

(19)日本国特許庁(JP)

(12)特許公報 ( B 2 )

(11)特許番号

特許第 3 2 3 5 7 1 3 号

( P 3 2 3 5 7 1 3 )

(45)発行日 平成13年12月4日(2001.12.4)

(24)登録日 平成13年9月28日(2001.9.28)

(51)Int. Cl.<sup>7</sup>

識別記号

F I

H 0 3 M 7/30

H 0 3 M 7/30

B

請求項の数 6

(全 7 頁)

(21)出願番号 特願平8-121690

(22)出願日 平成8年5月16日(1996.5.16)

(65)公開番号 特開平9-307450

(43)公開日 平成9年11月28日(1997.11.28)

審査請求日 平成10年11月17日(1998.11.17)

(73)特許権者 000004226

日本電信電話株式会社

東京都千代田区大手町二丁目3番1号

(72)発明者 守谷 健弘

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(72)発明者 岩上 直樹

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(74)代理人 100066153

弁理士 草野 卓

審査官 石井 研一

最終頁に続く

(54)【発明の名称】ベクトル量子化法及びその符号帳

1

(57)【特許請求の範囲】

【請求項 1】 複数サンプルよりなる入力ベクトルに対し、符号帳中の各候補ベクトルから歪の最も小さいものを選択して、上記入力ベクトルを量子化するベクトル量子化法において、

上記符号帳としてその候補ベクトルをその要素配列と直角方向に配列したとき、同一番号の要素はベクトル配列方向において0の要素または0でない要素が等間隔になるものを用い、

候補ベクトルの数だけ部分歪を記憶する歪記憶部を用意し、

上記入力ベクトルの指定された要素番号の要素と、上記各候補ベクトルの上記指定された要素番号で0でない要素との部分歪をそれぞれ計算し、その各計算された部分歪を対応する上記歪記憶部の部分歪に加算し、

2

上記要素番号の指定を変えて上記部分歪の計算を行って入力ベクトルと、各候補ベクトルとの歪を求めることを特徴とするベクトル量子化法。

【請求項 2】 複数サンプルよりなる入力ベクトルに対し、複数の符号帳中の候補ベクトルの和の組み合わせの中から、歪の最も小さい組み合わせを選択して、上記入力ベクトルを量子化するベクトル量子化法において、上記符号帳の少なくとも1つはその候補ベクトルを、その要素配列と直角方向に配列したとき、同一番号の要素は、ベクトル配列方向において、0の要素または0でない要素が等間隔になるものを用い、

候補ベクトルの和の各組み合わせと対応した部分歪を記憶する歪記憶部を用意し、

上記入力ベクトルの指定された要素番号の要素と、上記候補ベクトルの和の組み合わせの上記指定された要素番

10

号で 0 でない要素との部分歪をそれぞれ計算し、その各計算された部分歪を対応する歪記憶部の部分歪に加算し、

上記要素番号の指定を変えて上記部分歪の計算を行って、上記入力ベクトルと上記候補ベクトルの和の組み合わせとの歪を求めることを特徴とするベクトル量子化法。

【請求項 3】 入力ベクトルとの量子化歪の計算において、負の項となる部分について、まず計算し、その計算結果の大きいものと対応する複数の候補ベクトルを選出し、これら選出した複数の候補ベクトルについて、上記歪を求める計算を行うことを特徴とする請求項 1 または 2 記載のベクトル量子化法。

【請求項 4】 入力ベクトルの指定された要素番号の要素と、符号帳の各候補ベクトルの指定された要素番号で 0 でない要素との部分歪をそれぞれ計算し、その各計算された部分歪を各候補ベクトルごとに加算して、入力ベクトルと候補ベクトルの歪を求めるベクトル量子化法において、

上記符号帳は同一番号の要素はベクトル配列方向において、0 の要素または 0 でない要素が等間隔になっていることを特徴とするベクトル量子化法。

【請求項 5】 ベクトル量子化における歪計算における入力ベクトルの要素を含まない各計算項の各計算結果が、上記 0 でない要素と共に記憶されていることを特徴とする請求項 4 記載のベクトル量子化用符号帳。

【請求項 6】 周波数領域に変換された係数の量子化に用いられ、上記 0 でない要素は低次の上記要素側は密で、高次の上記要素側は疎となっていることを特徴とする請求項 4 又は 5 記載の符号帳。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は複数サンプルよりなる入力ベクトルと最も近い候補ベクトルを符号帳から選択して、量子化する方法及びこれに用いる符号帳に関する。

【0002】

【従来の技術】ベクトル量子化は図 5 A に示すように、入力端子 1 1 からの複数のサンプルを 1 つの入力ベクトルとしたものと、符号帳 1 2 から選択した候補ベクトルとの距離（歪）を距離計算部 1 3 で計算し、この距離（歪）が最も小さいものを符号帳 1 2 から選び出すように制御部 1 4 で制御し、その最小歪の候補ベクトルの番号を量子化結果として出力端子 1 5 へ出力する。

【0003】このベクトル量子化では符号帳中の候補ベクトルと入力ベクトルの距離計算（歪計算）が処理の大部分を占める。歪を小さくすること、距離計算を少なくすることを両立させることは難しく、ベクトル量子化を実用化する上での障害になっている。用途によっては符号帳中の候補ベクトルの要素の半数以上を 0 としても

歪が殆ど増加しない場合がある。この例を図 6 に示す。図 6 は ( P - 1 ) 次元の N 個の候補ベクトルを横方向を要素配列とし、縦方向をベクトル配列方向として並べた場合で、各要素中の 0 でないものを X で示し、0 の要素が可なり含まれている。距離計算は入力ベクトルの各要素と候補ベクトルの対応する要素との積で計算できるため、符号帳の候補ベクトルの要素が 0 の場合、その距離計算をする必要がなくなり、演算量を削減できる可能性がある。

【0004】

【発明が解決しようとする課題】しかし、信号処理プロセッサ ( DSP ) など前記距離計算を高速に処理させるためには、各候補ベクトルは 0 となる要素が等間隔または 0 とならない要素が等間隔であり、かつその 0 となる要素位置が予め知られ、0 となる要素に対する入力ベクトルの対応要素との乗算は行わないようにプログラムされる。

【0005】このように 1 つの候補ベクトルはその 0 の要素または 0 でない要素が等間隔でないと、入力ベクトルとの距離計算を高速に行うことができない。一方 0 の要素または 0 でない要素を等間隔にすると、入力ベクトルとの距離計算を高速に行うことができるが、そのような候補ベクトルは、ベクトルの自由度が低下して、量子化歪が大きいものとなる問題があった。この発明の目的は符号帳の中のベクトルの数多くの要素を規則的に 0 として、歪の増加を抑えつつ処理量を削減する方法及び符号帳を提供することにある。

【0006】

【課題を解決するための手段】この発明ではベクトル量子化の距離計算を候補ベクトルごとに行うのではなく、要素番号を指定して要素ごとにすべての候補ベクトルの距離を計算し、そのとき符号帳中のベクトルを 0 の要素または 0 でない要素をベクトル配列方向に等間隔に配置する。つまり、1 つの候補ベクトルに注目すると 0 または 0 でない要素が等間隔にならないことが従来方法と異なる。

【0007】

【発明の実施の形態】この発明で用いる符号帳の例を図 1 に示す。図と同じように、1 つの候補ベクトルの要素の配列を横方向とし、候補ベクトルの配列方向を縦方向として示してある。つまり、横方向の要素数 p - 1 は候補ベクトルの次元数であり、縦方向の候補ベクトル数 N - 1 は 2 のビット数乗個ある。図中の 0 はベクトルの要素が 0 であり、X は 0 でないことを示す。

【0008】一見不規則であるが、1 つの列を縦方向に見ると、つまり各候補ベクトルの i 番目の要素をベクトル配列方向に見ると、いずれも 0 でない要素が等間隔になっている。ただし、間隔は列によって異なっている。この例では 0 でない要素の間隔は i = 0 では 1 個間隔、i = 1, 2 ではそれぞれ 2 個間隔、i = 3, 4, 5, 6

ではそれぞれ 3 個間隔となっている。この結果、1 つの候補ベクトルについて見ると、すなわち、横方向に要素を見ると 0 でない要素は不規則で、候補ベクトルは数多くの種類が存在することがわかる。またこの例では  $i$  が小さいとき、すなわち低周波数に対応する要素の領域では、0 でない値が密で、 $i$  が大きいときには、0 でない要素が疎となるように配置してある。例えば音声信号のような音響信号を周波数領域の変換係数として量子化す\*

$$D_n = w(i) |x(i) - c(n, i)|^2$$

$$= w(i)x(i)^2$$

$$- 2 w(i)x(i)c(n, i)$$

$$+ w(i) c(n, i)^2$$

は  $i = 0$  から  $p - 1$  まで

右辺第 1 項は候補ベクトルと無関係になるため、この発明では第 2 項と第 3 項の和を  $D_n$  とし、 $n$  ごとに部分結果  $d_{ni}$  を記憶する。そして下記のように 1 つの  $i$  に対して  $n$  を増加させながら値を更新する。

$$d_{ni} = - 2 w(i)x(i)c(n, i) + w(i)c(n, i)^2 \quad \dots\dots (2)$$

その計算結果を対応する歪記憶部 2 1<sub>n</sub> 中の記憶歪  $d_n$  を次式で更新する。

$$d_n = d_n + d_{ni} \quad \dots\dots (3)$$

このようにして、要素番号  $i = 0$  から  $i = p - 1$  までについて順次、各  $N$  個の候補ベクトルの各要素  $c(n, i)$  と入力ベクトルの対応要素  $x(i)$  との部分歪を計算し、かつ、その計算結果をそれまでの対応候補ベクトルの部分歪に加算することが行われ、その際に、1 つの要素番号  $i$  について、候補ベクトルの要素  $(n, i)$  が 0 のものについての計算は省略して計算量を減少する。このようにして全てのベクトル要素について計算した後、歪記憶部 2 1<sub>0</sub> ~ 2 1<sub>N-1</sub> の各記憶部分歪  $d_n$  は、入力ベクトルと各候補ベクトルとの距離に対応したものとなり、これらの最小のものの候補ベクトル番号  $n$  を選出してベクトル量子化結果とする。

【0 0 1 2】例えば図 2 に示すように処理する。まず、歪記憶部 2 1<sub>1</sub> ~ 2 1<sub>N-1</sub> の各記憶内容  $d_n$  を 0 とし、要素番号  $i$  と候補ベクトル番号  $n$  とを 0 とし ( $S_1$ )、 $N$  個の候補ベクトルの  $i$  番目の要素の配列  $c(0, i), c(1, i), c(2, i) \dots, c(N - 1, i)$  において、最初に 0 でない候補ベクトルの番号  $n_0$  と、0 でない要素と次に 0 でない要素との間の 0 の数  $n$  を取得する ( $S_2$ )、これら  $n_0, n$  は予め既知であり、例えば図 3 A に示すように、各要素番号  $i = 0, 1, 2, \dots, p - 1$  についての  $N$  個の候補ベクトルの対応要素配列における最初の 0 でない候補ベクトル番号  $n_0$  と、0 でない要素内の候補ベクトル数  $n$  とがテーブルとして作られ、ステップ  $S_2$  においてその  $i$  の値に対する  $n_0, n$  を 2 つのテーブルから読み取る。なお、図 3 A のテーブルは図 1 の候補ベクトルの例について示しており、 $i = 0$  で  $n_0 = 0, n = 1, i = 1$  で  $n_0 = 1, n = 2, \dots$  となる。

\* する場合、このような配置のほうが都合がよいことがある。その周波数領域の変換係数としては、いわゆる平坦化された残差変換係数でもよい。

【0 0 0 9】ところで  $p - 1$  次元の入力ベクトルの  $i$  番目の要素を  $x(i)$ 、 $n$  番目の候補ベクトルの  $i$  番目の要素を  $c(n, i)$ 、 $i$  番目の要素に対する距離の重みを  $w(i)$  とすると、入力ベクトルと  $n$  番目の候補ベクトルとの距離  $D_n$  は次のようになる。

$$\dots\dots (1)$$

【0 0 1 0】つまり、図 1 に示すように各候補ベクトル  $c_n$  ごとに歪記憶部 2 1<sub>n</sub> ( $n = 0, 1, 2, \dots, N - 1$ ) を設け、例えば各ベクトルの要素番号  $i$  を順次指定し、その  $i$  ごとに次式を計算し、

【0 0 1 1】

20

$$\dots\dots (3)$$

【0 0 1 3】ステップ  $S_2$  で取得した  $n_0$  を  $n$  とし ( $S_3$ )、そのときの  $n, i$  を満たす、(2) 式 ( $x(i)$  と  $c(n, i)$  との距離に対応した値) を計算し ( $S_4$ )、その計算結果  $d_{ni}$  を対応部分歪記憶部 2 1<sub>n</sub> 内の記憶歪  $d_n$  に加算して、その値を  $d_n$  とし、つまり  $d_n$  の更新を行う ( $S_5$ )。次に  $n$  に  $n$  を加え ( $S_6$ )、その結果の  $n$  が  $N - 1$  より大かを調べ ( $S_7$ )、大でなければステップ  $S_4$  に戻って、次に 0 でない要素について部分歪を計算し、大であればその  $i$  番目の要素配列における候補ベクトル中の 0 でない各要素と入力ベクトルの対応要素との部分歪が全て計算されたことになり、 $i$  を  $+ 1$  し ( $S_8$ )、その更新された  $i$  が  $p + 1$  より大かを調べ ( $S_9$ )、大でなければステップ  $S_2$  に戻って、次の要素列中の 0 でない要素との部分歪の計算を行い、 $i$  が  $p + 1$  より大であれば、全ての候補ベクトルと入力ベクトルとの距離計算が終わったことになり、部分歪記憶部 2 1<sub>1</sub> ~ 2 1<sub>N-1</sub> に記憶されている歪  $d_0 \sim d_{N-1}$  中の最小のものを選択し ( $S_{10}$ )、その選択した最小距離が得られた候補ベクトルのベクトル番号  $n$  を量子化結果  $n_r$  として出力する。

30

40

50

【0 0 1 4】入力ベクトルと各候補ベクトルとの各距離は (1) 式によるから、図 2 を参照した量子化においても、実際の距離を求めたい場合は、(1) 式の右辺第 1 項  $\sum_{i=0}^{p-1} w(i)x(i)^2$  を 1 回計算して、その結果を記憶部 2 1<sub>0</sub> ~ 2 1<sub>N-1</sub> に記憶されている最終累積加算結果にそれぞれ加算すればよい。また (2) 式の第 2 項は候補ベクトルのみで決まるものであるから、各  $c(n, i)^2$  または各  $w(i)c(n, i)^2$  を予め計算して符号帳に追加しておけば、それだけ計算量を少

なくすることができる。つまり、例えば図 4 に示すように、各候補ベクトル中の 0 でない要素の部にはその要素の値  $X$  と、その自乗値  $X^2$  とを並べて記憶しておけばよい。この例は図 1 に示した符号帳に対するものである。 $X^2$  の代わり  $w(i) X^2$  を記憶しておいてもよい。

【0015】ところで上述した、この発明による符号帳（例えば図 1 に示すもの）を作成するには、例えば次のようにすればよい。これはいわゆる一般化ロイド法を変形したものであって、必要とする候補ベクトルの数（ $N$  個）の初期再生ベクトルを設定する（ $S_1$ ）、この設定は例えば乱数を発生させて、それぞれ与えればよい。次に全学習サンプルを最も近い再生ベクトルに帰属させ（ $S_2$ ）、各再生ベクトルごとにこれに帰属した学習サンプルの中心値（重心）を求める（ $S_3$ ）。一般化ロイド法では、この各中心値を新たな再生ベクトルとするが、この発明では、この各中心値に対し、例えば図 1 に示したベクトル配列方向における 0 でない要素の繰り返し位置を除き、他の全ての要素を 0 とする。つまり図 1 \*

$$\begin{aligned}
 D_{n,m} &= w(i) |x(i) - c_0(n,i) - c_1(m,i)|^2 \\
 &= w(i)x(i)^2 \\
 &\quad - 2 w(i)x(i)(c_0(n,i) + c_1(m,i)) \\
 &\quad + w(i)(c_0(n,i)^2 + c_1(m,i)^2 + 2 c_0(n,i) c_1(m,i)) \\
 &\quad \dots\dots (4)
 \end{aligned}$$

は  $i = 0$  から  $p - 1$  まで

この場合も第 1 の実施例と同様に右辺第 2 項と第 3 項のみを各要素の縦方向（ベクトル配列方向）に部分和を  $c_0(n,i)$  と  $c_1(m,i)$  で各々更新していけばよ

$$\begin{aligned}
 d_{n,m} &= - 2 w(i) x(i) (c_0(n,i) + c_1(m,i)) \\
 &\quad + w(i) (c_0(n,i)^2 + c_1(m,i)^2 + 2 c_0(n,i) c_1(m,i)) \\
 &\quad \dots\dots (5)
 \end{aligned}$$

この計算結果を、各組み合わせ候補ベクトルごとに設けられた部分歪記憶部  $2 1_{n,m}$ （図示せず）の対応するものに累積加算し、 $i = 0$  乃至  $i = p - 1$  まで（5）式を計算した後、部分歪記憶部  $2 1_{n,m}$  中の累積歪が最小のものに対応する候補ベクトルの組み合わせを量子化結果とする。

【0019】この場合、 $c_0(n,i)$  と  $c_1(m,i)$  の 0 でない要素の規則は互いに独立でよい。 $c_0(n,i)$  と  $c_1(m,i)$  とが共に 0 でないことは少なく、つまり（5）式右辺中最後の項は 0 となること  
40 が多く、つまり、この項を計算する必要が少なく、 $i$  ごとに僅かの計算でする場合が多い。このように 2 つの候補ベクトルの和で表す場合には  $w(i) x(i) (c_0(n,i))$  と  $w(i) x(i) (c_1(m,i))$  のそれぞれを計算し、つまり（5）式右辺中の第 1 項のかけを解いたものを計算し、それぞれの値の大きいものから複数の候補ベクトルを残し、これらの候補ベクトルについてのみ（5）式第 2 項、特に  $w(i) c_0(n,i) c_1(m,i)$  を計算することでさらに演算量を削減できる。つまり（5）式の右辺第 1 項は負の値である

\* 中の値の 0 の要素と対応する各中心値中の全ての要素を 0 とし、その新たな再生ベクトルとする（ $S_4$ ）。ステップ  $S_2$  に戻り、この新たな再生ベクトルに対して学習サンプルを帰属させることを行い、以下同様のことを歪が最小になるまで繰り返す。

【0016】図 5 B に示すように、2 つの符号帳  $1 2_0$  と  $1 2_1$  からそれぞれ各 1 つの候補ベクトル  $c_{0n}$ 、 $c_{1n}$  を取り出し、これらを加算し、その加算したものと入力ベクトル  $x$  との距離を計算部 1 4 で行い、その距離が最小となる候補ベクトル  $c_{0n}$ 、 $c_{1n}$  の組み合わせを制御部 1 4 で選択して、その候補ベクトルの番号を量子化結果として出力するベクトル量子化方法がある。この場合にも、この発明を適用できる。この場合の入力ベクトル  $x$  と組み合わせ候補ベクトル  $c_{0n}$ 、 $c_{1n}$  との距離、つまり歪  $D_{n,m}$  は次のようになる。ただし、 $m = 0, 1, 2, \dots, N - 1$  である。

【0017】

い。

【0018】つまり、要素番号  $i$  を指定して、各候補ベクトルの組み合わせについて、入力ベクトルとの要素間歪  $d_{n,m}$  を次式で計算し、

$$\dots\dots (5)$$

から、この絶対値が大きいものは歪が小さいから、歪の小さいものについて演算すればよい。第 1 の実施例についても  $w(i) x(i) c(n,i)$  の大きい複数の候補についてのみ計算してもよい。

【0020】また一方の符号帳  $1 2_0$  の  $c_0(n,i)$  は全ての要素が 0 でなく、他方の符号帳  $1 2_1$  の  $c_1(m,i)$  は 0 でない要素が等間隔になるように配置することで上記と同様の演算量削減効果があることは明らかである。また、この発明は 2 個以上の符号帳を持つベクトル量子化への拡張も容易である。

【0021】

【発明の効果】符号化を実時間で実行するために信号処理プロセッサがよく使われるが、このようなプロセッサでは積和演算と対象となるデータのアドレスの更新が並行して実行できるという利点がある。ただしアドレスの更新は等間隔であるという条件がつく。この発明ではベクトル量子化の距離計算を縦方向に 0 でない要素のみを等間隔で実行するため、つまり図 2 中のステップ  $S_6$  のアドレス更新が並行して実行でき、高速演算が可能である。一方、符号帳中の候補ベクトルの 1 つ 1 つは 0 でな

い要素がさまざまな間隔に配置されているため、すべての要素が0でない要素が等間隔に配置された場合に比較して歪の増加は僅かで抑えられる。また周波数領域の係数を量子化する際などに、低次の要素は密に高次の要素は疎にするといった調整が可能で演算量を効果的に削減できる。

【図面の簡単な説明】

【図1】この発明による符号帳の例を示す図。

【図2】この発明によるベクトル量子化法の処理手順の

例を示す流れ図。

【図3】Aは符号帳の各要素番号と最初に0でない要素のベクトル番号n。と、0でない要素間の0の数 nとの関係を示すテーブル、Bはこの発明の符号帳の作成方法の一例を示す流れ図である。

【図4】この発明による符号帳の他の例を示す図。

【図5】ベクトル符号化法を示す機能的ブロック図。

【図6】従来の符号帳の構成例を示す図。

【図1】

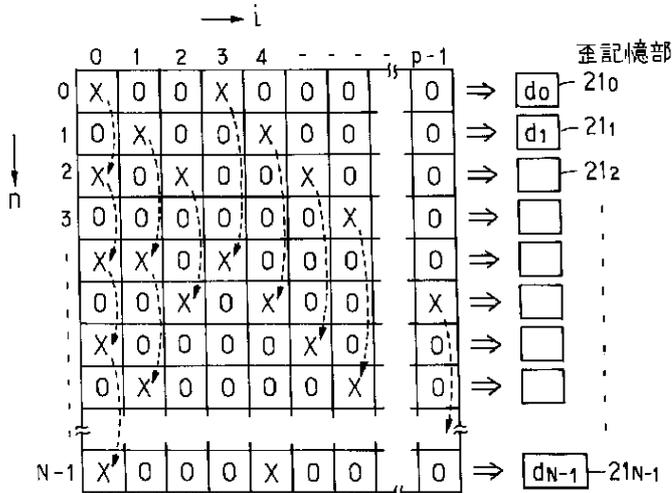


図 1

【図5】

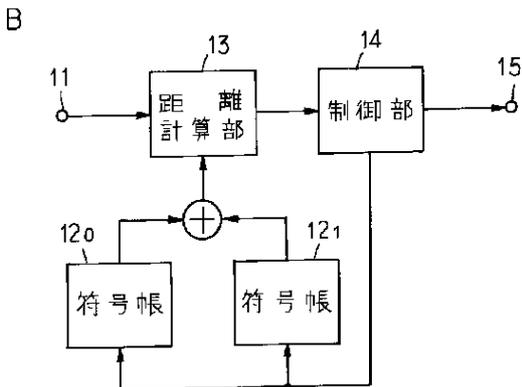
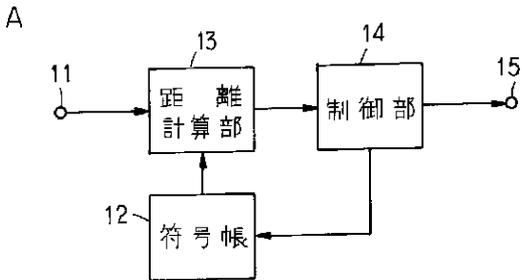


図 5

【図2】

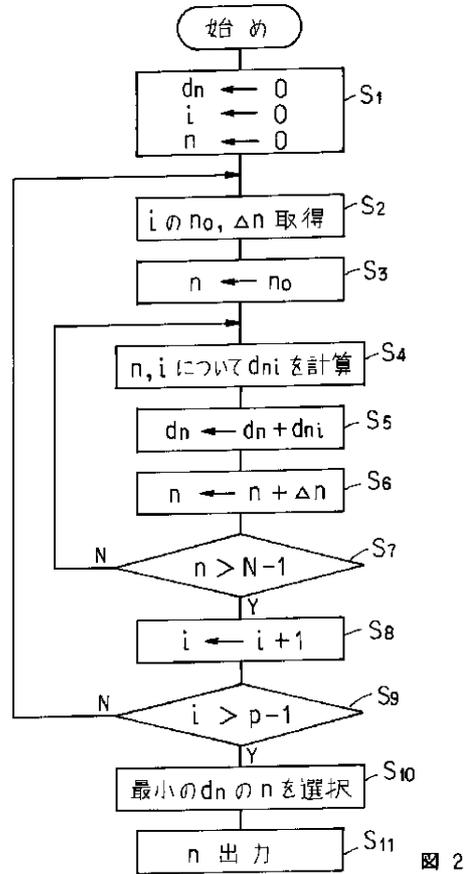


図 2

【図3】

A

i	0	1	2	3	- - - -	p-1
n <sub>0</sub>	0	1	3	0	- - - -	
Δn	1	2	2	3	- - - -	

B

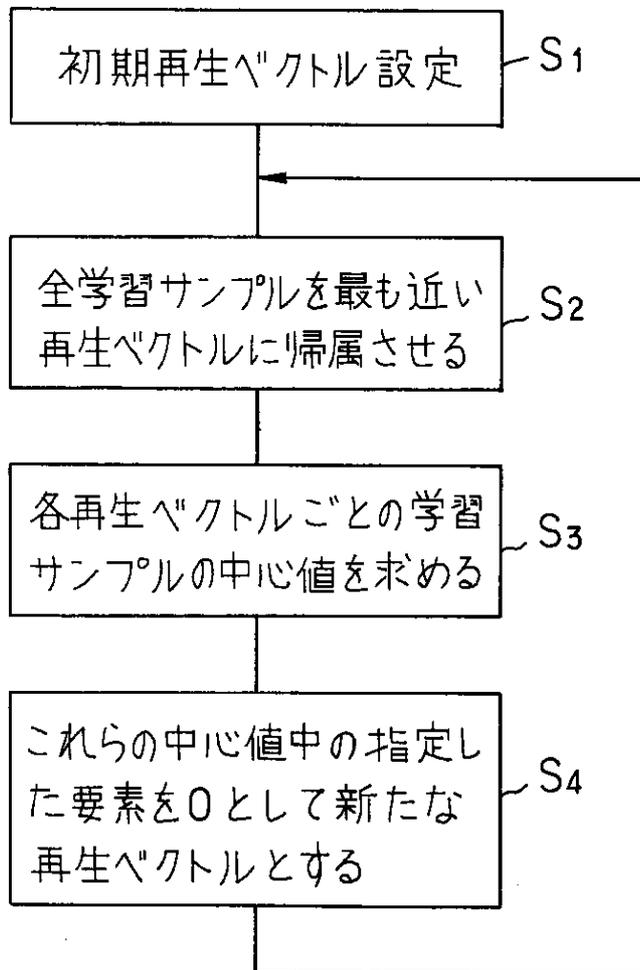


図 3

【図 4】

		→ i								
		0	1	2	3	-	-	-	-	p-1
0		X, X <sup>2</sup>	0	0	X, X <sup>2</sup>	0	0	0	0	0
1		0	X, X <sup>2</sup>	0	0	X, X <sup>2</sup>	0	0	0	0
2		X, X <sup>2</sup>	0	X, X <sup>2</sup>	0	0	X, X <sup>2</sup>	0	0	0
3		0	0	0	0	0	0	X, X <sup>2</sup>	0	0
n		X, X <sup>2</sup>	X, X <sup>2</sup>	0	0	0	0	0	0	0
		0	0	X, X <sup>2</sup>	X, X <sup>2</sup>	X, X <sup>2</sup>	0	0	0	X, X <sup>2</sup>
		X, X <sup>2</sup>	0	0	0	0	X, X <sup>2</sup>	0	0	0
		0	X, X <sup>2</sup>	0	0	0	0	X, X <sup>2</sup>	0	0
		X, X <sup>2</sup>	0	X, X <sup>2</sup>	0	0	0	0	0	0
N-1		X, X <sup>2</sup>	0	0	0	X, X <sup>2</sup>	0	0	0	0

図 4

【図 6】

		→ i								
		0	1	2	3	4	-	-	-	p-1
0		0	X	0	0	0	X	0	0	0
1		X	0	0	X	0	X	0	0	0
2		0	0	X	0	0	0	0	0	0
3		X	X	0	0	0	0	0	0	X
n		0	0	X	0	0	0	X	0	0
		0	0	0	X	0	0	0	0	X
		0	X	0	0	X	X	0	0	0
		X	0	0	0	0	X	0	0	X
N-1		0	0	0	X	0	0	X	0	0

図 6

フロントページの続き

- (56) 参考文献 特開 平 1 - 319799 ( J P , A )
- 特開 平 3 - 209920 ( J P , A )
- 特開 昭 62 - 188575 ( J P , A )
- 特開 平 1 - 218280 ( J P , A )
- 特開 平 4 - 170113 ( J P , A )
- 特開 平 5 - 37397 ( J P , A )

(58) 調査した分野(Int.Cl.<sup>7</sup>, DB 名)  
H03M 7/30