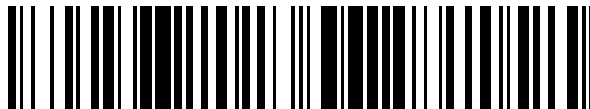


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 561 935**

21 Número de solicitud: 201400890

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 alto régimen binario, baja tasa de error y baja complejidad**

57 Resumen:

Sistemas y métodos para turbo decodificación iterativa de alto régimen binario, baja tasa de error y baja complejidad. La invención se refiere a un método para turbo decodificación en serie paralela iterativa de alto régimen binario, baja tasa de error y baja complejidad caracterizado porque 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 previo, dichos MLs intercambiables en la misma iteración. La invención refiere asimismo implementaciones de dicho método basadas en los algoritmos SOVA y BISOVA, así como una implementación basada en un nuevo algoritmo, ALSOVA, que constituya una modificación de SOVA, así como sistemas que implementan dichos métodos.

ES 2 561 935 B2

## DESCRIPCIÓN

**Sistemas y Métodos para turbo decodificación iterativa de alto régimen binario, 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. 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 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 "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 flechas 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 Transactions on 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 (Maximum A Posteriori) y SOVA (Soft Output Viterbi Algorithm). 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.

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 circuito integrado 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 aliviar esta degradación de prestaciones, algunas técnicas se han propuesto en la literatura, estando la mayoría de ellas centradas en el algoritmo MAP. Un solape entre sub-bloques es propuesto por Hsu y otros en "A parallel decoding scheme for turbo codes", publicado en el congreso de IEEE International Symposium on Circuits and Systems de 1998. Desafortunadamente esta técnica también aumenta la latencia de decodificación. Un método para la decodificación paralela basada en el algoritmo MAP que usa la información de borde entre sub-bloques fue propuesto por Yoon en "A parallel MAP algorithm for low latency turbo decoding", publicado en la revista IEEE Communications Letters, en 2002. Este método paralelo, conocido como SBI (Sub-block Boundary Information), consigue para el algoritmo MAP similares capacidades correctivas que la versión no paralela sin ningún impacto en la

latencia; cosa que no ocurre en las alternativas basadas en solape. No obstante al estar basado en el algoritmo MAP no ofrece las ventajosas características de bajo consumo de recursos que presenta el algoritmo SOVA.

5 Como con la decodificación paralela el consumo de área resulta un tema crítico el algoritmo SOVA se hace muy atractivo.

*Turbo decodificación clásica paralela con solape*

10 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  $K+T$  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 15 tiene una longitud de  $M$  bits.

20 Se define el lado izquierdo del sub-bloque  $n$ , identificado con el símbolo  $k_{l,n}$ , como la etapa en la que deberían inicializarse las métricas de camino o PMs (Path Metrics). El lado derecho, identificado con el símbolo  $k_{r,n}$ , se define como la etapa en la que el último recorrido hacia atrás o TB (Trace Back) debería ser iniciado. El índice de bloque  $n$  va de 1 a  $P$ . La inicialización de las PMs se realiza como sigue

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

25 donde  $\delta(\cdot)$  representa la función de Kronecker y  $PM^{(i,n)}(s, k_{l,n})$  es la PM para el sub-bloque  $n$ , estado  $s$ , iteración  $i$  y etapa  $k_{l,n}$ . La etapa en la que se realiza la inicialización de las PMs usando solape es  $k_{l,n}=0$  para  $n=1$  y  $k_{l,n}=(n-1)M-L/2$  para  $n>1$ . La ecuación anterior significa que para el primer sub-bloque ( $n=1$ ) el estado inicial es conocido, siendo el estado cero, mientras que para el resto de sub-bloques el estado inicial es desconocido con lo que se realiza un solape de  $L/2$  etapas para obtener PMs fiables.

30 El estado  $S_{kr,n}^{(i,n)}$  desde el que se inicia el último TB en la etapa  $k_{r,n}$  es el estado cero para el sub-bloque  $n=P$ , y cualquier estado para  $n<P$ , siendo  $k_{r,p}=K+T$  y  $k_{r,n}=nM+L/2$  para  $n<P$ . Las etapas de solape garantizan que los estados generados al hacer los recorridos hacia atrás converjan al camino ML (Maximum Likelihood).

35 En el caso del algoritmo SOVA, las PMs tendrán valores fiables cerca de los bordes de sub-bloque por el lado derecho siempre que la longitud de los sub-bloques sea mayor que la ventana de decodificación. Sin embargo, cuando los recorridos hacia atrás o TB empiezan desde el lado derecho, la incertidumbre en el estado desde el que se arranca el TB causa muchos errores en la secuencia decodificada en el lado derecho del sub-bloque. Como las PMs cercanas al lado derecho del sub-bloque son fiables, los valores a-posteriori  $L(u_k)$  tomarán valores grandes, indicando gran fiabilidad, para bits incorrectamente decodificados. Esta sobre-estimación provoca que aparezcan errores en los límites entre sub-bloques independientemente de la SNR 40 (Signal to Noise Ratio) que se tenga. En particular, las PMs por el lado derecho de los sub-bloques son fiables mientras que los estados usados para iniciar el proceso de TB, que se inicia

por la derecha de los sub-bloques, no lo son. Por la parte derecha de los sub-bloques la situación es la contraria.

## DESCRIPCIÓN DE LA INVENCION

5 La invención se refiere a una nueva arquitectura para la turbo decodificación en paralelo, identificada con el acrónimo PMSBx que viene del inglés Path Metric/State Border 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.

10 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  
15 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  
20 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  
25 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  
30 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.

40 *ALSOVA*

Una solución interesante al alto consumo de área del chip que tienen los decodificadores BISOVA es una novedosa modificación del algoritmo SOVA, denominada ALSOVA (ALternated direction SOVA). ALSOVA mejora considerablemente la capacidad de corrección

de errores del algoritmo SOVA, pero a diferencia del algoritmo BISOVA, no incrementa el consumo de área o la latencia. La combinación de ALSOVA y PMSBx permite alcanzar unas prestaciones en términos de BER cercanas a las que se consiguen con los decodificadores basados en el algoritmo MAP y a la vez mantienen el beneficio de la baja complejidad asociada al algoritmo SOVA.

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 delante 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 delante 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 delante 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 delante/atrás, o fb/bw del inglés forward/backward, el cual pueda realizar la decodificación en ambas direcciones en diferentes iteraciones.

- 5 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.
- 10

### **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.
- 15 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.
- 20 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.
- 25 La Fig. 10 ilustra la estructura de una posible realización de la invención, denominada PMSBx, usando la arquitectura en serie paralela adecuada para los algoritmos ALSOVA, SOVA y BISOVA.
- La Fig. 11 ilustra una estructura de la invención adecuada para los algoritmos SOVA, ALSOVA y BISOVA.
- 30 La Fig. 12 ilustra la estructura de una posible realización de la invención usando la arquitectura en serie paralela que es adecuada para el algoritmo 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 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.
- 35

Un aspecto interesante de la arquitectura propuesta para decodificación paralela, denominada PMSBx, es que puede ser usado con los algoritmos SOVA, ALSOVA y BISOVA. En el caso de BISOVA-PMSBx y ALSOVA-PMSBx existen dos tipos distintos de métricas de camino y de recorridos hacia atrás; unos para la decodificación hacia delante y otros hacia detrás. Para el caso de BISOVA este hecho no da lugar a ninguna modificación con respecto a SOVA-PMSBx ya que la inicialización se lleva a cabo en dos bancos independientes de decodificadores SOVA. Sin embargo, para ALSOVA-PMSBx no es posible intercambiar las PMs entre dos iteraciones consecutivas ya que éstas se han calculado en direcciones opuestas. Por tanto, en el caso de ALSOVA, el intercambio de PMs se realiza entre dos iteraciones en lugar de una como ocurre con SOVA y BISOVA. De manera más formal, la inicialización de PMs se puede describir 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  $k_{l,1}=0$  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 el estado desde el 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, BISOVA-PMSBx, y ALSOVA-PMSBx ya que los estados se intercambian en la misma iteración.

#### *ALSOVA-PMSBx 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. La arquitectura es la de un turbo decodificador serie paralelo. Este caso incluye a su vez el caso de un turbo decodificador en serie no-paralelo. El diagrama de bloques 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

5 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

10 métricas. En la figura, los buses que entregan P métricas LLR por ciclo se identifican con la letra P.

Cada uno de los decodificadores paralelos 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

15 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.

20 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. 8, permitiendo a dichas unidades 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

25 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 delante 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**. 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

30 trabajar tanto en la dirección hacia delante 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 delante 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

40 cada estado. Para ello, cada Elemento de Recursión **902** necesita conocer las medidas Ls, Lp y La. En el caso de una iteración completa, Lp es Lp1. En el caso de una semi-iteración, Lp es Lp2 y Ls ha sido entrelazado por el Banco de Memoria de Entrada **701**. Por otro lado, La 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 delante 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 delante o hacia atrás). Una posible manera de implementar este elemento es mediante un banco de multiplexores y registros para almacenar la PM previa.
- 10 Finalmente, el Banco Paralelo de Decodificadores SOVA fw/bw **704** tiene la arquitectura que aparece en la Fig. 10. En esta figura se pueden observar las conexiones entre decodificadores SOVA fw/bw que permiten el intercambio de información que alivia la incertidumbre entre sub-bloques. El intercambio de PMs se realiza gracias al elemento **1003** que aplica un retardo de dos iteraciones completas ( $D=2$ ), es decir cuatro semi-iteraciones, a las PMs. El intercambio de estados se realiza gracias a los registros que los almacenan durante la misma iteración en la que son usados. Estos registros aparecen como elemento **1002** en la Fig. 10. El intercambio de información (PMs y estados) y retraso de la misma es controlado por el elemento **1004** que genera las señales de control adecuadas para el resto de unidades de la Fig. 10. Además los decodificadores, que aparecen en la figura como unidades **1001**, son decodificadores SOVA fw/bw que siguen la arquitectura de la unidad **703** que aparece en la Fig. 8. Estos decodificadores son capaces de hacer la decodificación SOVA hacia delante y hacia atrás dependiendo de la iteración de la que se trate.

#### *SOVA-PMSBx para turbo decodificación en serie paralela*

- La Fig. 11 ilustra la estructura de la turbo decodificación en serie paralela, la cual se puede aplicar al caso de SOVA-PMSBx. La arquitectura es la de un turbo decodificador serie paralelo. Este caso incluye a su vez el caso de un turbo decodificador en serie no-paralelo. El diagrama de bloques 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 **1101**. 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 **1101** entrelaza las métricas  $L_s$  cuando lleva a cabo una semi-iteración, pudiendo así entregar dichas métricas al Banco de Decodificadores SOVA **1104**.

- El Banco de Decodificadores SOVA **1104** calcula métricas a-posteriori (sólo hacia delante). La Unidad de Control **1102** 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 **1106** calcula la información extrínseca utilizando las métricas LLR 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. La información extrínseca generada por el elemento **1106** se almacena en el elemento **1003**. Este último elemento se encarga de entregar la información extrínseca que se calculó en la semi-iteración anterior y de entrelazarla o desentrelazarla según se trate de una semi-iteración o de una iteración completa. En la figura, los buses que entregan  $P$  métricas LLR por ciclo se

identifican con la letra P. La estructura del Banco Paralelo de Decodificadores SOVA **1104** se presenta en la Fig. 10. En este caso se usa el parámetro  $D=1$ . Esto implica que las PMs son retrasadas por las unidades **1003** una iteración completa, es decir dos semi-iteraciones. En este caso cada elemento **1001** es un decodificador SOVA convencional que se ejecuta sólo en la dirección hacia delante. El elemento **1002** es un registro que almacena el estado desde el que se arranca el último TB en la iteración actual. La Unidad de Control **1004** garantiza que el intercambio de estados y PMs se realice acorde con el método SOVA-PMSBx.

*BISOVA-PMSBx para turbo decodificación en serie paralela*

La arquitectura de un turbo decodificador en serie paralelo usando el algoritmo BISOVA-PMSBx se ilustra en la Fig. 12. Esta arquitectura sigue la implementación directa del algoritmo BISOVA presentada por Chen en el año 2000. En el algoritmo BISOVA en la iteración  $i$ , un banco de decodificadores SOVA calcula las medidas a-posteriori hacia delante, identificadas por el símbolo  $L_f^{(i)}(u_k)$  mientras que el otro calcula las medidas a-posteriori hacia atrás, identificadas como  $L_b^{(i)}(u_k)$ .

$$L^{(i)}(u_k) = u_k \cdot \min(|L_f^{(i)}(u_k)|, |L_b^{(i)}(u_k)|) \quad (5)$$

Este método mejora significativamente la capacidad correctiva pero dobla el área ocupada por los bancos de decodificadores.

El Banco de Memoria de Entrada **1201** debe proporcionar las medidas  $L_s$  y  $L_p$  en la dirección hacia delante y hacia atrás simultáneamente. Por este motivo hay cuatro buses a la salida de esta unidad. La unidad **1205** está formada por dos bancos paralelos de decodificadores SISO, que aparecen en la figura como elementos **1203** y **1204**. La unidad **1203** se corresponde con un banco de decodificadores que trabaja en la dirección hacia delante. Esta unidad tiene la estructura que aparece en la Fig. 10 donde el retardo aplicado a las PMs es de una iteración completa ( $D=1$ ). Por otro lado la unidad **1204** es un banco de decodificadores SOVA paralelos que trabaja en la dirección hacia atrás. Esta unidad también tiene la estructura que se presenta en la Fig. 10 donde el retardo aplicado a las PMs es de una iteración ( $D=1$ ). Las medidas LLR a-posteriori calculadas en las direcciones hacia delante y hacia atrás se almacenan en la unidad **1206**. Esta unidad tiene dos bloques de memoria con dos puertos. Para alimentar cada banco de decodificadores al mismo tiempo, la unidad **1206** lee las correspondientes medidas LLR a-posteriori hacia delante y hacia atrás y calcula las medidas LLR a-posteriori finales siguiendo la ecuación (5). Posteriormente calcula la información extrínseca que será usada como información a-priori por ambos bancos de decodificadores en la siguiente semi-iteración. La unidad de control **1202** asegura que el funcionamiento del resto de unidades sea el correcto.

## REIVINDICACIONES

1. Método para turbo decodificación en serie paralela iterativa de alto régimen binario, baja tasa de error y baja complejidad, y basado en el algoritmo SOVA, caracterizado por que aplica en decodificación paralela y 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 previo, dichos estados de camino (ML) intercambiables en la misma iteración.
2. Método según la reivindicación anterior caracterizado por que además comprende:
- 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;
  - El entrelazado de las métricas LLR almacenadas cuando se lleva a cabo una semi-iteración y antes de la entrega de los sub-bloques correspondientes;
  - El cálculo de métricas a-posteriori por dichos dos o más decodificadores paralelos;
  - Y el cálculo por una unidad de cálculo extrínseca de la información extrínseca a partir de las métricas LLR y las métricas a-posteriori.
3. Método según la reivindicación anterior, y basado en el algoritmo ALSOVA, caracterizado por que comprende 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 delante en las iteraciones pares, y las medidas LLR hacia atrás en las iteraciones impares conforme a la ecuación (2)

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

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

4. Método según la reivindicación 2, y basado en el algoritmo BISOVA, caracterizado por que el cálculo de métricas a-posteriori comprende el cálculo de métricas a-posteriori hacia delante (por parte de decodificadores paralelos que trabajan hacia delante o fw) y el cálculo de métricas a-posteriori hacia detrás (por parte de decodificadores paralelos que trabajan hacia detrás o bw).
5. Sistema para turbo decodificación iterativa de alto régimen binario, baja tasa de error y baja complejidad que implementa un método conforme a la reivindicación 3, basado en el algoritmo SOVA; caracterizado por que comprende un Banco de Memoria de Entrada **1101**, una Unidad de Control **1102**, un Banco Paralelo de Decodificadores SOVA **1104**, un Banco de Memoria de Entrada **1105** y una Unidad de Cálculo Extrínseco **1106**; de forma que

- 5
- a. las métricas LLR se almacenan en el Banco de Memoria de Entrada **1101**, que entrelaza dichas métricas cuando lleva a cabo una semi-iteración, siendo entonces dicho Banco de Memoria de Entrada **1101** capaz de entregar el sub-bloque correspondiente a cada decodificador paralelo dentro del Banco de Decodificadores SOVA **1104**;
- b. el Banco de Decodificadores SOVA **1104** calcula métricas a-posteriori (sólo hacia delante);
- 10 c. la Unidad de Control **1102** determina el comportamiento de las diferentes unidades dependiendo de la iteración o semi-iteración que se lleve a cabo en cada momento;
- d. la Unidad de Cálculo Extrínseca **1106** calcula la información extrínseca utilizando las métricas LLR y las métricas a-posteriori;
- e. y un Banco de Memoria de Entrada **1105** en el que se almacena la información extrínseca calculada por la Unidad de Cálculo Extrínseca **1106**.
- 15
6. Sistema según la reivindicación anterior caracterizado por que el Banco Paralelo de Decodificadores SOVA **1104** comprende:
- a. Uno o más elementos 1003 que aplican un retardo de una iteración completa (dos semi-iteraciones) a las PMs para facilitar su intercambio entre los decodificadores 1001;
- 20 b. Dos o más decodificadores SOVA convencionales 1001 que se ejecutan sólo en la dirección hacia delante;
- c. Uno o más registros 1002 que almacenan el estado desde el que se arranca el último TB en la iteración actual;
- 25 d. y una Unidad de Control 1004 que garantiza el intercambio de estados y PMs conforme con el método objeto de la reivindicación 1.
7. Sistema para turbo decodificación iterativa de alto régimen binario, baja tasa de error y baja complejidad que implementa un método conforme a la reivindicación 5, basado en el algoritmo BISOVA; caracterizado por que comprende un Banco de Memoria de Entrada **1201**, una Unidad de Control **1202**, una unidad **1205** que comprende un Banco Paralelo de Decodificadores SOVA fw **1203** y un Banco de Decodificadores Paralelo SOVA bw **1204**, y una Unidad de Cálculo Extrínseco **1206**; de forma que
- 30 a. el Banco de Memoria de Entrada **1201** proporciona a la unidad **1205** las métricas LLR que se almacenan en él simultáneamente hacia delante y hacia atrás a través de buses de salida;
- b. el Banco Paralelo de Decodificadores SOVA fw **1203** trabaja hacia delante y el Banco Paralelo de Decodificadores SOVA bw **1204** trabaja hacia atrás, calculando ambas métricas a-posteriori;
- 40 c. la Unidad de Control **1202** 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 hacia delante, identificadas por el símbolo  $L_f^{(i)}(u_k)$ , y las métricas a-posteriori hacia atrás, identificadas como  $L_b^{(i)}(u_k)$ , conforme a la ecuación (5)
- 45

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

8. Sistema según la reivindicación anterior caracterizado por que
- a. el Banco Paralelo de Decodificadores SOVA fw **1203** comprende:
    - 5 i. Dos o más decodificadores SOVA convencionales que se ejecutan sólo en la dirección hacia delante;
    - ii. uno o más registros que almacenan el estado desde el que se arranca el último TB en la iteración actual;
    - 10 iii. y una Unidad de Control que garantiza el intercambio de estados y PMs conforme con el método objeto de la reivindicación 1;
  - b. y el Banco Paralelo de Decodificadores SOVA bw **1204** comprende:
    - 15 iv. Dos o más decodificadores SOVA convencionales que se ejecutan sólo en la dirección hacia atrás;
    - v. uno o más registros que almacenas el estado desde el que se arranca el último TB en la iteración actual;
    - vi. y una Unidad de Control que garantiza el intercambio de estados y PMs conforme con el método objeto de la reivindicación 1;
9. Sistema para turbo decodificación iterativa de alto régimen binario, 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
- 25 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 fw/bw **704**;
  - 30 b. cada decodificador SOVA fw/bw **703** comprendido en el Banco Paralelo de Decodificadores SOVA fw/bw **704** calcula métricas a-posterior hacia delante (fw) en las iteraciones impares y hacia atrás (bw) en las iteraciones pares;
  - 35 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 **706** calcula la información extrínseca utilizando las métricas LLR y las métricas a-posteriori.
10. Sistema según la reivindicación anterior caracterizado por que el Banco Paralelo de Decodificadores SOVA fw/bw **704** comprende
- 40 a. Uno o más elementos 1003 que aplican un retardo de dos iteraciones completas (cuatro semi-iteraciones) a las PMs para facilitar su intercambio entre los decodificadores 703;
  - b. uno o más registros 1002 que almacenan el estado desde el que se arranca el último TB en la iteración actual;;
  - 45 c. y una Unidad de Control 1004 que garantiza el intercambio de estados y PMs conforme con el método objeto de la reivindicación 1;

11. Sistema según la reivindicación anterior caracterizado por que cada decodificador paralelo SOVA fw/bw **703** comprende:
- 5 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;
  - 10 b. Un retardador **802** que aplica a las PMs un retardo en iteraciones;
  - 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;
  - 15 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;
  - 20 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.
  - 25
12. Sistema según la reivindicación anterior caracterizado por que la Unidad Recursiva **801**
- 30 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;
    - 35 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.
  - 40 13. Programa informático adaptado para la realización de un método conforme a cualquiera de las reivindicaciones 1 a 4, o que comprende instrucciones para llevar a cabo las etapas comprendidas en un método conforme a cualquiera de las reivindicaciones 1 a 4.
  - 45 14. 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 4.

15. Medio portador de grabación con un programa informático conforme a la reivindicación 13 grabado en dicho en dicho medio portador de grabación.
  16. Onda portadora de señal portando señales que incorporan un programa informático conforme a la reivindicación 13.
- 5



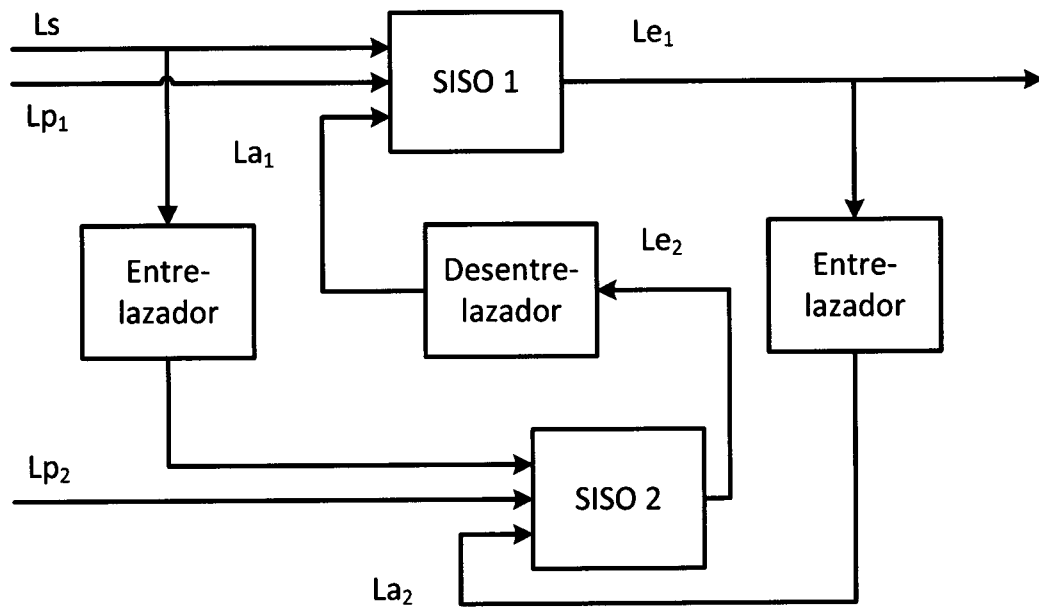


Fig. 1

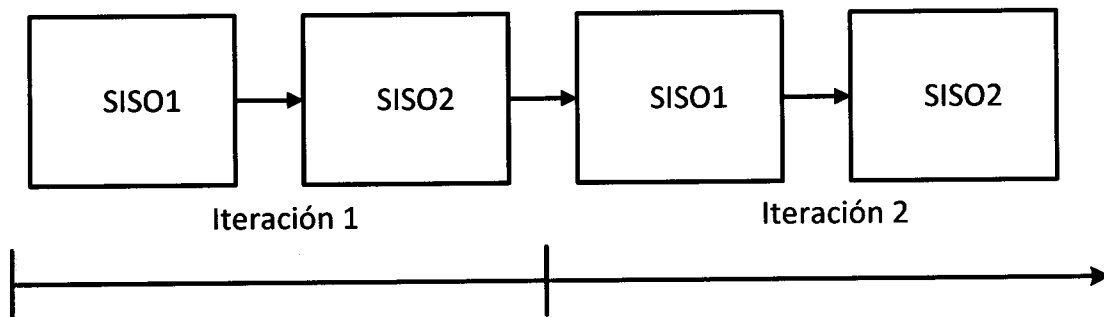


Fig. 2

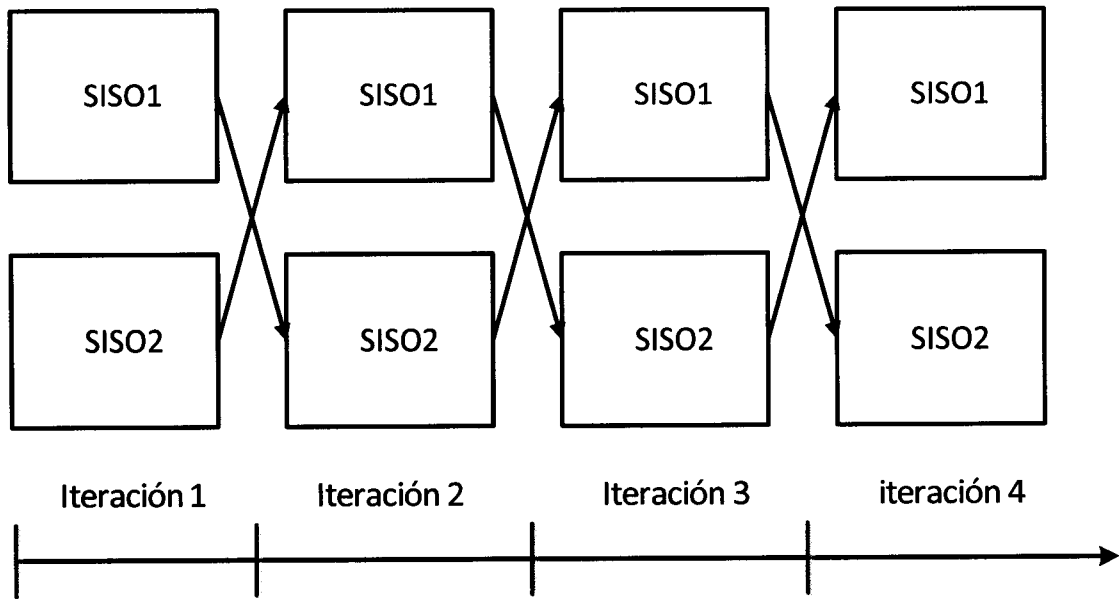


Fig. 3

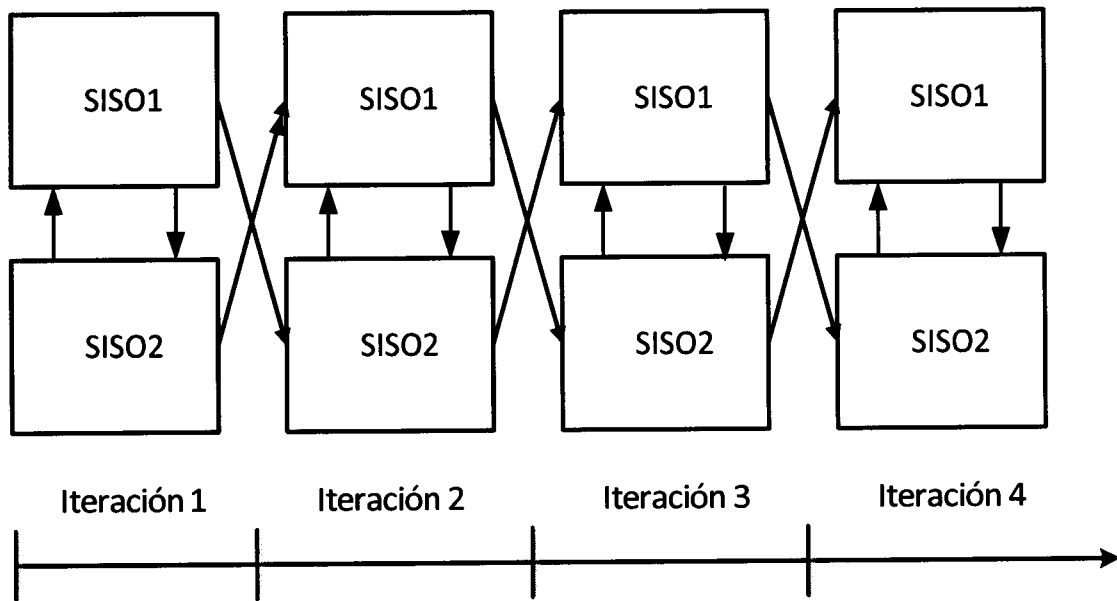


Fig. 4

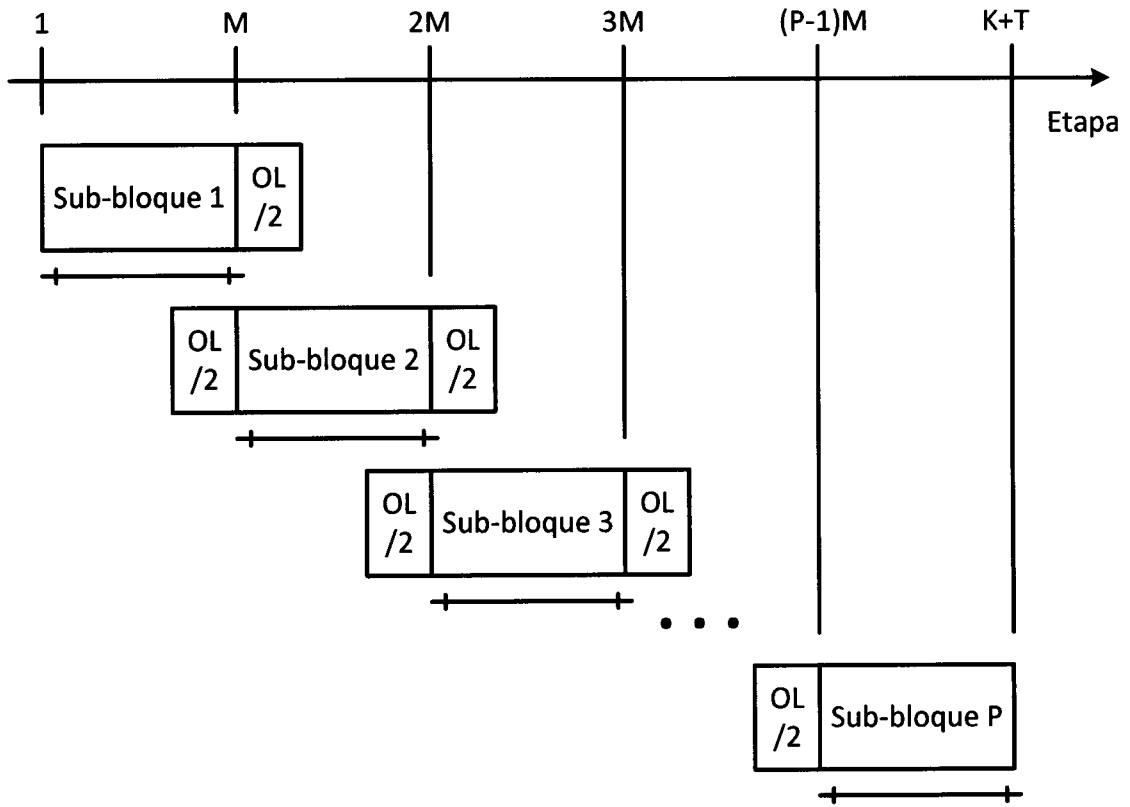


Fig. 5

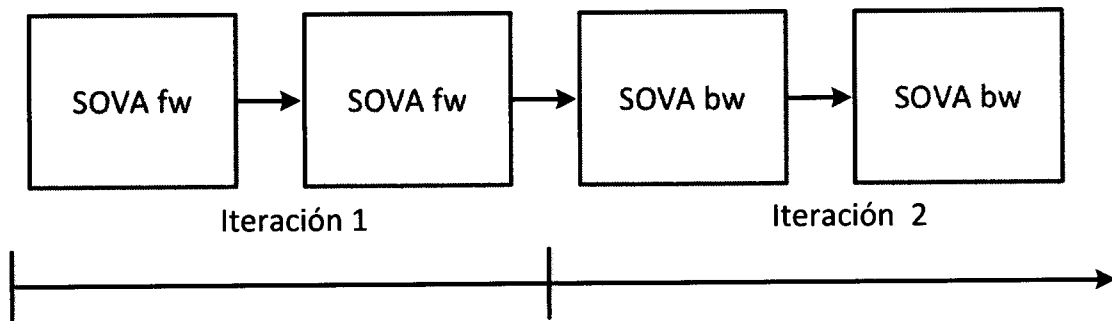


Fig. 6

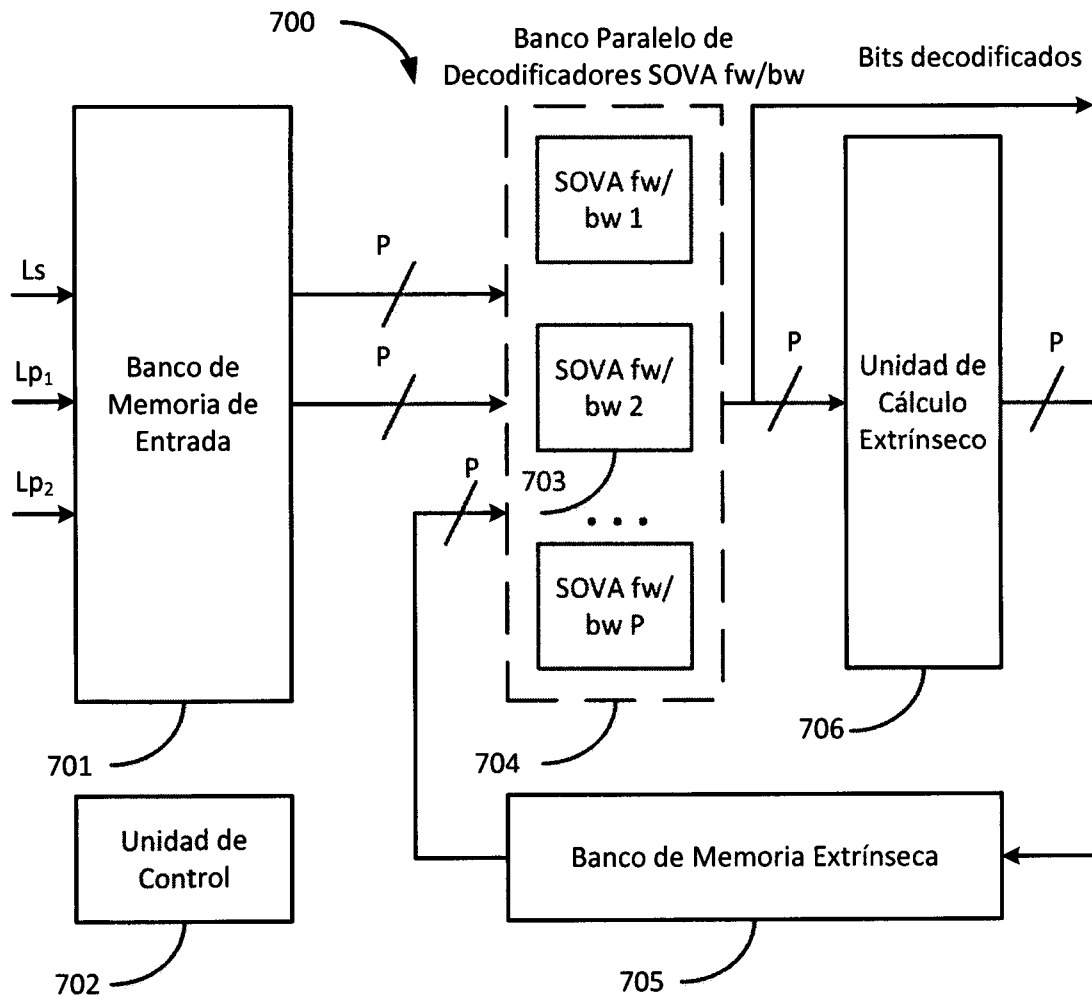


Fig. 7

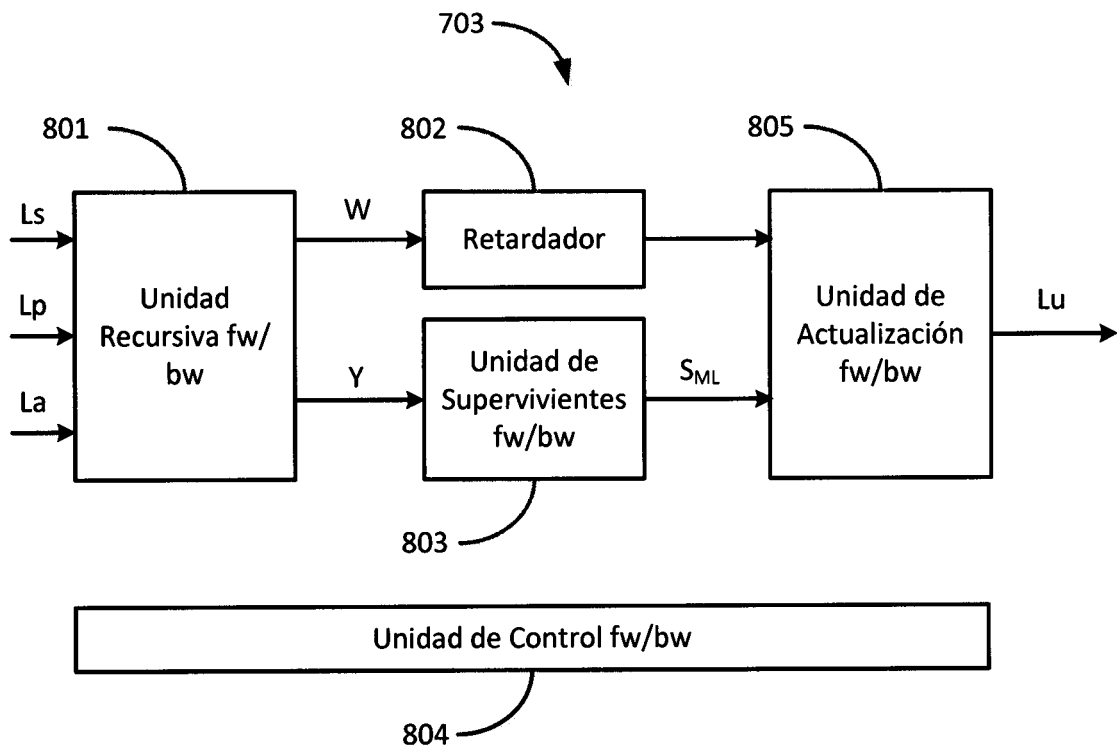


Fig. 8

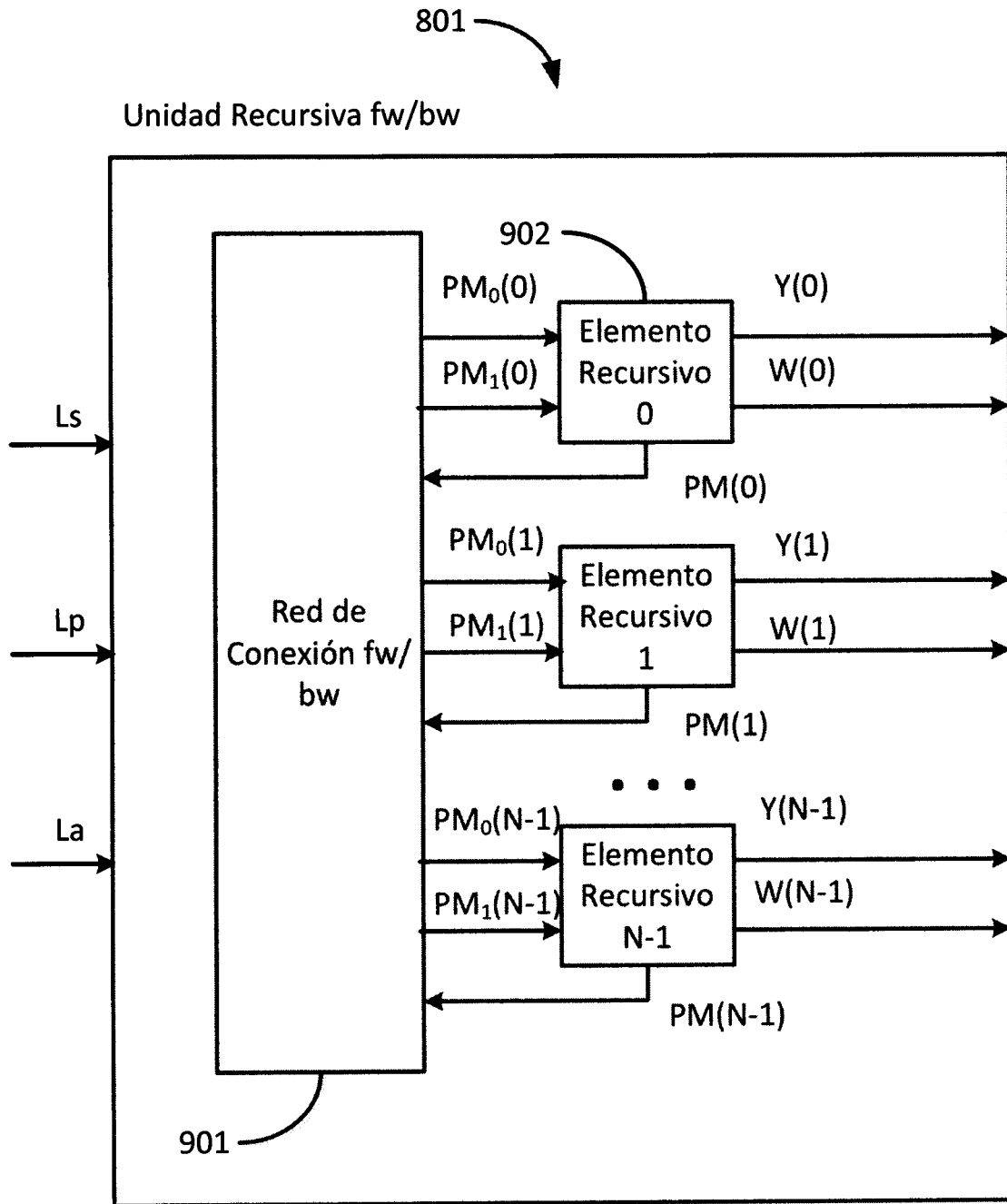


Fig. 9

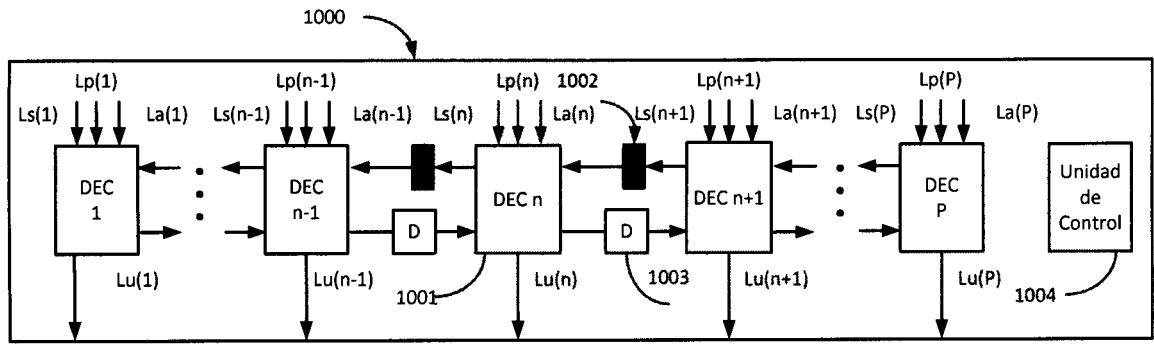


Fig. 10

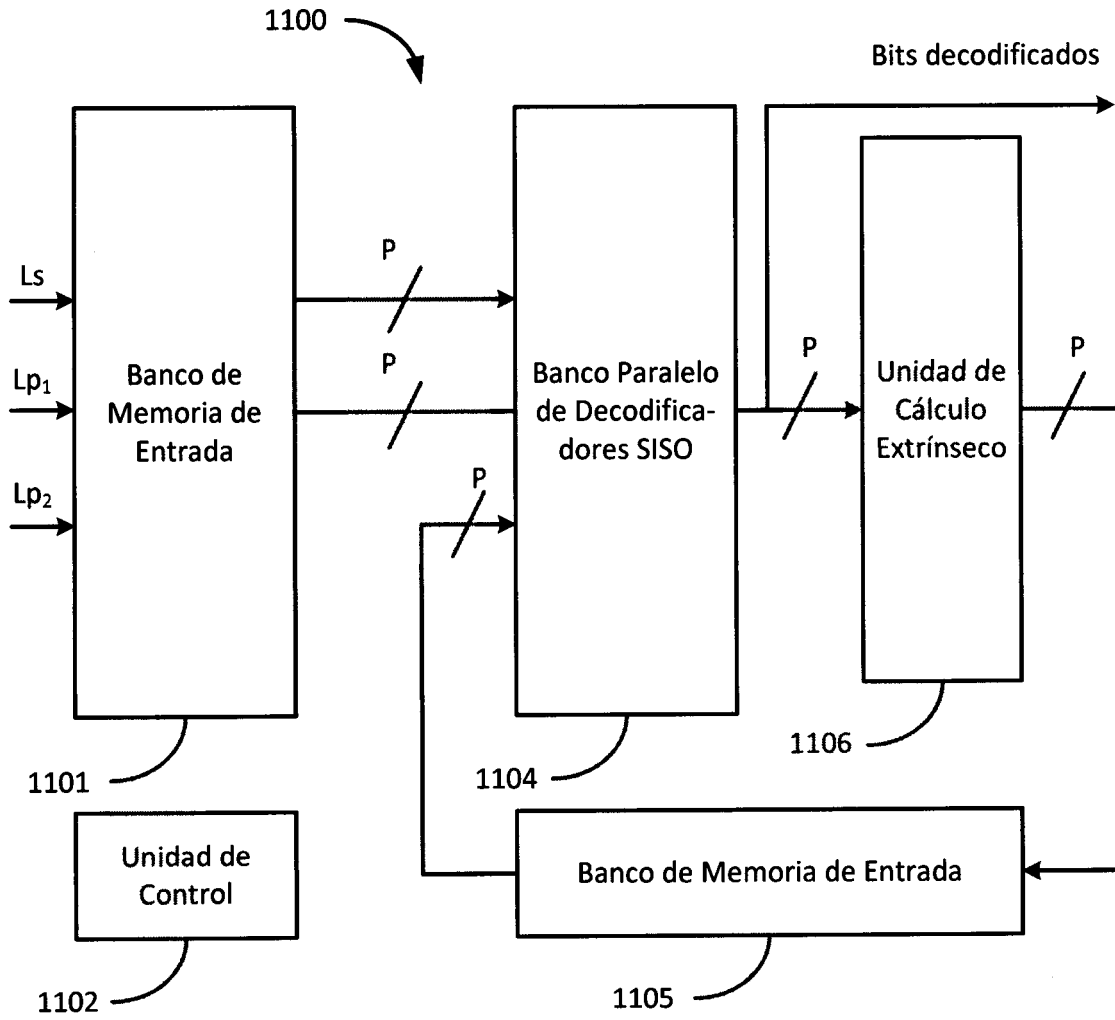


Fig. 11

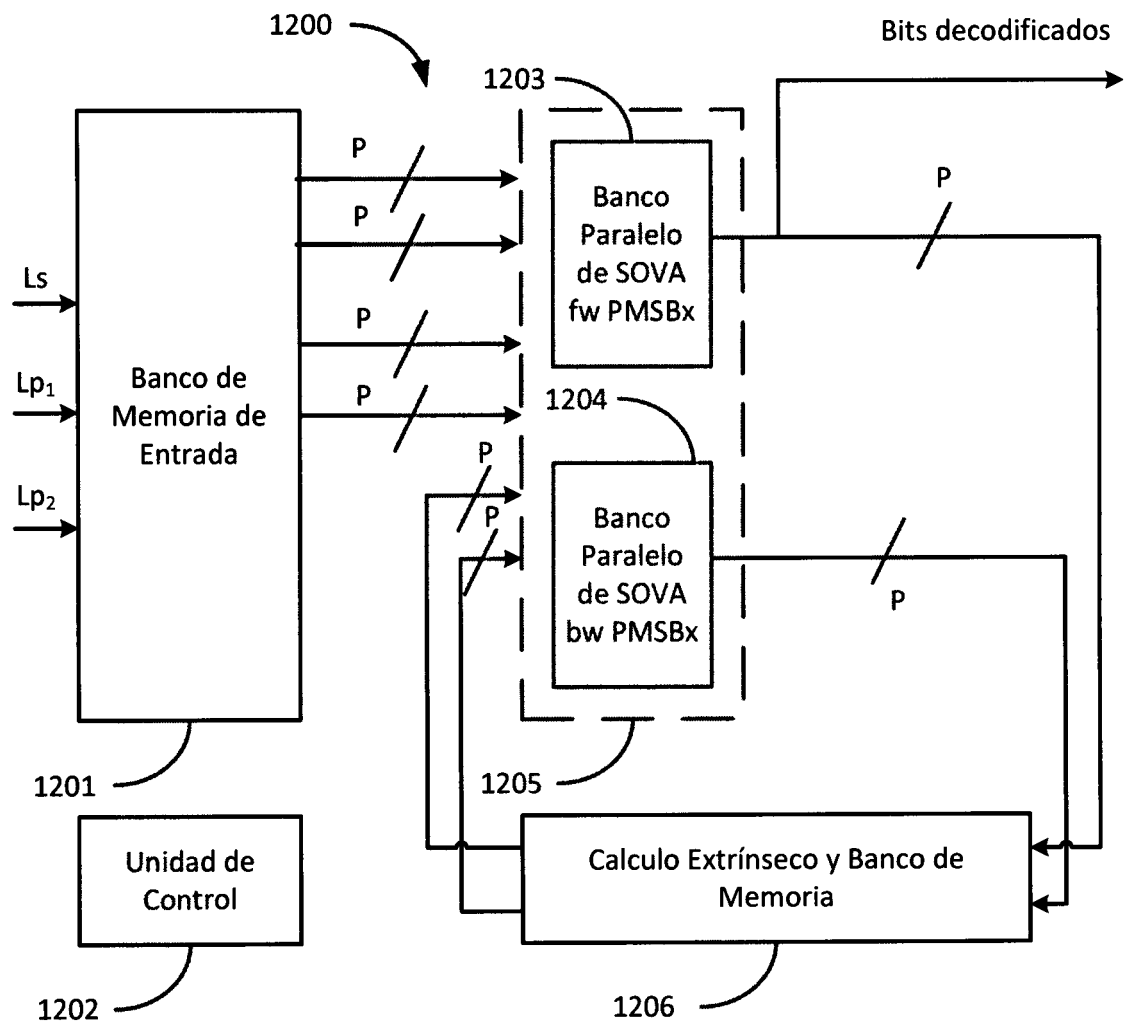


Fig. 12





- ②1 N.º solicitud: 201400890  
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 1391995 A2 (NEC ELECTRONICS CORP) 25.02.2004, párrafos [1],[3-5],[12-14],[18-19],[24],[37-39],[44],[46-51],[58].	1-16
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.	1-16
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.	1-16
A	CN 101373978 A (HUAWEI TECH CO LTD) 25.02.2009, resumen; figura 9.	1-16
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 >	1-16
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 >	1-16

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 1-16	<b>SI</b>
	Reivindicaciones	<b>NO</b>
<b>Actividad inventiva (Art. 8.1 LP11/1986)</b>	Reivindicaciones 1-16	<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 1391995 A2 (NEC ELECTRONICS CORP)	25.02.2004
D02	EP 1398881 A1 (ST MICROELECTRONICS NV et al.)	17.03.2004
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. El 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 del estado más próximo de la técnica es D01, que presenta varias mejoras en la decodificación turbo, entre ellas una relacionada con el uso de ventanas, donde la métrica de la siguiente se basa en la de la anterior, ya sea en una decodificación hacia atrás o hacia adelante.

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 en serie paralela iterativa de alto régimen binario, baja tasa de error y baja complejidad, y basado en el algoritmo SOVA (Párrafos 4, 12, 51) caracterizado por que aplica en decodificación paralela y porque 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 (párrafo 50); **[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 previo, dichos estados de camino (ML) intercambiables en la misma iteración.]**

Un experto en la materia aplicaría la solución expuesta en D01 para el aprovechamiento de métricas entre bordes de sub-bloques en el caso de un algoritmo BISOVA, sin necesidad de actividad inventiva, puesto que ello implicaría básicamente emplear en los módulos 21 y 22 bancos paralelos de decodificadores, una técnica muy conocida, como ilustra por ejemplo del documento D06.

La principal diferencia entre la invención reivindicada en 1 y el documento D01 es que este último no considera la posibilidad del intercambio de estados de camino y de su uso para el recorrido hacia atrás. El efecto técnico que tiene esta diferencia es el de reducir la sobreestimación de valores LLR a posteriori, merced a realizarse el recorrido hacia atrás a partir de estados fiables. Y el problema técnico a resolver sería el de mejorar el método de D01 para conseguir también un incremento de la fiabilidad del cálculo del camino hacia atrás. La resolución de este problema no sería evidente para un experto en la materia, a partir de D01, ni tampoco a partir de su combinación con cualquiera de los otros documentos (D02-D06), toda vez que en ninguno de ellos se contempla el papel de los estados ML, sino solamente de las métricas de camino.

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 4 y 13 a 16

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 13 a 16 son nuevas y poseen actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.

Reivindicaciones 5 a 12

Las referencias en las reivindicaciones 5, 7 y 9, a las reivindicaciones 3, 4 y 3 respectivamente, implican que el sistema reivindicado en las primeras se configura para implementar los métodos reivindicados en las segundas (se entiende que la referencia a la reivindicación 5 en la reivindicación 7 es en realidad a la 4, puesto que se precisa que dicha referencia es con respecto a un método, objeto que no es el de 5, pero sí el de 4). En este sentido, en D01 se encuentran o pueden derivarse sin necesidad de actividad inventiva por un experto en la materia, los componentes básicos mencionados en 17 a 19: banco de memoria de entrada (24), bancos de decodificadores bw/fw (21, 22), unidad de cálculo extrínseco (23), banco de memoria extrínseca (28) y unidad de control (párrafo 58). Sin embargo no es menos cierto que estos no estarían configurados para implementar la reutilización de estados del margen izquierdo para el camino hacia atrás desde el borde derecho adyacente. El efecto técnico de esta diferencia, es, como en el caso de la reivindicación 1, el de reducir la sobreestimación de valores LLR a posteriori, al realizarse el recorrido hacia atrás a partir de estados fiables. Y el problema técnico objetivo en cualquiera de las reivindicaciones, sería el de configurar el sistema presentado en D01 para conseguir también un incremento de la fiabilidad del cálculo del camino hacia atrás. Y, lo mismo que en el caso de 13, cabe concluir que semejante problema no puede ser resuelto por un experto en la materia si no es con el concurso de la actividad inventiva, puesto que implica un funcionamiento no evidente para un experto, ya sea a partir del descrito en D01, como de las funciones habituales y conocidas de sus componentes básicos.

Por lo tanto se concluye que, del examen de los documentos relevantes para la definición del estado de la técnica, las invenciones reivindicadas en 5, 7 y 9 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 5, 7 y 9, y las correspondientes relaciones de dependencia, se concluye que las invenciones reivindicadas en 6, 8 y 10 a 12 son nuevas y poseen actividad inventiva conforme ambos requisitos se definen respectivamente en los artículos 6 y 8 de la Ley de Patentes.