SPEECH SYNTHESIS USING WARPED LINEAR PREDICTION AND NEURAL NETWORKS

Matti Karjalainen[1], *Toomas Altosaar*[1] *and Martti Vainio*[2]

[1]Helsinki University of Technology
Lab. of Acoustics and Audio Signal Processing
P.O.Box 3000, FIN-02015 HUT, Finland
matti.karjalainen@hut.fi, toomas.altosaar@hut.fi

[2]University of Helsinki
Department of Phonetics, P.O.Box 35
FIN-00014 University of Helsinki, Finland
martti.vainio@helsinki.fi

ABSTRACT

A text-to-speech synthesis technique, based on warped linear prediction (WLP) and neural networks, is presented for high-quality individual sounding synthetic speech. Warped linear prediction is used as a speech production model with wide audio bandwidth yet with highly compressed control parameter data. An excitation codebook, inverse filtered from a target speaker's voice, is applied to obtain individual tone quality. A set of neural networks, specialized to yield synthesis control parameters from phonemic input in specific contexts, generate the detailed parametric controls of WLP. Neural nets are also used successfully to compute the prosodic parameters. We have applied this approach in prototyping highly improved text-to-speech synthesis for the Finnish language.

1. INTRODUCTION AND MOTIVATION

After a long period of successful developments in text-to-speech (TTS) synthesis, voice quality still remains a challenge. No practical technique yields wide audio bandwidth, near human quality, and individual sounding speech.

Our effort in this study was to find a strategy to improve TTS synthesis for the Finnish language. Earlier achievements were first based on traditional formant synthesis with rule-based control, SYNTE 2 and 3 [1], and then concatenation synthesis called microphonemic synthesis [2] similar to the PSOLA technique [3]. Concatenative synthesis, based on samples from human speech, easily captures the features from individual speakers. In order to approach full naturalness, however, a huge inventory of samples in different contexts is needed. The algorithms to select concatenative units and to join them in synthesis tend to become complex.

Source-filter models for speech synthesis, such as those used in linear prediction, have more flexibility and allow for easy analysis of control data. The problem remains how to code the excitation (source) and the filter control parameters in a compact way and be able to recompute them from phonemic/phonetic information. Hand-tuned rules and tables, as used in early synthesis, cannot produce highest quality speech. Tables of parameter trajectories have similar problems as concatenative synthesis: the size of such inventories grows beyond practical limits when contextual details are included. Among the techniques that are used to compress and generalize control parameter information through learning are, e.g., neural networks, hidden Markov models, and fuzzy or neuro-fuzzy rule systems.

The requirements dictating the choice of methods in our study were to obtain very high quality individual sounding synthesis, wide audio bandwidth (> 10 kHz), easy automation of tuning the synthesis to individual speakers using a speech database, moderate memory and processor requirements in implementation, easy integration of audio and visual synthesis (talking head), and preferably as much language independence as possible.

We first discarded the waveform concatenation methods due to the complexity of sample collection and even more due to the difficulty of controlling the detailed contextual effects. An LPC-like source-filter model was found to be more attractive. The success of this approach depends on several factors. A relatively small inventory of source excitations for the synthesis of all phones in the target language should be easily aquirable. The filter parameters should be represented compactly in a form that is suitable to automatic training, e.g., using neural nets.

The problem of ordinary linear prediction with wide bandwidths is that a high filter order is required and the high-frequency portion reserves too much resolution. For example, with a sampling rate of 22 kHz, the traditional rule of thumb leads to an LP filter order of about 24 and most of the filter parameters focus on frequencies above the important formant range below 3.4 kHz [4]. This problem was elegantly solved in our case by adopting warped linear prediction (WLP) [5], utilizing non-uniform frequency resolution and allowing moderate filter orders of 10 – 14 almost independently of the sampling rate.

The compactness of synthesis parameter information helped in modeling the generation of these parameters from phonemic input data. Neural networks have been shown to perform this mapping but not without problems. Possible candidates of neural nets are multilayer feedforward nets with phoneme string and synthesis position input, time delay neural networks (TDNN) with time frame input, and recurrent networks, see [6] and references in it. Our experience with neural nets has shown that for detailed modeling, specialization of nets is useful so that each individual net is applied only in a specific context.

In this paper the main features of our approach are described. We have studied the level of voice quality achievable using WLP and specialized neural nets. A full scale synthesizer is under development but already the experiments indicate that a very natural and individual sounding TTS synthesis, practical for implementation, can be obtained.

2. WARPED LINEAR PREDICTION

The first systematic formulation of warped linear prediction was presented by Strube [7]. Later, Laine et al. [5] have studied various formulations of efficient WLP. The idea of a warped frequency

0-7803-4428-6/98 $10.00 © 1998 IEEE

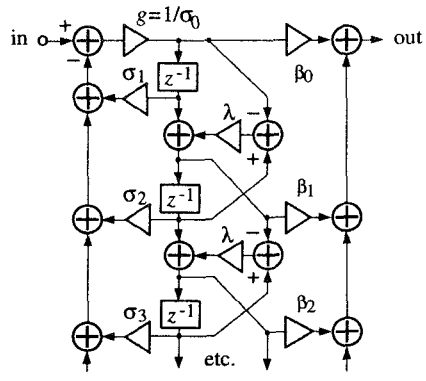

Figure 1: *A realizable WIIR structure with first-order allpass delays and a single unit delay.*

scale and related resolution is based on using allpass sections instead of unit delays in DSP structures, i.e.,

$$\tilde{z}^{-1} = D_1(z) = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}} \quad (1)$$

where λ, $-1 < \lambda < 1$, is a warping parameter and $D_1(z)$ is a warped (dispersive) delay element. With a proper value of λ, the warped frequency scale shows a good match to the psychoacoustically defined Bark scale [8], thus optimizing the frequency resolution from the point of view of auditory perception. For example, with a sampling rate of 22 kHz, Bark warping is obtained using λ = 0.63.

WLP analysis is easily realized by modifying only the autocorrelation computation using a version where unit delays are replaced by allpass sections. The same holds for inverse filtering to obtain the residual (excitation) signal. The synthesis filter, however, cannot be realized in such a simple manner since in recursive structures the replacement of Eq. (1) results in delay-free loops. Techniques to avoid this problem are discussed, e.g., in [9]. The filter structure shown in Fig. 1 has been used in our WLP synthesis experiments. The original (warped) denominator coefficients are mapped to new coefficients σ_i that are used as feedback coefficients. Otherwise, the WLP analysis and synthesis techniques are the same as with ordinary linear prediction.

The advantage gained when using Bark warping is that in wideband synthesis the filter order can be reduced remarkably without sacrificing the frequency resolution at low frequencies. At high frequencies the spectral resolution is worse, nevertheless this is exactly how hearing functions. We have experimentally evaluated the voice quality of WLP and normal LP for various filter orders when the sampling rate is 22 kHz. Ordinary LP yields good quality with orders of 20–24 while WLP works comparably with orders of 10–14. Figure 2 shows synthesis filter responses for a vowel spectrum (Finnish /a/) using ordinary LP and WLP.

The main advantage of WLP over LP is the compression of control parameter data which helps in the training of neural nets to generate these parameters. A lower filter order is also advantageous for fast computation but this is counteracted by the inherently more complex structure of the warped IIR filters (Fig. 1). It is also possible to expand the WIIR filter structure into an ordinary direct form IIR filter but the WIIR structure is numerically more robust as discussed in [9]. Since on modern processors (DSPs, Pentium, PowerPC) such filters consume only a few per cent of CPU resources, the robust and straightforward WIIR structure of Fig. 1 has been used in our synthesizer.

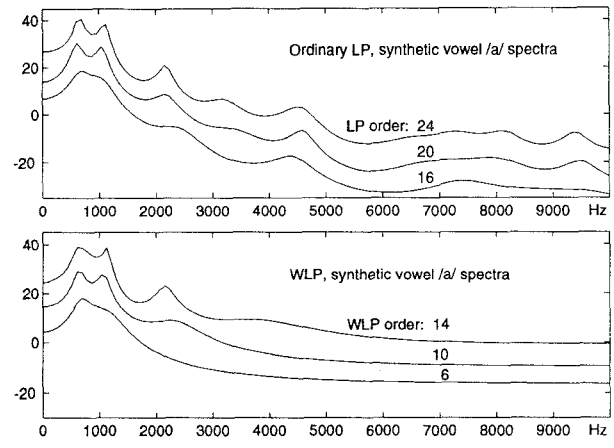

Figure 2: *LP and WLP spectra of vowel /a/ for different filter orders.*

As a final representation for WLP filter control parameters we used WLP lattice coefficients (reflection coefficients). This was due to the desirable characteristics of reflection coefficients whereby the stability of the synthesis filter can be guaranteed by limiting the coefficient values in the range $(-1, +1)$. This well-defined range of parameters also helps when generating them using neural networks as will be described below. The warped reflection coefficients were converted by the standard step-up procedure to warped polynomial coefficients for controlling the filter structure shown in Fig. 1.

3. SYSTEM CONFIGURATION

Figure 3 illustrates the block diagram of the synthesizer. The WLP synthesis structure consists of an excitation codebook, an overlap-add concatenator of excitation signals for pitch and duration generation, a gain multiplier, and a warped LP filter (WIIR synthesis filter). This voice synthesis chain is controlled by sets of context-specialized neural networks (netsets), for filter parameters, pitch, duration, and gain controls. Neural network inputs as well as the selection of a proper network within a netset is based on the phoneme to be synthesized, its phonemic context as well as other contextual information.

The input data in Fig. 3 is a string of phonemes. The preceding grapheme-to-phoneme conversion, which is exceptionally simple in the Finnish language, is not shown and discussed here. The phoneme to be synthesized as well as the neighbouring phonemes and other contextual information are used to compute numerically coded context vectors for the neural network inputs. Each netset in the diagram is a set of context-specialized feedforward neural nets. Only one of the networks within a netset is activated at any one time, depending on the unit to be synthesized and its context.

4. SPECIALIZED NEURAL NETS FOR FILTER PARAMETER CONTROL

Our experience with feedforward neural nets has shown that, instead of using a single large network, a complex input-output map-

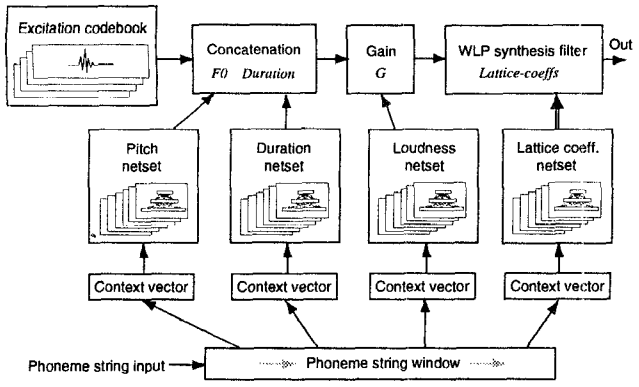

Figure 3: *Configuration of the speech synthesis system using warped linear prediction and specialized neural nets.*

ping is more easily and precisely learnt by a set of specialized networks, each one contributing only within a specific region of a multidimensional input data space. The same strategy, the utilization of specialized detectors and generators, is also found in biological and human neural systems.

In speech processing, this principle can be utilized in various ways. Earlier we have shown that the performance of prosodic feature models is improved when the mapping from phoneme string input to duration, pitch, or signal gain is properly partitioned [12].

4.1. Network Input/Output Coding

The input to the synthesizer consist of phonemic information (a string of phonemes converted from a string of graphemes) as well as phonetic information (e.g., factors affecting prosody indicated by punctuation). This symbolic information must be converted into numerical form to allow neural networks to be utilized in the generation of synthesis control parameters. We have used three types of information to constitute the input to the networks:

1. The phoneme to be synthesized is coded as three real numbers representing the broad class (e.g., vowel), the fine class (e.g., /a/), and the quantity (e.g., short vs. long). Neighboring phonemes (e.g., three previous as well as three future phonemes) are also coded in a similar way and thus the network is introduced to the specific context in which the phoneme to be generated exists. Therefore $(3 + 1 + 3) \cdot 3 = 21$ elements of the input vector are generated from the phonemic information in the above mentioned way.
2. The relative position of the phoneme to be synthesized in the word as well as the number of phonemes in the word are coded as two real numbers. This improves performance since the network then can infer stressed/unstressed syllables.
3. The relative point (time) within the phoneme to be synthesized is coded as a number between 0.0 and 1.0. This allows for the microstructure to be generated further improving the quality of the synthesis.

These $21 + 2 + 1 = 24$ values are combined into one input vector. Associated with each input vector is a target vector that indicates the desired output values of a neural net, i.e., the WLP lattice coefficients.

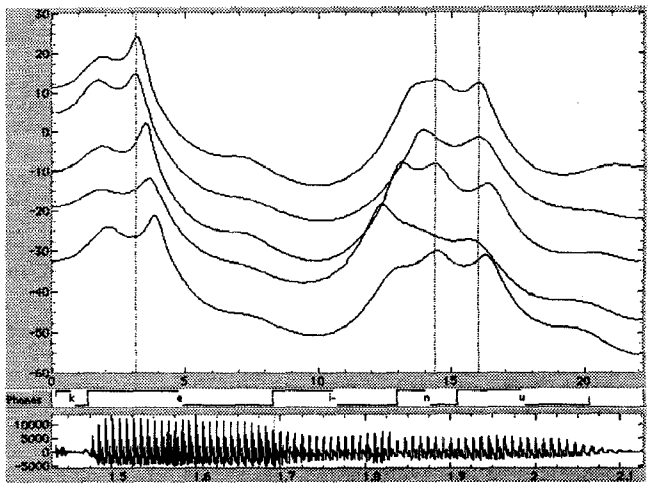

Figure 4: *WLP spectra (dB vs. Bark scale) at a certain time instant in an [e]–[i] transition of word /keinu/. The top curve is the target spectrum and the other ones are neural net generated cases (Table 1) in order of decreasing specialization.*

4.2. Network Specialization

Phoneme networks model the WLP coefficients at any temporal point within a phoneme. However, when moving across phoneme boundaries, switching in a new network may cause discontinuities to occur in the coefficients. To achieve more smooth transitional performance around these areas a set of diphone WLP synthesis networks are taught and utilized in a manner similar to the phoneme nets. Amplitude mixing (cross-fading) the outputs of both network types improves the quality of synthesis.

Table 1 shows the average absolute error of the lattice coefficients for a set of WLP diphone synthesis networks as a function of the degree of specialization. As specialization decreases the error increases. As an example of spectral error due to lattice coefficient error, Fig. 4 displays the WLP spectrum slightly past the diphthong transition [e]–[i] in the Finnish word /keinu/. The topmost curve represents the actual WLP spectrum at this point in the signal while the other curves (in order of decreasing specialization) represent the synthesized spectra using the networks listed in Table 1. The [e]–[i] specific network produces the most accurate spectral estimate (second topmost curve).

Table 1. *Lattice coefficient error vs. network specialization*

Specialization	Diphone Type	Coeff Error
specific	/e/ – /i/	5.0 %
...	front vowel – front vowel	5.3 %
...	vowel – vowel	6.1 %
general	any – any	7.5 %

4.3. Speech Database and Network Training

The speech material used for training and evaluating the networks consisted of approximately 2000 Finnish words spoken in isolation by a single male speaker. This manually segmented and phonetically transcribed material was divided into training and evaluation sets with a 2:1 ratio on a word basis. Each phone or diphone segment in either the training or evaluation set provided for 13

temporally nonlinearly spaced training elements. The number of elements in the training and evaluation sets for the most general diphone network exceeded 100,000 and 50,000, respectively. As the degree of specialization increased the size of the sets decreased.

For each degree of specialization the number of hidden nodes was systematically varied to determine the optimum network size so as to match the network to the difficulty of the mapping problem. Three hidden nodes was found to minimize the error for the most specialized network while the more general networks performed better with a substantially larger number of nodes. For example, the any-any diphone net displayed in Fig. 4 utilized 500 hidden nodes and this explains the relatively high level of spectral detail produced by this network.

5. EXCITATION CODEBOOK

The excitation codebook is an indexed table of residual signals, extracted from the speech database signal entries for the individual speaker to be modeled. In the most simple case a single excitation pattern may be used for all voiced sounds. However, a more natural voice quality is obtained if each phoneme has a different entry in the codebook, each representing a typical case of this specific phoneme. If desired, the codebook can be made even more specialized, e.g., by providing a separate entry for some critical allophones.

The entries of the excitations are concatenated during synthesis so that the desired pitch is generated according to the pitch target produced by the corresponding netset. For unvoiced sounds, white noise is used as an excitation signal.

6. PROSODY CONTROL

Prosody control is accomplished with three sets of networks for segmental durations, fundamental frequency, and gain (loudness). Their input is similar to the WLP networks' input with some difference in the phonetic information. Pitch nets are coded onto the semitone scale, loudness nets onto the phon scale, and the duration nets onto a logarithmic time scale. Again, specialization is utilized.

Our prosody control results were as follows: duration estimation was the most difficult task and specialization was needed for the error to decrease below 20%, the difference limen. A 2.2 phon error was achieved with loudness networks — one phon is generally considered just noticeable. An error of 3.5% was measured for the pitch networks: this amounts to about 0.6 semitones at 100 Hz and is well below the 1.5 to 2 semitone threshold for speech [10]. Prosody control is discussed in more detail in [11], [12], and [13].

7. SUMMARY

An experimental framework for individual sounding TTS utilizing WLP and specialized neural network sets for controlling spectral and prosodic parameters has been presented. The system described in this paper is in the development stage and so far has been trained and evaluated on isolated words. Future work includes extending the synthesizer to the sentence level as well as implementing a real-time version.

8. ACKNOWLEDGEMENT

This work has been supported by the Academy of Finland.

9. REFERENCES

[1] Karjalainen M, Laine, U. K., Toivonen R., "Aids for the Handicapped based on SYNTE 2 Speech Synthesizer". *Proc. of IEEE ICASSP-80*, Denver, 1980.

[2] Lukaszewicz K., Karjalainen M., "Microphonemic Method of Speech Synthesis," *Proc. of IEEE ICASSP-87*, Dallas, 1987.

[3] Moulines E., and Carpenter F., "Pitch-Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones," *Speech Communication*, 9(5/6):453–467, Dec 1990.

[4] Markel J. D., and Gray A. H., *Linear Prediction of Speech*, Springer Verlag, New York, 1976.

[5] Laine U. K., Karjalainen M., and Altosaar T., "Warped Linear Prediction (WLP) in Speech and Audio Processing," *Proc. IEEE ICASSP -94*, Adelaide, Australia, 1994.

[6] Karaali O. et al., "Text-to-Speech Conversion with Neural Networks: A Recurrent TDNN Approach," *Proc. Eurospeech-97*, Rhodes, 1997.

[7] Strube H. W., "Linear Prediction on a Warped Frequency Scale," *J. Acoust. Soc. Am.*, vol. 68, no. 4 (1980), pp. 1071-1076.

[8] Smith, J. O., and Abel, J. S. "The Bark Bilinear Transform," *Proc. IEEE ASSP Workshop*, Mohonk, New Paltz, 1995.

[9] Karjalainen M., Härmä A., and Laine U.K., "Realizable Warped IIR Filter and Their Properties", *Proc. IEEE ICASSP-96*, Munich, 1996.

[10] 't Hart J., Collier R., and Cohen, A., *A perceptual study of intonation*, Cambridge University Press, Cambridge, 1990.

[11] Karjalainen M., and Altosaar T., "Phoneme Duration Rules for Speech Synthesis by Neural Networks," *Proc. of Eurospeech-91*, Genoa, 1991.

[12] Vainio M., and Altosaar T., "Pitch, Loudness, and Segmental Duration Correlates: Towards a Model for the Phonetic Aspects of Finnish Prosody," *Proc. of ICSLP'96*, Philadelphia, 1996.

[13] Vainio M., and Altosaar T., "Pitch, Loudness, and Segmental Duration Correlates in Finnish Prosody," *NORDIC PROSODY Proc. of VIIth Conference*, Joensuu, Finland, 1996.