The Switched Split Vector Quantiser and its Application to LPC Parameter Quantisation in Wideband Speech Coding

Stephen So, *Member, IEEE,* and Kuldip K. Paliwal, *Member, IEEE*

***Abstract*— In this paper, we present an overview of the switched split vector quantiser (SSVQ), its advantages over the traditional split vector quantiser, and its application to LPC parameter quantisation in wideband speech coding. By utilising an unconstrained switch vector quantiser as an initial stage, the SSVQ is able to exploit global vector dependencies in order to compensate for rate-distortion (R-D) losses in the split vector quantiser, which are due to the structural constraints on the codebook. The resulting scheme is not only more efficient in the R-D sense, but also possesses lower computational complexity than the split vector quantiser. We apply the SSVQ to quantising line spectral frequency (LSF) parameters of wideband speech and compare its spectral distortion performance with the split vector quantiser (SVQ), PDF-optimised scalar quantiser, and the split-multistage vector quantiser (S-MSVQ) with MA predictor from the ITU-T G.222.2 AMR-WB speech coder. The results show that the SSVQ requires less bits for achieving transparent coding of wideband LSFs than SVQ and scalar quantisers. Finally, SSVQ (which is a memoryless scheme) achieves comparable spectral distortion with the S-MSVQ with MA predictor at 36 and 46 bits/frame.**

***Index Terms*— line spectral frequencies, LPC parameter coding, vector quantisers, wideband speech coding.**

I. Introduction

VECTOR quantisers (VQ) are well known to be the best quantiser one can design to achieve the lowest distortion for a given bitrate[1]. However, their computational and memory requirements increase exponentially with bitrate. For the application of LPC parameter quantisation in speech coding, estimates of the bitrate required to achieve transparent coding[2] range from 19 bits/frame [2] to 23 bits/frame [3] for narrowband (200–3400 Hz) speech and 35 bits/frame for wideband (50–7000 Hz) speech [4]. At these bitrates, unconstrained vector quantisers are too complex to design and operate, hence structural constraints need to be placed on the codebook to decrease the computational complexity.

The most common structurally-constrained vector quantisers are the *product code vector quantisers*, which include split vector quantisers (SVQ) [5], multistage vector quantisers (MSVQ) [5], [6], etc. The SVQ divides vectors into sub-vectors of lesser dimension and these are then quantised using independent codebooks, while in MSVQ, vector residuals are quantised in stages by coarser codebooks. While SVQ and MSVQ have considerably lower computational complexity and memory requirements, they suffer in terms of R-D efficiency.

The switched split vector quantiser (SSVQ) is a product code vector quantiser that is more efficient, in terms of rate-distortion (R-D) performance and computational complexity than the split vector quantiser. It consists of two-stages: an unconstrained vector quantiser that acts as a switch; and a series of split vector quantisers. In previous studies [7], we have shown that the SSVQ requires less bits and possesses lower complexity than SVQ for achieving transparent coding of narrowband LPC parameters, though at the expense of an increase in memory requirements.

S. So is with the School of Engineering, Gold Coast Campus, Griffith University. E-mail: s.so@griffith.edu.au

K.K. Paliwal is with the School of Microelectronic Engineering, Nathan Campus, Griffith University. E-mail: k.paliwal@griffith.edu.au

[1]According to Shannon asymptotic theory [1], the operating rate-distortion function of VQ approaches the Shannon limit as the vector dimension is made arbitrarily large. However, it can be shown that even for finite dimensional sources, VQ will always perform better than other quantisation schemes.

[2]Transparent coding means that the coded speech cannot be distinguished from the original speech.

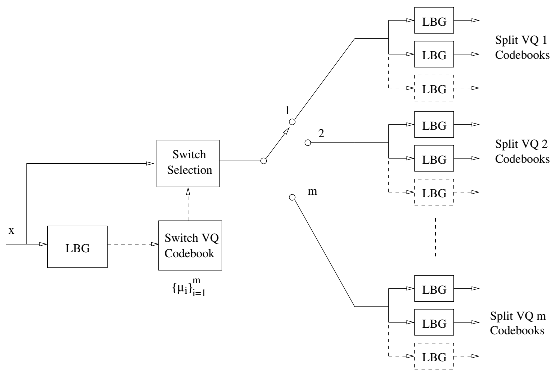

Fig. 1. SSVQ Codebook Training

In this paper, we review the switched split vector quantiser, as well as describe its advantages over the traditional SVQ using the idea of 'vector quantiser advantages' [8]. We also compare the spectral distortion performance of the SSVQ with other quantisation schemes for the coding of line spectral frequencies (LSF) derived from wideband speech. These schemes include PDF-optimised scalar quantisers, split vector quantisers, and the split-multistage vector quantiser with MA predictor from the ITU-T G.722.2 AMR-WB speech codec [9].

II. Switched Split Vector Quantisation

A. SSVQ Codebook Training

Fig. 1 shows a block diagram of the SSVQ codebook training. The LBG algorithm [10] is first applied on all vectors to produce m centroids (or means) $\{\boldsymbol{\mu}_i\}_{i=1}^m$. In the Euclidean distortion sense, these centroids are the 'best' representation of all the vectors in that Voronoi region. Hence, we can use them to form the *switch VQ codebook* which will be used for switch-direction selection. All the training vectors are classified based on the nearest-neighbour criterion:

$$j = \underset{i}{\operatorname{argmin}}\, d(\boldsymbol{x}, \boldsymbol{\mu}_i) \tag{1}$$

where $\boldsymbol{x}$ is the vector under consideration, j is the cluster (or, switching direction) to which the vector is classified, and $d(\boldsymbol{x}, \hat{\boldsymbol{x}})$ is the mean squared error between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$. With the training vectors classified to the m clusters, local SVQ codebooks are designed for each cluster (or, switching direction) using the corresponding training vectors.

B. SSVQ Coding

Fig. 2 shows a block diagram of SSVQ coding. Each vector to be quantised is first switched to one of the m possible directions based on the nearest-neighbour criterion defined by (1), using the switch VQ codebook, $\{\boldsymbol{\mu}_i\}_{i=1}^m$, and then quantised using the corresponding SVQ.

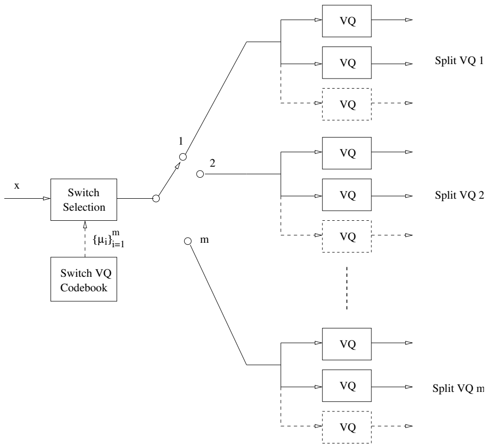

Fig. 2. SSVQ Coding

III. Advantages of SSVQ over SVQ

Lookabaugh and Gray [8] used high-resolution quantisation theory to explain why much vector quantisers performed better than scalar quantisers. They described three *vector quantiser advantages* [8]:

1) **Space-filling advantage:** Due to the increase in dimensionality, vector quantisers can use more efficient cell shapes that can fill more space than the rectangular-shaped cells of scalar quantisers;
2) **Shape advantage:** Code-vectors are free to appear anywhere in the higher-dimensional space which allows better matching of the marginal probability density function than scalar quantisers, whose code-points are constrained to lie on a rectangular grid; and
3) **Memory advantage:** Statistical dependencies (linear and non-linear) between the components are exploited by vector quantisers to reduce the 'wastage' of bits required to encode redundant information.

We can use these vector quantiser advantages to analyse the losses that are incurred when using split vector quantisation as well as explain how SSVQ alleviates two of these losses (more specifically, the memory and shape advantages).

A. The Memory Advantage of SSVQ

In order to illustrate the memory advantage of SSVQ over SVQ, we have generated some zero-mean Gaussian correlated random vectors of two dimensions with the following autocorrelation matrix:

$$\boldsymbol{R} = \begin{bmatrix} 4.5 & 4.0 \\ 4.0 & 6.0 \end{bmatrix} \tag{2}$$

We then quantised these random vectors using a two-part SVQ[3] and two-part SSVQ at 8 bits/vector. The SSVQ uses 16 switching directions and each part was allocated 2 bits. Fig. 3(a) and (b) shows the resulting product code-vectors of the SVQ and SSVQ, respectively. It can be observed in Fig. 3(a) that the SVQ has not exploited the correlation in the vectors, characterised by the tilt. This has resulted in a large amount of code-vectors that fall in areas where there are no vectors. In comparison, Fig. 3(b) shows the switch code-vectors (marked as crosses) have captured the major dependency between the components, x_1 and x_2. Each of the local SVQs have been placed according to these switch code-vectors where we can see a better utilisation of the code-vectors. The 8 bits/vector SSVQ has made a 2.58 dB gain in SNR over the SVQ.

[3]Note that two-part SVQ of two dimensional vectors is the same as applying PDF-optimised scalar quantisation on each dimension.

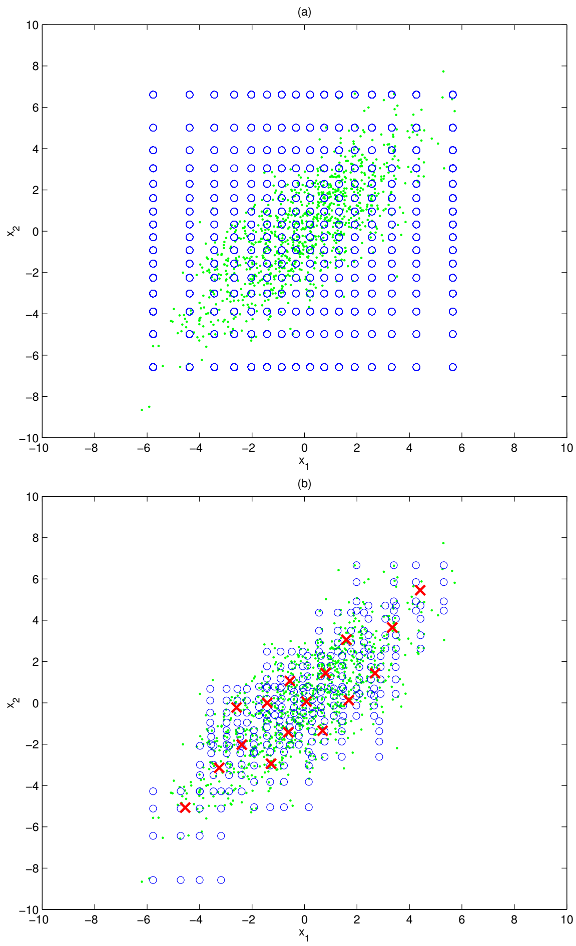

Fig. 3. Illustrating the memory advantage of SSVQ over SVQ: (a) 8 bits/vector split vector quantiser (SNR=20.32 dB); with (b) 8 bits/vector switched split vector quantiser (SNR=22.9 dB). The correlated Gaussian random vectors are represented as dots, the code-vectors as circles, and the switch code-vectors of SSVQ as crosses.

B. The Shape Advantage of SSVQ

In order to observe the shape advantage of the SSVQ over SVQ, we have generated some memoryless Gaussian random vectors with the following autocorrelation matrix:

$$\boldsymbol{R} = \begin{bmatrix} 6 & 0 \\ 0 & 1 \end{bmatrix} \tag{3}$$

Because the vectors have no memory, any gains in SNR are mostly due to the shape advantage. In this example, where the same vector splitting is used for both SSVQ and SVQ, there will be no space-filling advantage of SSVQ over SVQ since both split vectors into subvectors of the same dimensionality, which constrains the quantiser cell shapes to be the same (in this case, rectangular).

Figs. 4(a) and (b) show the product code-vectors of an 8 bits/vector split vector quantiser and 8 bits/vector switched split vector quantiser, respectively. When observing Fig. 4(a), we can see that the lattice of SVQ code-vectors does not match the marginal shape of the two dimensional PDF, which is elliptical. In comparison, the code-vectors resulting from SSVQ, as shown in Fig. 4(b), match the elliptical shape of the marginal PDF more closely than the rectangular shape of the

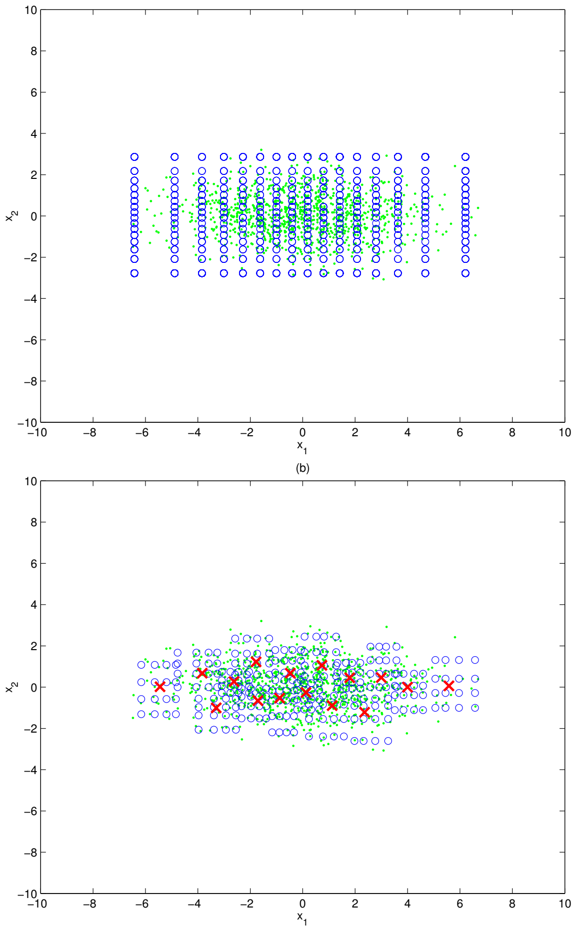

Fig. 4. Illustrating the shape advantage of SSVQ over SVQ: (a) 8 bits/vector split vector quantiser (SNR=21.29 dB); with (b) 8 bits/vector switched split vector quantiser (SNR=21.78 dB). The memoryless Gaussian random vectors with autocorrelation matrix of (3) are represented as dots, the code-vectors as circles, and the switch code-vectors of SSVQ as crosses.

SVQ. In terms of SNR, we see that the shape advantage gain of SSVQ over SVQ is approximately 0.5 dB.

IV. Line Spectral Frequency Representation of LPC Parameters

In the LPC analysis of speech, a short segment of speech is assumed to be the output of an all-pole filter, $H(z) = \frac{1}{A(z)}$, driven by either white Gaussian noise (for unvoiced speech) or a periodic sequence of impulses (for voiced speech), where $A(z)$ is the inverse filter given by [5]:

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n} \tag{4}$$

Here, n is the order of LPC analysis and $\{a_i\}_{i=1}^{n}$ are the LPC coefficients. The line spectral frequencies were proposed by Itakura [11] and are defined as the roots of the following polynomials:

$$P(z) = A(z) + z^{-(n+1)} A(z^{-1}) \tag{5}$$

and

$$Q(z) = A(z) - z^{-(n+1)} A(z^{-1}) \tag{6}$$

These two polynomials, $P(z)$ and $Q(z)$, are parametric models of the acoustic tube in two extremal states, where the $(n+1)$th stage (representing the glottis) is either completely closed or completely opened, respectively [12]. Consequently, LSFs have the following properties:

1) All zeros of $P(z)$ and $Q(z)$ lie on the unit circle;
2) zeros of $P(z)$ and $Q(z)$ are interlaced with each other; and
3) the minimum phase property of $A(z)$ is easily preserved after quantisation of the LSFs if the first two properties are satisfied.

Because of property 1, where each zero effectively represents a particular frequency (since it has no bandwidth), clusters of two to three LSFs define the location and bandwidth of formants in the power spectrum [5], [12]. Quantisation errors in the LSFs result in localised distortion in the power spectrum [5].

V. Average Spectral Distortion and Conditions for Transparent Coding

In order to objectively measure the distortion between a coded and uncoded LPC parameter vector, the spectral distortion is often used in narrowband speech coding [5]. For the ith frame, the spectral distortion (in dB), D_i, is defined as:

$$D_i = \sqrt{\frac{1}{F_s} \int_0^{F_s} \left[10 \log_{10} P_i(f) - 10 \log_{10} \hat{P}_i(f)\right]^2 df} \tag{7}$$

where F_s is the sampling frequency and $P_i(f)$ and $\hat{P}_i(f)$ are the LPC power spectra of the coded and uncoded ith frame, respectively. The conditions for transparent speech from narrowband LPC parameter quantisation are [5]:

1) The average spectral distortion (SD) is approximately 1 dB,
2) there is no outlier frame having more than 4 dB of spectral distortion, and
3) less than 2% of outlier frames are within the range of 2–4 dB.

According to Guibé *et al.* [13], listening tests have shown that these conditions for transparency, which are often quoted in the narrowband speech coding literature, also apply to the wideband case.

VI. Experimental Setup

The TIMIT database was used in the training and testing of the SSVQ, where speech is sampled at 16 kHz. We have used the preprocessing and LPC analysis of the AMR-WB speech codec (floating point version) to produce 16 linear prediction coefficients which are then converted to LSFs. The training set consists of 333789 vectors while the evaluation set, which consists of speech not contained in the training, has 85353 vectors. The weighted distance measure from [4], which emphasises LSFs that are situated near spectral peaks, was used in the SSVQ. Vectors are split into five parts for the SVQ and SSVQ $(3, 3, 3, 3, 4)$.

VII. Results and Discussion

Table I shows the spectral distortion, computational complexity, and memory requirements of the SSVQ for different switching directions, m. We can see that transparent coding of wideband LSFs requires 42 bits/frame. We note that as the number of switching directions is increased, there is generally a decrease in spectral distortion and percentage of outlier frames. Table II shows the average spectral distortion, computational complexity, and memory requirements of a five-part split vector quantiser, which uses unweighted mean squared error. It uses the same partition sizes as the five-part SSVQ. We can see that the five-part SVQ requires 46 bits/frame to achieve transparent coding. Comparing these results with Table I, we observe a saving of up to 4 bits/frame for transparent coding with the SSVQ. Also, the computational complexity of the transparent SSVQ is less than 40% of the complexity of the transparent SVQ. This confirms

TABLE I

AVERAGE SPECTRAL DISTORTION, COMPUTATIONAL COMPLEXITY, AND MEMORY REQUIREMENTS (ROM) OF THE FIVE-PART SWITCHED SPLIT VECTOR QUANTISER AS A FUNCTION OF BITRATE AND NUMBER OF SWITCH DIRECTIONS ON WIDEBAND LSF VECTORS FROM THE TIMIT DATABASE

m	Bits/ frame	Avg. SD (in dB)	Outliers (in %) 2–4 dB	Outliers (in %) > 4 dB	kflops/ frame	ROM (floats)
8	46	0.889	0.33	0.00	27.1	53376
	45	0.922	0.43	0.00	24.1	47232
	44	0.953	0.57	0.00	21.0	41088
	43	0.986	0.66	0.00	19.5	38016
	42	1.037	1.05	0.00	15.4	34944
16	46	0.878	0.34	0.00	24.6	94464
	45	0.906	0.44	0.00	21.5	82176
	44	0.936	0.50	0.00	20.0	76032
	43	0.975	0.64	0.00	18.4	69888
	42	1.018	0.83	0.00	17.7	66816

TABLE II

AVERAGE SPECTRAL DISTORTION, COMPUTATIONAL COMPLEXITY, AND MEMORY REQUIREMENTS (ROM) OF THE FIVE-PART SPLIT VECTOR QUANTISER AS A FUNCTION OF BITRATE ON WIDEBAND LSF VECTORS FROM THE TIMIT DATABASE

Bits/frame	Avg. SD (in dB)	Outliers (in %) 2–4 dB	Outliers (in %) > 4 dB	kflops/ frame	ROM (floats)
46	1.012	0.68	0.00	40.96	10240
45	1.061	0.99	0.00	32.76	8192
44	1.092	1.10	0.00	29.69	7424
43	1.151	1.70	0.00	26.62	6656
42	1.200	2.31	0.00	23.55	5888

TABLE III

AVERAGE SPECTRAL DISTORTION OF THE PDF-OPTIMISED SCALAR QUANTISERS AS A FUNCTION OF BITRATE ON WIDEBAND LSF VECTORS FROM THE TIMIT DATABASE

Bits/frame	Avg. SD (dB)	Outliers (in %) 2–4 dB	Outliers (in %) > 4 dB
61	0.918	0.82	0.00
60	0.970	0.95	0.00
59	1.011	1.18	0.01
58	1.080	1.64	0.01
57	1.120	1.88	0.01
56	1.162	2.26	0.01
55	1.219	2.98	0.03

TABLE IV

AVERAGE SPECTRAL DISTORTION AS A FUNCTION OF BITRATE OF THE SPLIT-MULTISTAGE VECTOR QUANTISER WITH MA PREDICTOR IN AMR-WB SPEECH CODEC ON WIDEBAND ISF VECTORS FROM THE TIMIT DATABASE

Bits/frame	Avg. SD (dB)	Outliers (in %) 2–4 dB	Outliers (in %) > 4 dB
46	0.894	0.76	0.01
36	1.304	5.94	0.03

the better rate-distortion and computational efficiency of the SSVQ over the SVQ, due to the former's compensation of the memory and shape advantage losses of the latter.

Table III shows the spectral distortion performance of using PDF-optimised scalar quantisers, where can see that 59 bits/frame are required for transparent coding. Therefore, we can conclude that the SSVQ requires 17 bits/frame less than scalar quantisers, due to the former's memory and shape advantages.

Finally, Table IV shows the spectral distortion performance of S-MSVQ with MA prediction from the ITU-T G.722.2 AMR-WB speech codec. We can see that the SSVQ achieves comparable spectral distortion at 46 bits/frame. Note that since SSVQ is a memoryless scheme (ie. it does not exploit interframe correlation), then we may conclude that SSVQ is a better scheme than S-MSVQ[4].

VIII. CONCLUSION

In this paper, we have presented an overview of the switched split vector quantiser and applied it to quantising line spectral frequencies derived from wideband speech. Moreover, we have shown, using the vector quantiser advantages, how SSVQ improves upon SVQ and demonstrated this via simple quantisation experiments. In our wideband LSF quantisation experiments, the SSVQ was shown to be better than SVQ and PDF-optimised scalar quantisers by 4 bits/frame and 17 bits/frame, respectively. SSVQ was also found to be comparable to the S-MSVQ with MA predictor at 46 bits/frame, though the former is a memoryless scheme.

REFERENCES

[1] C.E. Shannon, "Coding theorems for a discrete source with a fidelity criterion", *IRE National Convention Record*, part 4, pp. 142-163, 1959.

[2] K.K. Paliwal and W.B. Kleijn, "Quantization of LPC parameters" in *Speech Coding and Synthesis*, W.B. Kleijn and K.K. Paliwal, Ed. Amsterdam: Elsevier, 1995, pp. 443–466.

[3] P. Hedelin and J. Skoglund, "Vector quantization based on Gaussian mixture models", *IEEE Trans. Speech Audio Processing*, vol. 8, no. 4, pp. 385–401, July 2000.

[4] S. So and K.K. Paliwal, "A comparative study of LPC parameter representations and quantisation schemes in wideband speech coding", submitted to *Digital Signal Processing*, May 2005.

[5] K.K. Paliwal and B.S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame", *IEEE Trans. Speech Audio Processing*, vol. 1, no. 1, pp. 3–14, Jan. 1993.

[6] W.P. LeBlanc, B. Bhattacharya, S.A. Mahmoud, and V. Cuperman, "Efficient search and design procedures for robust multi-stage VQ for LPC parameters for 4 kb/s speech coding", *IEEE Trans. Speech Audio Processing*, vol. 1, no. 4, pp. 373–385, Oct. 1993.

[7] S. So and K.K. Paliwal, "Efficient product code vector quantisation using the switched split vector quantiser", to appear in *Digital Signal Processing*, 2005.

[8] T.D. Lookabaugh and R.M. Gray, "High-resolution quantization theory and the vector quantizer advantage", *IEEE Trans. Inform. Theory*, vol. 35, no. 5, pp. 1020–1033, Sept 1989.

[9] B. Bessette, R. Salami, R. Lefebvre, M. Jelfnek, J. Rotola-Pukkila, J. Vainio, H. Mikkola, and K. Jarvinen, "The adaptive multirate wideband speech codec (AMR-WB)", *IEEE Trans. Speech Audio Processing*, vol. 10, no. 8, pp. 620–636, Nov. 2002.

[10] Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. Commun.*, vol. COM-28, no. 1, pp. 84–95, Jan. 1980.

[11] F. Itakura, "Line spectrum representation of linear predictive coefficients of speech signals", *J. Acoust. Soc. Amer.*, vol. 57, p. S35, Apr. 1975.

[12] N. Sugamura and F. Itakura, "Speech Analysis and Synthesis Methods Developed at ECL in NTT–From LPC to LSP–", *Speech Commun.*, vol. 5, pp. 199–215, Jun. 1986.

[13] G. Guibé, H.T. How and L. Hanzo, "Speech spectral quantizers for wideband speech coding", *European Transactions on Telecommunications*, 12(6), pp. 535–545, 2001.

[4]Note that the AMR-WB uses immittance spectral frequencies (ISFs), which we have shown in [4] to be inferior to LSFs by approximately 1 bit/frame.