



**A METHOD AND DEVICE FOR ROBUST PREDICTIVE VECTOR  
QUANTIZATION OF LINEAR PREDICTION PARAMETERS IN VARIABLE  
BIT RATE SPEECH CODING**

5           **Owner/Applicant**

VoiceAge Corporation  
750 Chemin Lucerne, Suite 250  
Ville Mont-Royal, (Québec), H3R 2H6  
10          Canada

**Inventor (with contact information)**

*Milan Jelinek*

15          925, Walton,  
Sherbrooke, (Québec), J1H 1K4  
Canada

**BACKGROUND OF THE INVENTION**

**1. Field of the Invention**

20           The present invention relates to an improved technique for digitally encoding  
a sound signal, in particular but not exclusively a speech signal, in view of  
transmitting and synthesizing this sound signal. In particular, the present invention  
relates to the design of a vector quantization method of the linear prediction filter  
parameters in variable bit-rate linear prediction based speech coding.

## 2. Brief Description of the Prior Techniques

### 2.1 Speech coding and linear prediction (LP) parameters quantization

Digital voice communication systems such as wireless systems use speech coders to increase capacity while maintaining high voice quality. A speech *encoder* converts a speech signal into a digital bitstream which is transmitted over a communication channel or stored in a storage medium. The speech signal is digitized, that is, sampled and quantized with usually 16-bits per sample. The speech encoder has the role of representing these digital samples with a smaller number of bits while maintaining a good subjective speech quality. The speech *decoder* or synthesizer operates on the transmitted or stored bit stream and converts it back to a sound signal.

Digital speech coding methods based on linear prediction analysis have been very successful in low bit rate speech coding. In particular, *Code-Excited Linear Prediction* (CELP) coding is one of the best prior techniques for achieving a good compromise between the subjective quality and bit rate. This coding technique is a basis of several speech coding standards both in wireless and wireline applications. In CELP coding, the sampled speech signal is processed in successive blocks of  $N$  samples usually called *frames*, where  $N$  is a predetermined number corresponding typically to 10–30 ms. A linear prediction (LP) filter is computed, encoded, and transmitted every frame. The computation of the LP filter typically needs a *lookahead*, which consists of a 5–15 ms speech segment from the subsequent frame. The  $N$ -sample frame is divided into smaller blocks called *subframes*. Usually the number of subframes is three or four resulting in 4–10 ms subframes. In each subframe, an excitation signal is usually obtained from two components, the past excitation and the innovative, fixed-codebook excitation. The component formed from the past excitation is often referred to as the adaptive codebook or pitch excitation. The parameters characterizing the excitation signal are coded and transmitted to the decoder, where the reconstructed excitation signal is used as the input of the LP filter.

The linear prediction (LP) synthesis filter is given by

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{i=1}^M a_i z^{-i}}$$

where  $a_i$  are the linear prediction coefficients and  $M$  is the order of LP analysis. The LP synthesis filter models the spectral envelope of the speech signal. At the decoder,  
5 the speech signal is reconstructed by filtering the decoded excitation through the LP synthesis filter.

The set of *linear prediction coefficients*,  $a_i$ , are computed such that the prediction error

$$e(n) = s(n) - \tilde{s}(n) \tag{1}$$

10 is minimized, where  $s(n)$  is the input signal at time  $n$  and  $\tilde{s}(n)$  is the predicted signal based on the last  $M$  samples given by

$$\tilde{s}(n) = -\sum_{i=1}^M a_i s(n-i)$$

Thus the prediction error is given by

$$e(n) = s(n) + \sum_{i=1}^M a_i s(n-i)$$

15 This corresponds in the z-transform domain to

$$E(z) = S(z)A(z)$$

where  $A(z)$  is the linear prediction filter of order  $M$  given by

$$A(z) = 1 + \sum_{i=1}^M a_i z^{-i}$$

Typically, the LP coefficients  $a_i$  are computed by minimizing the mean-squared prediction error over a block of  $L$  samples. The computation of LP parameters is well known to people skilled in the art. An example computation is given in [1].

5           The prediction coefficients  $a_i$  can not be directly quantized for transmission to the decoder. The reason is that small quantization errors on the prediction coefficients can produce large spectral errors in the transfer function of the prediction filter, and can even cause filter instabilities. Hence, a transformation is applied to the prediction coefficients prior to quantization. The transformation yields what is called a  
10 *representation* of the prediction coefficients. After receiving the quantized, transformed prediction coefficients, the decoder can then apply the inverse transformation to obtain the quantized prediction coefficients. One widely used representation for the linear prediction coefficients is the Line Spectral Frequencies (LSF) also known as Line Spectrum Pairs (LSP). Details of the computation of the  
15 LSFs can be found in [2].

A similar representation is the Immitance Spectral Frequencies (ISF), which has been used in the AMR-WB coding standard [1]. Other representations are also possible and have been used. Without loss of generality, we consider in this invention the case of ISF representation.

20           The LP parameters are quantized either with scalar quantization (SQ) or vector quantization (VQ). In scalar quantization, the parameters are quantized individually and usually 3 or 4 bits per parameter are needed. In vector quantization, the parameters are grouped in a vector and quantized as an entity. A codebook, or a table, containing the set quantized vectors is stored. The quantizer searches the codebook  
25 for the codebook entry that is closest to the input vector according to a certain distance measure. The index of the selected quantized vector is transmitted to the

decoder. Vector quantization gives better performance than scalar quantization but at the expense of increased complexity and memory requirements.

Structured vector quantization is usually used to reduce the complexity and storage requirements of VQ. In split-VQ, the LP parameter vector is split into 2 or  
5 more subvectors which are quantized individually. In multistage VQ the quantized vector is the addition of entries from several codebooks. Both split VQ and multistage VQ result in reduced memory and complexity while maintaining good quantization performance. Further, an interesting approach is to combine multistage and split VQ to further reduce the complexity and memory. In reference [2] the LP vector is  
10 quantized in two stages where the second stage vector is split in two subvectors.

The LP parameters exhibit strong correlation between successive frames and this is usually exploited by the use of predictive quantization to improve the performance. In predictive vector quantization, a predicted LP vector is computed based on information from past frames. Then the predicted vector is removed from the  
15 input vector and the prediction error is vector quantized. Two kinds of prediction are usually used: auto-regressive (AR) prediction and moving average (MA) prediction. In AR prediction the predicted vector is computed as a combination of quantized vectors from past frames. In MA prediction, the predicted vector is computed as a combination of the prediction error vectors from past frames. AR prediction yields  
20 better performance, however, it is not robust to frame loss conditions which is encountered in wireless and packet-based communication systems. In case of lost frames, the error will propagate to consecutive frames since the prediction is based on previous corrupted frames.

## 2.2 Variable bit-rate (VBR) coding

25 In several communications systems, for example wireless systems using code division multiple access (CDMA) technology, the use of source-controlled variable bit rate (VBR) speech coding significantly improves the system capacity. In source-controlled VBR coding, the encoder operates at several bit rates, and a rate selection

module is used to determine the bit rate used for encoding each speech frame based on the nature of the speech frame (e.g. voiced, unvoiced, transient, background noise). The goal is to attain the best speech quality at a given average bit rate, also referred to as average data rate (ADR). The encoder can operate at different modes by tuning the rate selection module to attain different ADRs at the different modes where the encoder performance is improved at increased ADRs. This enables the encoder with a mechanism of trade-off between speech quality and system capacity. In CDMA systems (e.g. CDMA-one and CDMA2000), typically 4 bit rates are used and they are referred to as full-rate (FR), half-rate (HR), quarter-rate (QR), and eighth-rate (ER). In this system two rate sets are supported referred to as Rate Set I and Rate Set II. In Rate Set II, a variable-rate encoder with rate selection mechanism operates at source-coding bit rates to 13.3 (FR), 6.2 (HR), 2.7 (QR), and 1.0 (ER) kbit/s, corresponding to gross bit rates of 14.4, 7.2, 3.6, and 1.8 kbit/s (with some bits added for error detection).

A wideband codec known as adaptive multi-rate wideband (AMR-WB) speech codec was recently selected by the ITU-T (International Telecommunications Union – Telecommunication Standardization Sector) for several wideband speech telephony and services and by 3GPP (third generation partnership project) for GSM and W-CDMA third generation wireless systems. AMR-WB codec consists of nine bit rates in the range from 6.6 to 23.85 kbit/s. Designing an AMR-WB-based source controlled VBR codec for CDMA2000 system has the advantage of enabling the interoperation between CDMA2000 and other systems using the AMR-WB codec. The AMR-WB bit rate of 12.65 kbit/s is the closest rate that can fit in the 13.3 kbit/s full-rate of Rate Set II. This rate can be used as the common rate between a CDMA2000 wideband VBR codec and AMR-WB which will enable the interoperability without the need for transcoding (which degrades the speech quality). A half-rate at 6.2 kbit/s has to be added to enable the efficient operation in the Rate Set II framework. The codec then can operate in few CDMA2000-specific modes but it will have a mode that enables interoperability with systems using the AMR-WB codec.

Half-rate encoding is typically chosen in frames where the input speech signal is stationary. The bit savings (compared to the full rate) are achieved by updating encoder parameters less frequently or by using fewer bits to encode some parameters. Specifically, in stationary voiced segments, the pitch information is  
5 encoded only once in a frame, and fewer bits are used for the fixed codebook and the LP coefficients.

Since predictive VQ with MA prediction is typically applied to encode the LP coefficients, there is an unnecessary increase in quantization noise in the LP coefficients. MA prediction, as opposed to AR prediction, is used to increase the  
10 robustness to frame losses; however, in stationary frames the LP coefficients evolve slowly so using AR prediction in this case would have a smaller impact on error propagation in the case of lost frames. This can be seen by observing that, in the case of missing frames, most decoders apply a concealment procedure which essentially extrapolates the coefficients of the last frame. If the missing frame is stationary  
15 voiced, this extrapolation gives very similar values to the actual transmitted (but not received) LP parameters. The reconstructed LP vector is thus close to what would have been decoded if the frame had not been lost. In that specific case, using AR prediction in the quantization procedure of the LP coefficients can not have a very adverse effect on quantization error propagation.

20

## OBJECTIVE OF THE INVENTION

An objective of the present invention is therefore to provide a novel technique to improve a speech coder's LP quantizer efficiency while maintaining the robustness to channel errors in variable bit rate speech coding by switching between MA and AR prediction depending on the nature of the speech frames.

25

The above and other objects, advantages and features of the present invention will become more apparent upon reading of the following non restrictive description

of an illustrative embodiment thereof, given by way of example only with reference to the accompanying drawings.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

In the appended drawings:

5           Figure 1 is a schematic drawing showing the principle of multi-stage vector quantization;

Figure 2 is a schematic drawing showing the principle of split-vector vector quantization;

10           Figure 3 is a schematic drawing showing the principle of autoregressive predictive vector quantization;

Figure 4 is a schematic diagram showing the principle of predictive vector quantization using Moving Average (MA) prediction;

15           Figure 5 is a schematic block diagram showing the basic steps of disclosed switched predictive vector quantization at the encoder, according to an illustrative embodiment of present invention;

Figure 6 is a schematic block diagram showing the basic steps of disclosed switched predictive vector quantization at the decoder, according to an illustrative embodiment of present invention;

20           Figure 7 is an illustrative drawing showing how the ISFs are distributed over the frequency range – each distribution is the probability function of an ISF at a given position in the ISF vector; and

Figure 8 is a graph showing typical evolution of ISF coefficients through successive speech frames.

### DETAILED DESCRIPTION OF THE ILLUSTRATIVE EMBODIMENT

Most recent speech coding techniques are based on linear prediction analysis  
5 such as CELP coding. The linear prediction (LP) parameters are computed and  
quantized in frames of 10-30 ms. In this illustrative embodiment 20 ms frames are  
used and a 16 order LP analysis is assumed. An illustrative example of the  
computation of the LP parameters in a speech coding system is found in reference [1].  
In this illustrative example, the preprocessed speech signals is windowed and the  
10 autocorrelations of the windowed speech are computed. The Levinson-Durbin  
recursion is then used to computed the prediction coefficients  $a_i$ ,  $i=1,\dots,M$  from the  
autocorrelations  $R(k)$ ,  $k=0,\dots,M$ , where  $M$  is the predictor order.

The prediction coefficients  $a_i$  cannot be directly quantized for transmission to  
the decoder. The reason is that small quantization errors on the prediction coefficients  
15 can produce large spectral errors in the transfer function of the prediction filter, and  
can even cause filter instabilities. Hence, a transformation is applied to the prediction  
coefficients prior to quantization. The transformation yields what is called a  
*representation* of the prediction coefficients. After receiving the quantized,  
transformed prediction coefficients, the decoder can then apply the inverse  
20 transformation to obtain the quantized prediction coefficients. One widely used  
representation for the linear prediction coefficients is the Line Spectral Frequencies  
(LSF) also known as Line Spectrum Pairs (LSP). Details of the computation of the  
LSFs can be found in reference [2]. The LSFs consists of the poles of the polynomials

$$P(z) = \left( A(z) + z^{-(M+1)} A(z^{-1}) \right) / (1 + z^{-1})$$

25 and

$$Q(z) = (A(z) - z^{-(M+1)} A(z^{-1})) / (1 - z^{-1})$$

For even values of  $M$ , each polynomial has  $M/2$  conjugate roots on the unit circle  $(e^{\pm j\omega_i})$ , therefore, the polynomials can be written as

$$P(z) = \prod_{i=1,3,\dots,M-1} (1 - 2q_i z^{-1} + z^{-2})$$

5 and

$$Q(z) = \prod_{i=2,4,\dots,M} (1 - 2q_i z^{-1} + z^{-2}),$$

where  $q_i = \cos(\omega_i)$  with  $\omega_i$  being the line spectral frequencies (LSF) and they satisfy the ordering property  $0 < \omega_1 < \omega_2 < \dots < \omega_M < \pi$ .

A similar representation is the Immitance Spectral Pairs (ISP) or the  
10 Immitance Spectral Frequencies (ISF), which has been used in the AMR-WB coding standard. Details of the ISF computation can be found in reference [1]. Other representations are also possible and have been used. Without loss of generality, we consider in this invention the case of ISF representation as an illustrative example.

For an  $M$ th order LP filter, where  $M$  is even, the ISPs are defined as the roots  
15 of the polynomials

$$F_1(z) = A(z) + z^{-M} A(z^{-1})$$

and

$$F_2(z) = (A(z) - z^{-M} A(z^{-1})) / (1 - z^{-2})$$

Polynomials  $F_1(z)$  and  $F_2(z)$  have  $M/2$  and  $M/2-1$  conjugate roots on the unit  
20 circle  $(e^{\pm j\omega_i})$ , respectively. Therefore, the polynomials can be written as

$$F_1(z) = (1 + a_M) \prod_{i=1,3,\dots,M-1} (1 - 2q_i z^{-1} + z^{-2})$$

and

$$F_2(z) = (1 - a_M) \prod_{i=2,4,\dots,M-2} (1 - 2q_i z^{-1} + z^{-2})$$

where  $q_i = \cos(\omega_i)$  with  $\omega_i$  being the immittance spectral frequencies (ISF) and  $a_M$  is the last prediction coefficient. The ISFs satisfy the ordering property  
 5  $0 < \omega_1 < \omega_2 < \dots < \omega_{M-1} < \pi$ . Thus the ISF parameters consist of  $M-1$  frequencies in addition to the last prediction coefficients. In this illustrative embodiment the ISFs are mapped into frequencies in the range 0 to  $f_s/2$ , where  $f_s$  is the sampling frequency, using the following relation

$$f_i = \frac{f_s}{2\pi} \arccos(q_i), \quad i = 1, \dots, M-1,$$

10 and

$$f_M = \frac{f_s}{4\pi} \arccos(a_M)$$

LSFs and ISFs have been widely used due to several properties which make them suitable for quantization purposes. Among these properties are the well defined dynamic range, their smooth evolution resulting in strong inter and intra-frame  
 15 correlations, and the existence of the ordering property which guarantees the stability of the quantized LP filter.

We will describe here the main properties of ISFs to understand the quantization approaches used. Figure 7 shows a typical example of the probability distribution function (PDF) of ISF coefficients. Each curve represents the PDF of an  
 20 individual ISF coefficient. The mean of each distribution is shown on the horizontal axis ( $\mu_k$ ). For example, the curve for ISF<sub>1</sub> indicates all values, with their probability of occurring, that can be taken by the first ISF coefficient in a frame. The curve for ISF<sub>2</sub> indicates all values, with their probability of occurring, that can be taken by the second ISF coefficient in a frame, and so on. The PDF function is typically obtained  
 25 by applying a histogram to the values taken by a given coefficient as observed

through several consecutive frames. We see that each ISF coefficient occupies a restricted interval over all possible ISF values. This effectively reduces the space that the quantizer has to cover and increases the bit-rate efficiency. It is also important to note that, while the PDFs of ISF coefficients can overlap, ISF coefficients in a given  
5 frame are always ordered ( $\text{ISF}_{k+1} - \text{ISF}_k > 0$ , where  $k$  is the position of the ISF coefficient within the vector of ISF coefficients).

With frame lengths of 10 to 30 ms typical in a speech coder, ISF coefficients exhibit interframe correlation. Figure 8 illustrates how ISF coefficients evolve across frames in a speech signal. Figure 8 was obtained by performing LP analysis over 30  
10 consecutive frames of 20 ms in a speech segment comprising both voiced and unvoiced frames. The LP coefficients (16 per frame) were transformed into ISF coefficients. We see that the lines never cross, which means that ISFs are always ordered. We also see that ISF coefficients typically evolve slowly, compared to the frame rate. This means in practice that predictive quantization can be applied to  
15 reduce the quantization error.

Figure 3 shows the principle of autoregressive (AR) predictive quantization. As per this Figure, a prediction error vector  $e_n$  is first obtained by subtracting (Processor 301) a prediction vector  $p_n$  from the input parameter vector to be quantized  $x_n$ . The symbol  $n$  here refers to the frame index in time. The prediction  $p_n$  is computed  
20 by a predictor P (Processor 302) using the past quantized vectors  $\hat{x}_{n-1}$ ,  $\hat{x}_{n-2}$ , etc. The prediction error vector is then quantized (Processor 303) to produce an index  $i$  (for transmission) and a quantized prediction error  $\hat{e}_n$ . The total quantized vector  $\hat{x}_n$  is obtained by adding (Processor 304) the quantized prediction error vector and the prediction vector  $p_n$ . A general form of the predictor P in Processor 302 is

$$25 \quad p_n = A_1 \hat{x}_{n-1} + A_2 \hat{x}_{n-2} + \dots + A_K \hat{x}_{n-K}$$

where  $A_k$  are prediction matrices of dimension  $M \times M$  and  $K$  is the predictor order. A simple form for the predictor P in Processor 302 is the use of first order prediction

$$\mathbf{p}_n = \mathbf{A}\hat{\mathbf{x}}_{n-1} \quad (2)$$

where  $\mathbf{A}$  is a prediction matrix of dimension  $M \times M$ , where  $M$  is the dimension of LP parameter vector. A simple form of the prediction matrix is a diagonal matrix with diagonal elements  $\alpha_1, \alpha_2, \dots, \alpha_M$ , where  $\alpha_i$  are prediction factors for individual LP parameters. If the same factor  $\alpha$  is used for all LP parameters then equation 2 reduces to

$$\mathbf{p}_n = \alpha \hat{\mathbf{x}}_{n-1} \quad (3)$$

Using the simple prediction form of Equation (3), then in Figure 3, the quantized vector  $\hat{\mathbf{x}}_n$  is given by the following autoregressive (AR) relation

$$\hat{\mathbf{x}}_n = \hat{\mathbf{e}}_n + \alpha \hat{\mathbf{x}}_{n-1} \quad (4)$$

The recursive form of Equation (4) implies that, when using an AR predictive quantizer of the form of Figure 3, channel errors will propagate across several frames. This can be seen more clearly if we write Equation (4) in the following mathematically equivalent form :

$$\hat{\mathbf{x}}_n = \hat{\mathbf{e}}_n + \sum_{k=1}^{\infty} \alpha^k \hat{\mathbf{e}}_{n-k} \quad (5)$$

In this form, we see clearly that in principle each past decoded prediction error vector  $\hat{\mathbf{e}}_{n-k}$  contributes to the value of the quantized vector  $\hat{\mathbf{x}}_n$ . Hence, in the case of channel errors, which would modify the value of  $\hat{\mathbf{e}}_n$  received by the decoder relative to what was sent by the encoder, the decoded vector  $\hat{\mathbf{x}}_n$  obtained in Equation 4 would not be the same at the decoder and at the encoder. Because of the recursive nature of the predictor, this encoder-decoder mismatch will propagate in the future and affect the next vectors  $\hat{\mathbf{x}}_{n+1}$ ,  $\hat{\mathbf{x}}_{n+2}$ , etc., even if there are no channel errors in later frames. In short, predictive vector quantization is not robust to channel errors, especially when the prediction factors are high ( $\alpha$  close to 1 in Equations 4 and 5).

To alleviate this propagation problem, Moving Average (MA) prediction can be used instead of AR prediction. In MA prediction, the infinite series in Equation (5) is truncated to a finite number of terms. The idea is to approximate the autoregressive form of the predictor in Equation (4) by using a small number of terms in Equation  
5 (5). Note that the weights in the summation can be modified to better approximate the predictor of Equation (4).

The MA predictive quantization is shown in Figure 4. A general form of the predictor P in Processor 402 is

$$\mathbf{p}_n = \mathbf{B}_1 \hat{\mathbf{e}}_{n-1} + \mathbf{B}_2 \hat{\mathbf{e}}_{n-2} + \dots + \mathbf{B}_K \hat{\mathbf{e}}_{n-K}$$

10 where  $\mathbf{B}_k$  are prediction matrices of dimension  $M \times M$  and  $K$  is the predictor order. Note that in MA prediction, transmission errors propagate only into next  $K$  frames.

A simple form for the predictor P in Processor 402 is to use first order prediction

$$\mathbf{p}_n = \mathbf{B} \hat{\mathbf{e}}_{n-1} \quad (6)$$

15 where  $\mathbf{B}$  is a prediction matrix of dimension  $M \times M$ , where  $M$  is the dimension of LP parameter vector. A simple form of the prediction matrix is a diagonal matrix with diagonal elements  $\beta_1, \beta_2, \dots, \beta_M$ , where  $\beta_i$  are prediction factors for individual LP parameters. If the same factor  $\beta$  is used for all LP parameters then Equation (6) reduces to

$$20 \quad \mathbf{p}_n = \beta \hat{\mathbf{x}}_{n-1} \quad (7)$$

Using the simple prediction form of Equation (7), then in Figure 4, the quantized vector  $\hat{\mathbf{x}}_n$  is given by the following moving average (MA) relation

$$\hat{\mathbf{x}}_n = \hat{\mathbf{e}}_n + \beta \hat{\mathbf{e}}_{n-1} \quad (8)$$

The structure of predictive vector quantization using MA prediction is shown in Figure 4. In this figure, the predictor memory in Processor 402 is formed by the past decoded prediction error vectors  $\hat{e}_{n-1}$ ,  $\hat{e}_{n-2}$ , etc. Hence, the maximum number of frames over which a channel error can propagate is the order of the predictor  $P$  (processor 402). In illustrative predictor example of Equation (8), for 1<sup>st</sup> order prediction is used so the MA prediction error can only propagate over one frame.

While more robust to transmission errors than AR predictors, MA predictors do not achieve the same prediction gain for a given prediction order. The prediction error has consequently a greater dynamic range, and can require more bits to achieve the same coding gain as with AR predictive quantization. The compromise is thus robustness to channel errors versus coding gain at a given bit rate.

In source-controlled variable bit rate (VBR) coding, the encoder operates at several bit rates, and a rate selection module is used to determine the bit rate used for encoding each speech frame based on the nature of the speech frame (e.g. voiced, unvoiced, transient, background noise). The goal is to attain the best speech quality at a given average bit rate, also referred to as average data rate (ADR). As an illustrative example, in CDMA systems (e.g. CDMA-one and CDMA2000), typically 4 bit rates are used and they are referred to as full-rate (FR), half-rate (HR), quarter-rate (QR), and eighth-rate (ER). In this system two rate sets are supported referred to as Rate Set I and Rate Set II. In Rate Set II, a variable-rate encoder with rate selection mechanism operates at source-coding bit rates to 13.3 (FR), 6.2 (HR), 2.7 (QR), and 1.0 (ER) kbit/s.

In VBR coding, a classification and rate selection mechanism is used to classify the speech frame according to its nature (voiced, unvoiced, transient, noise, etc.) and selects the bit rate needed to encode the frame according to the classifier information and the required average data rate. Half-rate encoding is typically chosen in frames where the input speech signal is stationary. The bit savings (compared to the full rate) are achieved by updating encoder parameters less frequently or by using fewer bits to encode some parameters. Further, these frames exhibit strong correlation

which can be exploited to reduce the bit rate. Specifically, in stationary voiced segments, the pitch information is encoded only once in a frame, and fewer bits are used for the fixed codebook and the LP coefficients. In unvoiced frames, no pitch prediction is needed and the excitation can be modeled with small codebooks in HR  
5 or random noise in QR.

Since predictive VQ with MA prediction is typically applied to encode the LP coefficients, this results in an unnecessary increase in quantization noise. MA prediction, as opposed to AR prediction, is used to increase the robustness to frame losses; however, in stationary frames the LP coefficients evolve slowly so using AR  
10 prediction in this case would have a smaller impact on error propagation in the case of lost frames. This can be seen by observing that, in the case of missing frames, most decoders apply a concealment procedure which essentially extrapolates the coefficients of the last frame. If the missing frame is stationary voiced, this extrapolation gives very similar values to the actual transmitted (but not received) LP  
15 parameters. The reconstructed LP vector is thus close to what would have been decoded if the frame had not been lost. In that specific case, using AR prediction in the quantization procedure of the LP coefficients can not have a very adverse effect on quantization error propagation.

Thus, in the this invention a predictive VQ method for LP parameters is  
20 disclosed whereby the predictor is switched between MA and AR prediction according to the nature of the speech frame being processed. More specifically, in transient and nonstationary frames MA prediction is used while in stationary frames AR prediction is used. Further, due to the fact that AR prediction results in a prediction error vector  $\mathbf{e}_n$  with a smaller dynamic range than MA prediction, then it is  
25 not efficient to use the same quantization tables for both types of prediction. To overcome this problem, the prediction error after AR prediction is properly scaled so that it can be quantized using the same quantization tables as in the MA prediction case. When multistage VQ is used to quantize the prediction error, the first stage can be used for both types of prediction after properly scaling the AR prediction error.

Since it is sufficient to use split VQ in the second stage which doesn't require large memory, the second stage quantization tables can be trained and designed separately for both types of prediction. Note that instead of designing the 1<sup>st</sup> stage tables with MA prediction and scaling the AR prediction error, the opposite is also valid, that is,  
5 the 1<sup>st</sup> stage can be designed for AR prediction and the MA prediction error vector is scaled.

Thus, it is also disclosed in this invention a predictive vector quantization method for quantizing LP parameters in a variable bit rate speech codec whereby the predictor is switched between MA and AR prediction according to classification  
10 information regarding the nature of the speech frame being processed, and whereby the prediction error vector is properly scaled such that the same first stage quantization tables in a multistage VQ of the prediction error can be used for both types of prediction.

An illustrative embodiment of the disclosed invention is given below.  
15 Figures 1 shows an illustration of a two-stage VQ. An input vector  $\mathbf{x}$  is first quantized with the quantizer Q1 in processor 101 to produce a quantized vector  $\hat{\mathbf{x}}_1$  and a quantization index  $i_1$ . The difference between the input vector and first stage quantized vector is computed and further quantized with a second stage VQ to produce the quantized second stage error vector  $\hat{\mathbf{x}}_2$  with quantization index  $i_2$ . The  
20 indices of  $i_1$  and  $i_2$  are transmitted and the quantized vector is reconstructed at the decoder as  $\hat{\mathbf{x}} = \hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$ .

Figure 2 shows an illustrative example of split vector quantization. An input vector  $\mathbf{x}$  of dimension  $M$  is split into  $K$  subvectors of dimensions  $N_1, N_2, \dots, N_K$ , and quantized with vector quantizers  $Q_1, Q_2, \dots, Q_K$ , respectively. The quantized  
25 subvectors  $\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \dots, \hat{\mathbf{y}}_K$ , with quantization indices  $i_1, i_2$ , and  $i_K$ , are found. The quantization indices are transmitted and the quantized vector  $\hat{\mathbf{x}}$  is reconstructed by simple concatenation of quantized subvectors.

An efficient approach for vector quantization is to combine both multistage and split VQ which results in a good trade-off between quality and complexity. In a first illustrative example, a two-stage VQ can be used whereby the second stage error vector  $\hat{x}_2$  is split into several subvectors and quantized with second stage quantizers  $Q_{21}, Q_{22}, \dots, Q_{2K}$ , respectively. In an second illustrative example, the input vector can be split into two subvectors, then each subvector is quantized with two-stage VQ using further split in the second stage as in the first illustrative example.

Figure 5 is a schematic block diagram showing an illustrative embodiment of a switched predictive VQ according to the present invention. In Processor 501, the vector of mean LP parameters  $\mu$  is removed from the input LP parameter vector  $z$  to produce the mean-removed LP parameter vector  $x$ . Note that the LP parameter vectors can be vectors of LSF parameters, ISF parameters, or any other relevant LP parameter representation. Removing the mean vector is optional and results in improved prediction performance. If Processor 501 is disabled then the vector  $x$  will be the same as  $z$ . Note that the frame index  $n$  used in Figures 3 and 4 has been dropped here for simplification. The predicted error vector is then computed and removed from the mean-removed vector  $x$  to produce the prediction error vector  $e$  (Processor 502). Now, according to the present invention, based on frame classification information, if the frame is stationary voiced then AR prediction is used and the error vector  $e$  is scaled by a certain factor to obtain the scaled error vector  $e'$ . The scaling factor is typically larger than 1 and results in upscaling the dynamic range of the prediction error so that it can be quantized with a quantizer designed for MA prediction. The value of the scaling factor depends on the coefficients used for MA and AR prediction. Typical values are: MA prediction coefficient  $\beta=0.33$ , AR prediction coefficient  $\alpha=0.65$ , and scaling factor equal 1.25. Note that if the quantizer is designed for AR prediction then the opposite will apply, that is, the error vector in case of MA prediction will be scaled and the scaling factor will be less than 1.

The scaled prediction error vector  $e'$  is then vector quantized in 508 to produce the quantized scaled error vector  $\hat{e}'$ . In this illustrative embodiment, the

vector quantizer 508 consists of a two-stage quantizer where split VQ is used in both stages and whereby the first stage vector quantization tables are the same for both MA and AR prediction. The vector quantizer 508 consists of blocks 504, 505, 506, 507, and 509. In the first stage quantizer Q1, the scaled prediction error vector is quantized to produce the first stage quantized vector  $\hat{\mathbf{e}}_1$ . This vector is removed from the input scaled error vector in Processor 505 to produce the second stage vector  $\mathbf{e}_2$ . The vector  $\mathbf{e}_2$  is then quantized in 506 by either second stage quantizer  $Q_{MA}$  or second stage quantizer  $Q_{AR}$  to produce the second stage quantized vector  $\hat{\mathbf{e}}_2$ . The choice of the second stage quantizer depends on the frame classification information. The scaled quantized error vector is reconstructed in Processor 509 by the addition of the quantized vectors from the two stages. That is  $\hat{\mathbf{e}}' = \hat{\mathbf{e}}_1 + \hat{\mathbf{e}}_2$ . Finally, inverse scaling is applied to the quantized scaled error vector in Processor 510 to produce the quantized prediction error vector  $\hat{\mathbf{e}}$ . Note that in this illustrative example the vector dimension is 16, and split VQ is used in both stages. The quantization index sets  $\mathbf{i}_1$  and  $\mathbf{i}_2$  are multiplexed and transmitted in Processor 507.

The prediction vector  $\mathbf{p}$  is computed in either Processor 511 or Processor 512 depending the frame classification information. If the frame is stationary voiced then the prediction vector is equal to the output of the AR predictor 512. Otherwise the prediction vector is equal to the output of the MA predictor 511. As explained above the MA predictor 511 operates on the quantized error vectors from previous frames while the AR predictor 512 operates on the quantized input vector from previous frames. The quantized input vector (mean-removed) is constructed by adding the quantized error vector to the prediction vector in Processor 514. That is,  $\hat{\mathbf{x}} = \hat{\mathbf{e}} + \mathbf{p}$ .

Figure 6 is a schematic block diagram showing an illustrative embodiment of a switched predictive VQ at the decoder according to the present invention. At the decoder side, the received sets of indices  $\mathbf{i}_1$  and  $\mathbf{i}_2$  are used by the quantization tables 601 and 602 to produce the first and second stage quantized vectors  $\hat{\mathbf{e}}_1$  and  $\hat{\mathbf{e}}_2$ . Note that the second stage quantization 602 consists of two sets of tables for MA and AR

prediction as at the encoder side of Figure 5. The scaled error vector is then reconstructed in Processor 603 by the addition of the quantized vectors from the two stages. That is,  $\hat{e}' = \hat{e}_1 + \hat{e}_2$ . Inverse scaling is applied in Processor 609 to produce the quantized prediction error vector  $\hat{e}$ . Note that the inverse scaling is a function of the received frame classification information. The quantized (mean-removed) input vector is then reconstructed in Processor 604 by adding the prediction vector  $\mathbf{p}$  to the quantized error vector  $\hat{e}$ . That is,  $\hat{\mathbf{x}} = \hat{e} + \mathbf{p}$ . In case the mean vector has been removed at the encoder side, it is added in Processor 608 to produce the quantized LP parameter vector  $\hat{\mathbf{z}}$ . Note that as in the case of the encoder side of Figure 5, the prediction vector  $\mathbf{p}$  is either the output of the MA predictor 605 or the AR predictor 606 depending on the frame classification information (according to the logic in Processor 607).

Note that despite the fact that only the output of either the MA predictor or the AR predictor is used in a certain frame, the memories of both predictors need to always be updated in each frame. This is valid for both the encoder and decoder sides.

To optimize the encoding gain, some vectors of the first stage, designed for MA prediction, can be replaced by new vectors designed for AR prediction. In a second illustrative embodiment, the first stage codebook size is 256, and has the same content as in the AMR-WB standard at 12.65 kbit/s, and 28 vectors are replaced in the first stage codebook when using AR prediction. An extended, first stage codebook is thus formed as follows: first, the 28 first stage vectors less used when applying AR prediction are placed at the beginning of a table, then the remaining  $256 - 28 = 228$  first stage vectors are appended in the table, and finally 28 new vectors are put at the end of the table. The table length is thus  $256 + 28 = 284$  vectors. When using MA prediction, the first 256 vectors of the table are used in the first stage; when using AR prediction the last 256 vectors of the table are used. To ensure interoperability with the AMR-WB standard, a table is used which contains the mapping between the position of a first stage vector in this new codebook, and its original position in the AMR-WB first stage codebook.

To summarize, the novelty of the present invention, with respect to Figures 5 and 6, lies in the following aspects:

- 5                   - Switched AR/MA prediction is used depending on the encoding mode of the variable rate coder (which depends on the nature of the present speech frame).
- 10                  - Essentially the same first stage quantizer is used whether AR or MA prediction is applied (memory savings). In an illustrative embodiment, 16<sup>th</sup> order LP prediction is used and the LP parameters are represented in the ISF domain. The first stage codebook is the same as the one used in the 12.65 kbit/s mode of the AMR-WB encoder where the codebook was designed using MA prediction.
- 15                  - Instead of MA prediction, AR prediction is used in stationary modes, specifically half-rate voiced mode; otherwise, MA prediction is used.
- 20                  - In the case of AR prediction, the first stage of the quatizer is the same as the MA prediction case, however, the second stage can be properly designed and trained for AR prediction.
- 25                  - To take into account this switching in the predictor mode, the memories of both MA and AR predictors have to be updated in each frame, assuming both MA or AR prediction can be used for the next frame.
- Further, to optimize the encoding gain, some vectors of the first stage, designed for MA prediction, can be replaced by new

vectors designed for AR prediction. In this illustrative embodiment, 28 vectors are replaced in the first stage codebook when using AR prediction.

- 5                   -       An enlarged, first stage codebook is thus formed as follows: first, the 28 first stage vectors less used when applying AR prediction are placed at the beginning of a table, then the remaining  $256-28 = 228$  first stage vectors are appended in the table, and finally 28 new vectors are put at the end of the table. The table length is thus  $256 + 28 = 284$  vectors. When using MA prediction, the  
10                   first 256 vectors of the table are used in the first stage; when using AR prediction the last 256 vectors of the table are used.
- 15                   -       To ensure interoperability with the AMR-WB standard, a table is used which contains the mapping between the position of a first stage vector in this new codebook, and its original position in the AMR-WB first stage codebook.
- 20                   -       Since AR prediction achieves lower prediction error energy than MA prediction (when used on stationary signals), a scaling factor has to be applied to the prediction error. In this illustrative embodiment, the scaling factor is 1 when MA prediction is used, and  $1/0.8$  when AR prediction is used. This increases the AR prediction error to a dynamic equivalent to the MA prediction error. Hence, the same quantizer can be used for both MA and AR prediction in the first stage.

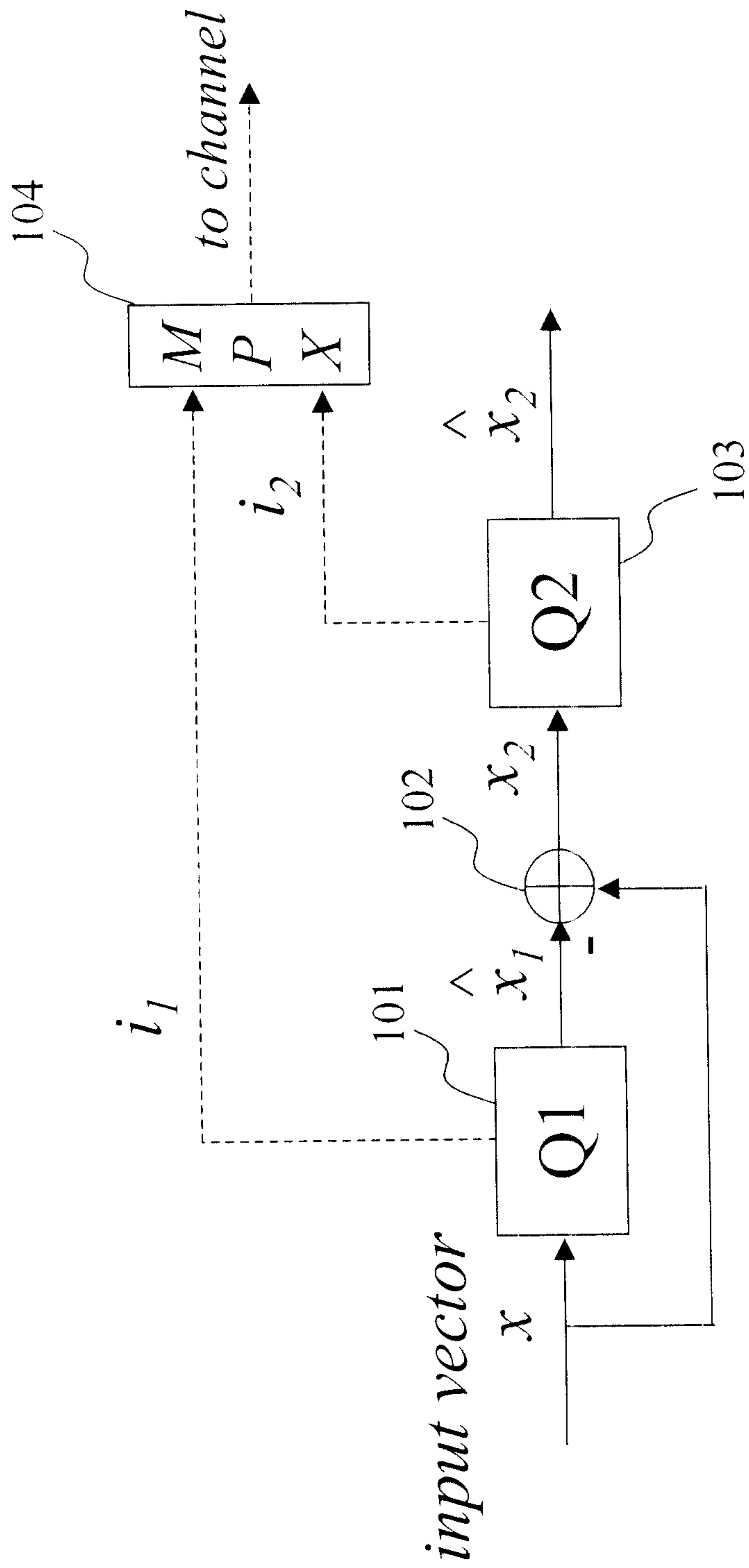
## REFERENCES

- 25       [1] ITU-T Recommendation G.722.2 "Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB)", Geneva, 2002.

A method and device for robust predictive vector quantization of  
linear prediction parameters in variable bit rate speech coding

**23 of 23**

- [2] ITU-T Recommendation G.729 “Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP),” Geneva, March 1996.



$$\hat{x} = \hat{x}_1 + \hat{x}_2$$

Figure 1. Principle of a two-stage vector quantizer

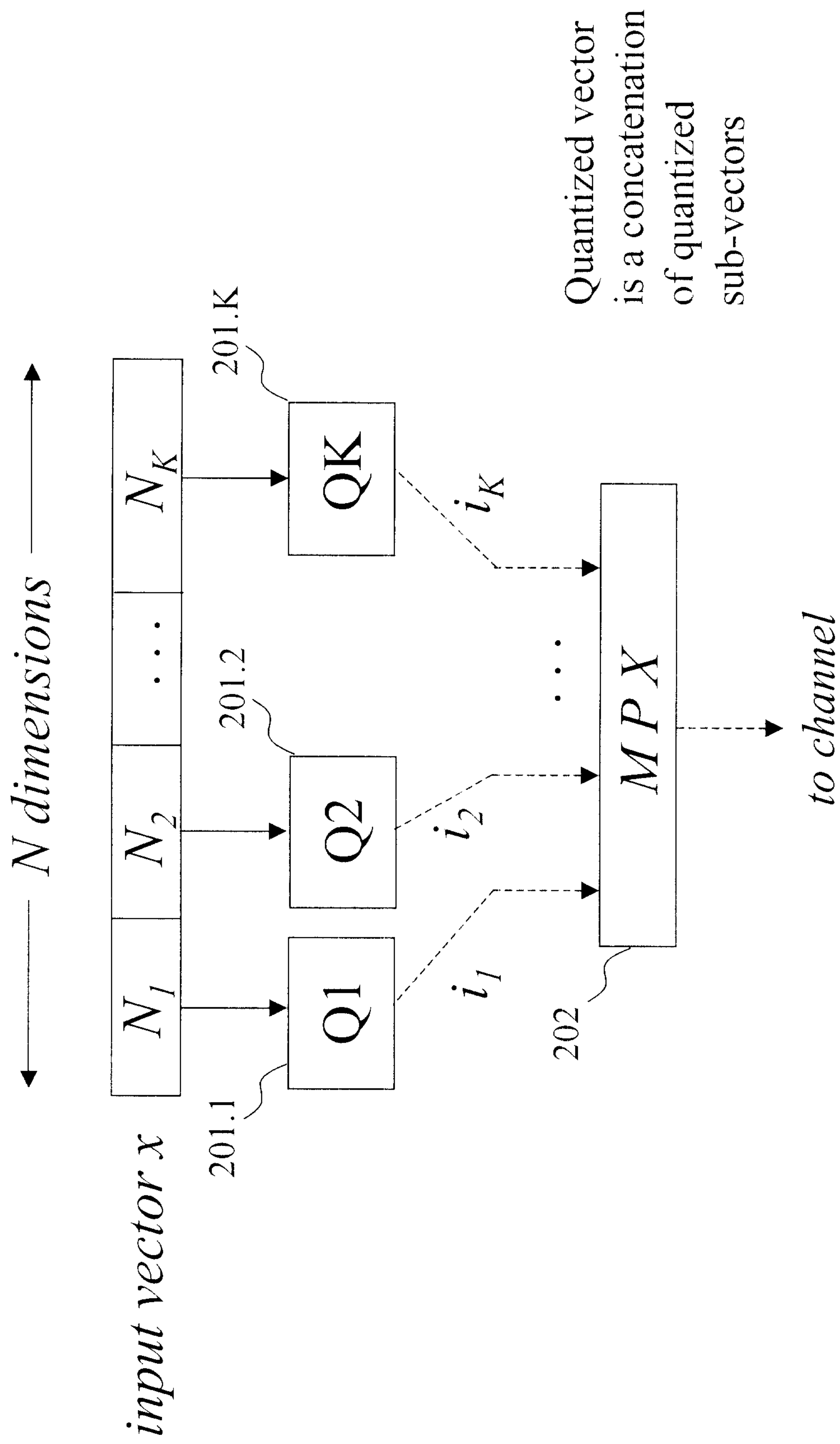


Figure 2. Principle of a split vector quantizer

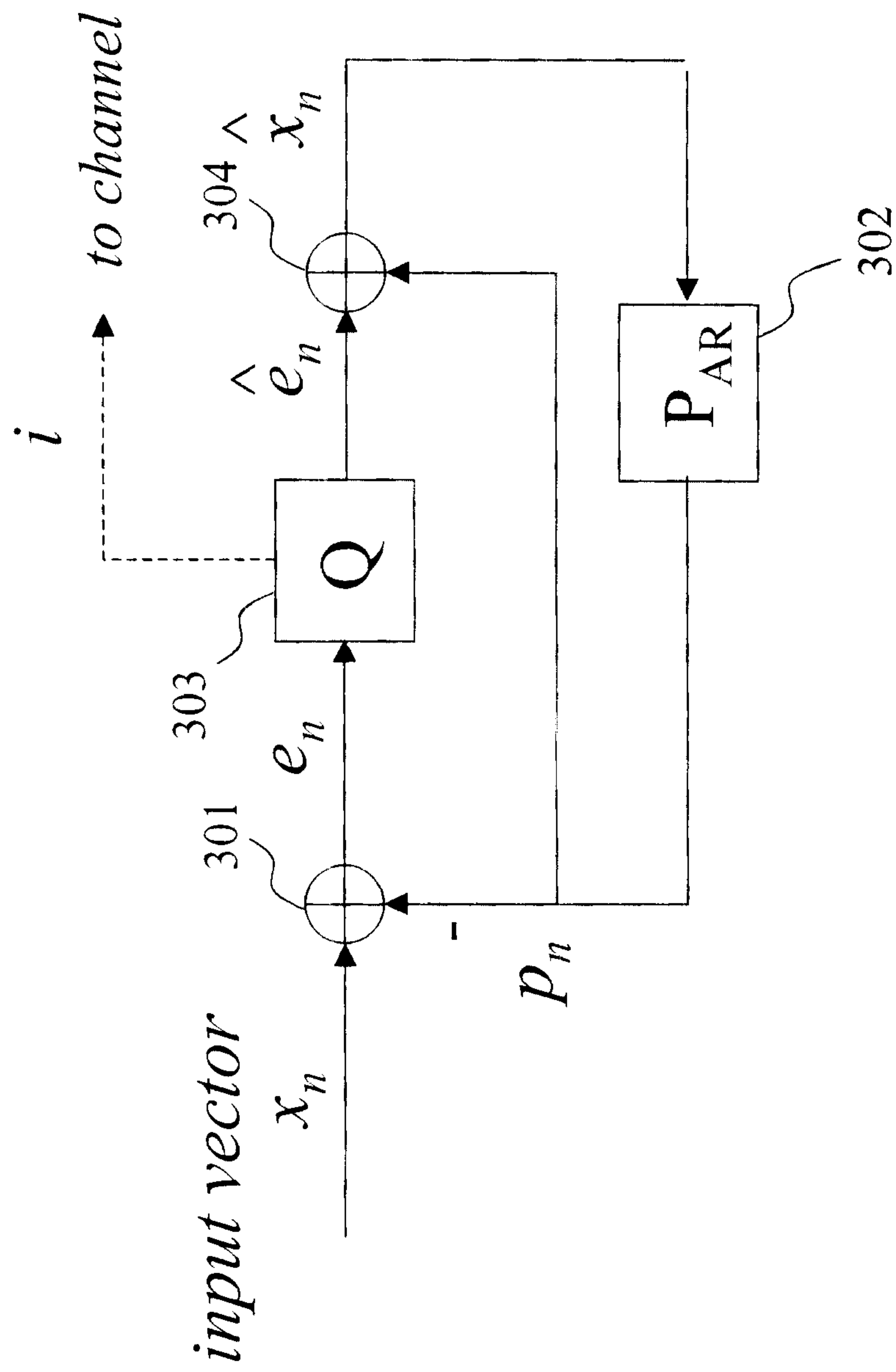


Figure 3. Principle of a predictive vector quantizer using autoregressive (AR) prediction.

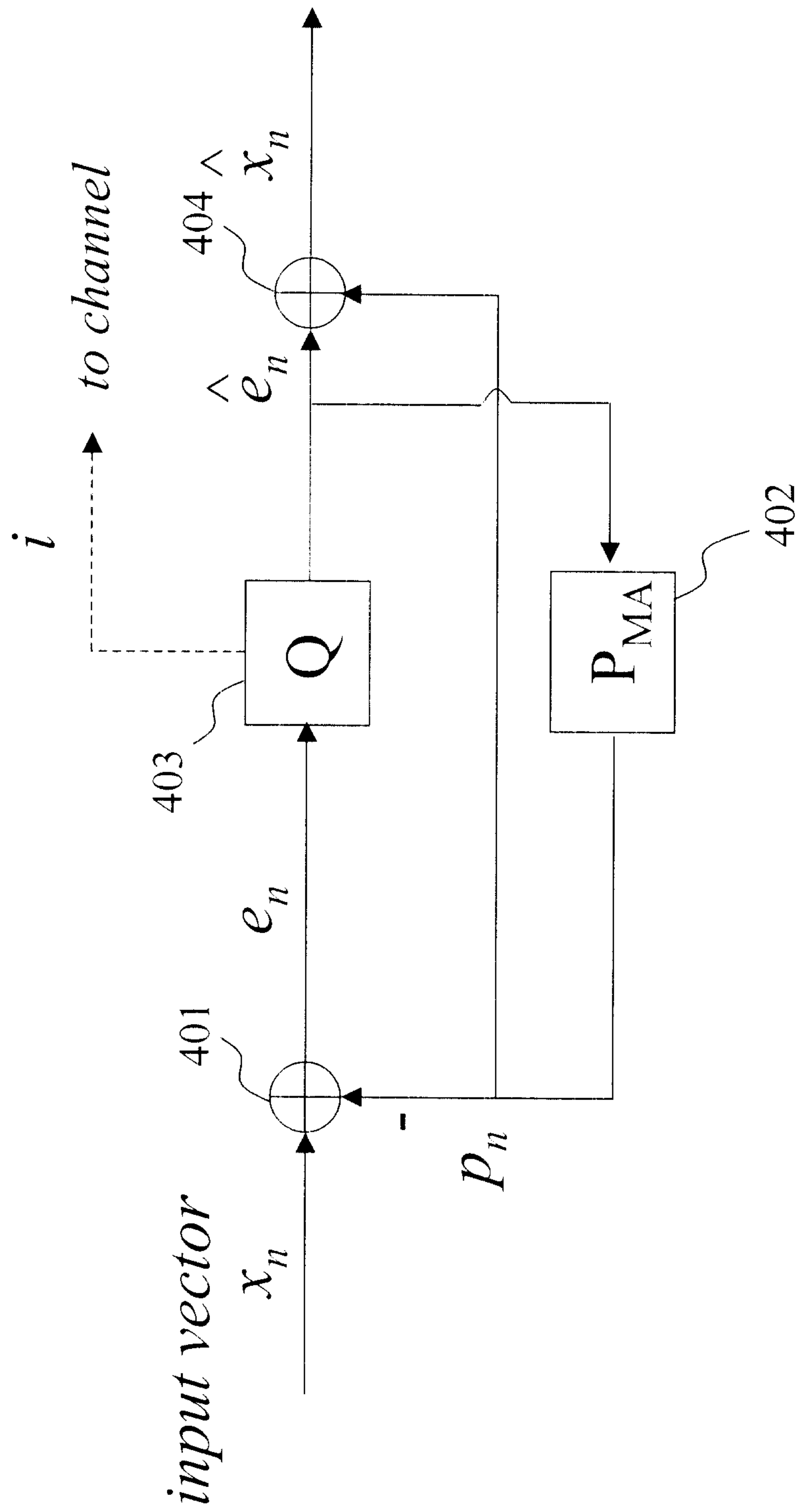


Figure 4. Principle of a predictive vector quantizer using Moving Average (MA) prediction

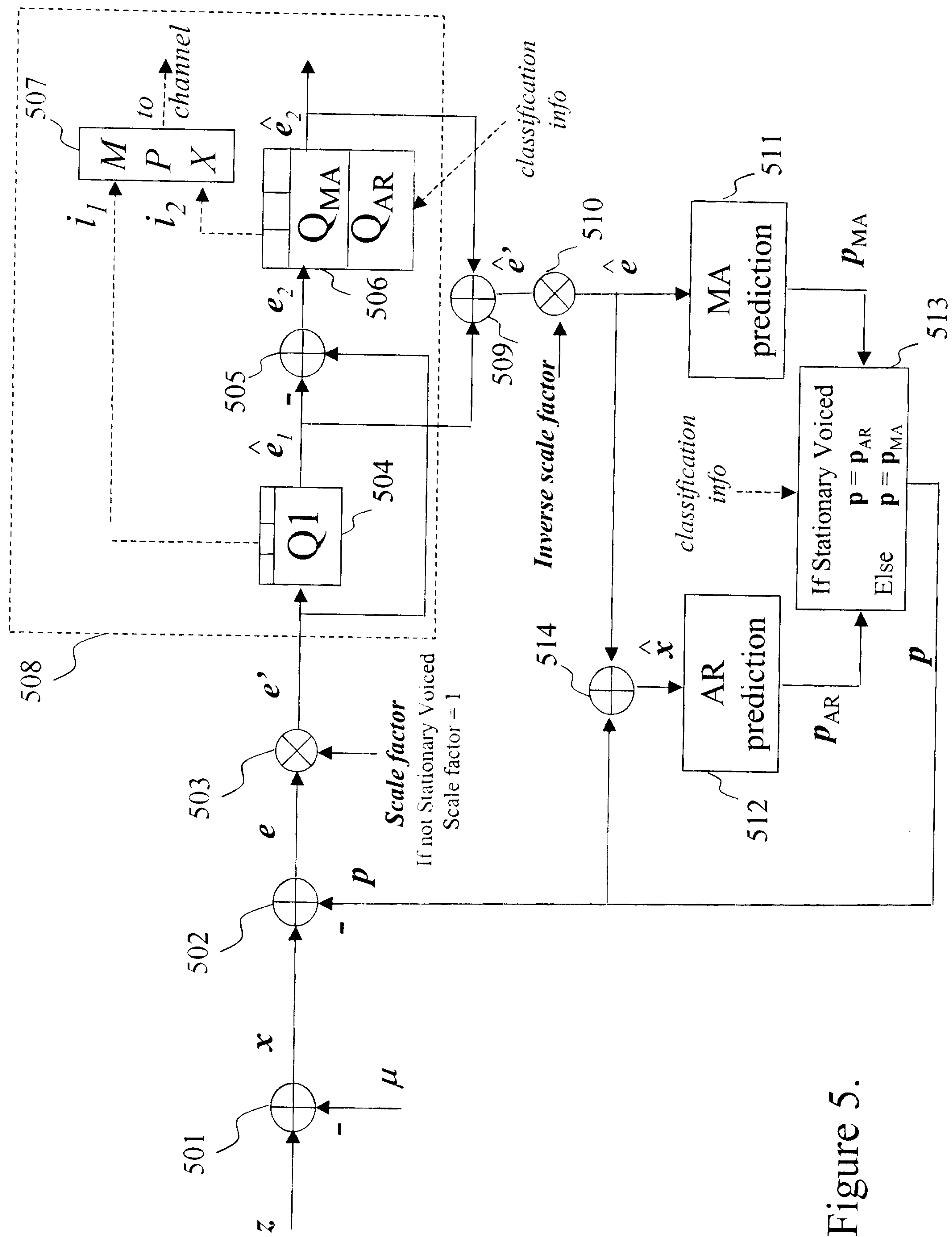


Figure 5.

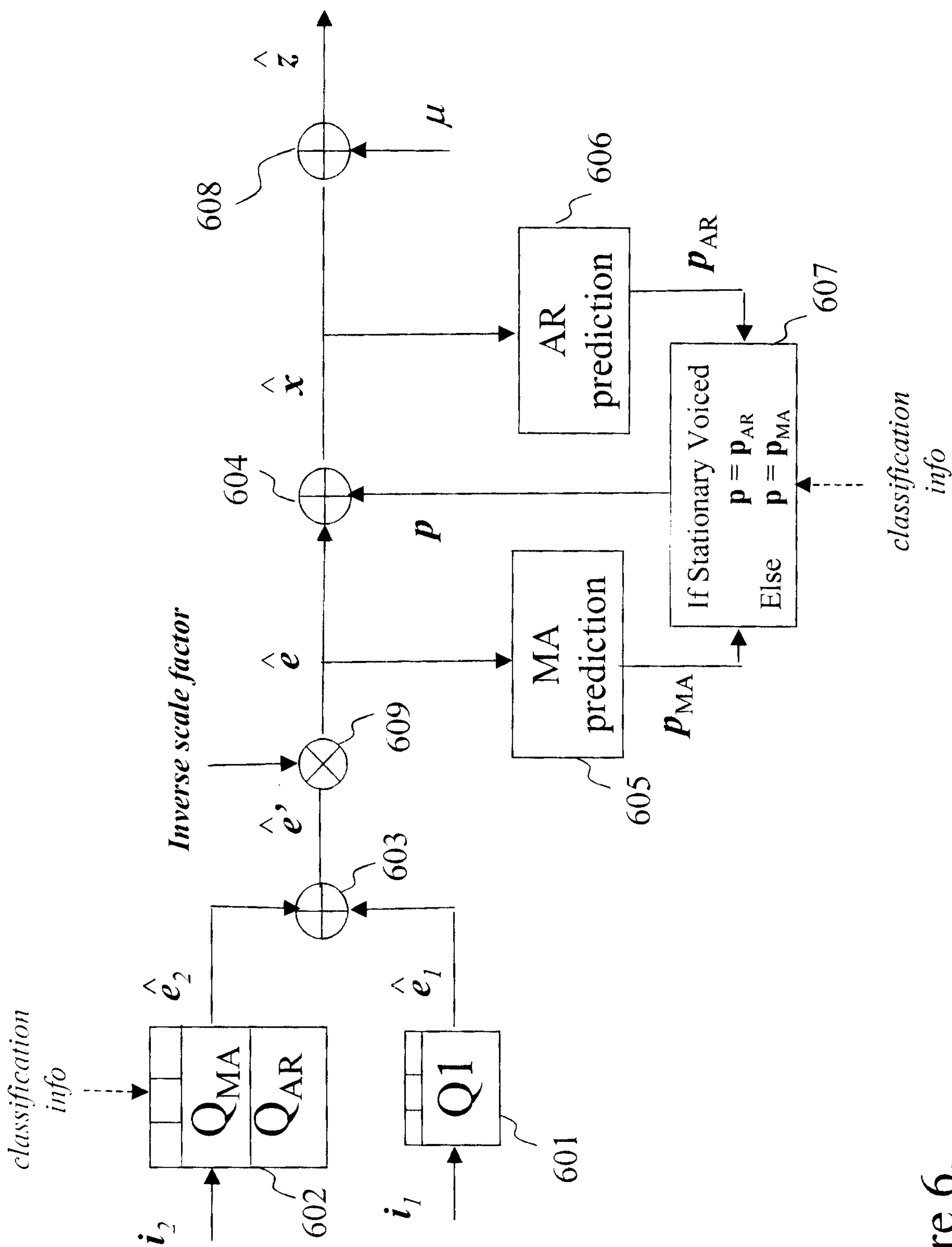


Figure 6.

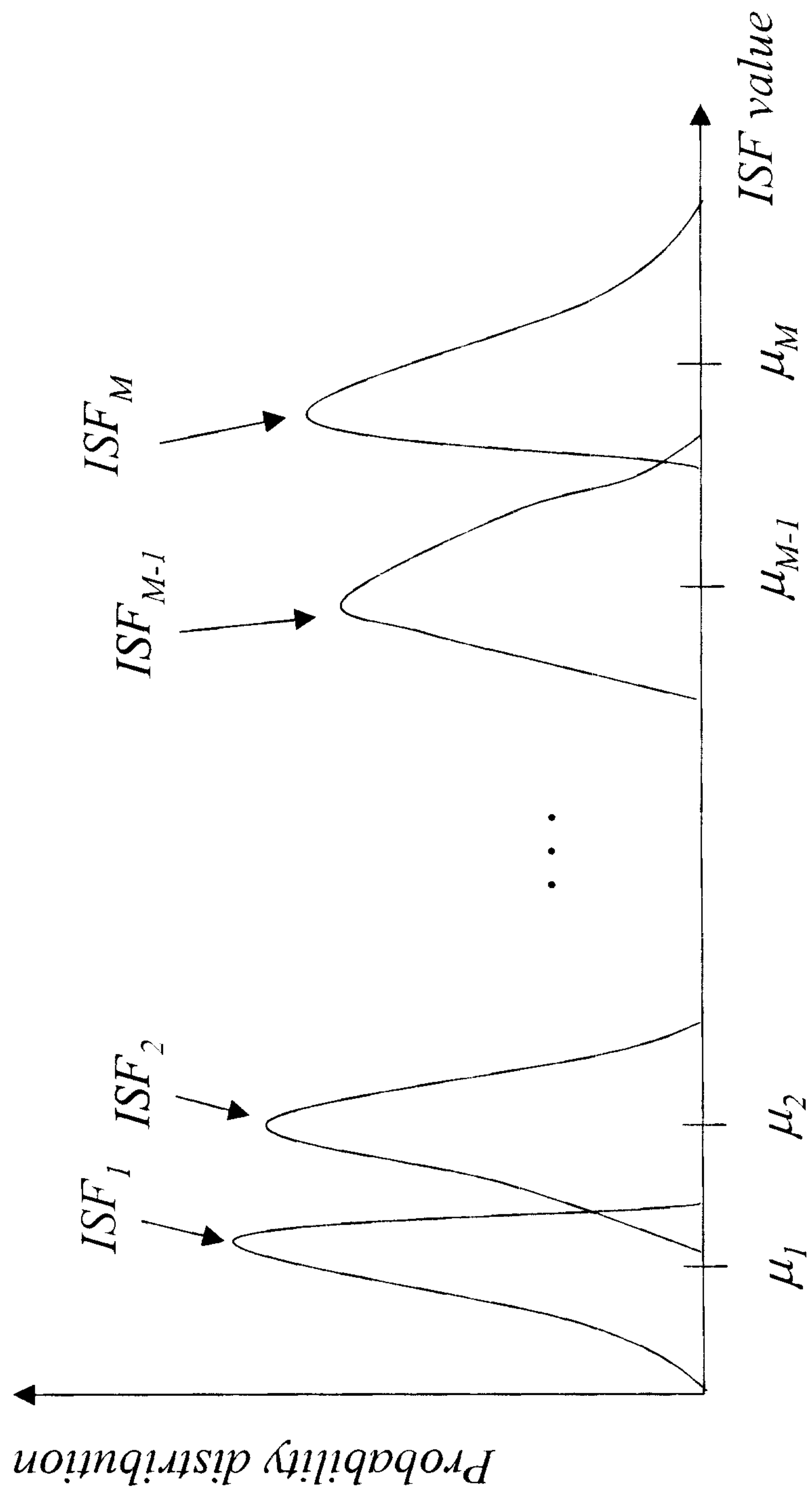


Figure 7. Example distribution of ISF coefficients

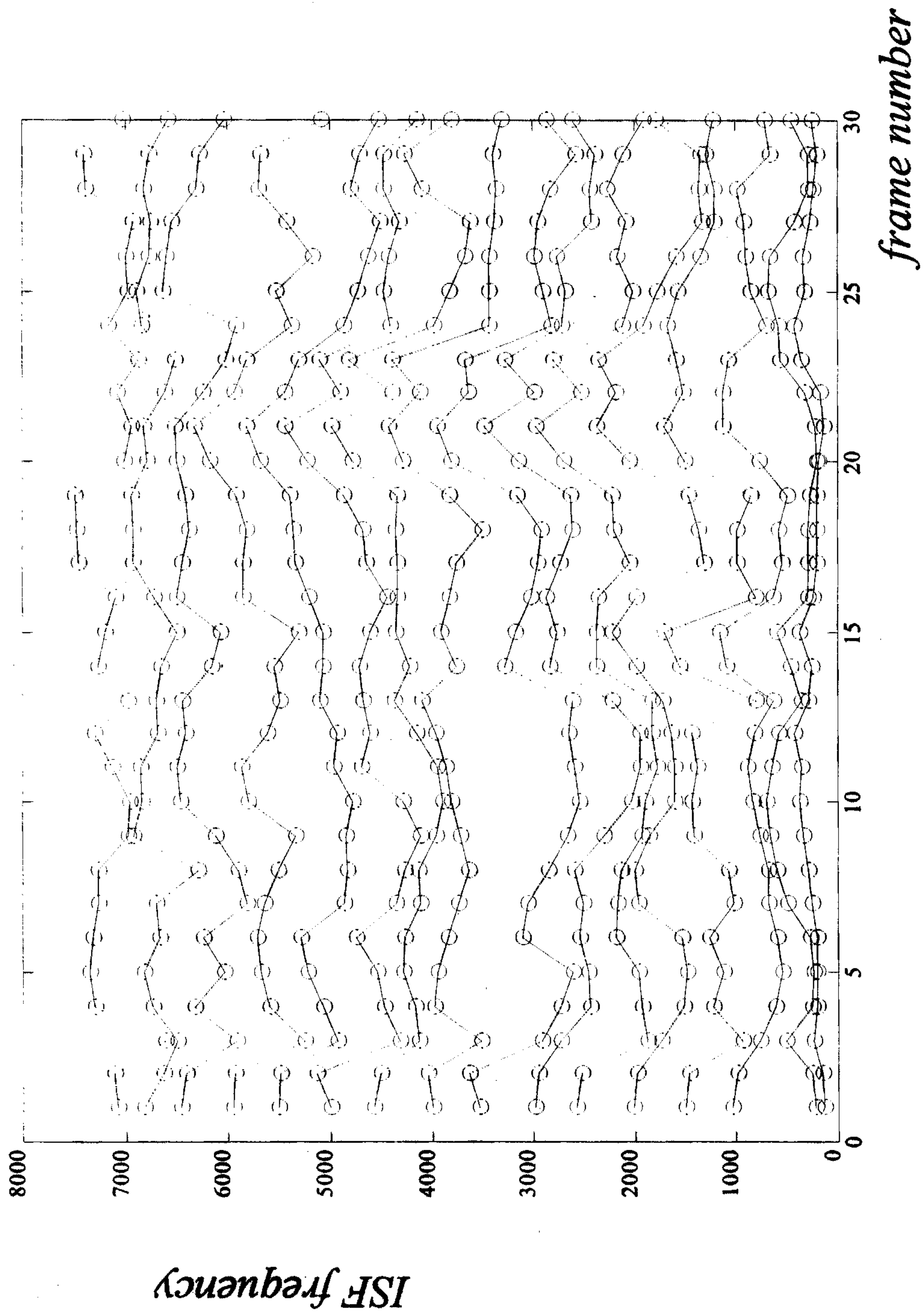


Figure 8. Typical evolution in time of ISF coefficients  
(horizontal axis is frame number)

