifican la ecuación en el nodo de control utilizando dicho símbolo determinado en el nodo de variable determinado, por el valor máximo de los mensajes recibidos en el nodo de control desde nodos de variable que no sean el nodo de variable determinado y referentes, cada uno de ellos, al símbolo asociado a este otro nodo de variable en la secuencia que verifica la ecuación.

13. Dispositivo de decodificación según la reivindicación 11 ó 12, que comprende medios de determinación de los mensajes de un nodo de variable hacia un nodo de control referentes al conjunto de los símbolos de tal manera que el valor mínimo de dichos mensajes sea nulo.

14. Dispositivo de decodificación según una de las reivindicaciones 11 a 13, en el cual los medios de inicialización del mensaje de un nodo de variable comprenden:

- medios para determinar, para cada símbolo del alfabeto, la suma de las relaciones de verosimilitud logarítmica binarias referentes a los bits no nulos del símbolo y en la posición correspondiente al nodo de variable y;

- medios para determinar el mínimo de las sumas determinadas;

- medios para restar, de cada suma determinada, el mínimo determinado.

15. Dispositivo de decodificación según una de las reivindicaciones 11 a 14, que comprende medios de determinación de una información *a posteriori* referente a un nodo de variable y a un símbolo, como la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable y referentes al símbolo.

16. Dispositivo de decodificación según la reivindicación 15, que comprende:

- medios para determinar, para cada nodo de variable, el símbolo para el cual la información *a posteriori* es mínima;

- medios para utilizar la secuencia de los símbolos así determinados para el conjunto de los nodos de variable como palabra estimada si dicha secuencia es una palabra de código.

17. Dispositivo de decodificación según una de las reivindicaciones 13 a 16, considerándose las reivindicaciones 14 y 15 en dependencia de la reivindicación 13, en el cual los medios de determinación de los mensajes de un nodo de variable hacia un nodo de control dado comprenden:

- medios para determinar, para cada símbolo, la suma del mensaje inicial referente al símbolo en dicho nodo de variable y del conjunto de los mensajes recibidos en el nodo de variable procedentes de un nodo de control que no sea el nodo de control dado y referentes al símbolo;

- medios para determinar el mínimo de entre las sumas determinadas;

- para cada símbolo, medios para restar el mínimo determinado, de la suma determinada referente al símbolo.

18. Dispositivo de decodificación según una de las reivindicaciones 12 a 17, considerándose las reivindicaciones 13 a 15 en dependencia de la reivindicación 12, en el cual los medios de determinación de por lo menos un mensaje de un nodo de control hacia un nodo de variable determinado comprenden:

- medios para determinar, para cada símbolo, un valor intermedio igual al valor mínimo adoptado por el valor máximo de mensajes recibidos en el nodo de control desde una parte solamente de los nodos de variable asociados al nodo de control;

- medios para determinar el valor mínimo adoptado por el valor máximo de entre los valores intermedios y los mensajes recibidos de un nodo de variable asociado al nodo de control y que no pertenecen a dicha parte.

19. Dispositivo de decodificación según una de las reivindicaciones 11 a 18, en el cual medios para determinar el mínimo, entre las parejas de dos símbolos que pertenecen al alfabeto del código, del máximo de dos valores asociados respectivamente a los dos símbolos de cada pareja, comprenden:

- medios de determinación de conjuntos que agrupan, cada uno de ellos, los símbolos a los cuales están asociados valores comprendidos en un intervalo determinado;

- medios de selección de conjuntos, de entre los conjuntos determinados, de tal manera que la reunión de los conjuntos seleccionados contiene por lo menos un número predeterminado de símbolos;

- medios que utilizan, como máximo entre dos valores, el valor asociado al símbolo comprendido en el conjunto correspondiente al intervalo superior cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a intervalos distintos;

- medios para determinar por comparación el máximo entre dos valores cuando los valores están asociados a símbolos comprendidos en conjuntos seleccionados correspondientes a un mismo intervalo.

20. Dispositivo de decodificación según una de las reivindicaciones 11 a 19, en el cual el código es un código de alfabeto no binario.

21. Aparato de comunicación que comprende un dispositivo según una de las reivindicaciones 11 a 20.

$$H = \begin{pmatrix} 1 & 3 & 4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 7 & 1 & 0 \\ 5 & 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 3 & 0 & 2 & 6 \end{pmatrix}$$

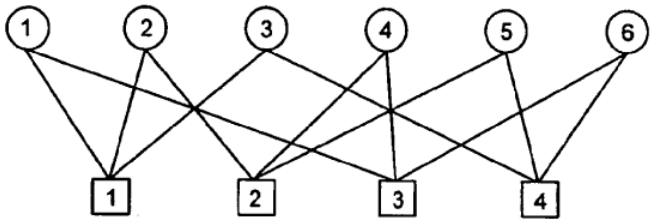

Fig. 1

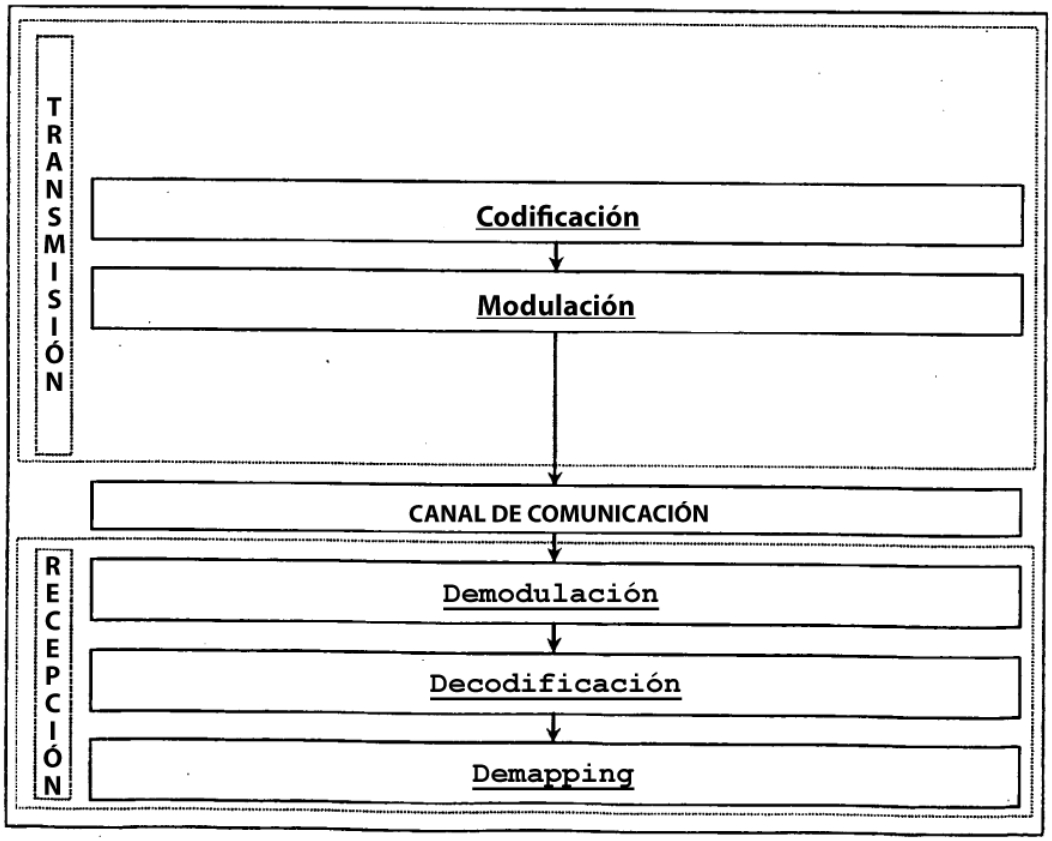

Fig. 2

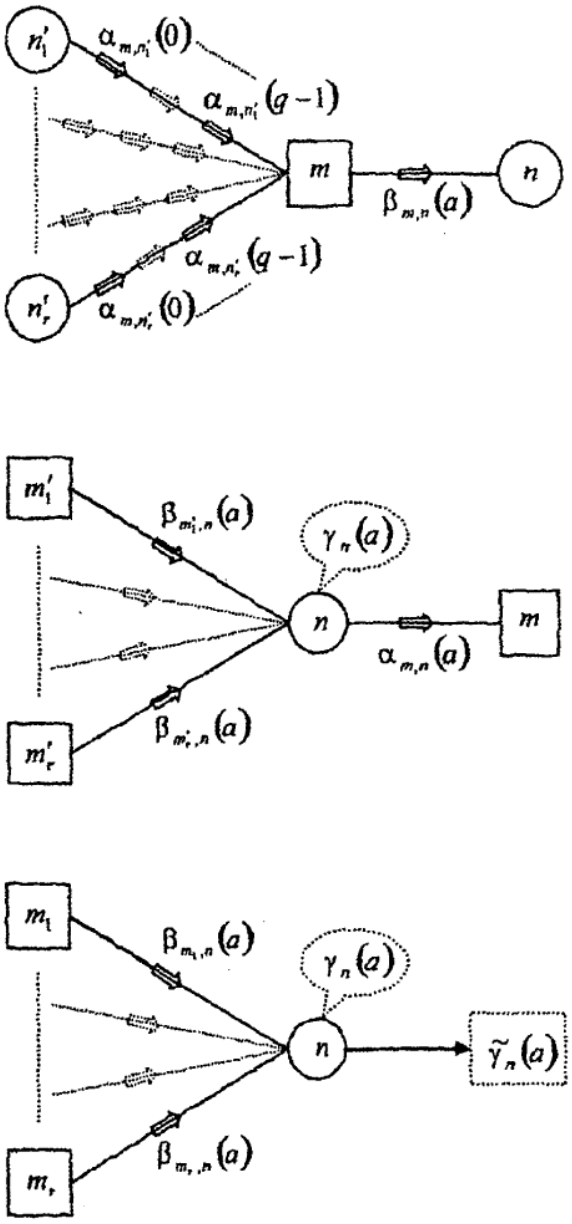
n'_1
$\alpha_{m,n'_1}(0)$
$\alpha_{m,n'_1}(q-1)$
m
n
$\beta_{m,n}(a)$
$\alpha_{m,n'_r}(q-1)$
n'_r
$\alpha_{m,n'_r}(0)$
m'_1
$\beta_{m'_1,n}(a)$
$\gamma_n(a)$
n
m
$\alpha_{m,n}(a)$
$\beta_{m'_r,n}(a)$
m'_r
m_1
$\beta_{m_1,n}(a)$
$\gamma_n(a)$
n
$\tilde{\gamma}_n(a)$
$\beta_{m_r,n}(a)$
m_r

Fig. 3

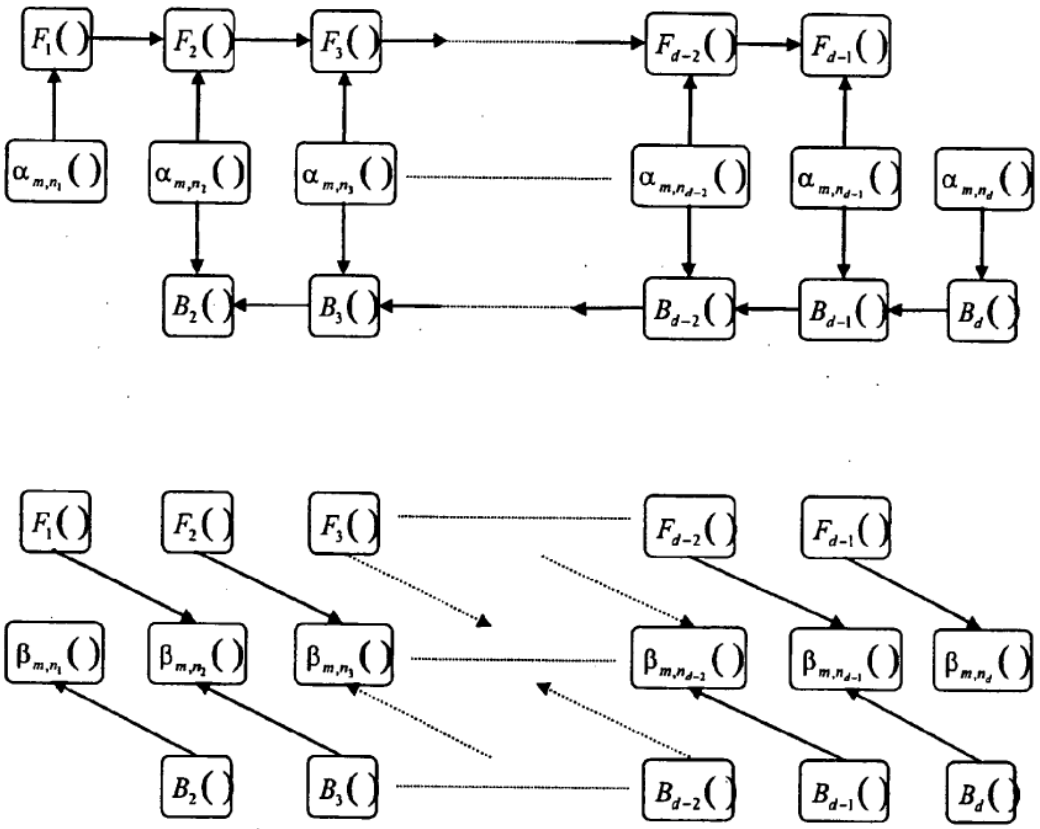
$F_1()$
$F_2()$
$F_3()$
$F_{d-2}()$
$F_{d-1}()$
$\alpha_{m,n_1}()$
$\alpha_{m,n_2}()$
$\alpha_{m,n_3}()$
$\alpha_{m,n_{d-2}}()$
$\alpha_{m,n_{d-1}}()$
$\alpha_{m,n_d}()$
$B_2()$
$B_3()$
$B_{d-2}()$
$B_{d-1}()$
$B_d()$
$F_1()$
$F_2()$
$F_3()$
$F_{d-2}()$
$F_{d-1}()$
$\beta_{m,n_1}()$
$\beta_{m,n_2}()$
$\beta_{m,n_3}()$
$\beta_{m,n_{d-2}}()$
$\beta_{m,n_{d-1}}()$
$\beta_{m,n_d}()$
$B_2()$
$B_3()$
$B_{d-2}()$
$B_{d-1}()$
$B_d()$

Fig. 4

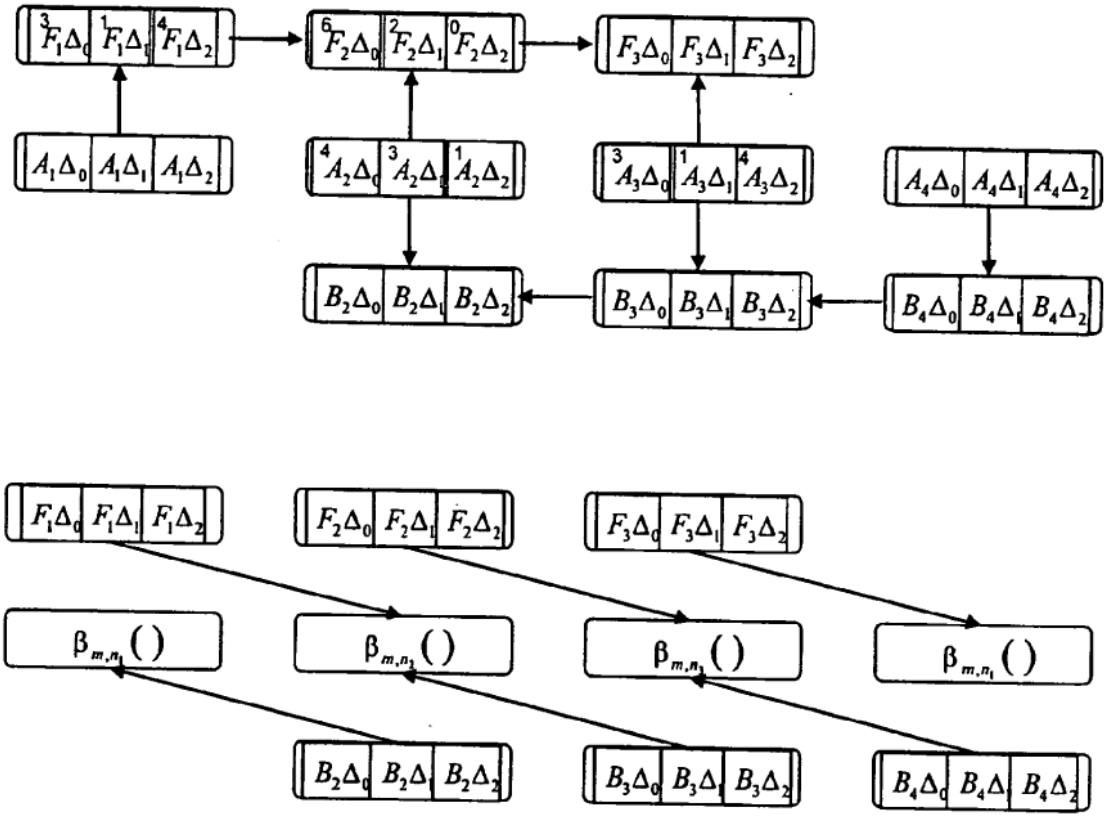
$^{3}F_1\Delta_0$ $^{1}F_1\Delta_1$ $^{4}F_1\Delta_2$
$^{6}F_2\Delta_0$ $^{2}F_2\Delta_1$ $^{0}F_2\Delta_2$
$F_3\Delta_0$ $F_3\Delta_1$ $F_3\Delta_2$
$A_1\Delta_0$ $A_1\Delta_1$ $A_1\Delta_2$
$^{4}A_2\Delta_0$ $^{3}A_2\Delta_1$ $^{1}A_2\Delta_2$
$^{3}A_3\Delta_0$ $^{1}A_3\Delta_1$ $^{4}A_3\Delta_2$
$A_4\Delta_0$ $A_4\Delta_1$ $A_4\Delta_2$
$B_2\Delta_0$ $B_2\Delta_1$ $B_2\Delta_2$
$B_3\Delta_0$ $B_3\Delta_1$ $B_3\Delta_2$
$B_4\Delta_0$ $B_4\Delta_1$ $B_4\Delta_2$
$F_1\Delta_0$ $F_1\Delta_1$ $F_1\Delta_2$
$F_2\Delta_0$ $F_2\Delta_1$ $F_2\Delta_2$
$F_3\Delta_0$ $F_3\Delta_1$ $F_3\Delta_2$
$\beta_{m,n_1}()$
$\beta_{m,n_2}()$
$\beta_{m,n_3}()$
$\beta_{m,n_4}()$
$B_2\Delta_0$ $B_2\Delta_1$ $B_2\Delta_2$
$B_3\Delta_0$ $B_3\Delta_1$ $B_3\Delta_2$
$B_4\Delta_0$ $B_4\Delta_1$ $B_4\Delta_2$

Fig. 5

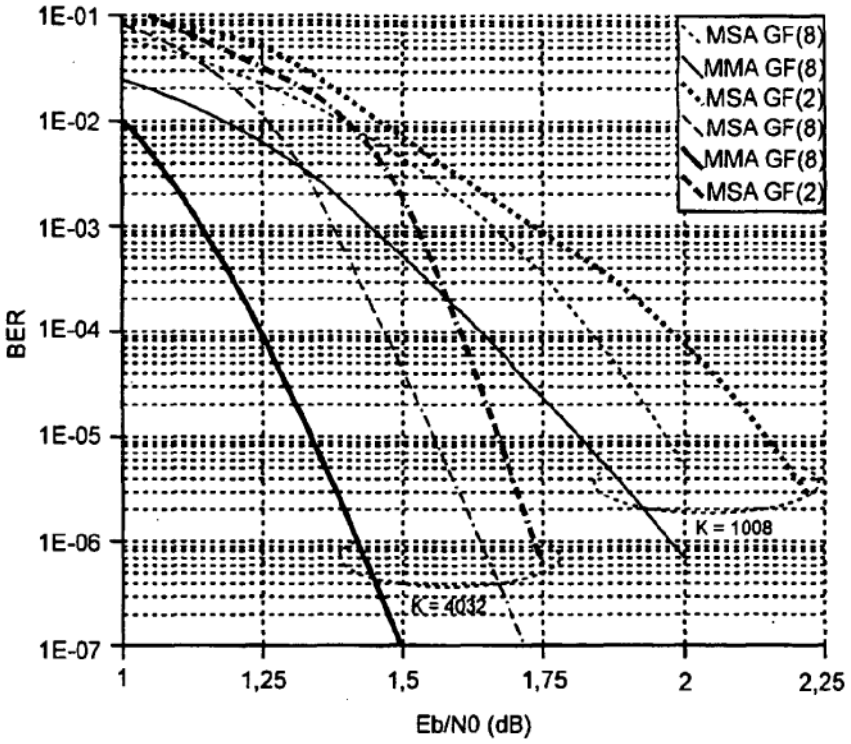
1E-01
1E-02
1E-03
1E-04
1E-05
1E-06
1E-07
BER
1
1,25
1,5
1,75
2
2,25
Eb/N0 (dB)
MSA GF(8)
MMA GF(8)
MSA GF(2)
MSA GF(8)
MMA GF(8)
MSA GF(2)
K = 1008
K = 4032

Fig. 6

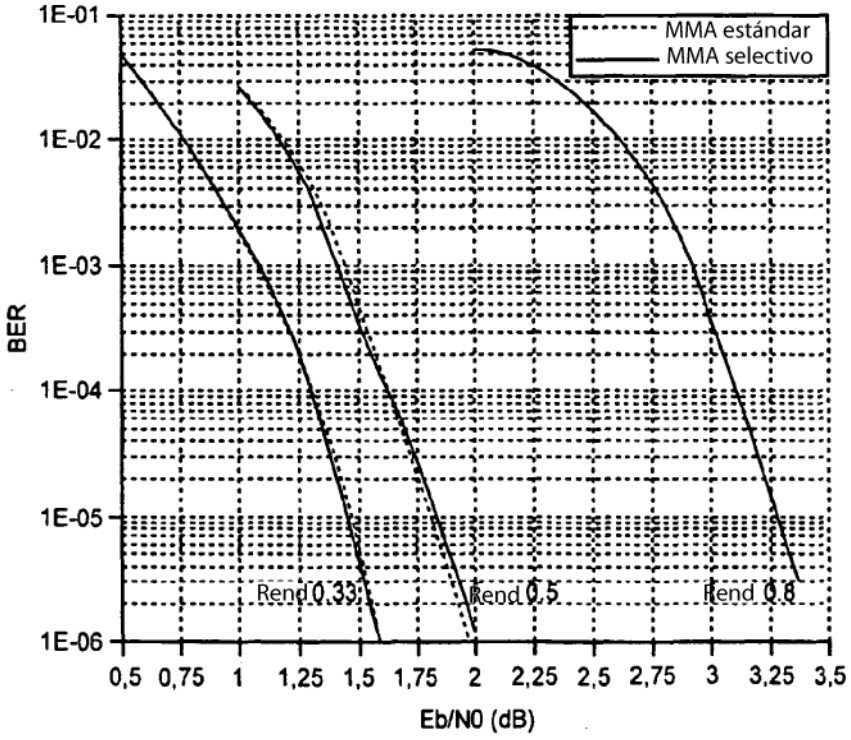
1E-01
1E-02
1E-03
1E-04
1E-05
1E-06
BER
0,5
0,75
1
1,25
1,5
1,75
2
2,25
2,5
2,75
3
3,25
3,5
Eb/N0 (dB)
MMA estándar
MMA selectivo
Rend 0,33
Rend 0,5
Rend 0,8

Fig. 7

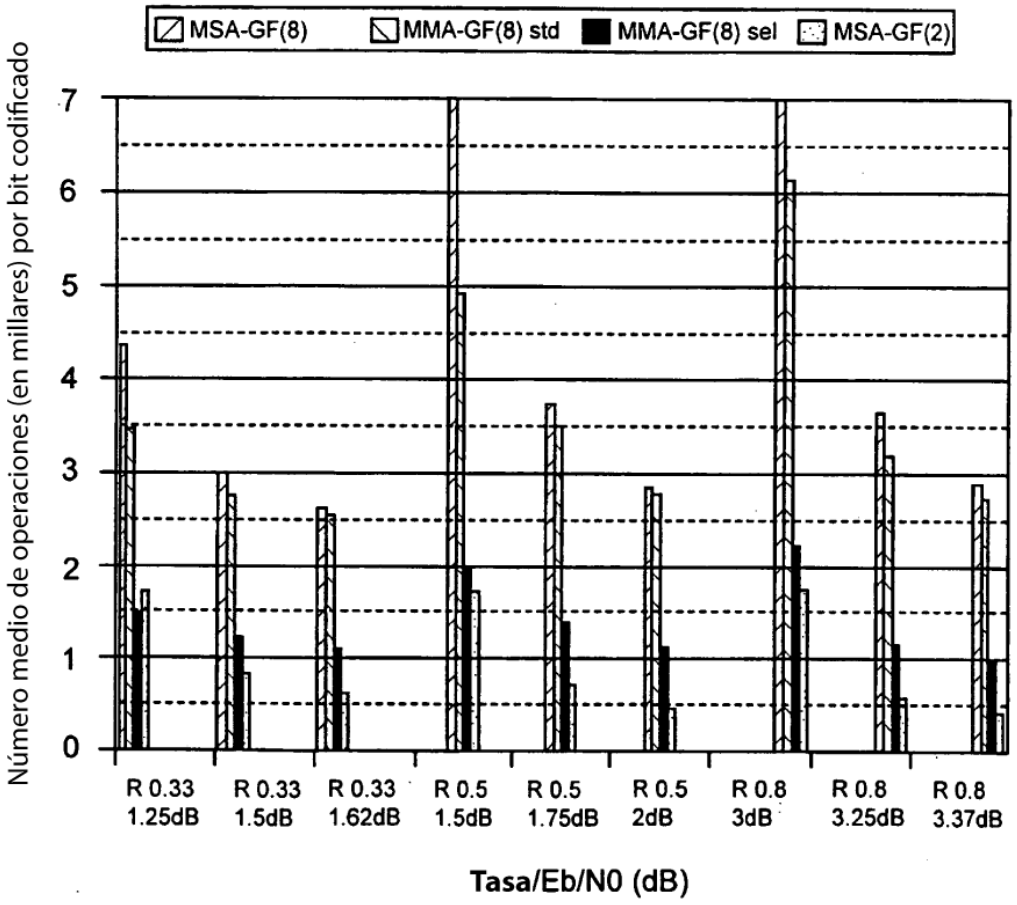
MSA-GF(8)
MMA-GF(8) std
MMA-GF(8) sel
MSA-GF(2)
Número medio de operaciones (en millares) por bit codificado
0
1
2
3
4
5
6
7
R 0.33 1.25dB
R 0.33 1.5dB
R 0.33 1.62dB
R 0.5 1.5dB
R 0.5 1.75dB
R 0.5 2dB
R 0.8 3dB
R 0.8 3.25dB
R 0.8 3.37dB
Tasa/Eb/N0 (dB)

Fig. 8