

(19)日本国特許庁(JP)

(12)特許公報 ( B 2 )

(11)特許番号

第 2 8 9 7 9 4 0 号

(45)発行日 平成11年(1999)5月31日

(24)登録日 平成11年(1999)3月12日

(51) Int. Cl. <sup>6</sup>	識別記号	F I
G 1 0 L	9/14	G 1 0 L 9/14 J
	9/18	9/18 E
H 0 3 M	7/30	H 0 3 M 7/30 B

請求項の数 2

(全 9 頁)

(21)出願番号	特願平3-180819	(73)特許権者	000004226 日本電信電話株式会社 東京都新宿区西新宿三丁目19番2号
(22)出願日	平成3年(1991)7月22日	(72)発明者	大室 伸 東京都千代田区内幸町一丁目1番6号 日本 電信電話株式会社内
(65)公開番号	特開平5-27798	(72)発明者	守谷 健弘 東京都千代田区内幸町一丁目1番6号 日本 電信電話株式会社内
(43)公開日	平成5年(1993)2月5日	(72)発明者	間野 一則 東京都千代田区内幸町一丁目1番6号 日本 電信電話株式会社内
審査請求日	平成6年(1994)5月26日	(74)代理人	弁理士 草野 卓
		審査官	酒井 伸芳

最終頁に続く

(54)【発明の名称】音声の線形予測パラメータ符号化方法

1

(57)【特許請求の範囲】

【請求項 1】 音声スペクトル包絡特性を表す線形予測パラメータを、フレームと呼ばれる一定時間間隔で算出し、

そのパラメータを、複数フレーム分蓄えて行列形式に変換し、

その行列形式で表現されたパラメータの入力行列を、あらかじめ決められた代表行列で表現して量子化する線形予測パラメータ符号化方法において、

あらかじめ決められた代表行列のうちで、上記入力行列との誤差が最小となる第 1 段の代表行列を決定し、

2 以上の任意の数 N について、上記あらかじめ決められた代表行列のうちで、既に決定されている第 1 段～第 N - 1 段までの代表行列の和で表現される行列と、上記入力行列とを、それぞれより小さいサイズの複数の副行列

2

に区分けし、

それぞれ区分けされた小さいサイズの副行列をあらかじめ決められた小さいサイズの副代表行列で表し、

それぞれ上記和の副行列に対応する副代表行列を加算した副行列と、上記入力行列の対応する副行列との誤差が最小となるように副代表行列を決定し、

これら決定された副代表行列を上記区分け前の行列形式に合成して、

第 N 段の代表行列を決定し、

上記入力行列の量子化値を上記第 1 段～第 N 段までの代表行列の和で表現することを特徴とした音声の線形予測パラメータ符号化方法。

【請求項 2】 上記第 1 段～第 N - 1 段までの代表値を決定する際に、最適な代表行列ただ 1 つに決定してしまいうのではなく、いくつかの候補を残した後、第 N 段の代

表行列を決定した時点で、第 1 段～第 N 段までの代表行列の和で表される行列と、入力行列との誤差が最小となるように第 1 段～第 N 段までの代表行列を決定することを特徴とする請求項 1 記載の音声の線形予測パラメータ符号化方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は音声スペクトル包絡特性を表す線形予測パラメータを複数フレーム分蓄えて行列形式とし、この行列形式のパラメータを、あらかじめ

【0002】

【従来の技術】従来の音声符号化系において、音声スペクトル包絡特性を表す線形フィルタの係数は、線形予測分析を一定時間間隔で行って計算され、偏自己相関 (PARCOR) 係数や、線スペクトル対 (LSP) などのパラメータに変換、量子化され、デジタル符号に変換した後、記憶、または伝送された。これらの方法の詳細は、例えば、古井貞熙著「デジタル音声処理」(東海大学出版会)に記載されている。

【0003】この符号化において、係数を更新する時間間隔を長くすれば、記憶または伝送のための情報量を減らすことができるが、あまり長くしすぎると音声を作成するとき記憶または伝送しなかった時間におけるフィルタ係数の推定精度が低下し、再生音声の品質劣化につながった。そこで、一般には上記時間間隔を 20 ミリ秒程度以下に設定している。

【0004】このような点から、より高効率な符号化を行うための方法として、マトリクス量子化と呼ばれる方法がある。これは、線形予測分析は短い時間間隔、例えば 10 ミリ秒から 20 ミリ秒程度の間隔で行っており、数組の分析結果をひとまとめにして量子化する方法である。線形予測パラメータは、時間方向、およびパラメータの次元方向にそれぞれ相関性がある。マトリクス量子化はこの相関性をうまく利用して量子化の効率をあげる方法である。

【0005】

【発明が解決しようとする課題】しかしこの方法によって、直接数組の予測パラメータを量子化しようとすると、非常に多くの計算量と、符号帳のための多くの記憶とが必要になり、現実的なハードウェア規模のもとではそれほど効率をあげることができない。この発明の目的は、マトリクス量子化が量子化歪みを小さくする観点で高効率であるという利点を生かしつつ、現実的な計算量と記憶量のもとでの高効率な線形予測パラメータの符号化方法を提供することにある。

【0006】

【課題を解決するための手段】この発明によればあらかじめ決められた代表行列のうちで、入力行列との誤差が

最小となる第 1 段の代表行列を決定し、2 以上の任意の数 N についてあらかじめ決められた代表行列のうちで、既に決定されている第 1 段～第 N - 1 段までの代表行列の和で表現される行列に、第 N 段の代表行列を加算した行列と、入力行列との誤差が最小となるように第 N 段の代表行列を決定し、入力行列の量子化値を第 1 段～第 N 段までの代表行列の和で表現するが、特にこの発明では計算量と記憶量を低減するため、2 段またはそれ以上に分割し、2 段のときは 2 段目を、それ以上のときは 2 段目以降の任意の段を複数のサブマトリクス(副行列)に区分けし、各サブマトリクスごとに最適量子化する。

【0007】一般に線形予測パラメータを次元方向と時間方向とに並べたマトリクスの各行、各列の要素の間には相関性が大きい。適当なビット数のマトリクス量子化器で量子化した、誤差マトリクスの各要素は、ガウス分布に従い、相関性はないかまたは最初のマトリクスに比べて非常に小さいものと考えられる。従って、2 段目以降の量子化には、最初のマトリクスを適当なサイズに区分けして、それぞれに最適量子化をしても、品質の劣化は小さい。しかも 1 段目と 2 段目の間に前記ディレイドディシジョンを用いれば、更に劣化を小さく抑えることができる。

【0008】このようにして線形予測パラメータのマトリクス量子化において、量子化の手順を複数の簡易な手順の組に分解することにより、計算量と記憶量とを現実的なハードウェア規模において実現可能なレベルにまで低減する。しかも簡易な手順の組に分解する際に、量子化効率の低下を極力抑えるように、各段の間に、ディレイドディシジョンと呼ばれる手法を用いる。これは、各段ごとに最適な量子化値をひとつに決定してしまうのではなく、いくつかの量子化候補を残しておき、全段における最適量子化値をダイナミックプログラミングの手法によって決定する。このとき、残しておく候補数は多いほど量子化性能の低下を防ぐことができるが、逆に計算量は候補数に比例して増加するため、目的とするハードウェア規模との兼ねあいになる。一般には 4～8 個あれば十分である。

【0009】

【作用】この発明において、複数の簡易な手順の組に対して、適当なビット数を割り当てることによって量子化効率の低下を抑えながら計算量の低減を実現することができる。なお、簡易な手順の組に処理を分ける際に、細かく割れば割るほど計算量は低減できるが、除々に量子化品質が劣化することはやむを得ない。

【0010】また、分割後のマトリクスサイズや、分割後の各サブマトリクスの重要性(聴覚への影響など)の違いによって、ビット数の割当を不均一にして効果をも上げることができる。

【0011】

【実施例】以下、この発明の説明に先立ち、行列の多段

ベクトル量子化方法を図面を参照して説明する。図 1 に音声の線形予測パラメータ符号化法を適用した符号化装置の構成例を示す。入力端子 1 からは、標準化されたデジタル化された音声信号  $s(t)$  が入力される。線形予測分析部 2 では、音声信号  $s(t)$  の  $N$  個のサンプルをいったんデータバッファに蓄えた後、これらのサンプルに対して線形予測分析を行って、一組の線形予測係数  $a_i(k)$ , ( $i = 1, 2, \dots, p$ ) を算出する。ここで、パラメータ  $p$  は分析次数と呼ばれ、一般には 10 ~ 16 程度の値が用いられる。また上記  $N$  サンプルの単位はフレーム更新周期または単にフレーム周期と呼ばれる。この結果、 $N$  個の入力サンプルに対して  $p$  個の線形予測係数が出力されることになる。そこでここでは、線形予測係数の時刻を表す単位として、 $N$  個のサンプルを単位とした時間単位を  $k$  で表し、「第  $k$  フレームの  $i$  次の線形予測係数  $a_i(k)$ 」などと呼ぶことにする。なお、各フレーム時刻  $k$  において、 $p$  個ずつの線形予測係数が得られることは言うまでもない。これらの処理方法の詳細は、前述の古井の著書に記載されている。

【0012】線スペクトル対算出部 3 では、 $p$  個の線形予測係数を同じく  $p$  個の線スペクトル対、 $f_i(k)$  に変換する。ここで線形予測係数を線スペクトル対に変換したのは、線スペクトル対の性質が、前述の手順の分割に対して、品質の劣化が少ないという利点を利用するためである。しかし、この線スペクトル対算出部 3 は必ずしも必要ではなく、線形予測係数  $a_i(k)$  を直接バッファ部 4 に入力してもよい。また、線スペクトル対のほか、偏自己相関 (PARCOR) 係数など、線形予測係数と相互変換が可能な任意のパラメータに変換してもよい。

【0013】バッファ部 4 では、 $p$  個の線スペクトル対を  $m$  フレーム分、即ち  $m \times p$  個のパラメータ  $f_i(k-j)$ ,  $i = 1, 2, \dots, p$   
 $j = m-1, m-2, \dots, 1, 0$   
 を蓄え、 $m$  フレーム分たまった時点で、これらの値をマトリクス量子化部 5 に供給する。図 1 では説明を簡単にするため、 $m = 4$  の場合について示した。

【0014】マトリクス量子化部 5 によって量子化された  $m \times p$  個のパラメータの組すなわちマトリクス

【0015】

【数 1】

$$F_{ij}^*(k')$$

は、符号化部 6 に送られ、伝送路または蓄積形態に合った形式に符号化され、伝送または蓄積される。ここで、時刻を表すインデックスとして  $k$  を用いたのは、インデックス  $k$  で表される  $m$  フレーム分を 1 単位とする時間単位であるためである。

【0016】再生側では、上記符号からマトリクス

【0017】

【数 2】

$$F_{ij}^*(k')$$

を生成し、マトリクス分解部 8 によって各フレームごとの線スペクトル対に戻した後、バッファ部 9 に蓄える。バッファ部 9 は、 $m$  フレーム分の線スペクトル対を 1 フレームずつ線形予測係数変換部 10 に送る。線形予測係数変換部 10 は、受け取った線スペクトル対の量子化値を線形予測係数  $a_i^*(k)$  に変換して出力する。

【0018】図 2 は図 1 中のマトリクス量子化部 5 を、簡易な (小規模な) マトリクス量子化器の縦続接続で実現した一例である。図は簡単のため、3 段の構成を示す。ここで、図 1 におけるマトリクス量子化部 5 への入力

$$f_i(k-j), i = 1, 2, \dots, p$$

$$j = m-1, m-2, \dots, 1, 0$$

をマトリクス形式に並べたものを

$$F_{ij}(k)$$

と表す。段数を  $N$  とすると図 2 に示すように、 $N$  組の符号帳 11, 14, 18、歪判定部 12, 16, 20、 $(N-1)$  組の行列加算器 15, 19 の縦続的な組み合わせからなる。行列加算器は 1 段目が不要である。

【0019】まず、第 1 符号帳 11 に保持する  $R_1$  個の各コードマトリクス (代表行列)  $C_{ij}^{(1)}(r)$  を順次歪み判定部 12 に送る。歪み判定部 12 では予め定義された歪み尺度に基づいて入力行列  $F_{ij}(k)$  とコードマトリクス  $C_{ij}^{(1)}(r)$  との歪みを計算し、最も歪みの小さい順に  $U$  個のコードマトリクス  $C_{ij}^{(1)}(r_u)$ , ( $u = 1, 2, \dots, U$ ) を 1 段目の量子化候補値

【0020】

【数 3】

$$F_{ij}^{*(1)}(u)$$

として出力する。ここで予め定義された歪み尺度とは、任意の尺度を用いてよいが、通常は線スペクトル対のユークリッド距離や、いったんケプストラムと呼ばれるパラメータに変換したのち、ユークリッド距離をはかる等の尺度が用いられる。上記 2 つの尺度では、後者の方が人間の聴覚特性に似ているという観点から、一般に優れていると言われている。次に 2 段目の量子化部では、第 2 符号帳 14 の保持する  $R_2$  個のコードマトリクス (代表行列) のうち、 $r$  番目のマトリクスを  $C_{ij}^{(2)}(r)$  とし、これを 1 段目の第  $u$  番目の量子化候補値

【0021】

【数 4】

$$F_{ij}^{*(1)}(u)$$

に行列加算器 1 5 で加えてマトリクス

【 0 0 2 2 】

【 数 5 】

$$F_{ij}^{*(1)}(u) + C_{ij}^{(2)}(r)$$

を求める。歪み判定部 1 6 では、1 段目と同様にしま 10  
ず入力行列

$F_{ij}(k)$

と

【 0 0 2 3 】

【 数 6 】

$$F_{ij}^{*(2)}(u) = F_{ij}^{*(1)}(u_r) + C_{ij}^{(2)}(r)$$

$$u = 1, 2, \dots, U$$

を 2 段目の量子化候補値として量子化候補保持部 1 7 に 20  
保持する。以後、各段において同様の手順を繰り返すこ  
とにより、N 段の小規模マトリクス量子化器の縦続接続  
によるひとつのマトリクス量子化器を構成する。なお、  
最終段においては、U 個の量子化候補を保持する必要が  
なく、最も歪みの小さい最適な量子化値に決定してよ  
い。

【 0 0 2 6 】 図 3 は図 2 の構成よりもさらに計算量を削 5  
減するためのこの発明を適用したマトリクス量子化部 5  
の構成の一例である。図 3 の構成は、図 2 の構成におい  
て、段数を 2 段にし、第 2 段目を変形した形になってい  
る。1 段目量子化部 3 9 は図 2 における 1 段目と全く同  
様に、U 個の量子化候補値

【 0 0 2 7 】

【 数 9 】

$$F_{ij}^{*(1)}(u), (u = 1, 2, \dots, U)$$

を得る。次に、この量子化候補値を順に 2 段目の量子化  
部 4 0 に送る。2 段目の量子化部 4 0 では、まず、サブ  
マトリクス分割部 2 4 で 1 段目の量子化 (候補) マトリ 40  
クスを、適当なサイズのサブマトリクスに区分けする。  
図 3 では、簡単のため 4 つに区分けしたものと示し  
た。区分けする個数及び区分けされたサブマトリクスの  
サイズは、量子化の品質 (能率) という観点からすれ  
ば、個数は少なく、サイズは大きい方が優れている。し  
かし、この発明では、計算量をハードウェア規模に見合  
った量に削減することを目的としているので、ハードウ  
ェア規模によって個数とサイズが制約されることにな

$$F_{ij}^{*(1)}(u) + C_{ij}^{(2)}(r)$$

との歪みが最小になるような u を選び、 $u_r$  とする。さ  
らに、

$F_{ij}(k)$

と

【 0 0 2 4 】

【 数 7 】

$$F_{ij}^{*(1)}(u_r) + C_{ij}^{(2)}(r)$$

との歪みが小さい順に U 個のマトリクス

【 0 0 2 5 】

【 数 8 】

る。現在のハードウェア技術の進歩を考慮すれば、1 段  
目も含めて、各小規模マトリクス量子化器に割り当てる  
ビット数が、8 ~ 1 0 ビット程度になるようなサブマト  
リクスの個数と、サイズに設定すれば、実時間処理が可  
能であると考えられる。ただし、ハードウェア価格を下  
げたい場合には、さらに多くのサブマトリクスに区分け  
しても良い。ここでは、一例として、図 4 に示すように  
区分けされたものとする。また、区分けするとき、必  
ずしも連続する行、および連続する列でサブマトリクス  
をつくる必要はなく、例えば、もとのマトリクスの 1 行  
目と 8 ~ 1 0 行目、1 列目と 4 列目で 4 x 2 のサブマト  
リクスを作っても良い。

【 0 0 2 8 】 2 段目量子化部 4 0 では、サブマトリクス  
分割部 4 1 で入力マトリクス

$F_{ij}(k)$

も同様の区分けをする。各サブマトリクスに区分けされ  
た

$F_{ij}(k)$

と

【 0 0 2 9 】

【 数 1 0 】

$$F_{ij}^{*(1)}(u), (u = 1, 2, \dots, U)$$

の第 s 番目のサブマトリクスのサイズを  $p_s \times m_s$  と  
し、それぞれ

【 0 0 3 0 】

【 数 1 1 】

$$F_{s_{ij}}^{(k')}$$

10

$$F_{s_{ij}}^{*(1)}(u), \quad (u = 1, 2, \dots, U)$$

$$(i = n_s, n_s + 1, \dots, n_s + p_s - 1)$$

$$(j = k_s, k_s + 1, \dots, k_s + m_s - 1)$$

とおく。ただし、

$$s p_s = p$$

$$s m_s = m$$

とする。

【0031】区分けされた1段目の第u番目の量子化候補サブマトリクスは、それぞれs番目の行列加算器に送られる。第2, s符号帳は $p_s \times m_s$ のコードマトリクス(代表行列)を $R_{2s}$ 個蓄えている。これを

$$C_{ij}^{(2s)}(r)$$

とする。これらを順に行列加算器に送り、

$$F_{s_{ij}}(k)$$

と

【0032】

【数12】

$$F_{s_{ij}}^{*(1)}(u) + C_{ij}^{(2s)}(r)$$

との歪みが最も小さくなるコードrを求める。すべてのsについて1段目の第u番目の量子化候補に対する最適量子化サブマトリクスを決定した後、すべてのサブマトリクスは、マトリクス合成部37に送られ、もとのマトリクス形式に戻される。これを

【0033】

【数13】

$$F_{ij}^{*(2)}(u)$$

で表す。これと、入力マトリクス

$$F_{ij}(k)$$

との歪みが最小になるように一段目の候補の中から最適な候補uを歪み判定部38で決定し、マトリクス量子化器5の出力とする。

【0034】なお、各符号帳の作成方法については説明を省略したが、LBGアルゴリズムによって高性能な符号帳を設計することができる。LBGアルゴリズムについては、文献Y. Linde, A. Buzo, R. M. Gray: "An Algorithm for Vector Quantizer Design", IEEE Trans. Comm. COM-28, p. p 84-95 1980に記載されている。

【0035】この発明は、音声符号化への応用のほか、

10 音声認識や話者認識などの、線形予測パラメータを音声の特徴量として用いるアプリケーションへの応用が可能である。またこの発明は、その一部または、全部を論理回路の組み合わせによるハードウェアで実現してもよいし、一部または全部をソフトウェアプログラムの形で実現してもよい。

【0036】

【発明の効果】この発明によって実際にどれだけ計算量が削減されるかを次の例によって示す。例えば、10次元の線形予測パラメータを時間方向に4つまとめたマトリクスを作り、これを量子化するために40ビットを与えたとする。これをそのままマトリクス量子化しようとすると、記憶パターンは

2<sup>40</sup>(パターン)

計算量は、

$$80 \times 2^{40} \text{ (回)}$$

の加減算、

$$40 \times 2^{40} \text{ (回)}$$

の乗算と、

$$2^{40} \text{ (回)}$$

20 の比較演算が必要となり、現実的なハードウェア規模のもとでは実現不可能である。

【0037】図2に示す構成によって、10ビットずつ4段に分割したとすると、ディレイドディシジョンを用いない、即ち候補が1つの場合の記憶量は、

$$2^{10} \times 4 = 4096 \text{ (パターン)}$$

計算量は、

$$80 \times 2^{10} \times 4 = 327680 \text{ (回)}$$

の加減算、

$$40 \times 2^{10} \times 4 = 163840 \text{ (回)}$$

40 の乗算と、

$$2^{10} \times 4 = 4096 \text{ (個)}$$

の比較演算となり、十分実現可能である。ディレイドディシジョンを用いても、加減算、乗算、比較演算が候補数に比例して増えるのみであるから、4~8個の候補であれば十分可能である。ハードウェアが許せば、分割する段数は少ない方が分割による品質の劣化が少ない。

【0038】この発明の方法、すなわち図3に示す構成によって、1段目のマトリクス量子化に10ビット、2段目を図4の様に6x2のマトリクスを2つと、4x4のマトリクス1つに区分けし、それぞれに10ビットを

割り当てて量子化をしたとすると、ディレイドディジジョンを用いない、即ち候補が1つの場合の記憶量は、 $2^{10} \times 4 = 4096$  (パターン)

計算量は、  
 $(80 + 32 + 24 \times 2) \times 2^{10} = 163840$  (回) の加減算、  
 $(40 + 16 + 12 \times 2) \times 2^{10} = 81920$  (回) の乗算と、  
 $2^{10} \times 4 = 4096$  (個)

の比較演算となり、ビットの分割が上記ひとつめの方法と同じでも、計算量はさらに半分になっている。上記例ではビットの分割を同じにして計算量の比較ができるようにしたが、実際にはもっと細かく割ってもよく、細かく割るほど計算量は低減される。その特別な例として、サブマトリクスが時間方向1次、パラメータ次元方向全次元のベクトルになるように分割すれば、マトリクス量子化の誤差に対して、ベクトル量子化を適用することになり、さらに小さく分割して1行列要素ごとになるように分割すれば、マトリクス量子化の誤差に対してスカラ量子化を適用したことになる。

【0039】さらに、計算量が低減できるのみならず、

2段目の計算を並列処理プロセッサによって、並列処理することが可能になる。マトリクス量子化において最も計算量がかかるのが、コードマトリクスを加算して、歪みを判定し、最適コードを決定する処理である。ひとつめの構成は縦続接続であるため、前から順に処理しなければならないが、2つめの方法では、並列処理ができるため、並列化した分さらに高速に処理することができる。

【図面の簡単な説明】

【図1】マトリクス量子化法による音声の線形予測パラメータ符号化法の一般的構成例を示すブロック図。

【図2】この発明の前提となる構成の要部であり、小規模なマトリクス量子化器を縦続に接続して符号化するマトリクス量子化部の一例を示すブロック図。

【図3】この発明の実施例の要部であり、図2の構成において、2段の量子化部を縦続接続し、さらに2段目を複数のサブマトリクスに区分けして符号化するマトリクス量子化部の一例を示すブロック図。

【図4】サブマトリクスに区分けする区分けの一例を示す図。

【図1】

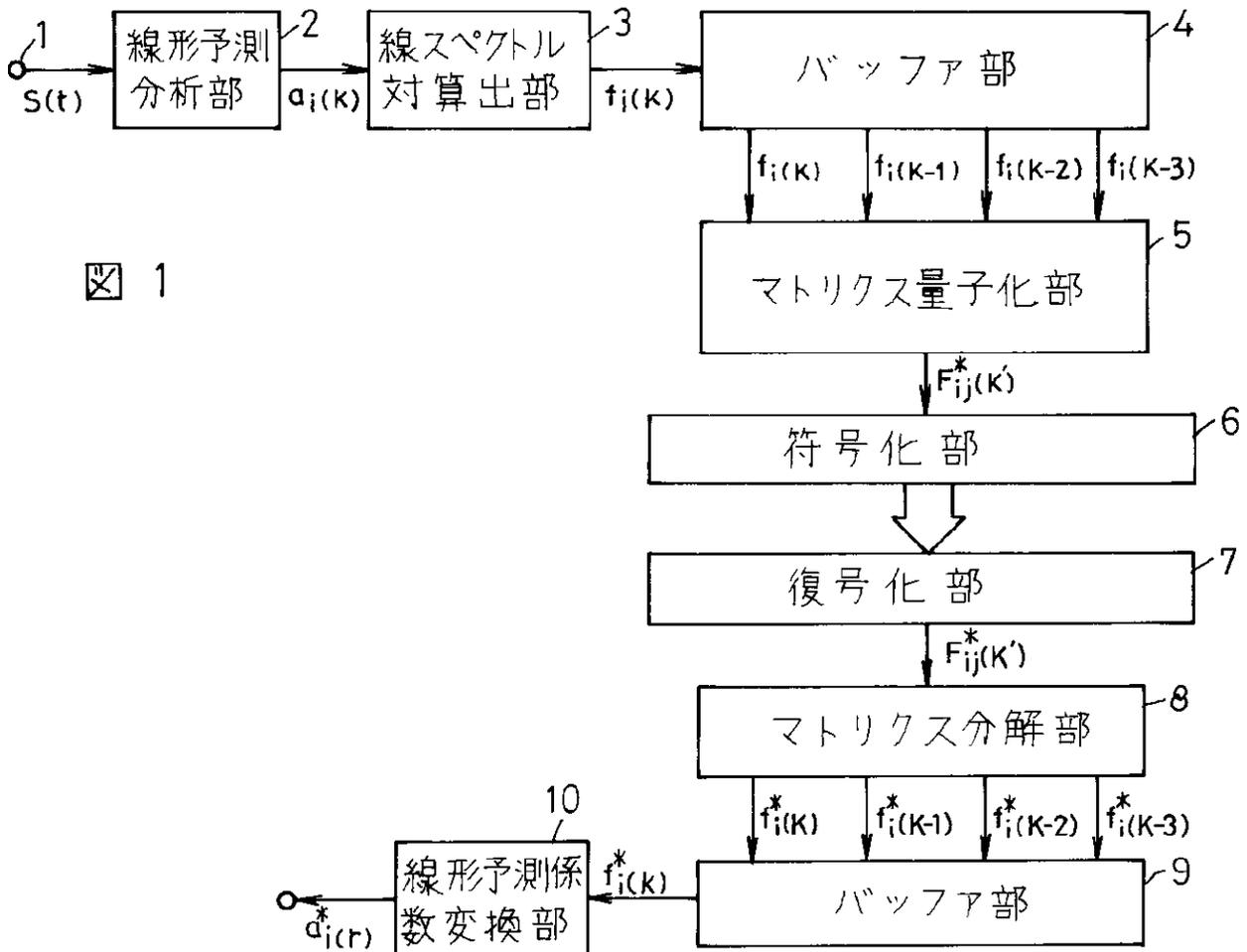
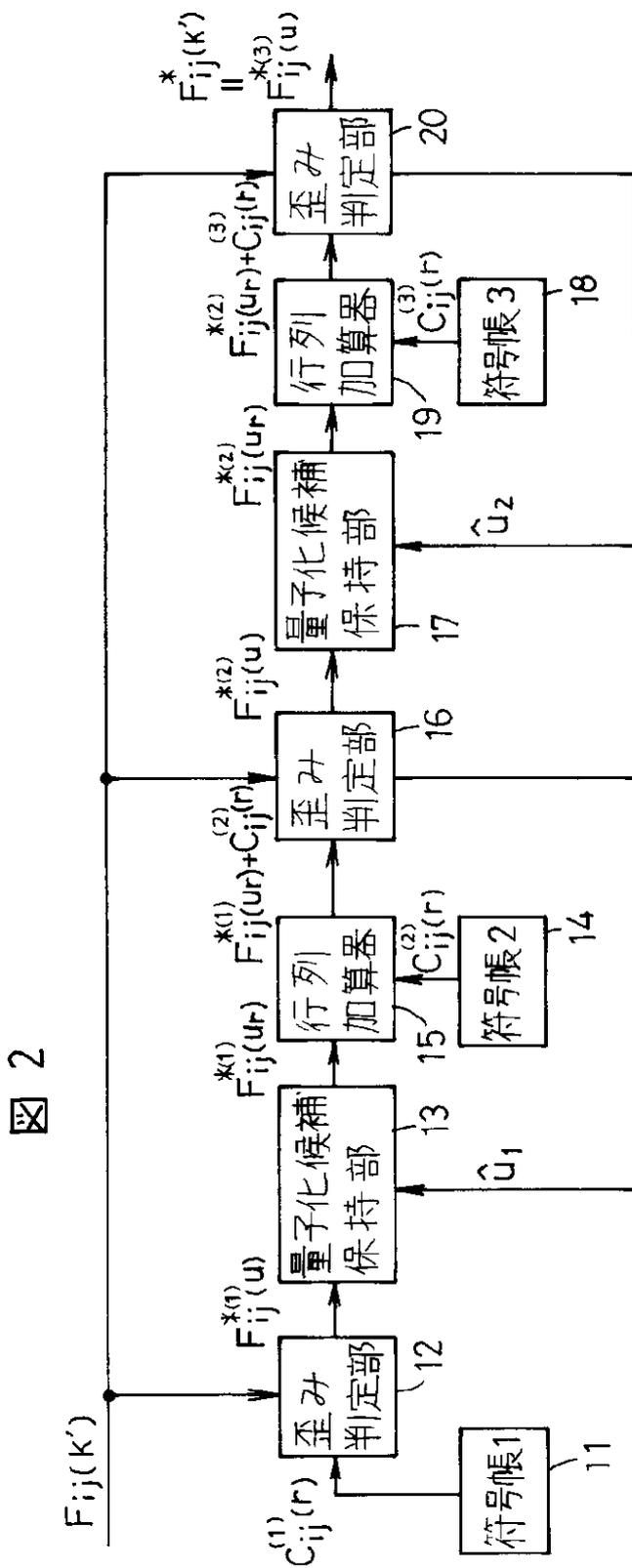


図 1

(7)

【図 2】



【図3】

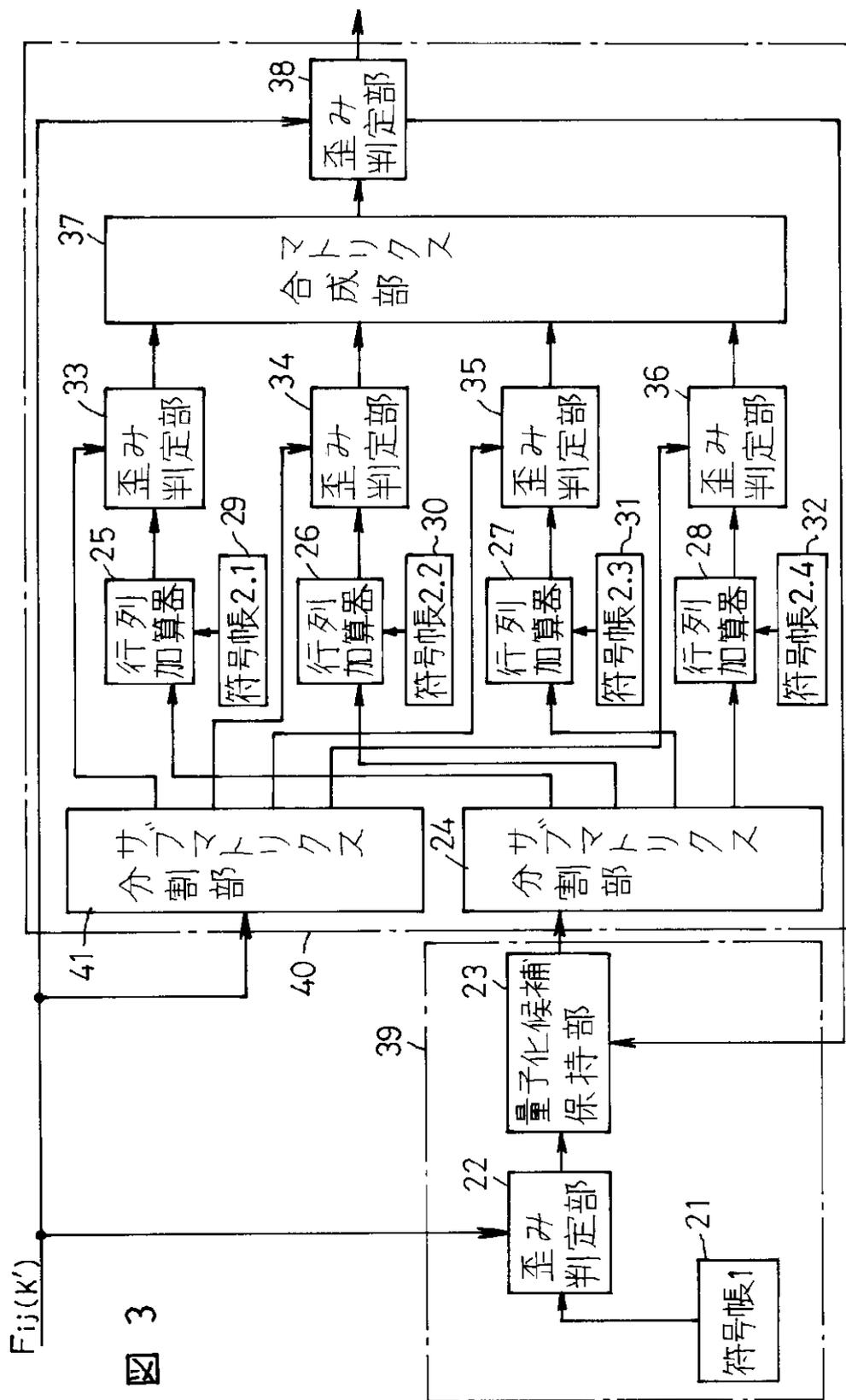
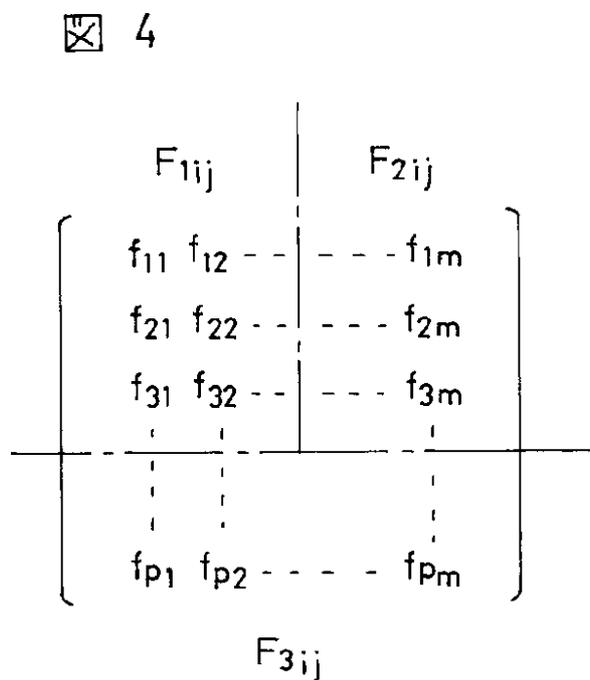


図 3

【図 4】



フロントページの続き

(56)参考文献 特開 昭60 - 224341 ( J P , A )  
 特開 平 1 - 239600 ( J P , A )  
 特開 平 3 - 60529 ( J P , A )  
 特開 平 1 - 233499 ( J P , A )  
 Nam, Nariman, 守谷、"多  
 段ベクトル量子化を用いた L S P パラメ  
 ータの情報源-通信路結合符号化"、電  
 子情報学会技術研究報告 S P 90 - 52、p  
 p . 63 - 70 ( 1990 )

(58)調査した分野(Int.Cl.<sup>6</sup>, DB名)  
 G10L 3/00 - 9/18  
 H03M 7/30  
 J I C S T ファイル ( J O I S )