

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 782 830**

51 Int. Cl.:

H03M 7/30 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.09.2014 PCT/US2014/054993**

87 Fecha y número de publicación internacional: **19.03.2015 WO15038648**

96 Fecha de presentación y número de la solicitud europea: **10.09.2014 E 14776776 (8)**

97 Fecha y número de publicación de la concesión europea: **01.01.2020 EP 3044881**

54 Título: **Procedimiento para la detección comprimida de datos de transmisión continua y el aparato para realizar el mismo**

30 Prioridad:

12.09.2013 US 201361877162 P
09.09.2014 US 201414481581

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

16.09.2020

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

FRERIS, NIKOLAOS;
OCAL, ORHAN y
VETTERLI, MARTIN

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 782 830 T3

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

DESCRIPCIÓN

Procedimiento para la detección comprimida de datos de transmisión continua y el aparato para realizar el mismo

5 **CAMPO DE LA INVENCION**

10 **[0001]** La presente invención está dirigida a un procedimiento para la detección comprimida de datos de transmisión continua y a los medios para realizar el mismo. Más específicamente, algunos modos de realización incluyen un procedimiento de detección comprimida de datos de transmisión continua que emplea un algoritmo recursivo para realizar la detección comprimida en datos de transmisión continua, y un aparato o sistema o producto de programa informático capaz de realizar el procedimiento para la detección comprimida de datos de transmisión continua.

15 **ANTECEDENTES DE LA INVENCION**

20 **[0002]** En el campo del procesamiento de señales, las señales de interés pueden representarse de forma dispersa usando pocos coeficientes en una base ortonormal seleccionada adecuadamente. De forma ejemplar, la base de Fourier se usa para señales de banda ilimitada o bases de onícula para señales continuas con función definida por partes, tal como imágenes. Si bien un pequeño número de coeficientes en las bases respectivas son suficientes para representar dichas señales, el teorema de muestreo de Nyquist/Shannon sugiere una tasa de muestreo que es al menos el doble del ancho de banda de la señal. Dicha tasa de muestreo se conoce en la técnica como la tasa de Nyquist. En muchos casos, la tasa de muestreo indicada es mucho más alta que la cantidad suficiente de coeficientes.

25 **[0003]** Recientemente, se introdujo el marco de detección comprimida (CS) con el objetivo de muestrear las señales no correspondientes a su ancho de banda, sino más bien de acuerdo con su contenido informativo, o sea, el número de grados de libertad de la señal. Este paradigma de muestreo sugiere una tasa de muestreo más baja en comparación con la teoría de muestreo clásica para señales que tienen una representación dispersa en algunas bases dadas. Las señales típicas que surgen naturalmente en astronomía y en la formación de imágenes biomédicas se ajustan a este modelo. El documento WO2012/094804 divulga un procedimiento de detección de compresión que aprovecha la naturaleza digital de las señales.

30 **SUMARIO DE LA INVENCION**

35 **[0004]** La presente invención se define por el conjunto de reivindicaciones.

BREVE DESCRIPCION DE LOS DIBUJOS

40 **[0005]** Otros objetos y ventajas de la invención se apreciarán a partir de la siguiente descripción, tomada junto con los dibujos incluidos, de los cuales:

La fig. 1 es un diagrama de bloques de un procedimiento de detección de datos de transmisión continua de acuerdo con un modo de realización de la presente invención;

45 la fig. 2 representa el tiempo de procesamiento promedio para la detección comprimida recursiva frente a un "enfoque ingenuo" en una sola ventana de tiempo;

la fig. 3 representa los resultados de la estimación del conjunto de soporte usando LASSO;

50 la fig. 4 representa gráficos de error para: a) estimaciones promediadas, b) estimaciones desesgadas y promediadas y c) estimaciones obtenidas mediante votación y promedio, y

la fig. 5 representa gráficos de error para la estimación promedio de LASSO y 'votación y cálculo de promedio' en datos de transmisión continua.

55 **DESCRIPCION DETALLADA DE LA INVENCION**

60 **[0006]** La siguiente descripción del mejor modo contemplado actualmente de practicar la invención no debe tomarse en un sentido limitante, sino que se hace simplemente con el propósito de describir los principios generales de la invención. El alcance de la invención debe determinarse con referencia a las reivindicaciones.

65 **[0007]** Aunque se ha avanzado mucho en el campo de la detección comprimida, todavía no se ha abordado un algoritmo recursivo para realizar la detección comprimida de datos de transmisión continua. La complejidad computacional y la estabilidad de la estimación de señal a partir de muestras ruidosas mediante la aplicación de detección comprimida en una secuencia de entrada a través del establecimiento de ventanas sucesivas aún no se han abordado en la técnica.

[0008] Como apreciará un experto en la técnica, los aspectos de la presente invención pueden realizarse como un sistema, procedimiento o producto de programa informático. En consecuencia, los aspectos de la presente invención pueden adoptar la forma de un modo de realización enteramente de hardware, un modo de realización de un programa informático (incluyendo firmware, programas informáticos residentes, microcódigo, etc.) o un modo de realización que combina aspectos de programas informáticos y hardware, los cuales en general se pueden denominar en el presente documento como "circuito", "unidad" o "sistema". Además, los aspectos de la presente invención pueden adoptar la forma de un producto de programa informático realizado en uno o más medios/soportes legibles por ordenador (por ejemplo, medios/soportes de almacenamiento de datos) que tienen un código de programa legible por ordenador registrado en los mismos.

[0009] Se puede usar cualquier combinación de uno o más medios/soportes legibles por ordenador. El medio legible por ordenador puede ser un medio de señal legible por ordenador o un medio de almacenamiento legible por ordenador. Un medio de almacenamiento legible por ordenador puede ser, por ejemplo, pero sin estar limitado a, un sistema, aparato o dispositivo electrónico, magnético, óptico, electromagnético, infrarrojo o semiconductor o cualquier combinación adecuada de los anteriores. Ejemplos más específicos (una lista no exhaustiva) del medio de almacenamiento legible por ordenador incluirían los siguientes: un disquete de ordenador portátil, un disco duro, una memoria de acceso aleatorio (RAM), una memoria de solo lectura (ROM), una memoria de solo lectura programable y borrable (EPROM o memoria *flash*), una fibra óptica, una memoria de solo lectura de disco compacto portátil (CD-ROM), un dispositivo de almacenamiento óptico, un dispositivo de almacenamiento magnético o cualquier combinación adecuada de los anteriores. En el contexto de este documento, un medio de almacenamiento legible por ordenador, por ejemplo, un medio de almacenamiento de datos, puede ser cualquier medio tangible que pueda contener o almacenar un programa para su uso por o en relación con un sistema, aparato o dispositivo de ejecución de instrucciones.

[0010] Un medio de señal legible por ordenador puede incluir una señal de datos propagada con código de programa legible por ordenador incorporado, por ejemplo, en banda base o como parte de una onda portadora. Dicha señal propagada puede tomar cualquiera de una variedad de formas, que incluyen, pero no se limitan a, electromagnética, óptica o cualquier combinación adecuada de las mismas. Un medio de señal legible por ordenador puede ser cualquier medio legible por ordenador que no sea un medio de almacenamiento legible por ordenador y que pueda comunicar, propagar o transportar un programa para su uso por o en relación con un sistema, aparato o dispositivo de ejecución de instrucciones.

[0011] El código del programa incorporado en un medio legible por ordenador puede transmitirse usando cualquier medio apropiado, incluyendo pero sin limitarse a, inalámbrico, por cable, cable de fibra óptica, RF, etc., o cualquier combinación adecuada de los anteriores.

[0012] El código de programa informático para llevar a cabo las operaciones de los aspectos de la presente invención puede escribirse en cualquier combinación de uno o más lenguajes de programación, incluyendo un lenguaje de programación orientado a objetos, tal como Java, Smalltalk, C++ o similares, y lenguajes de programación de procedimientos convencionales, tales como el lenguaje de programación "C" o lenguajes de programación similares. El código de programa puede ejecutarse completamente en el ordenador del usuario, parcialmente en el ordenador del usuario, como un paquete de software autónomo, parcialmente en el ordenador del usuario y parcialmente en un ordenador remoto o completamente en el ordenador o el servidor remoto. En el último caso, el ordenador remoto puede estar conectado al ordenador del usuario a través de cualquier tipo de red, incluyendo una red de área local (LAN) o una red de área amplia (WAN), o la conexión puede realizarse con un ordenador externo (por ejemplo, a través de Internet usando un proveedor de servicios de Internet).

[0013] Aspectos de la presente invención se describen a continuación con referencia a ilustraciones de diagramas de flujo y/o diagramas de bloques de procedimientos, aparatos (sistemas) y productos de programas informáticos de acuerdo con los modos de realización de la presente invención. Ha de entenderse que cada bloque de las ilustraciones de diagramas de flujo y/o diagramas de bloques, y las combinaciones de bloques en las ilustraciones de diagramas de flujo y/o diagramas de bloques, pueden implementarse mediante instrucciones de programa informático. Estas instrucciones de programa informático se pueden proporcionar a un procesador de ordenador de uso general, un ordenador de uso especial u otro aparato de procesamiento de datos programable para producir una máquina, el cual ejecutado por medio del procesador del ordenador u otro aparato de procesamiento de datos programable, crea medios para implementar las funciones/actos especificados en el diagrama de flujo y/o el bloque o bloques del diagrama de bloques.

[0014] Estas instrucciones de programa informático también pueden almacenarse en un medio legible por ordenador que puede dirigir un ordenador, otro aparato de procesamiento de datos programable u otros dispositivos para que funcionen de una manera particular. Las instrucciones de programa informático también pueden cargarse en un ordenador, otro aparato de procesamiento de datos programable u otros dispositivos para hacer que se realicen una serie de etapas operativas en el ordenador, otros aparatos programables u otros dispositivos para producir un proceso implementado por ordenador de modo que las instrucciones que se ejecutan en el ordenador u

otro aparato programable proporcionen procesos para implementar las funciones/actos especificados en el bloque o bloques del diagrama de flujo y/o diagrama de bloques.

5 [0015] A lo largo de la siguiente descripción, las letras mayúsculas y en negrita indican matrices (por ejemplo, **A**) y las letras minúsculas en negrita indican vectores (por ejemplo, **x**).

[0016] En lo sucesivo, x_i se usa para indicar la i .ª entrada de vector **x** y a_i se usa para indicar la i .ª columna de matriz **A**.

10 [0017] El i .º vector de muestra, tal como la i .ª ventana de la secuencia de entrada o la i .ª matriz de muestreo se indica por superíndice (por ejemplo, $x^{(i)}$ o $A^{(i)}$).

[0018] El producto interno entre dos vectores **a** y **b** se indica como $\langle a, b \rangle := a^T b$.

15 [0019] $|S|$ es indicativo de la cardinalidad de un conjunto **S**.

[0020] $E_x[\cdot]$ es indicativo de la expectativa condicional $E_x[\cdot] = E[\cdot|x]$

20 [0021] En el campo de la detección comprimida, los términos usados con frecuencia para caracterizar vectores o matrices son “dispersión k ”, “coherencia mutua”, “propiedad de isometría restringida” y “modelo genérico de dispersión k ”. En lo sucesivo, dichos términos se elaboran brevemente para ilustrar el significado deseado a través de este documento.

25 [0022] Para un vector $x \in R^n$ su soporte se define como $\text{supp}(x) := \{i: x_i \neq 0\}$.

[0023] Para el mismo vector $x \in R^n$ su pseudonorma l_0 se define como $\|x\|_0 := |\text{supp}(x)|$, en la que x_0 es la cardinalidad del soporte.

30 [0024] Un vector **x** es disperso si y solo si

$$\|x\|_0 \leq k.$$

[0025] Típicamente $k \ll n$, siendo n el número de columnas de la matriz **A**.

35 [0026] De forma ejemplar si $x = [0 \ 0 \ 0 \ 3 \ 0 \ -1 \ 0 \ 0 \ 5 \ 0]$, $x \in R^{10}$, $\|x\|_0 = 3$, y **x** es 3 disperso.

[0027] Para una matriz $A \in R^{m \times n}$ la coherencia mutua se define como el producto interno normalizado más grande entre dos columnas diferentes de la matriz **A**, como sigue:

40
$$\mu(A) := \max_{1 \leq i, j \leq n, i \neq j} \frac{|\langle a_i, a_j \rangle|}{\|a_i\|_2 \cdot \|a_j\|_2}.$$

[0028] Para una matriz $A \in R^{m \times n}$ y dado $0 < k < n$, la matriz **A** se dice que satisface la propiedad de isometría restringida (RIP) si existe $\delta_k \in [0, 1]$ de modo que:

45
$$(1 - \delta_k) \|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k) \|x\|_2^2.$$

[0029] Esta condición debe ser válida para todos los vectores dispersos k $x \in R^n$, donde δ_k necesita ser lo suficientemente pequeño.

50 [0030] Se conoce que las matrices aleatorias, tales como Gaussian, Bernoulli, filas seleccionadas aleatoriamente de matrices DFT se han usado como matrices para la detección comprimida en la literatura, ya que satisfacen la propiedad de isometría restringida con alta probabilidad. Los ejemplos de matrices que satisfacen la propiedad de isometría restringida son:

- 55 a) vectores aleatorios n muestreados de la esfera de unidad dimensional m ,
- b) matrices de Fourier parciales aleatorias obtenidas de forma aleatoria y uniforme seleccionando filas m de la matriz de Fourier dimensional n ,
- 60 c) matrices gaussianas aleatorias que tienen $A_{i,j} \sim N(0, 1/m)$,

d) matrices aleatorias de Bernoulli donde $A_{i,j} \in \left\{ \frac{1}{\sqrt{m}}, -\frac{1}{\sqrt{m}} \right\}$ con igual probabilidad.

[0031] Para los últimos dos casos, la matriz **A** satisface un prescrito $\bar{\delta}_k$ para cualquier $k \leq c_1 m / \log(n/k)$ con una probabilidad mayor o igual a $1 - 2e^{-c_2 m}$, donde c_1 y c_2 son constantes que solo dependen de $\bar{\delta}_k$.

[0032] Otros términos empleados en este documento se definen a continuación:

[0033] “Recursivamente” se emplea para indicar que el procedimiento se realiza repitiendo los elementos de una manera similar. En matemáticas e informática, el término se refiere a un procedimiento para definir funciones en el que la función que se define se aplica dentro de su propia definición. Específicamente, esto define un número infinito de instancias (valores de función), usando una expresión finita que para algunas instancias se puede referir a otras instancias, pero de manera tal que no se puede producir un bucle o una cadena infinita de referencias. El término también se usa de manera más general para describir un proceso de repetición de objetos de manera autosimilar. La recursión es el proceso al cual se somete un procedimiento cuando una de las etapas del procedimiento implica invocar el procedimiento por sí mismo. Se dice que un procedimiento que se somete a recursión es “recursivo”.

[0034] En el procesamiento de señales, una función de ventana (también conocida como función de apodización o función de disminución gradual) es una función matemática que tiene un valor cero fuera de algún intervalo seleccionado. Por ejemplo, a una función que es constante dentro del intervalo y cero en otro lugar se le llama ventana rectangular, que describe la forma de su representación gráfica. Cuando otra función o forma de onda/secuencia de datos se multiplica por una función de ventana, el producto también tiene un valor cero fuera del intervalo: todo lo que queda es la parte donde se superponen, la “vista a través de la ventana”. Las aplicaciones de las funciones de ventana incluyen análisis espectral, diseño de filtros y formación de haz. En aplicaciones típicas, las funciones de ventana usadas son curvas suaves “en forma de campana” no negativas, aunque se pueden usar rectángulos, triángulos y otras funciones. Una definición más general de las funciones de ventana no requiere que estas sean idénticamente cero fuera de un intervalo, siempre que el producto de la ventana multiplicado por su argumento sea integrable al cuadrado y, más específicamente, que la función vaya suficientemente rápido hacia cero. Cuando la longitud de un conjunto de datos a transformar es mayor de lo necesario para proporcionar la resolución de frecuencia deseada, una práctica común es subdividirlo en conjuntos más pequeños y establecerlos en una ventana individualmente. Para mitigar la “pérdida” en los bordes de la ventana, los conjuntos individuales pueden superponerse en el tiempo.

[0035] Si x es un vector lineal $x \in R^n$ y $y \in R^m$ es el vector de las muestras obtenidas

$$y = Ax \quad (1)$$

con $A \in R^{m \times n}$ siendo la matriz de muestreo (detección), la presente invención propone un procedimiento y medios para recuperar x cuando $m \ll n$. El sistema indicado con la ecuación (1) es un sistema lineal subdeterminado. De acuerdo con la detección comprimida, el resultado principal es que si x es k -disperso y $k < c_m / \log(n/k)$, es posible una solución a este sistema lineal indeterminado.

[0036] El problema P_0 que figura a continuación se debe resolver para encontrar el vector más disperso x que da lugar a la medición y . Este problema es, el tiempo polinómico no determinista complejo o NP-complejo, que requiere la búsqueda en todos los subconjuntos de columnas de A . Como se conoce en la técnica, un problema H es NP-complejo si y solo si hay un problema NP-completo L que es tiempo polinómico con reducibilidad de Turing a H (es decir, $L \leq \tau H$). En otras palabras, L puede resolverse en tiempo polinómico mediante una máquina de oráculo con un oráculo para H .

[0037] El problema P_0 se define como

$$\text{minimizar } \|x\|_0 \text{ sujeto a } Ax = y$$

[0038] Dado que el problema P_0 en general es NP-complejo, este problema tiene que ser ‘aproximado’ por procedimientos manejables. Se utilizan dos problemas de optimización convexa para recuperar vectores dispersos de mediciones lineales.

[0039] Un primer problema de optimización se denomina “búsqueda de base” en el que resolver el problema P_0 es equivalente a resolver el problema de minimización l_1 BP:

$$\text{minimizar } \|x\|_1 \text{ sujeto a } Ax = y$$

para todos los vectores x k -dispersos, si A satisface la propiedad de isometría restringida (RIP) con $\delta_{2k} \leq \sqrt{2} - 1$, el problema mencionado anteriormente es equivalente a P_0 . El problema de optimización (BP) se llama búsqueda de base. Dado que (BP) se puede refundir como un programa lineal, la resolución (BP) es más eficiente desde el punto de vista computacional, por ejemplo, por medio de procedimientos de punto interior, a diferencia de resolver el problema (P_0) que en general es irresoluble para grandes instancias.

[0040] Un segundo problema de optimización se denomina como el operador de contracción y selección menos absoluta (LASSO). En presencia de ruido, el modelo de medición $y = Ax$ se convierte en:

$$y = Ax + w; \tag{2}$$

donde $w \in R^m$ representa el ruido de medición aditiva.

[0041] En este contexto, se puede aplicar una variante de búsqueda de base, a saber la eliminación de ruido de búsqueda de base (BPDN), mejor conocida como LASSO en la literatura estadística, como:

$$\text{minimizar } \|x\|_1 \text{ sujeto a } \|Ax - y\|_2 \leq \hat{\sigma} \tag{3}$$

donde $\hat{\sigma} \in R^+$ se selecciona adecuadamente para tener en cuenta el ruido aditivo en las mediciones.

[0042] Como se conoce en la técnica, por dualidad, el problema puede plantearse de manera equivalente como un problema de optimización sin restricciones:

$$\text{minimizar } \|Ax - y\|_2^2 + \lambda \|x\|_1 \tag{4}$$

donde λ es el parámetro de regularización que controla la compensación entre la dispersión y el error de reconstrucción.

[0043] Un teorema útil para desarrollar el esquema recursivo para realizar la detección comprimida en los datos de transmisión es el teorema con respecto al error de LASSO que indica que si A satisface la propiedad de isometría restringida (RIP) con

$\delta_{2k} \leq \sqrt{2} - 1$ la solución x^* a la ecuación (3) cumple la fórmula de:

$$\|x^* - x\|_2 \leq C_0 \cdot \|x - x_k\|_1 / \sqrt{k} + C_1 \cdot \tilde{\sigma} \tag{5}$$

en la que $C_0 \cdot \|x - x_k\|_1 / \sqrt{k}$ es indicativo de emparejamiento erróneo de modelo y $C_1 \cdot \tilde{\sigma}$ es indicativo de ruido. C_0 y C_1 son constantes, donde x_k es el vector x con todos los componentes k más grandes establecidos en 0. En otras palabras, x_k es el vector obtenido preservando muchos elementos- k de x que tienen la mayor magnitud.

[0044] En particular, si x es k -disperso y $\delta_{2k} \leq \sqrt{2} - 1$ luego $\|x^* - x\|_2 \leq C_1 \cdot \tilde{\sigma}$.

[0045] El teorema indica que el error de reconstrucción está limitado por la suma de dos términos: el primero es el error debido al emparejamiento erróneo de modelo, y el segundo es proporcional a la variación del ruido de medición.

[0046] La clave es que la suposición sobre la constante de isometría se satisface con una alta probabilidad mediante matrices obtenidas de vectores aleatorios muestreados de la esfera de la unidad, matrices gaussianas aleatorias y matrices aleatorias de Bernoulli si

$$m \geq C_k \log(n/k)$$

donde C es una constante que depende de cada instancia.

[0047] En la técnica se demostró que el error de LASSO se puede usar para la detección de soporte casi ideal. Para establecer la característica de resultado para la detección de soporte casi ideal, se define un modelo genérico k -disperso.

[0048] Si $x \in \mathbb{R}^n$ indica una señal k-dispersa y $I_x := \text{supp}(x)$ es su conjunto de soporte, se dice que la señal x se genera por un modelo genérico k-disperso si:

- 1) Soporte $I_x \subset \{1, 2, \dots, n\}$ de x se selecciona uniformemente de forma aleatoria, y $|I_x| = k$, y
- 2) Condicionado a I_x , los signos distintos de ceros tienen la misma probabilidad de ser -1 y 1.

[0049] El conjunto de soporte de la señal k-dispersa se puede detectar como sigue:

[0050] Asumiendo que $\mu(A) \leq c_1 / \log n$ para una constante $c_1 > 0$, x se genera a partir de un modelo genérico k-disperso y

$$k \leq c_2 n / (\|A\|_2^2 \log n) \text{ para una constante } c_2.$$

[0051] Si, para $I_x := \text{supp}(x)$,

$$\min_{i \in I_x} |x_i| > 8\sigma \sqrt{2 \log n}$$

la estimación de LASSO obtenida al seleccionar $\lambda = 4\sigma \sqrt{2 \log n}$ para mediciones donde

$$w \sim N(0, \sigma^2 I)$$

obedece:

$$\begin{aligned} \text{supp}(\hat{x}) &= \text{supp}(x) \\ \text{sgn}(\hat{x}_i) &= \text{sgn}(x_i) \text{ para } i \in I_x \end{aligned}$$

con probabilidad al menos

$$1 - \frac{2}{n} \left(\frac{1}{\sqrt{2\pi \log n}} + \frac{|I_x|}{n} \right) - O\left(\frac{1}{n^{2 \log 2}}\right).$$

[0052] Para la matriz de muestreo que tiene elementos

$$A_{i,j} \sim N(0, 1/m)$$

la norma del operador es $\|A\|_2 \sim \sqrt{n/m}$, por tanto, la condición de dispersión en el teorema se convierte en

$$k \leq c_1 m / \log n.$$

[0053] Para que se satisfaga esta desigualdad, el valor de m debe ser

$$m = c_1' k \log n.$$

[0054] Además, para matrices gaussianas aleatorias $\mu(A) \sim \sqrt{2 \log n / m}$ la condición sobre la coherencia se cumple si $m > O(\log^3 n)$.

[0055] Combinando los límites, se requiere que m sea

$$m \geq \min \{O(k \log n), O(\log^3 n)\}.$$

50

[0056] La búsqueda de emparejamiento ortogonal (OMP) es un algoritmo codicioso que tiene como objetivo recuperar vectores x dispersos de una medición silenciosa $y = Ax$. El algoritmo genera un subconjunto de columnas de A , seleccionando iterativamente la columna minimizando el error residual de aproximar y proyectando al tramo lineal de las columnas ya seleccionadas. Se muestra que OMP recupera señales k -dispersas de mediciones sin ruido si la coherencia mutua de la matriz de medición A satisface

$$\mu(A) < \frac{1}{2k-1}.$$

[0057] Existen algoritmos desarrollados para LASSO, inspirados en algoritmos proximales para la optimización convexa no uniforme: ISTA es un procedimiento de gradiente proximal, FISTA es un procedimiento de gradiente proximal acelerado y SALSA es una aplicación del procedimiento alternativo de dirección de multiplicadores.

[0058] Para el error definido como $G(x_t) - G(x^*)$ donde $G(x)$ es la función objetivo de LASSO en la ecuación (4) y $x^* = \arg \min G(x)$, el error se descompone como $1/t$ para ISTA, $1/t^2$ para FISTA y $1/t^2$ para SALSA donde t es el número de iteraciones. En la técnica también se conoce un procedimiento de tipo Newton para la optimización convexa no uniforme, en el que la tasa de convergencia no es peor que $1=t^2$, pero es localmente cuadrática.

[0059] La señal de interés es una secuencia infinita, $\{x_i\}_{i=0,1,\dots}$, como cuando se trata de datos de transmisión continua. Para dicha señal de interés, la i -ésima ventana tomada de la señal de transmisión continua se define como

$$x^{(i)} := [x_i x_{i+1} \dots x_{i+n-1}]^T$$

[0060] Si $x^{(i)}$ se conoce que es disperso, las herramientas analizadas anteriormente son aplicables para recuperar la porción de señal en cada ventana, de ahí el flujo de datos. Sin embargo, las operaciones involucradas son costosas y una implementación en línea eficiente es incierta.

[0061] Por lo tanto, de acuerdo con la presente invención, se propone un procedimiento para muestrear y recuperar datos de transmisión continua de manera eficiente. Dicho procedimiento es un procedimiento de detección comprimida recursivo, y se describirá en detalle a continuación. Como se mostrará en detalle más adelante en el presente documento, el procedimiento de la invención presenta baja complejidad en las partes de muestreo y estimación, lo que hace que sus algoritmos sean adecuados para una implementación en línea eficiente.

[0062] Con $\{x_i\}$ se indica la secuencia infinita $\{x_i\}_{i=0,1,\dots}$, y con $x^{(i)} \in R^n$ la i -ésima ventana de longitud n .

[0063] Un modelo dinámico con entrada en R se representa como

$$x^{(i)} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} x^{(i-1)} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} x_{i+n-1} \quad (6)$$

[0064] El procedimiento de la presente invención permite el diseño de un algoritmo robusto de ventana deslizante de baja complejidad que proporciona estimaciones $\{\hat{x}\}$ que usa mediciones sucesivas $y^{(i)}$ de la forma

$$y^{(i)} = A^{(i)} x^{(i)} \quad (7)$$

donde $\{A^{(i)}\}$ es una secuencia de matrices de medición. El diseño de un algoritmo robusto de ventana deslizante de baja complejidad es posible si $\{x_i\}$ es suficientemente disperso en cada ventana, a saber si $\|x^{(i)}\|_0 \leq k$ para cada i , donde $k \ll n$ o si esta condición se respeta con una probabilidad suficientemente alta.

[0065] Si $A^{(i)}$ satisface la propiedad de restricción de isometría con $\delta_{2k} \leq \sqrt{2} - 1$, los procedimientos revisados anteriormente en el presente documento pueden aplicarse para estimar progresivamente $\{x^{(i)}\}$. Ejecutar dicho algoritmo en línea es costoso y, por lo tanto, es conveniente diseñar una alternativa a dicho procedimiento que, en el mejor de los casos, pueda calificarse como ad-hoc.

[0066] De acuerdo con la presente invención, se potencia la superposición de señal entre ventanas sucesivas, que consiste en muestreo recursivo (una etapa de codificación) y estimación recursiva (una etapa de decodificación) como sigue:

5 Con respecto a la etapa del muestreo recursivo: Para evitar una multiplicación matricial para cada $y(i)$, $A(i)$ está diseñado de modo que $y(i)$ pueda volverse a usar en el cómputo $y(i+1)$ con una sobrecarga de cómputo mínima:

$$y^{(i+1)} = f(y^{(i)}, x_{i+n}, x_i)$$

10 **[0067]** Con respecto a la etapa de estimación recursiva: Para acelerar la convergencia de un esquema de optimización iterativo, se usa la estimación correspondiente a la ventana previa, $\hat{x}^{(i-1)}$, para derivar un punto de partida, $\hat{x}_{[0]}^{(i)}$ del procedimiento de estimación de $\hat{x}^{(i)}$:

$$\hat{x}_{[0]}^{(i)} = g(\hat{x}^{(i-1)}, \hat{y}^{(i)})$$

15 **[0068]** A continuación, se describirán con más detalle las dos etapas de muestreo y estimación recursiva.

[0069] Con respecto a la etapa de muestreo recursivo, se propone un esquema de muestreo recursivo que presenta una sobrecarga computacional mínima. En la primera iteración, $t = 0$, no hay una estimación previa, por lo tanto, lo que se calcula es

20
$$y^{(0)} = A^{(0)} x^{(0)}$$

[0070] Una secuencia de matrices de detección $A^{(i)}$ recursivamente se selecciona como:

25
$$A^{(i+1)} = \begin{bmatrix} a_2^{(i)} & a_3^{(i)} & \dots & a_n^{(i)} & a_1^{(i)} \end{bmatrix} = A^{(i)} P \quad (8)$$

donde $a_i^{(i)}$ es la i -ésima columna de $A^{(i)}$ y P es la matriz de permutación:

$$P := \begin{bmatrix} 0 & \dots & 0 & 1 \\ 1 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & 0 \end{bmatrix}$$

30 **[0071]** El siguiente lema garantiza el éxito de este esquema de codificación: Si $A^{(0)}$ satisface la propiedad de isometría restrictiva para la k dada con constante δ_k , a continuación $A^{(i)}$ como se define en la ecuación (8) anterior, satisface también la propiedad de isometría restrictiva, para la misma k , y δ_k .

35 **[0072]** De la ecuación (8) se desprende que $A^{(i)} = A^{(0)} P^i$, y $A^{(i)}$ se obtiene reordenando las columnas de $A^{(0)}$. Por definición, la propiedad de isometría restrictiva depende solo de $\|X\|_0$, y es insensible a las permutaciones de las entradas de x , equivalente a las permutaciones de las columnas de $A^{(0)}$.

[0073] Dada la selección recursiva particular de $A^{(i)}$, $y^{(i+1)}$ se calcula como:

40
$$\begin{aligned} y^{(i+1)} &= A^{(i+1)} x^{(i+1)} = \sum_{l=1}^n x_{i+l} a_l^{(i+1)} = \sum_{l=1}^{n-1} x_{i+l} \underbrace{a_l^{(i+1)}}_{a_{l+1}^{(i)}} + x_{i+n} \underbrace{a_n^{(i+1)}}_{a_1^{(i)}} = \sum_{l=2}^n x_{i-1+l} a_l^{(i)} + x_i a_1^{(i)} - x_i a_1^{(i)} + x_{i+n} a_1^{(i)} = \\ &= y^{(i)} + (x_{i+n} - x_i) a_1^{(i)}. \end{aligned}$$

[0074] Cuando la señal de interés comprende ruido, las mediciones ruidosas se indican como

45
$$y^{(i)} = b^{(i)} + w^{(i)} \quad \text{donde } b^{(i)} = A^{(i)} x^{(i)}$$

[0075] Por lo tanto,

$$b^{(i+1)} = A^{(i+1)} x^{(i+1)} = b^{(i)} + (x_{i+n} - x_i) a_1^{(i)}$$

50 **[0076]** Al sustituir el valor de $b^{(i+1)}$ en la ecuación para $y^{(i+1)}$ lo que se obtiene es:

$$y^{(i+1)} = b^{(i+1)} + w^{(i+1)} = b^{(i)} + (x_{i+n} - x_i) a_1^{(i)} + w^{(i+1)} = y^{(i)} + (x_{i+n} - x_i) a_1^{(i)} + w^{(i+1)} - w^{(i)}$$

[0077] Definiendo $z^{(i)} := w^{(i)} - w^{(i-1)}$, $z^{(i)}$ y $z^{(i+1)}$ son independientes si $\{w^{(i)}\}$ es un proceso de incremento independiente.

[0078] La selección particular de las matrices de muestreo $\{A^{(i)}\}_{i=0,1,\dots}$ dado en la ecuación (8) satisface la condición

$$A^{(i)} x^{(i)} = A^{(0)} P^i x^{(i)}$$

[0079] Al definir $v^{(i)}$ como $v^{(i)} := P^i x^{(i)}$ el muestreo recursivo se puede ver como codificación $v^{(i)}$ usando la misma matriz de medición $A^{(0)}$. Con la estructura particular de $x^{(i)}$ dada en la ecuación (6), todas las entradas de $v^{(i)}$ y $v^{(i-1)}$ son iguales excepto $v_{i-1}^{(i)}$.

[0080] Para la estimación recursiva de la señal de interés, una estimación de $\hat{x}^{(i+1)} = [\hat{x}_{i+1}^{(i+1)} \dots \hat{x}_{i+n}^{(i+1)}]$ se

$$\hat{x}^{(i)} = [\hat{x}_i^{(i)} \dots \hat{x}_{i+n-1}^{(i)}].$$

encuentra potenciando la estimación

[0081] En un algoritmo iterativo, la velocidad de convergencia depende de la distancia del punto de partida a la solución óptima. Para lograr una convergencia acelerada, se potencia la superposición entre las ventanas y se usa el siguiente punto de partida:

$$x_{[0]}^{(i)} = [\hat{x}_2^{(i-1)} \hat{x}_3^{(i-1)} \dots \hat{x}_{n-1}^{(i-1)} E_{x^{(i-1)}} [x_{i+n-1}]]^T$$

donde $\hat{x}_j^{(i-1)}$ para $j=2, \dots, n-1$ es la porción de la solución óptima basada en la ventana previa. Esto se denomina 'arranque en caliente' en la literatura de optimización. Al seleccionar el punto de partida como tal, se reduce el número esperado de iteraciones para la convergencia. Esto se demostrará realmente en una porción posterior de este documento donde los resultados cuantitativos se analizan específicamente.

[0082] A continuación, se analizan las señales dispersas de interés, ya que se encuentran en una base ortonormal dada.

[0083] Hasta ahora en este documento se suponía implícitamente que para un determinado $n \in \mathbb{Z}^+$, las ventanas $x^{(i)}$ de longitud n obtenidas de la secuencia $\{x_i\}$ satisfacen la restricción de dispersión $\|x^{(i)}\|_0 \leq k, \forall i$. En general, podría ser el caso de que $x^{(i)}$ no es disperso en sí mismo, pero se puede representar de forma dispersa en una base seleccionada apropiadamente.

[0084] $x^{(i)} \in \mathbb{R}^n$ es dispersamente representable en una base ortonormal dada Φ como $x^{(i)} = \Phi \alpha^{(i)}$, donde $\alpha^{(i)}$ es disperso.

[0085] Asumiendo una base común para toda la secuencia $\{x_i\}$ sobre ventanas de tamaño n , lo que se obtiene es:

$$y^{(i)} = A^{(i)} x^{(i)} = A^{(i)} \Phi \alpha^{(i)}.$$

[0086] Para que la estimación de detección comprimida se transfiera, $A^{(i)} \Phi$ necesita satisfacer la propiedad de isometría restringida. La propiedad de isometría restringida se satisface con alta probabilidad para el producto de una matriz aleatoria $A^{(i)}$ y cualquier matriz fija. En este caso, el problema de LASSO para recuperar la señal en la base de dispersión, $\alpha^{(i)}$ se expresa como:

$$\text{minimizar } \|A^{(i)} \Phi \alpha^{(i)} - y\|_2^2 + \lambda \|\alpha^{(i)}\|_1$$

donde la señal de entrada viene dada por $x^{(i)} = \Phi \alpha^{(i)}$.

[0087] El problema que debe resolverse es cómo encontrar una actualización recursiva para $\alpha^{(i+1)}$ basada en $\alpha^{(i)}$ para tener una buena estimación inicial de la convergencia acelerada en la estimación recursiva, como $E[\alpha^{(i+1)} | \alpha^{(i)}]$.

[0088] Usando la ecuación (6) lo que se obtiene es:

$$x^{(i+1)} = \prod x^{(i)} + \begin{bmatrix} 0_{n-1} \\ 1 \end{bmatrix} (x_{i+n} - x_i)$$

donde 0_{n-1} es un vector de longitud $n-1$ que tiene todos los elementos establecidos en 0.

5 **[0089]** Ya que $x^{(i)} = \Phi \alpha^{(i)}$, se obtiene:

$$x_i = x_0^{(i)} = [1 | 0_{n-1}] \Phi \alpha^{(i)}$$

y

10

$$x_{i+1} = x_{n-1}^{(i+1)} = [0_{n-1} | 1] \Phi \alpha^{(i+1)}.$$

[0090] Multiplicando a la izquierda ambos lados por $\Psi = \Phi^{-1}$ en la ecuación previa, sigue:

$$\begin{aligned} \alpha^{(i+1)} &= \Psi x^{(i+1)} = \Psi \prod x^{(i)} + \Psi \begin{bmatrix} 0_{n-1} \\ 1 \end{bmatrix} (x_{i+n} - x_i) = \\ &= \Psi \prod x^{(i)} + (x_{i+n} - x_i) \psi_{n-1} = \\ &= \Psi \prod \Phi \alpha^{(i)} + \psi_{n-1} \left((\phi_{n-1}^T)^T \alpha^{(i+1)} - (\phi_0^T)^T \alpha^{(i)} \right) \end{aligned} \quad (11)$$

15

donde $\Pi := P^T$ es la matriz de permutación dada en la ecuación (9), y $(\phi_0^T)^T$ y $(\phi_{n-1}^T)^T$ son la primera y la última fila de la base ortonormal Φ , respectivamente.

20 **[0091]** La base de Fourier es de particular interés para Φ ya que se puede derivar una regla de actualización eficiente para dicha base. El muestreo recursivo para la base de Fourier es el siguiente:

[0092] Si Φ es la matriz $n \times n$ inversa de la matriz Transformada de Fourier discreta (IDFT) con entradas

25

$$\Phi_{i,j} = \omega^{(i-1)(j-1)} / \sqrt{n}$$

Donde $\omega := e^{j \frac{2\pi}{n}}$, en dicho caso:

$$\alpha^{(i+1)} = \Omega_n \alpha^{(i)} + \psi_{n-1} \left((\Phi_{n-1}^T)^T \alpha^{(i+1)} - (\Phi_0^T)^T \alpha^{(i)} \right) \quad (12)$$

30 donde Ω_n es la matriz diagonal $n \times n$ con $(\Omega_n)_{i,i} = \omega^{(i-1)}$, y $\Psi = \Phi^{-1}$ es la base de Fourier ortonormal.

[0093] En este caso particular, la ecuación (11) se simplifica a la ecuación (12) ya que el desplazamiento circular en el dominio del tiempo corresponde a la multiplicación por exponenciales complejos en el dominio de Fourier, como $\Psi \Pi = \Omega_n \Psi$ y $\Psi \Phi = I$.

35

[0094] De lo anterior se puede ver que aunque el número de cálculos para calcular $\alpha^{(i+1)}$ basado en $\alpha^{(i)}$ es $O(n^2)$ en general, para la base de Fourier es $O(n)$ ya que la matriz que multiplica $\alpha^{(i)}$ es diagonal.

40 **[0095]** Como se muestra anteriormente, en presencia de ruido, la estimación de $E[\alpha^{(i+1)} | \alpha^{(i)}]$ se usa como punto de partida en el solucionador iterativo LASSO para un arranque en caliente, para lograr una convergencia acelerada.

[0096] Una forma de mejorar la varianza del error es promediando las estimaciones obtenidas de ventanas sucesivas. Usando las estimaciones $\hat{x}^{(j)}$ la estimación promedio, \bar{x}_i , se define para la i .ª entrada de la señal de transmisión continua, es decir, x_i , como:

45

$$\bar{x}_i := \frac{1}{\min\{i+1, n\}} \sum_{j=\max\{0, i-n+1\}}^i \hat{x}_{i-j}^{(j)} \quad (13)$$

donde promediamos n muchas estimaciones para $i \geq n-1$ e $i+1$ muchas estimaciones para $i < n-1$.

[0097] Considerando $i \geq n-1$ para simplicidad de notación, lo que se obtiene adicionalmente es:

$$\frac{1}{n} \sum_{j=i-n+1}^i (\hat{x}_{i-j}^{(j)} - x_i)^2 \stackrel{(a)}{\geq} \left(\frac{1}{n} \sum_{j=i-n+1}^i (\hat{x}_{i-j}^{(j)} - x_i) \right)^2 = (\bar{x}_i - x_i)^2$$

donde (a) se desprende de la desigualdad de Jensen aplicada a x^2 . La desigualdad implica que el error de reconstrucción solo puede reducirse promediando las estimaciones.

[0098] A continuación, la esperada norma l_2 del error de reconstrucción $|x - \hat{x}|^2$ se analiza, expandiendo la ecuación y usando

$$Cov[\hat{x}_{j+1}^{(i-j)}, \hat{x}_{k+1}^{(i-k)}] = 0 \text{ para } j \neq k.$$

[0099] Por conveniencia de notación se considera el caso $i \geq n-1$, siendo similar al caso $i < n-1$.

$$E_x[(\bar{x}_i - x_i)^2] = E_x \left[\left(\frac{1}{n} \sum_{j=i-n+1}^i \hat{x}_{i-j}^{(j)} - x_i \right)^2 \right] = (E_x[\hat{x}_1^{(i)}] - x_i)^2 + \frac{1}{n} E_x \left[(\hat{x}_1^{(i)} - E_x[\hat{x}_1^{(i)}])^2 \right]$$

[0100] La igualdad resultante es la llamada descomposición de la varianza de sesgo del estimador. Se ve que a medida que aumenta la longitud de la ventana, el segundo término se acerca a cero y el error de reconstrucción asintóticamente converge al cuadrado del sesgo de LASSO.

[0101] A continuación, se analizará el algoritmo propuesto de acuerdo con la presente invención.

[0102] En general, el operador de selección y contracción menos absoluta (LASSO) produce un estimador sesgado a medida que mapea $R^m \rightarrow R^n$ donde $m \leq n$. Si las superposiciones entre ventanas se usan promediando las estimaciones de LASSO directamente, el error de reconstrucción no llega a cero debido al sesgo. Por otra parte, la estimación de mínimos cuadrados (LSE) es un estimador imparcial para un sistema sobredeterminado; como se muestra anteriormente en este documento, LASSO se puede usar para estimar el conjunto de soporte de la señal en la que LSE se puede aplicar posteriormente. Basado en estas observaciones, se propone un procedimiento de estimación de dos etapas para recuperar la señal muestreada para reducir el error de estimación. En primer lugar, las estimaciones de LASSO $\hat{x}^{(i)}$ se obtienen, las cuales a continuación se usan como entrada para un algoritmo desesgado. Para el desesgado, la LSE se realiza en el conjunto de soporte de la estimación LASSO para obtener $\tilde{x}^{(i)}$, que proporciona un estimador imparcial de los verdaderos distintos de cero de la señal cuando $\|x^{(i)}\|_0 \leq m$ y el soporte se identifican correctamente. Posteriormente se promedian las estimaciones desesgadas obtenidas en ventanas sucesivas. El diagrama de bloques del procedimiento y el pseudocódigo para el algoritmo se pueden ver en las figuras adjuntas y el Algoritmo 1 a continuación, respectivamente.

[0103] Algoritmo 1 Detección comprimida recursiva:

Entrada: $A^{(0)} \in R^{m \times n}$, $\{x\}$, $\lambda \geq 0$

Salida: estimación $\{x\}$

1. la selección de soporte cuenta: $\{v\} \leftarrow \{0\}$, estimación de señal: $\{x\} \leftarrow \{0\}$

2. para $i = 0, 1, 2, \dots$ do

3. $x^{(i)} \leftarrow (x_i x_{i+1} \dots x_{i+n-1})^T$

4. $y^{(i)} \leftarrow A^{(i)} x^{(i)} + w^{(i)}$ ► codificación

5. $\hat{x}^{(i)} \leftarrow \underset{x \in R^n}{\operatorname{arg\,mín}} \|A^{(i)} x - y^{(i)}\|_2^2 + \lambda \|x\|_1$ ► LASSO

6. $l \leftarrow \sup \rho(\hat{x}^{(i)})$ ► estimación de soporte

7. $v_{hi} \leftarrow v_{hi} + 1$ ▷ incrementar el recuento de selección en soporte

8. $\tilde{x}^{(i)} \leftarrow \underset{x \in R^d}{\operatorname{arg\,mín}} \left\| A_i^{(i)} x - y^{(i)} \right\|_2^2$ ▷ LSE en conjunto de soporte

9. $\bar{x}_i^{(i)} \leftarrow \left((v_i^{(i)} - 1) \otimes \bar{x}_i^{(i)} + \tilde{x}^{(i)} \right)$ ▷ actualizar las estimaciones promedio donde los operadores cruzados por cero indican multiplicación y división por elementos sabios, respectivamente

10. $A^{(i)} \leftarrow A^{(i-1)} P$ ▷ para muestreo recursivo

11. fin.

[0104] En la figura 1, se representa un diagrama de bloques del procedimiento para detectar datos de transmisión continua de acuerdo con un modo de realización de la presente invención. Un procedimiento de acuerdo con este modo de realización comprende la etapa de muestrear recursivamente un flujo de entrada de datos usando las etapas de superposición de ventanas para obtener al menos una medición previa, y emplear dicha medición previa para obtener una medición posterior. Más precisamente, el procedimiento comprende la etapa de muestreo recursivo 102 de un flujo de entrada de datos. Los datos muestreados durante la etapa del muestreo recursivo 102 se procesan por medio de una etapa de estimación recursiva 104. La información con respecto a una estimación previa, obtenida durante una etapa de obtención de estimación previa 116 previa al muestreo recursivo, también se imputa durante la etapa de estimación recursiva 104. Los datos obtenidos durante la etapa de estimación recursiva 104 se usan durante la etapa de detección de soporte 106, como se describe anteriormente en relación con la etapa 6 del algoritmo de muestreo recursivo. La información procesada como resultado de la etapa de detección de soporte se puede usar para la estimación de conteo, como se muestra en la etapa 110 de la figura 1 y como corresponde a la etapa 7 del algoritmo de muestreo recursivo. La información obtenida en la etapa 106 se usa durante la etapa 108 de calcular el LSE en el conjunto de soporte, como se describe anteriormente en relación con la etapa 8 del algoritmo de muestreo recursivo. Posteriormente, el LSE en el conjunto de soporte y el recuento de estimación se promedian en la etapa 112, como se describe anteriormente en relación con la etapa 9 del algoritmo de muestreo recursivo. De acuerdo con un modo de realización del procedimiento, la estimación previa obtenida durante la etapa 116 también se promedia en la etapa 112. Se obtiene una nueva estimación en la etapa 114, como se describe anteriormente en relación con la etapa 10 del algoritmo de muestreo recursivo.

[0105] El procedimiento de acuerdo con un modo de realización de este tipo también es capaz de analizar la complejidad computacional y el error de estimación del procedimiento.

[0106] Más precisamente, a continuación se analiza la complejidad y la varianza de error del procedimiento, y se introduce un algoritmo de votación para una detección de soporte más robusta.

[0107] El diagrama de bloques de la figura 1 que representa un modo de realización del procedimiento puede extenderse a un procedimiento que comprende una superposición variable entre ventanas sucesivas.

[0108] Lo que se toma en consideración es una generalización en la que la detección se realiza por medio del establecimiento de ventanas recurrentes con $0 < \tau \leq n$ superposiciones.

[0109] Si $A \in R^{m \times n}$ indica la matriz de muestreo, i indica el índice de la ventana y η_i es la eficiencia de muestreo, es decir, la proporción del total de las muestras tomadas hasta el tiempo $(n + i)$ con respecto al número de entradas detectadas. Para una ventana, la eficiencia de muestreo es m/n ya que la matriz de muestreo es $A \in R^{m \times n}$.

[0110] Al final de la i -ésima ventana, $n + (i - 1)\tau$ elementos se han recuperado al tomar m muestras de im . La eficiencia de muestreo asintótico es:

$$\eta := \lim_{i \rightarrow \infty} \eta_i = \lim_{i \rightarrow \infty} \frac{im}{n + (i - 1)\tau} = \frac{m}{\tau}$$

[0111] Si, en cambio, la codificación se usa en una codificación en, usando una actualización de rango- τ , es decir, usando la matriz obtenida por el desplazamiento circular de la matriz de muestreo τ veces, la eficiencia de muestreo asintótico se convierte en:

$$\eta = \lim_{i \rightarrow \infty} \frac{m + (i - 1)\tau}{n + (i - 1)\tau} = 1$$

[0112] En el último caso, el enfoque de muestreo recursivo es asintóticamente equivalente a tomar una muestra para cada instancia de tiempo. El beneficio de dicho enfoque radica en la supresión de ruido. Al tomar ventanas superpuestas, cada elemento se detecta como mínimo $\lfloor n/\tau \rfloor$ muchas veces, por lo tanto, se puede usar la decodificación colaborativa usando múltiplos para incrementar la exactitud de la estimación.

[0113] La aplicación de LASSO para estimar el soporte de señales se discutió anteriormente en este documento. Además, en este documento, se introduce un procedimiento que usa el soporte estimado en ventanas sucesivas para una detección de soporte más robusta en ruido de alta medición. Cuando las magnitudes de la señal no son lo suficientemente altas como para estar por encima del nivel de ruido, el mecanismo de detección de soporte mencionado anteriormente puede perder posiciones distintas de cero (falso negativo) o detectar falso soporte (falso positivo). Por lo tanto, el teorema de detección de soporte analizado anteriormente en el presente documento podría usar una constante de regularización más pequeña, λ , para detectar el soporte para valores mínimos menores distintos de cero con el coste de reducir la probabilidad de éxito. Suponiendo que los falsos positivos se distribuyen aleatoriamente en el conjunto $\{0, \dots, n-1\} \setminus \text{supp}(x^{(i)})$ para todo i , al introducir un umbral en el número mínimo de veces que una posición necesita que se detecte como soporte, podemos disminuir los falsos positivos en el régimen de baja señal a ruido.

[0114] Al introducir una 'máquina de votación', el algoritmo de dos etapas obtenido con votación puede representarse dentro del bloque de detección de soporte 106 en la figura 1. El algoritmo de dos etapas funciona resolviendo LASSO como

$$\hat{x}^{(i)} = \arg \min_{x \in \mathbb{R}^n} \left(\|A^{(i)} x - y^{(i)}\|_2^2 + \lambda \|x\|_1 \right).$$

[0115] Si ξ_1 se establece como > 0 , los índices donde el valor absoluto de $\hat{x}^{(i)}$ es mayor que ξ_1 están destinados a estimar el soporte de $x^{(i)}$; en la i .ª ventana

$$S_t^{(i)} = \{j : |\hat{x}_j^{(i)}| \geq \xi_1\}.$$

[0116] La secuencia que contiene los votos se define como $\{v_j\}$ y la cantidad de veces que se usa un índice i en la estimación de mínimos cuadrados (LSE) como $\{L_i\}$. Al comienzo del algoritmo, el $\{v_j\}$ y el $\{L_i\}$ se establecen en cero. Para cada ventana, se agregan los votos sobre las posiciones que están en el conjunto, indicándose el conjunto

mediante $S_t^{(i)}$ como $v_{S_t^{(i)}-1} \leftarrow v_{S_t^{(i)}+1} + 1$, donde el subíndice $S_t^{(i)} + i$ se usa para traducir los índices dentro de la ventana a índices globales en los datos de transmisión continua. Aplicando el umbral $\xi_2 \in \mathbb{Z}^+$ sobre el número de votos $\{v_j\}$, los índices que se han votado suficientes veces para ser aceptados como distintos de cero se encuentran y se almacenan en

$$S_g^{(i)} = \{j : v_{j+i} \geq \xi_2\}.$$

[0117] El umbral ξ_2 se selecciona de modo que $|S_g^{(i)}| < m$, por lo tanto, proporcionando un sistema sobredeterminado para la LSE. El problema de mínimos cuadrados sobredeterminado se resuelve en base a estos índices,

$$\tilde{x}^{(i)} = \left(A_{S_g^{(i)}}^{(i)T} A_{S_g^{(i)}}^{(i)} \right)^{-1} A_{S_g^{(i)}}^{(i)T} y^{(i)}$$

[0118] Donde $A_{S_g^{(i)}}^{(i)}$ es la matriz obtenida extrayendo columnas de $A^{(i)}$ indexadas por el conjunto $S_g^{(i)}$. A continuación, para realizar un promedio de las estimaciones de mínimos cuadrados, el número de recuperaciones se incrementa para las entradas usadas en el procedimiento de LSE como

$$L_{S_g^{(i)}+i} \leftarrow L_{S_g^{(i)}+i} + 1.$$

[0119] El vector $\bar{x}^{(i)}$ es en $\mathbb{R}^{|S_g^{(i)}|}$ y el vector $x' \in \mathbb{R}^n$ se genera estableciendo los elementos indexados por el $S_g^{(i)}$ a $x'_{S_g^{(i)}} = \tilde{x}^{(i)}$ y todos los otros índices a 0. A continuación, las estimaciones promedio se actualizan como:

$$\bar{x}_{i+j} \leftarrow \frac{L_i - 1}{L_i} \bar{x}_{i+j} + \frac{1}{L_i} x_j \text{ para } j=0, 1, \dots, n-1.$$

5 **[0120]** La varianza del error de estimación y la complejidad computacional del procedimiento de acuerdo con un modo de realización se analiza a continuación.

[0121] Antes de realizar el análisis de la varianza del error de estimación del procedimiento RCS de dos etapas, se introduce el siguiente lema:

10 **[0122]** Lema 5.1: indicando la solución a LASSO en la ecuación (4) por \hat{v} , su soporte establecido por $I := \text{supp}(\hat{v})$, y la solución de mínimos cuadrados restringida a columnas A_I por

$$\tilde{v} := \arg \min_{\substack{v \in R^n \\ v_{I^c} = 0}} \|Av - y\|_2^2,$$

15 se satisface la siguiente ecuación:

$$\|A_I(\tilde{v}_I - \hat{v}_I)\|_2^2 \leq \left(\|\tilde{v}_I\|_1 - \|\hat{v}_I\|_1 \right) \frac{\lambda}{2}.$$

20 **[0123]** Independientemente del soporte seleccionado, se cumple lo siguiente para la estimación de LASSO y los mínimos cuadrados aplicados en el soporte de la estimación de LASSO:

$$\hat{v} = \arg \min_{\substack{v \in R^n \\ v_{I^c} = 0}} \|Av - y\|_2^2 + \lambda \|v\|_1 = \arg \min_{\substack{v \in R^n \\ v_{I^c} = 0}} \|Av - y\|_2^2 + \lambda \|v\|_1$$

$$\tilde{v} = \arg \min_{\substack{v \in R^n \\ v_{I^c} = 0}} \|Av - y\|_2^2$$

25 donde $I = \text{supp}(\hat{v})$. Las condiciones de optimización del rendimiento de LASSO:

$$A_I^T (y - A_I \hat{v}_I) = \frac{\lambda}{2} \text{sgn}(\hat{v}_I) \tag{15}$$

30 **[0124]** donde $\text{sgn}(v)$ es la función de signo extendida a vectores. Ecuaciones normales leídas

$$A_I^T A_I \tilde{v}_I = A_I^T y$$

[0125] Por lo tanto:

$$\begin{aligned} \|A_I(\tilde{v}_I^{(i)} - \hat{v}_I^{(i)})\|_2^2 &= (\tilde{v}_I - \hat{v}_I)^T A_I^T A_I (\tilde{v}_I - \hat{v}_I) \stackrel{(a)}{=} (\tilde{v}_I - \hat{v}_I)^T \frac{\lambda}{2} \text{sgn}(\hat{v}_I) \leq \left(\|\tilde{v}_I\|_1 - \|\hat{v}_I\|_1 \right) \frac{\lambda}{2} \stackrel{(c)}{=} \\ &= \left(\|\tilde{v}_I\|_1 - \|\hat{v}_I\|_1 \right) \frac{\lambda}{2} \end{aligned}$$

35 donde (a) se deriva de la ecuación (15) y las ecuaciones normales, (b) se derivan de

$$\hat{v}_I^T \text{sgn}(\hat{v}_I) = \|\hat{v}_I\|_1$$

40 y

$$\tilde{v}_I^T \text{sgn}(\hat{v}_I) \leq \|\tilde{v}_I\|_1$$

45 y (c) se deriva como $\hat{v}_I = \tilde{v}_I = 0$.

[0126] Si se supone que el ruido no está correlacionado independientemente de la detección de soporte, es decir, $\text{Cov}[w^{(i)}, w^{(j)} | A_i, A_j = 0]$ para $i \neq j$ donde A_i indica el evento de que la detección de soporte tiene éxito en la ventana i .^a,

50

$$A_i := \left\{ \text{supp}(\hat{x}^{(i)}) = \text{supp}(x^{(i)}) \right\}$$

[0127] El siguiente teorema proporciona un límite superior en la varianza del error de reconstrucción del algoritmo de dos etapas propuesto.

5 **[0128]** De acuerdo con el teorema con respecto a la varianza de error para el algoritmo propuesto, en las condiciones analizadas anteriormente en relación con la detección de soporte y bajo el supuesto de que el ruido no está correlacionado independientemente de la detección de soporte para k y n fijos, la siguiente condición es válida:

$$E\left[(\bar{x}_i - x_i)^2\right] \leq P(A) \frac{\sigma^2}{\left(1 - \sqrt{k/m}\right)} + \frac{P(A^c)}{n} \sum_{i=0}^{q-1} E\left[\frac{\lambda \left(\|\hat{x}_i^{(i)}\|_1 - \|\hat{x}_i^{(i)}\|_1 \right)}{2(1 - \sqrt{k/m})} + \|\hat{x}^{(i)} - x^{(i)}\|_2^2 \right],$$

10 donde

$$P(A) \geq 1 - 2 \left(\frac{1}{\sqrt{2\pi \log n}} + \frac{|I|}{n} \right) - \frac{c}{n^{2 \log 2 - 1}}$$

15 y

$$P(A^c) = 1 - P(A).$$

20 **[0129]** Para demostrar la validez del teorema mencionado anteriormente, se deben cumplir las siguientes tres condiciones:

[0130] La primera condición es:

25 Dado m , y k donde $k \leq m$, para la matriz $m \times k$ A donde $A_{ij} \sim N(0, 1/m)$, los valores singulares $s_1(A) \geq \dots \geq s_k(A)$ cumplen la condición:

$$\max \left\{ P\left(s_1(A) \geq 1 + \sqrt{k/m} + t\right), P\left(s_k(A) \leq 1 - \sqrt{k/m} - t\right) \right\} < \exp(-mt^2/2)$$

30 **[0131]** La segunda condición es:

$$P(A_i) \geq 1 - \frac{2}{n} \left(\frac{1}{\sqrt{2\pi \log n}} + \frac{|I_x|}{n} \right) - O\left(\frac{1}{n^{2 \log 2}}\right).$$

35 **[0132]** Por la unión obligada,

$$P\left(\bigcap_{i=1}^n A_i\right) = 1 - P\left(\bigcup_{i=1}^n A_i^c\right) \geq 1 - 2 \left(\frac{1}{\sqrt{2\pi \log n}} + \frac{|I|}{n} \right) - \frac{c}{n^{2 \log 2 - 1}}$$

40 **[0133]** $\hat{v}^{(i)}$ se define como la estimación de Lasso. Se obtiene por mínimos cuadrados aplicados sobre el soporte de Lasso, $\hat{\Omega}^{(i)}$.

[0134] La estimación de LASSO se define, $\hat{\Omega}^{(i)}$, y la estimación se obtienen por mínimos cuadrados aplicados en el soporte de LASSO, $\hat{V}^{(i)}$ como:

$$\hat{v}^{(i)} = \arg \min_{v \in \mathbb{R}^n} \|Av - y^{(i)}\|_2^2 + \lambda \|v\|_1$$

$$\hat{v}^{(i)} = \arg \min_{\substack{v \in \mathbb{R}^n \\ v_{I^c} = 0}} \|Av - y^{(i)}\|_2^2$$

45 donde $I = \text{supp}(\hat{\Omega}^{(i)})$.

50 **[0135]** En el caso donde el soporte en todas las ventanas consecutivas n se detecte correctamente,

$A_1 \cap \dots \cap A_n$, la estimación de mínimos cuadrados ordinarios es

$$\tilde{v}^{(i)} = \left(A_{I^{(i)}}^T A_{I^{(i)}} \right)^{-1} A_{I^{(i)}}^T y^{(i)}.$$

5 **[0136]** La matriz $A_{I^{(i)}}^T A_{I^{(i)}}$ es invertible ya que los valores singulares de $A_{I^{(i)}}$ son distintos de cero con alta probabilidad.

[0137] Desde LSE estándar, $E[\tilde{v}^{(i)}] = v_{I^{(i)}}^{(i)}$ y $Cov[\tilde{v}^{(i)}] = \sigma^2 \left(A_{I^{(i)}}^{(i)T} A_{I^{(i)}}^{(i)} \right)^{-1}$.

10 **[0138]** Para cualquier entrada fija, tal como $v_{n-1} = x_{n-1}$ que es la primera entrada en la señal que se estima n veces, dado $A_0^{n-1} := A_0 \cup \dots \cup A_{n-1}$:

$$\begin{aligned} E_v \left[\left(\frac{1}{n} \sum_{i=0}^{n-1} \tilde{v}_{n-1}^{(i)} - v_{n-1} \right)^2 \middle| A_0^{n-1} \right] &= \frac{1}{n^2} \sum_{i=0}^{n-1} E_v \left[\left(\tilde{v}_{n-1}^{(i)} - v_{n-1} \right)^2 \middle| A_0^{n-1} \right] + \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{\substack{j=0 \\ j \neq i}}^{n-1} E_v \left[\left(\tilde{v}_{n-1}^{(i)} - v_{n-1} \right) \left(\tilde{v}_{n-1}^{(j)} - v_{n-1} \right) \middle| A_0^{n-1} \right] \\ &= \frac{1}{n^2} \sum_{i=0}^{n-1} E_v \left[\left(\tilde{v}_{n-1}^{(i)} - v_{n-1} \right)^2 \middle| A_0^{n-1} \right] = \frac{1}{n^2} \sum_{i=0}^{n-1} \sigma^2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}^T \left(A_{I^{(i)}}^T A_{I^{(i)}} \right)^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &\leq \frac{1}{n^2} \sum_{i=0}^{n-1} \sigma^2 \frac{1}{\sigma_k(A_{I^{(i)}})} \leq \frac{\sigma^2}{n(1 - \sqrt{k/m})} \end{aligned}$$

15 m ~ k log (n/k)

20 **[0139]** Este valor para m disminuye a casi 1/n, por lo tanto, la varianza de reconstrucción llega a cero en el caso en que la detección de soporte tiene éxito en ventanas consecutivas n.

$$\left(\bigcap_{i=1}^n A_i \right)^c.$$

[0140] En tercer lugar, es importante saber qué sucede en el complemento del caso previo, Para $\tilde{v}^{(i)}$ y $\check{v}^{(i)}$ el lema anterior genera:

$$\|A_I(\tilde{v}_I^{(i)} - \check{v}_I^{(i)})\|_2^2 \leq \left(\|\tilde{v}_I^{(i)}\|_1 - \|\check{v}_I^{(i)}\|_1 \right) \frac{\lambda}{2}$$

25 **[0141]** Por lo tanto:

$$\|\tilde{v}_I^{(i)} - \check{v}_I^{(i)}\|_2^2 \leq \left(\|\tilde{v}_I^{(i)}\|_1 - \|\check{v}_I^{(i)}\|_1 \right) \frac{\lambda}{2(1 - \sqrt{k/m})}$$

30 **[0142]** Usando la desigualdad del triángulo obtenemos un límite superior para la norma '2 de la diferencia entre la estimación LSE en el conjunto de soporte y la señal real:

$$\|\tilde{v}^{(i)} - v^{(i)}\|_2^2 \leq \|\tilde{v}^{(i)} - \check{v}^{(i)}\|_2^2 + \|\check{v}^{(i)} - v^{(i)}\|_2^2 \leq \frac{\lambda \left(\|\tilde{v}_I^{(i)}\|_1 - \|\check{v}_I^{(i)}\|_1 \right)}{2(1 - \sqrt{k/m})} + \|\check{v}^{(i)} - v^{(i)}\|_2^2$$

35 **[0143]** Al combinar estos resultados, lo que se obtiene es:

$$E \left[\left(\tilde{v}_{n-1} - v_{n-1} \right)^2 \right] = E \left[\left(\frac{1}{n} \sum_{i=0}^{n-1} \left(\tilde{v}_{n-1}^{(i)} - v_{n-1} \right) \right)^2 \right] \leq E \left[\frac{1}{n} \sum_{i=0}^{n-1} \left(\tilde{v}_{n-1}^{(i)} - v_{n-1} \right)^2 \right] \leq \frac{1}{n} \sum_{i=0}^{n-1} E \left[\|\tilde{v}^{(i)} - v^{(i)}\|_2^2 \right]$$

40 **[0144]** La ecuación anterior se obtiene sustituyendo la desigualdad 17, usando probabilidades para A_0^n y $(A_0^n)^c$, y observando que v, \tilde{v} , y \check{v} se pueden intercambiar con x, \tilde{x} y \check{x} respectivamente.

[0145] A continuación, se analizará la complejidad computacional del algoritmo de detección comprimida recursivo.

5 [0146] Si i es el índice de la ventana, $A^{(i)} \in \mathbb{R}^{m \times n}$ es la matriz de muestreo, y τ es indicativo de la etapa deslizante, el número de ranuras deslizantes entre ventanas sucesivas, al final de la i .ª ventana $n+(i-1)\tau$ entradas se han recuperado.

10 [0147] La primera ventana se muestra por $A^{(0)}x^{(0)}$. Esto requiere $O(mn)$ operaciones básicas, tales como sumas y multiplicaciones. Después de la ventana inicial, el muestreo de la ventana

$$x^{(i)} = [x_{i\tau} x_{i\tau+1} \dots x_{i\tau+n-1}]$$

15 se logra mediante muestreo recursivo con actualización de rango τ con complejidad $O(m\tau)$. Por lo tanto, al final de la i .ª ventana, la complejidad total del muestreo es $O(mn) + O(m\tau)i$, para una complejidad promedio de $O(m\tau)$ para muestreo recursivo.

20 [0148] La otra contribución a la complejidad computacional proviene del solucionador iterativo. La complejidad esperada atribuye al solucionador iterativo que se puede calcular como el número de operaciones en cada iteración multiplicado por el número esperado de iteraciones para la convergencia. Mientras que el primero depende del algoritmo particular, el segundo es una función de la distancia del punto de partida a la solución óptima, que está relacionado con el uso de la estimación recursiva como sigue:

[0149] Usando:

25
$$\hat{x}_{[0]}^{(i)} = [x_{i\tau}^{(i-1)} \dots x_{i\tau+n-1}^{(i-1)} 0^T]^T$$

[0150] Como punto de partida, lo que se obtiene es:

30
$$\|\hat{x}_{[0]}^{(i)} - x_*^{(i)}\|_2 \leq C_0 \|x^{(i-1)} - x_k^{(i-1)}\|_1 / \sqrt{k} + C_0 \|x^{(i)} - x_k^{(i)}\|_1 / \sqrt{k} + C_1 \tilde{\sigma} + \|[x_{n-\tau}^{(i)} \dots x_{n-1}^{(i)}]\|_2$$

[0151] La validez de la declaración hecha anteriormente se demuestra como sigue:

35
$$e^{(i)} := [x_{i\tau}^{(i-1)} \dots x_{i\tau+n-1}^{(i-1)} 0^T]^T - [x_{*0}^{(i)} \dots x_{*n-1}^{(i)}]^T$$

$$e^{(i)} := x_*^{(i)} - x^{(i)}$$

$$e^{(i+1)} = [x_{i\tau}^{(i)} \dots x_{i\tau+n-1}^{(i)} 0^T]^T - x^{(i)} + x^{(i)} - [x_{*0}^{(i)} \dots x_{*n-1}^{(i)}]^T$$

40 [0152] Al tomar la norma y usar la desigualdad triangular se genera:

$$\|e^{(i+1)}\|_2 \leq \|e^{(i-1)}\|_2 + \|e^{(i)}\|_2 + \|[x_{n-\tau}^{(i)} \dots x_{n-1}^{(i)}]\|_2$$

[0153] Con vista al teorema que proporciona el error de Lasso, lo que se obtiene es:

45
$$\|e^{(i-1)}\|_2 \leq C_0 \|x^{(i-1)} - x_k^{(i-1)}\|_1 / \sqrt{k} + C_0 \|x^{(i)} - x_k^{(i)}\|_1 / \sqrt{k} + C_1 \tilde{\sigma} + \|[x_{n-\tau}^{(i)} \dots x_{n-1}^{(i)}]\|_2 \quad (18)$$

50 [0154] La complejidad computacional exacta de cada iteración depende del algoritmo. Como mínimo, el solucionador iterativo para LASSO requiere la multiplicación de la matriz de muestreo y la estimación en la iteración que requiere operaciones $O(mn)$. En un algoritmo donde la función de coste se descompone sublinealmente (por ejemplo, $1/t^2$), como en FISTA, el número de iteraciones necesarias para obtener \hat{x} de modo que $G(\hat{x}) - G(x^*) \leq \epsilon$

donde x^* es la solución óptima, es proporcional a $\frac{\|x_{[0]} - x_*\|_2}{\sqrt{\epsilon}}$, donde x es el punto de partida del algoritmo. A partir de este límite, se ve que el número promedio de iteraciones es proporcional a la distancia Euclidiana del punto inicial del algoritmo desde el punto óptimo.

[0155] Para la secuencia $\{x_i\}_{i=0,1,\dots}$ donde $\|x^{(0)}\|_0 \leq k$ y $\max_{j=0,\dots,n-1} |x_j^{(i)}| \leq C$ para todo i donde $C > 0$ es constante, el número esperado de iteraciones para la convergencia de algoritmos donde la función de coste se descompone sublinealmente es $O(\sqrt{\tau})$ para mediciones silenciosas y $O(\sqrt{\tau}) + O(\sqrt{m})$ para el ruido de medición i.i.d.

5 **[0156]** La afirmación anterior es verdadera ya que: Ya que $x^{(0)}$ es k -disperso, los términos $\|x^{(i-1)} - x_k^{(i-1)}\|_1$ y $\|x^{(i)} - x_k^{(i)}\|_1$ se descartan de acuerdo con la ecuación 18.

[0157] Por $|x_i| \leq C$, se obtiene

10
$$\| [x_{n-\tau}^{(i)} \dots x_{n-1}^{(i)}] \|_2 \leq C\sqrt{\tau}.$$

[0158] Con mediciones ruidosas, el término $C_i \tilde{\sigma}$ está relacionado con el nivel de ruido. Como el ruido tiene reparto $N(0, \sigma^2 I)$, con alta probabilidad $\tilde{\sigma}^2 = \sigma^2 (m + \lambda \sqrt{2m})$ por lo tanto $\tilde{\sigma} = O(\sqrt{m})$ por las condiciones avanzan por el teorema relacionado con el error en LASSO. Usando este resultado en la ecuación (18), se observa que el número esperado de iteraciones es $O(\sqrt{\tau}) + O(\sqrt{m})$ en el caso ruidoso.

15 **[0159]** En el análisis anterior, la complejidad promedio es $O(m\tau)$ para muestreo recursivo y $O(mn)$ para algoritmo de optimización (mínimamente). Para recuperar cada componente del flujo de entrada y no omitir elementos, $\tau \leq n$, por lo tanto se ve que la complejidad de la optimización iterativa domina la complejidad promedio.

20 **[0160]** Usando la condición $m \geq Ck \log(n/k)$ para vectores aleatorios muestreados desde la esfera de la unidad, las matrices gaussianas aleatorias y las matrices de Bernoulli aleatorias cumplen la condición $\delta_{2k} \leq \sqrt{2} - 1$. La siguiente tabla 1 muestra la complejidad computacional promedio para varias clases de dispersión en k .

| k | Complejidad computacional |
|---------------|--------------------------------------|
| $O(1)$ | $O(n \log n)$ |
| $O(\log n)$ | $O(n \log n \cdot \log(n / \log n))$ |
| $O(\sqrt{n})$ | $O(n^{3/2} \log n)$ |
| $O(n)$ | $O(n^2)$ |

25 **RESULTADOS EXPERIMENTALES**

[0161] Los datos utilizados en las simulaciones se generan a partir del modelo aleatorio:

30
$$f_x(x) = \begin{cases} (1-p)\delta(x) + \frac{1}{2p} & \text{if } x \in [-1, 1] \\ 0 & \text{else} \end{cases} \quad (19)$$

con $p = 0,05$ a menos que se establezca lo contrario.

35 **[0162]** El modelo de medición es

$$y^{(i)} = A^{(i)} x^{(i)} + w^{(i)}$$

con $w^{(i)} \sim N(0, \sigma^2 I)$

40 donde $\sigma > 0$

y la matriz de muestreo es

$$A^{(0)} \in R^{m \times n}$$

5 donde m es seis veces el número esperado de entradas distintas de cero en una ventana, es decir, $m = 6pn$, y donde n es igual a la longitud de la ventana.

10 **[0163]** Se ha demostrado experimentalmente cuál es la ganancia de velocidad lograda por RCS comparando el tiempo promedio requerido para estimar una ventana dada mientras se usa FISTA, el procedimiento de gradiente proximal acelerado, para resolver LASSO. RCS se compara con el llamado 'enfoque ingenuo', donde el muestreo se realiza mediante la multiplicación de matrices en cada ventana y FISTA se inicia desde todos los vectores cero. El tiempo promedio requerido para recuperar una ventana en cada caso se muestra en la figura 2. Como se puede observar en la figura 2, el tiempo de procesamiento promedio para RCS está por debajo del tiempo de procesamiento requerido por otros procedimientos.

15 **[0164]** Los resultados de los experimentos en la estimación de soporte que usan LASSO se analizan a continuación. En las mediciones $x \in R^{6000}$, $\|x\|_0 = 60$, $A \in R^{m \times 6000}$ se genera por una distribución gaussiana con $A_{ij} \sim N(0, 1/m)$ y w tiene $\sigma = 0,1$. Como se sugiere en el teorema de LASSO, para estos parámetros, LASSO se resuelve con $\lambda = 4\sigma\sqrt{2 \log n}$, y las amplitudes distintas de cero de x cumplen $\min_{i=1,2,\dots,n} |x_i| \geq 3.34$ por muestreo de U $([-3,34, -4:34] \cup [3,34, 4,34])$.

20 **[0165]** En las simulaciones, el número de muestras tomadas varía de la señal, m, y la precisión de la estimación del soporte se estudia usando la tasa de detección igual a $|\text{soporte detectado} \cap \text{soporte verdadero}| / |\text{soporte verdadero}|$ y una tasa de falsos positivos = $|\text{soporte detectado} \setminus \text{soporte verdadero}| / |\text{soporte verdadero}|$, donde $|\cdot|$ indica la cardinalidad de un conjunto y \setminus es el operador de diferencia de conjunto. Debe tenerse en cuenta que con esta definición, la tasa de falsos positivos puede ser mayor que uno.

25 **[0166]** Se comparan dos procedimientos para la detección de soporte. El primer procedimiento es resolver LASSO y tomar las posiciones distintas de cero como soporte, y el segundo es tomar las posiciones donde la magnitud absoluta de la estimación es mayor que un umbral. La figura 3 muestra las curvas resultantes para los dos procedimientos, obtenidas generando aleatoriamente la señal de entrada 20 veces para cada valor de m y promediando las tasas de detección resultantes y los falsos positivos. Como se puede observar en la figura, aunque la tasa de detección se comporta de manera similar en ambos procedimientos, los falsos positivos se pueden reducir significativamente ajustando adecuadamente el umbral en las estimaciones de LASSO resultantes.

30 **[0167]** Lo que se ilustra específicamente en la Fig. 3 es la estimación del conjunto de soporte usando LASSO: para $n = 6000$, $\sigma=0.1$, $\min |x_i| \geq 3.34$, umbral $\xi_i = 0,01, 0,10$ y $1,00$. Los marcadores de círculo representan la tasa de detección, y los marcadores cuadrados representan la tasa de falsos positivos.

35 **[0168]** Como se analiza en detalle en lo anterior, el procedimiento de LASSO se puede usar junto con una estrategia de votación y una estimación de mínimos cuadrados para proporcionar un estimador imparcial. La figura 4 muestra la comparación del desempeño de a) estimaciones de LASSO individuales, b) estimaciones promediadas, c) estrategia de votación, y d) desesgado y cálculo de promedio.

40 **[0169]** La figura se obtiene usando un $x \in R^n$ fijo y tomando mediciones con ruido no correlacionado. Se puede ver que el error no disminuye a cero para la estimación promedio, debido a que LASSO es un estimador sesgado, como se explicó anteriormente en este documento. Específicamente, la Fig. 4 es una representación de gráficas de error para a) estimaciones promediadas, b) estimaciones desesgadas y promediadas y c) estimaciones obtenidas mediante votación y cálculo de promedio.

45 **[0170]** Con referencia ahora a la Fig. 5, la figura ilustra las gráficas de error para la estimación promedio de LASSO y 'la votación y cálculo de promedio' en datos de transmisión continua.

50 **[0171]** La figura 5 muestra una comparación entre el error de reconstrucción de la ventana obtenido con las estimaciones promedio de LASSO y el algoritmo de 'votación y cálculo de promedio' en datos de transmisión continua. Los datos se generan aleatoriamente cuando se selecciona una entrada como soporte con $p = 0,05$ y su amplitud se dibuja independientemente y se reparte idénticamente desde un reparto uniforme

55

$$60 \quad U([-1.3847, -2.3847] \cup [1.3847, 2.3847])$$

para satisfacer las condiciones del teorema de detección de soporte, en el que $n = 400$, $m = 5np = 100$ y la matriz de muestreo es gaussiana. Las mediciones están contaminadas con

$$W^{(i)} \sim N(0, \sigma^2 I_m)$$

donde $\sigma = 0,05$, y el algoritmo de votación usa $\xi_1 = 0,5$ y $\xi_2 = 20$. Como se puede observar, la votación da un error de reconstrucción que muestra saltos debido a la espera de la recolección de votos ξ_2 para usar una entrada en LSE. Sin embargo, también se puede ver que después de que la posición se acepta como parte del soporte, el error cae instantáneamente a valores menores en vez de simplemente promediar las estimaciones de LASSO.

[0172] Por lo tanto, lo que se propone de acuerdo con un modo de realización es un procedimiento eficiente para el muestreo recursivo y la recuperación iterativa pertinente a la detección comprimida en datos de transmisión continua. El procedimiento potencia las superposiciones de señal entre ventanas de procesamiento sucesivas para obtener una velocidad de convergencia más rápida para las estimaciones de la señal mientras se logra una reducción de la variación de la estimación en presencia de ruido.

[0173] Además, de acuerdo con otro modo de realización, se propone un procedimiento de estimación en dos etapas para aproximar un estimador imparcial de la señal basado en LASSO donde a) la detección de soporte se realiza resolviendo LASSO, y b) la estimación de señal se obtiene resolviendo mínimos cuadrados ordinarios en el conjunto de soporte estimado. La complejidad computacional del algoritmo es $O(mn)$ donde m es el número de muestras tomadas y n es la longitud de la ventana. Los experimentos muestran que el tiempo de convergencia es apropiado para la implementación en línea en datos de transmisión continua.

[0174] Por lo tanto, para resumir, los modos de realización incluyen un procedimiento para la detección comprimida de datos de transmisión continua que implica realizar un esquema recursivo para realizar la detección comprimida de datos de transmisión continua, que también es capaz de analizar la complejidad computacional y el error de estimación del procedimiento. De acuerdo con un modo de realización, el flujo de entrada de datos se muestrea de forma recursiva por medio de ventanas superpuestas mientras se hace uso de la medición previa para obtener la siguiente. Al potenciar la estructura de ventana superpuesta, se usa la estimación de la señal de la ventana anterior, para lograr una convergencia más rápida en un algoritmo de optimización iterativo, para decodificar la nueva ventana. La exactitud de la estimación se mejora promediando las estimaciones de señal obtenidas a partir de múltiples ventanas. Para eliminar el sesgo del estimador, se propone un procedimiento de estimación de dos etapas de acuerdo con un modo de realización que comprende detección de conjunto de soporte y estimación de amplitud de señal. Además, un modo de realización incluye un esquema de votación para una estimación robusta del soporte en presencia de ruido de medición elevado.

[0175] Los resultados de la simulación obtenidos mientras se emplean los medios de acuerdo con un modo de realización para la detección comprimida de datos de transmisión continua muestran una velocidad de diez veces con respecto a la aplicación de la detección comprimida tradicional en un flujo de datos, mientras se obtiene un error de reconstrucción significativamente menor en afecciones leves en las magnitudes de la señal y el nivel de ruido.

[0176] De acuerdo con otro modo de realización, también se proponen medios para detectar datos de transmisión continua. Dichos medios para detectar datos de transmisión continua comprenden medios para muestrear recursivamente un flujo de entrada de datos, y medios para emplear mediciones previas para obtener una medición posterior. Los medios para muestrear recursivamente un flujo de datos de entrada son capaces de usar las etapas de superposición de ventanas para obtener al menos una medición previa.

[0177] Un modo de realización incluye el rango con señales de banda ultraancha. Por ejemplo, un dispositivo puede monitorear continuamente los pulsos de retorno a picos transmitidos regularmente (la señal emitida es periódica). En general, el dispositivo recibe un eco principal, además de múltiples ecos agregados. Si el dispositivo se mueve, el eco principal cambia ligeramente de un período a otro, al igual que los ecos múltiples. Esta no es una señal periódica, solo una señal casi periódica, y un algoritmo de ventana deslizante rastreará naturalmente estos ecos cambiantes. En dichos modos de realización, los datos de eco de un sensor acústico se transforman de este modo en datos de intervalo.

[0178] En otro ejemplo, los datos de muestras biomédicas pueden transformarse de acuerdo con los procedimientos y sistemas descritos anteriormente. Por ejemplo, las muestras de uno o más sensores eléctricos configurados para recibir señales eléctricas de un cuerpo humano pueden procesarse de acuerdo con los modos de realización descritos en el presente documento para reconstruir señales electrocardiográficas o electroencefalográficas. En algunos modos de realización, la señal reconstruida puede coincidir con las firmas características de una o más afecciones de diagnóstico.

[0179] En otro ejemplo, los datos de muestra de un sensor de cámara, fotoeléctrico u otro sensor de conjunto de píxeles (por ejemplo, configurado para recibir luz visible o luz infrarroja) se pueden procesar como se describe en el presente documento. Por ejemplo, usando dichas técnicas, un aparato puede mejorar la eficiencia de potencia o incrementar la sensibilidad efectiva del sensor.

[0180] En otros modos de realización, los datos del sensor ambiental tales como la temperatura, la velocidad del viento, la dirección del viento, la precipitación pueden procesarse como se describe en el presente documento.

5 **[0181]** Los medios de la presente invención pueden implementarse como medios de programa informático, medios de hardware o una combinación de los mismos.

[0182] Los sistemas y procedimientos descritos anteriormente pueden realizarse, por ejemplo, en un módem inalámbrico o dispositivo inalámbrico que integra dicho módem.

10 **[0183]** Debería entenderse que dependiendo del modo de realización, determinadas acciones o eventos de cualquiera de los procedimientos descritos en el presente documento se pueden realizar en una secuencia distinta, pueden añadirse, fundirse u omitirse por completo (por ejemplo, no todas las acciones o eventos descritos son necesarios para la puesta en práctica del procedimiento). Además, en determinados modos de realización, las acciones o eventos se pueden realizar simultáneamente, por ejemplo, a través de procesamientos de múltiples subprocesos, procesamientos de interrupciones o múltiples procesadores, en lugar de secuencialmente.

15 **[0184]** Los expertos reconocerán que los diversos bloques lógicos, módulos, circuitos y etapas de algoritmo ilustrativos descritos en relación con los procedimientos, sistemas y aparatos divulgados en el presente documento se pueden implementar como hardware electrónico, programa informático ejecutado por un procesador o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, anteriormente se han descrito diversos componentes, bloques, módulos, circuitos y etapas ilustrativos, en general, en lo que respecta a su funcionalidad. Que dicha funcionalidad se implemente como hardware o software depende de la aplicación particular y de las restricciones de diseño impuestas en el sistema general. Los expertos en la técnica pueden implementar la funcionalidad descrita de formas distintas para cada solicitud particular, pero no debería interpretarse que dichas decisiones de implementación suponen apartarse del alcance de la presente invención.

20 **[0185]** Además, los modos de realización divulgados en el presente documento pueden implementarse o realizarse con un dispositivo electrónico o circuito tal como un procesador de uso general, un procesador de señales digitales (DSP), un circuito integrado específico de la aplicación (ASIC), un conjunto de puertas programables (FPGA) u otro dispositivo de lógica programable, lógica de transistor o de puertas discretas, componentes de hardware discretos o cualquier combinación de los mismos diseñada para realizar las funciones descritas en el presente documento. Un procesador de uso general puede ser un microprocesador pero, como alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador o máquina de estados convencional. Un procesador también puede implementarse como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores junto con un núcleo de DSP o cualquier otra configuración de este tipo.

30 **[0186]** Las etapas de un procedimiento o algoritmo descrito en relación con los modos de realización divulgados en el presente documento se pueden realizar directamente en hardware, en un módulo de software ejecutado por un procesador o en una combinación de los dos. Un módulo de software puede residir en una memoria RAM, en una memoria flash, en una memoria ROM, en una memoria EPROM, en una memoria EEPROM, en registros, en un disco duro, en un disco extraíble, en un CD-ROM o en cualquier otro medio de almacenamiento conocido en la técnica. Un medio de almacenamiento a modo de ejemplo está acoplado al procesador de manera que el procesador puede leer información de, y escribir información en, el medio de almacenamiento. De forma alternativa, el medio de almacenamiento puede estar integrado en el procesador. El procesador y el medio de almacenamiento pueden residir en un ASIC. El ASIC puede residir en un terminal de usuario. De forma alternativa, el procesador y el medio de almacenamiento pueden residir como componentes discretos en un terminal de usuario.

40 **[0187]** Específicamente de acuerdo con un modo de realización de la presente invención, un aparato para realizar la detección comprimida de datos de transmisión continua, comprende una muestra recursiva dispuesta para muestrear recursivamente un flujo de entrada de datos usando ventanas superpuestas para obtener al menos una medición previa con respecto al flujo de datos de entrada, y una unidad que emplea al menos una medición previa para obtener una medición posterior.

50 **[0188]** El aparato propuesto de acuerdo con un modo de realización de la presente invención comprende una muestra recursiva que se basa en un esquema recursivo para realizar la detección. El aparato para realizar la detección comprimida de datos de transmisión continua de la presente invención también comprende una unidad de procesamiento para procesar el flujo de entrada de datos muestreados durante la etapa de muestreo recursivo por medio de estimación recursiva.

60 **[0189]** De acuerdo con la presente invención, el aparato para realizar la detección comprimida de datos de transmisión continua también comprende medios de almacenamiento para entregar la información de entrada con respecto al flujo de datos de una estimación previa obtenida durante una etapa de obtención de estimación previa. Además, el aparato para realizar la detección comprimida de datos de transmisión continua de la presente invención comprende un contador dispuesto para realizar la estimación de recuento en base a la información obtenida durante una etapa de detección de soporte de flujo de datos. Aún más, el aparato de la presente invención también puede

5 comprender una calculadora para calcular un valor de estimación de mínimos cuadrados (LSE) para un conjunto de soporte de flujo de datos basado en los datos obtenidos durante dicha etapa de estimación recursiva. Una unidad de procesamiento para promediar el valor de estimación de mínimos cuadrados calculado, el valor de estimación de recuento y la estimación previa para calcular un valor promedio para obtener un valor promedio también está comprendido por el aparato propuesto de acuerdo con un modo de realización de la presente invención. También se prevé que un estimador para estimar un grado de error del procedimiento para detectar datos de transmisión continua esté comprendido en el aparato propuesto de acuerdo con la presente invención. Además, una unidad de procesamiento para promediar estimaciones de señal obtenidas de una pluralidad de ventanas también puede estar comprendido en el aparato propuesto por la presente invención.

10 **[0190]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento para realizar la detección comprimida de datos de transmisión continua, que comprende:

5 muestreo recursivo de un flujo de entrada de datos usando ventanas superpuestas sucesivas, al menos en parte:

seleccionando recursivamente una secuencia de matrices de detección A para las ventanas superpuestas en el flujo de entrada de datos, definiéndose la matriz de detección $A^{(i+1)}$ para la ventana superpuesta $(i-2)^{a}$ por la siguiente ecuación: $A^{(i+1)} = [a_2^{(i)}, a_3^{(i)}, \dots, a_n^{(i)}, a_1^{(i)}]$, donde $a_L^{(i)}$ es la L.^a columna de la matriz de detección $A^{(i)}$, n es la longitud de cada ventana superpuesta, e i es mayor o igual a 0; y

10

obteniendo una medición $y^{(i+1)}$ para la $(i+2)^{a}$ ventana superpuesta usando la siguiente ecuación: $y^{(i+1)} = y^{(i)} + (x_{i+n} - x_i)a_1^{(i)}$, donde i es mayor o igual a 0 y la señal $x^{(i)}$ del flujo de entrada de datos correspondiente a la $(i+1)^{a}$ ventana superpuesta se define mediante la siguiente ecuación: $x^{(i)} = [x_i \ x_{i+1} \ \dots \ x_{i+n-1}]^T$, y la medición $y^{(0)}$ para la ventana superpuesta inicial se define mediante la siguiente ecuación: $y^{(0)} = A^{(0)}x^{(0)}$.

15

2. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, en el que la etapa de obtener la medición $y^{(i+1)}$ se realiza sin realizar una multiplicación matricial que involucre una de las matrices de detección A.

20

3. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, en el que la etapa de obtener la medición $y^{(i)}$ comprende procesar dicho flujo de entrada de datos muestreados durante la etapa de muestreo recursivo por medio de la estimación recursiva.

25

4. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 3, que comprende además:

recibir información con respecto a dicho flujo de datos de entrada, en el que dicha información se refiere a una estimación previa obtenida durante una etapa de obtención de estimación previa, realizándose dicha etapa de obtención de estimación previa antes de la etapa de muestreo recursivo.

30

5. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 3, que comprende además:

35

detectar un soporte de flujo de datos, y

realizar una estimación de conteo basada en la información obtenida durante la etapa de detección de soporte de flujo de datos.

40

6. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 3, que comprende además calcular un valor de estimación de mínimos cuadrados (LSE) para un conjunto de soporte de flujo de datos basado en los datos obtenidos durante dicha etapa de estimación recursiva.

45

7. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 6, que comprende además promediar dicho valor de estimación de mínimos cuadrados calculado, dicho valor de estimación de conteo y dicha estimación previa para calcular un valor promediado.

50

8. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 7, que comprende además emplear dicho valor promedio para obtener una nueva estimación para dichos datos de transmisión continua.

55

9. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, que comprende además analizar una complejidad computacional de dicha detección comprimida de datos de transmisión continua.

60

10. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 9, que comprende además estimar un grado de error de dicho procedimiento para detectar datos de transmisión continua.

65

11. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, que comprende además obtener la convergencia en un algoritmo de optimización iterativo para decodificar una nueva ventana, en el que dicha etapa de obtención comprende potenciar una estructura de ventana superpuesta empleada por la etapa de usar ventanas superpuestas sucesivas y una estimación de señal con respecto a una ventana superpuesta previa.

12. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, que comprende además promediar estimaciones de señal obtenidas de las ventanas superpuestas sucesivas, en el que realizar la detección comprimida de datos de transmisión continua comprende:

5 realizar la detección de conjunto de soporte, y
 estimación de amplitud de señal.

10 13. El procedimiento para realizar la detección comprimida de datos de transmisión continua de la reivindicación 1, que comprende además aplicar un esquema de votación para una estimación de soporte robusta en presencia de un ruido de medición alto.

15 14. Un medio de almacenamiento legible por ordenador no transitorio que ha registrado en el mismo un programa informático para detectar datos de transmisión continua, comprendiendo el programa informático un algoritmo capaz de realizar el procedimiento de acuerdo con cualquiera de las reivindicaciones precedentes.

15 15. Un aparato para realizar la detección comprimida de datos de transmisión continua, que comprende:

20 una memoria configurada para almacenar datos de un flujo de entrada de datos del sensor; y
 un procesador configurado para:

 muestrear recursivamente un flujo de entrada de datos del sensor usando ventanas superpuestas sucesivas, al menos en parte:

25 seleccionando recursivamente una secuencia de matrices de detección A para las ventanas superpuestas en el flujo de entrada de datos del sensor, definiéndose la matriz de detección $A^{(i+1)}$ para la $(i+2)$.ª ventana superpuesta mediante la siguiente ecuación: $A^{(i+1)} = [a_2^{(i)}, a_3^{(i)}, \dots, a_n^{(i)}, a_1^{(i)}]$, donde $a_L^{(i)}$ es la L.ª columna de la matriz de detección $A^{(i)}$, n es la longitud de cada ventana superpuesta, e i es mayor o igual a 0; y

30 obteniendo una medición $y^{(i+1)}$ para la $(i+2)$.ª ventana superpuesta usando la siguiente ecuación: $y^{(i+1)} = y^{(i)} + (x_{i+n} - x_i)a_1^{(i)}$, donde i es mayor o igual a 0 y la señal $x^{(i)}$ del flujo de entrada de datos del sensor correspondiente a la $(i+1)$.ª ventana superpuesta se define mediante la siguiente ecuación: $x^{(i)} = [x_i \ x_{i+1} \dots \ x_{i+n-1}]^T$, y la medición $y^{(0)}$ para la ventana superpuesta inicial se define mediante la siguiente ecuación: $y^{(0)} = A^{(0)}x^{(0)}$.

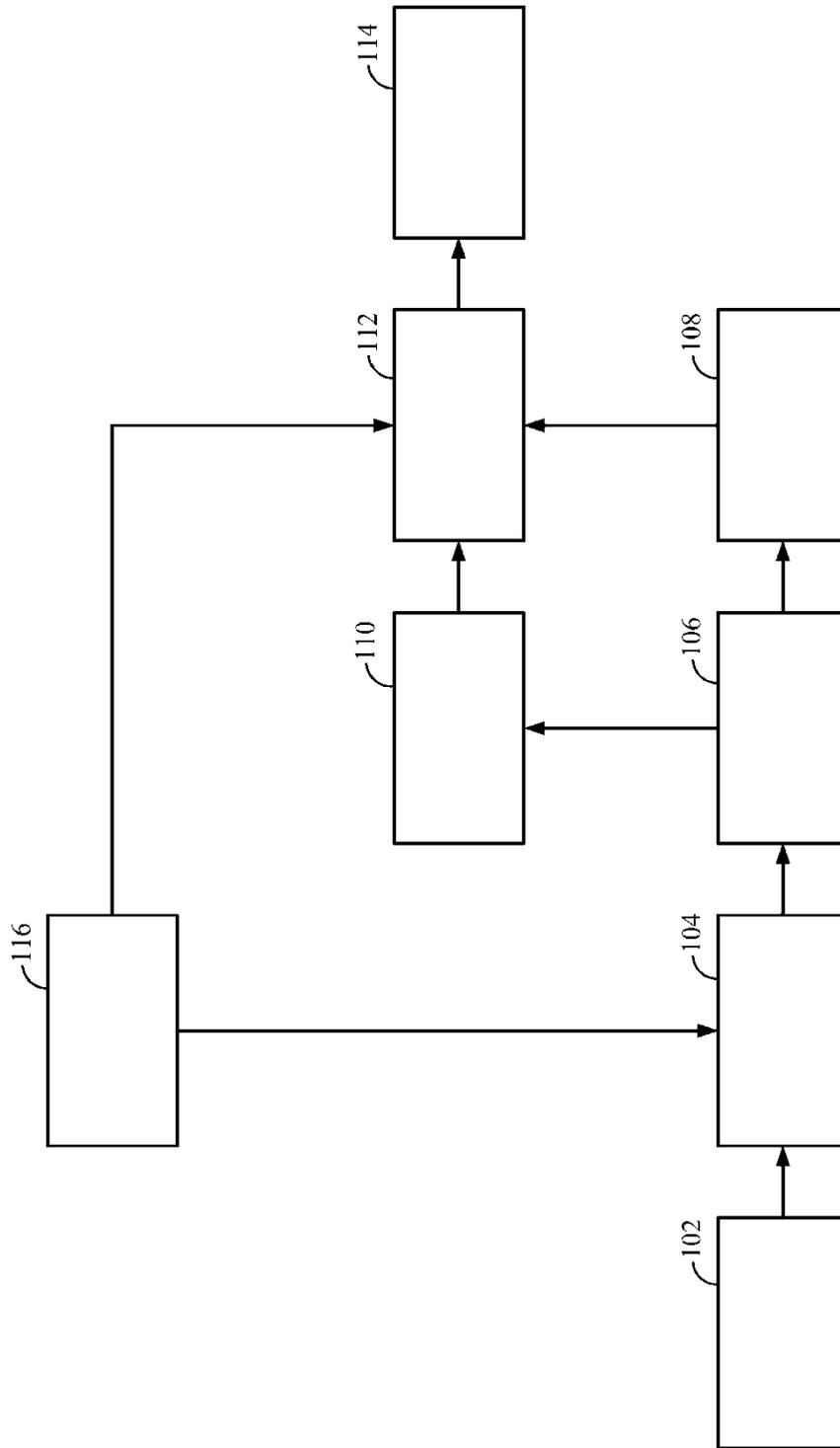


FIG. 1

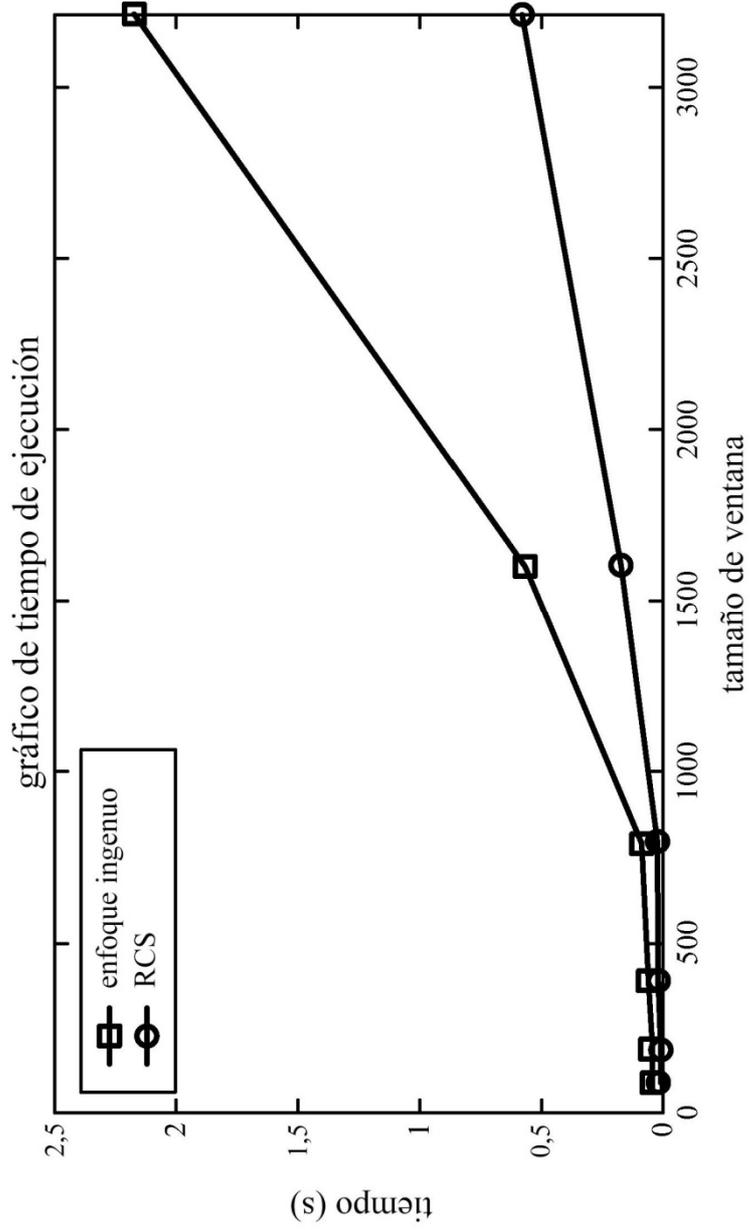


FIG. 2

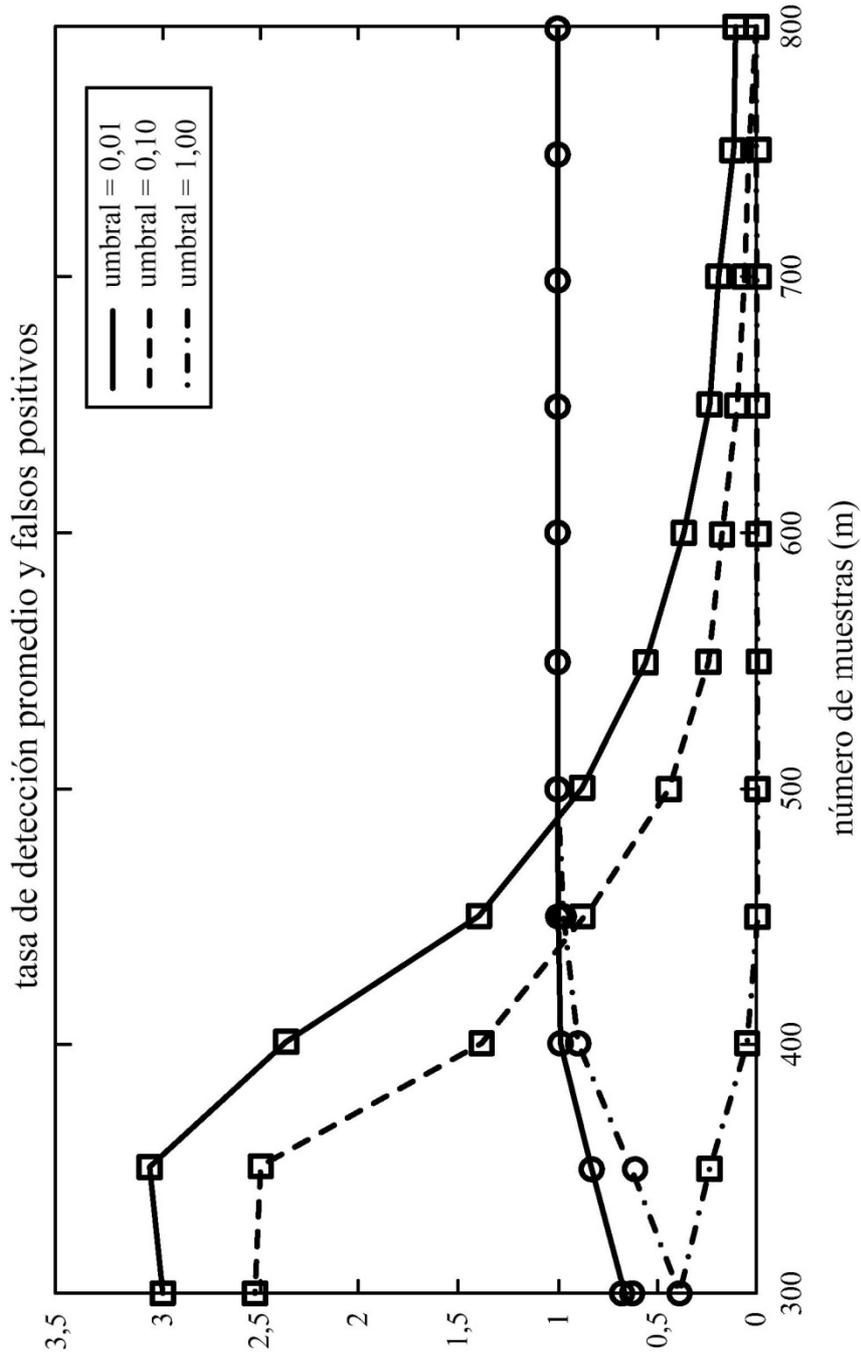


FIG. 3

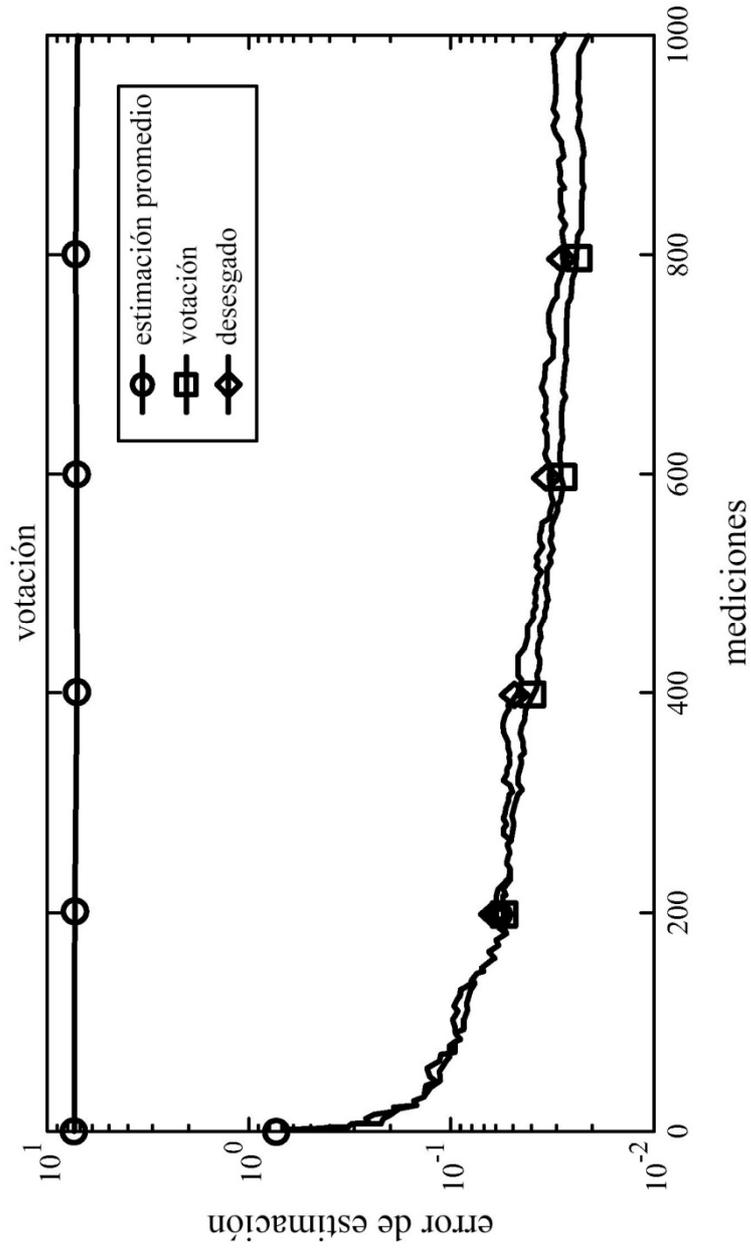


FIG. 4

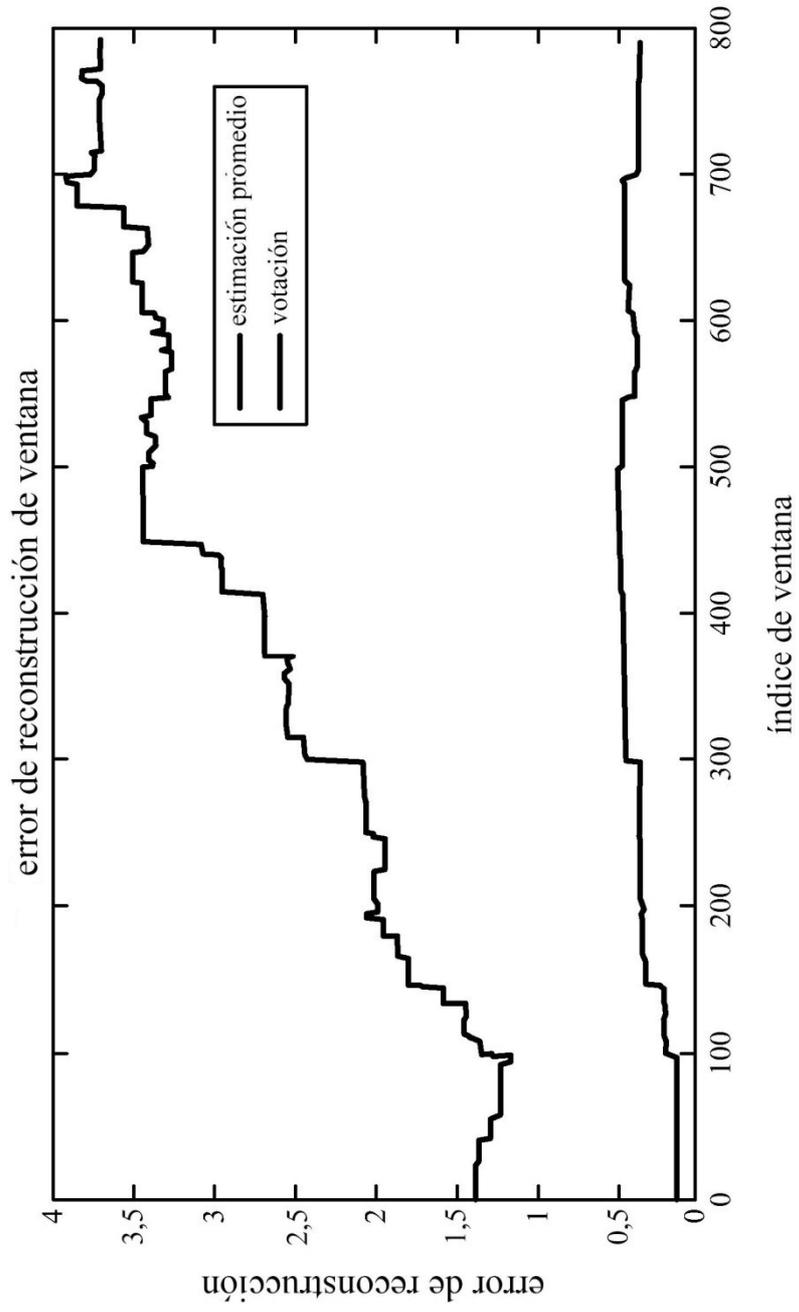


FIG. 5