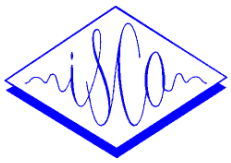

First International Conference on
Spoken Language Processing (ICSLP 90)
Kobe, Japan
November 18-22, 1990

# PERFORMANCE OF NONLINEAR PREDICTION OF SPEECH

*Shihua Wang, Erdal Paksoy and Allen Gersho*

Center for Information Processing Research
Dept. of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106 U.S.A.

## ABSTRACT

A novel method for nonlinear prediction of speech is introduced which does not require a parametric model of the predictor. The observable past is vector quantized and a nonlinear prediction is obtained by a table lookup, addressed by the index of the quantized input vector. The table is designed with speech training data. Experimental results for a moving-average process confirm that nearly optimal nonlinear prediction is achievable. Results for speech show that the performance depends on both the size of the vector quantizer codebook and the size of the training set. The method is applied to DPCM and some useful performance gain is demonstrated.

## 1. INTRODUCTION

Linear prediction (LP) and linear predictive coding (LPC) techniques have played a central role in speech processing. LP provides a valuable and convenient representation of the spectral envelope of a speech frame and its parameters are easily computed from a small number of autocorrelation values. Also, a variety of well-studied methods exist for adapting a linear predictor to match the source statistics. LP is able to remove a substantial amount of redundancy from a speech waveform by a simple filtering operation specified by only a few parameters.

For a Gaussian source, the linear predictor is the optimal least-squares estimator. Speech, however, is a non-Gaussian signal whose higher order statistics may contain important information which cannot be extracted by LP. A method limited to examining only the linear statistical dependencies of a block of speech samples overlooks important nonlinear dependencies that may be present. Therefore, LP is in general not an optimal solution for removing redundancy or modeling a speech segment. Furthermore, the maturity of LP analysis has led to a saturation of the benefits obtainable with this technique, motivating the search for alternative methods that exploit the nonlinear dependency among speech samples.

Some efforts have been made in recent years to develop methods for nonlinear prediction. In [1], an analysis of speech with a nonlinear dynamical system model leads to a neural network implementation of a nonlinear predictor. Another nonlinear prediction method uses a truncated Volterra series [2, 3]. In [4], samples of a signal are predicted by inspecting the past for patterns of samples that match the most recent set. All of these methods attempt to exploit the nonlinear dependencies between source samples and provide encouraging results, but there is still an inadequate understanding of the potential benefits for speech coding.

When the linearity constraint is abandoned, the best least-squares estimate of a random variable is its conditional expectation given the observed variables. This generally requires a nonlinear operation, thus leading to nonlinear prediction (NLP). The difficulty of analysis for NLP is that a multivariate probability density function (pdf) of the speech waveform, rather than the autocorrelation function, is required for computing the conditional expectation. In general, the joint pdf of a block of speech samples is not available and even if it were possible to find some model for this pdf, the evaluation of the conditional expectation from the pdf is likely to be analytically and computationally intractable.

The design objective in NLP is to find a method for evaluating the conditional expectation given any set of past observable values. In the Volterra and neural network methods, a parametric model is found which approximates the desired functional mapping from observable past samples of the speech waveform to a prediction of the next sample. Once such a model is found, it is relatively simple to perform NLP. However, the accuracy of the model is limited by its particular structure, the number of parameters it has, and by the amount of empirical data (training data) used to compute the parameter values.

In this paper, we present a new *nonparametric* approach to NLP that does not require any predefined model of the mapping. Instead, we directly compute the numerical values of the optimal nonlinear mapping from the training data. To reduce the enormity of the computational task, the set of possible values for the past observables is represented by means of vector quantization. Therefore, instead of conditioning the desired random variable on the exact values of the past observables, we condition it on a quantized version of the past. In other words, we approximate the conditional expectation $E\{x|y\}$ by $E\{x|Q(y)\}$, where $Q(y)$ is the quantized value of $y$. As the quantizer resolution gets higher the approximation will get more and more accurate. Hence, the estimate will approach, in an asymptotic sense, the optimal mean square estimate.

## 2. NONLINEAR PREDICTION

Let $\{x_k\}$ be a sequence of samples from a random process. We wish to predict the sample $x_{k+1}$ given a finite number of past samples, $x_k, x_{k-1}, \cdots, x_{k-P+1}$, where P is the prediction order. In other words, our objective is to design a mapping $g(\cdot)$ which has the observable vector $\underline{x}_k = [x_k, x_{k-1}, \cdots, x_{k-P+1}]$ as its input and a scalar value as its output, which is the prediction of the true value $x_{k+1}$. We seek the optimal mapping $g(\underline{x}_k)$ that minimizes the mean squared error $E[x_{k+1} - g(\underline{x}_k)]^2$. It is well known that the optimal prediction is achieved by evaluating the expected value of $x_{k+1}$ given the set of observables $\underline{x}_k$. Thus,

$$g(\underline{x}_k) = E\{x_{k+1} | \underline{x}_k\} \tag{1}$$

Evaluation of the above expectation requires the knowledge of the joint probability distribution of $x_{k+1}, x_k, x_{k-1}, \cdots, x_{k-P+1}$, but this information is not available.

Instead of proposing an *ad hoc* model for the function $g(\cdot)$ in terms of a few parameters, we wish to directly compute and store a numerical tabulation of this function. Ideally, we need a table containing all possible sequences of P consecutive speech samples and for each such vector $\underline{x}_k$, the expected value (ensemble average) of the corresponding next sample $x_{k+1}$. Then for each input vector to the predictor, we would search the entire table to determine the

This work was supported by the National Science Foundation, the University of California MICRO program, Bell Communications Research, Bell-Northern Research, and Rockwell International Corporation.

corresponding output. However, we face two major limitations.

First of all, for a random sequence with continuously distributed amplitudes, there is an infinite number of possible sequences of P samples (vectors $\underline{x}_k$). Even if we assume that each sample is specified with a finite number of bits of resolution as in a 12 bit per sample digitized speech signal, the required memory size and the search time would be astronomic even for moderate predictor orders P. Secondly, we need to evaluate the conditional expectation which would require knowledge of the conditional pdf $f(x_{k+1} \mid \underline{x}_k)$ for all $\underline{x}_k$. We solve the first problem with the help of vector quantization (VQ) and the second problem through the use of training data.

We assume that we have a training set consisting of a large number, say M, of empirically observed pairs $(\underline{v}_k, v_{k+1})$ from a speech sequence $\{v_k\}$, our objective is to determine a table of constrained size N that specifies the optimal mapping $g(\underline{x}_k)$.

Using a standard VQ design algorithm with the training set of vectors $\underline{v}_k$ (taken from the given training pairs), we design a VQ codebook C for the vectors $\underline{y}_k$. As usual, the ratio M/N, called the *training ratio*, must be reasonably large for effective codebook design. Thus, we obtain a vector quantizer $Q(\cdot)$ which partitions the P-dimensional space of observable vectors $\underline{v}_k$ into N regions, $R_i$, for $i = 1, 2, \cdots, N$. The corresponding code vector $\underline{y}_i$ then is the representative value that approximates any input vector in $R_i$. Then, the vector quantizer approximates an input $\underline{x}_k$ by $Q(\underline{x}_k)$ where $Q(\underline{x}_k) = \underline{y}_i$ when $\underline{x}_k$ is closer to $\underline{y}_i$ than to any other code vector in the codebook C. The use of VQ reduces both the storage requirement and the search time for the nonlinear predictor table. If the resolution of the quantizer is high enough, for a given input vector $\underline{x}_k$, $Q(\underline{x}_k)$ will approximate $\underline{x}_k$ fairly well. Hence the optimal predictor $g(\underline{x}_k)$ in (1) may be approximated by

$$\tilde{g}(\underline{x}_k) = E\{x_{k+1} \mid Q(\underline{x}_k)\} \tag{2}$$

Since we cannot compute expectations directly, we again revert to the training data to estimate the ensemble averages. Assume that the random process $\{x_k\}$ is stationary and ergodic in the sense that $E\{x_{k+1} \mid Q(\underline{x}_k)\}$ can be approximated with sample averages if the number of samples in the training set is sufficiently large. Let $R_i$ denote the partition region containing all training vectors $\underline{v}_k$ which are closer to $\underline{y}_i$ than to any other code vector and $M_i$ be the number of training vectors in $R_i$. Then the approximate prediction mapping $\hat{g}(\underline{x}_k)$ becomes:

$$\hat{g}(\underline{x}_k) = \frac{1}{M_i} \sum_{\underline{v}_k \in R_i} v_{k+1} \text{ if } \underline{x}_k \in R_i . \tag{3}$$

It is important to emphasize that as the resolution of the VQ increases, $\tilde{g}(\underline{x}_k)$ in (2) approaches $g(\underline{x}_k)$ as given by (1). On the other hand, assuming the needed ergodicity property, as the training set size increases, $\hat{g}(\underline{x}_k)$ as given in (3) becomes an increasingly accurate approximation of $\tilde{g}(\underline{x}_k)$. Therefore a nonlinear predictor designed in this way is asymptotically optimal.

The nonlinear predictor can be seen as a special type of vector quantizer in which the encoder and decoder codebooks have different dimensions. This structure is shown in Fig.1. The predictor consists of a P-dimensional VQ encoder with size N, which performs a nearest neighbor search for the best code vector in codebook CB1 that approximates the input $\underline{x}_k$ and specifies this match by the index I. The index I is then applied to the decoder which simply performs a table lookup to find the scalar predicted value in the codebook CB2. Note that CB2 consists of N scalar entries. Prediction is then performed by vector quantization of the observable vector, followed by a table lookup of the corresponding predicted value. The approach to NLP introduced here is in fact a special case of nonlinear interpolative vector quantization (NLIVQ) reported recently in [5], where the theoretical justification of the method is given.

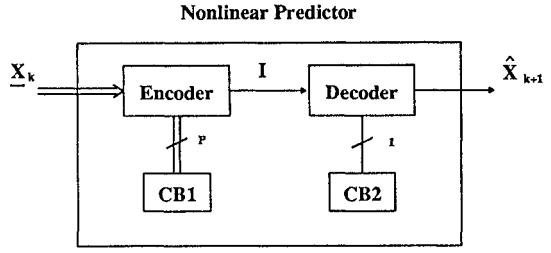


**Figure 1.** Nonlinear Predictor Structure

## 3. APPLICATION OF NLP TO A MOVING AVERAGE PROCESS

The nonlinear predictor was simulated and tested on a synthetic random process which satisfies the stationarity and ergodicity conditions. We used a moving average random process described by the following difference equation:

$$x_k = u_k + a u_{k-1}, \tag{4}$$

where $u_k$ is an i.i.d., uniformly distributed, zero mean, unit variance random process, and $a$ is a real number. For such a process, the optimal linear and nonlinear predictors and their mean square error performances have been previously analytically derived and compared for different values of $a$ and for different prediction orders [6]. In particular, with $a = 1.6$ and a prediction order of $P = 2$, the prediction gain of the optimal nonlinear predictor exceeds the prediction gain for the best possible linear predictor by 0.3385 dB. We trained a nonlinear predictor using a 12-bit codebook and compared the results with an experimental linear predictor, both designed using the same training data. We simulated this process and designed a nonlinear predictor with a training set of size $10^6$. Using the prediction gain as a performance measure, we found that the nonlinear predictor outperformed the linear predictor by 0.3371 dB, which is remarkably close to the theoretical results.

This experiment seems to confirm that for a stationary and ergodic process, VQ based nonlinear prediction approaches optimal prediction, if the quantizer resolution is sufficiently high.

## 4. APPLICATION OF NLP TO SPEECH

### 4.1. Training of the Nonlinear Predictor

In contrast with a moving average process, speech waveforms present a number of difficulties. There is a large amount of variation between different phonemes, utterances, and speakers. For this reason in the design of a nonlinear predictor for speech the training procedure is of great importance. The principal problem is then how to design a nonlinear predictor that would be robust to different speakers and utterances. This can be solved by using a large codebook size, and a large training ratio, especially for the design of the prediction table since the sample averages approach ensemble averages only if the number of samples involved in the averaging is large enough.

We divided a large speech database into a voiced training set and an unvoiced training set in order to separately design predictors for these two types of speech sounds. For the design of the VQ codebooks we used training ratios in the order of 50 to 100. The training set size used to design the prediction table was varied in order to better understand its importance. The corresponding curves illustrating the effect of the codebook size and training set size on the performance of the predictor for voiced and unvoiced speech are presented in Figure 2.

As expected, the prediction gain increases steadily as the codebook size used to quantize the observables increases. On the other hand, as the number of samples over which averaging is performed increases, the entries in the prediction table approach the ensemble

averages given by (2) and the prediction gain increases. However, since speech is not a stationary process, the values in the prediction table obtained by sample averages will not necessarily converge to the optimal prediction values. In terms of the segmental prediction gain, for voiced speech, the nonlinear prediction gain surpasses the linear prediction gain for very large codebooks (above 14 bits) and training set sizes. On the other hand, for unvoiced speech, a nonlinear predictor with small memory size and search complexity performs consistently better than a linear predictor.

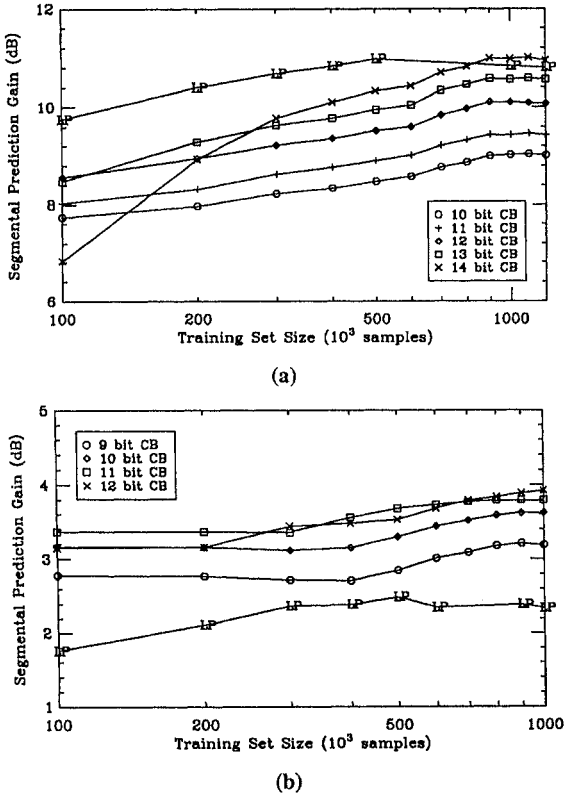


Figure 2. Comparison of 5th order LP and NLP: (a) voiced segments; (b) unvoiced segments.

### 4.2. Prediction Order and Codebook Size

If the prediction was performed directly using equation (1) (i.e. without quantization), then, clearly, increasing the prediction order would lead to an increase in the prediction gain. But the quantization of the observable vector leads to a trade-off between the quantizer resolution and the prediction order. For relatively small codebook sizes, it is better to use a low prediction order (low vector dimension) in order to accurately represent the input observable vector (high resolution VQ). Conversely, for large codebook sizes, the resolution is sufficiently high for higher dimensional vectors, so that some benefit can be obtained from increased memory size.

It must be noted that, in general, the benefit obtained from increased prediction order is not very large. We used voiced speech to train NLP's with different prediction orders and codebook sizes, and tested the predictors with a voiced speech segment outside of the training set. The resulting prediction gains are summarized in Figure 3. The results show that for codebook sizes below 12 bits the predictors with smaller prediction order perform better. Table 1 compares the results obtained from second order linear and nonlinear predic-

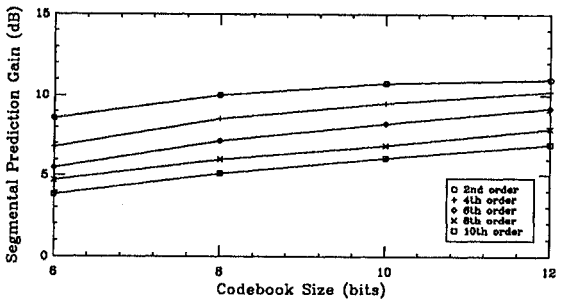


Figure 3. Nonlinear Prediction Gain vs. Prediction Order

tion with speech test sentences from male and female speakers and for two different codebook sizes (1024 and 4096). Both the prediction gain (PG) and the segmental prediction gain (SEGPG) in dB are shown.

Table 1. Prediction gains for 2nd order LP and NLP (in dB)

| Prediction Type | Female | | Male | |
|---|---|---|---|---|
| | PG | SEGPG | PG | SEGPG |
| LP | 9.76 | 10.71 | 9.43 | 9.21 |
| NLP (VQ=10bits) | 9.70 | 11.48 | 9.14 | 9.68 |
| NLP (VQ=12bits) | 9.80 | 11.73 | 9.29 | 9.88 |

### 4.3. Spectral Flattening

It is significant to note that, while the prediction gain for a 12-bit 5th order NLP is slightly less than that of the LP, NLP appears to perform better in terms of perceptual quality. In informal listening tests, we observed a major drop in the intelligibility of the prediction residual for NLP compared to LP. The spectrum of the residual, shown in Figure 4., confirms this observation. The NLP residual spectrum is considerably flatter than the LP residual spectrum. In particular, we observe that NLP is successful in eliminating the higher harmonics of the fundamental pitch frequency. Due to this fact, we find that short term NLP substantially reduces the need for pitch prediction.

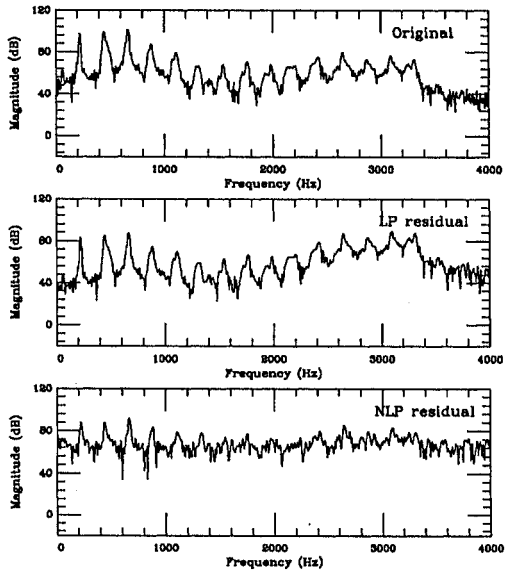


Figure 4. Spectra of original and residual with LP and NLP

## 5. APPLICATION TO DPCM

In order to assess the performance of NLP in a simple speech coder, we incorporated it into standard DPCM, where instead of the usual linear predictor we used a nonlinear predictor. The system configuration is shown in Figure 5. Voiced speech was used as the input. The prediction order was 5, the codebook size used for NLP was 4096. The system was tested with 3, 4, and 5 bit nonuniform scalar quantizers. The curve in Figure 6 compares the SNR of the DPCM system for linear and nonlinear predictors. We observe that nonlinear prediction outperforms linear prediction especially for low bit rates.

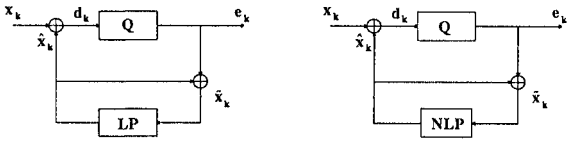


**Figure 5.** DPCM Systems with LP and NLP

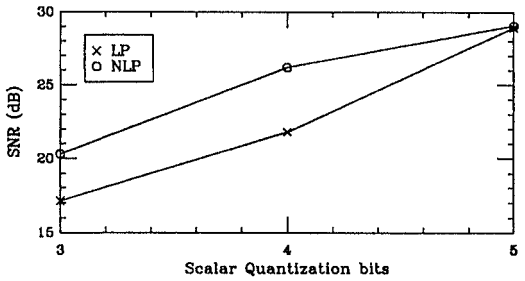


**Figure 6.** SNR Performance of DPCM Systems with LP and NLP

## 6. CONCLUDING REMARKS

The proposed nonlinear prediction method may be viewed as a numerical analysis method for optimal prediction which makes use of vector quantization techniques. NLP is asymptotically optimal for the prediction of stationary and ergodic processes. For such processes, if the codebook size is large enough, NLP outperforms LP in terms of the prediction gain. For nonstationary signals such as speech however, an improved prediction gain is achieved only for very large codebooks so that storage and computational complexity become potential problems. These problems may be avoided by using structured codebooks, as in multistage or tree-structured VQ.

The NLP method used here was based on the use of conventional VQ design methods for quantizing the observable input vector. Some performance improvement can be obtained by jointly optimizing the VQ codebook and the prediction table. This and other improvements are currently being studied by the authors.

For application to speech coding, adaptive prediction is usually needed. NLP can be made adaptive by continuously updating the prediction table according to the input but this introduces an excessive amount of side information if forward adaptation is used. This suggests that a simple parametric model of NLP might be superior to our method for forward adaptive NLP. Another difficulty arises in applying NLP to analysis-by-synthesis speech coding methods. It is possible that NLP could lead to a nonlinear synthesis filter for such coders. However, superposition can not be used, so that the search complexity will not benefit from the usual separation of the zero-input response and zero-state response with a linear synthesis filter.

This work represents only a preliminary study of a nonlinear prediction method with possible application to speech processing. It is evident that this subject is a difficult and challenging one with many unsolved problems remaining at this time. Many further studies are needed to explore the potential of NLP for speech processing.

### References


[1] N. Tishby, "A Dynamical Systems Approach To Speech Processing," *Proc. IEEE Conf. Acoust., Speech, Sig. Processing*, pp. 365-368, Albuquerque, April 1990.

[2] E. Biglieri, "Theory of Volterra Processors and Some Applications," *Proc. IEEE Conf. Acoust., Speech, Sig. Processing*, pp. 294-297, Paris, France, May 1982.

[3] G. L. Sicuranza and G. Ramponi, "Adaptive nonlinear prediction of TV image sequences," *Electronics Letters*, vol. 25, pp. 526-527, 13 April 1989.

[4] R. E. Bogner and T. Li, "Pattern Search Prediction of Speech," *Proc. IEEE Conf. Acoust., Speech, Sig. Processing*, pp. 180-183, Glasgow, UK, May, 1989.

[5] A. Gersho,, "Optimal Nonlinear Interpolative Vector Quantization," *IEEE Trans. Commun.*, vol. vol. COM-38, no. 9 , September, 1990.

[6] L. Shepp, D. Slepian, and A. Wyner, "On Prediction of Moving Average Processes," *Bell Systems Technical Journal*, pp. 367-415, May 1980.