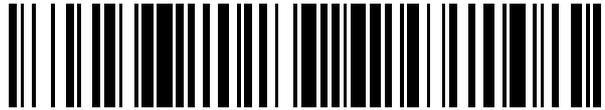


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 561 913**

21 Número de solicitud: 201400891

51 Int. Cl.:

**H03M 13/29** (2006.01)

12

PATENTE DE INVENCION CON EXAMEN PREVIO

B2

22 Fecha de presentación:

**06.11.2014**

43 Fecha de publicación de la solicitud:

**01.03.2016**

Fecha de la concesión:

**03.08.2016**

45 Fecha de publicación de la concesión:

**10.08.2016**

73 Titular/es:

**UNIVERSIDAD DE MÁLAGA (100.0%)  
Avda Cervantes, 2  
29071 Málaga (Málaga) ES**

72 Inventor/es:

**MARTÍN VEGA, Francisco Javier;  
BLÁNQUEZ CASADO, Francisco;  
LÓPEZ MARTÍNEZ, Francisco Javier;  
GÓMEZ PAREDES, Gerardo y  
ENTRAMBASAGUAS MUÑOZ, José Tomás**

54 Título: **Sistemas y métodos para turbo decodificación iterativa de baja tasa de error y baja complejidad**

57 Resumen:

Sistemas y métodos para turbo decodificación iterativa de baja tasa de error y baja complejidad. La invención se refiere a un método de turbo decodificación iterativa de baja tasa de error y baja complejidad caracterizado porque comprende (a) el almacenamiento de las métricas LLR en un banco de memoria de entrada que posteriormente entrega, bien el bloque correspondiente a un decodificador no paralelo que trabaja hacia delante (fw) y hacia atrás (bw), bien los sub-bloques correspondientes a dos o más decodificadores paralelos fw/bw; (b) el cálculo de métricas a-posteriori por el o los decodificadores; (c) el cálculo por una o más unidades de cálculo extrínsecas de la información extrínseca; y (d) la comparación de las medidas LLR a-posteriori fw y bw a lo largo de distintas iteraciones. La invención refiere diferentes implementaciones del método así como sistemas que implementan dicho método y sus variantes.

ES 2 561 913 B2

**Sistemas y Métodos para turbo decodificación iterativa de baja tasa de error y baja complejidad**

**SECTOR TÉCNICO**

La invención se refiere a comunicaciones digitales, en particular a sistemas y métodos asociados a la decodificación iterativa basada en el algoritmo SOVA (*Soft Output Viterbi Algorithm*) que pueden ser aplicados a estándares de comunicaciones móviles como 3GPP (*3rd Generation Partnership Project*), LTE (*Long Term Evolution*) y UMTS (*Universal Mobile Telecommunications System*). Más particularmente, la invención se refiere a sistemas y métodos para turbo decodificación iterativa paralela para los que no se requiere el solapamiento entre sub-bloques adyacentes, reduciendo así la latencia de decodificación de forma drástica.

**ESTADO DE LA TÉCNICA**

Los turbo códigos han sido ampliamente usados en estándares de comunicaciones móviles modernos debido a su enorme capacidad correctiva que permite transmisiones de datos fiables cercanas a la capacidad ergódica del sistema. El turbo codificador consiste en la concatenación en paralelo de dos codificadores RSC (*Recursive Systematic Convolutional*) separados por un entrelazador que aporta aleatoriedad al código. La decodificación consiste en un proceso iterativo en el que se genera (en cada iteración) la información extrínseca, que es usada como información a-priori en la siguiente iteración. Esto permite usar un criterio MAP (*Maximum A-Posteriori*) que minimiza la tasa de error o BER (*Bit Error Rate*). El turbo decodificador clásico usa un decodificador SISO (*Soft In Soft Out*) que genera información extrínseca en la forma de valores LLR (*Log Likelihood Ratio*), y dos entrelazadores, uno para los datos recibidos del canal y otro para la información extrínseca. La Fig. 1 ilustra el diagrama funcional de un turbo decodificador clásico. En dicha figura, el elemento SISO 1 decodifica el mensaje mientras que el elemento SISO 2 decodifica el mensaje entrelazado. Se dice que iteraciones completas decodifican el mensaje mientras que las semi-iteraciones decodifican el mensaje entrelazado. El diagrama funcional de la Fig. 1 puede ser implementado de diferentes formas que se identifican en este documento como: (1) turbo decodificación en serie, (2) turbo decodificación concurrente y (3) turbo decodificación barajada.

En la Fig. 2 se muestra un diagrama funcional de turbo decodificación en serie, donde la línea horizontal representa el tiempo. Nótese que, debido al hecho de que el decodificador SISO 2 tiene que esperar hasta que SISO 1 haya decodificado el bloque completo, esta arquitectura se puede implementar usando sólo un decodificador SISO. La Fig. 3 es un esquema funcional de la turbo decodificación concurrente. Nótese que en la literatura el término turbo decodificación paralela es comúnmente usado para referirse al método que hemos denominado aquí turbo decodificación concurrente; sin embargo en este documento hemos preferido reservar el término turbo decodificación paralela para otro tipo de arquitectura que va a ser explicada posteriormente. En la turbo decodificación concurrente las iteraciones completas y semi-iteraciones se realizan al mismo tiempo por un decodificador SISO distinto: uno decodifica el mensaje y el otro el mensaje entrelazado. Por tanto se incrementa la velocidad de convergencia de la decodificación. El intercambio de información extrínseca está representado con líneas negras en dicha figura. En la Fig. 4 se representa otra arquitectura conocida como turbo decodificación barajada. Esta arquitectura requiere dos decodificadores SISO como en la arquitectura concurrente. Sin embargo en este caso se permite que la información a-priori sea intercambiada entre ambos decodificadores en la misma iteración siempre que dicha información esté disponible. Dicha arquitectura acelera la velocidad de convergencia aún más

que la turbo decodificación concurrente, sin embargo requiere un mecanismo de acceso a memoria bastante complejo. En Juntan Zhang y otros "Shuffled iterative decoding" trabajo publicado en la revista IEEE Transactionson Communications, vol. 53, 2005 y en las referencias presentes en dicho trabajo se puede obtener más información acerca de esta arquitectura para la decodificación con turbo códigos. Debido al hecho de que la turbo decodificación en serie es la más extendida normalmente nos referiremos a esta arquitectura en el presente documento.

Existen esencialmente dos familias de algoritmo SISO para la turbo decodificación: MAP y SOVA (Soft Output Viterbi Algorithm). Una descripción detallada de estos algoritmos se puede encontrar en "Comparative study of turbo decoding techniques: an overview", publicada en la revista IEEE Transactionon Vehicular Technology, vol. 49, 2000 por Woodard y otros. El algoritmo MAP ofrece una ganancia de decodificación de aproximadamente 0.6 dB sobre el SOVA a expensas de tener mayor latencia y mayor consumo de área del chip incluso en sus versiones simplificadas Max-Log-MAP y Log-MAP. Se han realizado algunas propuestas con el fin de reducir esta diferencia en ganancia de decodificación debido a las atractivas características del algoritmo SOVA para implementación eficiente en el chip. En concreto, Fossorier y otros presentaron en el artículo titulado "On the equivalence between SOVA and max-log-MAP decodings", publicado en la revista IEEE Communication Letters, vol. 2, 1998 una modificación del algoritmo SOVA que lo hace equivalente al algoritmo Max-Log-MAP. Desafortunadamente el algoritmo propuesto es significativamente más complejo que el algoritmo SOVA original. En el artículo "Adaptive SOVA for 3GPP-LTE Receivers" de Blanquez-Casado y otros, publicado en la revista IEEE Communications Letters, vol.18, junio 2014 se presenta una modificación del algoritmo SOVA que mejora significativamente la capacidad correctiva, especialmente cuando se emplean altas tasas de modulación y codificación. Sin embargo el algoritmo propuesto requiere emplear una ventana de actualización de pesos de tamaño variable que hace difícil su implementación en un chip.

Una alternativa la constituye el algoritmo conocido como BISOVA (Bi-directional SOVA) en el que dos decodificadores SOVA decodifican la secuencia recibida en cada iteración haciendo que la capacidad correctiva sea similar a la del algoritmo MAP; no obstante la complejidad se duplica también. Esta propuesta fue presentada por primera vez por Chen y otros con el artículo titulado "Bi-directional SOVA decoding for turbo-codes" publicado en la revista IEEE Communication Letters, vol. 4, 2000. Una arquitectura para implementar dicho algoritmo que es eficiente desde el punto de vista del uso de la memoria fue presentada por Efimov y otros en la patente titulada "HIGH-THROUGHPUT MEMORY-EFFICIENT BI-SOVA DECODER ARCHITECTURE", US 2008/0152045 A1. No obstante esta arquitectura al estar basada en el algoritmo BISOVA sigue presentando mayor complejidad que las basadas en el algoritmo SOVA.

BISOVA requiere dos decodificadores SOVA, o dos bancos de decodificadores SOVA en el caso paralelo trabajando a la vez en cada iteración. La idea clave detrás del algoritmo BISOVA es que realizar el algoritmo SOVA en diferentes direcciones permite obtener una información distinta que ayuda a mejorar la turbo decodificación. En el algoritmo SOVA existe un proceso de actualización que persigue mejorar la calidad de los pesos, o medidas de fiabilidad, asociadas a cada bit decodificado para la etapa k con el fin de obtener las medidas a-posteriori LLR finales. Este proceso de actualización compara fiabilidades o pesos asociadas al camino ML (Maximum Likelihood) en la etapa k con las fiabilidades de los caminos competidores al camino ML que se unen a él y son descartados en etapas posteriores a k. Sin embargo, hay caminos que no se tienen en cuenta en el proceso de actualización porque son descartados antes

de mezclarse con el camino  $\overline{ML}$ . Este hecho da lugar a una sobrestimación de las medidas a-posteriori en SOVA que reduce su capacidad correctora. No obstante, algunos caminos que son descartados antes de unirse al camino  $ML$  en una dirección pueden no ser descartados antes de unirse a dicho camino  $ML$  en la otra dirección. Por tanto, realizar el algoritmo SOVA en ambas direcciones puede mejorar la turbo decodificación. En el algoritmo BISOVA, en la iteración  $i$ , uno de los decodificadores calcula las medidas LLR a-posteriori hacia adelante, identificadas con el símbolo  $L_f^{(i)}(u_k)$ , mientras que el otro las calcula hacia atrás, identificadas con el símbolo  $L_b^{(i)}(u_k)$ . Entonces, las medidas a-posteriori finales en BISOVA pueden ser calculadas como

$$L^{(i)}(u_k) = u_k \min(|L_f^{(i)}(u_k)|, |L_b^{(i)}(u_k)|) \quad (1)$$

Esto mejora notablemente la capacidad correctora a costa de doblar el consumo de área del chip.

La naturaleza iterativa de la turbo decodificación tiene asociada una latencia considerable que hace difícil cumplir los requisitos de régimen binario asociados con estándares modernos como LTE (Long Term Evolution). Debido a esto la decodificación paralela se hace imprescindible. En la decodificación paralela cada bloque a decodificar es dividido en varios sub-bloques que son decodificados al mismo tiempo por un decodificador SISO diferente. A pesar de que esta estrategia permite reducir la latencia eficientemente, la paralelización tiene dos grandes inconvenientes: (1) el consumo de área del chip es mayor y (2) hay una pérdida no despreciable en capacidad correctiva debido a la incertidumbre que existe en los bordes entre sub-bloques. Con el propósito de mitigar el segundo problema se suele usar solape entre sub-bloques para disminuir la incertidumbre entre sub-bloques a costa de aumentar la latencia de la decodificación. Como el decodificador SOVA es menos complejo que el MAP, sus implementaciones consumen menos área del chip y por tanto el primero está especialmente indicado para turbo decodificadores paralelos donde el consumo de área es un tema primordial. Sin embargo el algoritmo MAP ha sido preferido tradicionalmente debido a su mayor capacidad correctora. Estos hechos justifican la necesidad de idear modificaciones del algoritmo SOVA que permitan obtener capacidades correctoras cercanas al algoritmo MAP manteniendo la baja complejidad del algoritmo SOVA.

### DESCRIPCIÓN DE LA INVENCIÓN

La invención se refiere a una modificación del algoritmo SOVA que, a diferencia de las propuestas mencionadas anteriormente, mejora su capacidad correctora cuando se aplica a la turbo decodificación sin ningún incremento en el consumo de área ni de latencia. Esta modificación denominada ALSOVA (ALternated direction SOVA) mejora la capacidad correctiva del algoritmo SOVA sin ningún incremento en área ni latencia. ALSOVA mejora la capacidad correctiva del algoritmo SOVA cuando éste se aplica a la decodificación iterativa sin ningún incremento en consumo de área del chip ni de latencia. Por tanto este método puede aplicarse a la turbo decodificación siguiendo diferentes arquitecturas como la arquitectura en serie, la concurrente y la barajada usando realizaciones paralelas y no paralelas del banco de decodificadores SOVA. El algoritmo ALSOVA también puede aplicarse a la ecualización iterativa, de la misma forma que los algoritmos MAP y SOVA son usados en este campo.

El algoritmo ALSOVA se basa en el mismo principio que el algoritmo BISOVA, esto es, lleva a cabo el proceso de actualización en diferentes direcciones de forma que se reduce la sobrestimación de las medidas a-posteriori al tener en cuenta más caminos. Sin embargo, al contrario que el algoritmo BISOVA, el algoritmo ALSOVA no realiza la decodificación en

ambas direcciones durante la misma iteración. En cambio, el algoritmo ALSOVA compara las medidas LLR a-posteriori calculadas hacia delante y hacia atrás a lo largo de distintas iteraciones mediante el intercambio de la información extrínseca entre iteraciones. Esto permite utilizar un único decodificador SOVA en lugar de dos. El decodificador SOVA tiene que ser capaz de realizar la decodificación tanto hacia adelante como hacia atrás; sin embargo, esta cuestión tiene un impacto mínimo en el consumo de área ya que la mayoría de las unidades que forman el decodificador no resultan afectadas. Por tanto, la invención propone calcular las medidas LLR hacia adelante en las iteraciones pares, y las medidas LLR hacia atrás en las iteraciones impares como se indica a continuación:

$$L^{(i)}(u_k) = \begin{cases} L_f^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) == 1 \\ L_b^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) == 0 \end{cases} \quad (2)$$

Siendo  $\lfloor i \rfloor$  la función suelo. En la expresión previa se considera que en una iteración completa el mensaje es decodificado, mientras que en cada semi-iteración el mensaje entrelazado es decodificado. Las iteraciones comienzan en  $i=1$ , mientras que las semi-iteraciones comienzan en  $i=1.5$ .

La sobrestimación de bits erróneamente decodificados tiene un considerable efecto en la turbo decodificación porque puede propagar errores a lo largo de iteraciones. Nuestro enfoque propone realizar la decodificación hacia adelante y hacia atrás en ejecuciones pares e impares del decodificador, respectivamente. Por tanto, si en la etapa  $k$  un bit decodificado erróneamente tiene una LLR sobrestimada, esta sobrestimación puede propagarse hasta la siguiente semi-iteración. Sin embargo, debido a que la otra iteración va en la dirección contraria, otros caminos pueden ser considerados, mitigando la sobrestimación y evitando la propagación del error.

El algoritmo ALSOVA propuesto puede implementarse en turbo decodificación en serie, paralela o barajada usando tanto una implementación no paralela de cada decodificador SISO constituyente como una realización paralela de cada decodificador SISO constituyente. En el último caso tenemos de hecho un banco de decodificadores SISO. El algoritmo ALSOVA puede usarse también en turbo ecualización: el algoritmo ALSOVA requiere la implementación de un decodificador SOVA hacia adelante/atrás, o fw/bw del inglés forward/backward, el cual pueda realizar la decodificación en ambas direcciones en diferentes iteraciones. En este documento se proporciona una realización de dicho decodificador. Además se proporciona una posible realización de un turbo decodificador usando el algoritmo ALSOVA para las arquitecturas en serie paralela y concurrente paralela.

#### *PMSBx*

Adicionalmente, la invención se refiere también a una nueva arquitectura para la turbo decodificación en paralelo, identificada con el acrónimo PMSBx que viene del inglés PathMetric/StateBorder Exchange. Dicha arquitectura mitiga la degradación de prestaciones en términos de BER (Bit Error Rate) o tasa de error de bit asociada a la decodificación en paralelo. PMSBx emplea el hecho de que las métricas de camino son fiables en el borde derecho de los sub-bloques y por tanto pueden ser usadas como métricas de inicialización para el siguiente sub-bloque de la siguiente iteración.

PMSBx consigue mejores prestaciones que los métodos clásicos basados en solapamiento de sub-bloques. Al contrario que otros enfoques, PMSBx no requiere solapamiento entre sub-bloques, por tanto reduce la latencia de la decodificación. El solapamiento tiene la desventaja de que incrementa la latencia de la decodificación, reduciendo la tasa binaria alcanzable. Por otra parte, la BER de la turbo decodificación en paralelo sin solapamiento sufre una degradación significativa debido a la incertidumbre en los bordes de los sub-bloques.

PMSBx aprovecha la diferente fiabilidad de las PMs y los estados del camino ML para mejorar las prestaciones de la decodificación y se basa principalmente en dos premisas:

- Convergencia de las métricas de camino: debido a que el borde izquierdo de un sub-bloque se corresponde con el borde derecho del sub-bloque previo, las PMs del borde derecho de un sub-bloque pueden ser usadas como PMs iniciales para los siguientes sub-bloques en la siguiente iteración. Esto aprovecha el hecho de que dichas PMs son fiables. Esta idea fue explotada por Yoon, "A parallel MAP algorithm for low latency turbo decoding", IEEE Communications Letters, 2002, donde se presenta un método para el algoritmo MAP paralelo que consigue alcanzar la misma capacidad correctiva que en el caso no paralelo pero sin ningún incremento en latencia. Este método es identificado en este documento como SBI. En el caso del algoritmo SOVA, las PMs son fiables en el borde derecho de un sub-bloque. Por tanto, éstas métricas pueden ser usadas como métricas de inicialización para el siguiente sub-bloque en la siguiente iteración para el caso del algoritmo SOVA.
- Convergencia de los estados para el camino ML: Haciendo la inicialización con valores de métrica fiables, los estados del camino ML serán fiables en el borde izquierdo de los sub-bloques. Estos estados producidos por un decodificador SOVA pueden ser usados para hacer el último recorrido hacia atrás o TB desde un estado fiable del decodificador SOVA previo, reduciendo considerablemente la sobrestimación de los valores LLR a-posteriori, la cual degrada las prestaciones en términos de BER. Debido a que los estados en el sub-borde izquierdo están disponibles antes de que se lleve a cabo el TB en el borde izquierdo, estos estados se pueden intercambiar en la misma iteración. Hay que tener en cuenta que esto no es aplicable para las métricas de camino ya que éstas son intercambiadas entre iteraciones completas.

La invención también se refiere a (i) programas de ordenador adaptados para, o que comprenden código software adaptado para, llevar a cabo los métodos de la invención; (ii) medios de almacenamiento legibles en computador que comprenden dichos programas informáticos o que comprenden instrucciones para hacer que un aparato de procesamiento de datos lleve a cabo los métodos de la invención; (iii) medios portadores de grabación con dichos programas informáticos grabados en ellos; (iv) ondas portadoras de señal portando señales que incorporan dichos programas informáticos.

### **BREVE EXPLICACIÓN DE LAS FIGURAS**

La Fig. 1 ilustra el diagrama funcional de la turbo decodificación.

La Fig. 2 muestra un diagrama de tiempos de la turbo decodificación en serie.

La Fig. 3 muestra un diagrama de tiempos de la turbo decodificación concurrente.

La Fig. 4 muestra un diagrama de tiempos de la turbo decodificación barajada.

La Fig. 5 ilustra un diagrama funcional de decodificación paralela usando solape entre sub-bloques.

La Fig. 6 muestra un diagrama de tiempos de la arquitectura propuesta de turbo decodificador en serie usando ALSOVA.

5 La Fig. 7 ilustra la estructura de una posible realización de la invención usando la arquitectura en serie paralela.

La Fig. 8 ilustra una posible realización de los decodificadores SOVA fw/bw.

La Fig. 9 ilustra una posible realización de la Unidad de Recursión fw/bw del decodificador SOVA fw/bw.

10 La Fig. 10 proporciona un diagrama temporal asociado a la arquitectura concurrente usando el algoritmo propuesto ALSOVA.

La Fig. 11 ilustra una estructura de una posible realización de la invención usando la arquitectura concurrente paralela.

La Fig. 12 ilustra la estructura de la invención PMSBx que se puede aplicar a los algoritmos  
15 SOVA, ALSOVA y BISOVA.

### **MODOS DE REALIZACIÓN DE LA INVENCION**

En esta solicitud se referencian varias publicaciones. Los contenidos y revelaciones incluidas en éstas, así como en la totalidad de las referencias citadas por dichas publicaciones, se incorporan  
20 en la presente solicitud con el fin de describir de una manera completa y detallada el estado del arte al que pertenece esta invención. La terminología que se emplea en adelante es empleada con el propósito de describir de manera precisa determinados conceptos, y no debe considerarse como limitante.

La invención mejora la capacidad correctora de errores de un turbo decodificador basado en el  
25 algoritmo SOVA clásico sin ningún incremento en la complejidad. Esta invención puede usarse con turbo decodificación paralela y no paralela. En la Fig. 5 se ilustra una descripción funcional de la decodificación paralela usando solape entre sub-bloques. Se asumen códigos con terminación trellis como es el caso del código de LTE, es decir, el estado inicial y final es conocido, siendo comúnmente el estado cero. La longitud del bloque a decodificar es  $K+T$   
30 donde  $K$  es la longitud del mensaje y  $T$  el número de etapas requeridas para volver al estado cero. En el caso de tener  $P$  decodificadores SISO trabajando en paralelo cada sub-bloque tiene una longitud de  $M+L$  etapas excepto el último sub-bloque que tiene una longitud de  $M+L/2+T$  etapas y el primer sub-bloque de  $M+L/2$ , siendo  $M=K/P$  y  $L$  el número de etapas de solape. La secuencia decodificada por cada decodificador tiene una longitud de  $M$  bits.

35 *ALSOVA para turbo decodificación en serie paralela*

En la Fig. 6 se muestra un diagrama de tiempos del método ALSOVA propuesto, para decodificación serie de turbo códigos. En concreto se muestran dos iteraciones completas del algoritmo, aunque el proceso de decodificación puede extenderse a cualquier número de iteraciones según el criterio de parada determinado.

En la Fig. 7 se presenta una posible realización de la invención siguiendo la arquitectura de un turbo decodificador en serie paralelo. El diagrama de bloques de dicha figura muestra las medidas LLR del canal asociadas con un turbo código de tasa 1/3 con un bit sistemático y dos bits de paridad como es el caso del empleado en la tecnología LTE. Estas métricas LLR se representan en la figura como  $L_s$ ,  $L_{p1}$  y  $L_{p2}$  respectivamente. Las métricas  $L_s$  y  $L_{p1}$  están relacionadas con el mensaje, mientras que la métrica  $L_{p2}$  está relacionada con el mensaje entrelazado. Estas métricas se almacenan en el Banco de Memoria de Entrada **701**. Dicha unidad es capaz de entregar a cada decodificador paralelo dentro del banco de decodificadores paralelos el sub-bloque correspondiente. Además, el Banco de Memoria de Entrada **701** entrelaza las métricas  $L_s$  cuando lleva a cabo una semi-iteración, pudiendo así entregar dichas métricas al Banco de Decodificadores SOVA fw(hacia delante) y bw (hacia atrás) **704**.

El Banco de Decodificadores SOVA fw/bw **704** calcula métricas a-posteriori en la dirección hacia delante (iteraciones impares) y hacia atrás (iteraciones pares), respectivamente. De este modo, tanto el Banco de Memoria de Entrada **701** como el Banco de Memoria Extrínseca **705** pueden entregar datos al Banco de Decodificadores SOVA fw/bw **704** en ambas direcciones. La Unidad de Control **702** determina el comportamiento de las diferentes unidades dependiendo de la iteración o semi-iteración que se lleve a cabo en cada momento. La Unidad de Cálculo Extrínseca **706** calcula la información extrínseca utilizando las métricas LLR del canal y las métricas a-posteriori. Esta unidad lleva a cabo una resta, aunque también puede llevar a cabo una multiplicación por cierto factor de escala, y/o una compresión o cuantificación de las métricas. En la figura, los buses que entregan  $P$  métricas LLR por ciclo se identifican con la letra  $P$ .

Nótese que a partir de la arquitectura representada en la Fig. 7 se puede obtener fácilmente la de un turbo decodificador en serie no paralelo: En tal caso (decodificador en serie no paralelo)  $P$  sería igual a 1, y por tanto el elemento **704** contendría tan sólo un decodificador SOVA fw/bw **703**; además los elementos **701** y **705** tendrían que entregar información relativa a un solo bloque, en lugar de entregar información simultánea de varios sub-bloques, es decir, los buses que aparecen en la figura conectando a los elementos **701** y **705** con **704** sólo entregarían una métrica por bus ( $P=1$ ); por último, como en la decodificación no paralela se tiene un bloque a decodificar cuyos estados de inicio y fin son conocidos, no hay que aplicar ningún tipo de solape lo cual simplifica el comportamiento de los elementos **701** y **705**.

Cada uno de los decodificadores paralelo SOVA fw/bw **703** tiene la estructura que se muestra en la Fig. 8. La Unidad Recursiva fw/bw**801** lleva a cabo las siguientes funciones: (i) calcular la métrica de camino o PM para cada uno de los  $N$  estados; (ii) seleccionar el camino superviviente y (iii) obtener el peso de esta decisión como la diferencia de las PMs entre los caminos supervivientes. La decisión relativa a los caminos supervivientes y los pesos correspondientes para cada estado se usa para construir los vectores  $Y$  y  $W$ , de longitud  $N$ , como se muestra en la Fig. 8. Esta unidad se encarga de realizar las funciones mencionadas anteriormente, tanto en la dirección hacia delante como en la dirección hacia atrás.

La Unidad de Control **804** implementa una máquina de estados finitos que activa las señales de control apropiadas para el resto de unidades presentadas en la Fig. 7, permitiendo a dichas unidades el realizar su función en la dirección hacia delante o hacia atrás, dependiendo de la iteración en la que se encuentre. Las decisiones se entregan a la Unidad de Memoria de Supervivientes fw/bw **803**. Esta unidad realiza el recorrido hacia atrás o TB para encontrar el camino ML. Como esta unidad debe llevar a cabo el cálculo del recorrido hacia atrás, la lógica

combinacional que calcula el estado previo para un estado determinado debe ser diferente dependiendo de la dirección. Una considera la máquina de estados hacia adelante y la otra la máquina de estados hacia atrás. Los estados de este camino se introducen a la Unidad de Proceso de Actualización fw/bw **805**. Como el elemento **803** requiere cierto número de ciclos de reloj para obtener los estados de dicho camino, el elemento retardador **802** aplica a los pesos  $W$  (diferencias entre las PMs) el mismo retardo en ciclos de reloj. Dicha unidad se encarga de la actualización de los pesos calculados por la Unidad Recursiva fw/bw **801**. Para llevar a cabo este proceso de actualización, hay que realizar un doble recorrido hacia atrás o TB: un TB obtiene el camino ML, mientras que el otro obtiene el camino competidor. En cada etapa en la que el bit decidido para ambos caminos difiere, se lleva a cabo una actualización. Dicha actualización consiste en la selección del peso mínimo entre el camino competidor y los caminos ML. Esta unidad también debe trabajar tanto en la dirección hacia adelante como hacia atrás al llevar a cabo el TB. Finalmente, tras el proceso de actualización, se obtienen las métricas LLR a-posteriori en las direcciones hacia adelante o hacia atrás dependiendo de la iteración actual.

En la Fig. 9 se ilustra la estructura de la Unidad Recursiva fw/bw **801**. Esta unidad se compone de  $N$  Elementos de Recursión **902**, que calculan la PM y llevan a cabo las tareas (i)-(iii) para cada estado. Para ello, cada Elemento de Recursión **902** necesita conocer las medidas  $L_s$ ,  $L_p$  y  $L_a$ . En el caso de una iteración completa,  $L_p$  es  $L_{p1}$ . En el caso de una semi-iteración,  $L_p$  es  $L_{p2}$  y  $L_s$  ha sido entrelazado por el Banco de Memoria de Entrada **701**. Por otro lado,  $L_a$  es la información a priori, o de manera equivalente las métricas extrínsecas de la semi-iteración previa.

Para calcular las PMs para cada estado  $S_i$  en la etapa  $k$ , se necesitan las PM de los caminos que confluyen en  $S_i$  en la etapa previa  $k-1$ . Uno de dichos caminos, el asociado con  $u_k=1$ , se identifica como  $PM_1(S_i)$ , mientras que el otro, asociado con  $u_k=0$ , se identifica como  $PM_0(S_i)$ . Como los caminos que confluyen en cada estado son diferentes en las direcciones hacia adelante y hacia atrás, se requiere una red de interconexión. Esta tarea es realizada por la Unidad de Conexión fw/bw **901**. Dicha unidad entrega el PM adecuado de la etapa previa a cada Elemento de Recursión **902**, dependiendo de la dirección (hacia adelante o hacia atrás). Una posible manera de implementar este elemento es mediante un banco de multiplexores y registros para almacenar la PM previa.

#### *ALSOVA para la turbo decodificación concurrente paralela*

La Fig. 10 muestra un diagrama de tiempos del método ALSOVA propuesto, para decodificación concurrente paralela, mientras que la Fig. 11 muestra una posible implementación de la misma. En esta implementación, el Banco de Memoria de Entrada **1101** debe alimentar simultáneamente dos Bancos de decodificadores SOVA fw/bw. De esta forma, éste debe proporcionar la métrica  $L_s$  tanto de forma entrelazada como no entrelazada. Existen dos bancos de información extrínseca. Por un lado, el Banco de Información Extrínseca 1 **1103** siempre desentrelaza las medidas extrínsecas. Sin embargo, en las iteraciones impares se lee y escribe hacia adelante mientras que en las iteraciones pares se lee y escribe hacia atrás. Por otro lado, el Banco de Información Extrínseca 2 **1104** siempre entrelaza las medidas extrínsecas. Este banco trabaja hacia adelante en iteraciones impares y hacia atrás en las pares. Las Unidades de Cálculo Extrínseco **706** y el Banco de decodificadores SOVA fw/bw son equivalentes al caso mostrado en la sección anterior.

*ALSOVA-PMSBx para la turbo decodificación en serie paralela*

La técnica PMSBx puede combinarse con los algoritmos SOVA, ALSOVA y BISOVA. En los casos de BISOVA-PMSBx y ALSOVA-PMSBx existen dos tipos diferentes de métricas de camino o PMs y TBs, una en la dirección hacia adelante y la otra en la dirección hacia atrás.

- 5 Para el caso BISOVA, este hecho no supone ninguna modificación con respecto a SOVA-PMSBx puesto que la inicialización se realiza dentro de dos bancos independientes de decodificadores SOVA. Sin embargo, en el caso de ALSOVA-PMSBx, se tienen que intercambiar las PMs entre dos iteraciones dado que las PMs de la iteración anterior han sido calculadas en una dirección diferente. Formalmente, la inicialización de las PMs puede ser  
10 descrita con la siguiente ecuación.

$$PM^{(i,n)}(s, k_{l,n}) = \begin{cases} PM^{(i-D,n-1)}(s, k_{l,n}), & n \neq 1 \\ \log(\delta(s)), & n=1 \end{cases} \quad (3)$$

donde el símbolo D representa el retardo en iteraciones que se aplica a las PMs que se intercambian entre decodificadores paralelos. Para el caso de SOVA y BISOVA D=1 mientras que para ALSOVA D=2. La etapa en la que se realiza la inicialización para el sub-bloque n es  
15  $k_{l,1}=0$  para  $n=1$  y  $k_{l,n}=(n-1)M$  para  $n>1$ .

La inicialización de estados de cara a hacer los TBs es la que sigue

$$S^{(i,n)}(s, k_{r,n}) = \begin{cases} S^{(i,n+1)}(s, k_{r,n}), & n \neq P \\ 0, & n=P \end{cases} \quad (4)$$

donde la etapa en la que se realiza el último TB es  $k_{r,P}=K+T$  para  $n=P$  y  $k_{r,n}=nM$  para  $n<P$ . La ecuación anterior es válida para SOVA-PMSBx, ALSOVA-PMSBx y BISOVA-PMSBx dado  
20 que los estados se intercambian en la misma iteración.

La arquitectura propuesta para un turbo decodificador serie paralelo usando ALSOVA-PMSBx se muestra en la Fig. 7. La dirección de decodificación se cambia a lo largo de las iteraciones (como se describió anteriormente) con objeto de mejorar la capacidad correctora sin aumentar significativamente la latencia o el consumo de área. El Banco de Memoria a la Entrada **701** y el Banco de Memoria Extrínseco **705** son capaces de alimentar el Banco de decodificadores SISO **704** en la direcciones hacia adelante y hacia atrás. La Unidad de Control **702** determina el comportamiento de la diferentes unidades dependiendo de la iteración (o media iteración) a realizar en cada momento. No obstante, el caso del Banco de decodificadores SISO paralelo **704** tiene otra arquitectura que se muestra en la Fig. 12, conforme a la cuál dicho Banco de  
25 decodificadores SISO paralelo **704** comprende, entre otros, una Unidad de Control **1304**. Como se puede observar, las conexiones entre los decodificadores SOVA fw/bw **703** permiten el intercambio de información para paliar la incertidumbre en el borde de los sub-bloques. El intercambio de PM se realiza gracias a la unidad **1303**, la cual aplica un retardo a las PMs de dos iteraciones completas (D=2), o lo que es lo mismo, a cuatro medias iteraciones. El  
30 intercambio de estados se realiza gracias a registros de memoria **1302** que almacenan dicha información durante la iteración en la que es requerida. Además, los decodificadores **703** son decodificadores fw/bw con una arquitectura como la mostrada en la Fig. 8. Estos decodificadores son capaces de realizar la decodificación SOVA tanto hacia adelante como hacia atrás, dependiendo de la iteración en cuestión.

**REIVINDICACIONES**

1. Método para turbo decodificación iterativa de baja tasa de error y baja complejidad caracterizado por que comprende:

- 5 a. El almacenamiento de las métricas LLR en un banco de memoria de entrada que posteriormente entrega, bien el bloque correspondiente a un decodificador no paralelo que trabaja hacia delante (fw) y hacia atrás (bw), bien los sub-bloques correspondientes a dos o más decodificadores paralelos fw/bw dichos dos o más decodificadores paralelos fw/bw comprendidos en uno o más bancos paralelos de decodificadores SOVA fw/bw;
- 10 b. el cálculo de métricas a-posteriori bien por dicho decodificador no paralelo fw/bw bien por dichos dos o más decodificadores paralelos fw/bw;
- c. el cálculo por una o más unidades de cálculo extrínsecas de la información extrínseca a partir de las métricas LLR y las métricas a-posteriori;
- 15 d. y la comparación de las medidas LLR a-posteriori calculadas hacia delante y hacia atrás a lo largo de distintas iteraciones mediante el intercambio de la información extrínseca entre iteraciones calculando para ello las medidas LLR hacia adelante en las iteraciones pares, y las medidas LLR hacia atrás en las iteraciones impares conforme a la ecuación (2)

$$L^{(i)}(u_k) = \begin{cases} L_f^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) = 1 \\ L_b^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) = 0 \end{cases}$$

20 siendo  $\lfloor i \rfloor$  la función suelo.

2. Método según la reivindicación anterior caracterizado por que el banco de memoria de entrada entrega el bloque correspondiente a un decodificador no paralelo fw/bw, dicho decodificador no paralelo fw/bw responsable del cálculo de métricas a-posteriori.

25

3. Método según la reivindicación 1 caracterizado por que comprende:

- 30 a. El almacenamiento de las métricas LLR en un banco de memoria de entrada que posteriormente entrega los sub-bloques correspondientes a dos o más decodificadores paralelos fw/bw, dichos dos o más decodificadores paralelos fw/bw comprendidos en uno o más bancos paralelos de decodificadores SOVA fw/bw;
- b. El entrelazado de las métricas LLR antes de la entrega de los sub-bloques correspondientes por parte del banco de memoria de entrada a dos o más decodificadores paralelos fw/bw, dichos dos o más decodificadores paralelos fw/bw comprendidos en un banco paralelo de decodificadores SOVA fw/bw;
- 35 c. el cálculo de métricas a-posteriori por dichos dos o más decodificadores paralelos fw/bw;
- d. el cálculo por una o más unidades de cálculo extrínsecas de la información extrínseca a partir de las métricas LLR y las métricas a-posteriori;
- 40 e. y la comparación de las medidas LLR a-posteriori calculadas hacia delante y hacia atrás a lo largo de distintas iteraciones mediante el intercambio de la información extrínseca entre iteraciones calculando para ello las medidas LLR

hacia adelante en las iteraciones pares, y las medidas LLR hacia atrás en las iteraciones impares conforme a la ecuación (2)

$$L^{(i)}(u_k) = \begin{cases} L_f^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) == 1 \\ L_b^{(i)}(u_k) & \text{if } \text{mod}(\lfloor i \rfloor, 2) == 0 \end{cases}$$

siendo  $\lfloor i \rfloor$  la función suelo.

5

4. Método según la reivindicación anterior caracterizado por que comprende el uso de las métricas de camino (PMs) en el borde derecho de los sub-bloques como métricas de inicialización para el siguiente sub-bloque de la siguiente iteración; y el uso de los estados del camino (ML) del borde izquierdo de los sub-bloques para hacer el último recorrido hacia atrás (TB) desde un estado fiable del decodificador SOVA fw/bw previo, dichos estados de camino (ML) intercambiables en la misma iteración.

10

5. Método según la reivindicación 3 que comprende

15

a. La entrega de los sub-bloques correspondientes por parte del banco de memoria de entrada a dos o más decodificadores paralelos fw/bw, dichos dos o más decodificadores paralelos fw/bw comprendidos en dos o más bancos paralelos de decodificadores SOVA fw/bw;

20

b. el cálculo por dos o más unidades de cálculo extrínsecas de la información extrínseca a partir de las métricas LLR y las métricas a-posteriori;

c. el desentrelazado de la información extrínseca calculada por al menos una de las unidades de cálculo extrínseca por al menos un banco de información extrínseca que trabaja hacia delante (fw) en las iteraciones impares mientras que trabaja hacia atrás (bw) en las iteraciones pares;

25

d. y el entrelazado de la información extrínseca calculada por al menos una de las unidades de cálculo extrínseca no implicadas en la etapa (c) por al menos un banco de información extrínseca no implicado en la etapa (c) y que trabaja hacia atrás (bw) en las iteraciones pares mientras que trabaja hacia delante (fw) en las iteraciones impares.

30

6. Sistema para turbo decodificación iterativa de baja tasa de error y baja complejidad que implementa un método conforme a la reivindicación 2 caracterizado por que comprende un banco de memoria de entrada, una unidad de control, un decodificador SOVA no paralelo fw/bw, un banco de memoria extrínseca y una unidad de cálculo extrínseco; de forma que,

35

a. el banco de memoria de entrada almacena las métricas LLR, las entrelaza cuando lleva a cabo una semi-iteración, y proporciona el bloque al correspondiente decodificador SOVA no paralelo fw/bw;

b. el decodificador SOVA no paralelo fw/bw calcula métricas a-posterior hacia delante en las iteraciones impares y hacia atrás en las iteraciones pares;

40

c. la unidad de control determina el comportamiento de las diferentes unidades dependiendo de la iteración o semi-iteración que se lleve a cabo en cada momento;

d. y la unidad de cálculo extrínseco calcula la información extrínseca utilizando las métricas LLR y las métricas a-posteriori.

7. Sistema para turbo decodificación iterativa de baja tasa de error y baja complejidad que implementa un método conforme a la reivindicación 3 caracterizado por que comprende un Banco de Memoria de Entrada **701**, una Unidad de Control **702**, un Banco Paralelo de Decodificadores SOVA fw/bw **704** que comprende dos o más decodificadores SOVA fw/bw **703**, un Banco de Memoria Extrínseca **705**, y una Unidad de Cálculo Extrínseco **706**; de forma que
- a. el Banco de Memoria de Entrada **701** almacena las métricas LLR, las entrelaza cuando lleva a cabo una semi-iteración, y proporciona el sub-bloque correspondiente a cada decodificador SOVA fw/bw **703** comprendido en el Banco Paralelo de Decodificadores SOVA **704**;
  - b. cada decodificador SOVA fw/bw **703** comprendido en el Banco Paralelo de Decodificadores SOVA fw/bw **704** calcula métricas a-posterior hacia delante en las iteraciones impares y hacia atrás en las iteraciones pares;
  - c. la Unidad de Control **702** determina el comportamiento de las diferentes unidades dependiendo de la iteración o semi-iteración que se lleve a cabo en cada momento;
  - d. y la Unidad de Cálculo Extrínseca **1206** calcula la información extrínseca utilizando las métricas LLR y las métricas a-posteriori.
8. Sistema según la reivindicación anterior caracterizado por que el Banco Paralelo de Decodificadores SOVA fw/bw **704** comprende
- a. Uno o más elementos **1303** que aplican un retardo de dos iteraciones completas (cuatro semi-iteraciones) a las PMs para facilitar su intercambio entre los decodificadores;
  - b. y uno o más registros **1302** que almacenan el estado durante la misma iteración en la que son usados.
9. Sistema según la reivindicación anterior caracterizado por que cada decodificador paralelo SOVA fw/bw **703** comprende:
- a. Una Unidad Recursiva **801** que realiza las siguientes funciones, tanto hacia delante como hacia atrás: (i) calcular la métrica de camino o PM para cada uno de los N estados; (ii) seleccionar el camino superviviente y (iii) obtener el peso de esta decisión como la diferencia de las PMs entre los caminos supervivientes, utilizándose dichas decisiones relativas a los caminos supervivientes y los pesos correspondientes para cada estado, para construir los vectores Y y W, de longitud N;
  - b. Un retardador **802** que aplica a los pesos W (diferencias entre las PMs) un retardo en ciclos de reloj;
  - c. Una Unidad de Memoria de Supervivientes **803** que, en base a las decisiones relativas que recibe, realiza el recorrido hacia atrás (TB) para encontrar el camino ML
  - d. Una Unidad de Control **804** que implementa una máquina de estados finitos que activa las señales de control apropiadas para el resto de unidades del decodificador paralelo SOVA **703**, permitiendo a dichas unidades el realizar su función en la dirección hacia delante o hacia atrás dependiendo de la iteración en la que se encuentre;

- 5 e. y una Unidad de Actualización **805** que, en función de los estados del camino ML, actualiza los pesos calculados por la Unidad Recursiva **801** realizando un doble recorrido hacia atrás (un TB obtiene el camino ML, mientras que el otro obtiene el camino competidor) de forma que, en cada etapa en la que el bit decidido para ambos caminos difiere, se lleva una actualización que consiste en la selección del peso mínimo entre el camino competidor y los caminos ML, obteniéndose finalmente las métricas LLR a-posterior en las direcciones hacia delante o hacia atrás dependiendo de la iteración actual.
- 10 10. Sistema según la reivindicación anterior caracterizado por que la Unidad Recursiva **801** comprende:
- 15 a. Una Red de Conexión **901** que entrega el PM adecuado de la etapa previa a cada Elemento Recursivo **902** dependiendo de la dirección (hacia delante o hacia atrás), implementando para ello, por ejemplo, un banco de multiplexores y registros para almacenar la PM previa;
- b. y dos o más Elementos Recursivos **902** que (i) calculan la métrica de camino o PM para cada uno de los N estados; (ii) seleccionan el camino superviviente y (iii) obtienen el peso de esta decisión como la diferencia de las PMs entre los caminos supervivientes.
- 20 11. Sistema para turbo decodificación iterativa de baja tasa de error y baja complejidad conforme cualquier de las reivindicaciones 7 a 10 que implementa un método conforme a la reivindicación 4 caracterizado por que la Unidad de Control **1304** garantiza el intercambio de estados y PMs conforme con el método objeto de la reivindicación 4.
- 25 12. Sistema para turbo decodificación iterativa de baja tasa de error y baja complejidad que implementa un método conforme a la reivindicación 5 caracterizado por que comprende un Banco de Memoria de Entrada **1101**, una Unidad de Control **1102**, dos o más Bancos Paralelos de Decodificadores SOVA fw/bw **704** que comprenden dos o más decodificadores SOVA fw/bx, uno o más Bancos de Memoria Extrínsecos **1103**, uno o más Bancos de Memoria Extrínsecos **1104**, y dos o más Unidades de Cálculo Extrínseco **706**; de forma que
- 30 a. el Banco de Memoria de Entrada **1101** alimenta simultáneamente a los dos o más decodificadores SOVA fw/bw **703** comprendidos en los dos o más Bancos Paralelos de Decodificadores SOVA fw/bw **704**;
- 35 b. el al menos un Banco de Memoria Extrínseco **1103** siempre desentrelaza la información extrínseca, trabajando hacia delante en las iteraciones impares y hacia atrás en las iteraciones pares;
- 40 c. el al menos un Banco de Memoria Extrínseco **1104** siempre entrelaza la información extrínseca, trabajando hacia delante en las iteraciones impares y hacia atrás en las iteraciones pares;
- d. la Unidad de Control **702** determina el comportamiento de las diferentes unidades dependiendo de la iteración o semi-iteración que se lleve a cabo en cada momento;
- 45 e. y las dos o más Unidades de Cálculo Extrínseco **706** calculan la información extrínseca utilizando las métricas LLR y las métricas a-posteriori.

13. Sistema según la reivindicación anterior caracterizado por que el Banco Paralelo de Decodificadores SOVA fw/bw **704** comprende
- 5
- a. Uno o más elementos que aplican un retardo de dos iteraciones completas (cuatro semi-iteraciones) a las PMs para facilitar su intercambio entre los decodificadores;
  - b. y uno o más registros que almacenan el estado durante la misma iteración en la que son usados.
14. Sistema según la reivindicación anterior caracterizado por que cada decodificador paralelo SOVA fw/bw **703** comprende:
- 10
- a. Una Unidad Recursiva **801** que realiza las siguientes funciones, tanto hacia delante como hacia atrás: (i) calcular la métrica de camino o PM para cada uno de los N estados; (ii) seleccionar el camino superviviente y (iii) obtener el peso de esta decisión como la diferencia de las PMs entre los caminos supervivientes, utilizándose dichas decisiones relativas a los caminos supervivientes y los pesos correspondientes para cada estado, para construir los vectores Y y W, de longitud N;
  - 15
  - b. Un retardador **802** que aplica a los pesos W (diferencias entre las PMs) un retardo en ciclos de reloj;
  - 20
  - c. Una Unidad de Memoria de Supervivientes **803** que, en base a las decisiones relativas que recibe, realiza el recorrido hacia atrás (TB) para encontrar el camino ML;
  - d. Una Unidad de Control **804** que implementa una máquina de estados finitos que activa las señales de control apropiadas para el resto de unidades del decodificador paralelo SOVA fw/bw **703**, permitiendo a dichas unidades el realizar su función en la dirección hacia delante o hacia atrás dependiendo de la iteración en la que se encuentre;
  - 25
  - e. y una Unidad de Actualización **805** que, en función de los estados del camino ML, actualiza los pesos calculados por la Unidad Recursiva **801** realizando un doble recorrido hacia atrás (un TB obtiene el camino ML, mientras que el otro obtiene el camino competidor) de forma que, en cada etapa en la que el bit decidido para ambos caminos difiere, se lleva una actualización que consiste en la selección del peso mínimo entre el camino competidor y los caminos ML, obteniéndose finalmente las métricas LLR a-posterior en las direcciones hacia delante o hacia atrás dependiendo de la iteración actual.
  - 30
  - 35
15. Sistema según la reivindicación anterior caracterizado por que la Unidad Recursiva **801** comprende:
- a. Una Red de Conexión **901** que entrega el PM adecuado de la etapa previa a cada Elemento Recursivo **902** dependiendo de la dirección (hacia delante o hacia atrás), implementando para ello, por ejemplo, un banco de multiplexores y registros para almacenar la PM previa;
  - 40
  - b. y dos o más Elementos Recursivos **902** que (i) calculan la métrica de camino o PM para cada uno de los N estados; (ii) seleccionan el camino superviviente y (iii) obtienen el peso de esta decisión como la diferencia de las PMs entre los caminos supervivientes.
  - 45

16. Programa informático adaptado para la realización de un método conforme a cualquiera de las reivindicaciones 1 a 5, o que comprende instrucciones para llevar a cabo las etapas comprendidas en un método conforme a cualquiera de las reivindicaciones 1 a 5.
- 5 17. Medio de almacenamiento legible en computador que comprende un programa informático conforme a la reivindicación anterior, o instrucciones para hacer que un aparato de procesamiento de datos lleve a cabo las etapas comprendidas en un método conforme a cualquiera de las reivindicaciones 1 a 5.
- 10 18. Medio portador de grabación con un programa informático conforme a la reivindicación 16 grabado en dicho medio portador de grabación.
19. Onda portadora de señal portando señales que incorporan un programa informático conforme a la reivindicación 16.

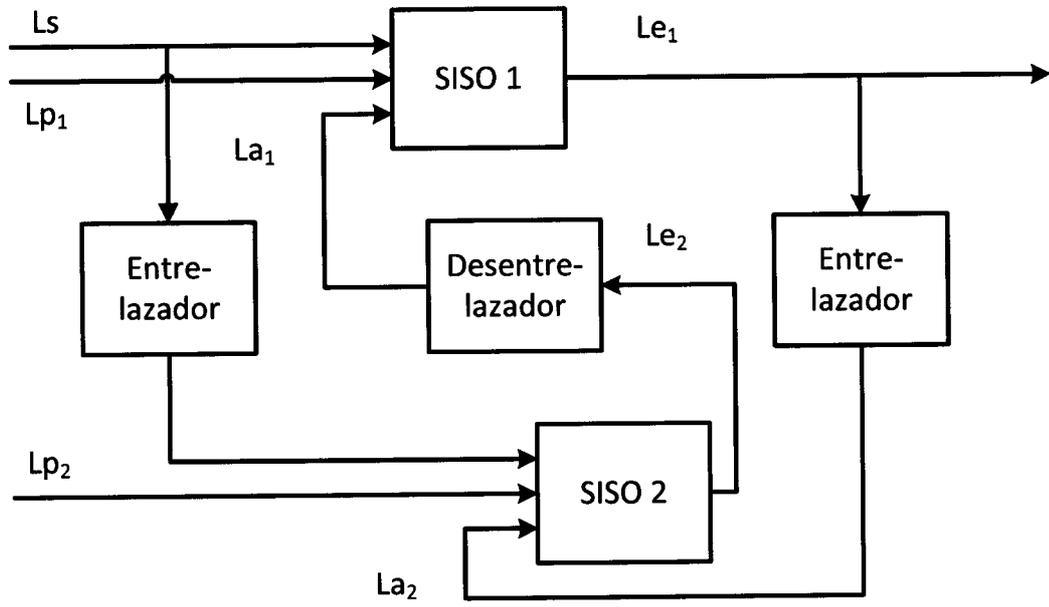


Fig. 1

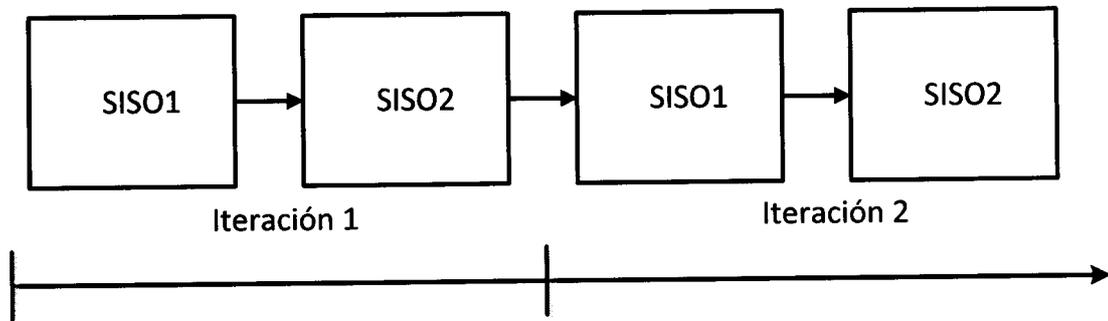


Fig. 2

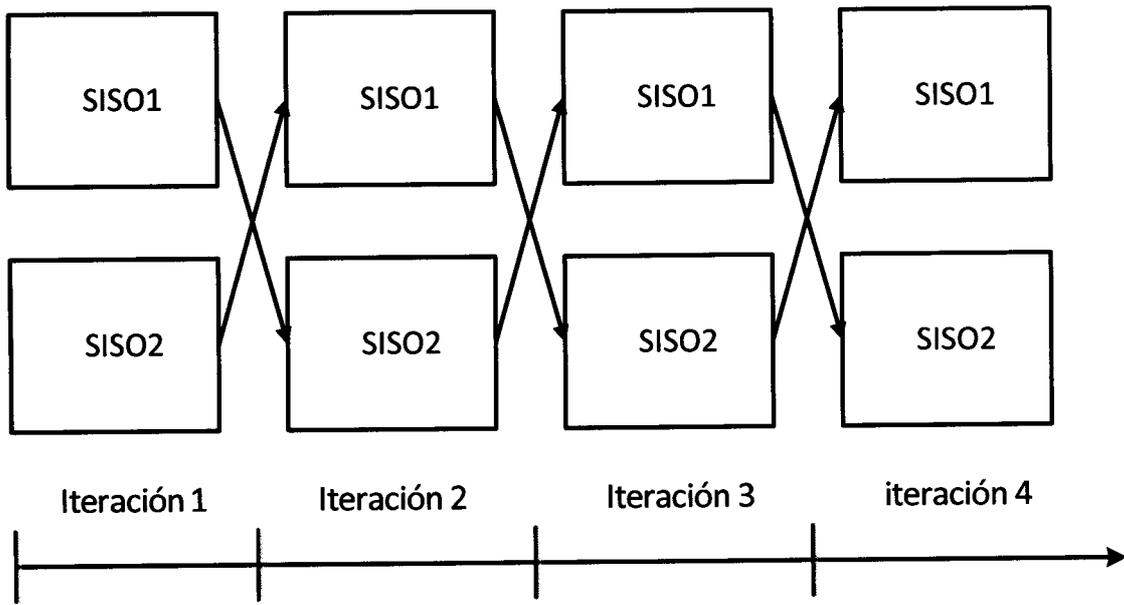


Fig. 3

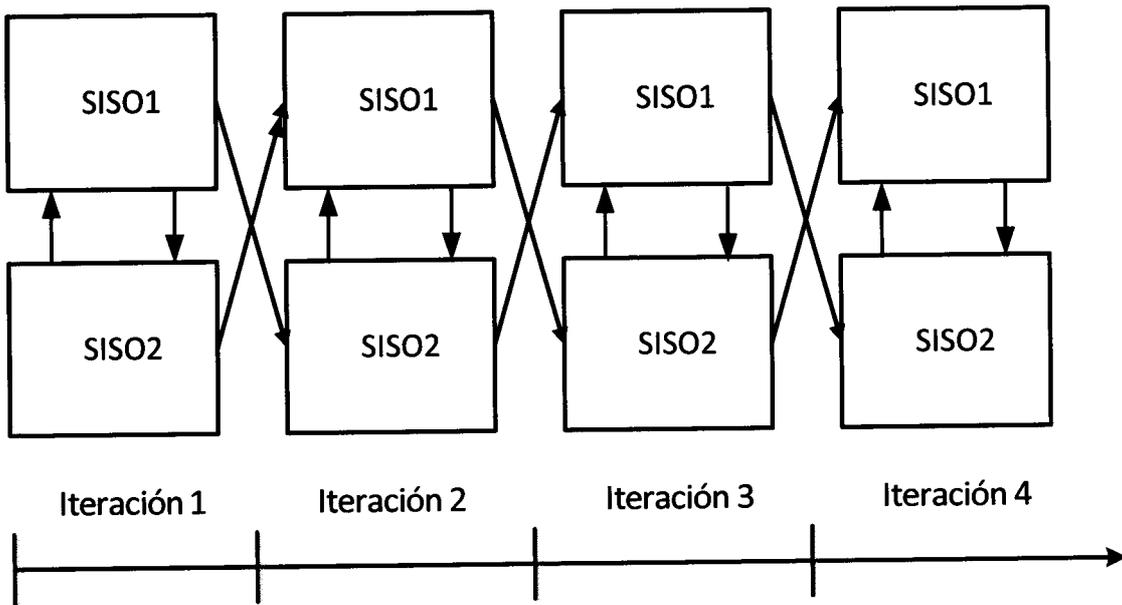


Fig. 4

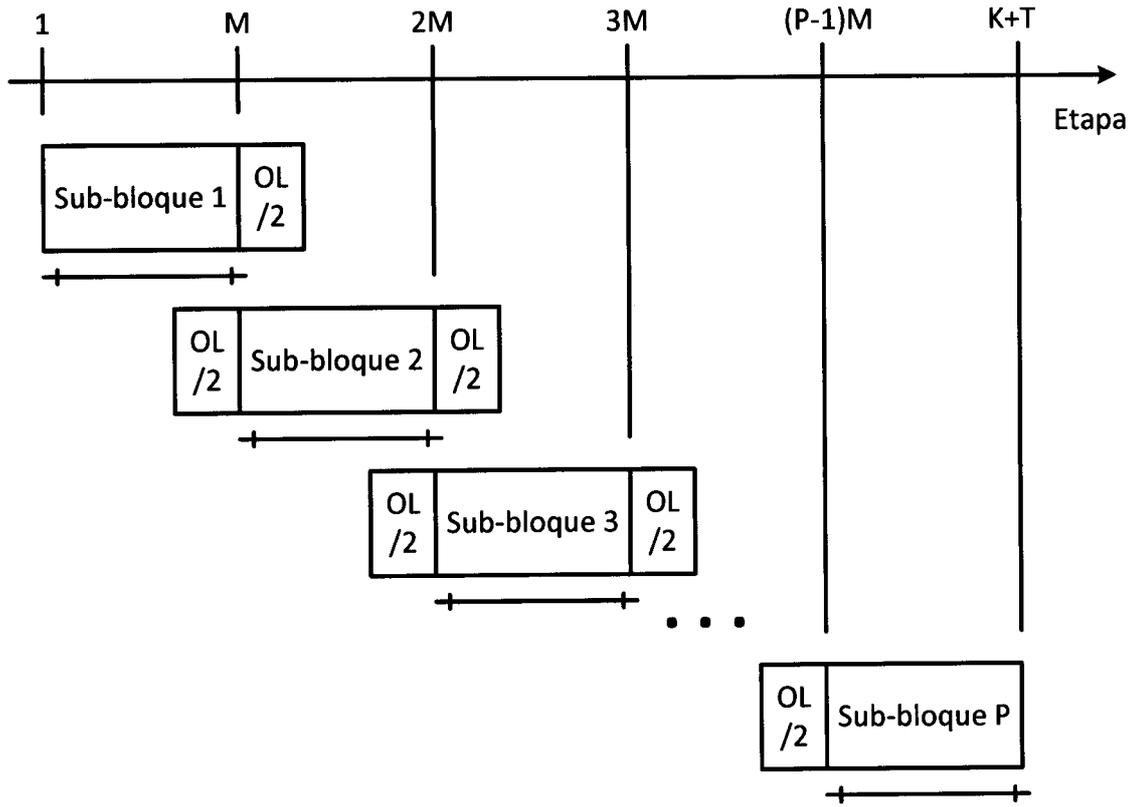


Fig. 5

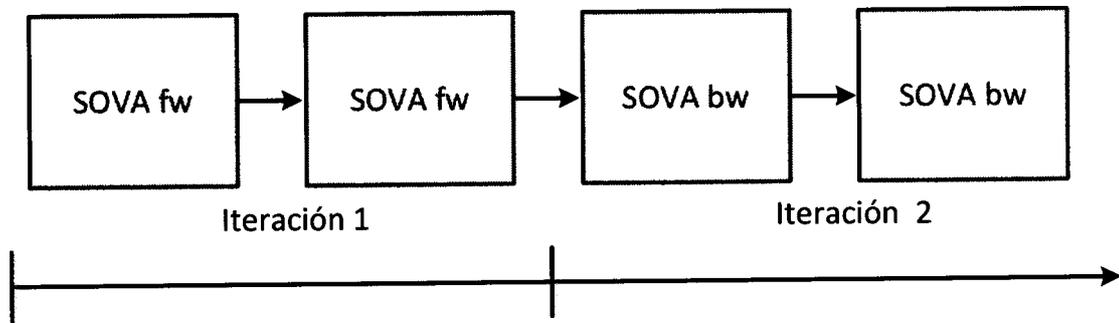


Fig. 6

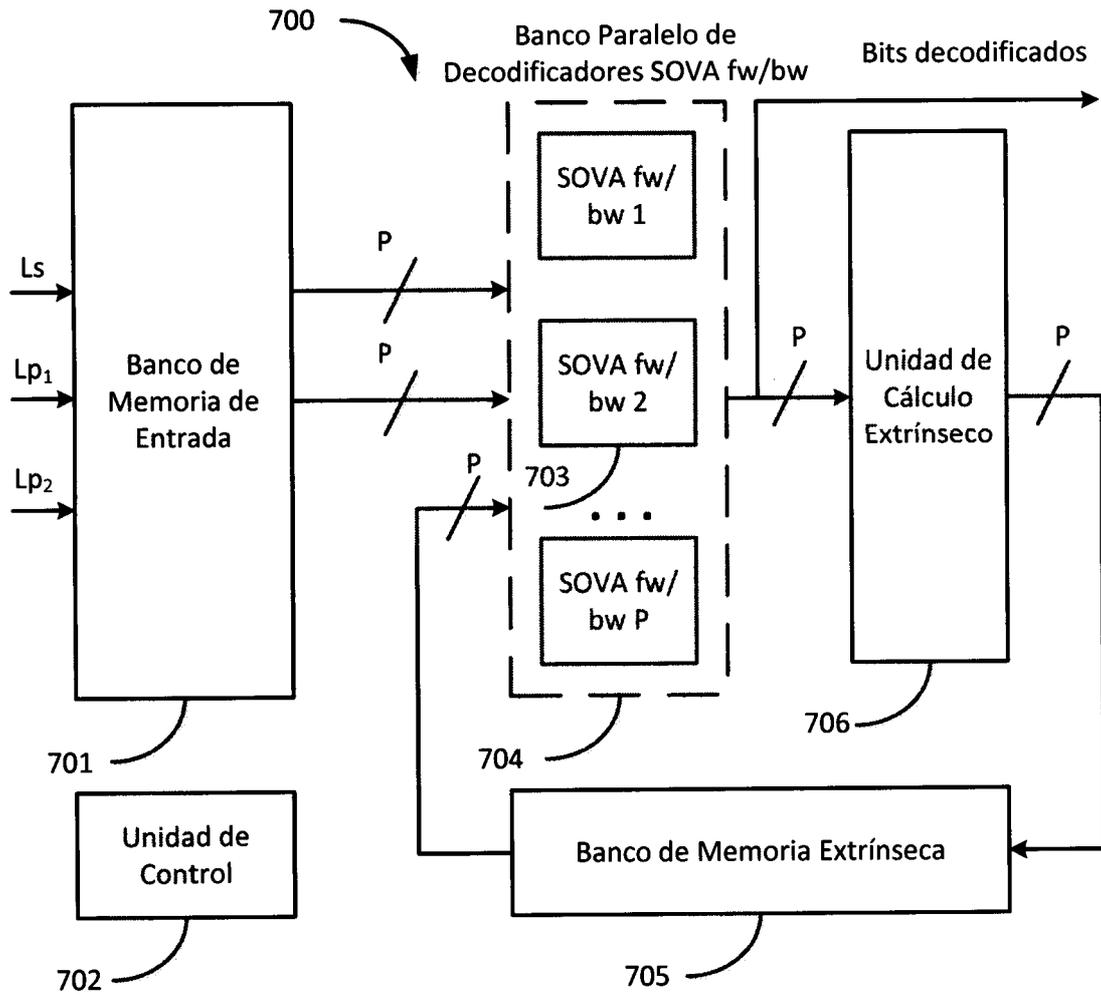


Fig. 7

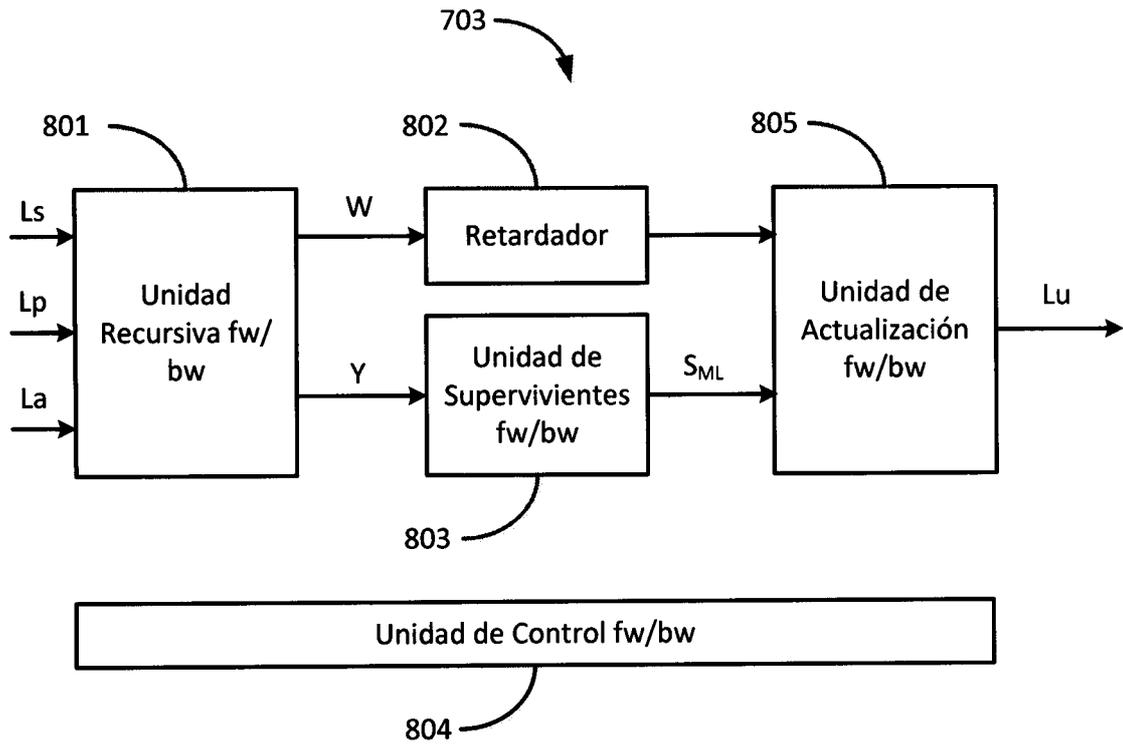


Fig. 8

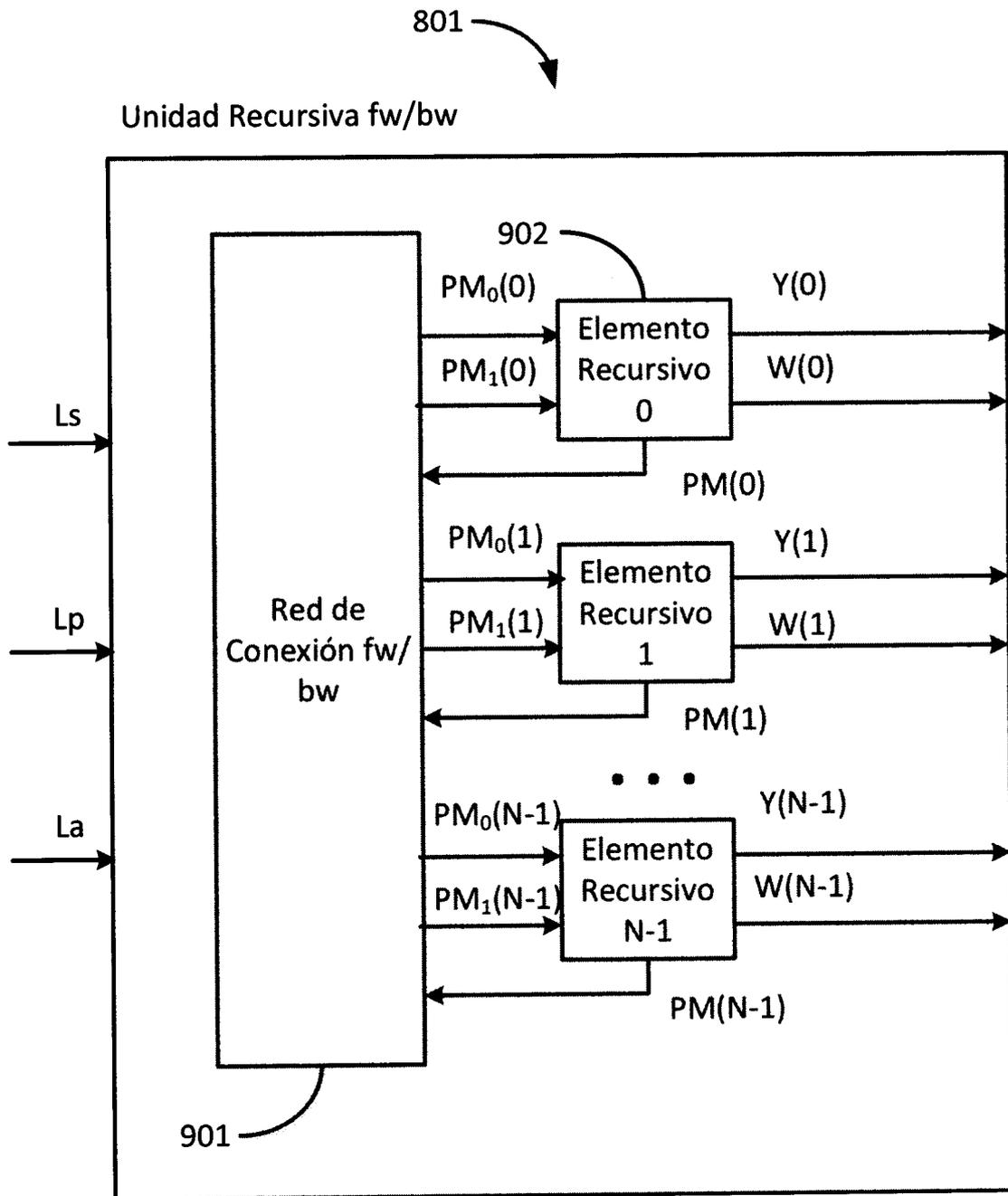


Fig. 9

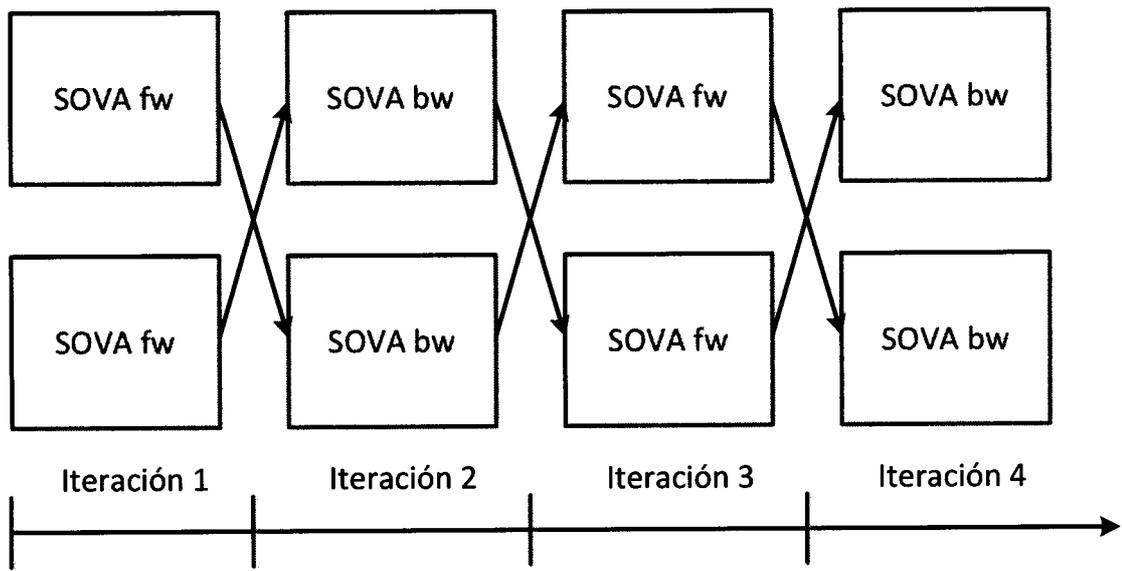


Fig. 10

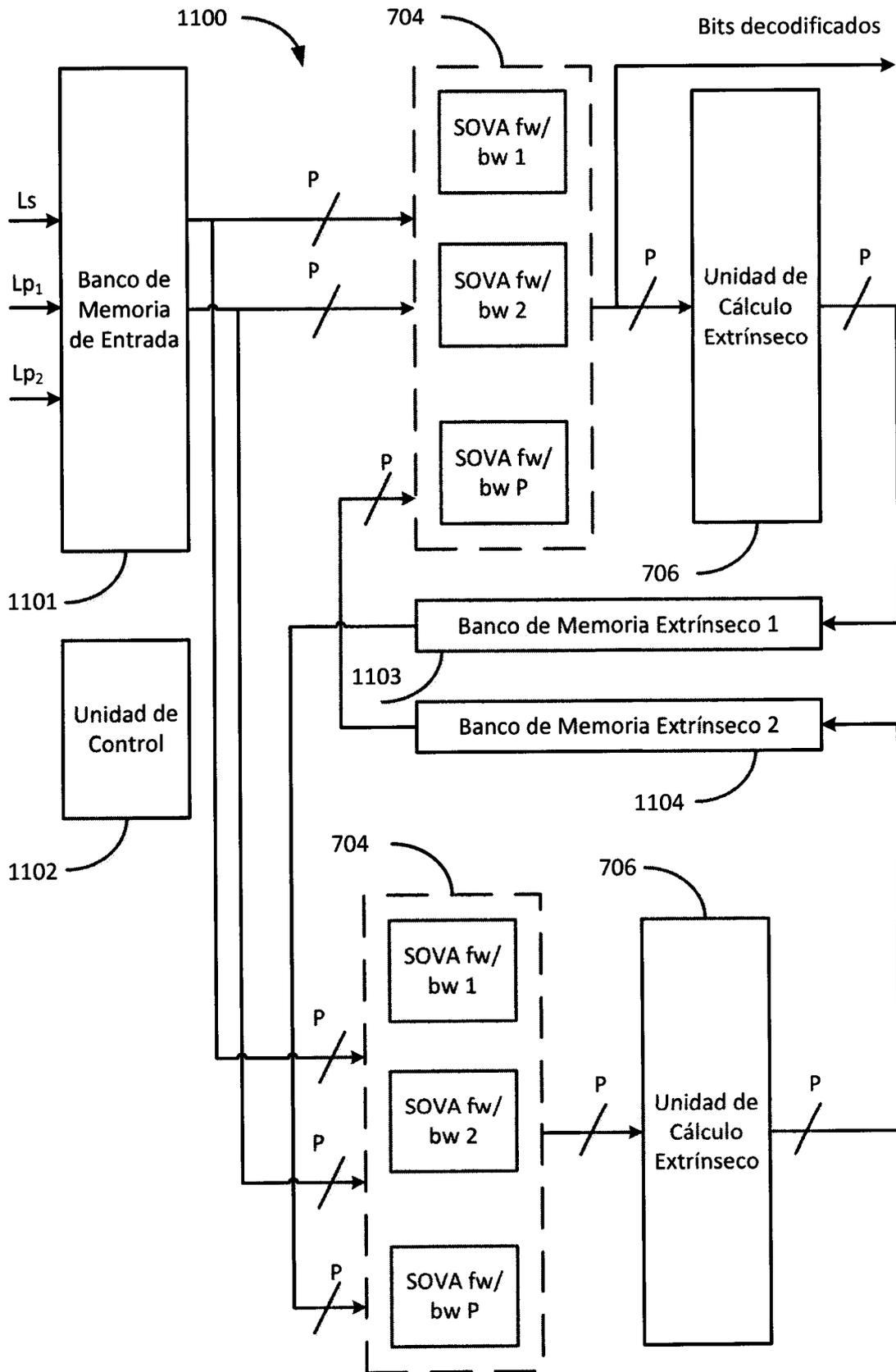


Fig. 11

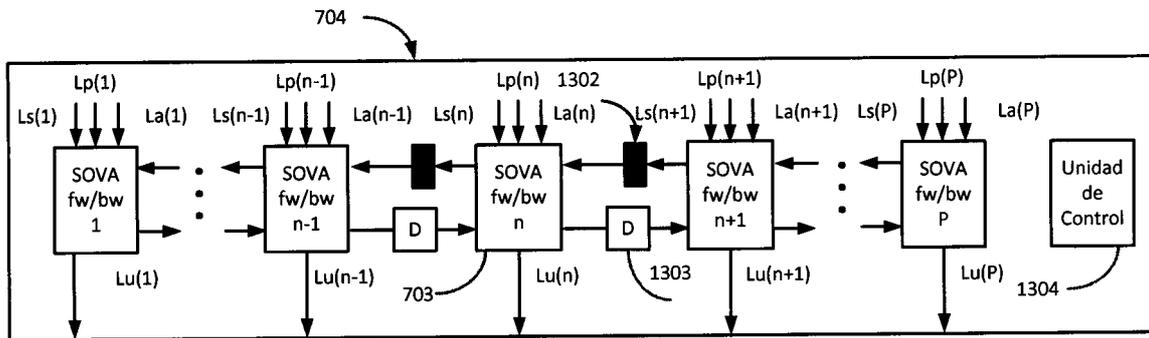


Fig. 12



- ②1 N.º solicitud: 201400891  
②2 Fecha de presentación de la solicitud: 06.11.2014  
③2 Fecha de prioridad:

INFORME SOBRE EL ESTADO DE LA TECNICA

⑤1 Int. Cl.: **H03M13/29** (2006.01)

DOCUMENTOS RELEVANTES

Categoría	⑤6 Documentos citados	Reivindicaciones afectadas
A	EP 1398881 A1 (ST MICROELECTRONICS NV et al.) 17.03.2004, párrafos [1],[4-6],[9-18],[25-32],[55-56],[60-61],[84-86],[98-136],[141],[153],[155],[167],[171-172]; figuras 3-4,6-7,9.	1-19
A	EP 1391995 A2 (NEC ELECTRONICS CORP) 25.02.2004, párrafos [1],[3-5],[12-14],[18-19],[24],[37-39],[44],[46-51],[58].	1-19
A	US 2001044919 A1 (EDMONSTON BRIAN S et al.) 22.11.2001, párrafos [4],[7],[9-10],[12],[14-18],[21-23],[34-38]; figura 4.	4,16-19
A	CN 101373978 A (HUAWEI TECH CO LTD) 25.02.2009, resumen; figura 9.	1-19
A	U. DASGUPTA et al. "Parallel decoding of turbo codes using soft output T-algorithms". IEEE VTC 2000. Vol 3. Septiembre 2000 [en línea] [recuperado el 25.01.2016] Recuperado de Internet < URL: <a href="http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=886292">http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=886292</a> > < DOI: 10.1109/VETECF.2000.886292 >	
A	ZHIYONG HE et al. "Highly-parallel decoding architectures for convolutional turbo codes". IEEE Transactions on very large scale integration (VLSI) systems. Vol 14, nº 10, Octubre 2006 [en línea] [recuperado el 25.01.2016] Recuperado de Internet <URL: <a href="http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1715351&amp;tag=1">http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1715351&amp;tag=1</a> > <DOI: 10.1109/TVLSI.2006.884172 >	4,16-19

Categoría de los documentos citados

X: de particular relevancia  
Y: de particular relevancia combinado con otro/s de la misma categoría  
A: refleja el estado de la técnica

O: referido a divulgación no escrita  
P: publicado entre la fecha de prioridad y la de presentación de la solicitud  
E: documento anterior, pero publicado después de la fecha de presentación de la solicitud

**El presente informe ha sido realizado**

para todas las reivindicaciones

para las reivindicaciones nº:

Fecha de realización del informe  
19.02.2016

Examinador  
J. M. Vázquez Burgos

Página  
1/5

Documentación mínima buscada (sistema de clasificación seguido de los símbolos de clasificación)

H03M

Bases de datos electrónicas consultadas durante la búsqueda (nombre de la base de datos y, si es posible, términos de búsqueda utilizados)

INVENES, EPODOC,WPI, INTERNET

Fecha de Realización de la Opinión Escrita: 19.02.2016

**Declaración**

<b>Novedad (Art. 6.1 LP 11/1986)</b>	Reivindicaciones 1-19	<b>SI</b>
	Reivindicaciones	<b>NO</b>
<b>Actividad inventiva (Art. 8.1 LP11/1986)</b>	Reivindicaciones 1-19	<b>SI</b>
	Reivindicaciones	<b>NO</b>

Se considera que la solicitud cumple con el requisito de aplicación industrial. Este requisito fue evaluado durante la fase de examen formal y técnico de la solicitud (Artículo 31.2 Ley 11/1986).

**Base de la Opinión.-**

La presente opinión se ha realizado sobre la base de la solicitud de patente tal y como se publica.

**1. Documentos considerados.-**

A continuación se relacionan los documentos pertenecientes al estado de la técnica tomados en consideración para la realización de esta opinión.

Documento	Número Publicación o Identificación	Fecha Publicación
D01	EP 1398881 A1 (ST MICROELECTRONICS NV et al.)	17.03.2004
D02	EP 1391995 A2 (NEC ELECTRONICS CORP)	25.02.2004
D03	US 2001044919 A1 (EDMONSTON BRIAN S et al.)	22.11.2001
D04	CN 101373978 A (HUAWEI TECH CO LTD)	25.02.2009
D05	U. DASGUPTA et al. "Parallel decoding of turbo codes using soft output T-algorithms". IEEE VTC 2000. Vol 3. Septiembre 2000 [en línea] [recuperado el 25.01.2016] Recuperado de Internet < URL: <a href="http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=886292">http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=886292</a> > < DOI: 10.1109/VETECF.2000.886292 >	28.09.2000
D06	ZHIYONG HE et al. "Highly-parallel decoding architectures for convolutional turbo codes?". IEEE Transactions on very large scale integration (VLSI) systems. Vol 14, nº 10, Octubre 2006 [en línea] [recuperado el 25.01.2016] Recuperado de Internet <URL: <a href="http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1715351&amp;tag=1">http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1715351&amp;tag=1</a> > <DOI: 10.1109/TVLSI.2006.884172 >	31/09/2006

**2. Declaración motivada según los artículos 29.6 y 29.7 del Reglamento de ejecución de la Ley 11/1986, de 20 de marzo, de Patentes sobre la novedad y la actividad inventiva; citas y explicaciones en apoyo de esta declaración**

La invención divulga sistemas y métodos de turbo decodificación con bajas tasa de error y complejidad. Un primer método consiste en: almacenar las métricas LLR en un banco de memoria de entrada, cálculo de las métricas a posteriori por los decodificadores, cálculo de la información extrínseca por unidades de cálculo ad hoc, y comparación de las medidas LLR a posteriori a lo largo de diferentes iteraciones. La invención incluye sistemas que implementan ambos métodos. Un segundo método utiliza métricas de camino en el borde derecho de los sub-bloques como métricas de inicialización para el siguiente sub-bloque de la siguiente iteración; y utiliza los estados del camino del borde izquierdo de los sub-bloques para hacer el último recorrido hacia atrás desde un estado fiable del decodificador SOVA previo.

El documento D01 del estado de la técnica más próximo presenta una serie de mejoras en los sistemas y métodos de decodificación de turbo códigos, enfocados a optimizar la memoria y el procesamiento de ambos.

Reivindicación 1

Con el fin de ilustrar de la mejor manera las diferencias entre la invención reivindicada en 1 y el documento D01, se reproduce a continuación el texto de dicha reivindicación, extrayendo del mismo sus referencias originales, e insertando únicamente las de D01. Asimismo, aquellas partes del texto que no estuviesen recogidas en D01 se señalarían entre corchetes y en negrita.

Método para turbo decodificación iterativa de baja tasa de error y baja complejidad caracterizado por que comprende:

- El almacenamiento de las métricas LLR en un banco de memoria de entrada (párrafos 17,129) que posteriormente entrega, bien el bloque correspondiente a un decodificador no paralelo que trabaja hacia delante (fw) y hacia atrás (bw) (párrafos 17, 86, 99, 129-132, 171-172; figura 9), bien los subbloques correspondientes a dos o más decodificadores paralelos fw/bw dichos dos o más decodificadores paralelos fw/bw comprendidos en uno o más bancos paralelos de decodificadores SOVA fw/bw;
- el cálculo de métricas a-posteriori bien por dicho decodificador no paralelo fw/bw bien por dichos dos o más decodificadores paralelos fw/bw (párrafos 129-132; figura 9);
- el cálculo por una o más unidades de cálculo extrínsecas de la información extrínseca a partir de las métricas LLR y las métricas a-posteriori (párrafos 129-132; figura 9);
- [y la comparación de las medidas LLR a-posteriori calculadas hacia delante y hacia atrás a lo largo de distintas iteraciones mediante el intercambio de la información extrínseca entre iteraciones calculando para ello las medidas LLR hacia adelante en las iteraciones pares, y las medidas LLR hacia atrás en las iteraciones impares conforme a la ecuación**

$$L^{(i)}(u_k) = \begin{cases} L_f^{(i)} & \text{if } \text{mod}([i], 2) == 1 \\ L_b^{(i)} & \text{if } \text{mod}([i], 2) == 0 \end{cases}$$

Siendo  $[i]$  la función suelo]

Aunque el documento D01 centra la mayor parte de su exposición en el algoritmo MAP de decodificación, incluye también (párrafo 172) la posibilidad de extender su solución a otros algoritmos, entre ellos el SOVA, mencionado también en el mismo documento (párrafos 55-56). Por ello se considera que D01 incluye la aplicación de su solución al algoritmo SOVA.

La principal diferencia entre la invención reivindicada en 1 y el documento D01 del estado de la técnica más próximo es que la primera realiza una comparación, para los mismos bloques de bits ( $U_k$ ), de las medidas LLR a posteriori hacia delante y hacia atrás, mientras que D01 realiza tan sólo dichas medidas para cada bloque en uno u otro sentido, pero no en ambos (esto es, un segmento de bits en un sentido, y el siguiente en otro, pero no cada uno de ellos en ambos), utilizando los resultados de uno para los cálculos del siguiente (figura 9). El efecto técnico que tiene esta diferencia es que, en la invención reivindicada en 1, además de un ahorro en memoria y procesamiento, se consigue una reducción de la probabilidad de propagación de un error por sobreestimación de medidas LLR en un sub-bloque, ya que ahora dicha propagación no se produce en caso de haber una sobreestimación en el cálculo en uno de los dos sentidos, sino que es preciso que esto ocurra en ambos (mucho menos probable). Algo que no es posible en D01. El problema técnico objetivo a resolver sería entonces el de, a partir de un método de ventana deslizante como el de D01, conseguir no sólo una mejora en el consumo de memoria y procesamiento, sino en la vulnerabilidad a errores por sobreestimación de medidas LLR. La resolución de semejante problema lleva aparejado un cambio profundo de la técnica adoptada, que no sería evidente para un experto en la materia, quien tendría que recurrir a la actividad inventiva para ello. A este respecto, tampoco cualquiera de los documentos D02 a D06 aporta elementos que, combinados con los de D01 puedan llevar a un experto en la materia a derivar de manera evidente una solución como la reivindicada en 1.

En consecuencia, cabe concluir que, del examen de los documentos relevantes para la definición del estado de la técnica, la invención reivindicada en 1 es nueva y posee actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.

#### Reivindicaciones 2 a 5 y 16 a 19

Teniendo en cuenta la conclusión alcanzada para la reivindicación 1, y las correspondientes relaciones de dependencia, se concluye que las invenciones reivindicadas en 2 a 5 y 16 a 19 son nuevas y poseen actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.

#### Reivindicaciones 6 a 15

Las referencias en las reivindicaciones 6, 7 y 12, a las reivindicaciones 2, 3 y 5 respectivamente, implican que el sistema reivindicado en las primeras se configura para implementar los métodos reivindicados en las segundas. En este sentido, aunque el sistema mostrado en D01 contiene bancos de memoria, decodificadores comunes a ambos sentidos, unidad de control y unidades de cálculo extrínseco (párrafos 105-118, 171; figura 7), lo cierto es que estos no se hayan configurados para implementar métodos como los de 2, 3 o 5. El efecto técnico de esta diferencia, es, como en el caso de la reivindicación 1, el de conseguir una reducción de la probabilidad de propagación de un error por sobreestimación de medidas LLR en un sub-bloque, ya que ahora dicha propagación no se produce en caso de haber una sobreestimación en el cálculo en uno de los dos sentidos, sino que es preciso que esto ocurra en ambos (mucho menos probable). Y el problema técnico en cualquiera de las reivindicaciones, sería el de configurar las unidades de cálculo extrínseco y de decodificación, para organizar las decodificaciones hacia delante y hacia atrás, así como el procesado de las medidas LLR de ambas, con el fin de conseguir dicha mejora, más allá de las de memoria y procesamiento que aporta la solución en D01. Y, lo mismo que en el caso de 1, cabe concluir que semejante problema - que implica una concepción totalmente distinta del modo en que funcionan estas unidades en D01 - no puede ser resuelto por un experto en la materia si no es con el concurso de la actividad inventiva.

Por lo tanto se concluye que, del examen de los documentos relevantes para la definición del estado de la técnica, las invenciones reivindicadas en 6, 7 y 12 son nuevas y poseen actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.

Por otro lado, teniendo en cuenta la conclusión alcanzada para las reivindicaciones 7 y 12, y las correspondientes relaciones de dependencia, se concluye que las invenciones reivindicadas en 8 a 11 y 13 a 15 son nuevas y poseen actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.