Proceedings of first International Conference on Conventional and Knowledge-based Intelligent Electronic Systems, KES '97, 21st-23rd May 1997, Adeliade, Australia, Editor: L.C. Jain

A NEURAL NETWORK BASED DYNAMIC RECONSTRUCTION FILTER FOR DIGITAL AUDIO SIGNALS[1]

Hossein L. Najafi[2]
Department of Computer Science
University of Wisconsin, River Falls
River Falls, WI 54022
e-mail hossein.najafi@uwrf.edu

D. W. Moses, C. H. Hustig, and J. Kinne
Wadia Digital Corporation
Troy Street, River Falls, WI 54022

Keywords: Reconstruction filter, brickwall filter, monotonic filter, cubic spline filter, neural networks, and backpropagation.

Abstract

The goal of any digital audio system is to sample and reconstruct an analog audio signal, without noticeable changes to the original signal. Currently, two major types of reconstruction filters, brickwall and monotonic filters, are used to smooth a sampled analog audio signal during its reconstruction. Brickwall filters work best on reconstruction of smooth signals and the monotonic filters are best for reconstruction of transient signals. Since audio is composed of mixed transient and smooth signals, both of these filters will introduce undesirable artifacts to the signal during its reconstruction. This paper presents a new neural network based dynamic reconstruction filter that can change its behavior to best match the type of signal that is being filtered

1. Introduction

The goal of any digital audio system is to sample and reconstruct an analog audio signal, without noticeable changes to the original signal. If, for example, the audio signal is sampled at a recording studio and the digital samples are stored on a CD, then the CD player must retrieve the digital samples and reconstruct the wave form of the audio signal as closely as possible to the wave form of the original analog signal.

In general, the architecture shown in Figure 1 is used to digitally process or store/retrieve an analog signal. Theoretically, any analog signal can be sampled and reconstructed, provided that the sampling rate is at least twice the bandwidth of the analog signal (Nyquist theorem). However, in practice, this is not feasible and the sampling rate is governed by the property of the medium that stores and/or transmits the signal. A sampling frequency lower than the Nyquist frequency will cause aliasing. Aliasing is caused by high frequency components to be erroneously represented as low frequencies in the digital signal. In order to prevent aliasing, at the input to the analog-to-digital (A/D) converter, a band-limiting filter is used to eliminate audio frequencies above one-half the Nyquist frequency. A similar over-sampling low-pass filter is used to reconstruct and smooth the

[1] Major support for this work was provided by Wadia Digital Inc. U.S. patent #5,532,950, July 2, 1996
[2] Responsible for correspondance.

digital signal before digital-to-analog (D/A) conversion.

Currently, two major types of reconstruction filters, brickwall and monotonic filters, are used to smooth a sampled analog audio signal during its reconstruction. The brickwall filter has a relatively flat passband (has small amplitude ripples in the passband) , and a steep transition band (see Figure 2a). Although, the brickwall filter has a good image rejection property (see Figure 4), due to its steep transition band, when it is driven by a transient signal it generates undesirable overshoot, ripple and ringing which is known as Gibbs phenomenon [Antoniou 79] (see Figure 3). In addition, its passband ripple, even though it is relatively small, causes an echo in the impulse response which is one of the contributors to Digital Time Displacement Error [Papoulis 62]. The monotonic filter (see Figure 2b), on the other hand, is characterized by its wide transition band (i.e., usually a drop off of 6 dB at half the sampling rate) and its smooth ripple-free passband [Papoulis 62]. Its smooth ripple-free passband eliminates the echo and its wide transition band provides for a better impulse response making it a better filter for reconstructing transient signals (see Figure 3). However, the monotonic filter has an image-energy problem (Figure 4).

An ideal reconstruction filter would have flat response over the bandwidth of interest, high attenuation above that frequency, and no ringing when presented with a transient signal such as a step or an impulse [Baker 95]. Although such a filter is not theoretically possible, here we present a new neural network based approach that would make such a filter design a practical possibility. A multilayer feed-forward neural network that uses the backpropagation algorithm [Werbos 74; Rumelhart 86; Vogl 88; Jacobs 88; Tollenaere 90; Rigler 90; Nguyen 90; Hagan 94] is trained to generate a fuzzy membership value that would indicate to what degree transient behavior is present in the sampled signal that is presented at its input. The output of the trained network is then used to dynamically change the parameters of the reconstruction filter to best match the property of the signal that is under reconstruction.

2. Neural Network Training

The basic idea behind the method proposed here is to find a fuzzy membership function [Zadeh 65; Mitra 92] for the fuzzy set transient. This fuzzy set represents the set of all signals that show transient behavior to some degree and the membership function defines the degree of the membership. An example of such a membership function f(X) is given in Figure 5. Here X represents the signal vector, and f(X) represent the degree of transient behavior in the signal. The membership function f(X) should be one when the signal is a pure transient signal (such as an impulse) and zero when it is a pure non-transient signal (such as a sine).

A three layer feed-forward network was trained to learn this membership function. The network consisted of sixteen input units, five hidden units, and a single output unit. The network was trained using the backpropagation algorithm. The size of the network was chosen using standard pruning methods. The training set consisted of sets of pure transient and non-transient signals such as impulses, ramps, square waves, sweeping frequencies, single tone frequencies, white noise, etc. For each type of the signals, the frequency, phase, amplitude and rate of swept was varied and for each combination of these parameters a 32-sample vector was generated and preprocessed in the following manner:

- Fast Fourier Transform (FFT) was used to generate a 32-sample Fourier spectrum of the 32-sample signal vector.
- The lower 16-sample of the resulting signal was extracted (i.e., the lower FFT image).
- The resulting signal was normalize by the largest sample value.

The target value of each transient signals was set to one and that of each non-transient signals was set to zero. In addition to these pure transient/non-transient signals, complex signals were formed and added to the training set. The complex signals were created by adding and/or multiplying the pure signals with each other. The target value of the complex signals were set to one if there was any transient behavior in the signals. Using this training set, the network was trained to recognize transient signals form non-transient signals.

3. Architecture of the Dynamic Filter

Two different versions of the dynamic filter were created in the following manner. In the first version, in addition to the trained network, a brickwall filter and a monotonic cubic spline filter were used. The output of the neural network was rate limited to smooth out rapid transient to non-transient transitions. The rate limited network's output was then used in the following manner to combine the output of the two filters:

$$f_d = \alpha f_m + (1-\alpha) f_b$$

Here, f_d is the output of the dynamic filter, f_m is the output of the monotonic filter, f_b is the output of the brickwall filter, and α is the rate limited output of the neural network. When the signal is pure transient, the network's output will be one, and the output of the dynamic filter would be the same as that of the monotonic filter. On the other hand, if the network's output is zero, the signal is classified as non-transient and the output of the dynamic filter would be the same as the brickwall filter's output. For any other values of α the networks output would be a linear combination of the output of the two filters. Smoother input signals (i.e., less transient signals) would generate network output values closer to one forcing the output of the dynamic filter to be closer to that of the monotonic filter. On the other hand, for more transient signals, the newtork's output will be closer to zero and the dynamic filter's output would be closer to that of the brickwall filter. The overall architecture of this version of the system is given in Figure 6.

The second version of the dynamic filter is more efficient in terms of computational cost. In this version, only a single reconstruction filter is used. However, the coefficients of the dynamic filter are dynamically changed according to the following formula:

$$\sigma_d = \alpha\sigma_m + (1-\alpha)\sigma_b$$

Here, σ_d is the coefficient vector of the dynamic filter, σ_m is the coefficient vector of the monotonic filter, σ_b is the coefficient vector of the brickwall filter, and α is the rate limited output of the neural network. This filter is more efficient, since the signal passes through a single filter that dynamically changes its behavior. The overall architecture of this version of the system is given in Figure 7.

4. Simulation Results

The performance of the system was measured in two steps. First we tested the generalization capability of the neural network (i.e., how well it performs on signals it has not seen before) using a test set. The test set was composed of the same type of signals that were in the training set, but with different set of parameters. For example, sine waves in the test set had different amplitude, frequency and phase. For each signal in the test set, the output generated by the network was compared with the actual target value of the signal and if the network's output was within 10% range of the expected target value, the signal was considered to be correctly classified. For example, since the target value of a pure transient signal should be one, if the network output for an impulse was greater than or equal to 0.9, then the signal was considered to be correctly classified.

The overall classification accuracy (i.e., the percentage of the signals that were correctly classified) for the pure signals was 96.6% and that of the complex signals was 81.3%.

After testing the generalization capability of the network, a software simulation of the dynamic filter was developed using Matlab's Simulink software. Figure 8 shows the impulse response of the three filters the response of the filters to a 20 KHz sine wave. Note that the dynamic filter outperforms both the brickwall and the monotonic filter on either type of signals. For transient signals (e.g., the impulse and the step), the dynamic filter generates no ringing and for non-transient smooth signals it has good image rejection.

5. Conclusions and future work

A new dynamic reconstruction filter that can change its behavior based on the type of signal that is being filtered was presented. A feed-forward backpropagation network was used for signal classification. Simulation results show that due to its dynamic behavior, this filter outperforms the two major types of reconstruction filters (i.e., brickwall and monotonic) in the market.

Presently the filter is being implemented in hardware. Preliminary data indicates good system performance.

The same idea can also be applied to Digital Image processing for reduction of image artifacts. This concept is currently under investigation.

Acknowledgments

We would like to thank Wadia Digital Inc. for providing support for this project. Also, we would like to thank Jason Goray, Jason Triplat, and Keith Jaco for their help in writing the simulation programs for this project.

References

[Antoniou 79] Antoniou A., *Digital Filters : Analysis and Design*, McGraw-Hill, New York, 1979.

[Papoulis 62] Papoulis A., *The Fourier Integral and its Applications*, McGraw-Hill, New York, 1962.

[Baker 95] Baker D., Fibush D. and Penny B., *Digital to Analog Conversion-Data and Filter Requirements*, pp. 120-124, SMPTE Journal, March 1995.

[Lippman 87] Lippman R.P., *An Introduction to Computing with Neural Nets*, IEEE ASSP Magazine, pp. 4-22, 1987.

[Werbos 74] Werbos P.J., *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, Ph.D. Thesis. Harward University, Cambridge, MA, 1974.

[Rumelhart 86] Rumelhart D.E., Hinton G.E. and Williams R.J., *Learning Internal Representation by Error Propagation, Parallel Distributed Processing*, Chapter 8, MIT Press, 1986.

[Vogl 88] Vogl T.P., Mangis J.K., Zigler A.K., Zink W.T. and Alkon, D.L., *Accelerating the convergence of the backpropagation method*, Biological Cybernetics, Vol. 59, P.P. 256-264, Sept. 1988.

[Jacobs 88] Jacobs R.A., *Increased rates of convergence through learning rate adaptation*, Neural Networks, Vol. 1, No. 4, pp. 295-308, 1988.

[Tollenaere 90] Tollenaere T., *SuperSBA: Fast adaptive backpropagation with good scaling properties*, Neural Networks, Vol. 3, No. 5, pp. 561-573, 1990.

[Rigler 90] Rigler A.K., Irvine J.M. and Vogl T.P., *Rescaling of variables in backpropagation learning*, Neural Networks, Vol. 3, No. 5, pp. 561-573, 1990.

[Nguyen 90] Nguyen D. and Widrow B., *Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights*, Proceedings of the IJCNN, Vol. 3, pp. 21-26, July 1990.

[Hagan 94] Hagan M.T. and Menhaj M., *Training feedforward networks with Marquardt algorithm*, IEEE transactions on Neural Networks, Vol. 5, No. 6, 1994.

[Hopfield 82] Hopfield, J.J., *Neural Networks and Physical Systems with Emergent Collective Computational Abilities*, Proceedings of the National Academy of Sciences, USA, National Academy of Sciences, Washington, D.C., Vol. 79, pp. 2554-2558, 1982.

[Kohonen 82] Kohonen T., *Self-Organized Formation of Topologically Correct Feature Maps*, Biological Cybernetics 43, Spring-Verlag, 59-69, 1982.

[Zadeh 65] Zadeh L.A., Info. Control 8, 338-352, 1965.

[Mitra 92] Mitra s. and Pal S.K., *Multilayer perceptrons, fuzzy sets and classification*, IEEE Transactions on Neural Networks, Vol. 3, No 5, 672-683, 1992.

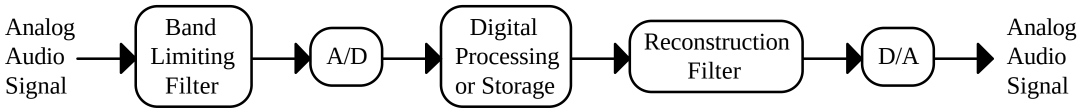
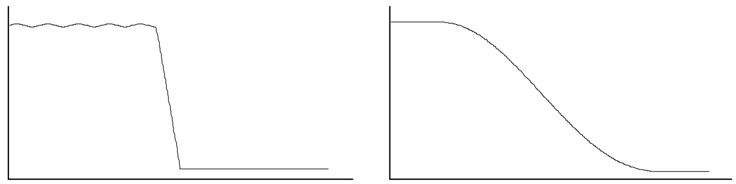


Figure 1. Architecture used for digital processing and storage of analog signals



a) Brickwall filter

b) Monotonic filter

Figure 2. Frequency response of the two major types of reconstruction filters

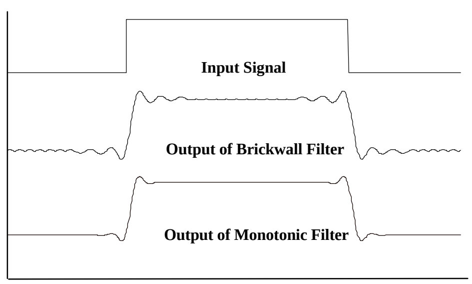


Figure 3. Ringing in a brickwall filter

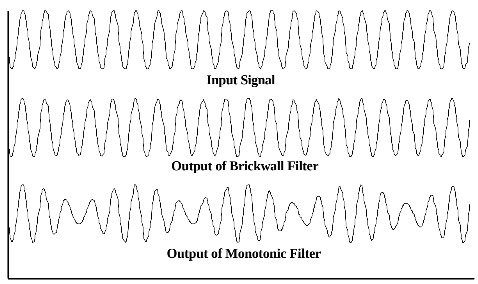


Figure 4. Image-energy problem of a monotonic filter

Figure 5. Example of a fuzzy membership function for the fuzzy set transient

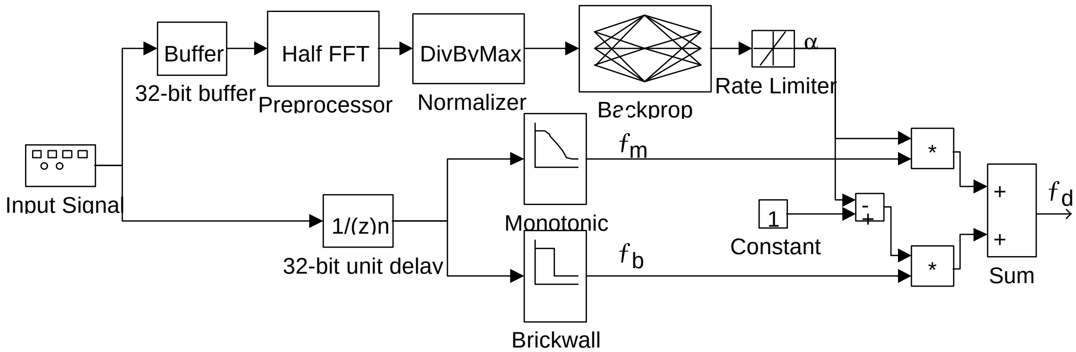


Figure 6. Overall architecture of the first version of the dynamic filter

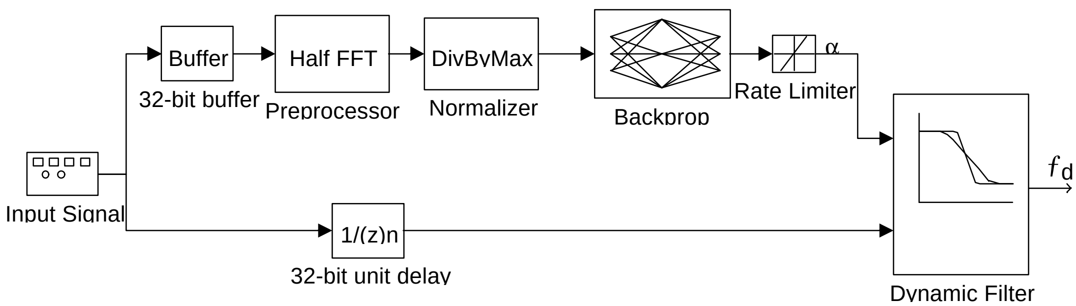


Figure 7. Overall architecture of the second version of the dynamic filter

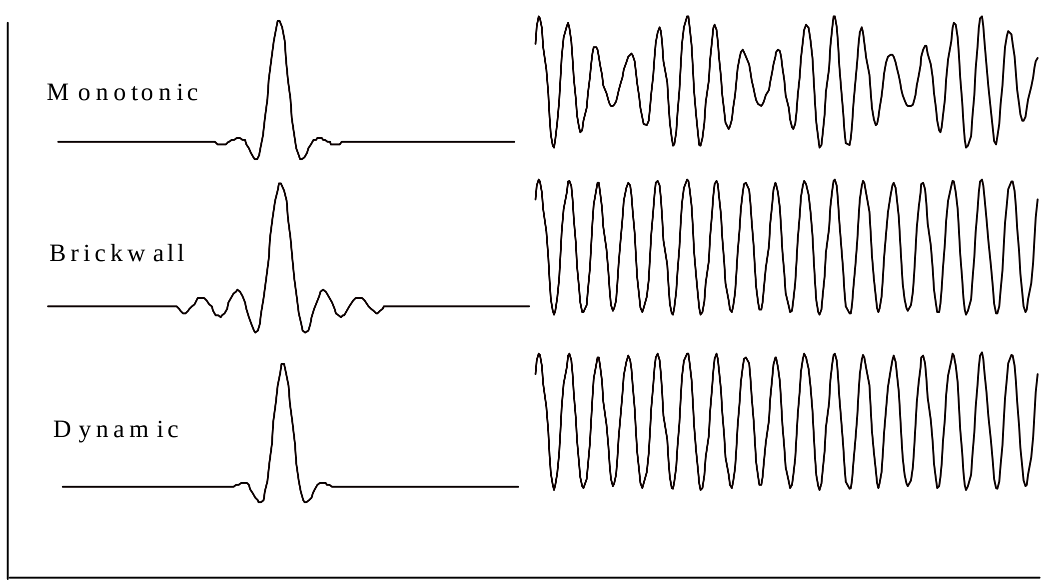


Figure 8. Performance of the Dynamic filter in comparison to that of the Brickwall and Monotonic Filters.