ANALYSIS AND COMPENSATION OF PACKET LOSS IN DISTRIBUTED SPEECH RECOGNITION USING INTERLEAVING

B.P. Milner and A.B. James

School of Computing Sciences, University of East Anglia, Norwich, UK
{b.milner, a.james}@uea.ac.uk

ABSTRACT
The aim of this work is to improve the robustness of speech recognition systems operating in burst-like packet loss. First a set of highly artificial packet loss profiles are used to analyse their effect on both recognition performance and on the underlying feature vector stream. This indicates that the simple technique of vector repetition can make the recogniser robust to high percentages of packet loss, providing burst lengths are reasonably short. This leads to the proposal of interleaving the feature vector sequence, prior to packetisation, to disperse bursts of packet loss throughout the feature vector stream.

Recognition results on the Aurora connected digits database show considerable accuracy gains across a range of packet losses and burst lengths. For example at a packet loss rate of 50% with an average burst length of 4 packets (corresponding to 8 static vectors) performance is increased from 49.4% to 88.5% with an increase in delay of 90ms.

1. INTRODUCTION
The increasing trend of using mobile and handheld devices for speech communication has resulted in distributed speech recognition (DSR) systems being developed. The European Telecommunication Standards Institute (ETSI) Aurora DSR standard [1] offers good robustness to noise by replacing the low bit-rate speech codec on the terminal device with the static MFCC feature extraction component of the speech recogniser. Vector quantisation of the resulting stream leads to a bit rate of 4800bps. Further revisions of the standard incorporate noise compensation on the terminal device which gives enhanced performance in noisy environments. However, it is often the case that DSR systems transmit speech data across networks which do not guarantee reliable delivery (e.g. IP and mobile networks). In this case packet loss may occur which results in the loss of valuable static feature vectors. The situation is made worse by the fact that packet loss typically occurs in bursts meaning that long periods of speech data may be lost.

Early work on packet loss compensation for speech recognition [2] used interpolation methods to estimate feature vectors lost as a result of packet loss. These techniques gave good recovery for short duration bursts (<4 vectors) but for longer bursts performance gains reduced. Other work has considered protecting the underlying bit stream of the feature vectors using Reed-Solomon coding [3]. Interpolation was then used to correct any remaining frame errors. An extension to this was proposed in [4][5] where the Viterbi decoding algorithm was modified such that contributions made by observation probabilities associated with feature vectors estimated in times of packet loss were reduced.

This work considers the effect of an unreliable channel at the packet level; the packet either arrives or is lost. The aim of this work is twofold; first to analyse the effect of packet loss on the feature vector stream and its subsequent effect on recognition performance. Secondly to use the result of this analysis to improve the performance of speech recognisers operating in more burst-like packet loss. Section 2 describes the analysis which uses highly structured, but artificial, packet loss profiles to develop an understanding of the processes occurring. This leads to the proposal of incorporating interleaving into the DSR system and this is described in section 3. Section 4 presents a set of experimental results, now using realistic packet loss profiles. Finally a conclusion is made in section 5.

2. ANALYSIS OF THE EFFECT OF PACKET LOSS
Before proceeding with a formal strategy for packet loss compensation it is useful to gain an understanding of the effect packet loss has on the stream of feature vectors and see how this effects recognition performance. To achieve this a set of highly artificial packet loss characteristics has been used to corrupt the static feature vector stream.

In this analysis the effect of packet loss is simulated by allowing a single static feature vector to be received followed by the loss of the next $\beta$ vectors, where $\beta$ is the burst length which remains constant throughout the test. For example with a burst length of $\beta=4$ and no compensation, the following sequence of static vectors is delivered to the recogniser $\{x_1, x_6, x_{11}, x_{16}, \ldots \}$ which corresponds to an 80% loss of vectors. To compensate for packet loss the simple method of static vector repetition is employed to restore the static feature vector stream. Therefore with $\beta=4$, the restored feature vector stream becomes $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, \ldots \}$. Temporal derivatives are calculated from the restored static feature vectors. It is important to note that these loss characteristics are highly artificial but form a useful and constrained method of analysing the effect of packet loss. Formal testing in section 4 uses realistic packet loss profiles.

2.1. Effect of Packet Loss on Recognition Accuracy
To examine the effect of this packet loss on recognition accuracy the Aurora connected digit database was used [1] (configuration details are given in section 4). Figure 1 shows the effect of burst length, $\beta$, on digit accuracy for both the uncompensated and compensated (vector repetition) systems.

![Figure 1: Accuracy under controlled packet loss conditions](image-url)

For the system with no compensation, recognition performance falls drastically – from a no loss figure of 98.5%
to 63.7% with $\beta=1$ and to 23.2% with $\beta=2$. The system employing repetition performs remarkably better. No noticeable reduction in accuracy occurs until the burst length exceeds $\beta=4$ vectors – this is equivalent to 80% of vectors being lost. Even at a burst length of $\beta=10$ vectors (equivalent to 91% of vectors being lost) digit accuracy is over 80%. To gain further insight into this result it is useful to examine the effect packet loss has on the individual static, velocity and acceleration components of the feature vector stream.

2.2. Effect of Packet Loss on the Temporal Components of the Feature Vector Stream

Figure 2-a shows 50 frames of static MFCC(1) extracted from a connected digit string comprising the digits “seven-oh”. Two bursts of packet loss have been introduced; a single vector loss at frame 11 and a burst of 8 vectors lost starting at frame 31. The solid line shows the original loss-free static feature vector component. The dashed line shows the corresponding feature vector stream with repetition used to estimate lost vectors. The step-like structure introduced by repetition is clearly evident.

Figure 2-b illustrates the resultant velocity component of MFCC(1) computed from both the original and compensated static feature vector streams. The plot shows that the single repeated static vector (occurring at frame 11) distorts the velocity component from frame 8 to frame 14, or $+/-V$ frames either side of the loss, where $V=2*1$ is the window width used to compute the velocity component (in this case $V=3$). In general, the total number of velocity vectors affected by packet loss is $\beta+2V$, although for a single loss the velocity measurement at the point of loss is not distorted. For the longer duration loss starting at frame 31, the distortion of the velocity component is considerably more severe – both in its duration and amplitude. The larger number of static vectors in error results in 14 velocity vectors being distorted. When the number of static vectors duplicated exceeds the width of the velocity window the velocity measurements at the centre of the burst become zero. This occurs at frames 33, 34 and 35. Figure 2-c shows the acceleration component computed from velocity using a window width of 5 frames. The distortion of the acceleration component is extended to $\beta+2(V+A)$ vectors, where $A=2+1$ is the window width used to compute the acceleration component (in this case $A=2$). For the single packet loss at frame 11 the distortion effects 11 acceleration measurements and for the burst of 8 frames, 18 acceleration measurements are corrupted. This corresponds to a width of $+/- (V+A)$ frames either side of the loss.

To gain more insight into the effect of this distortion on recognition performance three further recognition tests have been employed using the static, velocity and acceleration components alone. The artificial packet loss characteristics of section 2.1 have been used and affect the static measurements only after which repetition is used to estimate lost vectors. Velocity and acceleration are computed from the resulting static vector stream. Figure 3 shows recognition performance using repetition to estimate lost static vectors. For comparison the performance of the full (39-D) system is shown.

Figure 3: Recognition accuracy using the temporal derivatives

With no packet loss the baseline performance of the four configurations is full 98.5%, static 94.8%, velocity 95.3% and acceleration 92.5%. Recognition accuracy of the static-only system is considerably higher than both the velocity-only and acceleration-only configurations. This is more noticeable at longer burst lengths where the effect of static vector repetition severely corrupts both the velocity and acceleration measurements. The performance of the acceleration-only configuration begins to deteriorate at burst lengths of 3 frames, while the velocity-only configuration falls at burst lengths of 5 frames. This is to be expected as the effect of packet loss is spread wider for acceleration and velocity measurements. Performance of the full configuration gives highest performance up to burst lengths of 9 frames. Beyond this the distortion of the velocity and acceleration components brings down the performance of the full system to below that of the static-only system.

The robust performance of the static-only configuration can be explained by considering the modelling properties of the HMM. Within a state of the HMM the feature vectors are assumed to be independent and identically distributed. However the feature vector stream does posses temporal information (from both the underlying speech production mechanism and overlapping nature of feature extraction) but the HMM makes no use of this correlation and it makes little difference whether the feature vector stream consists of a series of duplicate frames or a natural progression through the feature space within the state. Both give similar observation probabilities. Figure 4 illustrates a 2-D feature space of an HMM state and shows a smooth trajectory path as a solid line.

Figure 4: Static feature vector path through a state of an HMM
The dotted line shows a similar static feature vector sequence where repetition has compensated for packet loss. In terms of the observation probability, the repetition of a single point vector does not significantly corrupt the measurement. For longer duration repetition (over state and model boundaries) observation probabilities will be severely corrupted.

The deterioration in performance of the velocity-only and acceleration-only systems comes from the more severe mismatch in the state-space introduced by packet loss and its subsequent correction. Short duration losses cause a minor distortion of the velocity and acceleration vector streams from their correct values. Longer duration losses cause much more severe distortions – consider the 8 frame loss shown in figure 2 where the velocity component actually takes on a zero amplitude value. This makes a good match with the state-space of the correct model more unlikely.

In fact, an examination of the static vector state sequence showed only minor corruption. However for the velocity and acceleration components the disruption to the static sequence was much more severe.

3. INTERLEAVING FOR PACKET LOSS COMPENSATION

The analysis in the previous section has shown that a speech recogniser can tolerate very high percentages of lost vectors (~80%), provided that the average burst length is reasonably short (<5 vectors). In effect this results in a series of target vectors from which the missing feature vectors can be estimated using repetition or interpolation techniques.

In reality the occurrence of packet loss is burst like and does not occur uniformly. However it is possible to distribute long bursts of packet loss into series of shorter bursts by incorporating interleaving at the sender [6]. Interleaving operates by re-ordering the sequence of feature vectors being packetised such that consecutive vectors are not contained within the same or neighbouring packets. This means that a burst of packet loss does not result in the loss of a sequence of consecutive feature vectors. Instead the loss is spread out in the feature vector stream. Figure 5 shows the proposed DSR architecture which includes interleaving prior to packetisation.

A block interleaver [6] of depth \(d\) is proposed, whereby a sequence of \(d+1\) vectors is arranged in a block and the interleaved sequence found by transposing the traversal order within that block. Initial experiments have used a block interleaver of depth 4 as is illustrated in figure 6. Each packet transports two feature vectors, taken from different sections of the feature vector stream. Figure 6 also illustrates how feature vectors lost within a burst of 5 packets (10 vectors) are distributed to give a maximum gap in the feature vector stream of either 2 or 3 vectors. Careful selection of the interleaving depth should ensure that sufficient target vectors exist to enable good estimation of those missing feature vectors.

![Figure 6: Interleaving of feature vectors prior to packetisation using a block interleaver of depth 4.](image)

Accuracy gains made by interleaving are offset against the increased delay that re-ordering of the feature vector stream introduces. The delay introduced by a block interleaver of depth \(d\) is \((d−1)^2\) vectors. Therefore, for the configuration in figure 6 the maximum delay is 9 vectors or 90ms (with a 10ms frame rate). For speech recognition applications the decoding time of the recogniser and typical nature of applications means that an additional hundred milliseconds of delay should not be too significant to the overall quality of service. However, this is not true for real-time communication, such as VoIP, where an additional delay of a few hundred milliseconds can cause a significant reduction in quality.

4. EXPERIMENTAL RESULTS

The experiments in this section examine the effect interleaving has on recognition accuracy in burst-like packet loss. The tests used the ETSI Aurora connected digits database [1]. Static feature vectors were extracted every 10ms, interleaved (using an interleaving depth of 4) and then packetised – as in figures 5 and 6. For efficiency and consistency with ETSI Aurora recommendations, each packet carried 2 static vectors. A packet number was also added to enable packet loss detection [2]. Upon receipt of the packets, repetition of the preceding vector was used to restore the static feature vector stream in the event of packet loss. Temporal derivatives were computed from the resulting sequence of static feature vectors.

For testing, realistic packet loss profiles were used generated by a 3-state Markov chain [2]. These were produced at rates of 10%, 30% and 50% packet loss with average burst lengths of 1, 4, 8, 12 and 20 packets.

4.1. Interleaving Results

For each packet loss characteristic three DSR configurations were compared; no compensation (splicing), repetition, and interleaving (using a depth of 4, as shown in figure 6) followed by repetition. Figures 7, 8 and 9 show recognition accuracy at 10%, 30% and 50% packet loss with varying average burst lengths - note different vertical scales are used for clarity.
Figure 7: Recognition accuracy at a packet loss rate of 10%.

Figure 8: Recognition accuracy at a packet loss rate of 30%.

Figure 9: Recognition accuracy at a packet loss rate of 50%.

The results show an overall decrease in recognition accuracy as both the percentage of packets lost and the average burst length increases. It is interesting to observe that as the average burst length increases, the overall digit accuracy converges to

\[(1 - \text{proportion of packets lost}) \times \text{baseline recognition accuracy}\]

This is to be expected as at longer burst lengths entire digits will be lost. The rate of convergence to this figure also increases as the overall percentage of packet loss increases.

The figures show that the simple technique of repetition gives good performance improvements for short burst lengths. However as burst length increases, the gains made by repetition begin to reduce. Interleaving the feature vectors prior to transmission results in a significant increase in accuracy – improving performance by 10% at 50% packet loss. This is due to the bursts of loss being distributed which reduces the length of gaps in the static feature vector stream. As burst length increases still further, accuracy reduces as burst lengths begin to exceed the depth of the interleaver.

One anomaly in the results occurs at the 50% loss rate at an average burst length of 1 packet. The interleaved result is significantly lower than the non-interleaved result. This is because the packet loss profile is essentially 1 packet received-1 packet lost. Applying interleaving to this creates a more burst-like packet loss, hence the reduction in accuracy. This is a well-known artefact of interleaving.

4.2. Optimising the Interleaving Depth

As noted in section 4.1, the gains made by interleaving are reduced when burst lengths exceed the interleaving depth. This suggests that increasing the interleaving depth will have a positive effect for longer burst lengths. However, this is at the expense of increased delay, due to the additional buffering needed.

Figure 10 shows the effect of increasing the interleaving depth for vectors being transmitted over a channel with 50% packet loss and an average burst length of 4 packets (8 vectors).

Figure 10: Accuracy using varying interleaving length.

The results show that increasing the interleaving depth aids recognition performance. Indeed, for the above channel, recognition accuracy reaches a maximum when the interleaving depth is 7 vectors or greater. However, an interleaving depth of 7 corresponds to a 7×7 block (49 vectors) with a resultant buffering delay of 36 vectors (360 ms).

5. CONCLUSION

Using some constrained packet loss profiles this work has shown that the simple technique of repetition can make the recogniser robust to very high rates of packet loss (~80%) provided burst lengths are reasonably short (<5 vectors). However, for realistic packet loss characteristics, burst lengths may be much longer than this. As a result, the technique of interleaving feature vectors prior to packetisation has been used to disperse the effect of a burst of packet loss throughout the feature vector stream. This has given substantial improvements in recognition accuracy, when used in combination with vector repetition, in the event of packet loss. Interleaving has, however, introduced a delay in the system and this must be considered as a trade-off against robustness to burst-like packet loss.

Future work will consider better methods for estimating the temporal components of the feature vector and more optimised methods of interleaving.

6. REFERENCES