



(19) 中華民國智慧財產局

(12) 發明說明書公告本

(11) 證書號數：TW I434536 B

(45) 公告日：中華民國 103 (2014) 年 04 月 11 日

(21) 申請案號：098123659

(22) 申請日：中華民國 98 (2009) 年 07 月 13 日

(51) Int. Cl. : **H04B7/08 (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 100 號

(72) 發明人：黃崇榮 HUANG, CHUNG JUNG (TW)；宋志晟 SUNG, CHIH SHENG (TW)；李大嵩 LEE, T. S. (TW)

(74) 代理人：蔡清福

(56) 參考文獻：

US 2008/0240277A1

US 2008/0298478A1

WO 2005/076491A1

L. G. Barbero and J. S. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," IEEE Transactions on Wireless Communications, pp. 2131-2142, June 2006. ^&rn^

審查人員：易志孝

申請專利範圍項數：5 項 圖式數：9 共 39 頁

(54) 名稱

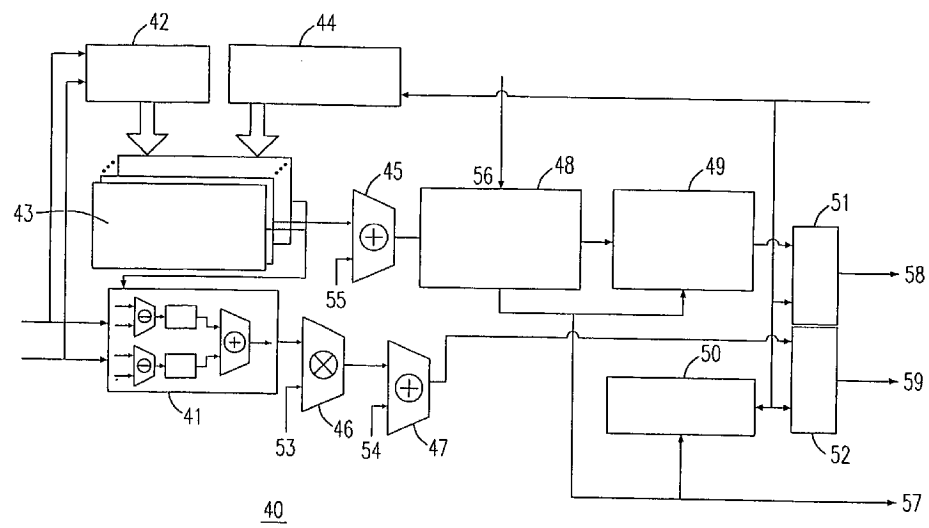
通訊系統接收器及其解碼方法

A RECEIVER IN COMMUNICATION SYSTEM AND DECODING METHOD THEREOF

(57) 摘要

針對 K-Best 演算法的缺失，提出一個適用於大型積體電路架構的高效率球型 K-Best 解碼器。透過複數平面星座圖的規律特性來簡化星座點之搜尋，以及提出一高速平行比較之架構，省卻大量的資料排序動作及節點的運算量，藉以提供一高效率，低功率消耗且具有固定吞吐量的 K-Best 解碼器。更進一步地，透過結合部份最大可能決策器與所提出的高效率搜尋演算法，使得在相同的環境及 K 值下，該系統能提供接近於最大可能搜尋之性能，且較傳統的 K-Best 演算法有更低的運算量。

In this invention, we propose a VLSI architecture of the modified K-Best sphere decoding algorithm with high-efficiency and fixed throughput. By the regular and symmetric properties of coordination in complex plane, an efficient symbol search method in complex plane is proposed. We also propose a high-speed and highly parallelism comparison architecture to reduce the computational complexity and sorting burden. Furthermore, combining the proposed near-ML search strategy and the efficient sorting architecture, the proposed K-Best sphere decoder has better performance and lower complexity than the conventional K-Best sphere decoder. The apparatus has high-efficiency, low-power characteristics and especially appropriates for personal wireless communication systems.



第五圖(a)

- 40 . . . 複數平面路徑計算模組及候選點產生器(CCPG)
- 41 . . . 路徑計算器
- 42 . . . 群組記憶體
- 43 . . . 子節點記憶體
- 44,50 . . . 上數計數器
- 45 . . . 座標轉移器
- 46 . . . 權重調整器
- 47 . . . 父節點權重器
- 48 . . . 邊界檢查器
- 49 . . . 象限座標還原器

發明專利說明書

(本說明書格式、順序，請勿任意更動，※記號部分請勿填寫)

※申請案號：98123659

※申請日：98.7.13

※IPC 分類：H04B 7/08 (2006.01)

一、發明名稱：(中文/英文)

通訊系統接收器及其解碼方法/A RECEIVER IN
COMMUNICATION SYSTEM AND DECODING METHOD
THEREOF

二、中文發明摘要：

針對 K-Best 演算法的缺失，提出一個適用於大型積體電路架構的高效率球型 K-Best 解碼器。透過複數平面星座圖的規律特性來簡化星座點之搜尋，以及提出一高速平行比較之架構，省卻大量的資料排序動作及節點的運算量，藉以提供一高效率，低功率消耗且具有固定吞吐量的 K-Best 解碼器。更進一步地，透過結合部份最大可能決策器與所提出的高效率搜尋演算法，使得在相同的環境及 K 值下，該系統能提供接近於最大可能搜尋之性能，且較傳統的 K-Best 演算法有更低的運算量。

三、英文發明摘要：

In this invention, we propose a VLSI architecture of the modified K-Best sphere decoding algorithm with high-efficiency and fixed throughput. By the regular and symmetric properties of coordination in complex plane, an efficient symbol search method in complex plane is proposed. We also propose a high-speed and highly parallelism comparison architecture to reduce the computational complexity and sorting burden. Furthermore, combining the proposed near-ML search strategy and the efficient sorting architecture, the

proposed K-Best sphere decoder has better performance and lower complexity than the conventional K-Best sphere decoder. The apparatus has high-efficiency, low-power characteristics and especially appropriates for personal wireless communication systems.

四、指定代表圖：

(一)本案指定代表圖為：第(五)圖(a)。

(二)本代表圖之元件符號簡單說明：

40：複數平面路徑計算模組及候選點產生器(CCPG)

41：路徑計算器

42：群組記憶體

43：子節點記憶體

44, 50：上數計數器

45：座標轉移器

46：權重調整器

47：父節點權重器

48：邊界檢查器

49：象限座標還原器

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

色點代表接收點，顯示一接收信號(向量 y 的元素)在具有複數個晶格點的 $2N$ 維度的空間中的位置，每一個晶格點代表傳送點(信號)的座標。在第一圖中的箭頭表示一 $2N$ 維度超球面的半徑。箭頭的長度代表一預設值，藉由計算在 $2N$ 維度超球面的每一個晶格點與該接收點之間的距離，找出具有最小距離的候選點 即為該問題的解。

在 IEEE 的論文中，論文名稱 A Novel Approach for K-Best MIMO Detection and its VLSI Implementation，提出一種尋找候選點的方法。對每一個二維平面的搜尋是利用同心圓慢慢地去擴大半徑。請參閱第二圖，其為習知尋找候選點的示意圖。尋找候選點的方法以搜尋之中心點為圓心，一個初始半徑值，看看那些點會落在該圓內，即以初始半徑值當作有效傳送(信號)點的路徑值。接下來再一一將半徑慢慢擴大，直到擴大半徑的圓包圍到在複數平面上所有的有效的傳送(信號)點為止。在此過程中，當擴大半徑的圓碰觸至有效傳送(信號)點時，此時的半徑值被視為有效傳送(信號)點的路徑值。但是，此法有著相當大的缺點：(1)對一個圓而言，取樣點必須夠多，才能在所有的搜尋範圍內，有效率地接觸到所有的點。但對每個半徑而言，都需對同心圓上的每一個取樣點，進行座標的計算及比對。這樣的運算量相當的龐大。(2)路徑的計算值並不是精確的值，尤其是當半徑慢慢地變大時，會特別地明顯。(3)這個方法，若取的有效傳送(信號)點個數較多時，為了能碰觸到所有的合法訊號點，所取的取樣點必需足夠的多，而且會發生找到重覆的點，或遺漏合法點的現象。。

在球型解碼器中較為熟知的兩種搜尋演算法，也就是 Schnorr-Eucher(SE)列舉及 K-Best 法，為最廣泛被採用的方

法。但是，目前的方法找尋在複數平面上的最佳傳送點既不經濟且不可靠。

【發明內容】

為了克服以上種種的缺失，吾人提出一個適用於大型積體電路架構的高效率 K-Best 球型解碼器。透過複數平面星座圖的規律特性來簡化候選點之搜尋，以及提出一高速平行比較之架構，以省卻大量的資料排序動作及節點路徑值的運算量，藉以提供一高效率、降低功率消耗、及具有固定吞吐量的 K-Best 解碼器。更進一步地，透過結合部份最大可能決策器與所提出的高效率搜尋演算法，使得在相同的環境及 K 值下，該系統能提供接近於最大可能搜尋之性能，且較傳統的 K-Best 接收機有更低的運算量

本案的發明目的在於透過複數平面星座圖的規律特性來簡化候選點之搜尋，以省卻大量的資料排序動作及節點的運算量，並結合最大相似搜尋決策器提供一高性能、高效率、低功率消耗、及具有接近固定吞吐量的 K-Best 解碼器。

依照上述構想，一種針對在一通訊系統中的一接收點從複數個傳送點之中找出一特定數目具有最小累積路徑值的通訊點的方法被提出，該方法包含下列步驟：(a)在一複數平面上定義各該複數個傳送點與該接收點的一第一座標。(b)將該接收點相應的該第一座標轉換到相應的一第二座標，其中該接收點位於該複數平面的一已知區域。(c)依據該接收點的該第二座標找出該特定數目具有最小累積路徑值的傳送點。

平行計算器

根據上述構想，一種用以產生一特定數目具有相對最小累積路徑值的通訊候選節點的方法被提出，該方法包含下列步

驟：(a)提供複數個序列，其各自具有一累積路徑值的候選節點集合，且各序列根據該候選節點集合的該累積路徑值遵循一漸增的順序來排列。(b)處理該複數個序列，以找出該特定數目具有相對最小累積路徑值的通訊候選節點。

較佳地，其中在該步驟(b)中更包含下列子步驟：(b1)根據各該複數序列的該漸增順序分別輸入至少一候選節點至一平行處理器。(b2)利用該平行處理器找出至少一具有該相對最小累積路徑值的候選節點。(b3)輸入在該步驟(b2)所找出位在該候選節點的同一個序列的下一個候選節點進入該平行處理器。(b4)重複該步驟(b1)到該步驟(b3)直到該特定數目具有該相對最小累積路徑值的候選節點被找到。

較佳地，其中該平行處理器包含複數個輸入，且當該複數個序列的總數超出該平行處理器的輸入總數時，該方法更包含下列步驟：(a1)將該複數個序列分成複數個群組，其中各該群組序列的總數等於或小於該平行處理器的該輸入總數。(a2)對各該群組重複該步驟(b1)到該步驟(b4)，直到在各該群組中該特定數目具有該相對最小累積路徑值的候選節點被找到。(a3)將該複數個群組轉化成該複數個序列，各該序列包含該特定數目在各該群組中具有該最小累積路徑值的該等候選節點。

較佳地，其中該平行處理器包含 j 個輸入，該 j 為一偶整數，且當該複數序列為兩序列時，該步驟(b)包含下列步驟：(bb1)將在該兩序列中的一第一序列依據該由小到大順序輸入前 k 個候選節點到該平行處理器的一奇數位置的輸入，其中 $k=j/2$ 。(bb2)將在該兩序列中的一第二序列根據該漸增順序輸入前 k 個候選節點到該平行處理器的一偶數位置的輸入。

根據上述構想，

一種在一 MIMO 通訊系統中將 K-Best 最佳化的方法，該方法包含下列步驟：對該 MIMO 通訊系統的一通道矩陣執行一 QR 分解以獲得一具有複數個對角線元素的矩陣。基於該對角線元素的

值決定是否執行一 K-Best 演算法或最大相似搜尋。

藉由上述通訊系統中產生候選點的方法，俾能達到簡化候選點之搜尋，以省卻大量的資料排序動作及節點的運算量，藉以提供一高效率、降低功率消耗、及具有固定吞吐量的 K-Best 解碼器。其他的特徵及功效，可參閱過後文的實施方式後便能得到更進一步的了解。

【實施方式】

請參閱第三圖，其為本案所提具有 P 個狀態的搜尋樹的示意圖。本案較佳的實施例以兩個狀態為例，各種調變方法也都可以適用。該搜尋樹包含一節點 20、一節點 21、一節點 22、一節點 23、一節點 24、一節點 25、一節點 26、一節點 27、一節點 28、一節點 29、一節點 30、一節點 31、一節點 32、一節點 33、一節點 34、一傳送信號集合 213、一傳送信號集合 223、一傳送信號集合 233、一傳送信號集合 243、一傳送信號集合 253、一傳送信號集合 263、一傳送信號集合 273、一傳送信號集合 283、一傳送信號集合 293、一傳送信號集合 303、一傳送信號集合 313、一傳送信號集合 323、一傳送信號集合 333、一傳送信號集合 343、一父節點到子結點距離 201(假設其路徑值為 1)、一父節點到子結點距離 202(假設其路徑值為 2)、一父節點到子結點距離 211(假設其路徑值為 1)、一父節點到子結點距離 212(假設其路徑值為 2)、一父節點到子結點距離 221(假設其路徑值為 2)、一父節點到子結點距離 222(假設其路徑值為 1)、一父節點到子結點距離 231(假設其路徑值為 0)、一父節點到子結點距離 232(假設其路徑值

為 2)、一父節點到子結點距離 241(假設其路徑值為 1)、一父節點到子結點距離 242(假設其路徑值為 3)、一父節點到子結點距離 251(假設其路徑值為 5)、一父節點到子結點距離 252(假設其路徑值為 0)、一父節點到子結點距離 261(假設其路徑值為 1)、一父節點到子結點距離 262(假設其路徑值為 2)、一節點累積路徑值 200(其值為 0)、一節點累積路徑值 210(其值為 $1=1+0$)、一節點累積路徑值 220(其值為 $2=2+0$)、一節點累積路徑值 230(其值為 $2=1+1$)、一節點累積路徑值 240(其值為 $3=1+2$)、一節點累積路徑值 250(其值為 $4=2+2$)、一節點累積路徑值 260(其值為 $3=2+1$)、一節點累積路徑值 270(其值為 $2=2+0$)、一節點累積路徑值 280(其值為 $4=2+2$)、一節點累積路徑值 290(其值為 $4=3+1$)、一節點累積路徑值 300(其值為 $6=3+3$)、一節點累積路徑值 310(其值為 $9=4+5$)、一節點累積路徑值 320(其值為 $4=4+0$)、一節點累積路徑值 330(其值為 $4=3+1$)、一節點累積路徑值 340(其值為 $5=3+2$)。

假設有三個傳送天線(未顯示)與三個接收天線(未顯示)，每個天線傳送一個相位調變(PSK)的信號，該相位調變信號包含兩種可能的相位為 S_1 與 S_2 ， x_1, x_2, x_3 分別代表第一、第二、及第三根傳送天線所傳送的值， y_1, y_2, y_3 分別代表第一、第二、及第三根接收天線所接收的值，則各傳送信號集合、各父節點到子結點的距離、及各節點路徑值可以用第三圖中的搜尋樹來表示。

在方程式 1 中，如果 $N = 3$ ，表示共有 3 個傳送信號透過對 H 進行 QR 分解後， $y = Hx + n$ 可轉換為 $z = Rx + v$ ，其中

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ 0 & r_{2,2} & r_{2,3} \\ 0 & 0 & r_{3,3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = Q^H \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = Q^H \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

z_1 、 z_2 、及 z_3 是 y_1 、 y_2 、及 y_3 經過 Q^H 的轉換，因為 y_1 、 y_2 、 y_3 是接收到的值， Q 是 H 經過 QR 分解的結果，所以很容易求出 z_1 、 z_2 、及 z_3 。 x_1 、 x_2 、 x_3 分別為該第一、第二、第三傳送天線傳送的值，且該第一、第二、第三傳送天線傳送的值不是 S_1 就是 S_2 。

在第三圖中，首先開始解該第三傳送天線傳送信號的值 x_3 ，因為該第三根傳送天線傳送的信號值為未知數，根據 QR 分解後，由方程式 $z_3 = r_{3,3}x_3 + v_3$ 可以很快計算出 x_3 的結果(因為 $z_3, r_{3,3}$ 都知道)，由於我們不知道 v_3 的值，所求的 x_3 會受到雜訊 v_3 的影響，如果 x_3 受雜訊的干擾而產生解碼錯誤，則後面解出來的 x_2 及 x_1 都會是有問題的。所以想看看 x_3 跟 S_1, S_2 那一個比較可能，一種基本的想法是“路徑值越小越有可能”。所以用路徑值(距離之平方)當判斷的依據。因為該節點 20 是在複數平面上，所以可以藉由直接代入幾何距離公式而予以計算。

此時計算 $\|z_3 - r_{3,3}S_1\|$ 與 $\|z_3 - r_{3,3}S_2\|$ 的距離，其距離分別為 1，2，所以將 1 的值填入該父節點到子節點的距離 201，將 2 的值填入該父節點到子節點的距離 202。

接下來要解該第二根傳送天線傳送的值為何，透過 QR 分解後的方程式 $z_2 = r_{2,2}x_2 + r_{2,3}x_3 + v_2$ ，對節點 23 及節點 24 而言，假設第三根傳送天線是送 S_1 ，所以把 x_3 用 S_1 代入上面方程式

$z_2 = r_{2,2}x_2 + r_{2,3}S_1 + v_3$ ，以算出 x_2 的值，然後再計算相應的距離，而分別得到 1 與 2。

因為路徑會走到節點 23，就表示是經過節點 21(認為第三根傳送天線傳送 S_1)，所以必須把節點 21 的影響考慮進來，因此節點 23 的累積路徑值 230 應該是節點 21 的累積路徑值 210 加上節點 21 到節點 23 的父節點到子節點的路徑值 211，也就是 $d^2(x_2, S_1)$ ，同時節點 23 的傳送信號集合 233 表示為 $[?, S_1, S_1]$ ，其中?表示未知，其為 S_1 或 S_2 的其中之一。因為不知道該第一根傳送天線送 S_1 或 S_2 ，但假設該第二、三根傳送信號都送 S_1 ，所以在節點 23 的累積路徑值 230 就填入 $2(= 1+1)$ 。

接下來考慮節點 24，因為路徑會走到節點 24，就表示是經過節點 21(該第三根傳送天線被認為傳送 S_1)，因此必須把節點 21 的影響考慮進來，所以節點 24 的累積路徑值 240 應該是節點 21 的累積路徑值 210 加上父節點到子節點的路徑值 212，也就是 $d^2(x_2, S_2)$ 。節點 24 的傳送信號集合 243 表示為 $[?, S_2, S_1]$ 。所以在節點 24 的累積路徑值 240 就填入 $3(= 1+2)$ 。

對節點 25 及節點 26 而言，假設第三根傳送天線是送 S_2 ，所以把 x_3 用 S_2 代入上面方程式($z_2 = r_{2,2}x_2 + r_{2,3}S_2 + v_3$)，算出 x_2 的值。目前路徑是到節點 25，因為會路徑走到節點 25，就表示是經過節點 22(認為第三根天線傳送 S_2)，所以必須把節點 22 的影響考慮進來，節點 25 的累積路徑值應該是節點 22 的累積路徑值 220 加上父節點到子節點的路徑值 221，所以在節點 25 的累積路徑值 250 就填入 $4(= 2+2)$ 。同時，在節點的傳送信號集合 253 就表示為 $[?, S_1, S_2]$ 。因為路徑會走到節點 26，

就表是是經過節點 22(認為第三根天線傳送 s_2)，所以必須把節點 22 的影響考慮進來，節點 26 的累積路徑值 260 應該是節點 22 的累積路徑值 220 加上父節點到子節點 222 的路徑值(=1)，所以節點 26 就填入 3(= 2+1)。同時，在節點 26 的傳送信號集合 263 表示為[?, S_2 , S_2]。

值得注意的是，在計算累積路徑值 230 與累積路徑值 250 時(節點 21→節點 23 和 節點 22→節點 25)看來公式好像一樣，但在決定 x_2 時，雖是同樣的方程式，但 x_3 卻代入不一樣的數值(s_1 或 s_2)，所以求出的 x_2 的值也會不一樣，且求出的距離平方也會不一樣。如此一直類推下去，便可將所有可能的情況都列出來，因為該三根接收天線接收的值和猜測的所有情況對應的值之中，那一個總體的距離最小，所以必需把這些距離值加起來，看最後總合誰最小，那就是我們要的。

這樣的方式很沒效率，所以在每一層將所有的點的累積路徑值都算出來，保留累積路徑值比較小的點，繼續往下找。

{ 節點21, 節點22} 視為同一層，{ 節點23, 節點24, 節點25, 節點26}視為同一層，

{節點27, 節點28, 節點29, 節點30, 節點31, 節點32, 節點33, 節點34}視為同一層，因為最後要選累積路徑值最小的節點，所以如果一開始，發現某節點的累積路徑值很大，若再累積下去，值一定會越來越大(因為距離平方恆為正值)，所以有很高的機率認為該點不可能為真實的解。因此，乾脆到此就把它排除掉，如此某節點不再當成父節點往下繼續搜尋。因此，在每一層保留 K 個具有最小累積路徑值節點，這個方法稱為K-Best演算法，因為每一層，要保留累積路徑值較小的節點，所以需要排序。

最後，節點 27 的傳送信號集合 273 表示為 $[S1, S1, S1]$ 的節點 27 的累積路徑值 270 為最小，所以三根天線傳送的值應為 $[S1, S1, S1]$ 。這樣就可以順利將傳送信號值 $x1$ 、傳送信號值 $x2$ 、及傳送信號值 $x3$ 都解出來。

然而上述方法在進行每一層搜尋時，需要先將該層的路徑值全部計算出來後，才能進行排序的動作，然而這種方式會增加運算量與消耗大量記憶體。

請參閱第四圖(a)，其為本案所提特定區域的座標轉移的示意圖。在第四圖(a)中共有64個傳送點0~63置於座標為奇數的複數平面上，且64個傳送點0~63接近複數平面的原點。對於複數平面的點進行分析觀察，發現其具有相當的規律性及對稱性。透過此特性，首先將已經過干擾消除後的任何一個點例如a點，先將之平移至x-軸的範圍 $[-1, 1]$ ，y-軸的範圍 $[-1, 1]$ 區間內的b點，即在複數平面上已知的區域，該已知區域為第四圖(a)中虛線所框之區域，且以 $[dx, dy]$ 來表示b之座標值，也就是說 dx, dy 的絕對值小於1.0。再利用四個象限仍具有對稱性，所以利用該性質，只需建立第一象限的關係，其他象限的值便可輕易獲得。透過事先離線計算，可以得知在此範圍內的任一點，與其他所有候選節點相應路徑由小到大的關係。例如根據第四圖(a)，一接收信號位於位置a點，該位置a點被複數平面的四個傳送點41、42、49、及50所圍繞，四個傳送點為與a點之間具有相對最小距離的傳送點。除此之外，根據該位置a點到傳送點的位置可找到一序列49-50-41-42。現在考慮傳送點27、28、35、及36所形成的正方形區域，將傳送點41、42、49、及50的座標分別轉換至傳送點27、28、35、及36，對應於位置a點的位置b點也就能找到。根據從位置b點到傳送

點27、28、35、及36的新序列35-36-27-28也就能找到。在第四圖(a)中可以觀察到位置a點與其鄰近傳送點的關係在經過座標轉換到位置b之後，位置b點與其鄰近的傳送點之關係仍未改變。

請參閱第四圖(b)，其為本案所提候選點序列產生的示意圖。將第四圖(b)中的平面分成複數個區域，以標號01, 02, 03, ..., 29, 30表示，在同一個區域中的可能接收點對於周圍的候選節點都具有相同的序列關係(在此，以輸出11個候選點為例)，並可找到多組序列關係，分別以群組號碼加以編號從Group ID 01到Group ID 30，在第四圖(a)中接收信號點與周圍候選點的序列關係可以歸納如表2，其中的數字代表的是複數座標上的特定點。

表2

Group ID	Candidate Sequence
01	35→27→36→28→34→26→43→19→44→20→37
02	35→27→36→28→34→26→43→19→44→20→42
03	35→27→36→28→34→26→43→19→44→42→20
04	35→36→27→28→43→34→44→26→37→19→29
05	35→27→36→28→34→43→26→44→19→37→20
06	35→27→36→28→34→26→43→44→19→20→37
07	35→27→36→34→28→26→43→19→44→42→20
08	35→36→27→28→43→44→34→26→37→29→19
09	35→36→27→28→43→44→34→37→26→29→19
10	35→27→36→28→34→26→43→44→19→20→42
11	35→27→36→28→34→26→43→44→19→42→20
12	35→27→36→28→34→43→26→44→19→37→42
13	35→27→36→28→34→43→26→44→19→42→37

14	35→27→36→34→28→26→43→44→19→42→20
15	35→27→36→34→28→26→43→44→42→19→20
16	35→36→27→28→43→34→44→26→37→19→42
17	35→27→36→28→34→43→26→44→42→19→37
18	35→27→36→34→28→43→26→44→42→19→37
19	35→36→27→28→43→34→44→26→37→42→19
20	35→27→36→34→43→28→26→44→42→19→37
21	35→36→27→28→43→34→44→26→42→37→19
22	35→36→27→28→43→44→34→26→37→29→42
23	35→36→27→28→43→44→34→26→37→42→29
24	35→36→27→28→43→44→34→37→26→29→42
25	35→36→27→43→28→34→44→26→42→37→19
26	35→36→27→43→34→28→44→26→42→37→19
27	35→36→27→28→43→44→34→37→26→42→29
28	35→36→27→43→28→44→34→26→37→42→29
29	35→36→27→43→28→44→34→26→42→37→29
30	35→36→27→43→28→44→34→37→26→42→29

也就是說總共只有30種可能，可以直接透過 $[dx, dy]$ 當成索引值，直接取出群組編號(僅需 5個位元輸出)，利用這五個位元及原先點的座標象限及平移量，很容易產生出依路徑大小排列的11個候選節點(該11個候選節點其實是從13個點中挑選出來，表中內容亦具有相當大的重覆性)，在相應的路徑大小計算方面，因為是計算相對應路徑大小，所以與其座標的絕對值無關，只與相對應的座標位置有關。一種直覺的計算方法，直接採用以下的式子來進行路徑值的計算：

$$w = [cand_x - dx]^2 + [cand_y - dy]^2, \text{ 其中 } cand_x \text{ 為候選序列點對應的 } x$$

座標， can_y 為候選序列點對應的y座標， dx 為b點的x座標， dy 為b點的y座標。

透過路徑計算模組(未顯示)，任一個經過干擾消除後的接收座標點便能快速正確地被產生出所需的候選節點及其相應精確的路徑值，且以路徑值為依據由小到大依序排列產生。候選節點的產生及排序的動作可藉由本案所提的複數平面路徑計算模組及候選點產生器(CCPG)來達成。

本案較佳實施例提出了複數平面路徑計算模組及候選點產生器(CCPG)(未顯示，詳述於第五圖(a))。

在第五圖(a)中，因為CCPG會產生一個序列(序列的順序是 $d^2(x_2, S_1)$ ， $d^2(x_2, S_2)$ 誰小就排誰在前面)以第三圖為例：

若以節點23當父節點，底下節點27，節點28視為一個群組，CCPG所產出的序列是節點27，節點28。

若以節點24當父節點，底下節點29，節點30視為一個群組，CCPG所產出的序列是節點29，節點30。

若以節點25當父節點，底下節點31，節點32視為一個群組，CCPG所產出的序列是節點32，節點31。

若以節點26當父節點，底下節點33，節點34視為一個群組，CCPG所產出的序列是節點33，節點34。這個時候，真正的路徑值為何並不知道，因為CCPG內的表沒有這個訊息。

若要挑出 {節點27, 節點28, 節點29, 節點30, 節點31, 節點32, , 節點33, 節點34} 中哪一個節點具有最小的累積路徑值，因為CCPG顯示出節點27的順序先節點28的順序，節點29的順序先於節點30的順序，節點32的順序先於節點31的順序，節點33的順序先於節點34的順序，所以只要比較 {節點27, 節點29, 節點32, 節點33} 誰具有最小的累積路徑值 (所以這

時先計算出 {節點27, 節點29, 節點32, 節點33} 的累積路徑值), 那他一定就是 {節點27, 節點28, 節點29, 節點30, 節點31, 節點32, 節點33, 節點34} 中具有最小的累積路徑值。由第三圖中可知是節點27的累積路徑值270為最小。

假如要找次小的路徑值, 其實就是要找出 {節點28, 節點29, 節點32, 節點33} 哪一個節點具有最小的累積路徑值(節點28取代節點27的位置), 這時才須計算節點28的路徑值, 然後比較其累積路徑值。

值的注意的是, 若利用具有兩輸入的比較器所構成的電路架構來做平行計算器, 例如第一個比較器的第一輸入電連接到第二個比較器的輸出, 第一個比較器的第二輸入電連接到第三個比較器的輸出, 則當在做 {節點27, 節點29, 節點32, 節點33} 比較時, 節點27與節點29的累積路徑值會分別先輸入至第二個比較器的第一輸入與第二輸入比大小, 而節點32與節點33的累積路徑值會分別先輸入第三個比較器的第一輸入與第二輸入比大小, 兩者之中最小的再比一次, 找出的就是最小的累積路徑值。這時要比三次, 才知道誰是最小的。但下一步要比 {節點28, 節點29, 節點32, 節點33}, 只須要進行對 {節點28, 節點29} 比大小, 因為 {節點32, 節點33} 的大小關係剛剛就知道了。因此每求一個次小值, 只須進行一個路徑點的計算, 若只需求 K 個, 那做 K 次, 就可以停止, 所以並不須要對所有節點都進行路徑計算後再排序, 所以運算量將會很低。若需要比的集合越大, 每次要比的個數就會少很多, 這是四個元素做比較, 每次省一個, 若是八個做比較, 一次就省4個, 這就是本案較佳實施例所提平行計算器能減少運算量的優點。

再回到第四圖(a)與(b)，若要找出與位置a點之間相對最小距離的傳送點，就必須先根據b點座標找到b點位於哪個群組。例如b點在群組23中，根據表2，對應於群組23的候選點序列的前8個元素為35、36、27、28、43、44、34、及26，其與位置b點之間相對最小距離的傳送點便可找出。既然不同區域的空間結構相類似，因此位置b點分別到傳送點35、36、27、28、43、44、34、及26的距離的順序也會與位置a分別到傳送點49、50、41、42、57、58、48、及40的距離的順序相同。

請參閱第五圖(a)，其為本案所提複數平面路徑計算模組及候選點產生器(CCPG)。該複數平面路徑計算模組及候選點產生器包含一路徑計算器 41、一群組記憶體 42、一子節點記憶體 43、一上數計數器 44、一座標轉移器 45、一上數計數器 50、一邊界檢查器 48、一象限座標還原器 49、一父節點權重器 47、一暫存器 51、一暫存器 52、及一權重調整器 46。

該複數平面路徑計算模組及候選點產生器 40 能夠依照路徑大小依序產生出相應的候選子節點，再配合大型積體電路適於平行處理的特性，提出以下的架構，來達到高速排序的目的，透過兩者特性的組合，便可達到大幅降低路徑運算量及後選節點路徑值的比較。

該群組記憶體 42 儲存該群組編號，該子節點記憶體 43 儲存序列表，該座標轉移器 45 將候選節點還原至正確的位置，該象限座標還原器 49 將候選節點還原至正確的象限，該權重調整器 46 根據通道調整其路徑值，該邊界檢查器 48 檢候選節點的座標值是否超出內定的複數平面範圍。

請參閱第五圖(b)，其為本案所提平行計算器的示意圖。該平行計算器 60 包含一暫存器 61、一暫存器 62、一暫存器

63、一暫存器 64、一減法器 65、一減法器 66、一減法器 67、一多工器 68、一多工器 69、一多工器 70。在第五圖(a)中的路徑計算器 41 可擴展成 K 個形成該平行計算器 60。在第五圖(b)中的減法器 65、減法器 66、及減法器 67 的作用是將是將兩個路徑值相減後得到一判斷值，而多工器 68、多工器 69、及多工器 70 接收該判斷值後選擇輸出最小累積路徑值。

現在考慮四個候選節點，例如 A、B、C、及 D 被置於暫存器 61~64 中以比較其累積路徑值。對該平行計算器 60 只需花 3 個比較動作就能完成比較，並且找出具有最小累積路徑值的候選節點。首先 A 與 B 的累積路徑值比較而 C 與 D 的累積路徑值比較，假如 A 與 C 的累積路徑值為較小的，則 A 與 C 的累積路徑值將分別被多工器 68 與多工器 69 所傳送。然後 A 與 C 的累積路徑值比較，即可找出 4 個候選節點中哪一個候選節點具有最小累積路徑值。要找出第二最小的累積路徑值，假設 A 已經被找出為具有最小累積路徑值的候選節點，B 的累積路徑值被多工器 68 傳送且與 C 的累積路徑值比較，C 的累積路徑值已被多工器 69 所傳送。因此對該平行計算器 60 而言，只需要多花一個步驟就能找到 4 個候選節點中哪一個候選節點具有第二最小累積路徑值。然後第三最小累積路徑值與第四最小累積路徑值也是利用同樣的方法可以找到。由此可觀察到在 N 個候選節點中找出最小累積路徑值可藉由 N-1 個平行計算的步驟來完成。

本發明在不同的應用上可得到平行計算的方法的好處。本發明提供一種方法找出具有特定數目的候選節點，該候選節點具有相對最小路徑值，例如距離複數個候選節點的累積路徑值或距離。該複數個候選節點首先被分成一數目群組，其中該數

目與所使用的平行處理器的輸入個數相同。在本例中該平行計算器 60 如同第五圖(b)所示有 4 個輸入，該複數個候選節點倍分成 4 個群組，例如 A、B、C、及 D。此例的目的在於從多個候選節點中例如 64 個候選節點，選擇特定的數目，例如 4。該 64 個候選節點被分成 4 個群組，所以每個群組 A、B、C、及 D 各包含 16 個候選節點。

以群組 A 為例，群組 A 包含候選節點 A1~A16。群組 A 可分成 4 個子群組，Aa、Ab、Ac、及 Ad，每個子群組包含 4 個候選節點，例如：Aa 包含 A1~A4、Ab 包含 A5~A8、Ac 包含 A9~A12、及 Ad 包含 A13~A16。對於每個子群組而言，可藉由平行計算器 60 而獲得漸增的累積路徑值的序列。然後每個漸增序列的累積路徑值以其漸增順序填入平行計算器 60 的每一個輸入。例如：將子群組 Aa、Ab、Ac、及 Ad 的最小累積路徑值分別輸入到暫存器 61、暫存器 62、暫存器 63、及暫存器 64。在經過平行計算器 60 的平行比較後，具有最小累積路徑值的候選節點就能被找出，例如該子群組 Aa 中的候選節點 A1。然後，將子群組 Aa 中的第二最小累積路徑值輸入該平行計算器 60 的暫存器 61。該第二最小累積路徑值與暫存器中 62 的值比較，較小的值再與多工器 69 中的值比較，然後具有第二最小累積路徑值的候選節點就能找出，且第二最小路徑值從多工器 70 被輸出。同理，第三最小累積路徑值與第四最小累積路徑值經過同樣的比較過程可找出。所以就可以獲得以最小累積路徑值、第二最小累積路徑值、第三最小累積路徑值、及第四最小累積路徑值所形成的一組序列。

同理，可由相類似的方法在群組 B、C、及 D 中找出最小累積路徑值、第二最小累積路徑值、第三最小累積路徑值、及

第四最小累積路徑值。藉由依序輸入每個序列中的最小累積路徑值、第二最小累積路徑值、第三最小累積路徑值、及第四最小累積路徑值到平行計算器 60 的暫存器 61~64，並且當最小累積路徑值被找出時，則其所屬序列的下一個最小累積路徑值取代最小累積路徑值，最後就能找出 4 個最小累積路徑值。

根據另一個較佳的例子，從兩個群組的候選節點中找出特定數目具有相對最小累基路徑值的候選節點。首先，兩個群組皆根據候選節點的累積路徑值以漸增的方式排序，例如可使用如第五圖(a)中的 CCPG 40，或是透過表 2 的查詢得到兩個序列。假設得到序列 A 與序列 B，A 包含 16 個候選節點 A1~A16，序列 B 包含 16 個候選節點 B1~B16。

平行計算器 60 通常有偶數個輸入，例如 L 個輸入。如果每一個輸入用一整數來識別，共有 $L/2$ 個具有奇數識別的輸入及 $L/2$ 個具有偶數識別的輸入。序列 A 的前 $L/2$ 個候選節點的累積路徑值被輸入具有奇數識別的輸入，序列 B 的前 $L/2$ 個候選節點的累積路徑值被輸入具有偶數識別的輸入，既然平行計算器 60 有 4 個暫存器 61~64，暫存器 61 與暫存器 63 則代表奇數識別的輸入，暫存器 62 與暫存器 64 則代表偶數識別的輸入。在序列 A 中的前兩個候選節點 A1 與 A2 分別被置於暫存器 61 與暫存器 63 的輸入，在序列 B 中的前兩個候選節點 B1 與 B2 分別被置於暫存器 62 與暫存器 64 的輸入。在經過平行計算器 60 的第一次平行計算後，群組 A 與群組 B 中具有最小累積路徑值的候選節點就能被找出來。例如 A1 的累積路徑值小於 B1 的累積路徑值且 B2 的累積路徑值小於 A2 的累積路徑值，所以 A1 的累積路徑值被多工器 68 輸出且 B2 的累積路徑

值被多工器 69 輸出。假設 A1 的累積路徑值小於 B2 的累積路徑值，則 A1 的累積路徑值被多工器 70 輸出，因此在兩群組中的最小累積路徑值被找出。然後 A3 的累積路徑值從 A1 的同一個序列中被輸入到暫存器 61 的輸入。第二次的計算只需兩個步驟，因為 A2 的累積路徑值與 B2 的累積路徑值之前已經比較過，所以只需比較 A3 的累積路徑值與 B1 的累積路徑值，在將其兩者中最小的累積路徑值與 B2 的累積路徑值比較，則第二最小累積路徑值就能被找出來。然後，下一個最小累積路徑值可依照相同的方式從兩群組中被找出，直到特定數目的累積路徑值都被找出。

當採取搜尋樹尋找具有最小累積路徑值的候選節點時，每個父節點連接到大量的子節點，為了節省計算資源，其中之一的方法是在搜尋樹中的每層選擇一特定數目具有最小累積路徑值的子節點，此方法稱為 K-Best 法。例如在 MIMO 的通訊系統中共有 15 個傳送天線與 20 個接收天線。16-QAM 信號在每一個傳送天線被傳送，假設 K 值等於 8，作為特定的數目。搜尋樹可模擬上述 MIMO 通訊系統具有 15 層，其中每個父節點連接到 16 個子節點。在第一層的計算過程中，8 個子節點及其相對最小累積路徑值的節點被找出。其他的子節點則被捨棄。然後在第二層的計算過程中，128 個子節點只有 8 個子節點被保留，其餘則被丟棄。K-Best 法在解碼的過程中節省了大量的計算。

在第三圖中，該搜尋樹若對第 i 層的搜尋而言，每個存活的候選節點(父節點)集合的候選節點都已經是依相應的路徑大小由小到大排列。在第五圖(a)及第五圖(b)中，依序將每一個存活的候選節點(父節點)所對應的最小路徑節點(子節點)

取出並計算其相應的累積路徑值，然後再送入相應平行計算器 60 的暫存器 61~64 內來進行平行比較，如此便可順利地選出具有最小累積路徑值的節點，。為方便說明，在此假設該候選節點來自第 j 個存活的候選節點(父節點)。該選取候選節點即為整個搜尋範圍中所有可能候選節點中具有最小路徑值的節點。配合所提的平行計算器 60，則所須要的運算量為 K 個路徑值計算以及 $(K-1)$ 個比較選取動作， K 個路徑值計算。

接下來，僅需將前一時脈週期所選出的候選節點所相應的第 j 個父節點它對應的下一個具有最小路徑值的子節點(也就是說不含先前選取的候選節點)，透過 CCPG 40 便能很容易產生出來並計算其相應的累積路徑值，用來取代原先選取的子節點的累積路徑值以進行平行比較，而其餘的輸入暫存器的值都維持現狀不改變。此時平行比較器 60 會選出一個具有最小累積路徑值的子節點，該子節點即為搜尋剔除已被選取的最小累積路徑節點外所有可能候選節點中具有最小累積路徑值的候選節點。此時每挑出一個候選節點所須要的運算量為 $\log_2 K$ 個比較選取的運算量，以及單一個路徑值的計算。

當經過 K 個完整比較時脈週期，便可完成在眾多候選節點中正確無誤地搜尋挑選出 K 個具有最小路徑的候選節點。不論合法候選節點的個數有多少。只需針對 $(2K-1)$ 個真正參與最小路徑競爭的節點進行路徑計算。

K -Best 法所關心的在於：是否在前幾層計算的過程中丟棄許多節點是適當的，所以若在前幾層就丟棄節點，真正具有最小累積路徑值的候選節點就不可能被找到。所以有必要將前幾層的所有子節點都保留，當然當選取的 K 越大，整體的運算量也會大幅提升，功率消耗也會明顯加大。

有一些方法被提出來以解決上述問題如下：

方法一：對通訊系統中的通道矩陣進行一奇異值分解，然後根據藉由該奇異值分解過程被計算的條件值以決定 K 值。此目的是當通道矩陣的條件值很大時，則選擇一個較大的 K 值，使得最大相似解(maximum-likelihood solutions)可以被保留。但是此方法在做奇異值分解時會消耗大量的計算，結果會消耗大量功率且造成低資料量的傳輸。

方法二：直接選擇一個大的 K 值，以確保適當的節點被保留，此方法的代價是消耗大量的計算及高功率的消耗。

方法三：在搜尋樹的前幾層無例外地進行最大相似計算，然後再採用 K-Best 法。然而，並沒有任何理論上的分析以決定多少層需要最大相似計算，根據文獻的記載，不論是藉由複雜的分析或是電腦模擬，只有不同層數的最大相似計算的相對效能被比較。從硬體應用的觀點而言，越多層的最大相似搜尋將不止引起高功率的消耗，而且也需要大量的記憶體空間以儲存可能的候選節點之資訊。

方法四：在搜尋樹的前幾層無例外地進行最大相似計算，然後再採用 K 值較小的 K-Best 法，以降低計算量。方法三與方法四在理論上是相同的。

根據本發明的實施例，對所提到的缺點的最佳解法為：基於對通道矩陣的分析在搜尋樹的前幾層有條件地維持最大相似計算，然後再採取 K-Best 法。此外，此實施例建議當需要最大相似搜尋時，則利用平行計算處理以降低計算量。

請參閱第六圖(a)，其為傳送接收天線皆為四之通道矩陣之行向量經過以行模大小由小到大後排序並經過 QR 分解後，R 矩陣對角線元素平方值機率分佈圖。請參閱第六圖(b)，

其為傳送接收天線皆為八的之通道矩陣之行向量經過以行模大小排序並經過 QR 分解後，R 矩陣對角線元素平方值機率分佈圖。從該第六圖(a)及(b)可以看出，最後一層對應的對角線元素，有相當大的機率其數值會很小。當值很小的情況下，代回原來的方程式會使得整個複數平面上的有效訊號點個數變的很大，甚至於會將所有的點都包含進來。此時我們若仍維持先前的描述。在每層搜尋時，只保留 K 個具有最小路徑的存活節點，便有很大的機會，會將真實的解所對應的節點在此時就被丟棄掉，造成性能的下降。而進入到第三層搜尋時，該影響明顯會變小。因此，我們利用此特性，提出一個是否啟動最大可能搜尋的決策器。利用此決策器，提供一個判斷機制，避免一開始就把真實的解丟棄掉。值得注意的是：我們並不是對所有的通道條件下都進行最大可能搜尋。顯然地，可大幅降低不必要的動作，達到低功率消耗的目的。

由於要進行最大可能搜尋，基於先前所提出的 CCPG 40 及平行計算器 60 的架構下，只需針對控制程序稍做修改，使得在進行聯合第一二層最大相似搜尋時，仍以極高效率的方式進行，且可大幅減少硬體成本及功率消耗。

綜合以上的這些觀點，針對這個特性，提出一個最大可能搜尋決策方法，也就是說當滿足下列方程式時，最大可能搜尋才會被啟動：

$$\frac{T_r^2 \times D_{K, \min}^2}{E \{(d_c)^2\}} = 1 \quad (d_c)^2 = \|\mathbf{y} - \mathbf{H}\mathbf{x}_{ZF}\|^2$$

T_r 為欲求的臨界值

$D_{K, \min}^2$ 為取 K 個點對應的等效半徑的平方

$E\{(d_c)^2\}$ 為接收點與 zero-forcing 解點的距離的平方

之期望值，其中的 zero-forcing 解點為幾個較佳的例子之一。

也就是說： $r_{N,N}, r_{N-1,N-1}$ 作用於以先前預定 K 值所對應的等效搜尋半徑小於 zero-forcing 解點與接收點的距離。在此情況下，便可能將真實的解在進行 K 點保留時，被丟棄掉。因此，我們建議此時應啟動聯合第一二層的最大可能搜尋的機制來獲得較佳的性能。該聯合第一二層的最大相似搜尋演算法如下：

● 首先第一層會有 MD 個 (MD 為所有星座座標點有效點的個數) 有效點，若每個有效點視為一個父節點，則每個父節點會有 MD 個子節點。利用先前的電路 (CCPG 40 及平行計算器 60)，將 MD 個父節點拆成 m 個群組 (若 MD 能被 K 整除，則 $m = MD/K$ ，否則 $m = [MD/K] + 1$ ， $[\]$ 取整數之意)，將每個群組送入先前的電路 (CCPG 40 及平行計算器 60)，可以很有效率地找出屬於該群組內具有 K 個最小累積路徑值的候選節點。只要執行 m 次，便可找出 $m \times k$ 個內具有最小累積路徑值的候選節點。重覆使用該平行計算器 60 進行群組合併，便可達成聯合一、二層最大可能搜尋目的，找出具有 K 個最小累積路徑值的候選節點。

其中對於所選定的第 i -th、 j -th 個群組 (每個群組內含 K 個候選節點) 進行合併，其動作說明如下：

對第 i -th 個群組的前 $K/2$ 個元素置於該平行計算器 60 的偶數的位置。

對第 j -th 個群組的前 $K/2$ 個元素置於該平行計算器 60 的奇數的位置。

此時共須 K 次記憶體讀取的動作，且同時可以找出第 i -th、 j -th 個群組內的最小值。接下來，將最小值所屬的群組內所剩餘元素中具有最小值的元素補上，進行下一次的比較，便可順利找出第 i -th、 j -th 個群組內的次小值。總共經過 K 次比較便可完成兩個群組合併的動作。

透過改良式的平行比較方法，結合最大相似性搜尋決策器，透過嚴謹的實驗證明，在相同的環境下可得到較一般傳統最佳 K -Best 球形解碼器為佳的性能，且具有相當低的運算複雜度及記憶體存取次數，且達到高效率且低功率消耗的目的。

另一個好處是可以利用原先的 CGPG 40 及平行計算器 60 而不需另外建構一組新的電路。基於本案所提出的架構及演算法，確實可提供一個高效能且低功率消耗的解碼器，尤其特別適用於個人行動通訊裝置。

本發明的說明與實施例已揭露於上，然其非用來限制本發明，凡習知此技藝者，在不脫離本發明的精神與範圍之下，當可做各種更動與修飾，其仍應屬在本發明專利的涵蓋範圍之內。

【圖式簡單說明】

第一圖：習知球形解碼演算法的示意圖；

第二圖：其為習知尋找候選點的示意圖；

第三圖：其為本案所提具有 P 個狀態的搜尋樹的示意圖；

第四圖(a)：本案所提特定區域的座標轉移的示意圖；

第四圖(b)：本案所提候選點序列產生的示意圖；

第五圖(a)：本案所提複數平面路徑計算模組及候選點產生器 (CCPG)；

第五圖(b)：本案所提平行計算器的示意圖；

第六圖(a)：傳送接收天線皆為四的 R 矩陣對角線元素平方值機率分佈圖；

第六圖(b)：傳送接收天線皆為八的 R 矩陣對角線元素平方值機率分佈圖；

【主要元件符號說明】

20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
34: 節點

200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300,
310, 320, 330, 340: 累積路徑值

201, 202, 211, 212, 221, 222, 231, 232, 241, 242, 251,
252, 261, 262: 父節點到子節點的距離

213, 223, 233, 243, 253, 263, 273, 283, 293, 303, 313,
323, 333, 343: 傳送信號集合

40: 複數平面路徑計算模組及候選點產生器 (CCPG)

41: 路徑計算器

42: 群組記憶體

43: 子節點記憶體

44, 50: 上數計數器

45: 座標轉移器

46: 權重調整器

47: 父節點權重器

48: 邊界檢查器

49: 象限座標還原器

60: 平行計算器

61, 62, 63, 64: 暫存器

65, 66, 67: 減法器

68, 69, 70: 多工器

七、申請專利範圍：

1. 一種針對在一通訊系統中的一接收點從複數個傳送點之中找出一特定數目具有最小累積路徑值的通訊點的方法，該方法包含下列步驟：

(a) 在一複數平面上定義各該複數個傳送點與該接收點的第一座標；

(b) 將該接收點相應的該第一座標轉換到相應的一第二座標，其中該接收點位於該複數平面的一已知區域；及

(c) 依據該接收點的該第二座標找出該特定數目具有最小累積路徑值的傳送點，其中該步驟(c)更包含下列子步驟：

(c1) 提供複數個序列，其中各該序列包含各自具有一累積路徑值的複數個候選節點，且各該序列根據該累積路徑值以一由小到大的順序來排列；

(c2) 依據該接收點的該第二座標，從該複數個序列中選擇一序列；

(c3) 從該序列中挑選該特定數目具有最小累積路徑值的該等候選節點；

(c4) 分別找出該特定數目在該複數平面上具有最小累積路徑值的候選節點的一第二座標；及

(c5) 分別將該特定數目具有最小累積路徑值的該等候選節點的該第二座標還原為該第一座標。

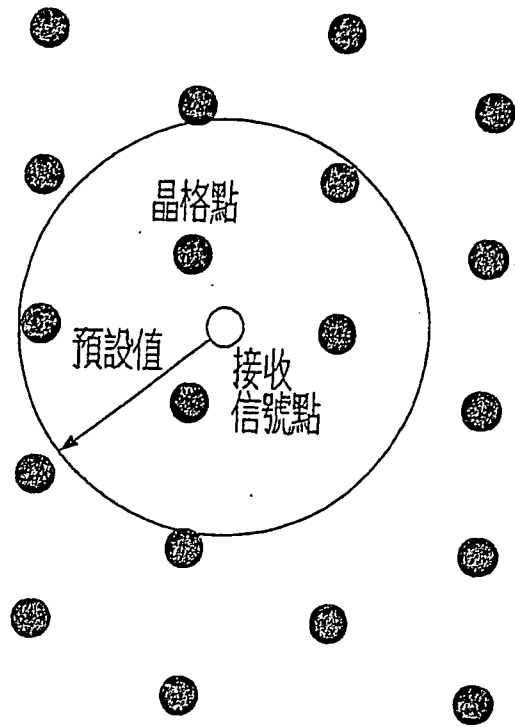
2. 如申請範圍第1項所述的方法其中在該子步驟(c1)中，該複數個序列是藉由計算該候選節點的該累積路徑值來獲得。

3. 如申請範圍第1項所述的方法，其中在該子步驟(c1)中，該複數個序列是藉由查詢一表格的來獲得。

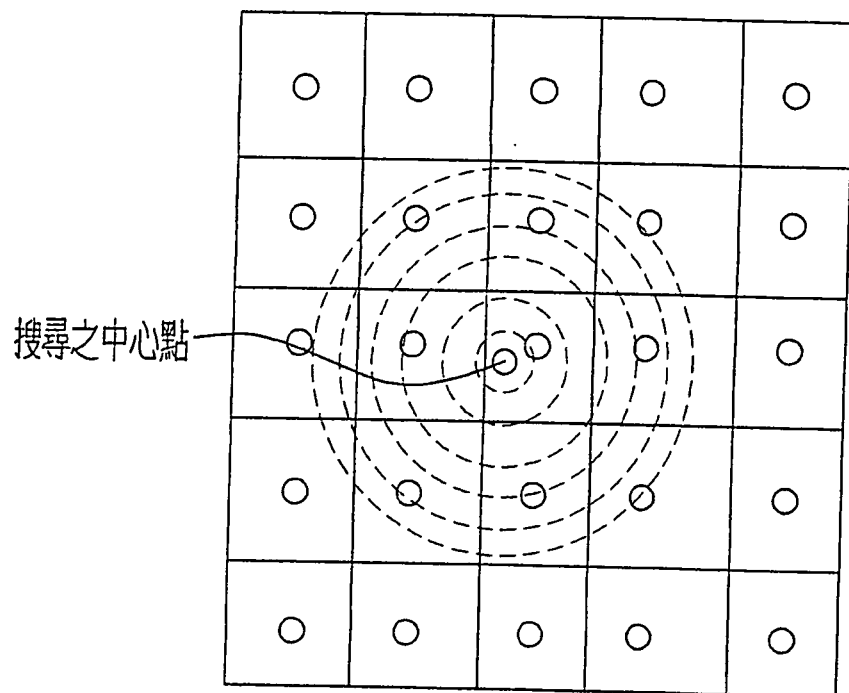
4. 如申請範圍第3項所述的方法，其中該表格係利用該第二座標平面上的一幾何關係，而將該等候選節點根據該累積路徑值的該由小到大的順序，而分別形成該等序列。

5. 如申請範圍第4項所述的方法，其中該表格包含該幾何關係中所有可能的該累積路徑值由小到大的排列順序。

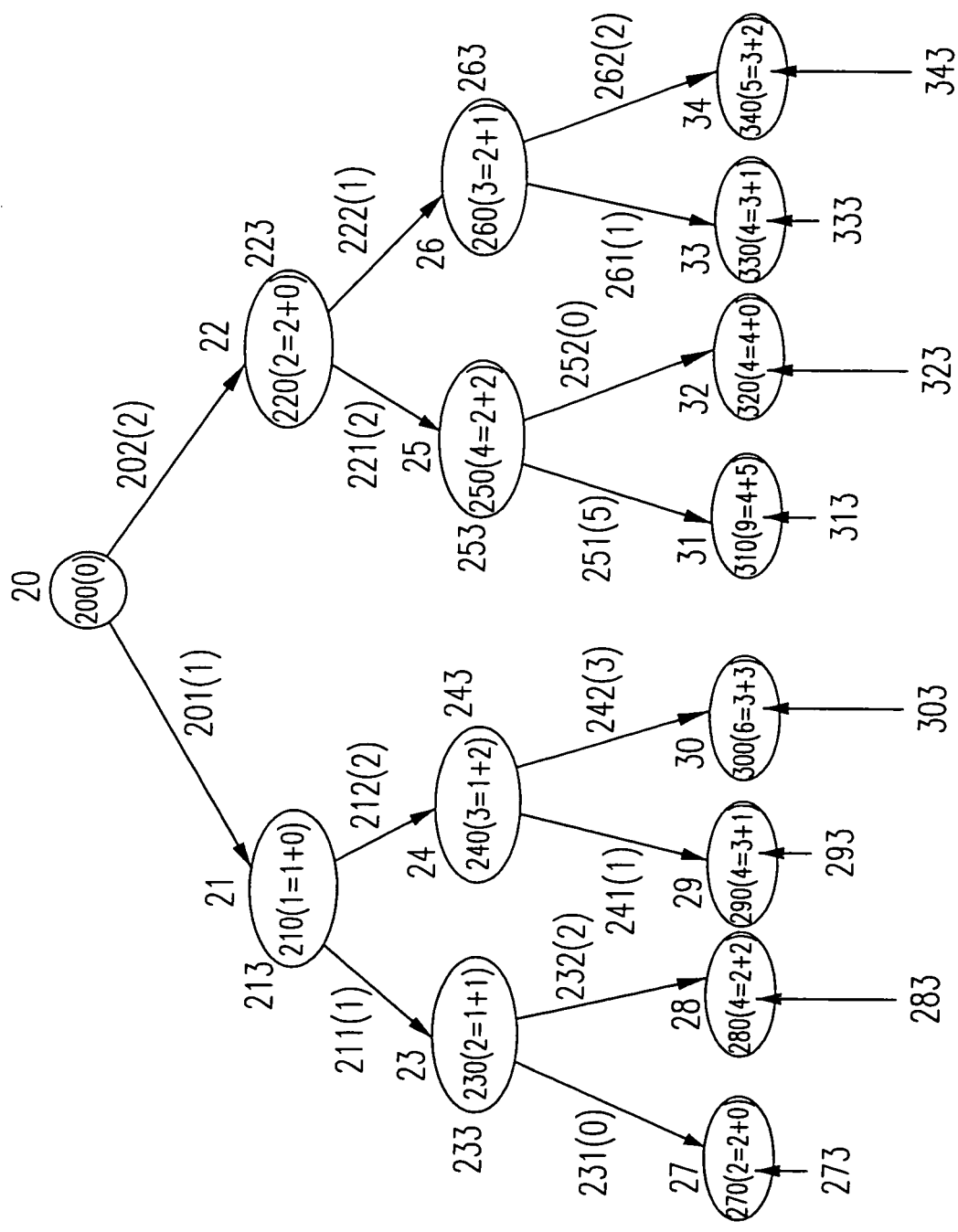
99年10月14日修正本



第一圖

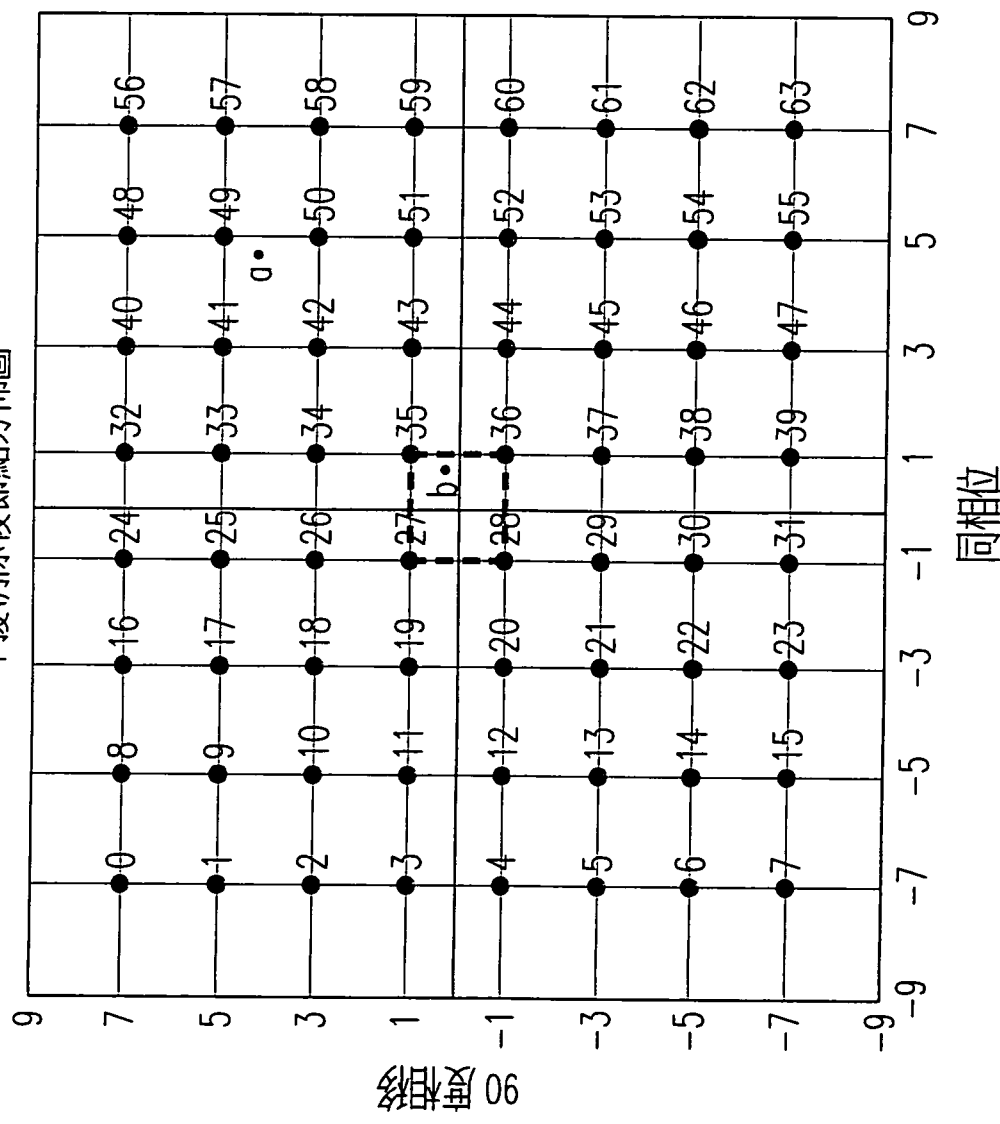


第二圖

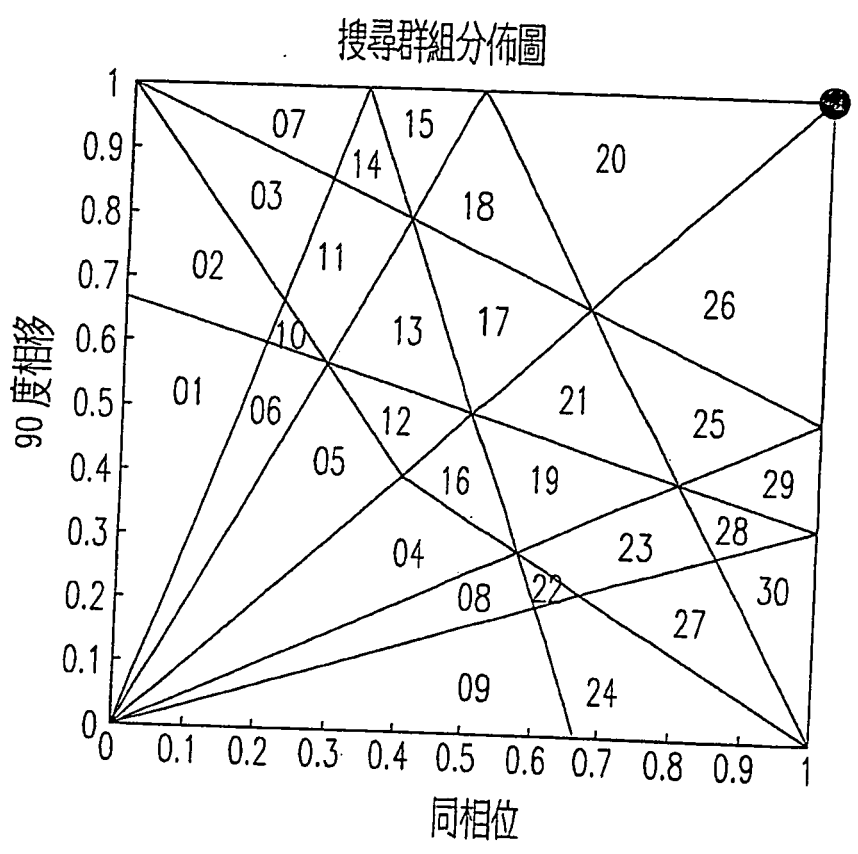


第三圖

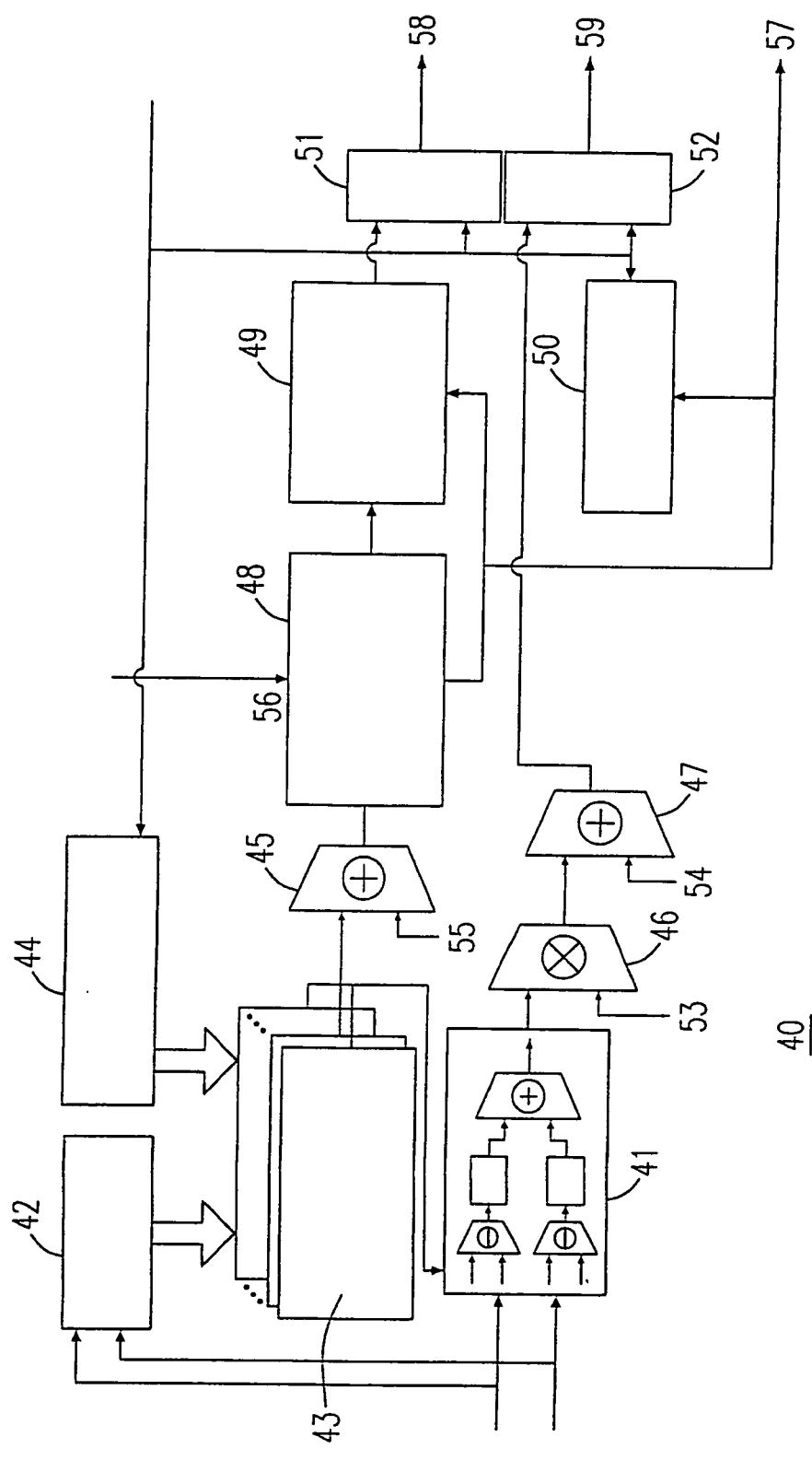
干擾消除後節點分佈圖



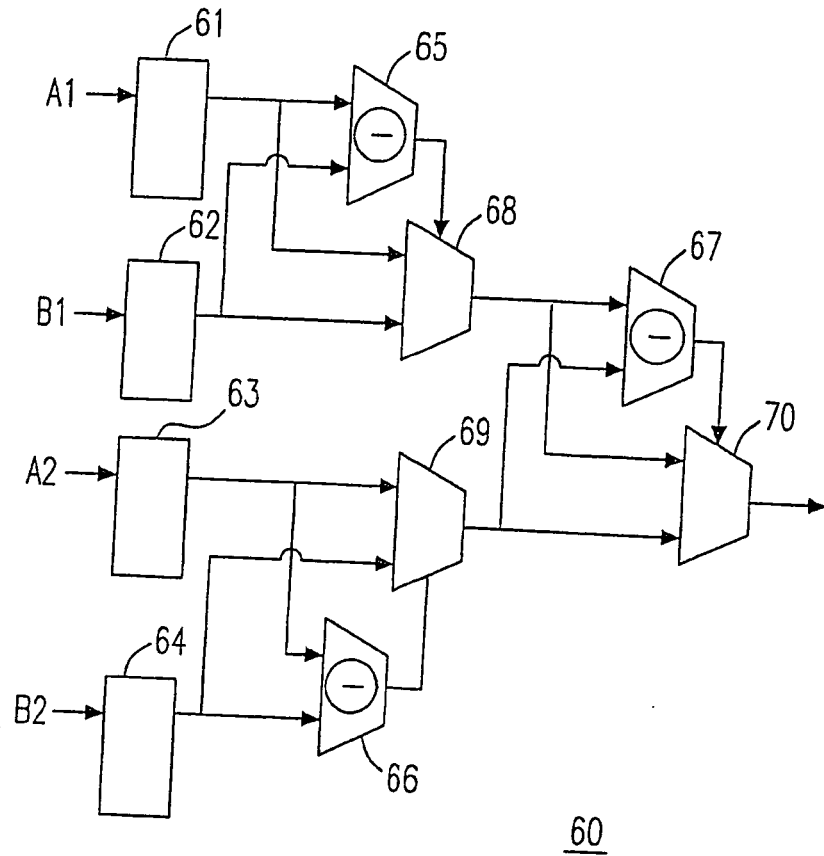
第四圖(a)



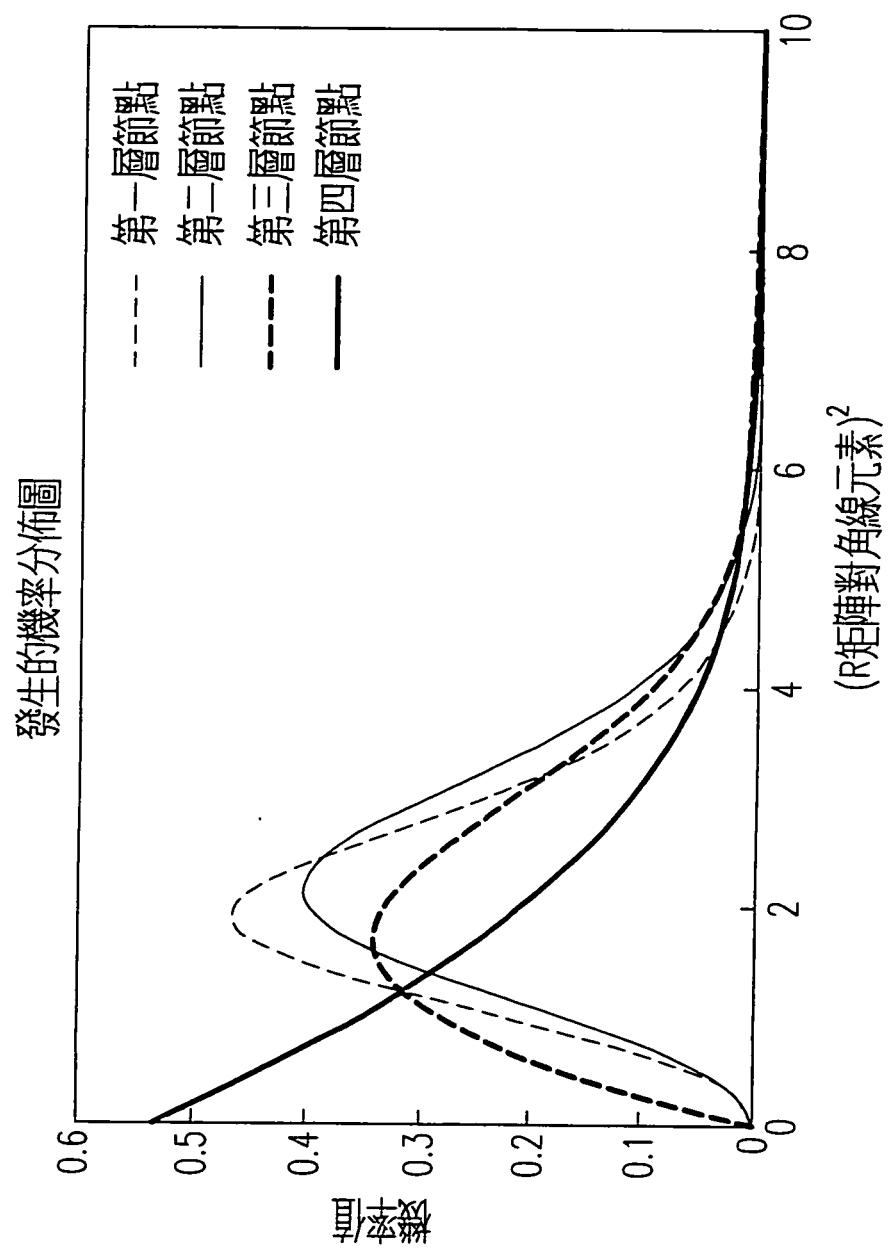
第四圖(b)



第五圖(a)

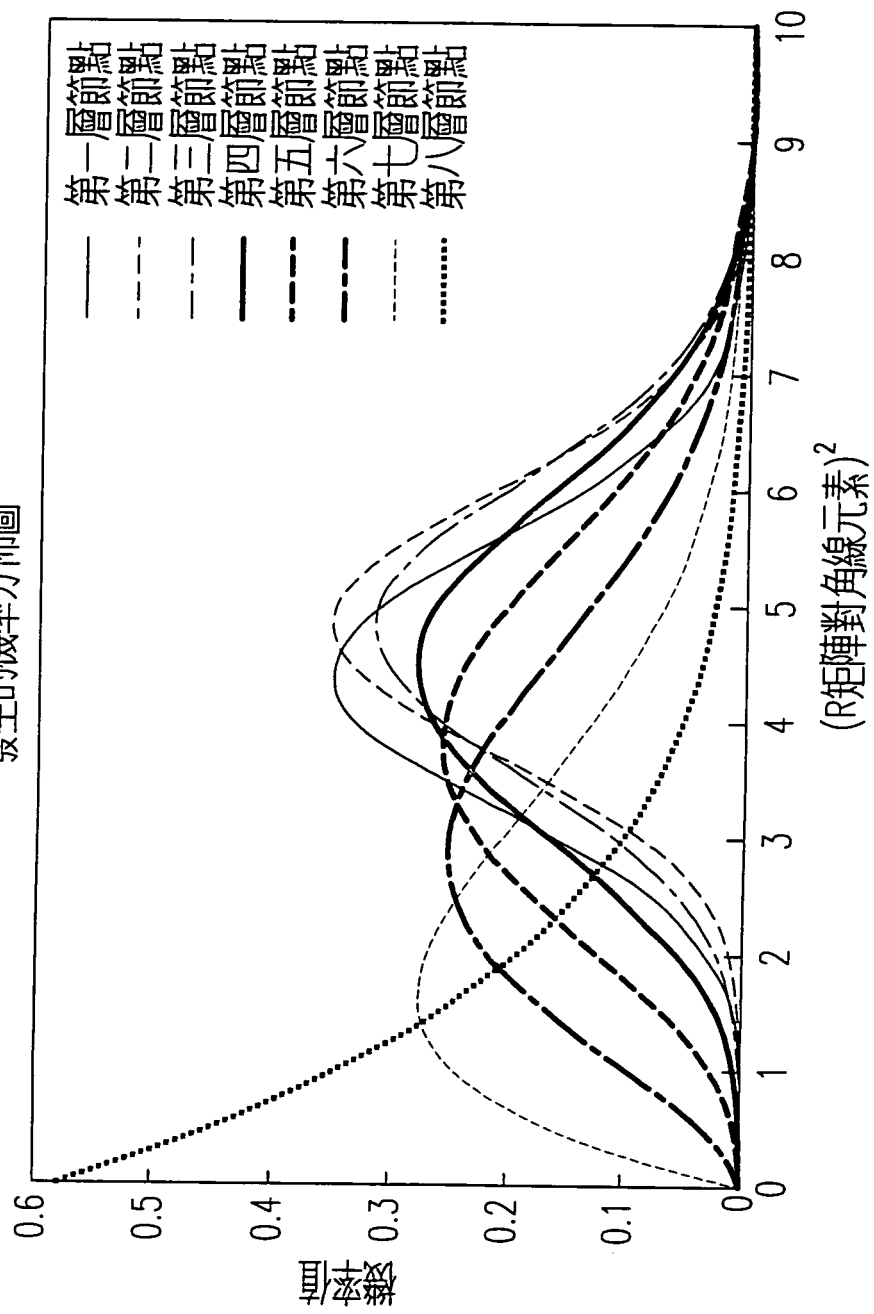


第五圖(b)



第六圖(a)

發生的機率分佈圖



第六圖(b)