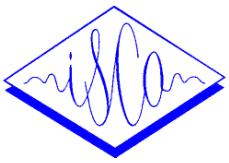
ISCA Archive
http://www.isca-speech.org/archive



3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

EMULATION OF A FORMANT VOCODER AT 600 AND 800 BPS

Nigel Sedgwick

Cambridge Algorithmica Ltd, Royston, Hertfordshire, United Kingdom SG8 0HF

ABSTRACT

A vocoder operating at a variety of low data rates has been implemented by software emulation. It uses Vector Quantisation (VQ), Variable Frame Rate Coding (VFR) with infil by replication and interpolation, and analysis by synthesis with a formant synthesis model. Experiments with a large talker-dependent codebook gives intelligible speech at 800 bps in context, and similar performance is expected with a talker-independent codebook. Intelligible speech at 600 bps is expected to be possible with further work. At 480 bps the speech is too slurred and further work using the same infil methods is not expected to give sufficient improvement.

Keywords: speech coding, vocoder, formant vocoder, very low rate vocoder.

10.21437/Eurospeech.1993-26

1. INTRODUCTION

Vocoders are mainly of interest for military secure speech communications. Vocoders operating at 2400 bits per second (bps) of speech, including those based on linear predictive coding (LPC) eg [1], have been in operational use since the early 1980s. More recently there has been interest in vocoders that operate at data rates of 800 bps and below, for use over frequency-agile HF radios and in poor propagating conditions.

Jaskie and Fette [2] give a comprehensive review of previously proposed technical approaches. The work reported here builds on that undertaken in the late 1970s and early 1980s at the UK Joint Speech Research Unit (JSRU). The JSRU Parallel Formant Synthesiser [3], when driven correctly, is known to give very good quality speech synthesis at 100 frames per second. The control parameters are frequency and amplitude for F1, F2 and F3, amplitude only for F4 (which is approximated by one or two fixed frequency resonators) and a low frequency formant FN (which is usually used with fixed frequency), and pitch and degree of voicing.

Dupree [4] implemented a near optimal method of VFR coding using Dynamic Programming (DP) based on the variable duration segmentation algorithm of Bridle and Sedgwick [5]; he reported improved speech quality from the VFR coding due to smoothing of inconsistent formant jumps from frame to frame; he also proposed, but did not implement, the use of VQ with his VFR scheme for the formant analyser.

The work reported here on is a formant vocoder that uses VQ and VFR coding and is based on analysis by synthesis and is an extension and improvement to that previously reported [6].

2. FORMANT VOCODER ANALYSER

Acoustic Analysis. This is done at a fixed rate of 100 frames per second. Speech is sampled at 10ks/s and analysed spectrally to give energies in 32 mel spectral channels covering the range 150Hz to 4500Hz. A DFT is used with rectangular window over a selected 4ms portion of pre-emphasised speech starting in the frame. The selected window is chosen by an excitation point finding algorithm that operates on the doubly pre-emphasised speech waveform and looks for maxima in the energy integrated over a 4ms window. Spectral analysis is therefore done during the closed glottis period for voiced speech and during higher energy portions for unvoiced speech. Pitch and degree of voicing are found using cepstral analysis over a fixed 50ms window centred on each frame. Currently 2 degrees of voicing are used. An allowance has been made to increase this to 4 levels, to improve vocoding of voiced fricatives.

Codebook Shortlist Lookup. For each input frame, the nearest few entries (typically 14) in the codebook are found using the spectral distance metric.

Dynamic Programming VQ/VFR Selection. Segment-based DP, extended from that of [5], is used to find simultaneously and efficiently the near-optimal set of segment boundaries, segment-final VQ codebook entries and infil method. The partial path score $F(m,k)$ is the error from the beginning of the speech to the end of the current frame k with a segment ending at VQ entry m (selected from only those VQ entries in the shortlist for frame k). It is calculated frame by frame using:

$$F(m,k) = \min_{a,j,t} [F(a,k-j) + smf(a,m,t,k-j+1,k) + traj(a,m,t,j)]$$

The segment matching function $smf(a,m,t,i,k)$ gives the total spectral distance (summed frame by frame across the segment) between the input speech from frame i to frame k and the formant synthesised speech generated

using a segment with infil by method t. Infil is by replication of VQ codebook entry m or linear interpolation between VQ codebook entries a (segment-final VQ entry for the preceding candidate segment) and m. The trajectory penalty $traj(a,m,t,j)$ depends on the infil method t, the formant frequency (but not amplitude) movement across the segment and the segment length j; it penalises hypothesised segments with large formant frequency jumps (for infil by replication) or high gradient formant frequency trajectories (for infil by linear interpolation). The minimisation is performed over all valid segment lengths j and all valid preceding segment-final codebook entries a.

Partial path traceback is used to output a segment every few frames, at the average VFR reduction rate. Where all paths to active nodes do not have a common initial subpath within the DP buffer (typically 50 frames), traceback is forced to chose the best path ending at the current frame; all partial paths that do not start from the same subpath are deleted. The output description of each segment therefore consists of the segment length, segment-final VQ codebook entry, and infil method.

Side Information Coding. For each segment, the degree of voicing is taken to be that of the segment final VQ entry. Pitch is coded with a segment-final value and a contour type. Currently only two contours are used: (i) constant pitch at the segment-final value, or (ii) linear interpolation from the previous segment-final value. Allowance has been made for parabolic contours (n and u with two sizes of excursion about the mean segment-final pitch of the preceding and current segment) and cubic contours (nu and un), making a total of 8 contour types. This is expected to code better the pitch variations on very long segments. Allowance has also been made to normalise the input frame energy level and spectral tilt for the term-averaged values (derived over many syllables); the normalisation value applied to each segment could then be transmitted to allow reconstitution at the vocoder synthesiser of the original energy level and spectral balance; this approach would allow codebooks to be derived without the need to model variations between and within talkers of speaking level (affecting energy) and vocal effort (affecting energy and spectral tilt).

Spectral Distance Metric. The distance between pairs of frames uses a weighted Euclidean distance with dimensions of the 32 mel spectral channels and the degree of voicing. Spectral channel energies for the formant synthesised values are calculated directly from the synthesiser control parameters, and are modified for the effect of the rectangular window used for spectral analysis. The weights for the spectral channels are varied frame by frame according to the spectral channel energies of the natural speech, in order to give more emphasis to spectral differences near the formants. An improved spectral distance metric will be used for future work [7].

3. CODEBOOK GENERATION

The VQ codebook is built from example frames of formant analysed natural speech, using a formant assignment process, similar to that in [4]. Analysis proceeds as follows.

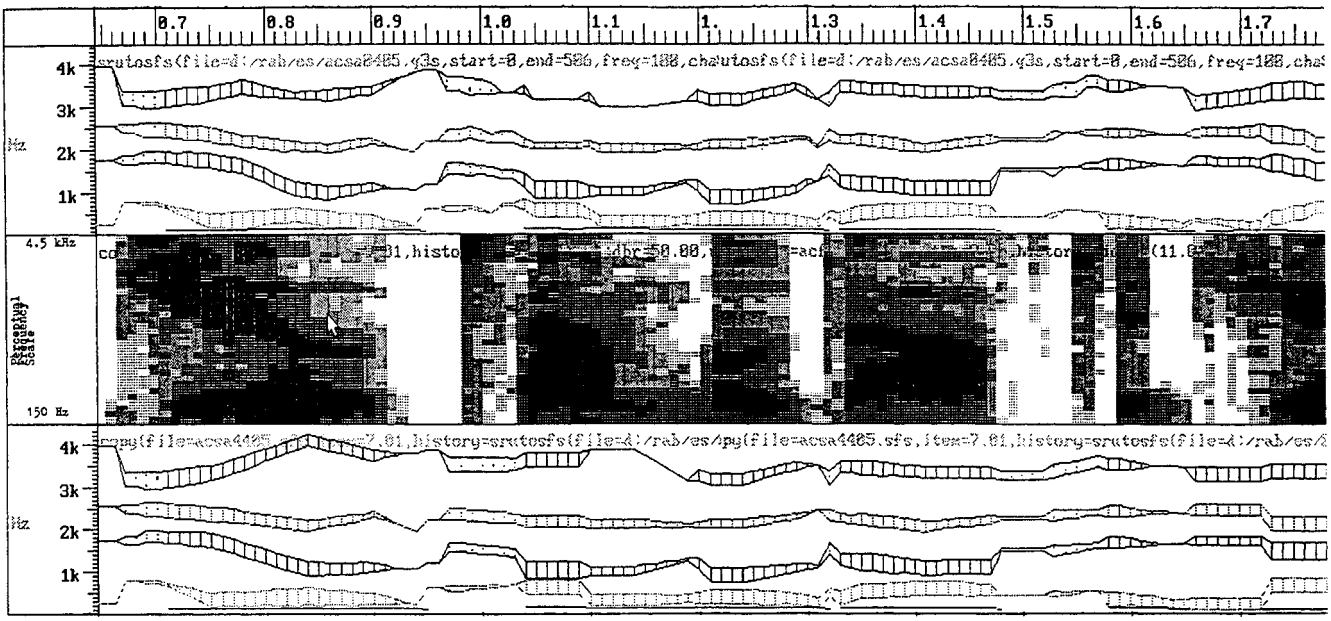
Acoustic Analysis. Spectral analysis and pitch and degree of voicing analysis is made in the same way as for the Formant Vocoder Analyser. In addition, an estimate is made of the frequencies of the resonances of the vocal tract. This is currently done using covariance LPC analysis on the 4ms windowed signal used for the spectral analysis.

Formant Hypotheses. All possible assignments are made of vocal tract resonances to formants, to give a set of hypothesised formant frequencies; constraints are applied on the valid formant frequency ranges and the ordering of formants. For each hypothesis, formant amplitudes are then estimated from the appropriate spectral channel energy. Then iterative refinement of each hypothesis is made by grid search over small perturbations of the formant amplitudes (and optionally frequencies) to minimise the spectral distance between the formant synthesised spectrum and the natural speech spectral analysis. Hypotheses that become very close to others are eliminated. Then the best few refined hypotheses (typically 14) are retained.

Dynamic Programming Smoothing. This is very similar to the DP in the Formant Vocoder Analyser. The only difference is that the Formant Hypotheses are used for each frame as the segment-final candidates, rather then the codebook shortlist. The effect of this is to produce variable duration segments that describe smoothed formant trajectories.

Smoothed, Unsmoothed and Segment-Final Output. Output of formant synthesiser control parameters is made are follows: (i) frame by frame the smoothed values (ie those interpolated from infil between the segment-final values); (ii) frame by frame, the nearest refined hypothesis (in formant space) to the smoothed values; (iii) for each segment, the segment-final value, to allow codebook generation only from these values. These outputs are used for codebook generation. Note, they can also be used as the output of a fixed rate or VFR formant vocoder without VQ.

Codebook Training Data. This consists of the smoothed, unsmoothed or segment-final output for appropriately large quantity of speech, from one or more talkers.



Codebook Initialisation. An initial VQ codebook is formed by clustering the codebook training data into a number of clusters somewhat larger than the required codebook size. This is currently done using the kD-tree algorithm [8].

Codebook Optimisation. Iterative refinement of the current codebook (initialised as above) is done using the LBG algorithm [9]. Note that the codebook size can shrink slightly.

Codebook Size Correction. To allow for codebook size shrinkage during optimisation, one normally starts with a slightly larger codebook than desired. If after shrinkage it is still not the required size, this final stage repeatedly merges the nearest pair of remaining codebook entries until a codebook of exactly the desired size is obtained.

Formant to Spectrum Transformation. As the Formant Vocoder Analyser needs codebook vectors defined in both formant and spectral spaces, this transformation is done vector by vector as part of codebook building.

Formant Distance Metric. This is calculated as the weighted Euclidean distance between two frame descriptions of formant synthesiser control parameters, including the voicing decision.

For codebook initialisation and optimisation, distances are calculated in formant space using the formant distance metric. Codebook vectors can be set to the mean of all input speech frames assigned to a particular vector, or can be the speech vector nearest to the mean.

4. FORMANT VOCODER SYNTHESISER

There are two obvious stages.

VQ/VFR Expansion. Each segment is expanded according to the appropriate infil method, segment length and segment-final VQ entry to produce a sequence of frames of control parameters for the JSRU Parallel Formant Synthesiser.

Parallel Formant Synthesis. This is done in real-time using a commercially available hardware implementation of the JSRU Parallel Formant Synthesiser, to produce an analogue speech signal. Note that, although the frequency of F4 is found by the analysis, synthesis is constrained to use a fixed frequency by the hardware implementation.

5. EXPERIMENTAL RESULTS

A VQ codebook was built from approximately 50,000 frames (about 8.3 minutes) of speech from a single male talker. Unsmoothed output of the formant assignment process was used, with smoothing from a VFR reduction of 3:1 (ie 33.33 segments per second). During codebook creation, vectors were set to the mean of assigned speech frames. The codebook was approximately 7,700 vectors long, but was assumed to be 8,192 long for assignment of transmitted data bits.

In informal intelligibility tests, meaningful sentences and a 40 second passage of speech were synthesised from the output of the smoothed and unsmoothed formant assignment process at the VFR reduction rate of 3:1. All listeners found all of this speech intelligible.

Speech from the same male talker (not used for codebook generation) was then vocoded using the Formant Vocoder Analyser and Formant Vocoder Synthesiser at a variety of different VFR reductions.

Each segment of vocoded speech was described as follows:

Segment-Final VQ Codebook Entry	13
Segment Length (1..16 frames)	4
Formant Infil Method	1
Segment-Final Pitch (log scale)	5
Pitch Contour	1
TOTAL BITS PER SEGMENT	**24**

Speech was vocoded using different VFR reductions as follows.

Vocoded Rate (bps)	Average Segments per sec	VFR Reduction	DP Optimisation Buffer Length (frames)
480	20	5:1	50
600	25	4:1	48
800	33.33	3:1	48

Various phonetically balanced and phonetically compact sentences and a 40 second meaningful passage were vocoded. None of this speech was used for codebook generation. Informal listening test with listeners not familiar with the test material indicated the following.

At 480 bps the speech was partially intelligible, but judged too slurred to be of practical use. At 600 bps most of the speech was intelligible in context. At 800 bps all of the speech was intelligible in context, with the exception of one or two words; listeners familiar with vocoders judged the speech quality as operationally usable.

The Figure shows an example of the operation of the Formant Vocoder Analyser on approximately 1.2 seconds of speech. The middle section gives a pseudo spectrogram of the input speech; it displays the 32 mel spectral channel energies. The upper and lower sections shows the control parameters for the JSRU Parallel Formant Synthesiser immediately before resynthesis of the speech waveform. The centres of the 4 bands give the frequencies of F1, F2, F3 and F4 according to the left-hand linear frequency scale. The widths of the bands give the amplitudes of these 4 formants. Line hatching within the bands indicates voiced speech and dotted filling indicates unvoiced speech. The upper section is for speech vocoded at 800 bps and the lower for speech at 600 bps.

6. CONCLUSIONS

A vocoder based on VQ/VFR coding of formant frequencies and amplitudes and analysis by synthesis has been implemented by software emulation. Using a talker-dependent codebook, informal listening tests indicate usable speech quality at 800 bps.

Given the large codebook size (specifically chosen to avoid problems with talker and language dependency), further work is expected to lead to equivalent speech quality with a talker-independent codebook.

At 600 bps the speech quality was marginally below that thought usable. Further work is expected to lead to improvements in speech quality here, perhaps raising it to an operationally useful level.

At 480 bps the speech quality was sufficiently far from usable and so slurred that there is doubt that the technical approach could be improved sufficiently to give operationally useful speech, without increasing the sophistication of the segment infil methods.

ACKNOWLEDGEMENTS

This work was funded by the Speech Research Unit (SRU) of the UK Defence Research Agency under contract number ML12A/1481. Thanks are due to Dr Keith Ponting of SRU for providing improved pitch track and degree of voicing values for all speech signals and for assistance with codebook generation and other computer runs. Thanks are also due to erstwhile colleagues, Peter Stephens and Robin Hernaman, who worked on the development.

REFERENCES

1. NATO Stanag 4198 Edn 1, "Parameters and Coding Characteristics that must be Common to assure Interoperability of 2400 BPS Linear Predictive Encoded Digital Speech", 1984.
2. C Jaskie and B Fette, "A Survey of Low Bit Rate Vocoders", Proc Voice Systems Worldwide 1992, London, UK, February 1992.
3. J M Rye and J N Holmes, "A Versatile Software Parallel-Formant Speech Synthesiser", JSRU Research Report No 1016, 1982.
4. B C Dupree, "Formant Coding of Speech using Dynamic Programming", Electronic Letters, Vol 20, No 7, pp279-280, March 1984.
5. J S Bridle and N C Sedgwick, "A Method of Segmenting Acoustic Patterns, with Applications to Automatic Speech Recognition", Proc IEEE Int Conf on Acoustics Speech and Signal Processing, Hartford USA, 1977.
6. N C Sedgwick, "A Formant Vocoder at 600 bps", IEE Colloquium of Speech Coding - Techniques and Applications, London, UK, April 1992.
7. R A Brierton and N C Sedgwick, "Talker Enrolment for Speech Recognition by Synthesis", Proc ESCA Eurospeech '93 Conf, Berlin, FDR, September 1993.
8. R F Sproull, "Refinements to Nearest-Neighbour Searching in k-Dimensional Trees", Algorithmica, Vol 6, pp579-589, 1991.
9. Y Linde, A Buzo and R M Gray, "An Algorithm for Vector Quantizer Design", IEEE Trans on Communications, Vol COM-28, No 1, January 1980.