

(19)



Europäisches Patentamt  
European Patent Office  
Office européen des brevets



(11) Publication number:

**0 614 075 A2**

(12)

**EUROPEAN PATENT APPLICATION**(21) Application number: **94103204.7**(51) Int. Cl.<sup>5</sup>: **G01L 9/14**(22) Date of filing: **03.03.94**(30) Priority: **03.03.93 IT MI930406**(43) Date of publication of application:  
**07.09.94 Bulletin 94/36**(84) Designated Contracting States:  
**DE FR GB IT**(71) Applicant: **ALCATEL ITALIA S.p.A.**  
**Via L. Bodio, 33/39**  
**I-20158 Milano (IT)**(72) Inventor: **Fratti, Marco**

**Via della Birona 9**  
**I-20052 Monza (Milano) (IT)**  
Inventor: **Cucchi, Silvio**  
**Via S. Ibenzio 9**  
**I-20090 Gaggiano (Milano) (IT)**

(74) Representative: **Pohl, Herbert, Dipl.-Ing et al**  
**Alcatel SEL AG**  
**Patent- und Lizenzwesen**  
**Postfach 30 09 29**  
**D-70449 Stuttgart (DE)**

(54) **Method for speech coding using Trellis Coded Quantization for Linear Predictive Coding quantization.**

(57) It is disclosed a method and related circuits for speech coding using Trellis Coded Quantization for encoding of Linear Predictive Coding Coefficienty.

In particular it uses Trellis Coded Quantization for coding Line spectrum Pairs and Reflection Coefficients parameters.

The bit allocation at each quantization step of the Trellis Coded Quantizer is made variable and the quantization error accumulated in quantizing the previous values will be taken in account in the search of the optimum quantization levels of the next values.

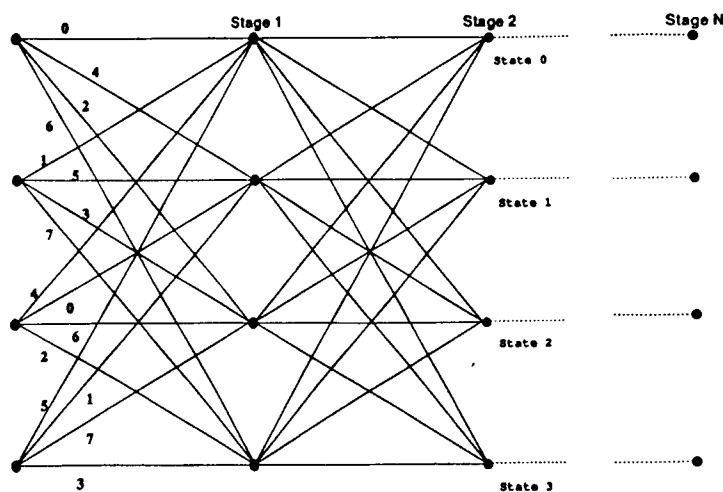


Figure 1

EP 0 614 075 A2

The present invention relates to a method for speech coding as set forth in the preamble of claim 1 and a speech coder as set forth in the preamble of claim 17.

In the telecommunication field it is useful to transmit the information both vocal and video using as less numbers of bits as possible without losing part of the information transmitted.

5 Such aim is achieved by means of suitable coding techniques.

There are a lot of these techniques: each of them has its characteristic features. From the field of the communication theory, and in particular from the modulation theory the Trellis Coded Modulation technique is well known. The Trellis coded Modulation paradigm, combined with well-known quantization theories, gave rise to the Trellis Coded Quantization (TCQ) algorithm.

10 TCQ is a recent technique for efficient scalar encoding of any source.

In particular, it has been introduced by M.W. Marcellin, T.G. Fisher, in "Trellis Coded Quantization of Memoryless and Gauss-Markov Sources", IEEE Trans. on Communication, vol. 38, No. 1, January 1990, for encoding memoryless Gauss-Markov sources. The feature of the trellis coded quantization approach is the use of a structured codebook with an expanded set of quantization levels. Based on the notion of set partitioning introduced by G. Ungerboeck, in "Channel Coding with Multilevel/Phase Signals", IEEE Trans. on Information Theory, Vol. IT-28, Jan. 1982, the trellis structure then prunes the expanded number of quantization levels down to the desired encoding rate. The encoder uses the Viterbi algorithm for finding a vector of quantized scalars that is closest (according to a predetermined metric) to the unquantized vector.

20 Because of the complexity of the matter, only a good knowledge of both the modulation theory (TCM) and of the quantization theory allows a proper implementation and exploitation of this quantization technique.

The main object of the present invention is therefore substantially an efficient and effective way how to apply the TCQ technique.

25 According to the invention, therefore, the method for speech coding is constructed as set forth in claim 1 and the speech coder as set forth in claim 17.

Further features of the invention are explained in the depending claims.

Enboudiments of the invention will now be explained in detail with reference to the accompanying drawings, in which figure 1 is showed a general structure of a Trellis Coded Quantizer; figure 2 is a Trellis Coded Quantizer with variable bit allocation; figure 3 is a TCQ scheme for LSP difference quantization; figure 4 is an updated paths in the TCQ.

The trellis encoder is completely specified by:

- The trellis topology (i.e. the connections among the successive states or, equivalently, the description of the underlying finite-state machine).
- The quantization levels associated with each state transition.

35 In Figure 1, a possible structure for a TCQ is depicted. The labels assigned to each trellis branch are arbitrary, as well as the quantization level partition.

This particular structure is depicted for the sake of clearness and may not correspond to a 'physical' one.

40 The 4-state trellis is fully connected. The number associated to each state transition (branch) represents the quantization value corresponding to that branch. The trellis is a N-stage one, that is, it is employed for coding a N-component input vector.

Note that 8 possible scalar quantization values are present; however, each trellis state 'sees' only a 4-value subset, as function of its transition to a future state.

45 In principle, an exhaustive procedure should be employed for identifying the best quantized vector with respect to a N-value input vector.

This means that one should identify each possible trellis path and, according to the quantization value associated to each path step, an overall quantization error should be constructed. In the example depicted in Figure 1, this exhaustive procedure would imply identifying  $4^N$  quantized vectors and measuring the quantization error for each of them.

50 This tremendous amount of computation is avoided by using the well-known Viterbi algorithm as described by G.D. Forney, in "The Viterbi algorithm", IEEE Trans. on Information Theory, Vol. IT-28, Jan. 1982, that, although operating in a step-by-step fashion, guarantees to find the optimal solution.

With reference to Figure 1, it is easily understood that, in case a scalar quantization technique was employed for encoding each input vector component,  $3N$  bits would be necessary.

55 On the contrary, by using a TCQ, the quantization process would require  $2N + 2$  bits (where the binary representation of the TCQ 'winning' best initial -or final- state is taken into account).

A trellis scheme like the one in Figure 1 is an example of what is generally found in the literature. That is, the topological configuration of the trellis is the same at each quantization step.

Similarly, the quantization level number is the same at each quantization step.

This configuration may be not the ideal one in case one needs to quantize a vector whose scalar components have a different 'importance scale', according to a predefined performance criterium. In this case, a different bit/sample number may be necessary for each vector component.

5 This problem can be solved in the following way: suppose to start with a given topological configuration for the trellis (that is, suppose to start with the trellis depicted in Figure 1); an increase-decrease in the bit/sample assignment at each quantization step can be obtained as follows.

- Bit/sample number increase:

10 this can be obtained by simply adding one or more parallel transitions to the state branches (that is, we have multiple branches at each state transition). The bit/sample number is thus increased according to the number of parallel transitions that are added.

Referring to the example in Figure 2, it is immediate to verify that in the first quantization step (Step 1) we have doubled the quantization level number (from 8 to 16) and, accordingly, the bit/sample configuration (i.e. from 2 to 3 bits).

15 The quantization level partition associated to each parallel transition can be derived from the optimal set partition theory, as described in M.W. Marcellin, T.G. Fisher, "Trellis Coded Quantization of Memoryless and Gauss-Markov Sources", IEEE Trans. on Communication, vol. 38, No. 1, January 1990, and G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals", IEEE Trans. on Information Theory, Vol. IT-28, Jan. 1982.

- Bit/sample number decrease:

20 This can be obtained by changing the trellis topology in a trivial way.

In particular, the state transition number can be pruned down to the desired encoding rate.

25 With reference to Figure 2, in the third quantization step (Step 3) the encoding rate is halved (and, obviously, the same is true for the quantization level number) since only a subset of the trellis states can be reached. In the third quantization step, the state transition choice is dichotomic, therefore allowing for a single bit/sample in the quantization of the corresponding vector component.

It is worth to note that from the implementation point of view, both the bit/sample increase and the decrease can be easily realized while carrying out the Viterbi algorithm in the encoding process. In particular, the addition of parallel transitions implies an increase in the evaluation of the local transition state metrics. On the contrary, pruning a state transition branch implies the assignment of a corresponding 'infinite' local transition state metric.

30 In recent years line spectrum pairs (LSP) representation of LPC parameters has become popular in speech coding applications. The LSP are frequency domain parameters strictly related to the formants: the position of a pair of frequencies gives the position of the formant, while their difference carries information about the width of the spectral peak.

The ordering property of the LSP parameters can be exploited by quantizing the differences between adjacent LSP frequencies instead of the absolute values of the LSP frequencies.

A proper bit allocation can be assigned to each LSP difference according to its perceptual importance.

40 When applied to the quantization of the LSP differences, the TCQ algorithm proves itself to be particularly effective, since the quantization error accumulated in quantizing the - say - first  $(i-1)$ -th LSP differences can be taken into account in the search of the optimum quantization level for the  $i$ -th LSP difference.

45 Each trellis state will be assigned a "history path", at each  $i$ -th trellis stage; each state transition belonging to this history path will correspond to a pointer to the quantization level of the corresponding LSP difference. By adding all the LSP differences of each state history path up to the  $i$ -th trellis stage, the  $i$ -th quantized LSP can be reconstructed (note that this reconstructed LSP will be different - in general - for each trellis state).

As an example, suppose that the  $i$ -th LSP difference must be quantized.

The following operations will be performed:

- 50 - For each  $j$ -th state of the  $(i-1)$ -th trellis stage, the corresponding  $(i-1)$ -th LSP is reconstructed, by adding all the quantized LSP differences belonging to the  $j$ -th state history path.
- For each  $j$ -th state, the LSP difference between the input  $i$ -th LSP and the reconstructed  $(i-1)$ -th LSP is computed.
- This difference is quantized in with a suitable metric, according to the  $i$ -th stage quantizer level partition 'seen' by each  $j$ -th state.
- 55 - The Viterbi algorithm is then acted upon on each trellis state, in order to determine the best previous state and, therefore, the updated history path. Furthermore, the quantization accumulated cost is updated for each state; in particular, this cost is consistent with the metric used for each LSP

difference quantization.

An example of this procedure is depicted in Figure 3, where a simple 4-state trellis with 4 quantization levels (for each quantization step) and 1 bit/branch is used.

Suppose that the  $n$ -th LSP is input. Hence, the corresponding LSP difference must be quantized.

In the trellis of Figure 3,

$$D_j^i$$

the quantity refers to the quantized  $i$ -th LSP difference, according to the corresponding quantization level belonging to the  $j$ -th state history path.

It is clear that by adding all the quantized LSP difference belonging to the generical state history path, the reconstructed  $n$ -th LSP may be obtained, as function of the state under consideration.

In particular, let:

$$LSP_0 = D_0^0 + D_0^1 \quad \text{be the reconstructed LSP along the state 0 path.}$$

$$LSP_1 = D_1^0 + D_1^1 \quad \text{be the reconstructed LSP along the state 1 path.}$$

$$LSP_2 = D_2^0 + D_2^1 \quad \text{be the reconstructed LSP along the state 2 path.}$$

$$LSP_3 = D_3^0 + D_3^1 \quad \text{be the reconstructed LSP along the state 3 path.}$$

For each  $j$ -th state the difference between the  $n$ -th input LSP and the reconstructed  $n$ -th LSP may be computed. Let this quantity be denoted as

$$D_j^2.$$

Afterward, the transition cost from each  $j$ -th state to each possible  $k$ -th future state is computed. This transition cost is related to the quantization level associated to the corresponding transition branch; with reference to Figure 3, the transition cost is denoted as  $C_{jk}$ . In particular,  $C_{jk}$  depends on the quantization error which is measured (according to a proper metric) as function of the "transitional" quantization level.

To be more specific, let  $L_{jk}$  the quantization level associated to the transition between the  $j$ -th state and the  $k$ -th one. We can write:

- Starting from state 0, compute:

$$C_{00} = m(D_0^2, L_{00}) \quad \text{transition cost associated to the state 0 - state 0 branch.}$$

$$C_{01} = m(D_0^2, L_{01}) \quad \text{transition cost associated to the state 0 - state 1 branch.}$$

- Starting from state 1, compute:

$C_{12} = m(D_1^2, L_{12})$  transition cost associated to the state 1 - state 2 branch.

5

$C_{13} = m(D_1^2, L_{13})$  transition cost associated to the state 1 - state 3 branch.

10

- Similarly, the procedure is repeated for state 2 and state 3.

After all the transition costs have been computed, the best state path up to the 3rd trellis stage must be updated (for each trellis state). The well-known Viterbi algorithm is used to this goal. As an example, referring to Figure 3, we have the following:

- 15 - Current 'observation' state: state 0  
 - Compute the candidate accumulated cost with respect to the previous state 0:  $A = 0(0) + C_{00}$ ,  $0(0)$  being the state 0 overall cost accumulated so far.  
 - Compute the candidate accumulated cost with respect to the previous state 2:  $A = 0(2) + C_{20}$ ,  $0(2)$  being the state 2 overall cost accumulated so far.  
 20 - If  $A < B$  the state 0 is considered to be the best previous state with respect to the current state 0; the new state 0 path is determined by concatenating the state 0 path with the state 0 - state 0 transition.  
 On the contrary, if  $A > B$ , the state 0 path is updated according to the state 2 path and to the state 2 - state 0 transition.  
 - The same procedure applies for the 'observation' states 2, 3, 4.

25 Finally, suppose that the following situation occurs:

- Best previous state with respect to state 0: **state0**
- Best previous state with respect to state 1: **state2**
- Best previous state with respect to state 2: **state3**
- Best previous state with respect to state 3: **state3**

30 The state paths are updated as depicted in Figure 4. Furthermore, the overall accumulated costs are updated for each trellis state.

We are ready for the next LSP quantization.

After the  $p$ -th LSP difference ( $p$  being the predictor order) has been quantized, the final state with the minimum accumulated cost is selected as the "winning" one. Its index (or, equivalently, the index of the corresponding initial state) is transmitted, together with the state transition labels of its history path.

35 At the decoder, the initial winning state index and the state transition labels of its history path are input. All the LSP differences can be recovered from the state transition label pointers to the quantization level table. Afterward, the LSP frequencies can be reconstructed.

Besides an intra-frame correlation of the LSP parameters (i.e. the ordering property), it is possible to take advantage of the strong inter-frame correlation; this may lead to efficient quantization schemes that can operate, for instance, in a two-dimensional differential fashion. A possible application is described in C.C. Kuo, F.R. Jean, H.C. Wang, "Low Bit-Rate Quantization of LSP Parameters Using Two-Dimensional Differential Coding", Proc. ICASSP '92, pagg. 97-100, where a two-dimensional differential coding scheme is shown to significantly improve the effectiveness of the quantization scheme as function of the desired encoding rate.

45 In particular, in C.C. Kuo, F.R. Jean, H.C. Wang, "Low Bit-Rate Quantization of LSP Parameters Using Two-Dimensional Differential Coding", Proc. ICASSP '92, pagg. 97-100, the inter-frame/intra-frame dependency of the  $i$ -th LSP at frame  $n$  may be expressed as:

$$50 \quad f_i(n) = a_i f_{i-1}(n) + b_i f_i(n-1), i = 1, 2, \dots, p \quad (1)$$

where  $p$  is the predictor order and  $a_i$  and  $b_i$  are the coefficients of an optimal two-dimensional (i.e. 2 - D) predictor; these coefficients can be estimated from a long sequence of speech, as described in C.C. Kuo, F.R. Jean, H.C. Wang, "Low Bit-Rate Quantization of LSP Parameters Using Two-Dimensional Differential Coding", Proc. ICASSP '92, pagg. 97-100,  $f_i(n)$  is the current LSP estimation;  $f_{i-1}(n)$  and  $f_i(n-1)$  are previously quantized parameters.

55 Therefore, it is possible to transmit the difference between the exact value and the estimated value of the current LSP.

It is clear that this quantization scheme may not be the optimal one in case channel errors occur during the transmission of the LSP difference information.

It is possible to cope with this problem by employing a careful design of the 2-D predictor coefficients. However; this issue will be described in greater details in a next paragraph. For the time being, we will assume to deal with proper 2-D predictor coefficients and will concentrate on the TCQ functioning in this case.

The working principle is analogous to the one described previously for the 1-D (i.e. intra-frame) case, which simply exploits the LSP ordering property. In particular, it is worth to note that the 1-D case can be considered as a particular case of the 2-D scheme, by putting  $a_i = 1$  and  $b_i = 0$ .

In particular, suppose that the  $i$ -th LSP 2-D difference must be quantized. The following operations will be performed:

For each  $j$ -th trellis state of the  $(i - 1)$ -th trellis stage, a corresponding (reconstructed)

$$LSP_j^{i-1}$$

( $n$ ) will be available ( $n$  is the current frame index).

- For each  $j$ -th state, the 2-D LSP difference is computed between the input  $i$ -th LSP and the weighted combination of:
- the reconstructed

$$LSP_j^{j-1}$$

- the corresponding  $i$ -th quantized LSP derived in the previous frame  $LSP^i(n-1)$ .

The combination weights are the 2-D predictor coefficients.

- The computed difference is then quantized according to the trellis topology and to the quantization level configuration. The quantization procedure is analogous to the one described in the 1-D case.
- Once all the LSP differences have been quantized and the state paths (as well as their accumulated costs) have been updated, the quantized

$$LSP_j^i$$

( $n$ ) can be derived, as function of each  $j$ -th state under consideration. In particular, we have:

$$LSP_j^i(n) = a_i LSP_{previous(j)}^{i-1}(n) + b_i LSP^i(n-1) + D_{LSP_j}^i(n), \quad (2)$$

where

$$D_{LSP_j}^i$$

is the best local quantized LSP difference appertaining to the  $j$ -th state and  $previous(j)$  is its previous state in the history path, as derived from the Viterbi algorithm.

Iterating this way, all the LSP can be quantized, according to the 2-D predictor behaviour.

More in general, a multi-coefficient 2-D predictor can be employed, both in the intra-frame and in the inter-frame sense, as follows:

$$\sum_{j=0}^J a_j f_{i-j}(n) + \sum_{k=0}^K b_k f_i(n-1) \quad (3)$$

It is worth to note that the predictor length is not necessarily the same in each dimension.

A further way to exploit the 'spatial-temporal' redundancy of the LSP parameters is to use a 3-D predictor as follows:

$$f_i(n) = a_i f_{i-1}(n) + b_i f_i(n-1) + c_i f_{i-1}(n-1) \quad (4)$$

that is, we introduce another inter-frame/intra-frame dependency, namely the one related to the previous (in the intra-frame sense) LSP of the previous (in the inter-frame sense) frame. The third weighting coefficient can be determined in an 'optimal' way, as will be described in a following section.

The concept can be extended further, by introducing a multi-coefficient multi-dimensional predictor, operating with different prediction orders, according to the prediction 'direction' (i.e. intra-frame, inter-frame, various intra-frame/inter-frame combinations).

Irrespectively of the predictor structure, a difference between a LSP and the corresponding estimated one will be quantized, following the trellis search procedure and the Viterbi algorithm described previously.

At the decoder site, all the LSP differences can be recovered from the best state information and the related history path. The LSP values can then be reconstructed by re-adding the previously (both in the intra-frame and in the inter-frame sense) reconstructed parameters, after weighting them by the corresponding predictor coefficients.

It is clear that the TCQ of the LSP parameters (in a generical differential sense) can be carried out according to any suitable metric that allows to measure an overall distortion as function of successive local distortions.

In particular, a simple mean squared error (MSE) could be used as the local metric for the quantization error. In this respect, the transition cost defined previously (i.e. see the 1-D case) could be defined as:

25

$$C_{jk} = (D_j^k - L_{jk})^2 \quad (5)$$

More in general, a weighted MSE could be employed, following (e.g.) the guidelines specified in K.K. Paliwal, B.S. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", Proc. ICASSP '91, pagg. 661-664, where the spectral content of the speech signal at the LSP frequency location is taken into account explicitly. Or, a WMSE criterion that considers the relative weight of the specific LSP that is being quantized could also be considered. In this case, formula (5) could be re-written as:

35

$$C_{jk} = f(D_j^k, L_{jk})(D_j^k - L_{jk})^2 \quad (6)$$

where  $f(.,.)$  is a (one-dimensional or two-dimensional) weighting function that would take into account the differential LSP to be quantized and/or the quantization level that is being considered.

Although LSP have proved to be an useful representation of the LPC coefficients with respect to quantization effectiveness, the reflection coefficients are also attractive, for some reasons like:

- Easy control of the filter stability.
- No need of complicated arithmetic procedures to convert them into LSP parameters.
- Possibility of implementing the necessary filter structures in lattice forms, with evident advantages for fix-point computation.

A recursive structure may be used for the computation of the reflection coefficients, starting either from the values of the autocorrelation function (and thereby using the well-known Leroux-Gueguen algorithm) or from the values of the signal covariance function (by employing the so-called covariance-lattice formulation, as explained in A. Cumani, "On a Covariance-Lattice Algorithm for Linear Prediction", Proc. ICASSP '82, pagg. 651-654).

In particular, the Leroux-Gueguen algorithm should be reformulated properly in order to take into account the eventual quantization of the reflection coefficients after their computation at each step of the recursion.

This gives rise to a slightly modified recursive algorithm in which, starting from the autocorrelation values, the reflection coefficients are computed as follows:

- Let  $f_j(n)$  be the forward residual of the lattice structure  $j$ -th stage and  $b_j(n)$  be the corresponding backward residual. Then, the expression of the  $j$ -th stage residuals as function of the  $(j - 1)$ -th stage ones is as follows:

$$f_j(n) = f_{j-1}(n) + K_j b_{j-1}(n - 1) \quad (7)$$

$$b_j(n) = b_{j-1}(n - 1) + K_j f_{j-1}(n), \quad (8)$$

$K_j$  being the  $j$ -th stage reflection coefficients.

- Defining the initial conditions:

$$f_0(n) = b_0(n) + s(n) \quad (9)$$

$s(n)$  being the lattice structure input signal.

- Defining also the following autocorrelation and cross-correlation functions:

$$R_F^j(k) = \sum_{n=0}^{N-1-k} f_j(n) f_j(n+k) \quad (10)$$

autocorrelation of the forward residual at the  $j$ -th lattice stage.

$$R_B^j(k) = \sum_{n=0}^{N-1-k} b_j(n) b_j(n+k) \quad (11)$$

autocorrelation of the backward residual at the  $j$ -th lattice stage.

$$R_{FB}^j(k) = \sum_{n=0}^{N-1-k} f_j(n) b_j(n+k-1), \quad (12)$$

cross-correlation between the forward and backward residuals at the  $j$ -th lattice stage.

- The forward residual autocorrelation at the  $j$ -th lattice stage may be expressed by means of the following recursive formula:

$$R_F^j(k) = R_F^{j-1}(k) + K_j R_{BF}^{j-1}(k+1) + K_j R_{FB}^{j-1}(k-1) + K_j^2 R_B^{j-1}(k) \quad (13)$$

- Therefore, the optimal value for the reflection coefficient  $K_j$  is the one that minimizes the forward residual energy  $R_F(0)$ .

- Once the  $K_j$  values is computed (and, eventually, quantized), the autocorrelation may be updated, as well as the quantities,  $R_B^j$ ,  $R_{FB}^j$ ,  $R_{BF}^j$  (using expressions similar to the one derived for  $R_F^j$ )

Both the covariance-lattice formulation and the modified autocorrelation one are particularly amenable to TCQ, in that each computed reflection coefficient can be quantized prior to the next recursion step.

Again, after defining the trellis topology and the quantization level number at each quantization step, each reflection coefficients can be computed as function of each particular state.

Its value can therefore take into account the quantization error accumulated along each branch of a generical state path.

Afterward, the computed reflection coefficient can be quantized according to the quantization level subset 'seen' by each particular trellis state.

In formulas, the recursive algorithm for reflection coefficient computation, with embedded TCQ may be stated as follows (only the formulation related to the covariance-lattice approach is given, since the corresponding formalism for the autocorrelation approach may be derived in an analogous way; also, note



that the formalism used resembles the one described in: A. Cumani, "On a Covariance-Lattice Algorithm for Linear Prediction", Proc. ICASSP '82, pagg. 651-654.

- Given a block of N signal samples:  $s(0), s(1), \dots, s(N-1)$ , compute the covariance  $\Phi_{ik}$  for  $i, k = 0, 1, \dots, p$  ( $p$  being the predictor order):

5

$$\Phi_{ik} = \sum_{n=p}^{N-1} s(n-1)s(n-k) \quad (14)$$

10

- Set up  $F_{ij}^0, B_{ij}^0, C_{ij}^0$  for  $i, j = 0, 1, \dots, p-1$  using formula (14) of the reference mentioned above. Also, set:  $m = 0$
- For each predictor stage  $m$ :
  - Compute the  $m$ -th reflection coefficient as function of the  $j$ -th TCQ state:

15 where  $C_{00}^{mj}, F_{00}^{mj}$  and  $B_{00}^{mj}$  are the 'forward-backward covariance

$$K_{m+1}^j = -2C_{00}^{mj} / (F_{00}^{mj} + B_{00}^{mj})$$

20

functions' that, once determined making use of formulas (12a,b,c) of A. Cumani, "On a Covariance-Lattice Algorithm for Linear Prediction", Proc. ICASSP '82, pagg. 651-654, take into account the quantization values of the previous reflection coefficients along the  $j$ -th TCQ state path.

- 25 - Quantize the reflection coefficient just computed according to the quantizer level partition appertaining to the  $j$ -th state. In particular, a non-linear transformation (e.g. log-area ratios) can be done prior to quantization. Each quantization level 'seen' by the  $j$ -th state will correspond to a particular state branch connecting the  $j$ -th state to a future  $i$ -th state (according to the trellis topology).
- The optimal local quantization level thus found is related to a local transition cost, between the  $j$ -th state and the  $i$ -th one.
- 30 - Proceed to the next trellis stage and update the accumulated cost of each state (making use of the accumulated cost of the previous trellis stage and of the local metrics just computed). Also, update the partial quantization path for each state. That is, for each  $i$ -th trellis state update the 'forward-backward covariance functions' making use of the function values in the previous  $j$ -th state and the quantization level of the reflection coefficient that corresponds to the  $j$ -th state --  $i$ -th state transition branch.
- 35 - Again, the state with the minimum overall accumulated cost is declared as 'winner'. Its value, together with the trellis labels defining its path, determines the quantized reflection coefficient vector.

40 As for the LSP case, a proper metric should be employed to carry out the quantization process; in particular, a metric that allows to measure an overall distortion as function of successive local distortions (such as a MSE- or WMSE-based metric) can be suitable.

The quantization procedures outlined in the previous paragraphs may not be the optimal ones (with respect to both the LSP and the reflection coefficients).

45 In particular, it has been stressed the importance of using suitable metrics, that allow the computation of the accumulated cost as sum of successive partial costs.

From the perceptual point of view, it is well known that the most reliable metric for measuring the effectiveness of the LPC parameter quantization is based on the cepstral coefficients (e.g. see K.K. Paliwal, B.S. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", Proc. ICASSP '91, pagg. 661-664). In particular, once two set of predictor coefficients (i.e. before and after quantization) are known, one should compute the corresponding cepstral coefficient sequences and then measure the MSE (namely, the cepstral distance CD).

50

However, this procedure is not feasible in carrying out the step-by-step quantization (i.e. as a new LPC parameter is available), as described in the previous TCQ procedures.

Therefore, in order to obtain the set of quantized coefficients that guarantee the best perceptual LPC reproduction, the following steps should be carried out:

55

- Reconstruct the decoding path in correspondence of each trellis state, therefore obtaining a set of quantized LPC parameter vectors (i.e. either in terms of LSP or in terms of reflection coefficients).

- For each vector, obtain the corresponding representation in terms of LPC cepstral coefficients and measure the CD with respect to the cepstral coefficient representation of the unquantized model.
- The trellis parameters to be transmitted should be the ones that define the best LPC vector (in terms of cepstral distance).

5 Obtaining the cepstral coefficients from a set of LPC parameters (i.e. LSP or reflection coefficients) is not a trivial task. Therefore, the outlined procedure is likely to be very time-consuming. It is possible to reduce the computation load by reconstructing the decoding path in correspondence of only a subset of the overall trellis states.

10 This implies that an implicit assumption is made: namely, the trellis state subset with lowest overall accumulated cost is likely to contain the best trellis state, in terms of CD.

For better quantization efficiency, the quantization levels are different for each trellis state; the following example clarifies this concept (assume that we are dealing with the quantization of the LSP parameters).

The same rationales apply for the quantization of the reflection coefficients as well).

15 Suppose that the  $i$ -th LSP difference must be quantized; its value is computed by taking the difference between the  $i$ -th LSF and the reconstructed (i.e. quantized)  $(i - 1)$ -th LSF. This reconstructed  $(i - 1)$ -th LSF is different for each trellis state; the  $i$ -th LSF difference thus obtained must be quantized accordingly to the level partition "seen" by the corresponding trellis state.

20 Assume that two generical trellis states point to the same subset of quantization level; in standard TCQ procedures (e.g. see M.W. Marcellin, T.G. Fisher, "Trellis Coded Quantization of Memoryless and Gauss-Markov Sources", IEEE Trans. on Communication, vol. 38, No. 1, January 1990) the subset quantization values are the same for the two states; they are only addressed in a different way.

On the contrary we use different values in the same quantization level subset, as function of TCQ state under consideration.

25 In order to obtain this, a proper TCQ training procedure can be adopted; in particular we start from a unique set of quantization values for each state subset; these values can be found by using a standard scalar quantization clustering procedure.

Afterwards an iterative procedure is adopted in which a long training sequence of LSP is input to the TCQ and the input LSP vector is then assigned to the "partition" corresponding to the obtained TCQ path.

30 At the end of the training procedure each possible quantization path is assigned a partition; the corresponding "cluster vector" can be derived by simply taking a proper mean of each partition value and assigning this mean value to the corresponding path state.

Next, the LSF value training sequence is again input to the TCQ; a new partition set can be generated and the corresponding set of cluster vectors can be found.

More in details, during a generical iteration step, the following operations can be performed:

- 35
- Reset all the partitions belonging to each trellis state. Note that the number of partitions appertaining to each trellis state is equal to the number of quantization levels that can be 'reached' from the state.
  - For each LSP input vector:
    - Find the optimal quantized vector according to a predefined metric. The quantized vector is identified by specifying the winning starting state and the branch labels along the state path.
  - Assign each  $j$ -th element of the input vector to a partition whose index corresponds to the branch label of the  $j$ -th quantization step along the winning state path. In particular, if the simple MSE is adopted in the quantization phase, the  $j$ -th input vector element is simply added to the previous partition value.
- 40
- At this point, all the partitions for each trellis state and for each quantization step have been constructed. Each partition 'centroid' can be recomputed by taking an appropriate mean as function of the accumulated partition value and of the number of elements inside the partition. In particular, if the MSE metric is adopted in the quantization phase, each centroid can be computed by taking the simple arithmetic mean of the partition accumulated value.
- 45

50 Iterating this way it can be observed that the quantization error (in a MSE sense, or in a WMSE sense, according to the metric adopted) is decreasing; although this may not correspond to a performance increase in the cepstral distance sense, it is possible to run the iterative procedure for a fixed iteration number and then choose the quantization level set that guarantees the best performance in terms of cepstral distance.

55 Finally, it is worth to note that this iterative reoptimization of the quantization levels is independent on the trellis topology as well as on the LPC parameters under consideration (i.e. LSP or reflection coefficients). However, care must be taken in the determination of the partition centroid, according to the metric used in the quantization phase.

Trellis Coded Vector Quantization (TCVQ) is a generalization of the TCQ concept. It has been introduced in T.G. Fisher, M.W. Marcellin, M.Wang, "Trellis Coded Vector Quantization", IEEE Trans. on Information Theory, Vol. IT-37, Nov. 1991. and, again, consists of using a structured codebook with an expanded set of quantization levels.

5 In particular, instead of dealing with scalar quantization levels, we have an expanded set of reproduction vectors. Again, the trellis structure prunes the expanded number of quantization reproduction vectors down to the desired encoding rate.

When applied to the LPC parameter quantization, the same strategies can be employed, whether we use the representation in terms of LSP or in terms of reflection coefficients.

10 It is clear that, for typical predictor orders (i.e. 10), it is not worth to use high-dimension vectors, in order to maintain a favourable trade-off between performance and encoding rate.

To be more specific, let's consider the following example:

- Predictor order = 10 (i.e. 10 LPC coefficients to be quantized)
- Scalar quantization versus TCQ case
  - 15 - Using 3 bits/coefficient (8 quantization levels), scalar quantization implies quantizing the LPC information with 30 bits.
  - Using 2 bits/coefficients (8 quantization levels) and a 16-state trellis (which is a good compromise between performance and computation load), TCQ implies quantizing the LPC information with  $4 + 20 = 24$  bits
- 20 - Vector quantization versus TCVQ case
  - Dividing the LPC vector in two subvectors of 5 coefficients each and quantizing each subvector with a  $2_{15}$ -element codebook (in order to maintain the same encoding rate as for the scalar quantization case), and using, again, a 16-state trellis, the TCVQ approach allows for an overall encoding rate of  $4 + 14 + 14 = 32$  bits for the LPC information (note that by using simple VQ, we
  - 25 - Dividing the LPC vector in 5 subvectors of 2 coefficients each and quantizing each subvector with a  $2_6$ -element codebook (thus obtaining again an encoding rate of 30 bits for the VQ case), the TCVQ approach with a 16-state trellis would allow to obtain  $4 + 5 \cdot 5 = 29$  bits.

30 Actually, the TCVQ technique acted upon subvectors of coefficient couples seems a good compromise between encoding rate and performance.

The TCVQ procedure is carried out in exactly the same way as for the TCQ counterpart, both in the 1-D case (i.e., taking into account only the intra-frame dependency of the LSP parameters), and in the case of multi-dimensional prediction (i.e., exploiting both the intra-frame and the inter-frame dependency of LSP parameters, with any prediction length in either direction).

35 Besides, when considering the reflection coefficient case (where the prediction-based solution may not be the optimal one), the TCVQ procedure can be carried out by recursive quantization of reflection coefficient couples (if the subvector dimension is actually 2), by using the same strategy employed for the TCQ case.

40 To be more specific, a brief description of the TCVQ procedure, when applied to the LSP in the 1-D case, is as follows (assuming to deal with subvectors of coefficient couples. Also, suppose we want to quantize the successive LSP differences): As an example, suppose that the  $i$ -th LSP and the  $(i + 1)$ -th one must be quantized.

The following operations will be performed:

- 45 - For each  $j$ -th state of the  $(i - 1)$ -th trellis stage, the corresponding  $(i - 1)$ -th LSP is reconstructed, by adding all the quantized LSP difference couples belonging to the  $j$ -th state history path.
- For each  $j$ -th state, the LSP difference between the input  $i$ -th LSP and the reconstructed  $(i - 1)$ -th LSP is computed. Furthermore, the LSP difference between the two input LSP parameters can be computed. This gives rise to a 2-component LSP difference vector to be quantized.
- This difference vector is quantized with a suitable metric, according to the  $i$ -th stage quantizer level partition 'seen' by each  $j$ -th state.
- 50 - The Viterbi algorithm is then acted upon on each trellis state, in order to determine the best previous state and, therefore, the updated history path. Furthermore, the quantization accumulated cost is updated for each state; in particular, this cost is consistent with the metric use for each LSP difference quantization.

55 Note that the TCVQ generalization for the LSP multi-dimensional predictor case and for the reflection coefficients case can be derived in a straightforward manner following the corresponding TCQ descriptions.

Finally, also the trellis level reoptimization procedure can be carried out in an analogous way as for the TCQ case. In particular, the vector clusters can be constructed in an iterative way, as function of the

different trellis states and of the corresponding encoding paths. These clusters are obtained as 'centroid' (according to a predetermined metric) of corresponding partitions of the input vector set.

# Claims

- 5 1. Method for speech coding comprising the steps of: receiving in input a sequence of LPC filter coefficients, quantizing said LPC filter coefficients, characterized by said quantization is made using TCQ technique.
- 10 2. Method according to claim 1 characterized by a variable bit allocation at each quantization step.
3. Method according to claim 2 characterized by the fact that a bit rate increase is obtained by adding one or more parallel transitions to the state branches, given a certain trellis topology.
- 15 4. Method according to claim 2 characterized by the fact that a bit rate decrease is obtained by deleting one or more state branches, given a certain trellis topology.
5. Method according to claim 1 characterized by the fact that of each quantization step the quantization error accumulated in quantizing the previous steps can be monitored and, eventually compensated.
- 20 6. Method according to claim 1 characterized by said LPC filter coefficients are the LSP parameters.
7. Method according to claims 5 and 6 characterized by the fact that at each quantization step each trellis state is assigned a history path, each path branch correspond to a pointer to the quantization level of the corresponding LSP value.
- 25 8. Method according to claims 7 characterized by the fact the history path contains the information associated to each quantization step.
- 30 9. Method according to claim 6 characterized by the fact that intraframe correlation can be exploited in quantizing the LSP parameters, by means of one dimensional differential prediction schemes along the frequency direction.
- 35 10. Method according to claim 6 characterized by the fact that interframe correlation can be exploited in quantizing the LSP parameters, by means of one dimensional differential prediction schemes along the time direction.
- 40 11. Method according to claim 6 characterized by the fact that both interframe correlation and intraframe correlation can be exploited in quantizing the LSP parameters, by means of multi dimensional differential prediction schemes.
12. Method according to claim 1 characterized by said LPC filter coefficients are the RC parameters.
- 45 13. Method according to claims 12 and 5 characterized by comprising the steps of defining trellis topology, defining quantization level number at each quantization step, and computing each RC as function of each particular state.
14. Method according to claim 1 characterized by the fact that the quantization error is computed using a metric that has the property of being additive at each quantization step.
- 50 15. Method according to claim 1 and 14 characterized by comprising the steps of reconstructing the encoding path in correspondence of each trellis state, obtaining a set of quantized LPC parameter vectors, obtaining for each vector, a corresponding representation in terms of LPC cepstral coefficients, measuring the cepstral distance with respect to the cepstral coefficient representation of the unquantized model, choosing the trellis parameters that define the LPC vector with optimal cepstral distance.
- 55 16. Method according to claim 1 characterized by adopting a proper TCQ training procedure for a quantization level reoptimization.

17. Method according to claim 16 characterized by comprising the step of starting from a set of quantization values for each subset, adopting a iterative procedure with a long training sequence of LSP as input, assigning to each input LSP vector a partition corresponding to the obtained TCQ path, taking a proper mean of each partition value assigning said mean to the corresponding path state branch.

5

18. Method according to claim 1 characterized by the fact that a TCVQ technique is used instead of a TCQ one.

19. Speech coder based on LPC techniques comprising means for quantizing said LPC information characterized by the use of TCQ technique.

10

15

20

25

30

35

40

45

50

55

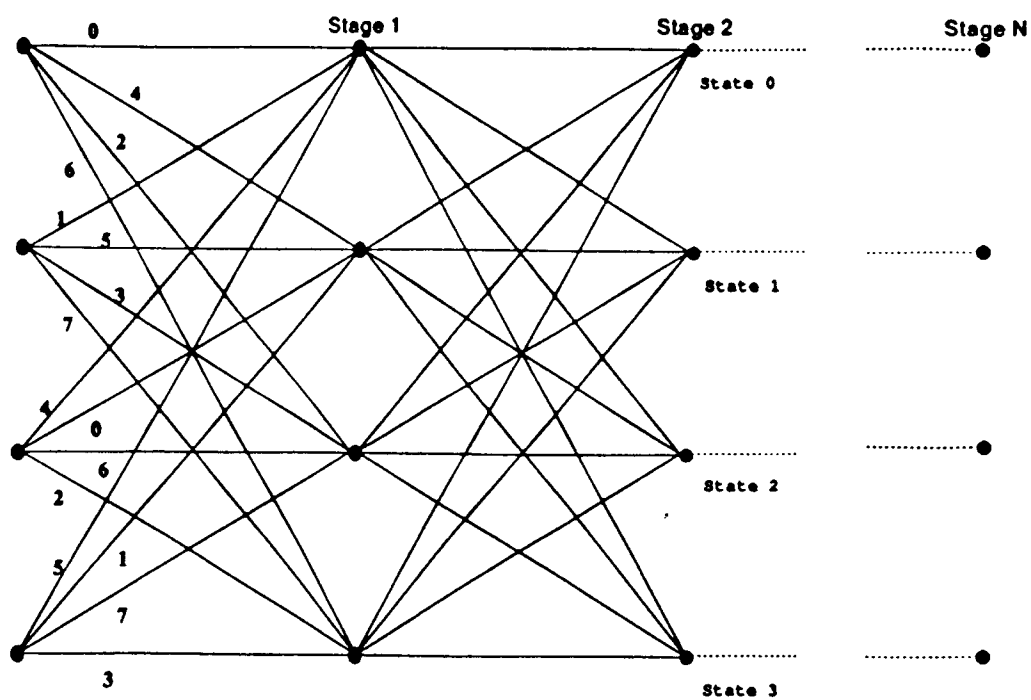


Figure 1

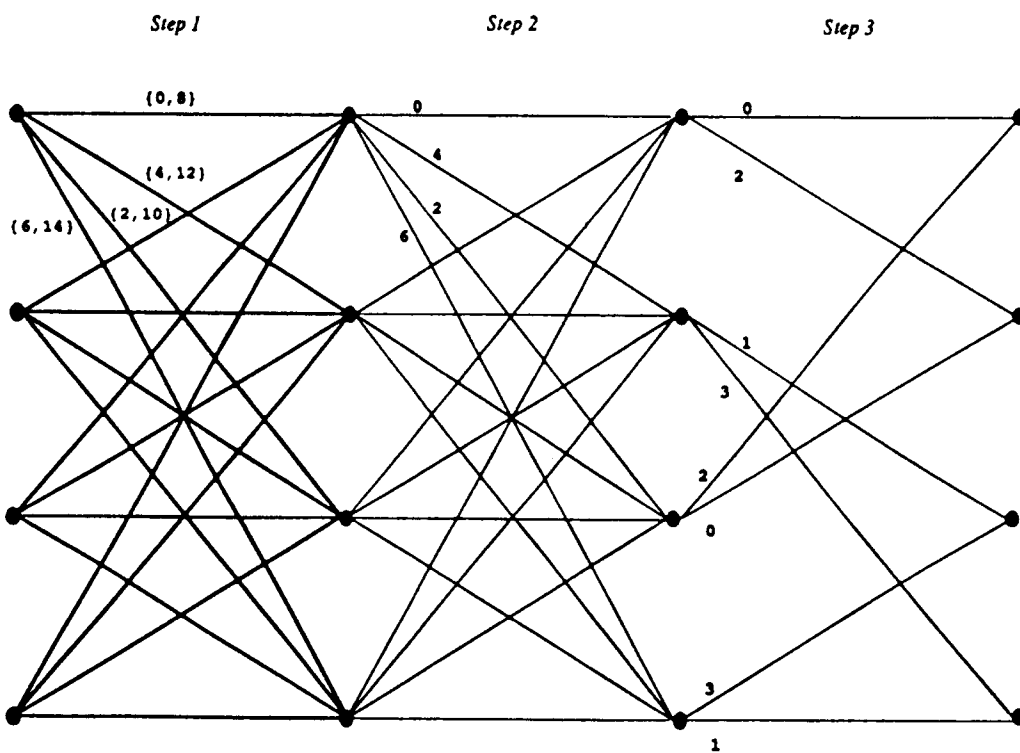


Figure 2

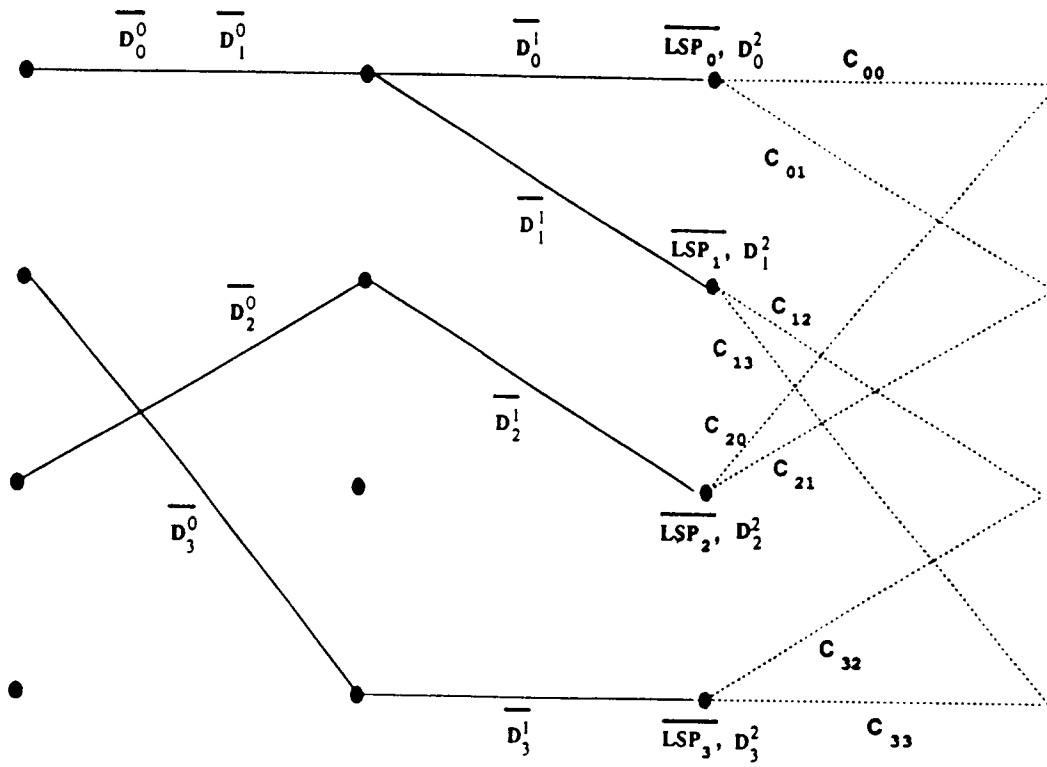


Figure 3

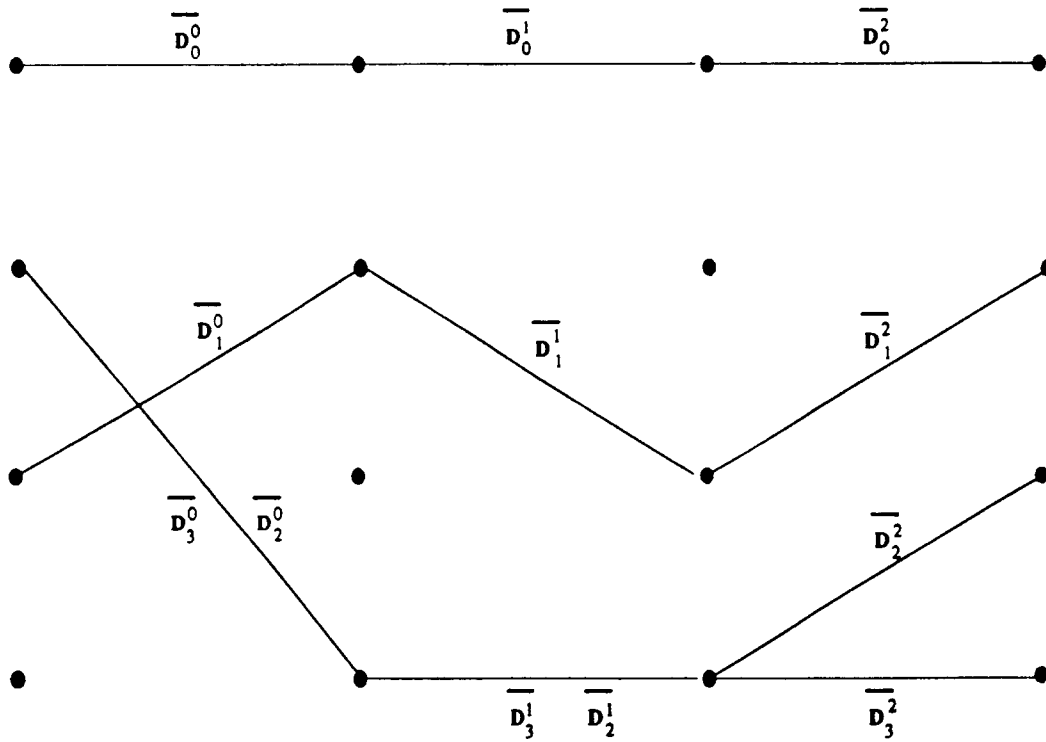


Figure 4