Fully Vector Quantized Neural Network-Based Code-excited Non-linear Predictive Speech Coding

Lizhong Wu & Frank Fallside

CUED/F-INFENG/TR.94

Submitted to IEEE Transactions on Signal Processing.

Cambridge University Engineering Department
Trumpington Street
Cambridge CB2 1PZ
England

March 1992

Email: lzw@eng.cam.ac.uk or fallside@eng.cam.ac.uk

Fully Vector Quantized Neural Network-Based Code-excited Non-linear Predictive Speech Coding

Lizhong Wu & Frank Fallside

Cambridge University Engineering Department
Trumpington Street, Cambridge CB2 1PZ, U.K.

Abstract

Recent studies have shown that non-linear prediction can be implemented with neural networks, and non-linear predictors will on average achieve about $2 - 3dB$ improvement in prediction gain over conventional linear predictors. In this paper, we take the advantage of non-linear prediction with neural network, apply it to predictive speech coding and attempt to improve the speech coding performance. Our studies are concentrated on non-linear prediction with neural network, non-linear predictive vector quantiser, non-linear long term (pitch) speech prediction, the output variance of the non-linear predictive synthesis filter to its input distortion and fully vector quantised code-excited non-linear predictive speech coding. The above studies have resulted in a fully vector quantised CENN [1] -excited non-linear predictive speech coder. Performance evaluations and comparisons with the linear predictive speech coding are made in this paper.

[1] Codebook-excited neural network, see (Wu and Fallside, 1992).

Contents

1 Introduction

Previous speech coding techniques are mostly based on linear prediction. Linear prediction speech coding systems can be described by a set of linear algebra equations, and designed with the optimal solution of these equations (Atal, 1986).

Linear prediction analysis is based on the assumption that the vocal tract can be approximated by a large number of short cylindrical segments, and transmission throughout the vocal tract is lossless (Rabiner and Schafer, 1978). Without an alternative, linear representation of speech signal is a unique choice when non-linear technique has not practically been applicable. Linear simplification will doubtless lead to representational inaccuracy, processing inefficiency and poor coding performance (Wang et al., 1990).

Recent studies have shown that non-linear prediction can be implemented with neural networks and various simulation results have demonstrated their applicability. In speech applications, it has been reported that neural network-based non-linear prediction will achieve an improvement over conventional linear prediction of about $2 - 3dB$ in prediction gain (Tishby, 1990).

In this paper, we take advantage of non-linear prediction with neural networks and apply it to speech coding. Our studies are focused on:

1. a neural network-based non-linear predictive model

   We develop a general neural network model which contains two non-linear hidden units. The outputs of the hidden units are time delayed and fed back to their input terminals. Linear prediction and most of recently studied non-linear prediction models are special cases of this model. A training algorithm is given in the paper. Simulations and performance comparisons are conducted with the Sunspot data set and speech.

2. a non-linear predictive vector quantiser

   When applied to speech coding, the non-linear predictive parameters should be compressed. Instead of directly quantising the weight parameters of each non-linear predictive neural network, we train a finite set of non-linear predictors to form a non-linear predictive vector quantiser. The non-linear prediction of each speech frame is thus encoded by the index of the selected predictor with the least predictive error.

3. non-linear long term speech prediction

   Voiced speech consists of two types of redundant information, one between successive samples and another around adjacent pitch periods. The former redundancy filtering is referred to as short term (formant) prediction, and the latter as long term (pitch) prediction. We demonstrate that not only can long term prediction be carried out with a non-linear model, but also that it outperforms a linear model. A non-linear long term predictive neural network model and the connection methods of the short term and long term predictors are studied in this paper.

4. tolerance of the non-linear predictive synthesis filter to its excitation distortion

   A fully vector quantised predictive speech coder consists of two main parts, a predictive vector quantiser and an excitation vector quantiser. For each frame of speech, an excitation codevector is chosen and applied to exciting the selected predictive synthesis filter to reconstruct the speech. With a linear predictive vector quantiser, the excitation vector quantiser can be formed by vector quantising the predictive residual. However, we find that this quantised residual-excited method is no longer suitable for the non-linear predictive coding due to the poor tolerance of the non-linear predictive synthesis filter to its input distortion.

   Two approaches have been considered for coping with this. One is the modification of the training cost function of the non-linear predictive vector quantiser by the equation of the output variance to the input disturbance derived in the appendix of this paper, so that the non-linear predictive synthesis filter becomes less sensitive to the distortion of its residual excitation inputs. Another approach is, instead of using the quantised residual, directly training the excitation vector quantiser with the analysis-by-synthesis method. We will concentrate on the second approach in this paper and discuss the first one elsewhere.

5. fully vector quantised code-excited non-linear predictive speech coding

   Unlike linear predictive coding, a complete form of the excitation vector quantiser no longer exists in the non-linear predictive coder. Therefore, we implement the excitation vector quantiser with a codebook-excited neural network (CENN) (Wu and Fallside, 1992). The CENN is formed by a multi-layer neural network excited by a Gaussian codebook. The network may be feedforward or recurrent, and maps the Gaussian sequence to any desired output to excite the non-linear predictive synthesis filters. The transform function of the CENN is trained directly to minimise the error between the predictive synthesis filter output and the original encoded speech.

The above studies result in a fully vector quantised CENN-excited non-linear predictive speech coder. Performance evaluations and comparisons are made of predictive gains, distortion-rate curves, SNRs of reconstructed speech, etc.

2 Neural Network-Based Non-linear Prediction

Time series prediction can be defined as: Given p previous observations of $s(t)$, $X = (s(t-1), \ldots, s(t-p))^T$, find out a function $g(\cdot)$ which minimises the predictive residual

$$D = \int \| s(t) - g(X) \|^2 P(X, s) dX ds. \tag{1}$$

The theoretical solution of eqn(1) is a posterior mean estimator:

$$g(X) = \int s(t) P_{s|X}(X, s) ds, \tag{2}$$

where $P_{s|X}(X, s)$ is the density function of the conditional probability of s given X.

With a multi-layer neural network, if the number of the input units is p, and there is only one output unit, the network can be trained as a p^{th}-order predictor. Assume that $F(\Phi, X)$ is the transfer function of the network. The aim of the training is to determine the architecture of the network and adjust its weights, Φ, so that $F(\Phi, X)$ approaches $g(X)$. This is a problem of nonparametric estimation with neural networks. The studies by White (White, 1989), etc, have demonstrated that single hidden layer feedforward networks are capable of learning an arbitrarily accurate approximation to an unknown function, provided that they increase in complexity at an approximate rate proportional to the size of training data.

Neural network-based predictors can be fitted to data without any specific prior assumption about the form of the non-linearity. Their advantages have been reported by a number of researchers, e.g. (Lapedes and Farber, 1987; Tishby, 1990). In this section, we study a general neural predictive model and extend the studies to the non-linear predictive vector quantiser and the long term (pitch) non-linear speech prediction.

2.1 A Neural Model of Non-linear Prediction

A general structure of the neural network-based non-linear predictor is shown in figure 1. It consists of three layers, which respectively contain N_i, N_h and 1 units, with N_i set to the given predictive order. To predict observations with any scale of amplitude, no non-linear activation function is imposed on the output unit. The hidden output is delayed with time unit τ and fed back to the inputs of the hidden units via a weighting matrix W. This predictor can be described by the equations:

$$\begin{aligned} Y(t) &= f\left(WY(t-\tau) + VX(t)\right) \\ z(t) &= UY(t) \end{aligned} \tag{3}$$

where U and V are respectively the weight vector between the output unit and the hidden layer and the weight matrix between the hidden layer and the input layer. $X(t) = (s(t-1), \ldots, s(t-N_i))^T$ is the input vector, $Y(t)$ the hidden vector, and $z(t)$ the output variable. $f(\cdot)$ is a differentiable non-linear function. In this paper, we define $f(\cdot)$ as the commonly used, sigmoid form, and set the time delay τ to one sample period.

In eqn(3), if $W = 0$, the network is of feedforward type. This form of predictor has been widely studied, e.g. (Lapedes and Farber, 1987). Moreover, if $N_h = N_i$, $V = I$, $f(X) = X$, $W = 0$ and $U = \alpha$, then $z(t) = \alpha X(t)$, and the predictor becomes the linear one.

After the architecture and size of the neural network have been decided, U, V and W can be trained. With Φ representing $\{U, V, W\}$,

$$\Delta\Phi = \eta e(t) \nabla_\Phi z(t) \tag{4}$$

where η is the learning rate and $e(t) = s(t) - z(t)$. Back-propagating $\nabla_\Phi z(t)$ to $\nabla_\Phi Y(t)$, we get

$$\begin{aligned} \nabla_U z(t) &= Y(t) \\ \nabla_W z(t) &= U\nabla_W Y(t) \\ \nabla_V z(t) &= U\nabla_V Y(t). \end{aligned} \tag{5}$$

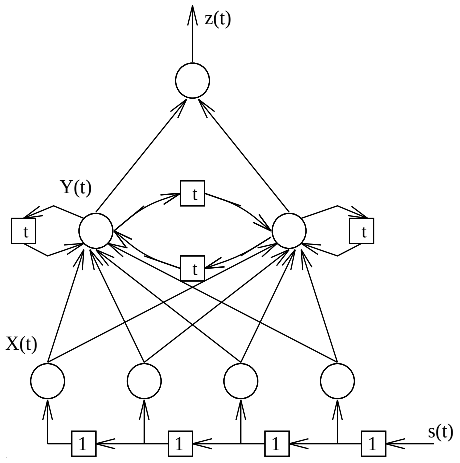

Figure 1: Neural network-based non-linear predictor. The small squares stand for time delay and the numbers inside the squares represent their time delay units.

By assigning a matrix $A = W\nabla_{\Phi}Y(t-\tau)$ and a column vector $G = f'(WY(t-\tau)+VX(t))$, $\nabla_W Y(t)$ and $\nabla_V Y(t)$ can recursively be computed by

$$\begin{aligned} \frac{\partial y_j(t)}{\partial w_{rs}} &= g_j[a_{rs} + \delta_{jr} y_s(t-\tau)] \\ \frac{\partial y_j(t)}{\partial v_{mn}} &= g_j[a_{mn} + \delta_{jm} s(t-n)], \end{aligned} \tag{6}$$

for $1 \leq j, r, s, m \leq N_h$ and $1 \leq n \leq N_i$.

2.2 Non-linear Predictive Vector Quantiser

Like a linear predictive vector quantiser (Wu and Fallside, 1991), a non-linear predictive vector quantiser consists of a set of non-linear predictors $\{F(\Phi_k, X),\ k = 1, \ldots, K\}$. The number of the predictors K is equal to 2^r, r (in bits) being the size of the non-linear predictive quantiser. In the same way as for linear predictive quantisers, the performance of non-linear predictive quantisers is evaluated by their distortion-rate functions. In the non-linear predictive quantiser, each predictor is trained to cover certain region of the non-linear predictive parameter space. The non-linear predictive quantiser is therefore expected to cope the non-stationary of the predicted signal and further improve the predictive performance of individual predictors. In quantisation, the predictor with the least predictive error is selected for each frame of signal. The process can be described by the equation:

$$c = \arg \min_{0 \leq k < K} \sum_{t}^{N} \| s(t) - F(\Phi_k, X(t)) \|^2, \tag{7}$$

N is the frame length. The non-linear prediction of the signal frame is then represented by the code symbol of r bit length instead of the weight parameters of the the selected predictor.

The training process of the non-linear predictive quantiser is the same as that for a single predictor, except the cost function becomes

$$D = \sum_{k}^{K} \sum_{S(t) \in k} \sum_{t}^{N} \| s(t) - F(\Phi_k, X(t)) \|^2, \tag{8}$$

and the updating equations become

$$\Delta \Phi_k = \eta \sum_{S(t) \in k} \sum_{t}^{N} \| s(t) - F(\Phi_k, X(t)) \| \nabla_{\Phi_k} z_k(t), \quad \text{for} \quad k = 1, \ldots, K. \tag{9}$$

Unlike linear prediction, however, the output of the non-linear predictor cannot simply be expressed as the sum of both the zero-state response and the zero-input response. The effect of the memory of the last selected predictor cannot be directly removed by subtracting the current frame signal with the zero-input response. The memory of the last frame should jointly be taken into account during the predictor selection for the current frame.

2.3 Long Term (Pitch) Non-linear Speech Prediction

Speech signal consists of two types of correlations: the correlation between successive speech samples and the correlation between adjacent pitch periods. The previous linear prediction-based speech production model contains two time-varying linear predictive synthesis filters, both the short term (formant) and the long term (pitch) linear predictive synthesis filters, describing the vocal tract. The former gives the spectral envelope of the speech signal and the latter re-produces the spectral fine structure. The long term linear prediction of speech has been studied, for example, by (Ramachandran and Kabal, 1989).

We study the implementation of long term non-linear prediction of speech using a neural network. The structure of the long term non-linear predictive model is the same as the short term one shown in figure 1, but the input is delayed by one pitch period before applying it to the input layer. A current sample is thus predicted by the samples around its last pitch period. The long term non-linear predictor is expressed by the following equation [2]:

$$\begin{aligned} Y_l(t) &= f(W_l Y_l(t-\tau) + V_l X_l(t-T)) \\ z_l(t) &= U_l Y_l(t), \end{aligned} \tag{10}$$

where T is the pitch period, which is found from the short term predictive residual,

$$r_s(t) = s(t) - z_s(t). \tag{11}$$

Long term prediction is always accompanied by short term prediction in speech coding. The long term predictor can be connected to the short term one in cascaded and parallel forms, as shown in figure 2. In the cascaded model, the pitch period delayed short term predictive residual, $r_s(t-T)$, is

[2] We use the subscript l to identify the long term predictor and s the short term one.

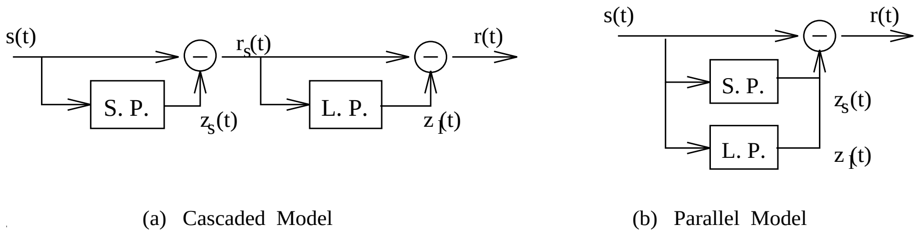

Figure 2: Short term (S.P.) and long term (L.P.) cascaded and parallel predictive models.

fed to the long term predictor, while, in the parallel model, the pitch period delayed original speech, $s(t-T)$, is applied to the long term predictor. So, in eqn(10),

$$X_l(t) = (r_s(t - N_{li}/2), \ldots, r_s(t + N_{li}/2))^T \tag{12}$$

for the cascaded model and

$$X_l(t) = (s(t - N_{li}/2), \ldots, s(t + N_{li}/2))^T \tag{13}$$

for the parallel model. The final residual of the short term and long term combined prediction for both the cascaded and parallel models is,

$$r(t) = s(t) - z_s(t) - z_l(t). \tag{14}$$

2.4 Non-linear Predictive Quantisation Performance

Besides its structure, two parameters of the predictive neural network, the predictive order and the total number of weights, also affect the property of the predictor. We compare the predictive quantisation performance between different structures under either of following two conditions: (a) same predictive order, or (b) same number of weights.

In all our studies, the neural network-based non-linear predictor contains only a two unit hidden layer.

2.4.1 Simulations with Sunspot Data Set

The sunspot data (Priestley, 1988) shown in figure 4 is a well-known data set which has been widely used as a benchmark in non-linear analysis. In our simulations, the data set is divided into two equally-sized sections. When the former section is as a training set, the latter one is as a test set, or vice versa. The results of the training and test performances are given by the average over the two sections. The preprocessing normalises and compacts the data into the range of $[-1, 1]$.

Figure 3 compares the learning curves of the predictors with different structures and sizes. We use the following notations, $LP(p)$ stands for a p^{th} order linear predictor, $FFNLP(N_i - N_h)$ for a

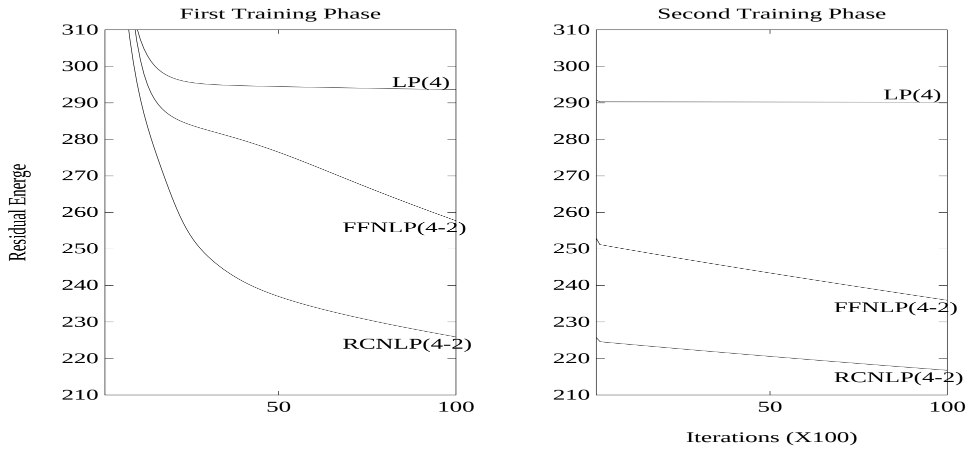

Figure 3: Comparing the learning processes between linear and non-linear predictors with the sunspot data set.

feedforward neural network-based non-linear predictor with N_i input units and N_h hidden units and $RCNLP(N_i - N_h)$ for that with recurrent connections. For these predictors, their learning processes are formed by two phases. The learning rate is set to 0.1 for linear predictors and feedforward net-based non-linear predictors, and 0.05 for recurrent ones in the first training phases, and the learning rate in the second phase is halved.

The decreasing slope of the learning curve is dependent on the magnitude of the learning rate, but different structures of predictors show different learning speeds. It is seen that (a) The predictive error of the linear predictor falls faster than the non-linear one at the early beginning of learning, but that it slows down immediately afterwards. (b) After certain updating iterations, the learning curve of the linear predictor becomes saturated while that of the non-linear one still reduces. (c) The magnitude of the minimal predictive error (after similar iterations) is: recurrent non-linear predictors, feedforward non-linear predictors, linear predictors, in increasing order.

In general, the learning curves will not monotonically decrease if the learning rate is not small enough. So the final weights of the predictor are set to those with the least predictive error in the whole training process instead of the result of the last updating iteration.

Table 1 compares the predictive quantisation performance between different structures and sizes of non-linear predictive quantisers with 1 bit sizes, under the condition either of the same predictive order or the equal number of weights. In both cases, we see: (a) All non-linear predictors are better than linear ones either with the same predictive order or with equal numbers of weights. (b) The recurrent non-linear predictors are better than the feedforward one. (c) All the one-bit quantisers outperform the zero-bit ones with the same structure and size. This benefit in the non-linear predictive quantiser is also greater than that in the linear one.

Figure 4 compares the predictive gain of the quantisers. We find that the non-linear predictive

Predictor Structure		*Predictive Quantisation Performance*
Same Predictive Order	LP(4)	10.783
	FFNLP(4-2)	12.900
	RCNLP(4-2)	13.222
Equal Number of Weights	LP(10)	12.892
	FFNLP(4-2)	12.900
	LP(14)	13.045
	RCNLP(4-2)	13.222

Table 1: Comparing the predictive quantisation performances (in dB) of different structures and sizes of linear and non-linear predictive quantisers with 1 bit sizes under the condition of either the same predictive order or the same number of weights, with the sunspot data set.

gain is almost always greater than the linear one using training data. Using test data, although there is a short segment where the non-linear predictive gain is worse than the linear one, on average, the former still outperforms the latter.

2.4.2 Simulations with Speech

Since linear prediction is an all-pole model, good results will only be obtained when the overall speech spectrum is like the response of an all-pole filter. For vowel sounds, the all-pole model can make good approximations to the spectral shapes of the formants. But for the nasals and some voiced consonant sounds the all-pole spectral modelling is poor. and its performance usually saturates at a very small predictive order. Therefore, a general non-linear model will be more suitable.

As an example with a frame of nasal speech, figure 5 compares the spectra of linear and non-linear predictive residuals. We see that when the predictive order is increased from 10 to 26, only a small improvement is obtained for the linear predictor. The $RCNLP(10-2)$, its predictive order is equal to 10 and the total number of weights is 26, outperforms both the $LP(10)$ and the $LP(26)$. Its predictive residual is closer to the white noise, and the formants have been filtered up in the predictive residual spectra.

The non-linear predictive quantiser is now evaluated with a set of speech data. Each speech frame contains 64 samples, and the error measure is also of the MSE type. The long term predictive quantisers are trained for the voiced segments only. Figure 6 compares the predictive gain-rate functions between the linear and non-linear predictive quantisers. In a manner consistent with the sunspot data set, all non-linear speech predictive quantisers outperform linear ones in all cases. The improvement in the large quantiser is also greater than that in the small one.

The improvements in the predictive gain of the non-linear predictive quantisers over the linear

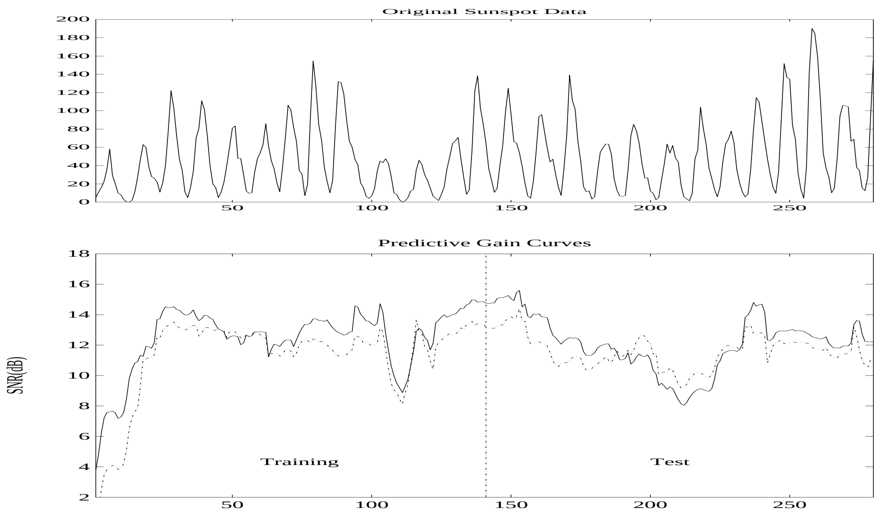

Figure 4: Predictive gains of the 1 bit sized linear (dotted line) and non-linear predictive quantisers using the sunspot data.

ones are given in figure 7, expressed as

$$p(\%) = \frac{2(SNR_{nlp} - SNR_{lp})}{SNR_{nlp} + SNR_{lp}}, \tag{15}$$

where SNR_{nlp} and SNR_{lp} are respectively the predictive gains of the non-linear and linear predictive quantisers. We see that in this example with the original speech shown in figure 8, about 30% improvement is achieved for the unvoiced speech, and 15% for the voiced segment. Figure 8 also plots the short term non-linear predictive residuals and the short term and long term cascaded non-linear predictive residuals. Figure 16 shows the spectrogram of another segment predictive residual. Compared with the original spectrogram, we see that both the formant and pitch strips have almost been removed in the residual spectrogram.

3 Code-excited Non-linear Predictive Speech Coding

In code-excited predictive speech coding as shown in figure 9 (Atal, 1986), the redundancy is first removed with predictors selected from the predictive quantisers, and the residual is then quantised by a vector quantiser. The residual vector quantiser usually performs the codevector search in an analysis-by-synthesis manner (Kroon and Deprettere, 1988). The predictive residual has a lower variance than the original speech, and can therefore be quantised more easily.

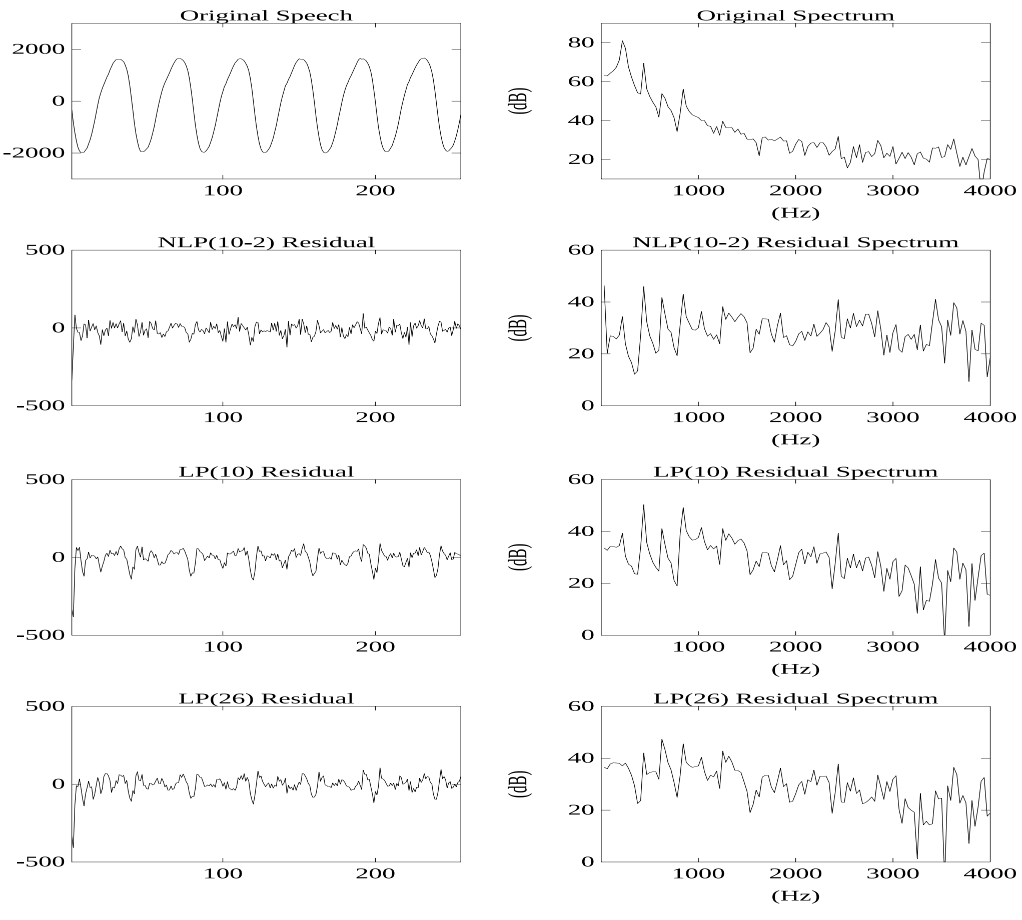

Figure 5: Linear and non-linear predictive residual spectra of nasal speech.

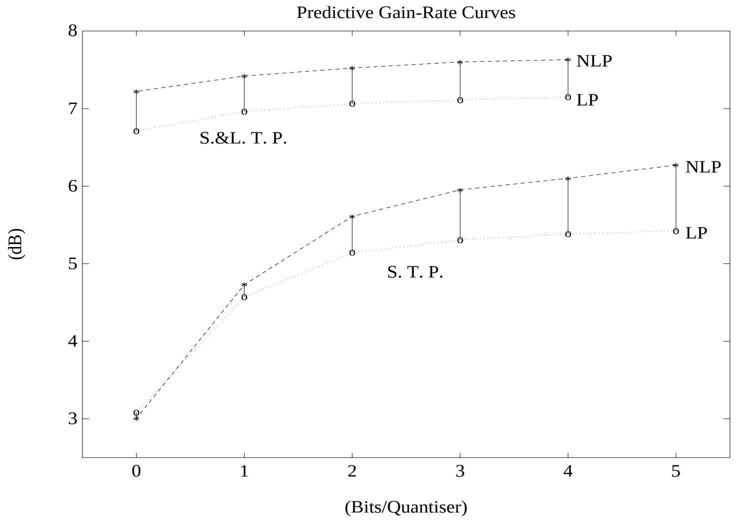

Figure 6: Predictive gain-rate functions of different structures and sizes of linear and non-linear speech predictive vector quantisers. The dashed curves connect the non-linear predictive vector quantisers, and dotted curves connect linear ones. The two groups of curves are respectively the short term predictive vector quantisers (S.T.P.) and the short term and long term cascaded predictive vector quantisers (S.&L.T.P.). The length of the vertical lines between the dashed and dotted curves indicate the improvement with non-linear prediction.

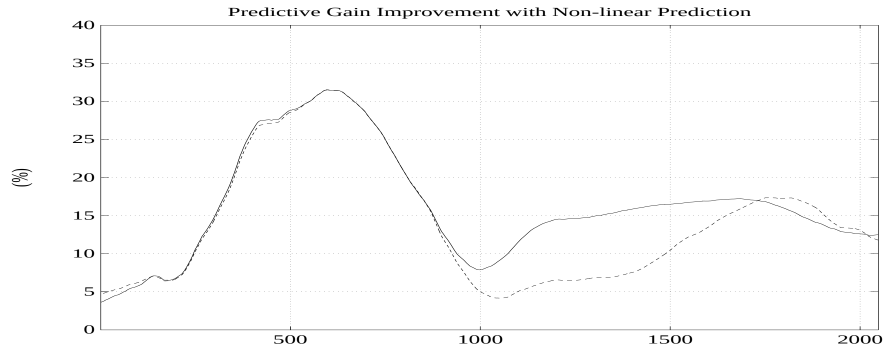

Figure 7: Predictive gain improvements of the non-linear speech predictive quantisers over the linear ones. The solid curve indicates the quantisers with the short term linear prediction, the dashed curve indicates those with the short term and long term cascaded prediction.

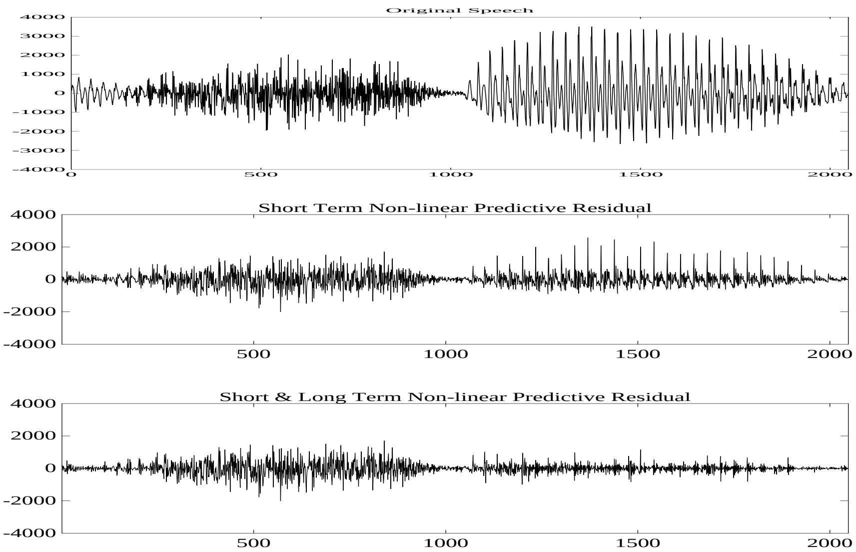

Figure 8: Non-linear predictive residuals of speech. From top to bottom, they are respectively original speech, the short term (formant) non-linear predictive residual, and the short term (formant) and long term (pitch) cascaded non-linear predictive residual.

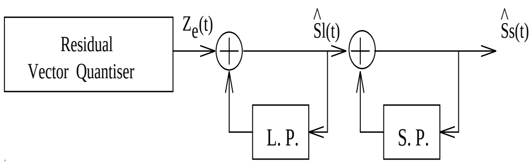

Figure 9: Code-excited predictive speech coder. L.P. stands for the long term (pitch) prediction, and S.P. for the short term (formant) prediction.

3.1 Tolerance of the Non-linear Predictive Synthesis Filter to Its Excitation Distortion

In linear predictive coding, the residual vector quantiser can be formed directly from the vector quantised residual samples. In this sub-section, however, we will show the vector quantised residual-excited model does not apply to the non-linear predictive coding due to its poor tolerance to its input disturbance.

Assume that the input of the predictive synthesis filter is changed from X_0 to X with $D_x = \| X - X_0 \|^2$, and their corresponding outputs are respectively z and z_0.

With linear prediction, $z = \alpha^T X$, where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_P)^T$ is the p^{th}-order linear predictive coefficient vector. We can easily get

$$D_z = \| z - z_0 \|^2 \leq \| \alpha \|^2 D_x \tag{16}$$

For non-linear prediction, we have derived (see Appendix)

$$D_z \leq \rho^2 D_x. \tag{17}$$

where

$$\rho = \frac{\gamma \| U \| \| V \|}{\gamma \| W \| - 1}, \tag{18}$$

$\| U \|$, $\| V \|$ and $\| W \|$ are respectively the Euclidean norms of U, V and W, $\gamma = \max \mid f'(O^*(t)) \mid$, $O^*(t) \in (O(t), O_0(t))$ and $O(t) = WY(t-r) + VX(t)$.

Assume that two predictive synthesis filters whose sensitivity parameters are ρ_1 and ρ_2. If they are connected in cascaded, the sensitivity parameter of the combined filter $\rho = \rho_1 \rho_2$, and if they are connected in parallel, the sensitivity parameter of the combined filter $\rho = \rho_1 + \rho_2$.

Eqns (16) and (17) respectively give the upper bounds of the output variations of the linear and non-linear predictive synthesis filters due to the input disturbance.
Since

$$0 \leq \gamma \| W \| < 1,$$[3]

so

$$(\gamma \| W \| - 1)^2 < 1.$$

If $(\gamma \| W \| - 1)^2$ is close to zero, then ρ becomes very large. Therefore, the output will greatly deviate from its original value when the input gets a small disturbance.

As an example, table 2 compares the sensitivity parameters, $c = \frac{D_z}{D_x}$, of the non-linear predictive synthesis filters in both linear and non-linear predictive vector quantisers with 4 bit sizes. All non-linear units are with a sigmoid function $f(x) = \frac{2}{1+e^{-2x}} - 1$, so $\gamma = 1$. We see that the the sensitivity parameters of non-linear predictive synthesis filters are much greater than the linear ones, in particular, for those with a large predictive order.

[3] see Appendix.

$Predictor$	$Predictive\ Order$			
	4		14	
	$Non-linear$	$Linear$	$Non-linear$	$Linear$
0	24.90	1.28	63.40	3.91
1	25.88	1.29	48.17	2.92
2	7.63	1.14	422.50	4.25
3	6.18	2.18	37.28	2.81
4	768.55	7.03	1004.43	2.65
5	470.50	1.48	530.01	2.11
6	70.82	5.20	19.76	2.04
7	47.15	1.37	13.93	1.84
8	23.73	0.35	66.55	1.19
9	65.67	0.35	58.79	1.03
10	25.75	0.21	27.03	0.92
11	12.76	0.25	10.37	0.80
12	37.72	0.35	15.09	0.57
13	7.23	0.24	6.42	0.44
14	2.50	0.18	11.50	0.23
15	1.23	0.14	6.59	0.58

Table 2: Compare the sensitivity parameters of the predictive synthesis filters in the 4 bit linear and non-linear predictive vector quantisers.

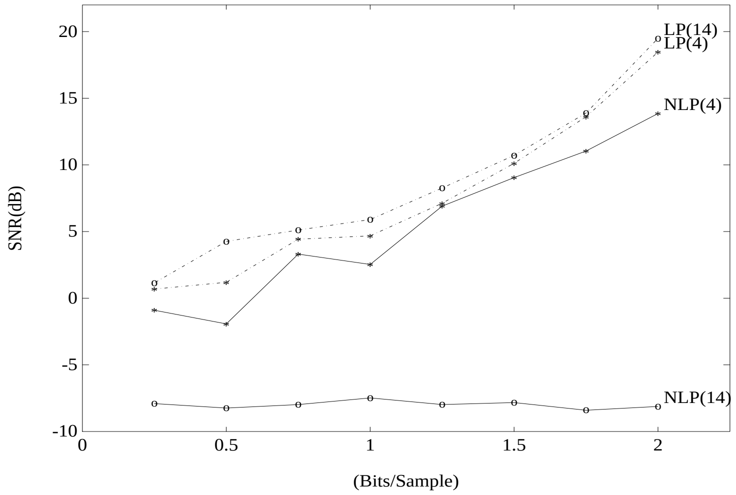

Figure 10: Coding performance to the excitation distortion in the vector quantised residual-excited predictive speech coders. Where the solid curves connect the coders with non-linear predictive quantisers and the dashed curves connect those with the linear predictive quantisers. The p in $LP(p)$ and $NLP(p)$ stands for the predictive order.

The strong sensitivity of the output variance to its input disturbance of the non-linear predictive synthesis filters can be further demonstrated by figure 10 with the vector quantised residual-excited predictive coders. The sizes of the predictive quantisers are 4 bits, and the predictive residuals are vector quantised with the rate from 0.25 to 2 bits/sample. This shows the reconstructed speech quality against the sizes of the residual vector quantisers. We see that the non-linear predictive coders are much poorer than the linear predictive ones, especially, for those with the 14^{th} predictive order.

To deal with the poor tolerance of the non-linear predictive synthesis filter to its input disturbance in non-linear predictive coding, two approaches can be applied.

1. as an accompanying training criterion, minimising $\frac{1}{\gamma||W||-1}$ at the same time as minimising the predictive error;
2. using the trained excitation codevectors in the code-excited non-linear predictive coder with the analysis-by-synthesis method, instead of using the vector quantised residual.

We will discuss the first approach elsewhere, and focus on training a codebook-excited neural network (CENN) (Wu and Fallside, 1992) as an excitation vector quantiser in non-linear predictive coding.

3.2 CENN-excited Non-linear Predictive Coding

In the non-linear predictive coder, its output cannot be expressed as the sum of both the zero-state response and the zero-input response. The complete form of the excitation signal does not exist and the conventional vector quantiser design technique, the LBG algorithm, is not applicable either. Therefore, we implement the excitation vector quantisation with the CENN.

With the neural network-based short term and long term combined non-linear predictive vector quantiser as the predictive part and the CENN as the excitation part, the predictive speech coder can be expressed by the following equations.

Assuming N_{eo} is the output dimension of the CENN, we define $\hat{S}(t)$, $\hat{S}_l(t)$, $Z_s(t)$, $Z_l(t)$ and $Z_e(t)$ as N_{eo} sized column vectors, and $\mathbf{Y}_s(t)$, $\mathbf{Y}_l(t)$, $\mathbf{X}_s(t)$ and $\mathbf{X}_l(t)$ [4] respectively as $N_{sh}\times N_{eo}$, $N_{lh}\times N_{eo}$, $N_{si}\times N_{eo}$ and $N_{li}\times N_{eo}$ matrices.

For the predictive part,

$$\begin{aligned} \mathbf{Y}_l(t) &= f(W_l\mathbf{Y}_l(t-\tau)+V_l\mathbf{X}_l(t-T)), & Z_l(t) &= [U_l\mathbf{Y}_l(t)]^T, \\ \mathbf{Y}_s(t) &= f(W_s\mathbf{Y}_s(t-\tau)+V_s\mathbf{X}_s(t)), & Z_s(t) &= [U_s\mathbf{Y}_s(t)]^T, \end{aligned} \tag{19}$$

where

$$\begin{aligned} \mathbf{X}_l(t) &= (\hat{S}_l(t-N_{li}/2),\ldots,\hat{S}_l(t+N_{li}/2))^T \\ \mathbf{X}_s(t) &= (\hat{S}(t-1),\ldots,\hat{S}(t-N_{si}))^T, \end{aligned} \tag{20}$$

and

$$\begin{aligned} \hat{S}_l(t) &= Z_l(t)+Z_e(t) \\ \hat{S}(t) &= Z_s(t)+\hat{S}_l(t). \end{aligned} \tag{21}$$

For the excitation part,

$$\begin{aligned} Y_e(t) &= f(W_eY_e(t-\tau)+V_eX_e(t)) \\ Z_e(t) &= U_eY_e(t). \end{aligned} \tag{22}$$

The coding distortion is

$$D(t) = \| E(t) \|^2 = \| S(t)-\hat{S}(t) \|^2. \tag{23}$$

The training process of U_e, V_e and W_e is outlined as follows. For simplity, we assume that the coder consists of a short term predictive quantiser only. For that with both the short term and the long term combined predictive quantisers, the training algorithm can be derived analogously.

Letting $\Phi \in \{U_e, V_e, W_e\}$, then

$$\Delta\Phi = \eta E(t)\nabla_\Phi \hat{S}(t). \tag{24}$$

Since

$$\hat{S}(t) = Z_s(t)+Z_e(t), \tag{25}$$

so

$$\nabla_\Phi \hat{S}(t) = \nabla_\Phi Z_s(t)+\nabla_\Phi Z_e(t). \tag{26}$$

[4] To avoid confusing $Y(t)$ and $X(t)$ in eqn(3) and eqn(10), we use the bold type style.

The back propagated $\nabla_\Phi Z_s(t)$ in the predictive neural network is

$$\begin{array}{rcl} \nabla_\Phi \mathbf{Y}_s(t) & = & \left[W_s \nabla_\Phi \mathbf{Y}_s(t-\tau) + V_s \nabla_\Phi \mathbf{X}_s(t)\right] {G_s}^T \\ \nabla_\Phi Z_s(t) & = & U_s \nabla_\Phi \mathbf{Y}_s(t), \end{array} \tag{27}$$

where

$$\begin{array}{rcl} \nabla_\Phi \mathbf{X}_s(t) & = & \left(\nabla_\Phi \hat{S}(t-1), \ldots, \nabla_\Phi \hat{S}(t-N_{si})\right)^T \\ G_s & = & f'(W_s \mathbf{Y}_s(t-\tau) + V_s \mathbf{X}_s(t)). \end{array} \tag{28}$$

The back-propagated $\nabla_\Phi Z_e(t)$ in the CENN is

$$\begin{array}{rcl} \nabla_{W_e} Z_e(t) & = & U_e \nabla_{W_e} Y_e(t) \\ \nabla_{V_e} Z_e(t) & = & U_e \nabla_{V_e} Y_e(t) \end{array} \tag{29}$$

and

$$\begin{array}{rcl} \dfrac{\partial z_{ei}(t)}{\partial u_{epq}} & = & \delta_{ip} y_{eq}(t) \\ \dfrac{\partial y_{ej}(t)}{\partial w_{ers}} & = & g_j[a_{rs} + \delta_{jr} y_{es}(t-\tau)] \\ \dfrac{\partial y_{ej}(t)}{\partial v_{emn}} & = & g_j[a_{mn} + \delta_{jm} x_{en}(t)], \end{array} \tag{30}$$

where $1 \leq i, p \leq N_{eo}$, $1 \leq j, q, r, s, m \leq N_{eh}$, $1 \leq n \leq N_{ei}$, a and g are respectively the elements of

$$\begin{array}{rcl} A & = & W_e \nabla_\Phi Y_e(t-\tau) \\ G & = & f'(W_e Y_e(t-\tau) + V_e X_e(t)). \end{array} \tag{31}$$

3.3 Coding Performance

With 4 bit predictive vector quantisers, the comparison of the coding performance of the CENN-excited, linear and non-linear predictive speech coders is given in figure 11. The sizes of the CENN are set to 4 bits and 6 bits. Its output dimension is varied from 4 to 16. The CENN is formed with three feedforward layer with two hidden units. The number of input units is equal to that of the output units. The excitation codebook of the CENN are set with the Gaussian samples. The above design results in a series of the speech coders with the transmitted rate varying from 0.375 to 1.625 (bits/sample). As shown by figure 11, the non-linear predictive coders outperform the linear ones in the all coders.

To show the advantage with the multi-layer CENN, we design an excitation vector quantiser without any network transformation. The excitation vector quantiser is initialised with Gaussian sequences and trained with a descent method. Figure 12 shows that the coders with this excitation vector quantiser are all worse than those with the multi-layer CENN.

A set of signal waveforms of the CENN-excited non-linear predictive speech coder are shown in figure 13. The CENN maps the Gaussian sequence to its output as the excitation of the predictive synthesis filter. This excitation is much closer to the actual predictive residual shown in figure 8 than the Gaussian sequence. With the long term predictive synthesis filter, impulses are formed at each pitch period of the voiced speech. The output of the long term predictive synthesis filter is then applied to the short term predictive synthesis filter to generate the synthetic speech.

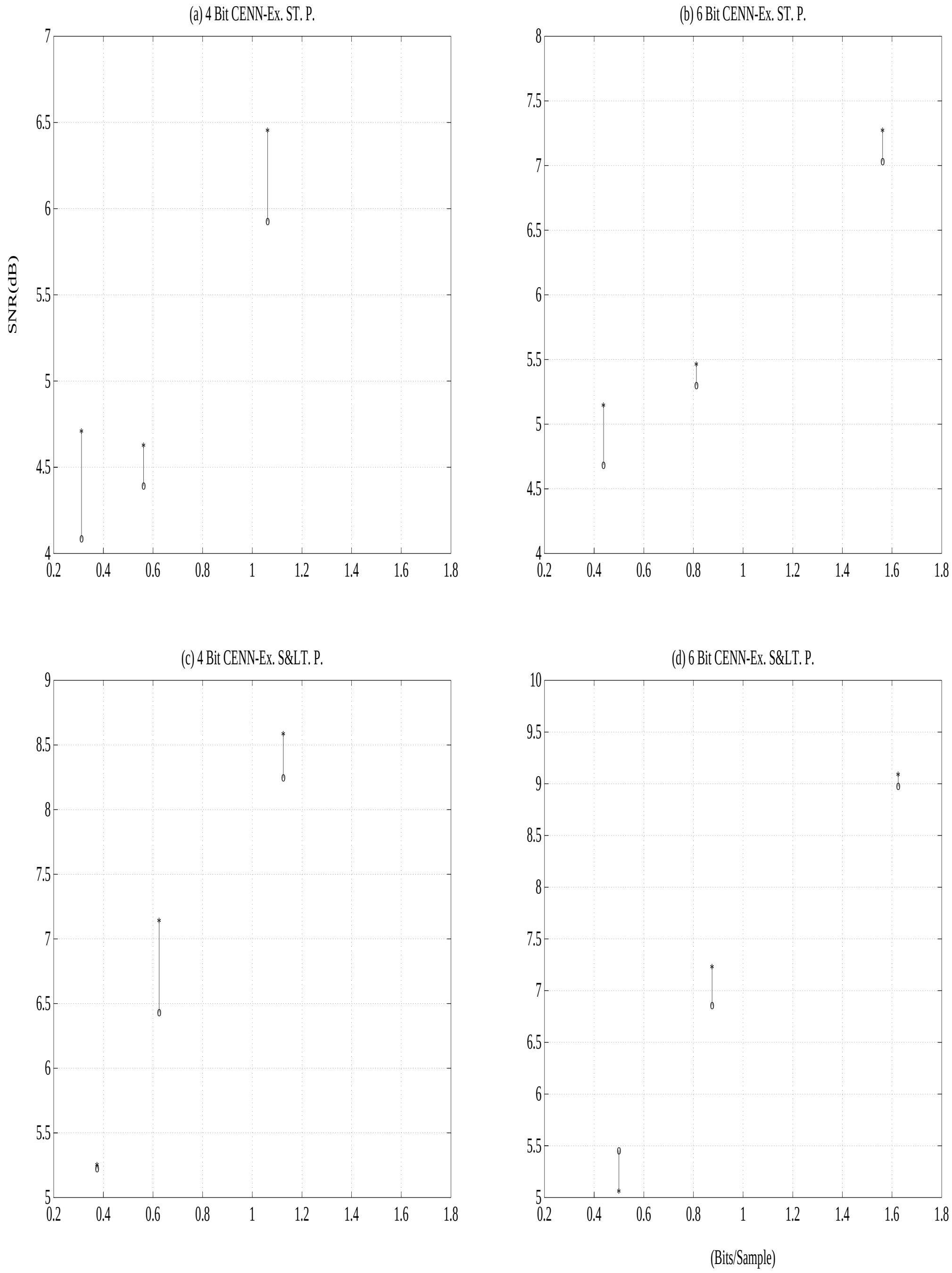

Figure 11: Compare the coding performances of the CENN-excited linear and non-linear predictive speech coders with different transmitted rate. The dimensions of the excitation codevectors are set to 4, 8 and 16 respectively. The coders shown in figure 11-(a) and (c) are with 4 bit CENNs, those in figure 11-(b) and (d) are with 6 bit CENNs. The coders shown in figure 11-(a) and (b) are with the short term predictive quantisers, those in figure 11-(c) and (d) are with the short term and long term cascaded predictive quantisers. In all sub-figures, "$*$" labels the non-linear predictive coding performance and "o" the linear one. The length of the vertical line between "$*$' and "o" indicates the coding performance improvement.

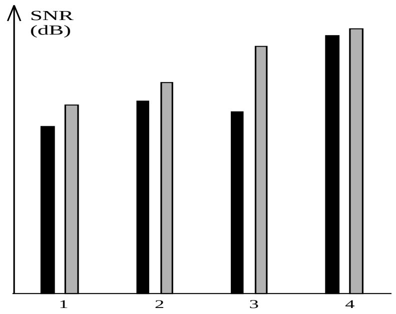

Figure 12: Compare the coding performance of non-linear predictive speech coders respectively excited by the vector quantiser trained with the descent algorithm (black lines) and excited by a three-layer CENN (gray lines). The size of the non-linear predictive quantiser is 4 bits. The output dimension of the CENN is equal to 4. The size of the CENN is 4 bits for the coders labelled by 1 and 3, and 6 bits for those labelled by 2 and 4. The coders labelled by 3 and 4 are with short term and long term cascaded non-linear prediction.

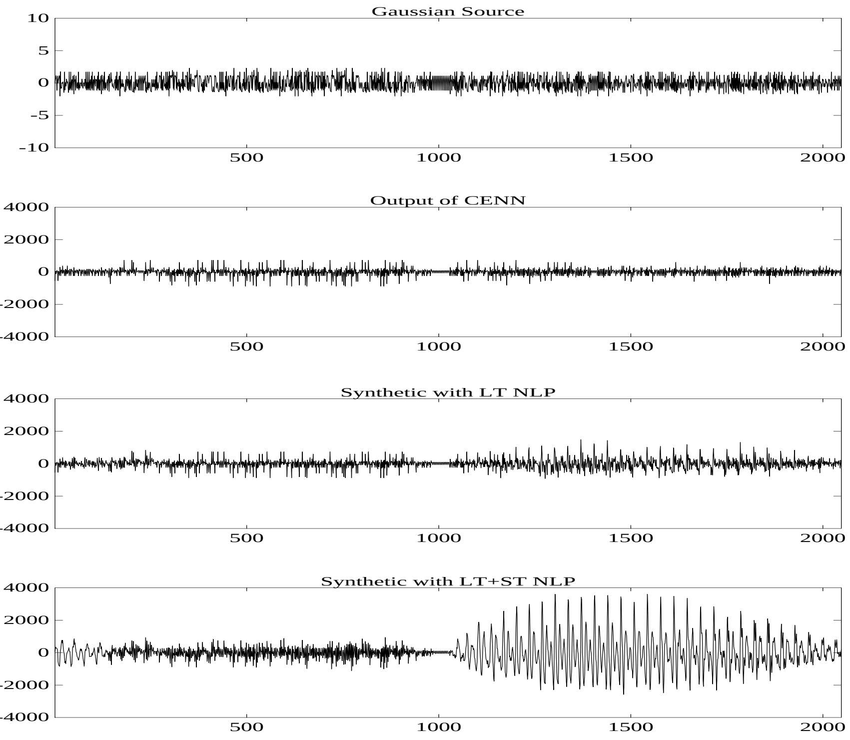

Figure 13: Signal waveforms of the CENN-excited, short term and long term cascaded, non-linear predictive speech coding. From top to bottom, they are respectively the source of the CENN (Gaussian sequence), the output of the CENN, the output of the long term predictive synthetic filter and the output of the short term predictive synthetic filter.

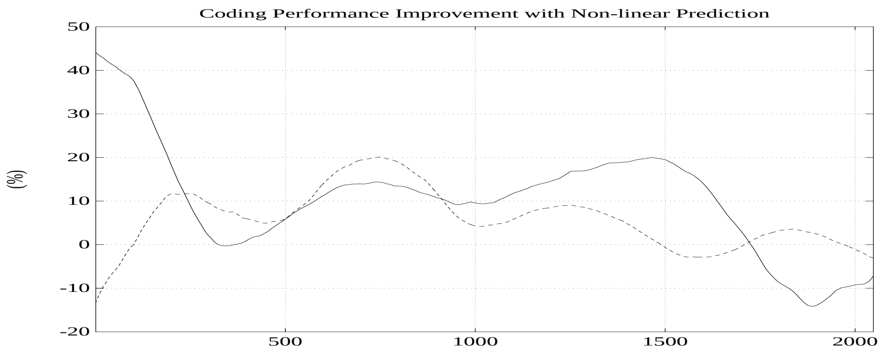

Figure 14: Coding performance improvements of the CENN-excited non-linear predictive speech coders over the linear ones. The solid curve indicates the coders with the short term linear prediction, the dashed curve indicates those with the short term and long term cascaded prediction.

The improvement of the coding performance with non-linear prediction is shown in figure 14. The measure used is the same as the eqn(15). We see that the improvements are greater than zero during most of frames. On average, about 15% improvement is achieved for both voiced and unvoiced speech.

In figure 15, a comparison is made of spectra. We see that for the voiced speech frame, the synthetic spectrum is fairly close to the original except in the high frequency part. Increasing the predictive order can be expected to reduce this distortion in the high frequency band. The main difference between the original and synthetic spectra of the unvoiced frame is their amplitudes. This can be improved by gain-adaptive coding as discussed in the next sub-section. Figure 16 shows the spectrogram of a segment of synthetic speech. We see the basic information such as the formant positions and amplitudes and its transition tendencies have been re-constructed in the synthetic spectrogram.

3.4 Discussion on Gain-adaptive Non-linear Predictive Coding

Although the non-linear predictive neural network and the CENN can be trained to fit the signal with any scale amplitude at their outputs since no non-linear activation function is imposed on their output units, gain normalisation will smooth the outputs and reduce the output variances so as to improve the adaptive ability of the coder. This issue consists of designing a gain separated non-linear predictive vector quantiser and inserting a gain term in the coder. Unlike linear predictive speech coding, the gain term is placed at the output of the non-linear predictive synthesis filter shown in figure 17, instead of at the output of the CENN (Both places are equivalent to the linear case). Such an architecture leads to that the $g(t)$ is independent of the $Z_p(t)$ and its optimal solution can be

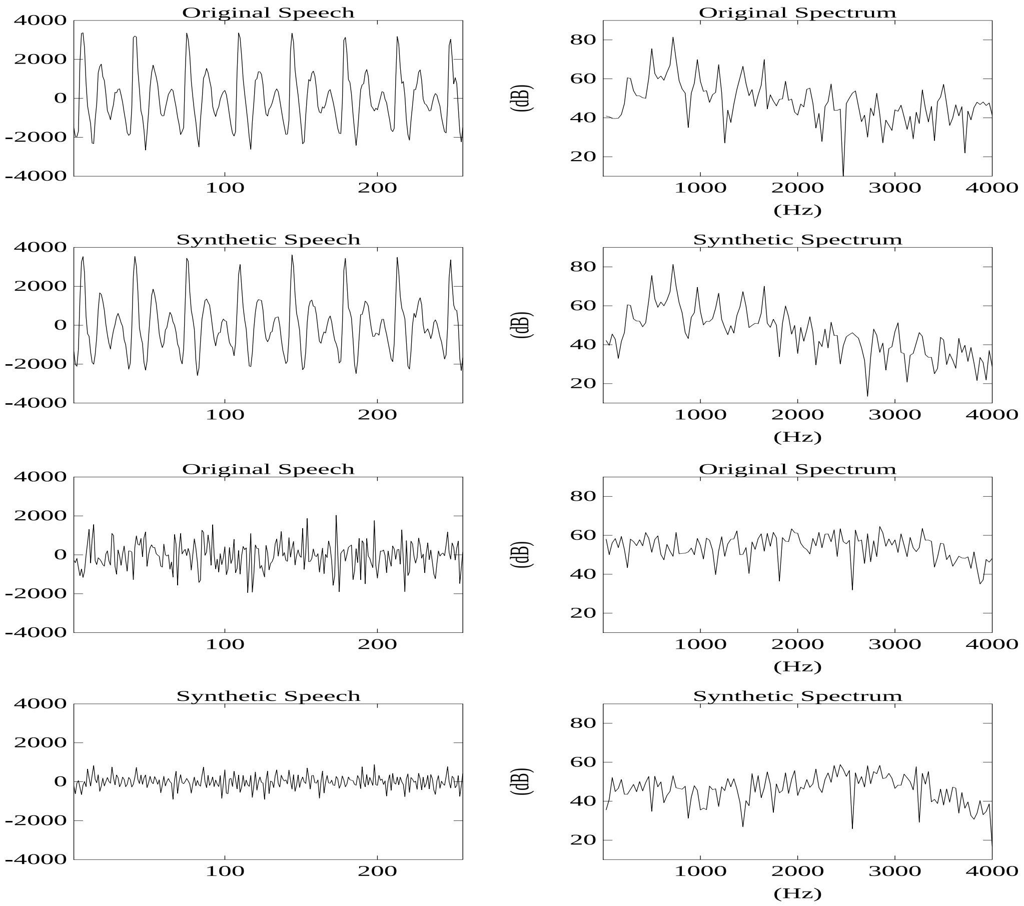

Figure 15: Spectra of original speech and synthetic one with the CENN-excited non-linear predictive coder.

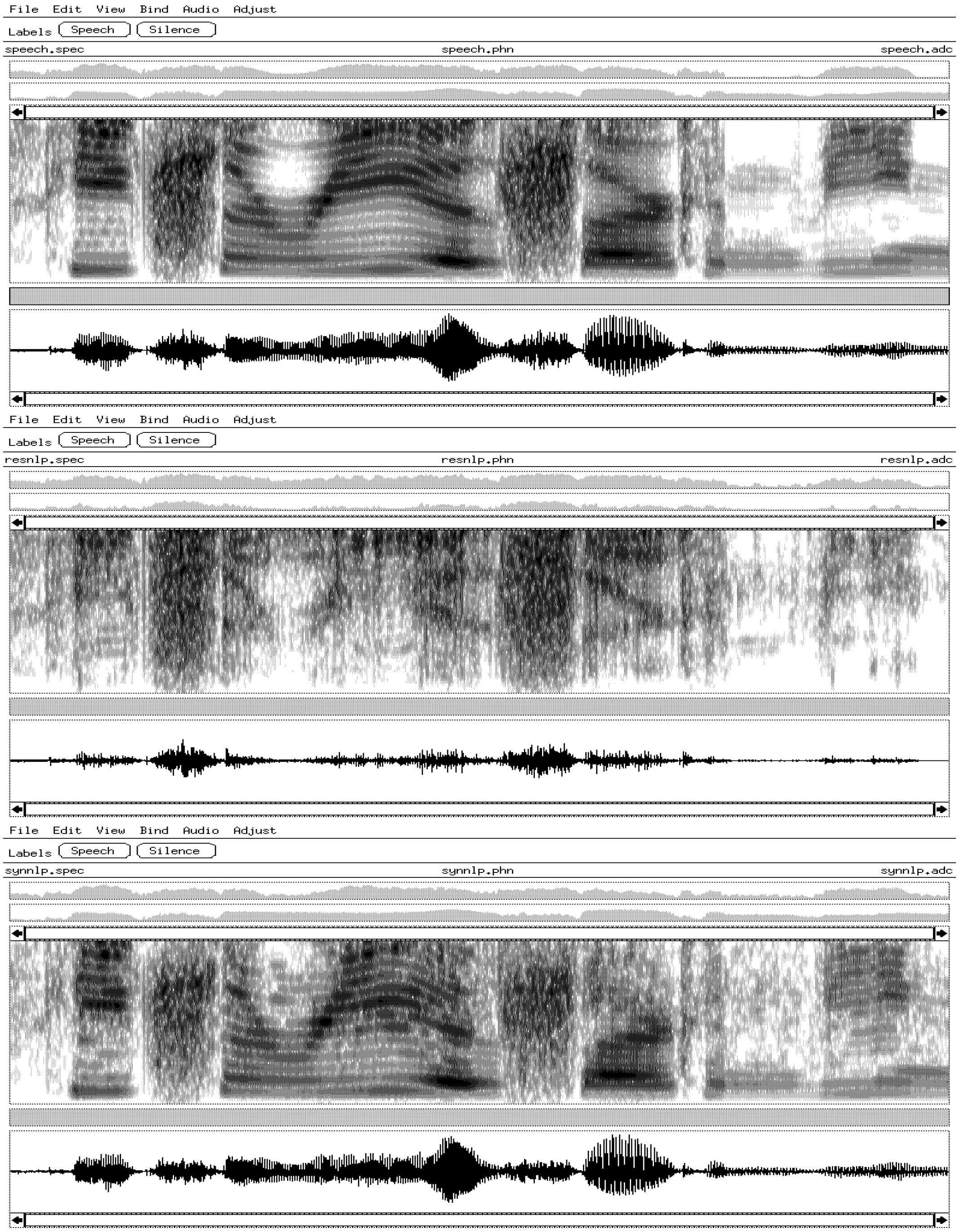

Figure 16: Spectrograms of original speech, the predictive residual of the non-linear predictive vector quantiser and the synthetic speech of the CENN-excited non-linear predictive coder (in the order from top to bottom).

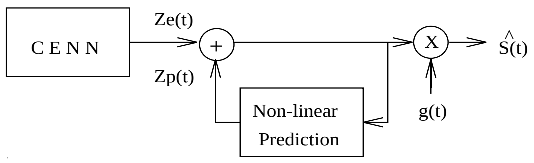

Figure 17: Gain-adaptive CENN-excited non-linear predictive speech coder.

found as

$$g(t) = \frac{[S(t)]^T [Z_e(t) + Z_p(t)]}{\| Z_e(t) + Z_p(t) \|^2}. \tag{32}$$

This results in the coding error

$$\begin{aligned} D &= \| S(t) - g(t)[Z_e(t) + Z_p(t)] \|^2 & (33) \\ &= \| S(t) \|^2 - \frac{\{[S(t)]^T [Z_e(t) + Z_p(t)]\}^2}{\| Z_e(t) + Z_p(t) \|^2}. & (34) \end{aligned}$$

Therefore, the gain-adaptive non-linear predictive speech coder can be trained by minimising the above D or maximising the second term of eqn(34).

4 Concluding Remarks

The above studies have shown the applicability of non-linear prediction to speech coding, the improvement of the coding performance with non-linear prediction, and the implementation of the fully vector quantised code-excited non-linear predictive speech coder with neural network. In continuing studies, other issues, e.g. improvement of the tolerance of the non-linear predictive synthesis filter to its input distortion, gain-adaptive non-linear predictive coding, will be further investigated, together with problems such as weighting, computational simplification.

5 Appendix: Output Variance of Neural Network to Its Input Perturbation

Defining a state vector $O(t) = WY(t-r) + VX(t)$, using the mean value theorem

$$f(O(t)) - f(O_0(t)) = f'(O^*(t))(O(t) - O_0(t)), \tag{35}$$

and Schwarz's inequality

$$\| f(O(t)) - f(O_0(t)) \| \leq \| f'(O^*(t)) \| \, \| O(t) - O_0(t) \|, \tag{36}$$

we get

$$\begin{aligned} D_z &= \| Uf(O(t)) - Uf(O_0(t)) \|^2 \\ &\leq \gamma^2 \| U \|^2 \| O(t) - O_0(t) \|^2 & (37) \end{aligned}$$

where $\gamma = \max \mid f'(O^*(t)) \mid$ and $O^*(t) \in (O(t), O_0(t))$.

τ is usually small, so the neural network satisfies the following dynamic function, [5]

$$\tau \frac{dO(t)}{dt} = Wf(O(t)) - O(t) + VX(t). \tag{38}$$

If we let

$$D_o(t) = \parallel O(t) - O_0(t) \parallel^2, \tag{39}$$

then

$$\frac{dD_o(t)}{dt} = 2(O(t) - O_0(t))^T \left(\frac{dO(t)}{dt} - \frac{dO_0(t)}{dt} \right). \tag{40}$$

With eqn(38)

$$\frac{dO(t)}{dt} - \frac{dO_0(t)}{dt} = \frac{1}{\tau} \{ W[f(O(t)) - f(O_0(t))] - (O(t) - O_0(t)) + V(X(t) - X_0(t)) \}. \tag{41}$$

Using the mean value theorem and Schwarz's inequality again, we get

$$\begin{aligned} \frac{dD_o(t)}{dt} &= \frac{2}{\tau} \{ \parallel O(t) - O_0(t) \parallel^T W[f(O(t)) - f(O_0(t))] \\ &\quad - \parallel O(t) - O_0(t) \parallel^2 + \parallel O(t) - O_0(t) \parallel^T V(X(t) - X_0(t)) \} \\ &\leq \frac{2}{\tau} \left[\xi D_o(t) + \zeta \sqrt{D_o(t)} \right], \end{aligned} \tag{42}$$

where

$$\xi = \gamma \parallel W \parallel -1, \tag{43}$$

$$\zeta = \sqrt{D_x} \parallel V \parallel. \tag{44}$$

If we let $D_o(t) = Q^2(t)$ and $Q(t) > 0$, then eqn(42) becomes

$$\frac{dQ(t)}{dt} \leq \frac{1}{\tau} [\xi Q(t) + \zeta], \tag{45}$$

which yields

$$Q(t) \leq \left[Q(0) + \frac{\zeta}{\xi} \right] \exp\left(\frac{\xi}{\tau} t \right) - \frac{\zeta}{\xi} \tag{46}$$

or

$$D_o(t) \leq \left\{ \left[\sqrt{D_o(0)} + \frac{\zeta}{\xi} \right] \exp\left(\frac{\xi}{\tau} t \right) - \frac{\zeta}{\xi} \right\}^2. \tag{47}$$

If $O_0(t)$ is the unique equilibrium for the input X_0, the network satisfies (Atiya, 1987),

$$\gamma \parallel W \parallel < 1, \tag{48}$$

[5] The variable t of $X(t)$ and $Z(t)$ indicates that the input and the output are time varying. The t in this dynamic function represents the settling time of the evolution process of the network from initial state to target state. During the evolution period, the input can be assumed as fixed, since the updating rate of the activation of the network is usually much greater than the input updating rate.

so $\xi < 0$ and

$$\sup_{t \to \infty} D_o(t) = \left(\frac{\zeta}{\xi}\right)^2. \tag{49}$$

By letting

$$\rho = \frac{\gamma \parallel U \parallel \parallel V \parallel}{\gamma \parallel W \parallel -1} \tag{50}$$

eqn(37) becomes

$$D_z \leq \rho^2 D_x. \tag{51}$$

References

Atal, B. (1986). High-quality speech at low bit rates: multi-pulse and stochastically excited linear predictive coders. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 1681–1684.

Atiya, A. (1987). Learning on a general network. In Anderson, D., editor, *Proceedings of Neural Information Processing Systems*, pages 22–30. American Institute of Physics, Denver.

Kroon, P. and Deprettere, E. F. (1988). A class of analysis-by-synthesis predictive coders for high quality speech coding at rates between 4.8 and 16 kbits/s. *IEEE Journal on Selected Areas in Communications*, 6(2):353–363.

Lapedes, A. and Farber, R. (1987). Nonlinear signal processing using neural networks: prediction and system modelling. Technical Report LA-UR-87-2662, Los Alamos National Laboratory, Los Alamos, New Mexico 87545.

Priestley, M. (1988). *Non-linear and non-stationary time series analysis.* Academic Press.

Rabiner, L. and Schafer, R., editors (1978). *Digital processing of speech signals.* Englewood Cliffs, Prentice-Hall.

Ramachandran, R. P. and Kabal, P. (1989). Pitch prediction filters in speech coding. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-37(4):467–478.

Tishby, N. (1990). A dynamical systems approach to speech processing. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 365–368.

Wang, S., Paksoy, E., and Gersho, A. (1990). Performance of nonlinear prediction of speech. In *Proceedings of International Conference on Spoken Language Processing, Kobe, Japan*, pages 2.1.1–2.1.4.

White, H. (1989). Learning in artificial neural networks: A statistical perspective. *Neural Computation*, 1:425–464.

Wu, L. and Fallside, F. (1991). On the design of connectionist vector quantizers. *Computer Speech and Language*, 5(3):207–230.

Wu, L. and Fallside, F. (1992). Source coding and vector quantization with codebook-excited neural networks. *Computer Speech and Language*, in press.