

3 Algoritmos de reconstrucción

En el apartado anterior vimos que la señal original x es la única solución al problema de minimización $\min \|s\|_1$ sujeto a $y = \Phi s$. Resolver este problema requiere de elevados recursos de cálculo, lo cual lo hace poco adecuado para el uso en aplicaciones en tiempo real, generando la necesidad de encontrar métodos que puedan resolver este problema de forma más eficiente.

En este capítulo se desarrollan dos algoritmos iterativos de reconstrucción de señales sparse que se emplean en la teoría del Compressive Sensing, Matching Pursuit y Orthogonal Matching Pursuit. Se presentarán sus diagramas de flujo así como una explicación detallada de los pasos a seguir para cada algoritmo. En los apéndices podemos encontrar los algoritmos programados en Matlab que se han usado para las simulaciones que se presentan en el proyecto.

Supongamos x una señal real de dimensión n con al menos k componentes distintas de cero, esta señal se denomina k -sparse. El conjunto $\{y_1, \dots, y_m\}$ es una secuencia de vectores de mediciones en \mathbb{R}^n , usamos estos vectores para obtener m medidas de la señal

$$(44) \quad \langle x, y_1 \rangle, \langle x, y_2 \rangle, \dots, \langle x, y_m \rangle$$

donde $\langle \cdot, \cdot \rangle$ denota producto interno. El problema de la recuperación de la señal nos plantea dos preguntas:

- 1) ¿Cuántas mediciones son necesarias para reconstruir la señal?
- 2) Dadas estas mediciones, ¿qué algoritmos realizan la tarea de reconstrucción?

A la primera pregunta podemos responder inmediatamente que serán necesarias no menos de k mediciones. Incluso si las mediciones estuvieran adaptadas a la señal, todavía podría tomar k piezas de información para determinar las componentes distintas de cero de una señal k -sparse. En la otra dirección, n mediciones no adaptativas siempre serán suficientes ya que podemos simplemente listar las n componentes de la señal

original. Esto no es obvio, ya que las señales sparse pueden ser recuperadas con mucha menos información.

En el artículo [11] encontramos una breve reseña histórica. El método para recuperación de señales sparse tiene su origen en La Segunda Guerra Mundial. La U.S. Army tenía un gran interés en la detección de soldados que padecían la sífilis, pero los test para detectar esta enfermedad eran muy caros, y la Armada se dio cuenta de que era inútil realizar ensayos individuales para detectar un caso ocasional. Su solución fue acumular sangre de grupos de soldados y testear esa sangre acumulada. Si uno de los lotes daba positivo, se realizaban pruebas adicionales. Este método, llamado “pruebas de grupo”, se estudió posteriormente en campos como la informática o la estadística.

La idea es la siguiente, combinando aleatoriamente las entradas de la señal sparse, es posible generar un resumen estadístico que nos permita identificar las componentes de la señal que son distintas de cero. El siguiente teorema extraído del documento de Candès-Tao [16] y Rudelson-Vershynin [17], describe un ejemplo de este fenómeno:

Teorema 4([12]): Sea $m \geq Ck \ln(n/k)$, tomamos m vectores $\{y_1, \dots, y_m\}$ independientes pertenecientes a una distribución Gaussiana en \mathbb{R}^n . La siguiente afirmación es cierta con una probabilidad superior a $1 - e^{-Cm}$. Es posible reconstruir cada señal k -sparse $x \in \mathbb{R}^n$ a partir de los datos $\{\langle x, y_i \rangle : i = 1, 2, \dots, m\}$.

En apartados anteriores vimos que la señal original x es la única solución al problema de minimización:

$$\min_{\tilde{x} \in \mathbb{R}^n} \|\tilde{x}\|_{\ell_1} \quad \text{sujeto a } \Phi\tilde{x} = y$$

en este caso, sacrificando el número de medidas y utilizando un método de resolución del problema de minimización de la norma ℓ_1 (Basis Pursuit), se requieren $m = ck$ mediciones, donde k es el nivel de escasez y c es un factor de medición siempre mayor que 1. Esto requiere de elevados recursos de cálculo, lo cual lo hace poco adecuado para el uso en aplicaciones en tiempo real, lo que sugiere una respuesta a la segunda pregunta sobre los algoritmos de reconstrucción de señales sparse, generando la necesidad de encontrar métodos que puedan resolver este problema más eficientemente.

3.1 Matching Pursuit (MP)

MP es un algoritmo iterativo usado para la descomposición de una señal en funciones bases de un diccionario. Su objetivo principal es encontrar las componentes del diccionario que forman parte de la señal.

Cualquier señal en \mathbb{R}^n puede ser representada en términos de una base de vectores $\{\Psi_i\}_{i=1}^n$. Usando la matriz $n \times n$ $\Psi = [\Psi_1, \Psi_2, \dots, \Psi_n]$, con los vectores Ψ_i como columnas, la señal x puede expresarse como:

$$(45) \quad \mathbf{x} = \sum_{i=1}^n s_i \Psi_i = \mathbf{s} \Psi$$

Si el diccionario Ψ es una base ortonormal, entonces los coeficientes s_i se obtienen simplemente del producto interno de las funciones del diccionario Ψ_i y la señal:

$$(46) \quad s_i = \langle \mathbf{x}, \Psi_i \rangle = \Psi_i^T \mathbf{x}$$

El algoritmo MP es un algoritmo iterativo que trata de encontrar el átomo del diccionario holográfico que aporta mayor contribución a la definición de la señal proyectada, encuentra el estimado de la contribución de dicho átomo a la conformación de la señal y remueve dicha contribución que este aporta a la estructura del conjunto de mediciones de la señal, definiendo así un residuo. Se procede iterativamente hasta que el criterio de parada se cumpla, ya sea que se exceda el número de iteraciones posibles o que la norma ℓ_2 del residuo sea menor a cierto valor definido.

3.1.1 Diseño del algoritmo MP

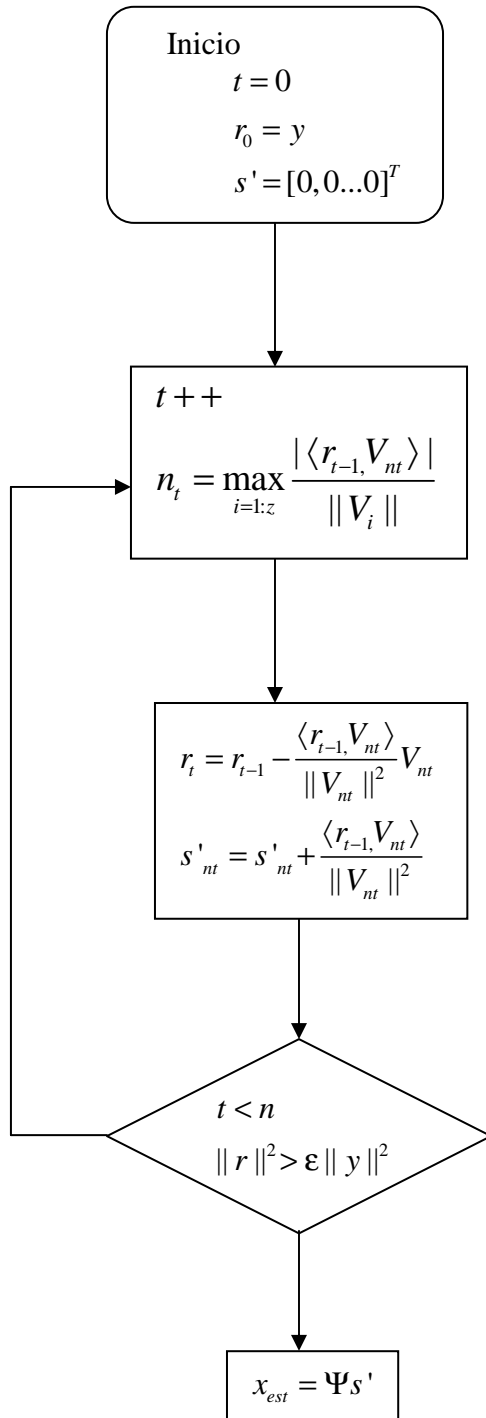
DATOS DE ENTRADA

- Matriz de medición: $\Phi \in \mathbb{R}^{m \times n}$
- Diccionario: $\Psi \in \mathbb{R}^{n \times n}$
- Vector de medición: $y = \Phi x \in \mathbb{R}^m$
- Diccionario holográfico: $V = \Phi \Psi = [V_1, V_2, \dots, V_l] \in \mathbb{R}^{m \times n}$
- Normas de las columnas del diccionario holográfico: $V_{mi} = \|\Psi_i\|_2$

SALIDA

- Estimado de la señal x_{est}

ALGORITMO



t: contador de iteraciones
r ₀ : residuo inicial
s': estimado de la señal
Φ: matriz de proyecciones aleatorias
Ψ: diccionario
y: proyecciones aleatorias
nt: posición del vector V _i más parecido al residuo
r _{t-1} : residuo en la iteración t-1
r _t : residuo en la iteración t
V _i : vector columna de V
V _{nt} : vector columna de V en la posición nt
s' _{nt} : estimado de la señal en la posición nt
⟨r _{t-1} , V _i ⟩: producto punto entre r _{t-1} y V _i
V _i : norma de V _i
x _{est} : estimado de la señal original

Fig15. Diagrama de flujo del algoritmo de reconstrucción Matching Pursuit.

El algoritmo MP procede de la siguiente forma:

Paso I: paso de inicialización, se inicia el contador de iteraciones en 1, en la primera iteración el residuo es el mismo vector de medición y finalmente el estimado de la señal va a ser un vector con todas sus componentes iguales a cero.

Paso II: en este paso se proyecta el residuo sobre cada una de las columnas del diccionario holográfico para conseguir cual de sus columnas es la que mayor semejanza tiene con el residuo. La columna con mayor semejanza al residuo es aquella que genera la máxima proyección y viene dada por aquella componente cuyo producto punto con respecto al residuo de el mayor valor.

Paso III: se encuentra el estimado para esa componente de la señal proyectando el residuo sobre la componente del diccionario holográfico que generó la máxima proyección. Se actualiza el residuo restándole la contribución que este átomo del diccionario holográfico aportó al conjunto de mediciones. Se suma el estimado de la señal para esta componente a la señal reconstruida.

Paso IV: si se cumple el criterio de parada, ya sea que se sobrepase el número de iteraciones máximas definido o que la norma del residuo sea menor al valor deseado, el algoritmo para, si no vuelve al paso II.

Paso VI: encontrar la señal reconstruida escogiendo los átomos del diccionario que originalmente conformaron la señal.

El problema principal de este algoritmo aparece al volver al paso II ya que se podría escoger un átomo que anteriormente ya había sido seleccionado, si esto sucede, la contribución que hace este átomo podría no ser muy significativa, en consecuencia el número de iteraciones crece si esto ocurre, aumentando así el tiempo y el costo para reconstruir la señal dada.

3.1.2 Reconstrucción de señales con MP

Mallat [15] introduce el algoritmo denominado Matching Pursuit para la descomposición de una señal en funciones bases de un diccionario. Las funciones que conforman el diccionario se elijen de forma que mejor se ajusten a la estructura de la señal a reconstruir. A pesar de que MP es no lineal, mantiene una conservación de energía que garantiza su convergencia. A continuación, Mallat muestra en su artículo el algoritmo general en el espacio de Hilbert:

Se puede definir a un diccionario \mathbf{D} como una familia de átomos o vectores $(g_\gamma)_{\gamma \in \Gamma}$, los cuales pertenecen al espacio de Hilbert \mathbf{H} , y poseen una norma unitaria $\|g_\gamma\| = 1$. Para un diccionario de átomos de tiempo-frecuencia, $\mathbf{H} = \mathbf{L}^2(\mathbf{R})$, cada uno de los vectores g_γ es un átomo definido por la ecuación :

$$(47) \quad g_\gamma(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) e^{i\xi t}$$

Si las expansiones lineales de los átomos de tiempo-frecuencia son densas en $L^2(\mathbf{R})$, entonces este diccionario está completo.

Sea $f \in \mathbf{H}$. Se desea calcular una expansión lineal de f sobre un conjunto de vectores seleccionados de \mathbf{D} , de modo que se obtenga la mejor correspondencia de su estructura interna. Esto se realiza empleando una aproximación sucesiva de f con proyecciones ortogonales sobre elementos de \mathbf{D} . Sea $g_{\gamma_0} \in \mathbf{D}$. El vector f se puede descomponer en:

$$(48) \quad f = \langle f, g_{\gamma_0} \rangle g_{\gamma_0} + R_f$$

dónde Rf es el vector residual después de aproximar f en dirección de g_{γ_0} . Claramente g_{γ_0} es ortogonal a Rf , por lo tanto:

$$(49) \quad \|f\|^2 = |\langle f, g_{\gamma_0} \rangle|^2 + \|R_f\|^2$$

Para minimizar $\|R_f\|$, se debe elegir $g_{\gamma_0} \in \mathbf{D}$ tal que $|\langle f, g_{\gamma_0} \rangle|$ sea máximo. En algunos casos, esto se puede lograr con tan solo encontrar un vector g_{γ_0} que sea el mejor en el sentido que cumpla con la condición:

$$(50) \quad |\langle f, g_{\gamma_0} \rangle| \geq \alpha \sup_{\gamma \in \Gamma} |\langle f, g_{\gamma} \rangle|$$

dónde α es un factor de optimización que satisface $0 < \alpha \leq 1$.

Matching Pursuit es un algoritmo iterativo que subdescompone el residuo Rf proyectándolo sobre un vector de \mathbf{D} , de modo que este vector corresponda lo mejor posible al residuo Rf ; este proceso se lleva a cabo de la misma manera como anteriormente se hizo para f . MP efectúa nuevamente este procedimiento con el siguiente residuo que se obtiene y así sucesivamente, hasta cumplir con alguna condición establecida, tal como el número de átomos a obtener o el valor de la relación $\|R^n f\| / \|f\|$.

Matching Pursuit se lleva a cabo de la siguiente manera:

Sea $R^0 f = f$. Suponiendo que se ha calculado el residuo de orden n : $R^n f$, para $n \geq 0$. A continuación se escoge un elemento $g_{\gamma} \in \mathbf{D}$ que corresponda lo mejor posible al residuo $R^n f$, bajo la siguiente condición:

$$(51) \quad |C(R^n f, g_{\gamma_n})| = \sup_{\gamma \in \Gamma} |C(R^n f, g_{\gamma})|$$

donde $C(R^n f, g_{\gamma})$ es una función de correlación que mide la similitud entre f y g_{γ} .

Entonces, el residuo $R^n f$ se subdescompone en:

$$(52) \quad R^n f = C(R^n f, g_{\gamma_n}) g_{\gamma_n} + R^{n+1} f$$

Se prueba entonces que el error $\|R^n f\|$ decae hacia cero. De modo que realizando iterativamente (51) se obtiene:

$$(53) \quad f = \sum C(R^n f, g_{\gamma_n}) g_{\gamma_n} \quad n = 0, 1, 2, \dots, \infty$$

La estructura del algoritmo Matching Pursuit permite que las operaciones se efectúen rápidamente hasta cumplir con la condición que se ha establecido.

3.2 Orthogonal Matching Pursuit (OMP)

3.2.1 Diseño del algoritmo OMP

Debido al gran número de iteraciones y mediciones necesarias para reconstruir una señal utilizando MP, surge la necesidad de desarrollar otro algoritmo iterativo que pueda mejorar estas desventajas.

Las mayores ventajas de este algoritmo son su velocidad y la facilidad de implementación. La evidencia empírica sugiere que el algoritmo OMP puede recuperar una señal k-sparse cuando el número de mediciones m es casi proporcional al nivel de escasez k. *Joel Tropp and Anna Gilbert [11]*.

La idea del OMP es básicamente la misma del MP, encontrar en cada iteración el átomo del diccionario holográfico que más se parece a la estructura del conjunto de mediciones proyectado y remover la contribución de dicho átomo conformando una señal residuo. La principal diferencia es la introducción de una nueva matriz de medición y hallar la solución de un problema de mínimos cuadrados, a partir de la cual se hallará el estimado para la señal.

Suponemos que x es una señal arbitraria k-sparse en \mathbb{R}^n , y s_1, s_2, \dots, s_m un conjunto de m vectores de mediciones. Formamos con estos vectores una matriz Φ de dimensión $m \times n$ cuyas filas son los vectores de mediciones, y tomamos el vector de datos como $y = \Phi x$ que será una combinación lineal de k columnas de Φ . Para identificar la señal original x necesitamos determinar qué columnas de Φ participan en la formación del vector y. En cada iteración tomaremos la columna de Φ cuya correlación con la parte restante de y sea mayor. Luego, sustraemos esa contribución del vector y y creamos el residuo. Se espera recuperar la señal tras k iteraciones.

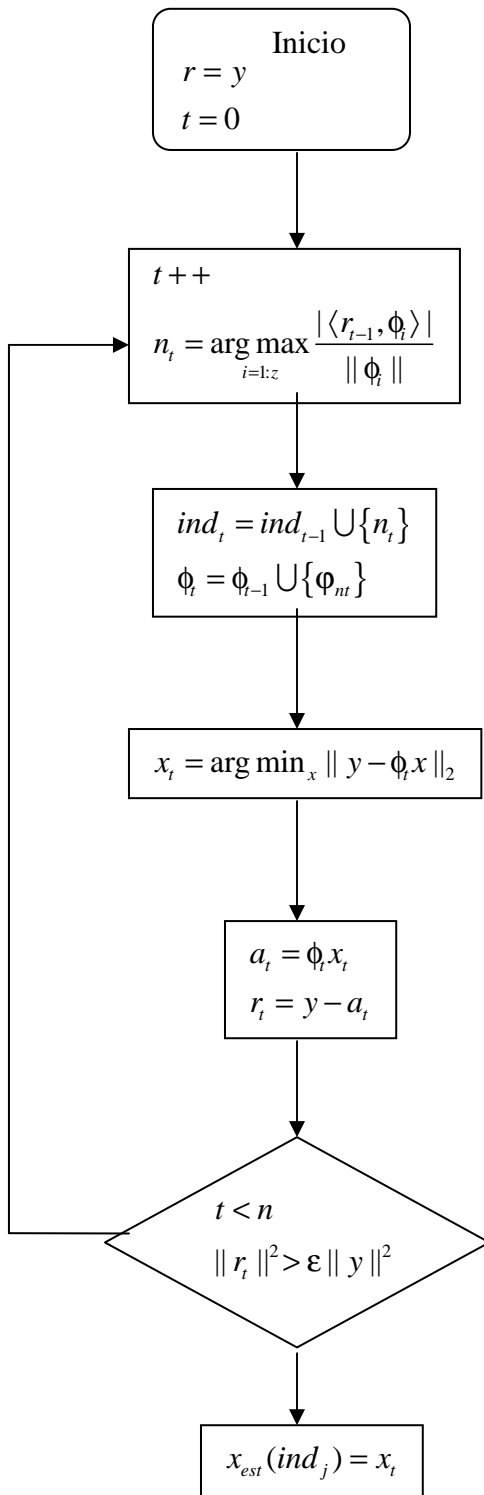
DATOS DE ENTRADA

- Matriz de medición: $\Phi \in \mathbb{R}^{m \times n}$
- Vector de medición: $y = \Phi x \in \mathbb{R}^m$
- Nivel de escasez de la señal k

SALIDA

- Estimado de la señal $x_{\text{est}} \in \mathbb{R}^n$
- Matriz formada por k elementos de $\{1, \dots, n\}$ ind_t
- Aproximación del vector y $a_t \in \mathbb{R}^n$
- Vector residuo $r_t \in \mathbb{R}^n$

ALGORITMO



t: contador de iteraciones

r_0 : residuo inicial

Φ : matriz de proyecciones aleatorias

y: proyecciones aleatorias

nt: posición del vector Φ_i más parecido al residuo

r_{t-1} : residuo en la iteración t-1

r_t : residuo en la iteración t

Φ_{nt} : vector columna de Φ en la posición nt

$\langle r_{t-1}, \Phi_i \rangle$: producto punto entre r_{t-1} y Φ_i

$\|\Phi_i\|$: norma de Φ_i

ind_t: matriz que almacena los índices de los átomos de Φ que producen la máxima proyección en cada iteración

Φ_t : matriz que almacena los átomos de Φ que producen la máxima proyección en cada iteración

x_t : solución al problema de mínimos cuadrados

a_t : aproximación de la señal en la iteración t

x_{est} : estimado de la señal original

Fig16. Diagrama de flujo del algoritmo de reconstrucción Orthogonal Matching Pursuit.

Paso I: paso de inicialización, se inicia el contador de iteraciones, en la primera iteración el residuo es el mismo vector de medición, finalmente se inicia el conjunto de índices a 0.

Paso II: en este paso se proyecta el residuo sobre cada una de las columnas de la matriz de mediciones para conseguir cual de sus columnas es la que mayor semejanza tiene con el residuo. La columna con mayor semejanza al residuo es aquella que genera la máxima proyección y viene dada por aquella componente cuyo producto punto con respecto al residuo de el mayor valor.

Paso III: en el conjunto ind se incorpora en cada iteración el índice correspondiente al átomo del diccionario holográfico que maximizó la proyección efectuada en el paso II. Además la matriz Φ_t va almacenando en cada iteración la componente de la matriz de mediciones que produce la máxima proyección

Paso IV: de la solución del problema de mínimos cuadrados se obtiene el estimado de la señal, la solución a este problema es la siguiente:

Sea el siguiente problema de minimización

$$x_t = \operatorname{argmin} \|y - \Phi_t x\|^2$$

Si se deriva e iguala a cero el lado izquierdo de la igualdad, se tiene:

$$\Phi_t'(y - \Phi_t x_t) = \Phi_t' y - \Phi_t' \Phi_t x_t = 0$$

Ahora despejando queda que:

$$x_t = (\Phi_t' \Phi_t)^{-1} \Phi_t' y$$

Paso V: calculamos el nuevo estimado de la señal al proyectar la solución obtenida para problemas de mínimos cuadrados sobre la nueva matriz de medición. Se actualizará el residuo restándole el estimado de la señal al conjunto de mediciones original.

Paso VI si se cumple el criterio de parada, ya sea que se sobrepase el número de iteraciones máximas definido o que la norma del residuo sea menor al valor deseado, el algoritmo para, si no vuelve al paso II.

Paso VII: obtener el estimado de la señal. El estimado de la señal x_{est} , tiene componentes distintas de cero en las posiciones indicadas por el conjunto de índices ind, el valor del estimado en la posición ind_j será igual a la componente j -ésima de x_t .

Debido a que r_t proviene de la solución del problema de mínimos cuadrados, este siempre va a ser ortogonal a las columnas del diccionario holográfico que fueron almacenadas en la matriz Φ_t , de modo que ninguno de los átomos del diccionario anteriormente utilizados aportará contribución alguna a la definición del residuo. Esto garantiza, que ninguna de las columnas utilizadas anteriormente sea indexada en la

nueva matriz de medición Φ_t , evitando que esta se pueda convertir en una matriz singular por lo tanto no invertible en la solución de mínimos cuadrados, reduciendo también el número de iteraciones necesarias para que el OMP pueda reconstruir una señal con éxito.

3.2.2 Reconstrucción de señales con OMP

Si tomamos medidas aleatorias de una señal sparse usando una matriz de mediciones admisible (vistas en el apartado 2.4.2), el algoritmo OMP puede emplearse para la recuperación de la señal original con alta probabilidad.

Teorema 5([11]): (OMP con matriz de mediciones admisible): fijamos $\delta \in (0, 0.36)$ y elegimos $m \geq Ck \ln(n/\delta)$. Suponemos que x es una señal k -sparse en \mathbb{R}^n y tomamos una matriz de mediciones admisible Φ de dimensión $m \times n$ independiente de la señal. Dado el dato $y = \Phi x$, OMP puede reconstruir la señal con una probabilidad mayor que $1 - \delta$. La constante cumple $C \leq 20$. Para valores elevados de k , se puede reducir ese valor a $C \leq 4 + \eta$ siendo η cualquier número positivo. [12].

Si consideramos el evento E_{succ} en el cual el algoritmo identifica correctamente la señal x tras k iteraciones. Sólo decrementamos la probabilidad de éxito si imponemos el requerimiento adicional de que el valor singular más pequeño de Φ_{opt} encuentre una cota inferior. Para definir Φ_{opt} asumimos que las k primeras entradas de la señal original x son distintas de cero y las siguientes $n-k$ entradas son cero. Luego, el vector y es una combinación lineal de las k primeras columnas de la matriz Φ . Representamos la matriz como $\Phi = [\Phi_{opt} | \Psi]$, por lo que Φ_{opt} tiene k columnas y Ψ tiene $n-k$ columnas. Entonces, como vemos en la demostración del *Teorema 5* en el Apéndice 5.

$$(54) \quad P(E_{succ}) \geq 1 - n^2 e^{-Cm/k}$$

Como conclusión, la elección de $m \geq C \cdot k \cdot \log(n/\delta)$ es suficiente para reducir la probabilidad de fallo por debajo de δ .

Aún así el algoritmo OMP puede fallar, demostramos la posibilidad de obtener un éxito o un fallo para matrices de mediciones Gaussianas en la siguiente proposición:

Proposición 7([11]): Elegimos una señal arbitraria x k -sparse en \mathbb{R}^n , con $m \geq 2k$. Suponemos que Φ es una matriz Gaussiana $m \times n$ y ejecutamos OMP con $y = \Phi x$. Si el residuo r_k después de k iteraciones es cero, entonces OMP ha identificado correctamente la señal x con probabilidad 1. Si el residuo después de k iteraciones no es cero, el algoritmo habrá fallado.

Prueba: si $r_k = 0$ pero $\tilde{x} \neq x$, será posible escribir el vector y como combinación lineal de k columnas de la matriz Φ de dos formas diferentes. En consecuencia, existe una dependencia lineal entre $2k$ columnas de la matriz Φ . Si Φ es una matriz Gaussiana $m \times n$ y $m \geq 2k$ esto no ocurrirá nunca.

Para medidas de Bernoulli, existe una proposición similar con probabilidad exponencial cercana a 1. Este resultado se deduce del hecho de que una pequeña fracción exponencial de matrices (cuadradas) son singulares. [13].

3.3 Conclusiones

El algoritmo MP es un algoritmo iterativo que trata de encontrar el átomo del diccionario holográfico que aporta mayor contribución a la definición de la señal proyectada, encuentra el estimado de la contribución de dicho átomo a la conformación de la señal y remueve dicha contribución que este aporta a la estructura del conjunto de mediciones de la señal, definiendo así un residuo. Se procede iterativamente hasta que el criterio de parada se cumpla. Debido al gran número de iteraciones y mediciones necesarias para reconstruir una señal utilizando MP, surge la necesidad de desarrollar otro algoritmo iterativo que pueda mejorar estas desventajas.

El problema principal de este algoritmo aparece al volver al paso II ya que se podría escoger un átomo que anteriormente ya había sido seleccionado, si esto sucede, la contribución que hace este átomo podría no ser muy significativa, en consecuencia el número de iteraciones crece si esto ocurre, aumentando así el tiempo y el costo para reconstruir la señal dada.

La idea del OMP es básicamente la misma del MP, encontrar en cada iteración el átomo del diccionario holográfico que más se parece a la estructura del conjunto de mediciones proyectado y remover la contribución de dicho átomo conformando una señal residuo. La principal diferencia es la introducción de una nueva matriz de medición y hallar la solución de un problema de mínimos cuadrados, a partir de la cual se hallará el estimado para la señal.

Las mayores ventajas del algoritmo OMP son su velocidad y la facilidad de implementación. La evidencia empírica sugiere que el algoritmo OMP puede recuperar una señal k -sparse cuando el número de mediciones m es casi proporcional al nivel de escasez k . También se ha demostrado en este apartado que fijando una $\delta \in (0, 0.36)$ y tomando $m \geq Ck \ln(n/\delta)$, el algoritmo OMP puede reconstruir la señal con una probabilidad mayor que $1 - \delta$ [11].