0.3dB and the number of processors in the Viterbi decoder is reduced to eight. This is acceptable in practical realisations. We note that the performance is greatly degraded in the case of two-state and four-state decoders.

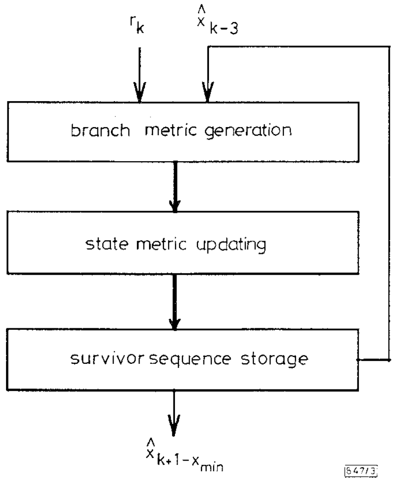

Fig. 3 *Block diagram of eight-state partial erasure decoder*

Table 1: Hardware requirements for EPR4/RSSE BMG modules

	EPR4	RSSE
Number of inputs	1	2
Multipliers	0	2
Adders	0	4
Shift registers	3	3
Mem. locations	0	4
Multiplexers	0	4

Implementation: If we compare the eight-state trellis from Fig. 1 to the EPR4 trellis we note that they are structurally identical. This means that they have the same number of states, the same number of branches, and the same connectivity. This suggests that the eight-state Viterbi decoder can be easily constructed by modifying an EPR4 Viterbi decoder. Basically the only difference between these two decoders is that in the EPR4 Viterbi decoder there is only one branch symbol associated with each branch. In the case of the eight-state Viterbi decoder, there are two possible branch symbols associated with a branch and the selections of branch symbols depend on $\hat{x}_{k-3}$ which is obtained from the path history of each reduced trellis state. This only affects the branch metrics generation (BMG) module. The hardware requirements for the branch metrics generator modules are compared in Table 1. The block diagram of the eight-state partial erasure Viterbi decoder is shown in Fig. 3; r_k denotes the noisy version of y_k and X_{min} denotes the truncation depth of the Viterbi decoder. Note that the two remaining modules (state metrics updating and survivor sequence storage) remain the same as in the EPR4 Viterbi decoder which means that we can use the same processor architectures and the same survivor management techniques as in the EPR4 Viterbi decoder.

 26 March 1996

Electronics Letters Online No: 19960928

R. He, V. Guranovic and J.R. Cruz (*Communications and Signal Processing Laboratory, School of Electrical Engineering, The University of Oklahoma, Norman, OK 73019, USA*)

References

1 LEE, I., YAMAUCHI, T., and CIOFFI, J.M.: 'Modified maximum likelihood sequence estimation in a simple partial erasure model'. Proc. Int. Conf. Commun., May 1994, Vol. 1, pp. 245-249

2 EYUBOGLU, M.V., and QURESHI, S.U.H.: 'Reduced-state sequence estimation with set partitioning and decision feedback', *IEEE Trans.*, 1988, **COM-36**, pp. 13-20

3 YAMAUCHI, T., and CIOFFI, J.M.: 'A nonlinear model in thin film disk recording system'. IEEE Int. Magn. Conf., April 1993, pp. EC-09

Robust speech pulse detection using adaptive noise modelling

N.B. Yoma, F. McInnes and M. Jack

Indexing terms: Speech recognition, Adaptive filters

The problem of speech pulse detection with additive noise at a signal-to-noise ratio (SNR) as low as 0 and –6dB is addressed. The noise is assumed to be reasonably stationary and correlated. Three techniques have been examined: the autoregressive analysis of noise; spectral density comparison; and the non-stationarity measure.

Introduction: The inaccurate detection of the endpoints is a major cause of errors in automatic speech recognition systems. Most of the endpoint detecting techniques are based on energy levels, pitch, zero- and/or level-crossing rates, and timing [1]. However, in many real environments the speech signal is corrnpted by additive noise and these parameters may be insufficient for the correct detection of a speech pulse if the signal-to-noise ratio (SNR) is low.

The contributions of this Letter concerns: (i) adaptive autoregressive modelling of noise in order to reduce the influence of the corrupting signal; and speech pulse detection aided by (ii) spectral density comparison between noise and noisy speech signals or (iii) non-stationarity measures.

The FIR filters used in the autoregressive analysis are trained with the LMS algorithm during non-speech intervals. The spectral density comparison is made between noisy speech frames and an estimation of noise in non-speech intervals. In contrast, non-stationarity measures are based on spectral distances between contiguous frames and do not require noise estimation. Preliminary experiments have shown that the AR analysis generally increases the discrimination between speech and noise, and that spectral density comparison and non-stationarity measures might be more effective than energy in indicating the presence of a speech pulse at low SNRs.

AR analysis of the noise signal: It is assumed that the noise $n(i)$ could be described by an AR process of order M, i.e. it would satisfy the following equation [2]:

$$H_A(z)N(z) = W(z) \qquad (1)$$

where $N(z)$ and $W(z)$ are the z transform of the noise and a white noise process, respectively, and $H_A(z)$ is defined as

$$H_A(Z) = 1 + \sum_{k=1}^{M} a_k z^{-k} \qquad (2)$$

If the noise is reasonably stationary, its autoregressive filter $H_A(Z)$ estimated in non-speech intervals may be used to increase the energy gap between the noise and the noisy speech signals. Since the speech signal is intrinsically non-stationary and has components in all the considered band (250–3200 Hz), its spectral density and that of the noise are likely to differ along time, even if the noise is correlated and mainly concentrated in low frequencies (below 1000Hz). Consequently, it is expected that the attenuation caused by $H_A(Z)$ will be lower on average for the speech than for the corrupting signal. The filter $H_A(Z)$ is transversal or FIR, and its coefficients can be estimated using the classical LMS algorithm. If the coefficients a_i are replaced with c_i, where $c_i = -a_i$, the tap weights adaptation is given by

$$c_k(i+1) = c_k(i) + \eta n(i-k)e(i) \qquad (3)$$

where η is the learning rate and $e(i)$ corresponds to the prediction error:

$$e(i) = n(i) - \sum_{k=1}^{M} c_k n(i-k) \qquad (4)$$

Spectral density comparison: If the noise is assumed to be reasonably stationary, the noise spectral density could be considered valid between two consecutive silence periods and could be useful in detecting speech pulses. In the results presented in this Letter, the spectral estimation was made with a 14 channel Mel-filter bank, the same used in recognition experiments [4], but neither logarithmic compression nor normalisation was applied. The spectral density comparison coefficient ($SD(i)$) for a frame i is defined, in the Euclidean metric context, as

$$SD(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_k^n - E_{i,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \qquad (5)$$

where $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ and $S^n = (E_1^n, E_2^n, E_3^n, ..., E_{14}^n)$ correspond, respectively, to the spectral estimation of frame i and that of the noise; E_k^n and $E_{i,k}$ represent the filter k output energies. The noise spectral estimation was computed as the average spectrum in 10 non-speech frames.

Stationarity coefficient: If the noise is reasonably stationary its statistical properties are constant or change slowly, or might even present fast but small variations along time. To use these features of the corrupting signal in speech pulse detection, the non-stationarity coefficient ($NST(i)$) for a frame i is defined, in the Euclidean metric context, as

$$NST(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_{i,k} - E_{i-1,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \qquad (6)$$

where $S_{i-1} = (E_{i-1,1}, E_{i-1,2}, E_{i-1,3}, ..., E_{i-1,14})$ and $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ correspond, respectively, to the spectral estimations of two contiguous frames.

Results: The experiments were carried out using the Noisex-92 database [3]. The signals were lowpass filtered using a 10th order Tchebychev filter with a cutoff frequency of 3700 Hz, downsampled from 16000 to 8000 sample/s, and highpass filtered using a fourth order Tchebychev filter with a cutoff frequency of 120Hz and a minimum attenuation equal to 25dB. The data signal was divided in 25ms frames without overlapping. Each frame was processed with a Hamming window before the frame energy and spectral estimation being computed.

Three noises from Noisex-92 were considered (car, speech and Lynx) and for each case one AR FIR filter was trained using the noise-only samples files and the LMS algorithm. The learning rate was made equal to $0.1/(M \times noise\_power)$ and the LMS algorithm was active for 10 training frames (250 ms). The FIR taps were set to 0 at the beginning of the iterative procedure. To determine the optimum prediction order, several configurations were tested and the one that gave the lowest prediction error was chosen. The clean and noisy speech signals belonged to the male speaker from the Noisex-92 database. Table 1 shows the optimum number of taps for each filter and the ratio G between the attenuation gain on clean speech signals and the attenuation gain on the training noise signal after the AR FIR filter being estimated. This quotient G gives an idea of the energy gap increase between noise and speech due to the AR FIR filter. The clean signals corresponded to 10 utterances (one per digit) automatically end detected.

Table 1: Optimum AR FIR order and quotient (G) between the energy attenuation gains on clean speech signal and training noise

Noise	Car	Speech noise	Lynx
Optimum FIR order	2	2	4
G (dB)	13.1	6.6	5.3

Figs. 1 and 2 present the power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The power envelope corresponds to the difference between the mean frame energy (dB) and the mean noise energy estimation (dB) made in 10 non-speech frames. The utterance corresponds to the digit 'one' in the car noise, with SNR equal to 0 and –6dB, respectively. The word 'one' was chosen because it presents a signal mainly concentrated in low frequencies, and constituted a more challenging problem than, e.g. the digit 'six'.

Discussion: As can be seen in Table 1, the AR analysis led to a higher attenuation on average for the noises than for speech signals. According to Figs. 1 and 2, the AR FIR filters increased the discrimination between the speech signal and backgound noise in the power, spectral comparison and non-stationarity coefficient domains. When compared with the power envelope, spectral comparison and non-stationarity coefficients slightly increased the difference between speech and non-speech pulses before FIR processing, but gave similar results after FIR at SNR equal to 0dB (Fig. 1). According to Fig. 2 (SNR = –6dB), these coefficients increased the difference between speech and non-speech pulses after FIR processing, but the improvement achieved before AR analysis was not enough to highlight the speech signal from the backgound.

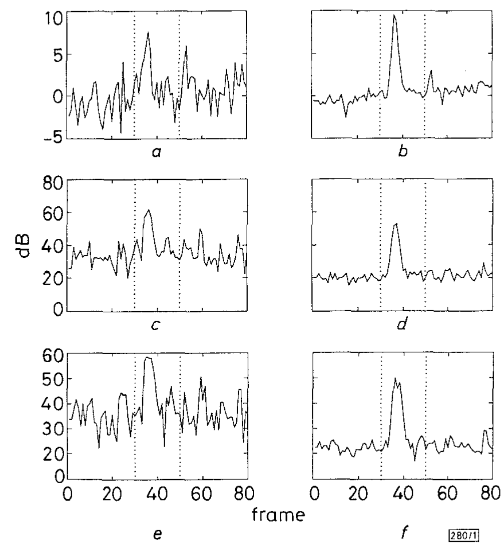

Fig. 1 *Power envelope spectral comparison and non-stationary coefficients before and after processing the signal with the AR FIR filter*

The utterance corresponds to the digit 'one' in car noise with SNR equal to 0dB
Dotted vertical lines: endpoints of speech signal
a Power envelope before FIR *b* Power envelope after FIR
c Spectral comp. before FIR *d* Spectral comp. after FIR
e Non-station. coeff. before FIR *f* Non-station. coeff. after FIR

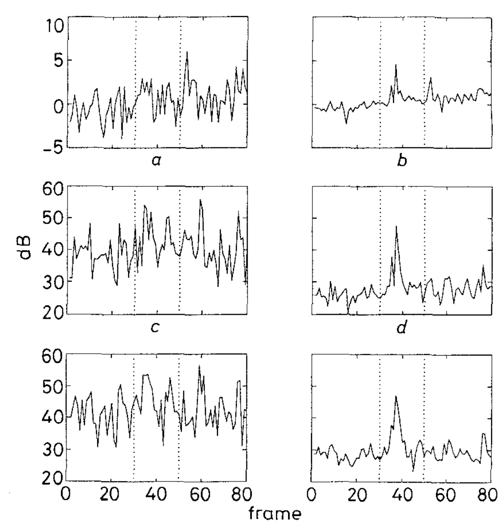

Fig. 2 *Power envelope spectral comparison and non-stationary coefficients before and after processing the signal with the AR FIR filter*

The utterance corresponds to the digit 'one' in car noise with SNR equal to –6dB
Dotted vertical lines: endpoints of speech signal
a–f As for Fig. 1

Concluding, the observed improvements were mainly due to the AR analysis, although spectral comparison and the non-stationarity coefficient might be useful at low SNRs. The AR FIR filters needed a low number of taps, the LMS algorithm seems to be fast enough to capture slow variations of the noise characteristics and only one microphone is necessary. Moreover, the AR analysis might be used by noise cancelling techniques in speech recognition. Future work includes some heuristics to develop an endpoint detector, automatic threshold estimation, and the study of AR adaptation techniques.

Acknowledgments: N.B. Yoma is supported by a grant from CNPq-Brasilia/Brasil.

7 May 1996
Electronics Letters Online No: 19960892

N.B. Yoma, F. McInnes and M. Jack (*Centre for Communication Interface Research, University of Edinburgh, 80 South Bridge, Edinburgh EH1 1HN, United Kingdom*)

References

1 JUNQUA, J.C., REAVES, B., and MAK, B.: 'A study of endpoint detection algorithms in adverse conditions: Incidence on a DTW and HMM recognizer'. Eurospeech'91, 1991, pp. 1371–1374
2 HAYKIN, S.: 'Adaptive filter theory' (Prentice Hall, Englewood Cliffs, NJ, 1991), 2nd edn.
3 VARGA, A., STEENEKEN, H.J.M., TOMLINSON, M., and JONES, D.: 'The Noisex-92 study on the effect of additive noise in automatic speech recognition'. Technical report, DRA Speech Research Unit, U.K., 1992
4 YOMA, N.B., MCINNES, F.R., and JACK, M.A.: 'Improved algorithms for speech recognition in noise using lateral inhibition and SNR weighting'. Eurospeech'95, 1995, pp. 461–464

Delay analysis of multiserver ATM buffers

B. Vinck and H. Bruneel

Indexing terms: Asynchronous transfer mode, Queueing theory

A discrete-time multiple-server queue with FIFO discipline and deterministic service times of one slot each is studied. A relationship between the mass-functions of the delay of an arbitrary customer, and the system contents, during an arbitrary slot is derived under the most general conditions possible.

Introduction: In [1] a general relationship between the probability distributions of the delay of an arbitrary customer, and the buffer contents, during an arbitrary slot was derived for a stable discrete-time single-server queue with deterministic service times equal to one slot. That relationship was 'general' in the sense that no modelling assumptions (concerning the arrival process), other than those necessary to guarantee the existence of the quantities appearing in the relationship, are required.

We extend that relationship to the case of a multiple-server queue with deterministic service times.

The result we obtain has been observed to hold under various more restrictive conditions, none of which is it necessary to impose when following the proof presented in this letter. In [2] a rather general proof of the result is given, but the authors require the system to reach a slot-wise limit distribution, thereby excluding all arrival processes that lead to a different limit behaviour. We also notice that our approach is mathematically much less complex than their approach.

Model and result: We consider a discrete-time queueing system with c servers and a deterministic service time of one slot for each customer. Owing to the discrete-time operation, time appears as being divided into fixed length slots $S_k = (t_k, t_{k+1})$; all customer arrivals and departures occur at the slot boundaries. Customers arrive according to a general arrival process and are queued for service according to a FIFO service discipline. The number of customers arriving at t_k is denoted as A_k. The buffer contents, i.e. the number of customers present, during slot S_k is denoted as U_k, and the delay of the jth customer C_j denoted as D_j.

For any sample path of the queueing process, the average arrival rate λ is defined as:

$$\lambda := \lim_{k\to\infty} \frac{1}{k}\sum_{i=1}^{k} A_i \qquad (1)$$

assuming the limit exists. Likewise, using the notation # to indicate the number of elements in a set, the distribution of the buffer contents U during an arbitrary slot can be characterised as:

$$\Pr[U = n] := \lim_{k\to\infty} \frac{\#\{i|U_i = n, 1 \le i \le k\}}{k} \qquad n \ge 0$$

and the delay distribution for an arbitrary customer is given by:

$$\Pr[D = n] := \lim_{l\to\infty} \frac{\#\{C_j|D_j = n, 1 \le j \le l\}}{l} \qquad n \ge 1$$

again, under the assumption that these limits exist. A sufficient condition for this is that the limit λ in eqn. 1 exists and be strictly less than c.

In this Letter we prove this regardless of the precise nature of the arrival process:

$$\Pr[D = n] = \frac{1}{\lambda}\sum_{p=-c+1}^{c-1} (c - |p|)\Pr[U = cn + p] \qquad n = 1, 2, \ldots \quad (2)$$

Proof: We can make use of the result obtained in [1] for the single-server queue with deterministic service times when we divide each slot in c mini-slots of equal length and consider an equivalent single-server queue that serves one customer per mini-slot and receives the same arrivals as the original queue. In the equivalent queue, customers are also served according to a FIFO discipline. The boundaries of the mini-slots are denoted as $\hat{t}_k$, whereby $\hat{t}_{kc} = t_k$, for any $k \in \mathbf{N}$, see Fig. 1.

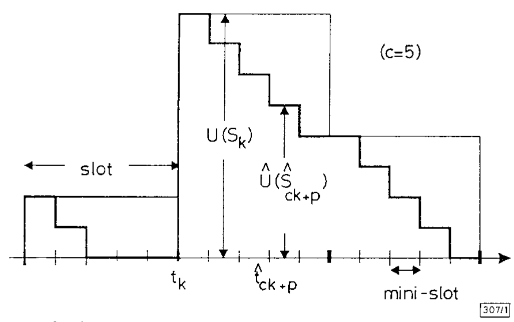

Fig. 1 *Multiple server queue*

In the equivalent single-server queue the number of 'time-units' is raised by a factor c, so that the load is $\lambda/c < 1$. From [1] we can copy the result for the 'equivalent' single-server queue:

$$\Pr[\hat{D} = n] = \frac{c}{\lambda}\Pr[\hat{U} = n] \qquad n \ge 1 \qquad (3)$$

where $\hat{U}$ is the system contents during an arbitrary mini-slot and $\hat{D}$ is the delay - expressed in a number of mini-slots - of an arbitrary customer in the equivalent single-server queue.

We now translate eqn. 5 into a relationship between the mass-functions of the stochastic variables related to the actual multiple-server queue.

First, we observe that during each slot the multiserver queue and the equivalent single-server queue serve the same customers. A customer that leaves the multiserver queue at t_{k+1} leaves the equivalent single-server queue at $\hat{t}_{kc+1}$, $\hat{t}_{kc+2}$, ... or $\hat{t}_{(k+1)c} = \hat{t}_{k+1}$, while the arrival instants are the same in both queues. Therefore,

$$D_j = n \Leftrightarrow \hat{D}_j \in \{cn, cn-1, \ldots, cn-c+1\}$$

so that for any $l \in \mathbf{B}$:

$$\#\{C_j|D_j = n, 1 \le j \le l\} = \sum_{p=0}^{c-1} \#\{C_j|\hat{D}_j = cn - p, 1 \le j \le l\}$$