NONLINEAR LONG-TERM PREDICTION OF SPEECH SIGNALS

Martin Birgmeier[*] *Hans-Peter Bernhard* *Gernot Kubin*[†]

Institute of Communications and Radio-Frequency Engineering
Vienna University of Technology, Gusshausstrasse 25/E389, A–1040 Vienna, Austria
Email: Martin.Birgmeier@nt.tuwien.ac.at, {G.Kubin,H.P.Bernhard}@ieee.org

ABSTRACT

This paper presents an in-depth study of nonlinear long-term prediction of speech signals. While previous studies of nonlinear prediction focused on short-term prediction (with only moderate performance advantage over adaptive linear prediction in most cases), successful long-term prediction strongly depends on the nonlinear oscillator framework for speech modeling. This hypothesis has been confirmed in a series of experiments run on a voiced speech database. We provide results for the prediction gain as a function of the prediction delay using two methods. One is based on an extended form of radial basis function networks and is intended to show what performance can be reached using a nonlinear predictor. The other relies on calculating the mutual information between multiple signal samples. We explain the role of this mutual information function as the upper bound on the achievable prediction gain. We show that with matching memory and dimension, the two methods yield nearly the same value for the achievable prediction gain. We try to make a fair comparison of these values against those obtained using optimized linear predictors of various orders. It turns out that the nonlinear predictor's gain is significantly higher than that for a linear predictor using the same parameters.

1. INTRODUCTION

Nonlinear prediction of speech signals has been subject to several studies over the past five years with highly varying outcomes, for an extended list of references see [6, section 2.2]. Still, an upsurge in the practical use of nonlinear prediction has not yet arrived; for a notable exception, see [7]. This contribution attempts to clarify what nonlinear prediction can do for speech processing. To this end, we first introduce a systematic description of *predictor characteristics*.

Linear predictors are usually specified by their *order* and their *prediction delay* L which is either one sampling interval (short-term prediction) or one pitch period (long-term prediction).

Nonlinear predictors require to break up the simple order concept into three different features: The *memory span* M denotes the duration (in physical time units) of the speech signal history stored in the predictor state memory. The *dimension* D denotes the number of predictor state vector components used as the input to the nonlinear map which computes the predicted signal value. Due to results from nonlinear dynamical systems theory, the dimension can often be much lower than the memory span divided by the sampling interval, i.e., a subsampled version of the predictor memory serves as the input to the nonlinear map implemented by the predictor. The *complexity* C of the predictor structure is not defined by its dimension alone, but one needs to take the degree of the nonlinearity into account (e.g., polynomial degree, number of hidden nodes in neural networks, etc.).

We have selected nonlinear long-term prediction of sustained voiced speech sounds as the experimental paradigm. This choice is motivated from the following:

- We have shown previously that linear autoregressive models are fully adequate for unvoiced speech [6, section 2.1].
- Many studies including our own [5, 4] have shown that it is difficult to design nonlinear short-term predictors for voiced speech which would substantially outperform linear predictors. This result remains unchanged even when the nonlinear predictor operates on the linear prediction residual.
- Continuous speech is characterized by frequent bifurcation-type transitions such as voiced/unvoiced transitions etc. No predictor can be expected to operate across such boundaries without performance degradation. As our focus is on the exploration of the margins of predictability to be gained by nonlinear methods, we want to exclude the influence of nonstationarity explicitly. Working with sustained speech sounds allows us to establish *upper bounds* on the prediction gain that will carry over to continuous speech.

In the following, we present two strategies to establish such upper bounds. Section 2 discusses the design and training of *radial basis function* network based *autoregressive models* (RBF-AR) as a constructive vehicle to approach the upper bounds of predictability. Section 3 presents a model-free, nonparametric algorithm which estimates the mutual information function of the speech signal and which directly relates to the maximum achievable prediction gain. Results for both strategies are summarized in section 4 where long-term prediction over a whole range of prediction delays (from zero to several pitch periods) is studied. Section 5 presents our conclusions.

2. NONLINEAR PREDICTION BY RBF-AR NETWORKS

We use RBF-AR networks [8] as nonlinear predictors since they have proved to provide very good nonlinear approxima-

[*] Martin Birgmeier is now with Philips Speech Processing, Computerstrasse 6, A–1101 Vienna, Austria.

[†] Support through grant P08779-TEC from the Austrian Science Fund (FWF) is gratefully acknowledged.

tion capabilities, and efficient training algorithms do exist [4].

The RBF-AR network can be described as a general mapping structure which computes a *locally linear* map from the vector[1] of inputs $\mathbf{i} = [i_1 \dots i_D]^T$ to the scalar output o:

$$o = \sum_{k=0}^{K} \varphi_k(\mathbf{i}) \sum_{l=0}^{D} w_{l,k}\, i_l \tag{1}$$

Here we have defined

$$i_0 \equiv 1 \quad \text{and} \quad \varphi_0 \equiv 1. \tag{2}$$

In addition, $\varphi_k(\mathbf{i})$ is the activation of hidden node k, which is taken to be a multidimensional Gaussian with center vector $\mathbf{t}_k$ and scalar covariance σ_k,

$$\varphi_k(\mathbf{i}) = \exp\left(-\frac{1}{2} \cdot \frac{\|\mathbf{i} - \mathbf{t}_k\|^2}{\sigma_k^2}\right). \tag{3}$$

If we use a vector of lagged speech signal samples $\mathbf{y}(t) = [y(t), y(t-\tau_1), \dots, y(t-\tau_{D-1})]^T$ for the input vector $\mathbf{i}$, we see from eq. (1) that the RBF-AR network computes a weighted sum of FIR-filtered versions of the input signal.

Training of the RBF-AR network is accomplished in the usual mixed unsupervised/supervised fashion using a set of input and target values, $\mathbf{i}(t)$ and $d(t)$. t numbers the set of training examples, which in our case are taken from a certain part of some test speech signal. First, a clustering procedure is used to determine the hidden nodes' centers $\mathbf{t}_k$ such that they represent the training vectors $\mathbf{i}(t)$ with minimum (quadratic) distortion. Each Gaussian's covariance σ_k^2 is chosen proportionally to the mean square distortion of the training vectors closest to center k. In the second step, the output weights $w_{l,k}$ are computed to minimize the energy of the residual,

$$\mathcal{E} = \sum_t \|d(t) - o(t)\|^2. \tag{4}$$

The target value $d(t)$ is given by the signal sample to be predicted, $y(t+L)$, and the output o accordingly is the prediction $\hat{y}(t+L)$ itself. The quotient of the target energy $\sum_t \|d(t)\|^2$ and the cost function is the *prediction gain* (cf. below). However, in order to verify the *generalization ability* of the predictor, we may alternatively use a separate set of test vectors; then the parameters of the network are calculated as just described, but the prediction gain is redefined by re-calculating both the cost function and the target energy using the test instead of the training set.

Varying the prediction delay from zero to some maximum value results in a characteristic variation of the prediction gain. In sec. 4, we shall show that it nearly reaches the upper bound on prediction gain estimated through calculation of the mutual information between the input vector $\mathbf{y}(t)$ and the target value $y(t+L)$.

3. MUTUAL INFORMATION AND PREDICTION GAIN

For a given *stochastic* process, we may ask what the *maximum achievable* prediction gain is. In prediction, we make use of the information contained in past samples to compute an estimate for a future sample. Again, we denote the sample to be predicted by $y(t+L)$, and the memory of past samples by the vector $\mathbf{y}(t)$, as in the nonlinear predictor above. For an additive model, the error between actual and estimated value is defined by

$$e(t+L) = y(t+L) - \hat{y}(t+L) = y(t+L) - f(\mathbf{y}(t)). \tag{5}$$

The performance measure is defined to be the prediction gain. Under the assumption of stationarity, we get for the prediction gain G as a function of the prediction delay L

$$G(L) = 10 \log_{10} \frac{\sigma_y^2}{\sigma_e^2(L)} \quad [\text{dB}] \tag{6}$$

where σ_y^2 denotes the process power, and the error power σ_e^2 depends on the prediction delay L. In [1], we have shown that an upper bound on the prediction gain $G(L)$ can be found using information theoretic considerations. This upper bound is *independent of the functional form of the predictor f*. It is given by the mutual information between $y(t+L)$ and $\mathbf{y}(t)$, i.e., $I(y(t+L); \mathbf{y}(t))$, corrected by the entropy difference Δ between $y(t+L)$ and a Gaussian random variable with the same variance σ_y^2:

$$G(L) \le 20 \log_{10}(2) \cdot \big(I(y(t+L); \mathbf{y}(t)) + \Delta\big) \quad [\text{dB}] \tag{7}$$

Here, the mutual information $I(y(t+L); \mathbf{y}(t))$ quantifies the information shared between two random variables $y(t+L)$ and $\mathbf{y}(t)$. It expresses how much we know about the predicted sample $y(t+L)$, if we consider the essential past contained in $\mathbf{y}(t)$. The additional term Δ describes the distance of the amplitude distribution of $y(t)$ from a Gaussian, which has the minimum variance for a given entropy $h(y)$. We have

$$\Delta = \frac{1}{2}\, \log_2(2\pi e \sigma_y^2) - h(y). \tag{8}$$

To estimate both the mutual information and the entropy difference, we have developed a fast algorithm [2] whose computational complexity increases only linearly with the input vector dimension D.

Again, the upper bound on prediction gain G depends on the prediction delay L.

4. RESULTS

In the following, we discuss one specific example from our database of sustained voiced speech sounds in detail. It has been produced by a male speaker and sampled at 48 kHz to achieve the high temporal resolution which optimizes long-term prediction performance. We use a linear predictor of order 144 as our baseline reference. This predictor has the same memory ($M = 3$ ms) as an order 24 predictor at a sampling frequency of 8 kHz. We call this a *short-memory* linear predictor as the memory spans still significantly less than the pitch period of the speaker. We also show results for linear predictors of orders 33 and 1 (the latter corresponds to the usual setting for oversampled pitch predictors), to provide some further reference points. The linear predictors are optimized according to the same least squares criterion as in eq. (4).

We estimate the upper bound on the prediction gain using the mutual information algorithm with a dimension $D = 3$ and lags $\tau_k = \{16, 32\}$. These values result from an optimum state-space reconstruction procedure based on the mutual information algorithm [2].

For the RBF-AR nonlinear predictor, we use the same dimension ($D = 3$) and lags, such that we can easily compare

[1] Vector quantities are denoted by boldface type.

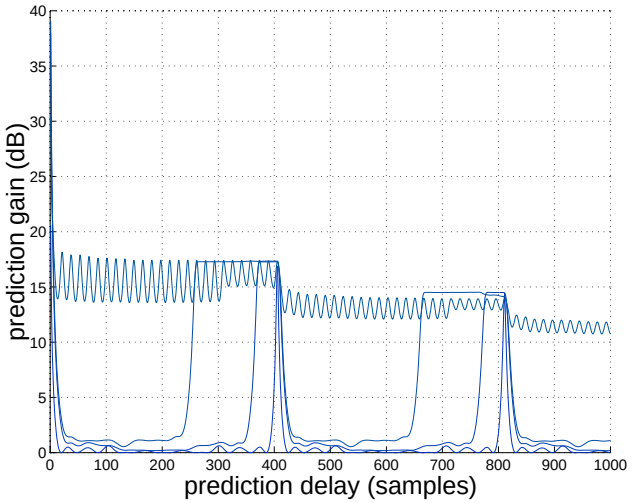

Figure 1. Linear prediction gain vs. prediction delay: Long-memory linear predictor (oscillating behavior); short-memory linear predictors, $D = 1/33/144$ (smooth curves with peaks at multiples of the pitch period).

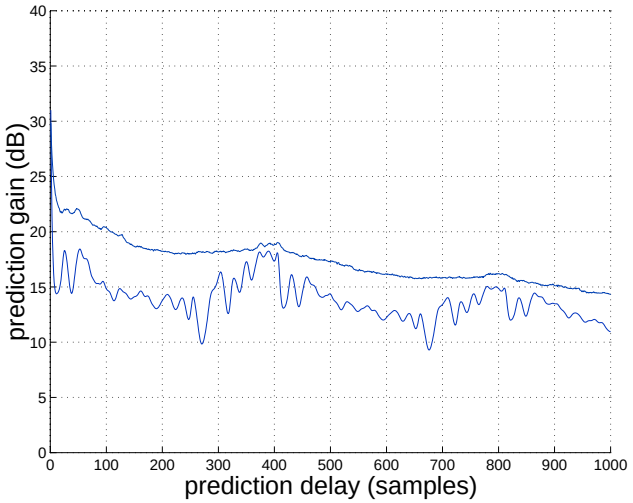

Figure 2. Nonlinear prediction gain vs. prediction delay: Upper bound obtained from the mutual information estimate (top); short-memory RBF-AR predictor, $D = 3$ (bottom).

the results to those obtained using the previous method. The network comprises $K = 40$ centers.

Finally, we also evaluate the performance of a *long-memory* linear predictor whose memory covers a full pitch period. For the fundamental frequency of 120 Hz, this requires to use a memory spanning at least 8.3 ms. To keep the solution for the predictor coefficients computationally tractable, we use lags $\tau_k = 16k, 1 \leq k \leq D-1$, and set the predictor dimension to $D = 32$, corresponding to a memory of around 10 ms. This choice of lags matches those of the mutual information and RBF-AR algorithms.

Figs. 1 and 2 show a comparison of the prediction gain vs. prediction delay calculated using these methods. Here, we calculate the prediction gain for the training sequence itself, which consists of samples $42001 \ldots 52000$ of the speech signal. We note the following results:

- The short-memory linear predictor shows a characteristic periodicity in prediction gain vs. prediction delay. This period corresponds to the fundamental frequency. Except for prediction delays which are multiples of the pitch period, the prediction gain is practically zero.
- As the signal is almost periodic, a straightforward linear predictor can be built by simply copying samples from the previous period. This requires the use of a long memory if arbitrary prediction delays are to be handled. The gain of this linear predictor is approximately constant over the whole pitch period. Due to the lags being multiples of 16, we have oscillations with a period of 16. We can easily explain this effect by noting that the long-memory predictor achieves most of its prediction gain by copying samples from the previous pitch period. But the predictor input vector only includes every 16th sample, which results in the observed oscillating prediction gain. (Note that there are no such oscillations for the short-memory linear predictors which do not use subsampling.) However, it should be clear that even using all samples, only the oscillations would vanish, with no additional prediction gain over the maxima of the oscillations.
- The prediction gain of the long-memory linear predictor falls in steps with each period, approximately paralleling the falling upper bound estimated by the mutual information algorithm. Within a source–filter model for speech, this stair-case behavior can be explained by locating the major nonlinearity of speech generation with the excitation pulse generator driving the filter.
- The RBF-AR predictor achieves at least the same prediction gain as the linear predictors despite being fed only three past samples from a memory of only 0.67 ms. It is mostly unaffected by the larger lags used for the construction of the state memory: Whereas the (long memory) linear predictor exhibits oscillations with a period corresponding to the lags (i.e., 16), this is much less the case for the nonlinear predictor. Furthermore, the decay of the nonlinear prediction gain follows the upper bound on the prediction gain without the step changes seen in the long-memory linear predictor.

Using a different test sequence (consisting of samples $53001 \ldots 63000$ of the same signal) even more supports the nonlinear prediction approach: There, all types of linear predictors incur a larger decrease in prediction gain (compared to the case where test and training sequences are the same) than the nonlinear predictor (cf. fig. 3). The RBF-AR predictor surpasses the prediction gain of the long-term linear predictor by up to 5 dB (e.g., for a prediction delay $L = 500$). Our interpretation is that the nonlinear predictor is both less susceptible to noise as well as to temporal variations in the signal, and better captures the nonlinearities present in the speech generation mechanism.

5. CONCLUSIONS

The most attractive result here certainly is the strong correspondence between the prediction gain values obtained using two very different nonlinear methods, namely the calculation of the mutual information function of signals, and prediction via an RBF-AR network. The values match rather well over a range of prediction delays. Hence, these methods can be seen as complementary to each other, and results obtained using one can be expected to be similar for the other.

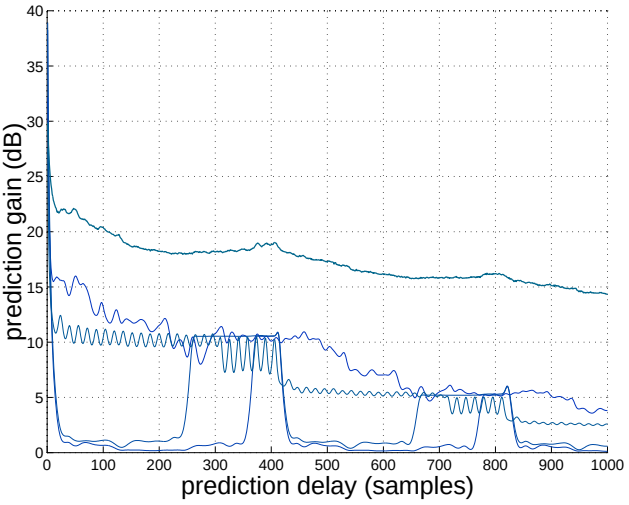

Figure 3. Prediction gains for a different test sequence: Shown are two short-memory linear predictors (peaking only at multiples of the pitch period), a long-memory linear predictor (oscillating behavior), the RBF-AR predictor with largest gain especially outside the pitch intervals, and the upper bound estimated by the mutual information algorithm (which is unchanged from fig. 2).

Another interesting feature of nonlinear long-term prediction is its slow and almost monotonic performance decay with increasing prediction delay. This performance is achieved even with short memory and low dimension, both for the maximum achievable gain and with the constructive RBF-AR method.

A point for critique might be the apparent difference in complexity between the linear and nonlinear predictors. In fact, the RBF-AR predictor in the configuration used in this paper comprises a total of 324 trained parameters, 160 in the hidden layer, plus 164 in the (linear) output layer. Conversely, the linear predictors shown here have a complexity (equal to the number of tap weights) of between 1 and 144. There are, however, still several points in favor of the nonlinear approach:

- Building a linear predictor with full performance over a range of delays requires different taps for each value of the delay, which is in contrast to the nonlinear predictor which operates on the same taps for all delays. Alternatively, we may use a dense set of taps over a full pitch period, which however would increase the linear predictor's complexity beyond that of the nonlinear one.
- It is possible to use a fixed hidden layer of Gaussian nodes, with only a minor decrease in performance. This reduces the complexity of the RBF-AR predictor to 164.
- For the case of equal training and test sets, the long-term linear predictor's performance matches that of the nonlinear nearly everywhere (cf. figs. 1 and 2). However, this is due to overtraining, as can nicely be seen in the last figure. Clearly, the RBF-AR predictor much better captures the underlying nonlinearities inherent in the speech generation mechanism. This is important in situations where the predictor operates on previously unseen data, which is the case in most practical applications.
- Finally, the map produced by a nonlinear (usually one-step) predictor may be *iterated*[2] [3]. The resulting time series resembles the training signal for at least several periods, and does not decay to zero as would be the case for a standard (stable) AR model without an excitation signal.

The last point suggests that nonlinear methods would allow to extrapolate voiced speech signals over relatively long intervals which could be exploited in the design of vector predictive coders or for the restoration of lost speech frames in mobile radio or packet-switched transmission systems.

Let us finally note that this study, too, arrives at the well-known result that a linear predictor already provides a very good approximation to the speech waveform. Any additional performance improvements are either rather moderate or can be gained only using complex structures. This additional complexity can, however, be tolerated if the speech modeling approach shifts from prediction to synthesis. The results reported here provide further support for the nonlinear oscillator model of speech [6].

REFERENCES

[1] Hans-Peter Bernhard. Determining the predictability of signals. In *Proceedings of the IEEE Digital Signal Processing Workshop*, Loen, Norway, 1996.

[2] Hans-Peter Bernhard and Gernot Kubin. A fast mutual information calculation algorithm. In M. J. J. Holt, Colin F. N. Cowan, Peter M. Grant, and William A. Sandham, editors, *EUSIPCO-94*, volume Signal Processing VII: Theories and Applications, pages 50–53, Amsterdam, September 1994. Elsevier Science Publishers.

[3] Martin Birgmeier. A fully Kalman-trained radial basis function network for nonlinear speech modeling. In *Proc. IEEE ICNN*, volume 1, pages 259–264, 1995.

[4] Martin Birgmeier. *Kalman-Trained Neural Networks for Signal Processing Applications.* Doctoral dissertation, Vienna University of Technology, Vienna, Austria, 1996.

[5] Martin Birgmeier. Nonlinear prediction of speech signals using radial basis function networks. In *Proc. VIII European Signal Processing Conference, EUSIPCO'96*, pages 459–462, Trieste, Italy, September 1996.

[6] Gernot Kubin. Nonlinear processing of speech. In W. B. Kleijn and K. K. Paliwal, editors, *Speech Coding and Synthesis*, chapter 16. Elsevier Science B. V., 1995.

[7] Jes Thyssen. *Non-Linear Analysis, Prediction, and Coding of Speech.* Ph.D. thesis, Technical University of Denmark, Electronics Institute, DK-2800 Lyngby, Denmark, July 1995.

[8] J. M. Vesin. Local models for nonlinear signal processing. In D. Docampo and A. R. Figueras, editors, *Adaptive Methods and Emergent Techniques for Signal Processing and Communications*, pages 384–390. Universidad de Vigo, Vigo, Spain, June 1993. Based on the proceedings of COST 229 action WG 1 and 2 workshop.

[2] This recovers the original definition of the RBF-AR network in [8].