



(19) 中華民國智慧財產局

(12) 發明說明書公開本

(11) 公開編號：TW 201228268 A1

(43) 公開日：中華民國 101 (2012) 年 07 月 01 日

(21) 申請案號：099146951

(22) 申請日：中華民國 99 (2010) 年 12 月 30 日

(51) Int. Cl. :

*H04B7/04 (2006.01)*

*H04L27/06 (2006.01)*

(71) 申請人：國立交通大學 (中華民國) NATIONAL CHIAO TUNG UNIVERSITY (TW)

新竹市大學路 1001 號

(72) 發明人：黃崇榮 HUANG, CHUNG JUNG (TW) ; 吳智湧 WU, CHIH YUNG (TW) ; 李大嵩 LEE, TA SUNG (TW)

(74) 代理人：黃于真；李國光

申請實體審查：有 申請專利範圍項數：10 項 圖式數：5 共 34 頁

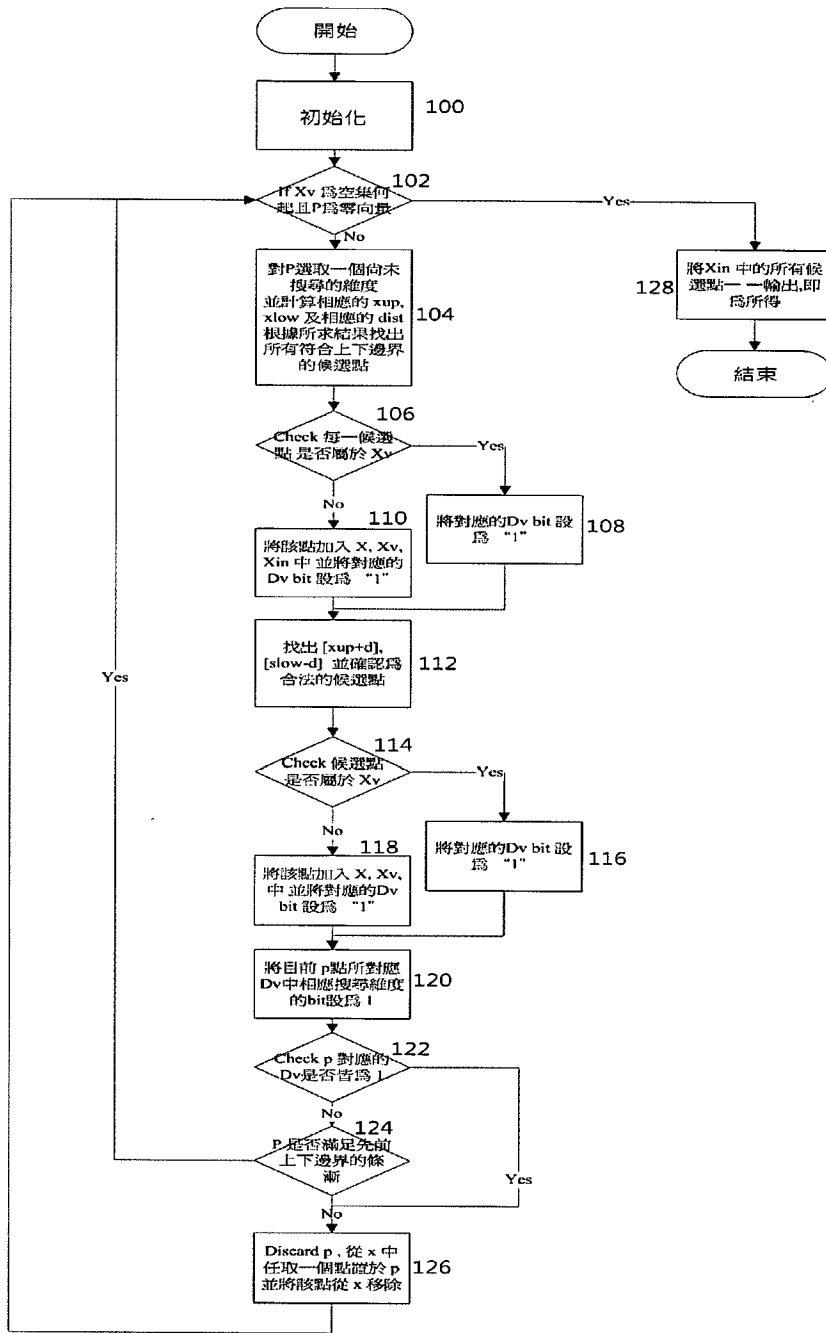
(54) 名稱

基於幾何方法欠定多輸入多輸出系統之高效率解碼器及其解碼方法

GEOMETRY BASED EFFICIENT DECODER FOR UNDERDETERMINED MIMO SYSTEMS AND DECODING METHOD THEREOF

(57) 摘要

本發明係揭露一種具有低解碼複雜度之欠定系統解碼器。該解碼器包含了兩個步驟：1. 藉由所提出的高效率的平面候選點搜尋器將所有所需的候選點一一找出。2. 針對這些候選點集合進行等效平面交集的動作並配合動態半徑調整機制來快速地找出該問題的解。本發明亦提出一可與該解碼器結合之通道矩陣行向量的排序策略，進而提供低運算需求及近似最大似然搜尋的解碼性能。



# 發明專利說明書

(本申請書格式、順序，請勿任意更動，※記號部分請勿填寫)

※ 申請案號： 99146951

※ 申請日： 99.12.30      ※IPC 分類： H04B 7/04 (2006.01)  
H04L 17/06 (2006.01)

一、發明名稱：(中文/英文)

基於幾何方法欠定多輸入多輸出系統之高效率解碼器  
及其解碼方法/GEOMETRY BASED EFFICIENT DECODER FOR  
UNDERDETERMINED MIMO SYSTEMS AND DECODING METHOD  
THEREOF

二、中文發明摘要：

本發明係揭露一種具有低解碼複雜度之欠定系統解碼器。該解碼器包含了兩個步驟：1.藉由所提出的高效率的平面候選點搜尋器將所有所需的候選點一一找出。2.針對這些候選點集合進行等效平面交集的動作並配合動態半徑調整機制來快速地找出該問題的解。本發明亦提出一可與該解碼器結合之通道矩陣行向量的排序策略，進而提供低運算需求及近似最大似然搜尋的解碼性能。

三、英文發明摘要：

A decoder for underdetermined MIMO systems with low decoding complexity is provided. The decoder consists of two stages: 1. Obtaining all valid candidate

points efficiently by slab decoder. 2. Finding the optimal solution by conducting the intersectional operations with dynamic radius adaptation to the candidate set obtained from Stage 1. A reordering strategy is also disclosed. The reordering can be incorporated into the proposed decoding algorithm to provide a lower computational complexity and near-ML decoding performance for underdetermined MIMO systems.

**四、指定代表圖：**

(一)本案指定代表圖為：第(1)圖。

(二)本代表圖之元件符號簡單說明：

略

**五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：**

## 六、發明說明：

### 【發明所屬之技術領域】

本發明是有關於一種欠定系統之解碼器，特別是有關於一種基於幾何方法之高效率欠定系統解碼器及其解碼方法。

### 【先前技術】

目前，為了滿足不斷增加的個人通訊需求，新一代無線通訊系統之設計必需支援高資料傳輸率及高行動力。然而，資料的連結品質會受到無線頻道中多路徑傳輸所造成的頻率或時間選擇衰減之影響。此外，無線通訊的品質及可靠性也會因多普勒效應及載波頻率/相位之因素而降低。再者，由於可用頻寬及傳輸功率上的限制，使無線通訊系統之設計變得更加困難。因此，在此領域中有許多新發明的技術被大量用來改善無線通訊連結之可靠性與頻譜效率，例如編碼多載波調變、智慧型天線及多輸入多輸出技術以及適應性調變。

在這些技術中，多輸入多輸出(MIMO)技術是最傑出的一項。MIMO 技術涉及了多個天線的使用，以改善連結表現。MIMO 技術有兩個主要的特徵：為增加資料傳輸率之空間多工處理，以及為改善連結品質之空間多樣化。空間多工處理藉由同時發射多個獨立的資料串流而使資料傳輸率線性增加。空間多樣化藉由在空間、時間及頻率上使用多個發射信號之複製信號(理論上是各自

獨立的)。這兩個特徵是必需相互平衡的，並可在高資料傳輸率及傳輸可靠性之間提供一有效、折衷的解決方案。

主要的 MIMO 信號偵測方式包括了線性偵測、接續干擾消除(SIC)及最大可能偵測(ML)。前兩者的好處是具有低解碼複雜度及容易實施，但其偵測表現並不是最佳的。最大可能偵測可以提供最佳的偵測表現，但其複雜度卻會隨信號點分佈區域的大小及發射天線的數目而成指數增加。因此，如何取得高效能及低複雜度特性是 MIMO 接收機設計的主要關鍵問題。為了降低最大可能偵測器之複雜度，球型解碼演算法(SDA)受到相當的關注，被認為是一種有效率的 MIMO 系統偵測方法。然而，傳統的 SDA 並不能直接使用於發射天線數目大於接收天線數目之 MIMO 系統中。

為了克服傳統 SDA 的上述缺點，傳統通用球型解碼器(GSD)、雙層球型解碼器(DLSD)及平面球型解碼器(SSD)被提出。這些解碼器會將欠定系統轉換成過定系統，而可以使用 SDA 來求解。由於 GSD 必需在傳送與接收天線數量差的維度大小上為最大可能之解進行竭盡式搜尋，其解碼複雜度會隨信號點分佈區域之大小及天線數目差值而增加。DLSD 使用了外球型解碼器來找出有效的候選點，然後再經由內球型解碼器來找出該問題的解。SSD 則是使用幾何的概念來找到有效的候選點，以降低 DLSD 的搜尋複雜度。然而，在 SSD 中，仍然會執行過多次數的 SDA。

**【發明內容】**

有鑑於上述習知技藝之問題，本發明之其中一目的就是在提供一種具有低解碼複雜度之欠定系統解碼器。該解碼器包含了兩個步驟：1.藉由所提出的高效率的平面候選點搜尋器將所有所需的候選點一一找出。2.針對這些候選點集合進行等效平面交集的動作並配合動態半徑調整機制來快速地找出該問題的解。本發明亦提出一可與該解碼器結合之通道矩陣行向量的排序策略，進而提供低運算需求及近似最大似然搜尋的解碼性能。

**【實施方式】**

依據本發明之實施例，提供了一種具有低解碼複雜度之欠定系統解碼器。該解碼器之演算法包括了兩個階段。第一階段包含了經由區塊解碼器有效率地取得所有候選點之步驟，而第二階段包含了經由使用依第一階段取得之候選點集合而調整的動態半徑來進行交集運算而找出最佳解答之步驟。此外，亦提供了一種排序策略(前置處理)，可以與上述演算法整合而為欠定系統提供一低運算需求及最大似然搜尋的解碼性能。

**第一階段**

請參閱第 1 圖，其係為本發明一實施例中基於幾何欠定系統解碼演算法之第一階段的步驟流程圖。其與 PDA(Plane Decoding Algorithm)及 SLA(Slab Decoding



Algorithm)之演算類似，但只使用了單一個演算程序就可以找出位於平面中的候選點，而不會對信號點及分佈區域進行重複搜尋。

假設一 MIMO 系統具有  $N_t$  個發射天線及  $N_r$  個接收天線。發射信號向量及接收信號向量間之關係可以表示成：

$$\tilde{\mathbf{y}} = \tilde{\mathbf{H}}\tilde{\mathbf{x}} + \tilde{\mathbf{n}}, \quad (1)$$

其中， $\tilde{\mathbf{y}} = [\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_{N_r}] \in \mathbb{C}^{N_r \times 1}$  代表接收信號向量，

$\tilde{\mathbf{x}} = [\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_{N_t}] \in \mathbb{C}^{N_t \times 1}$  代表發射信號向量， $\tilde{\mathbf{H}}$  代表頻率平緩衰減通道矩陣：

$$\tilde{\mathbf{H}} = \begin{bmatrix} \tilde{h}_{1,1} & \tilde{h}_{1,2} & \dots & \tilde{h}_{1,N_t} \\ \tilde{h}_{2,1} & \tilde{h}_{2,2} & \dots & \tilde{h}_{2,N_t} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{h}_{N_r,1} & \tilde{h}_{N_r,2} & \dots & \tilde{h}_{N_r,N_t} \end{bmatrix} \in \mathbb{C}^{N_r \times N_t}, \quad (2)$$

其中， $\tilde{\mathbf{H}}$  中的元素是具有正規分佈  $CN(0,1)$  之複數相等獨立分佈 (i.i.d) 高斯隨機變數，且  $\tilde{\mathbf{n}} = [\tilde{n}_1, \tilde{n}_2, \dots, \tilde{n}_{N_r}] \in \mathbb{C}^{N_r \times 1}$  代表具有正規分佈  $CN(0, \sigma^2 \mathbf{I})$  之複數加型高斯白雜訊 (AWGN)。在欠定系統中，是假定  $N_r < N_t$  的。

複數方程式 (1) 可以經由實數分解而被重寫成一個均等實數系統：

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \quad (3)$$

其中

$$\begin{aligned} \mathbf{y} &= \begin{bmatrix} \text{Re}\{\tilde{\mathbf{y}}\}^T & \text{Im}\{\tilde{\mathbf{y}}\}^T \end{bmatrix} \in \mathbf{R}^M \\ \mathbf{x} &= \begin{bmatrix} \text{Re}\{\tilde{\mathbf{x}}\}^T & \text{Im}\{\tilde{\mathbf{x}}\}^T \end{bmatrix} \in \mathbf{R}^N, \\ \mathbf{n} &= \begin{bmatrix} \text{Re}\{\tilde{\mathbf{n}}\}^T & \text{Im}\{\tilde{\mathbf{n}}\}^T \end{bmatrix} \in \mathbf{R}^M \end{aligned} \quad (4)$$

且

$$\mathbf{H} = \begin{bmatrix} \text{Re}\{\tilde{\mathbf{H}}\}^T & -\text{Im}\{\tilde{\mathbf{H}}\}^T \\ \text{Im}\{\tilde{\mathbf{H}}\}^T & \text{Re}\{\tilde{\mathbf{H}}\}^T \end{bmatrix} \in \mathbf{R}^{M \times N}, \quad (5)$$

其中， $M=2N_r$ ;  $N=2N_t$ 。

經由將  $y$  與格點之間的歐幾里德距離最小化的步驟可以得到  $x$  的最大可能估計  $\tilde{x}$ ：

$$\tilde{x} = \arg \min_{x \in \mathbf{Z}^N} \|y - \mathbf{H}x\|^2 = \arg \min_{x \in \mathbf{Z}^N} \|\mathbf{R}(\rho - x)\|^2, \quad (6)$$

其中，對 16-QAM 的情況來說， $\mathbf{Z} = \{-3, -1, 1, 3\}$  而  $\rho = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}y$ ， $\mathbf{Q}$  則是一  $M \times M$  單元矩陣， $\mathbf{R}$  是對應於  $\mathbf{H}$  之 QR 分解的  $M \times N$  上三角矩陣，意即  $\mathbf{H} = \mathbf{Q}\mathbf{R}$ 。為了有效率地求得方程式 (6) 中的  $\tilde{x}$ ，必需使用一種演算程序，在已知半徑  $C(>0)$  內找出所有符合以下條件之格點：

$$\|y - \mathbf{H}x\|^2 \leq C^2 \quad (7)$$

將  $\mathbf{H} = \mathbf{Q}\mathbf{R}$  代入方程式 (7) 中可以得到：

$$\|y' - \mathbf{R}x\|^2 \leq C^2, \quad (8)$$

其中， $y' = \mathbf{Q}^T y$ 。若  $N > M$ ，則在第  $M$  層中：

$$-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C, \quad (9)$$

方程式 (9) 涉及了  $N - M + 1$  個面之偵測。本實施例中之演算法即是以找出所有落於目標區塊 (slab) 中的信號點，意即符合方程式 (9) 之點。

考慮一個廣義區塊其對應方程式為  $|\sum_{i=1}^N w_i x_i - y| \leq C$ 。本實施例之演算法即是以找出所有滿足該方程式中的候選點。

$X$  為需要再被搜尋的點集合， $X_v$  為有被搜尋過的點集合， $X_{in}$  為滿足搜尋條件的點集合，而三者初始值皆為空集合。每一個候選點都會有一個對應的向量  $D_v$ ， $D_v$  中的每一個元素皆為二進制的值，「1」代表所對應的維度已經被搜尋過了，反之為「0」。在一開始，每一個候選點所對應  $D_v$  的初始值為  $D_v = [0, 0, \dots, 0] \in \mathbb{Z}^N$ ，代表所有對應的維度都還未搜尋。 $p$  為一暫存的變數，其值以  $p = [p_1 \ p_2 \ \dots \ p_N]^T \in \mathbb{R}^N$  表示，代表目前正在以該點出發進行搜尋。初始值為  $0$ 。

對於每一個候選點，我們定義其相對應的距離（表示該點與平面方程式  $y = \sum_{i=1}^N w_i x_i$  的幾何距離）為  $dist = \left| \sum_{i=1}^N w_i x_i - y \right|$ 。

首先，在步驟 100 中進行初始化動作，即選定空間中任一點當成初始點  $x_{int} = [x_1^{int} \ x_2^{int} \ \dots \ x_N^{int}]^T \in \mathbb{R}^N$ ，且令  $X = x_{int}$ 。接著從  $X$  取出存於  $p$  並從  $X$  中刪除該點。

接著，在步驟 102 中判斷  $X_v$  集合是否為空集合且  $p$  值是否為  $0$ ，若是，則執行步驟 128，代表  $X_{in}$  集合內的所有點與相對應的  $dist$  即為所求；否則，在步驟 104 中選取任一個  $k$  值，而其中  $k$  指的是這個點尚未被搜尋過的維度，也就是它所對應的  $D_v$  向量中的第  $k$  個元素為「0」。

接著計算：

$$x_{k,up} = p_k - \frac{\Delta y(p)}{w_k} + C,$$

$$x_{k,low} = p_k - \frac{\Delta y(p)}{w_k} - C.$$

其中  $\Delta y(p) = \sum_{i=1}^N w_i p_i - y$ 。

根據  $x_{k,up}, x_{k,low}$  的計算結果，將其間所有的合法座標點一一產生出來。在步驟 106 中判斷產生的候選點是否存在於  $X_v$

內，若否，則在步驟 110 中將該點分別加入  $X$ 、 $X_v$  和  $X_{in}$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」。若是，則在步驟 108 中僅將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」。透過先前計算所得的  $\Delta y(\bullet)$  來有效地計算出該候選點其相應的  $dist$  值。

接下來，在步驟 112 中，根據  $x_{k,up}, x_{k,low}$  產生出對應的  $|x_{k,up} + d|, |x_{k,low} - d|$  這兩個候選點(這兩點其實就是落在上下臨界點外的最接近 slab 的候選點)。其中， $d$  為座標點之間之最小距離。在步驟 114 中，判斷這些點是否存在於  $X_v$  之內，若否，則在步驟 118 中將其分別加入  $X$  和  $X_v$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」。若是，則在步驟 116 中僅將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」。需注意的是，此時並不需要計算其相應的  $dist$  值。

至此，目前以  $p$  點出發且對第  $k$  個方向已經搜尋完畢，所以在步驟 120 中，將該點相應  $D_v$  向量中的第  $k$  個元素設定為「1」。

然後，在步驟 122 中，判斷：(a)該相應的  $D_v$  向量內的所有元素是否都為「1」，若是，則執行步驟 126；若否，則在步驟 124 中再判斷：(b)該  $p$  值是否滿足目前的搜尋條件，若否，則執行步驟 126；若否，則跳回步驟 102 繼續搜尋。在步驟 126 中，從  $X$  中任取出一個候選點取代目前的  $p$  並將取出的候選點從  $X$  中刪除(在取候選點時，若  $X$  為空集合，則  $p$  的值為 0)。

最後，藉由循環執行上述之步驟，在  $X_{in}$  中的每一個

點都是落於目標區塊中的。上述的搜尋演算方法可以快速地找到某一已知  $C$  下符合公式(9)之所有有效候選點。傳統欠定系統解碼器在限制半徑不大時，會藉由 SDA 並使用所有的點來找出近似最大似然的解。然而，如上述，解碼複雜度與候選點數目是密切相關的。在天線數目差值很大時，由上述搜尋演算法所得到的候選點數目仍然很大。因此，依據本發明之實施例，最佳的解答是經由執行動態半徑調整之交集運算來得到，而非使用 SDA。以下將進行說明。

## 第二階段

第 2 圖係為本發明一實施例中基於幾何欠定系統解碼演算法之第二階段的步驟流程圖。公式(8)可以展開重新表示為：

$$\left\| \begin{bmatrix} y'_i \\ \vdots \\ y'_{M-1} \\ y'_M \end{bmatrix} - \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,M-1} & r_{1,M} & \cdots & r_{1,N} \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{M-1,M-1} & r_{M-1,M} & \cdots & r_{M-1,N} \\ 0 & \cdots & 0 & 0 & r_{M,M} & \cdots & r_{M,N} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \right\|^2 \leq C^2$$

$$\Rightarrow \sum_{i=1}^M \left[ y'_i - \sum_{j=1}^N r_{ij} x_j \right]^2 \leq C^2 \quad (11)$$

在方程式(11)中的平面方程式總數是  $M$ 。這些方程式可以以相對的識別代號來重新表示：

Slab equations ID	Slab equations	
Slab M	$-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C$	
$\vdots$	$\vdots$	(12)
Slab 2	$-C \leq y'_2 - [r_{2,2}x_2 + \dots + r_{2,N}x_N] \leq C$	
Slab 1	$-C \leq y'_1 - [r_{1,1}x_1 + \dots + r_{1,N}x_N] \leq C$	

在 SSD 中，只有最後一個平面方程式(Slab M)被使用。然而，在方程式(12)中平面方程式的總數是 M。因此，為了能降低解碼複雜度，本發明實施例乃是使用所有可得的平面方程式來找出候選點集合，而不能僅僅只使用 Slab M。

請參閱第 2 圖，在步驟 210 中， $i$  被設定為  $M$ ，並經由 Slab M 方程式  $-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C$  找到最初候選點集合 ( $\hat{x}_M$ )。其集合中每一個候選點相對之距離( $dist_M$ )可以經由第一階段的搜尋演算法得到。

在步驟 220 中，判斷  $i=1$  是否成立。若成立，則跳至步驟 240，否則，在步驟 221 中，經由以下的半徑限制找出相對於候選點集合中每一候選點之  $x_{i-1}$  的上限 ( $x_{i-1,up}$ ) 與下限 ( $x_{i-1,low}$ )：

$$D_{i,i-1} = \left\| \begin{bmatrix} y'_{i-1} \\ y'_i \end{bmatrix} - \begin{bmatrix} r_{i-1,i-1} & r_{i-1,i} & \dots & r_{i-1,N} \\ 0 & r_{i,i} & \dots & r_{i,N} \end{bmatrix} \begin{bmatrix} x_{i-1} \\ x_i \\ \vdots \\ x_N \end{bmatrix} \right\| \leq C^2$$

且

$$x_{i-1,up} = \frac{\sqrt{C^2 - (dist_i)^2} + (y'_{i-1} - \sum_{j=1}^N r_{i-1,j}x_j)}{r_{i-1,i-1}}$$

$$x_{i-1,low} = \frac{-\sqrt{C^2 - (dist_i)^2} + (y_{i-1}' - \sum_{j=i}^N r_{i-1,j} x_j)}{r_{i-1,i-1}}$$

因此， $dist_{i-1}$  可以經由下列公式得到：

$$dist_{i-1}^2 = \left( y_{i-1}' - \sum_{j=i-1}^N r_{i-1,j} x_j \right)^2 + (dist_i)^2.$$

在步驟 230 中，使用相對於每一個候選點之上限與下限可以建立一個新的候選點集合 ( $\hat{x}_i$ )：

$$\hat{\mathbf{x}}_{i-1} = \left[ \left[ x_{i-1,low}, x_{i-1,up} \right] \cap \mathbf{Z}, x_i, \dots, x_N \right]$$

選擇一與  $\hat{x}_{i-1}$  具有最小距離之候選點，並得出其對應的 ZF-SIC 之解。接著，再得到一新的半徑 ( $C_{new}$ )。

接著在步驟 232 中，根據目前所求得新的半徑 ( $C_{new}$ ) 及內部的控制變數「state」(初始值為 increment) 來對  $C$  進行更新調整。其調整的規則如下：

	State = increment	State = decrement
$C_{new} > C$	將原本 $C$ 加上一個小增量，state 保持在 increment	$C$ 不改變，state 值也不改變
$C_{new} \leq C$	$C = C_{new}$ ，將 state 設為 decrement	$C = C_{new}$ ，state 維持為 decrement

在步驟 233 中，將  $i$  設定為  $i-1$  後，跳至步驟 220。

在步驟 240 中，將與  $\hat{x}_i$  具有最小距離之候選點選擇為  $x$  之估計。

必需注意的是，若最初之候選點集合 $(\hat{x}_M)$ 是空集合， $C$  會被增加並重新執行第一階段的動作。並且，若候選點集合 $(\hat{x}_i, 1 \leq i \leq M-1)$ 是空集合時，其 ZS-SIC 之解為做為  $x$  之估計。

### 前置處理

在欠定系統中，會使用某些技術對通道矩陣進行前置處理之動作，這可以降低 SDA 之解碼複雜度或是改善其性能。這種前置處理技術有很多，例如晶格縮減、行置換及比例伸縮。然而，這些前置處理技術都無法在本發明實施例之欠定系統解碼器中發揮降低解碼複雜度之功效。如上述，解碼複雜度的主要關鍵是候選點集合之交集。因此，必需使交集運算之程序被執行得越少次越好。

首先，落於最初目標區塊中的候選點數目 $(N_p)$ 可以被定義為解碼複雜度之指標。因此必需  $N_p$  越小越好。最初目標區塊之公式可以被重新表示成：

$$-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C. \quad (13)$$

最初的候選平面可以經由通道矩陣之行重新排列來找出，且候選平面之數目為  $C(N, N-M+1)$ 。自候選平面中選擇一個最初目標區塊對本發明的解碼器來說是很重要的。以幾何上的概念來說， $|y'_M|$  代表從原點到目標區塊之距離，如第 3 圖所示。假設平面方程式是  $h_1x_1 + h_2x_2 = y$ ，其中  $h_i$  是常數而  $y$  是變數，且  $|y'_M|$  被定義為平面 $(\xi)$ 之位置指標。如第 3 圖所示，當  $\xi$  較大時， $N_p$  較小。因此，具有



最大  $\xi$  之平面將會被選為最初目標區塊。

第 4 圖係為本發明一實施例中基於幾何欠定系統解碼演算法之前置處理的步驟流程圖。方程式(8)中的  $y'$  可以被重新表示為：

$$y' = [y'_1 \ y'_2 \ \cdots \ y'_M]^T = [\bar{q}_1^T y \ \bar{q}_2^T y \ \cdots \ \bar{q}_M^T y]^T \in \mathbf{R}^M, \quad (14)$$

其中， $Q = [\bar{q}_1 \ \bar{q}_2 \ \cdots \ \bar{q}_M] \in \mathbf{R}^{M \times M}$ 。為了得到  $y'_M$ ，必需計算  $\bar{q}_M$ 。由於不同的通道矩陣的行重新排列會產生不同的  $\bar{q}_M$ ，欲解答的問題就變成了如何快速地找出  $\bar{q}_M$ 。通道矩陣之行重新排列可以被表示成  $\mathbf{HP}$ ，其中  $\mathbf{P}$  是一置換矩陣。

在步驟 410 中，由於  $\mathbf{P}=\mathbf{I}$ ，因此  $\mathbf{HP}=\mathbf{H}=[\bar{h}_1 \ \bar{h}_2 \ \cdots \ \bar{h}_N]$ 。 $\mathbf{HP}$  的 QR 分解可以表示成：

$$\mathbf{V}_1 \mathbf{H} = \begin{bmatrix} \mathbf{T}_1 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_1 \end{bmatrix}, \quad (15)$$

其中， $\mathbf{V}_1$  是  $\mathbf{H}$  的第一行向量之  $M \times M$  的 Householder 矩陣， $\mathbf{T}_1$  是一比例值而  $\tilde{\mathbf{H}}_1 = \mathbf{V}_1[2:M, 1:M] \mathbf{H}[1:M, 2:M]$ 。接著，定義一新矩陣  $\tilde{\mathbf{H}}_1 \in \mathbf{R}^{(M-1) \times (M-1)}$  以再進行相同的步驟：

$$\mathbf{V}_2 \tilde{\mathbf{H}}_1 = \begin{bmatrix} \mathbf{T}_2 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_2 \end{bmatrix}, \quad (16)$$

其中， $\mathbf{V}_2$  是  $\tilde{\mathbf{H}}_1$  的第一行向量之  $(M-1) \times (M-1)$  的 Householder 矩陣， $\mathbf{T}_2$  是一比例值而  $\tilde{\mathbf{H}}_2 = \mathbf{V}_2[2:M-1, 1:M-1] \tilde{\mathbf{H}}_1[1:M-1, 2:M-1]$ 。依此類推進行，可以得到  $\mathbf{H}$  的 QR 分解為：

$$\mathbf{Q}^T \mathbf{H} = \mathbf{R}, \quad (17)$$

其中

$$\mathbf{Q}^T = \begin{bmatrix} \bar{\mathbf{q}}_1^T \\ \bar{\mathbf{q}}_2^T \\ \vdots \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \mathbf{U}_{M-1} \cdots \mathbf{U}_2 \mathbf{U}_1$$

且

$$\mathbf{U}_i = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_i \end{bmatrix}$$

由於  $\bar{\mathbf{q}}_M$  是唯一必需考慮的，因此僅計算  $\mathbf{U}_{M-1}$  之最後一列，而可以得到：

$$\begin{bmatrix} \otimes \\ \vdots \\ \otimes \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes \\ \mathbf{0} & v_{M-1(2,1)} & v_{M-1(2,2)} \end{bmatrix} \Omega_1, \quad (18)$$

其中  $\Omega_1 = \mathbf{U}_{M-2} \cdots \mathbf{U}_1$ 。為了避免不必要的計算動作，只有  $\Omega_1$  的最後兩列被考慮。因此， $\mathbf{U}_{M-2}$  的最後兩列可以依以下進行計算：

$$\Omega_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes & \otimes \\ \mathbf{0} & v_{M-2(2,1)} & v_{M-2(2,2)} & v_{M-2(2,3)} \\ \mathbf{0} & v_{M-2(3,1)} & v_{M-2(3,2)} & v_{M-2(3,3)} \end{bmatrix} \Omega_2$$

其中， $\Omega_2 = \mathbf{U}_{M-3} \cdots \mathbf{U}_1$ 。經由重複循環此程序，可以得到  $\mathbf{P} = \mathbf{I}$  之  $\bar{\mathbf{q}}_M$ 。再從方程式(14)可以依此找出  $y'_M$ 。

在步驟 420 中，通道矩陣中另一行之  $y'_M$  可以在步驟 410 被執行後快速被找出。舉例來說，假設通道矩陣行

之排序是要將  $\bar{h}_{M-1}$  及  $\bar{h}_M$  交換為  $[\bar{h}_1 \cdots \bar{h}_M \bar{h}_{M-1} \bar{h}_{M+1} \cdots \bar{h}_N]$ 。可以看出，只有  $U_{M-1}$  需要進行計算。這個概念可以用來找出通道矩陣其他行排列的  $y_M$ 。最後，具有最大  $\xi$  的候選平面會被選為最初目標區塊。

上述之前置處理方法會對所有的  $\xi$  進行竭盡式搜尋。因此，當候選平面數目 ( $C(N, N-M+1)$ ) 很大時，運算的複雜度也很高。在本發明另一實施例中便提供了另一種可避免進行竭盡式搜尋之前置處理方法。

第 5 圖係為本發明另一實施例中基於幾何欠定系統解碼演算法之前置處理的步驟流程圖。在步驟 51 中，將向量集合  $V$  初始設定為  $V = \{\bar{h}_i, i=1, 2, \dots, N\}$ ，以及將參數  $k$  初始設定為 1。在步驟 52 中，找出對所有屬於  $V$  之中的  $\bar{h}_i$  其與  $y$  之間具有最小關聯值： $\bar{h}_j$  被選擇出來並經過正規化處理而設為  $\bar{q}_k$ 。在步驟 53 中，將  $\bar{h}_j$  自  $V$  中移除，然後將  $V$  更新為  $\bar{h}_i = \bar{h}_i - \bar{h}_i \cdot \bar{q}_k$ ，其中  $\forall i \neq j$  接著  $k$  被設定為  $k+1$ 。然後跳回步驟 52 直到  $\bar{q}_M$  被找出。最後再經由  $r_{jk} = \bar{q}_j \cdot \bar{h}_k$  來計算  $r_{jk}$  以得到重新排列後  $H$  之 QR 分解，其中  $j=1, 2, \dots, k-1$ 。

電腦模擬結果可以顯示出本發明之解碼演算法較 SSD 具有更低的解碼複雜度，並可以具備近似最大似然之性能。本發明中的前置處理方法更可進一步降低本發明解碼器之解碼複雜度，尤其是在信號點分佈區域較大時。在 64-QAM、4x3MIMO 中，本發明解碼器之解碼複雜度可以較 SSD 節省達 76%。若同時加入本發明之前置處理步驟，本發明解碼器之解碼複雜度可以較 SSD 節省

達 79%。

以上所述僅為舉例性，而非為限制性者。任何未脫離本發明之精神與範疇，而對其進行之等效修改或變更，均應包含於後附之申請專利範圍中。

**【圖式簡單說明】**

第 1 圖 係為本發明一實施例中基於幾何欠定系統解碼演算法之第一階段的步驟流程圖；

第 2 圖 係為本發明一實施例中基於幾何欠定系統解碼演算法之第二階段的步驟流程圖；

第 3 圖 顯示了具有不同  $y$  之平面的幾何圖；

第 4 圖 係為本發明一實施例中基於幾何欠定系統解碼演算法之前置處理的步驟流程圖；

第 5 圖 係為本發明另一實施例中基於幾何欠定系統解碼演算法之前置處理的步驟流程圖；

**【主要元件符號說明】**

略

## 七、申請專利範圍：

1. 一種基於幾何欠定系統之高效率解碼方法，包括以下步驟：

(1) 選定空間中任一點為一初始點  $x_{\text{int}} = [x_1^{\text{int}} \ x_2^{\text{int}} \ \dots \ x_N^{\text{int}}]^T \in \mathbb{R}^N$ ，且令  $X = x_{\text{int}}$ ，並將該初始點從  $X$  取出存於  $p$  並從  $X$  中刪除該點；

(2) 判斷  $X_v$  集合是否為一空集合且  $p$  值是否為 0，若是，則代表  $X_{\text{in}}$  集合內的所有點與相對應的  $\text{dist}$  即為所求，否則，執行步驟(3)；

(3) 選取任一個  $k$  值，而其中  $k$  指的是這個點尚未被搜尋過的維度，也就是它所對應的  $D_v$  向量中的第  $k$  個元素為「0」，並計算：

$$x_{k,\text{up}} = p_k - \frac{\Delta y(p)}{w_k} + C,$$

$$x_{k,\text{low}} = p_k - \frac{\Delta y(p)}{w_k} - C.$$

其中  $\Delta y(p) = \sum_{i=1}^N w_i p_i - y$ ，根據  $x_{k,\text{up}}, x_{k,\text{low}}$  的計算結果，將其間所有合法座標點一一產生出來；

(4) 判斷產生之該候選點是否存在於  $X_v$  內，若否，則將該候選點分別加入  $X$ 、 $X_v$  和  $X_{\text{in}}$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」，若是，則僅將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」；

(5) 透過先前計算所得的  $\Delta y(\bullet)$  來有效地計算出該候選點其相應的  $\text{dist}$  值；

(6) 根據  $x_{k,\text{up}}, x_{k,\text{low}}$  產生出對應的  $|x_{k,\text{up}} + d|, |x_{k,\text{low}} - d|$  兩個候選點，其中  $d$  為座標點之間之最小距離；

- (7) 判斷該些候選點是否存在於  $X_v$  之內，若否，則將其分別加入  $X$  和  $X_v$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」，若是，則僅將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」；
- (8) 判斷該相應的  $D_v$  向量內的所有元素是否都為「1」，若是，則執行步驟(9)，若否，則再判斷該  $p$  值是否滿足目前的搜尋條件，若否，則執行步驟(9)，若否，則跳回步驟(2)繼續搜尋；以及
- (9) 從  $X$  中任取出一個候選點取代目前的  $p$  並將取出的候選點從  $X$  中刪除。

2. 一種基於幾何欠定系統之高效率解碼方法，包括以下步驟：

- (1) 將一參數  $i$  設定為  $(M-1)$ ，並經由一 Slab  $M$  方程式  $-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C$  找到一最初候選點集合  $(\hat{x}_{i+1})$ ，並得出其相對之一距離  $(\text{dist}_{i+1})$ ；
- (2) 若  $i=1$  成立，則執行步驟(4)，否則，經由下列條件找出相對於候選點集合中每一候選點之  $x_{i-1}$  的上限  $(x_{i-1,up})$  與下限  $(x_{i-1,low})$ ：

$$D_{i-1} = \left\| \begin{bmatrix} y_{i-1} \\ y_i \end{bmatrix} - \begin{bmatrix} r_{i-1,i-1} & r_{i-1,i} & \dots & r_{i-1,N} \\ 0 & r_{i,i} & \dots & r_{i,N} \end{bmatrix} \begin{bmatrix} x_{i-1} \\ x_i \\ \vdots \\ x_N \end{bmatrix} \right\|^2 \leq C^2$$

且

$$x_{i-1,up} = \frac{\sqrt{C^2 - (dist_i)^2} + (y_{i-1} - \sum_{j=i}^N r_{i-1,j} x_j)}{r_{i-1,i-1}},$$

$$x_{i-1,low} = \frac{-\sqrt{C^2 - (dist_i)^2} + (y_{i-1} - \sum_{j=i}^N r_{i-1,j} x_j)}{r_{i-1,i-1}}$$

並經由下列公式得到  $dist_{i-1}$  :

$$dist_{i-1}^2 = \left( y_{i-1} - \sum_{j=i-1}^N r_{i-1,j} x_j \right)^2 + (dist_i)^2 . ;$$

(3) 使用相對於該每一個候選點之上限與下限建立一新候選點集合 ( $\hat{x}_i$ ) :

$$\hat{x}_{i-1} = \left[ \left[ x_{i-1,low}, x_{i-1,up} \right] \cap \mathbf{Z}, x_i, \dots, x_N \right]$$

並選擇一與  $\hat{x}_{i-1}$  具有最小距離之候選點，並得出其對應的 ZF-SIC 之解，並得到一新的半徑 ( $C_{new}$ )，再根據半徑 ( $C_{new}$ ) 及一初始值為 increment 之內部控制變數 state 對 C 進行更新調整，並將  $i$  設定為  $i-1$  後執行步驟(2)；以及

(4) 將與  $\hat{x}_i$  具有一最小距離之候選點選擇為  $x$  之一估計。

3.如申請專利範圍第 1 及 2 項所述之解碼方法，其中更包括將一通道矩陣  $\mathbf{H}$  重新排列。

4. 一種基於幾何欠定系統之高效率解碼方法，包括以下步驟：

(1)對一通道矩陣  $\mathbf{H}$  經由以下公式進行 QR 分解：

$$\mathbf{V}_1 \mathbf{H} = \begin{bmatrix} \mathbf{T}_1 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_1 \end{bmatrix}$$

其中， $\mathbf{V}_1$  係  $\mathbf{H}$  之一第一行向量之  $M \times M$  之 Householder 矩陣， $\mathbf{T}_1$  係一比例值而  $\tilde{\mathbf{H}}_1 = \mathbf{V}_1[2:M, 1:M] \mathbf{H}[1:M, 2:M]$ ，再定義一新矩陣  $\tilde{\mathbf{H}}_1 \in \mathbf{R}^{(M-1) \times (M-1)}$  以再進行：

$$\mathbf{V}_2 \tilde{\mathbf{H}}_1 = \begin{bmatrix} \mathbf{T}_2 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_2 \end{bmatrix}$$

其中， $\mathbf{V}_2$  係  $\tilde{\mathbf{H}}_1$  之一第一行向量之  $(M-1) \times (M-1)$  之 Householder 矩陣， $\mathbf{T}_2$  係一比例值而  $\tilde{\mathbf{H}}_2 = \mathbf{V}_2[2:M-1, 1:M-1] \tilde{\mathbf{H}}_1[1:M-1, 2:M-1]$ ，依此類推進行可以得到  $\mathbf{H}$  的 QR 分解為  $\mathbf{Q}^T \mathbf{H} = \mathbf{R}$ ，其中

$$\mathbf{Q}^T = \begin{bmatrix} \bar{\mathbf{q}}_1^T \\ \bar{\mathbf{q}}_2^T \\ \vdots \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \mathbf{U}_{M-1} \cdots \mathbf{U}_2 \mathbf{U}_1 \quad \text{且} \quad \mathbf{U}_i = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_i \end{bmatrix};$$

(2) 計算  $\mathbf{U}_{M-1}$  之最後一列而得到：

$$\begin{bmatrix} \otimes \\ \vdots \\ \otimes \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes \\ \mathbf{0} & v_{M-1(2,1)} & v_{M-1(2,2)} \end{bmatrix} \Omega_1$$

其中  $\Omega_1 = \mathbf{U}_{M-2} \cdots \mathbf{U}_1$ ，再依以下計算  $\mathbf{U}_{M-2}$  的最後兩列：



$$\Omega_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes & \otimes \\ \mathbf{0} & v_{M-2(2,1)} & v_{M-2(2,2)} & v_{M-2(2,3)} \\ \mathbf{0} & v_{M-2(3,1)} & v_{M-2(3,2)} & v_{M-2(3,3)} \end{bmatrix} \Omega_2$$

其中， $\Omega_2 = \mathbf{U}_{M-3} \cdots \mathbf{U}_1$ ，經由依此類推進行計算可以得到  $\mathbf{P}=\mathbf{I}$  之  $\bar{\mathbf{q}}_M$ ，再依此找出  $y'_M$ ；以及

(3) 找出該通道矩陣  $\mathbf{H}$  中另一行之  $y'_M$  並將具有最大  $\xi$  之候選平面選為該最初目標區塊。

5. 一種基於幾何欠定系統之高效率解碼方法，包括以下步驟：

(1) 將一向量集合  $\mathbf{V}$  初始設定為  $\mathbf{v} = \{\bar{\mathbf{h}}_i, i=1, 2, \dots, N\}$ ，並將一參數  $k$  初始設定為 1；

(2) 找出對所有屬於  $\mathbf{V}$  之中的  $\bar{\mathbf{h}}_i$  其與  $y$  之間具有最小關聯值： $\bar{\mathbf{h}}_j$  被選擇出來並經過正規化處理而設為  $\bar{\mathbf{q}}_k$ ；

(3) 將  $\bar{\mathbf{h}}_j$  自該向量集合  $\mathbf{V}$  中移除，並將該向量集合  $\mathbf{V}$  更新為  $\bar{\mathbf{h}}_i = \bar{\mathbf{h}}_i - \bar{\mathbf{h}}_i \cdot \bar{\mathbf{q}}_k$ ，其中  $\forall i \neq j$  且  $k$  被設定為  $k+1$ ；

(4) 重複執行步驟(2)直至  $\bar{\mathbf{q}}_M$  被找出；以及

(5) 經由  $r_{jk} = \bar{\mathbf{q}}_j \cdot \bar{\mathbf{h}}_k$  計算  $r_{jk}$  以得到重新排列後  $\mathbf{H}$  之 QR 分解，其中  $j=1, 2, \dots, k-1$ 。

6. 一種基於幾何欠定系統之高效率解碼器，該解碼器執行以下步驟：

(1) 選定空間中任一點為一初始點  $x_{int} = [x_1^{int} \ x_2^{int} \ \dots \ x_N^{int}]^T \in \mathbb{R}^N$ ，且令  $X = x_{int}$ ，並將該初始點從  $X$  取出存於  $p$  並從  $X$  中刪除該點；

(2) 判斷  $X_v$  集合是否為一空集合且  $p$  值是否為  $0$ ，若是，則代表  $X_{in}$  集合內的所有點與相對應的  $dist$  即為所求，否則，執行步驟(3)；

(3) 選取任一個  $k$  值，而其中  $k$  指的是這個點尚未被搜尋過的維度，也就是它所對應的  $D_v$  向量中的第  $k$  個元素為「0」，並計算：

$$x_{k,up} = p_k - \frac{\Delta y(p)}{w_k} + C,$$

$$x_{k,low} = p_k - \frac{\Delta y(p)}{w_k} - C.$$

其中  $\Delta y(p) = \sum_{i=1}^N w_i p_i - y$ ，根據  $x_{k,up}, x_{k,low}$  的計算結果，將其間所有合法座標點一一產生出來；

(4) 判斷產生之該候選點是否存在於  $X_v$  內，若否，則將該候選點分別加入  $X$ 、 $X_v$  和  $X_{in}$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」，若是，則僅將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」；

(5) 透過先前計算所得的  $\Delta y(\bullet)$  來有效地計算出該候選點其相應的  $dist$  值；

(6) 根據  $x_{k,up}, x_{k,low}$  產生出對應的  $[x_{k,up} + d], [x_{k,low} - d]$  兩個候選點，其中  $d$  為座標點之間之最小距離；

(7) 判斷該些候選點是否存在於  $X_v$  之內，若否，則將其分別加入  $X$  和  $X_v$  之中，同時並將其所對應  $D_v$  向量中的第  $k$  個元素設為「1」，若是，則僅將其所

對應  $D_v$  向量中的第  $k$  個元素設為「1」；

(8) 判斷該相應的  $D_v$  向量內的所有元素是否都為「1」，若是，則執行步驟(9)，若否，則再判斷該  $p$  值是否滿足目前的搜尋條件，若否，則執行步驟(9)，若否，則跳回步驟(2)繼續搜尋；以及

(9) 從  $X$  中任取出一個候選點取代目前的  $p$  並將取出的候選點從  $X$  中刪除。

7.如申請專利範圍第 6 項所述之解碼器，其中更執行以下步驟：

(1)將一參數  $i$  設定為  $(M-1)$ ，並經由一 Slab  $M$  方程式  $-C \leq y'_M - [r_{M,M}x_M + \dots + r_{M,N}x_N] \leq C$  找到一最初候選點集合  $(\hat{x}_{i+1})$ ，並得出其相對之一距離  $(dist_{i+1})$ ；

(2) 若  $i=1$  成立，則執行步驟(4)，否則，經由下列條件找出相對於候選點集合中每一候選點之  $x_{i-1}$  的上限  $(x_{i-1,up})$  與下限  $(x_{i-1,low})$ ：

$$D_{i,i-1} = \left\| \begin{bmatrix} y'_{i-1} \\ y'_i \end{bmatrix} - \begin{bmatrix} r_{i-1,i-1} & r_{i-1,i} & \dots & r_{i-1,N} \\ 0 & r_{i,i} & \dots & r_{i,N} \end{bmatrix} \begin{bmatrix} x_{i-1} \\ x_i \\ \vdots \\ x_N \end{bmatrix} \right\|^2 \leq C^2$$

且

$$x_{i-1,up} = \frac{\sqrt{C^2 - (dist_i)^2} + (y'_{i-1} - \sum_{j=i}^N r_{i-1,j} x_j)}{r_{i-1,i-1}},$$

$$x_{i-1,low} = \frac{-\sqrt{C^2 - (dist_i)^2} + (y'_{i-1} - \sum_{j=i}^N r_{i-1,j} x_j)}{r_{i-1,i-1}}$$

並經由下列公式得到  $dist_{i-1}$  :

$$dist_{i-1}^2 = \left( y_{i-1}' - \sum_{j=i-1}^N r_{i-1,j} x_j \right)^2 + (dist_i)^2 . ;$$

(3) 使用相對於該每一個候選點之上限與下限建立一新候選點集合 ( $\hat{x}_i$ ) :

$$\hat{x}_{i-1} = \left[ \lfloor x_{i-1,low}, x_{i-1,up} \rfloor \cap \mathbf{Z}, x_i, \dots, x_N \right]$$

並選擇一與  $\hat{x}_{i-1}$  具有最小距離之候選點，並得出其對應的 ZF-SIC 之解，並得到一新的半徑 ( $C_{new}$ )，再根據半徑 ( $C_{new}$ ) 及一初始值為 increment 之內部控制變數 state 對 C 進行更新調整，並將  $i$  設定為  $i-1$  後執行步驟(2)；以及

(4) 將與  $\hat{x}_i$  具有一最小距離之候選點選擇為  $x$  之一估計。

8.如申請專利範圍第 7 項所述之解碼器，其中該解碼器係將一通道矩陣  $\mathbf{H}$  重新排列。

9.如申請專利範圍第 8 項所述之解碼器，其中對該通道矩陣  $\mathbf{H}$  重新排列係經由以下步驟進行：

(1)對該通道矩陣  $\mathbf{H}$  經由以下公式進行 QR 分解：

$$\mathbf{V}_1 \mathbf{H} = \begin{bmatrix} \mathbf{T}_1 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_1 \end{bmatrix}$$

其中， $\mathbf{V}_1$  係  $\mathbf{H}$  之一第一行向量之  $M \times M$  之 Householder

矩陣， $\mathbf{T}_1$  係一比例值而  $\tilde{\mathbf{H}}_1 = \mathbf{V}_1[2:M, 1:M]\mathbf{H}[1:M, 2:M]$ ，再定義一新矩陣  $\tilde{\mathbf{H}}_1 \in \mathbf{R}^{(M-1) \times (M-1)}$  以再進行：

$$\mathbf{V}_2 \tilde{\mathbf{H}}_1 = \begin{bmatrix} \mathbf{T}_2 & \mathbf{A} \\ \mathbf{0} & \tilde{\mathbf{H}}_2 \end{bmatrix}$$

其中， $\mathbf{V}_2$  係  $\tilde{\mathbf{H}}_1$  之一第一行向量之  $(M-1) \times (M-1)$  之 Householder 矩陣， $\mathbf{T}_2$  係一比例值而  $\tilde{\mathbf{H}}_2 = \mathbf{V}_2[2:M-1, 1:M-1]\tilde{\mathbf{H}}_1[1:M-1, 2:M-1]$ ，依此類推進行可以得到  $\mathbf{H}$  的 QR 分解為  $\mathbf{Q}^T \mathbf{H} = \mathbf{R}$ ，其中

$$\mathbf{Q}^T = \begin{bmatrix} \bar{\mathbf{q}}_1^T \\ \bar{\mathbf{q}}_2^T \\ \vdots \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \mathbf{U}_{M-1} \cdots \mathbf{U}_2 \mathbf{U}_1 \quad \text{且} \quad \mathbf{U}_i = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_i \end{bmatrix};$$

(2) 計算  $\mathbf{U}_{M-1}$  之最後一列而得到：

$$\begin{bmatrix} \otimes \\ \vdots \\ \otimes \\ \bar{\mathbf{q}}_M^T \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes \\ \mathbf{0} & \nu_{M-1(2,1)} & \nu_{M-1(2,2)} \end{bmatrix} \Omega_1$$

其中  $\Omega_1 = \mathbf{U}_{M-2} \cdots \mathbf{U}_1$ ，再依以下計算  $\mathbf{U}_{M-2}$  的最後兩列：

$$\Omega_1 = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \otimes & \otimes & \otimes \\ \mathbf{0} & \nu_{M-2(2,1)} & \nu_{M-2(2,2)} & \nu_{M-2(2,3)} \\ \mathbf{0} & \nu_{M-2(3,1)} & \nu_{M-2(3,2)} & \nu_{M-2(3,3)} \end{bmatrix} \Omega_2$$

其中， $\Omega_2 = \mathbf{U}_{M-3} \cdots \mathbf{U}_1$ ，經由依此類推進行計算可以得到  $\mathbf{P} = \mathbf{I}$  之  $\bar{\mathbf{q}}_M$ ，再依此找出  $y'_M$ ；以及

(3) 找出該通道矩陣  $\mathbf{H}$  中另一行之  $y'_M$  並將具有最大  $\xi$  之候選平面選為該最初目標區塊。

10. 如申請專利範圍第 8 項所述之解碼器，其中對該通道矩陣  $\mathbf{H}$  重新排列係經由以下步驟進行：

(1) 將一向量集合  $\mathbf{V}$  初始設定為  $\mathbf{v} = \{\bar{\mathbf{h}}_i, i=1, 2, \dots, N\}$ ，並將一參數  $k$  初始設定為 1；

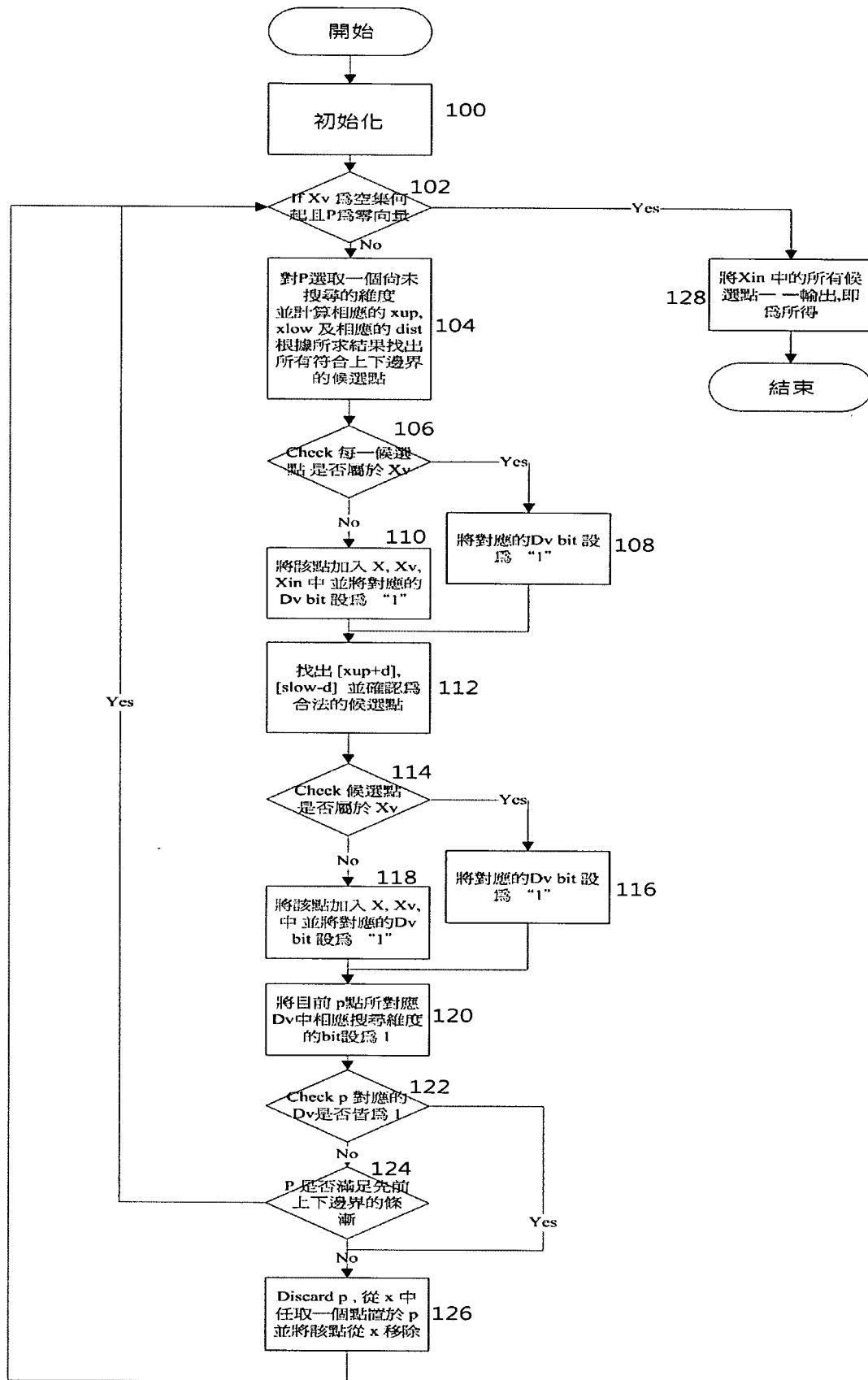
(2) 找出對所有屬於  $\mathbf{V}$  之中的  $\bar{\mathbf{h}}_i$  其與  $y$  之間具有最小關聯值： $\bar{\mathbf{h}}_j$  被選擇出來並經過正規化處理而設為  $\bar{\mathbf{q}}_k$ ；

(3) 將  $\bar{\mathbf{h}}_j$  自該向量集合  $\mathbf{V}$  中移除，並將該向量集合  $\mathbf{V}$  更新為  $\bar{\mathbf{h}}_i = \bar{\mathbf{h}}_i - \bar{\mathbf{h}}_i \cdot \bar{\mathbf{q}}_k$ ，其中  $\forall i \neq j$  且  $k$  被設定為  $k+1$ ；

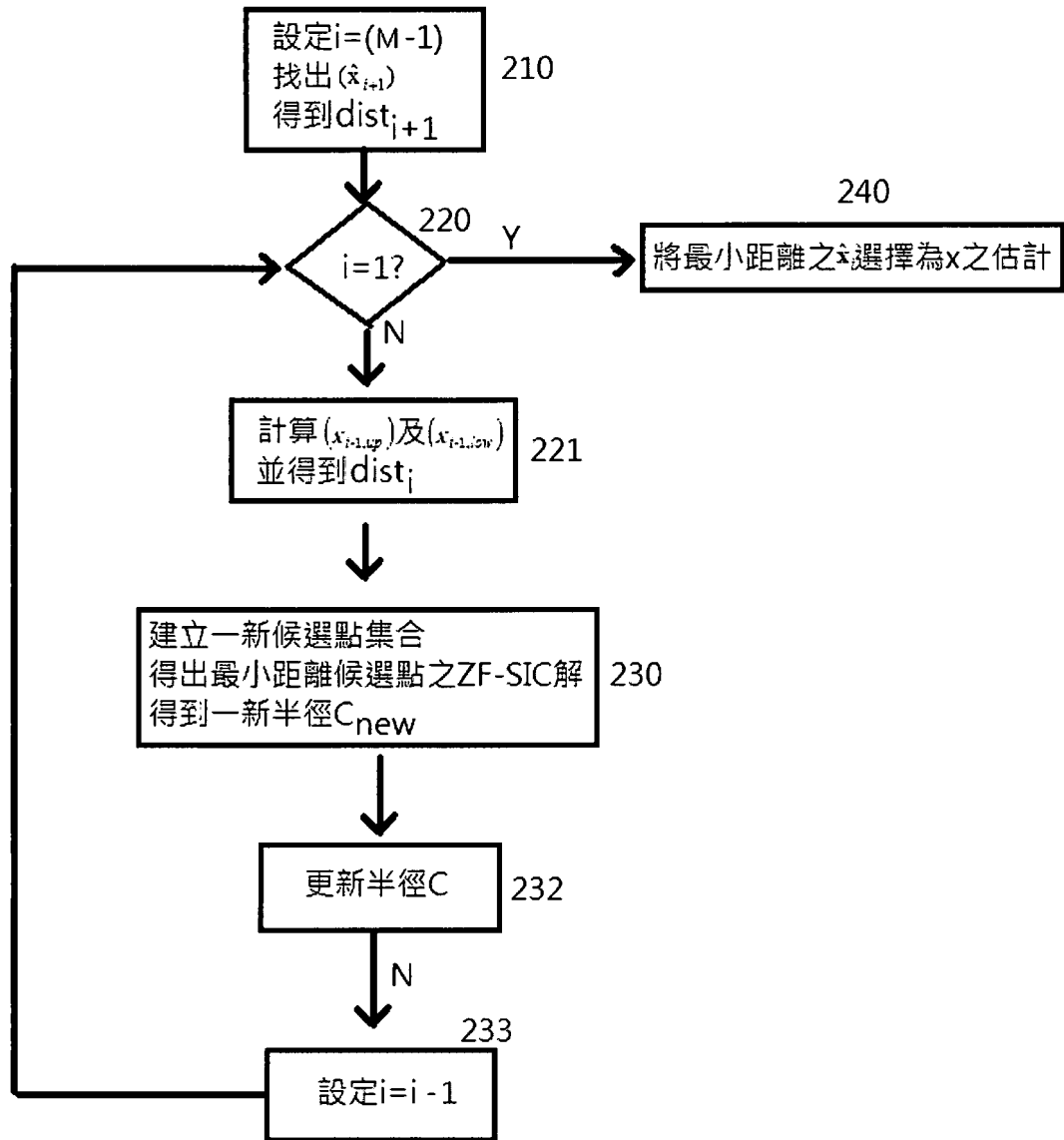
(4) 重複執行步驟(2)直至  $\bar{\mathbf{q}}_M$  被找出；以及

(5) 經由  $r_{jk} = \bar{\mathbf{q}}_j \cdot \bar{\mathbf{h}}_k$  計算  $r_{jk}$  以得到重新排列後  $\mathbf{H}$  之 QR 分解，其中  $j=1, 2, \dots, k-1$ 。

八、圖式：

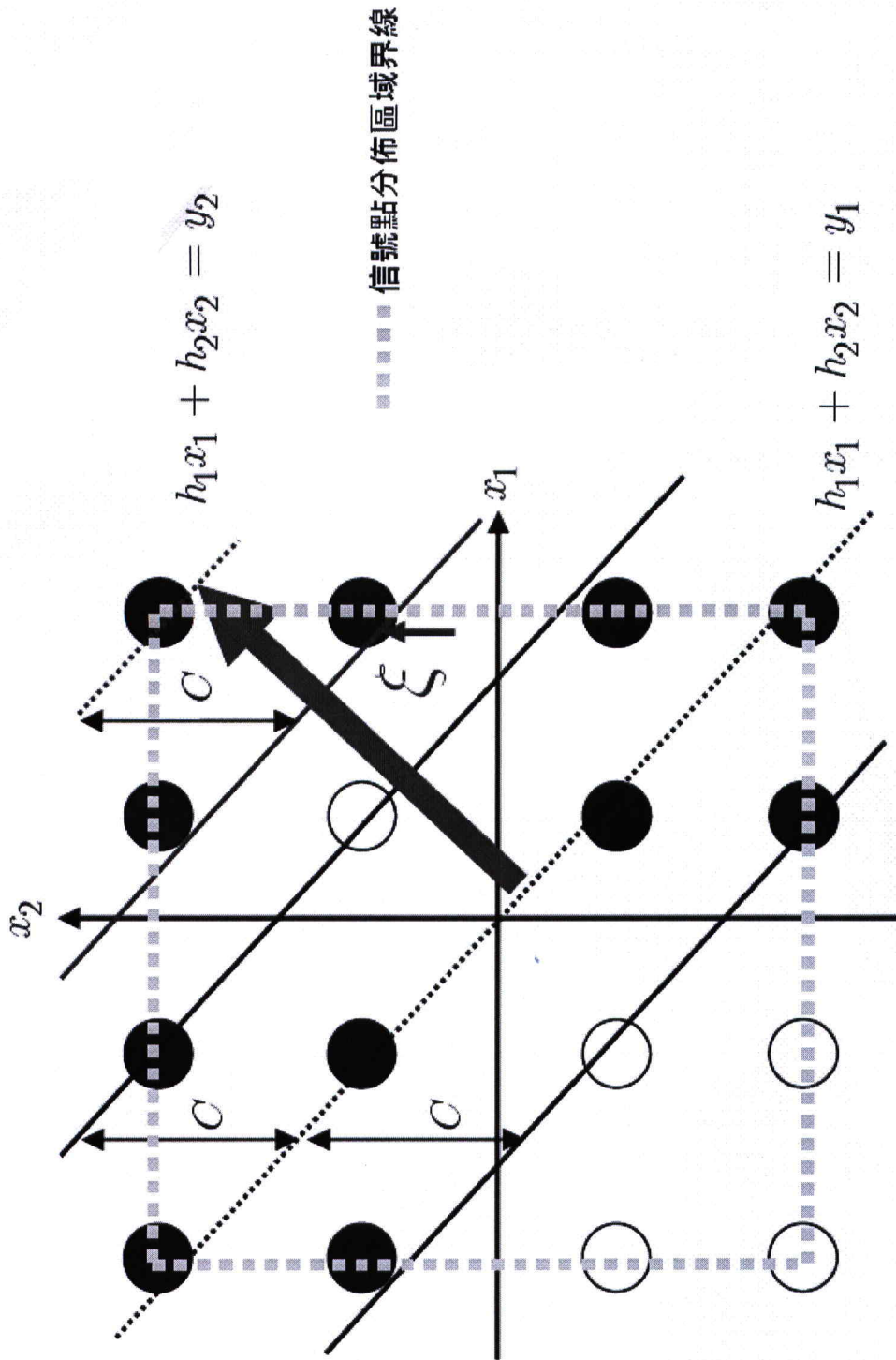


第 1 圖

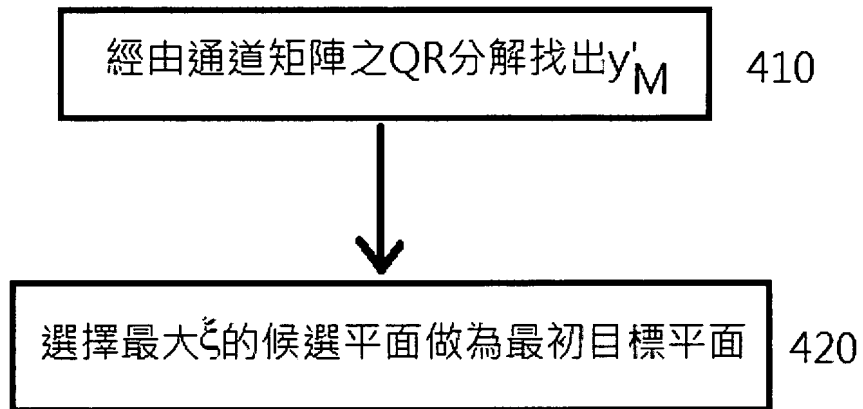


第 2 圖

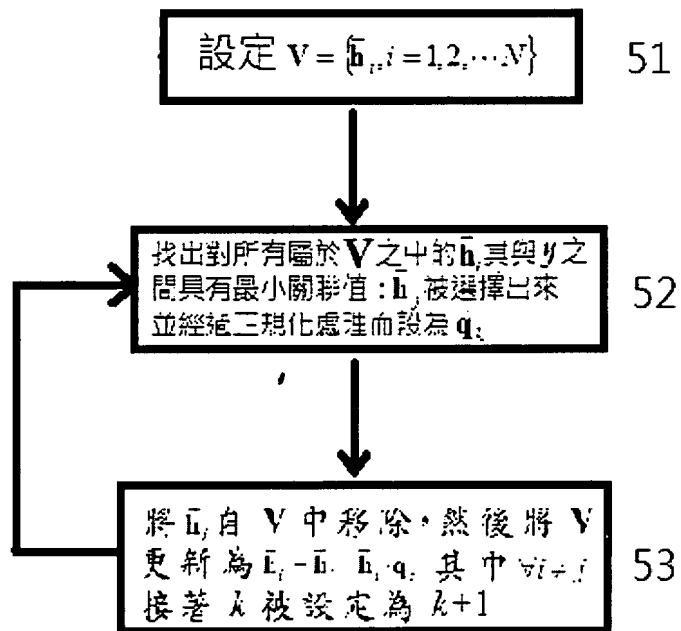




第 3 圖



第 4 圖



第 5 圖