ESTIMATION OF SINUSOIDS IN AUDIO SIGNALS USING AN ANALYSIS-BY-SYNTHESIS NEURAL NETWORK

Guillermo García

Creative Advanced Technology Center, 1500 Green Hills Rd., Scotts Valley, CA 95067 / guille@atc.creative.com
CCRMA, 626 Lomita Dr., Stanford University, CA 94305-8180 / guille@ccrma.stanford.edu

ABSTRACT

In this paper we present a new method for estimating the frequency, amplitude and phase of sinusoidal components in audio signals. An analysis-by-synthesis system of neural networks is used to extract the sinusoidal parameters from the signal spectrum at each window position of the Short-Term Fourier Transform. The system attempts to find the set of sinusoids that best fits the spectral representation in a least-squares sense. Overcoming a significant limitation of the traditional approach in the art, preliminary detection and interpolation of spectral peaks is not necessary and the method works even when spectral peaks are not well resolved in frequency. This allows for shorter analysis windows and therefore better time resolution of the estimated sinusoidal parameters. Results have also shown robust performance in presence of high levels of additive noise, with signal-to-noise ratios as low as 0 dB.

1. INTRODUCTION

The representation of audio signals in terms of time-varying sinusoidal components is well known in the fields of computer music ([2][6][7]), speech ([1]), and more recently audio coding ([3][4]). The sinusoidal model represents a signal x(n) with sampling period T as a finite sum of sinusoids with time-varying frequencies f_p and amplitudes a_p:

$$x(n) = \sum_{p=1}^{P} a_p(n)\cos(\varphi_p(n)) \qquad (1)$$

where the phase φ_p is updated as

$$\varphi_p(n+1) = \varphi_p(n) + 2\pi f_p(n)T \quad and \quad \varphi_p(0) = \phi_p \qquad (2)$$

Typically, the model parameters are estimated from the Short-Time Fourier Transform (STFT) representation of the signal being modeled, by means of automatic analysis techniques. The traditional method in the art ([1]) starts by performing an exhaustive search for "spectral peaks", i.e. main lobes of the analysis window, in the Discrete Fourier Transform (DFT) magnitude spectrum. The short-term frequency and amplitude of the sinusoids are then estimated by polynomial interpolation between the DFT points of each spectral peak. A second step consists of matching the peaks between successive STFT frames to determine the time-varying frequency and amplitude trajectories. Several tracking algorithms have been developed for this purpose ([1][2][5][6]).

The traditional method works well for low bandwidth speech and relatively stationary audio signals. However, the assumption that real audio signals can be decomposed in terms of sinusoids which are stationary enough, and sufficiently resolved in frequency, to be seen as discrete peaks in the spectrum is not valid for many types of audio signals. Polyphonic music or monophonic sounds with fast pitch variations are often poorly modeled since the peak detection algorithm fails to find unresolved peaks. Furthermore, the interpolation on detected peaks yields inaccurate results due to the temporal smoothing introduced by the analysis window, the influence of neighbor peaks, and the peak shape distortion introduced by additive noise.

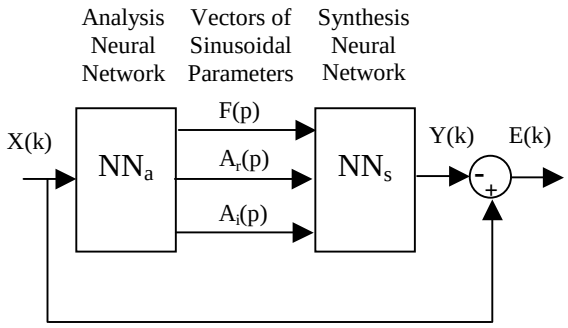

Fig. 1*: Analysis-by-Synthesis Neural Network System.*

In this paper we propose a method for estimating the sinusoidal parameters from the DFT spectrum without any peak detection/interpolation, based on an analysis-by-synthesis system of neural networks. The method allows for shorter analysis windows and thus less parameter smoothing, takes into account the effect of overlapping peaks, and is robust in presence of additive noise.

2. METHOD DESCRIPTION

Our method is based on the neural network analysis-by-synthesis system depicted in Figure 1. For a given input DFT X(k), the system attempts to find an output DFT Y(k) that best fits X(k) in a least-squares sense, i.e. that minimizes the energy of the DFT estimation error E(k).

The synthesis stage comprises a neural network (NN_s) that maps a set of P sinusoids onto the DFT of their sum Y(k). The sinusoidal parameters are coded using three vectors: frequencies F(p), real amplitudes $A_r(p)$ and imaginary amplitudes $A_i(p)$ (i.e. amplitude and phase are coded in cartesian coordinates). The synthesis neural network NN_s is not adaptive but is designed by hand to perform the synthesis mapping.

The analysis stage comprises a typical adaptive, single-layer or two-layer neural network (NN_a) with sigmoid nonlinearities, fully or partially connected ([8][9]). It maps an input DFT X(k) onto the sinusoidal parameters F(p), $A_r(p)$ and $A_i(p)$.

For each STFT spectrum X(k), the neural network system is trained with the back-propagation algorithm ([8]) so as to perform an identity map between input and output.

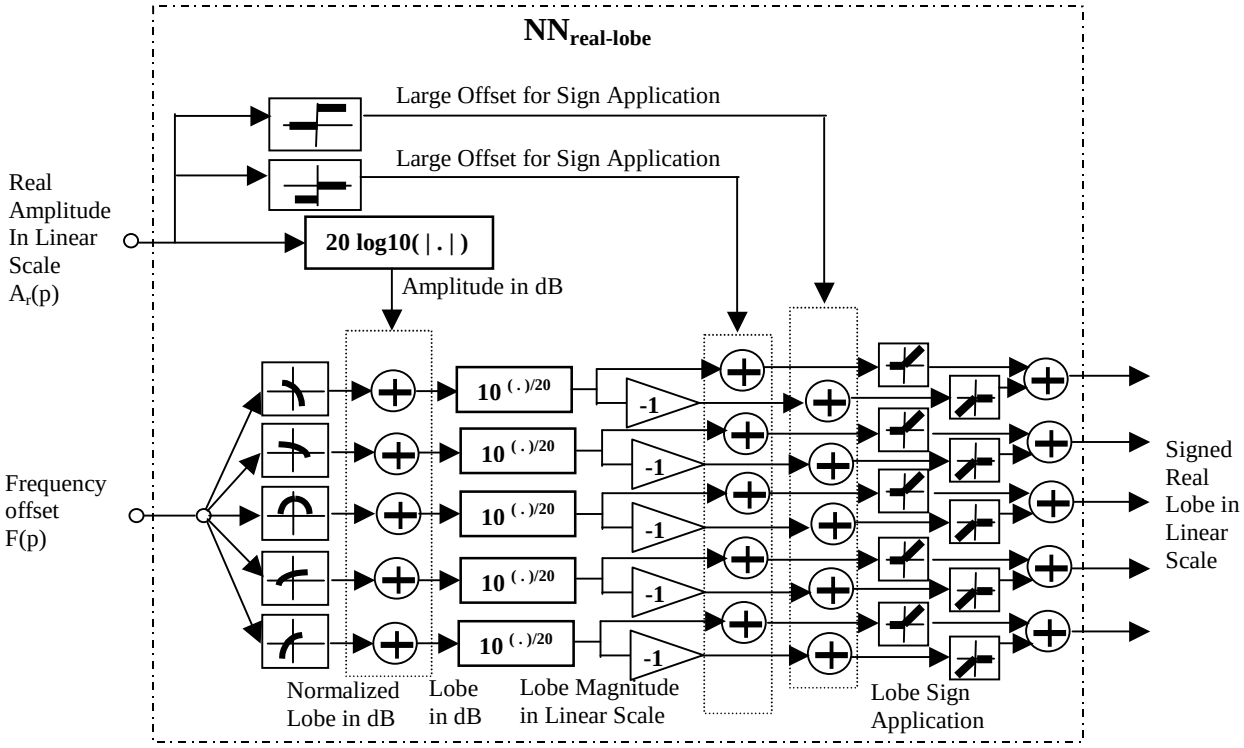

Fig. 2: *Neural network designed by hand to synthesize the central five points of the analysis window DFT (time origin of the DFT is at the center of the analysis window).*

Several adaptation iterations are performed while holding the input spectrum X(k) until the energy of the estimation error E(k) is below a specified threshold, or for a specified maximum number of iterations. Since the synthesis network NN_s is designed by hand to perform the synthesis mapping, only the weights of the analysis network NN_a are adapted during training. The synthesis stage does need to be implemented in the form of a neural network in order for the output error E(k) to be back-propagated through it during training of NN_a.

After convergence, the output of the analysis network NN_a corresponds to a set of sinusoidal parameters F(p), $A_r(p)$ and $A_i(p)$ that best model the input spectrum X(k) in a least-squares sense.

2.1 Design of the Synthesis Neural Network

The building block of the synthesis network NN_s is a "lobe synthesizer", a sub-network that maps a real-valued amplitude $A_r(p)$ or $A_i(p)$ and frequency F(p) of a single sinusoid onto a specified number of main-lobe points of the analysis window DFT. Amplitude is expressed in linear scale and frequency is expressed as an offset with respect to a reference DFT point associated with sinusoid p. Thus, F(p) = 0 means that the sinusoid frequency falls exactly on the reference DFT point. DFTs with time-origin at the center of the analysis window are used so that lobe phase is constant, thus allowing for real and imaginary parts of the lobes to be synthesized separately. The lobe synthesizer is depicted in figure 2.

The nonlinearities of the first layer correspond to the shape of the window main lobe in dB. The first layer produces a normalized lobe in dB scale centered on F(p). The second layer de-normalizes the lobe by adding the input amplitude in dB, and transforms the lobe to linear scale. The rest of the graph takes care of applying the correct sign (+ or -) to the lobe: it generates both positive and negative lobe versions and then uses a large offset, which is function of the amplitude sign, to gate the incorrect signed version and let pass the correct one.

The synthesis neural network NN_s uses an array of such lobe synthesizer units to generate one lobe for each sinusoid at its input. The lobe-synthesizer outputs are overlap-added onto the real and imaginary DFT buffers $Y_r(k)$ and $Y_i(k)$ respectively, as shown in figure 3 (for $Y_r(k)$ only).

Each of the P sinusoids at the input of the synthesis network is associated with a reference DFT point. The designer chooses the number of sinusoids P, the association between input sinusoids and reference DFT points, and the maximum excursion of the frequency offset.

The complete synthesis network NN_s uses the previously described unit to synthesize the real and imaginary parts of the estimated DFT Y(k), as shown in Figure 4.

2.2 Estimation Error Weighting

The estimation error E(k) must be expressed in signed linear scale in order for phase information to be tracked. However, we wish the error on the spectral magnitude, expressed in Decibels, to be uniform over the frequency axis.

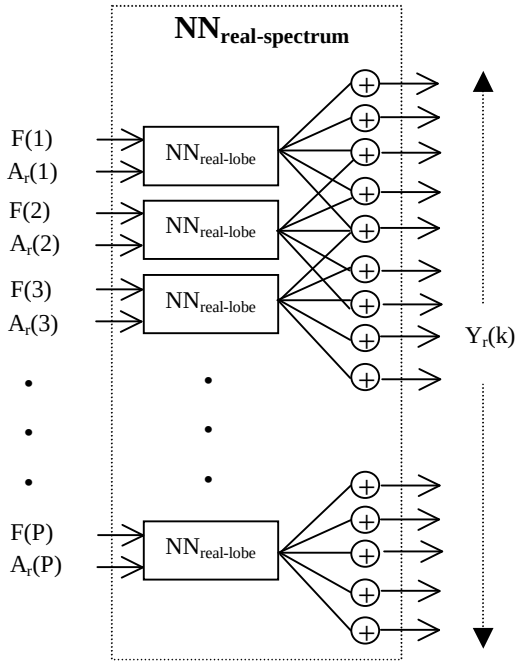

Fig. 3: *Synthesis of the real spectrum $Y_r(k)$ by overlap-add of lobe-synthesizer outputs.*

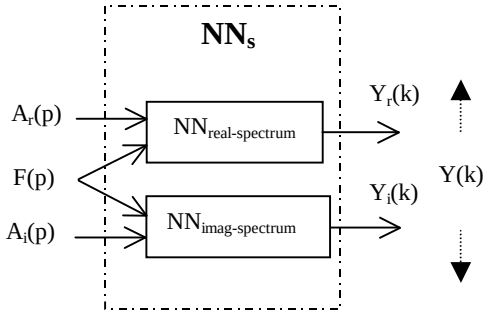

Fig. 4: *Synthesis of the estimated complex spectrum Y(k).*

To achieve this, we use an error weighting function W(k) defined as:

$$W(k) = \left(20\log_{10}\left|X(k)\right| - 20\log_{10}\left|Y(k)\right|\right)^{\alpha} \qquad (3)$$

with α=4 typically. The error used to train the network is then modified as $E'(k) = \left[C.W(k).E(k)\right]$, where C is a rescaling factor such that the error energy remains unchanged, i.e.:

$$C = \sqrt{\sum_{k=0}^{N/2}\left|E(k)\right|^2 \Big/ \sum_{k=0}^{N/2}\left|W(k)E(k)\right|^2} \qquad (4)$$

This rescaling is necessary to avoid divergence. Perceptual error weighting could also be performed by using psychoacoustics criteria, in order to obtain better estimation at perceptually relevant regions of the spectrum.

3. EXPERIMENTS AND RESULTS

In the figures below, the input spectrum X(k) is represented in solid line, and the dotted line corresponds to the estimated spectrum Y(k) at the output of the synthesis network. The 'x' marks correspond to the frequency and magnitude (phase is not shown) of simulated sinusoids, and the 'o' marks indicate the frequencies and amplitudes estimated by the analysis network after convergence. The horizontal axis represents the normalized frequency (f.T.N) where T is the sampling period and N is the DFT length. Amplitude in dB is represented on the vertical axis. We used T=1/16000 sec and N=64. A two-layer analysis network was initialized with random small weights in the range [–0.1; 0.1] and trained using back-propagation during 300 iterations with an adaptation constant μ=0.01.

Figure 5 shows the results on a signal with five sinusoids; with three peaks poorly resolved in frequency (less than 3 DFT bins apart). The mean absolute estimation error on the normalized frequency was 0.06, i.e. 6% of the DFT frequency resolution. The mean amplitude estimation error was 0.2dB, and mean phase error was 0.008 radians. The estimation errors on the unresolved peaks were of the same order of magnitude as on the resolved peaks. Figure 6 shows the results on a sum of eight sinusoids extremely close in frequency (about 2 DFT bins, equivalent to an analysis window size of two periods for a harmonic signal, whereas the minimum required by the peak detection approach is 4 periods). The traditional peak detection approach would have clearly failed in this case. Six of the sinusoids were estimated as single sinusoids, whereas the other two were modeled as sinusoid pairs. Over the six correctly estimated sinusoids, the mean normalized-frequency error was 0.08 DFT bins, mean amplitude error was 0.6dB and mean phase error was 0.01 radians.

Figures 7 and 8 show the results on a signal consisting of five sinusoids plus additive white Gaussian noise, with signal-to-noise ratios (SNR) of 3dB and 0dB respectively. Mean normalized frequency, amplitude and phase estimation errors were of 0.15 and 0.28 DFT bins, 1.05 and 1.19 dB and 0.06 and 0.09 respectively.

Finally, figure 9 shows a test on a real speech signal with an analysis window of two and a half periods and N = 128.

4. CONCLUSIONS

We have presented a new spectral analysis-by-synthesis method based on a neural network system, which improves the extraction of sinusoidal parameters from an audio signal. The method eliminates the need for spectral peak detection and interpolation, a main weak point of the traditional approach in the art. Instead, the sinusoidal parameters are found by global optimization according to a least-squares criterion, thus taking into account the effects of overlapping spectral peaks. Results have shown that our method performs well even when peaks are not resolved, allowing for shorter analysis windows (2 instead of 4 periods for harmonic signals) and thus improving the time resolution of the sinusoidal parameters. Additionally, performance in presence of high levels of additive white noise was robust.

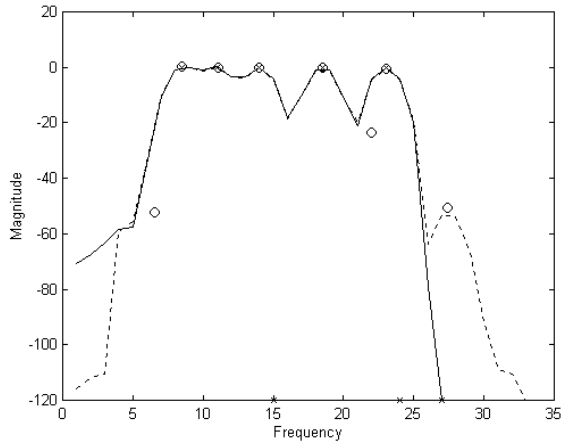

Fig. 5: *Five-sinusoid signal with three unresolved peaks.*

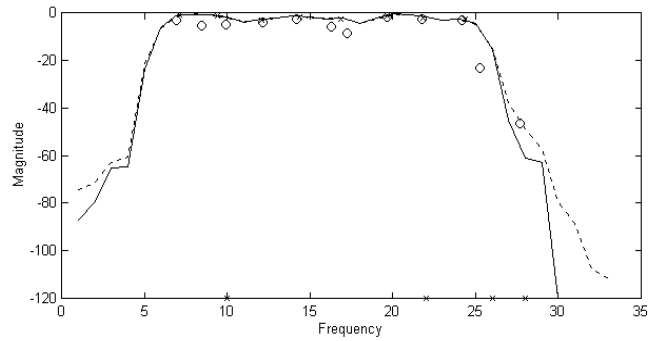

Fig. 6: *Eight-sinusoid signal with poor frequency resolution.*

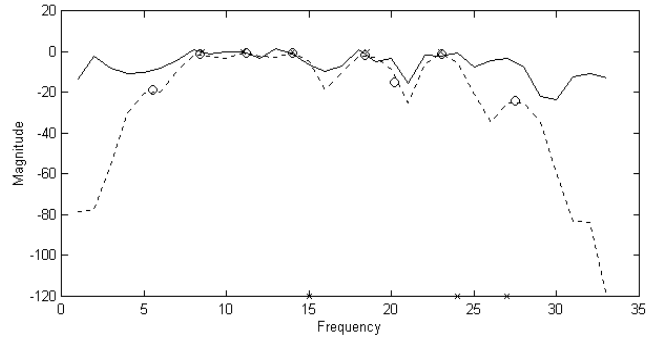

Fig. 7: *Five sinusoids embedded in white Gaussian noise with SNR =3 dB.*

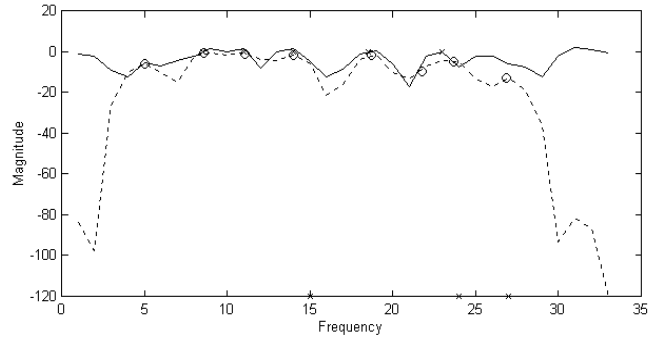

Fig. 8: *Five sinusoids embedded in white Gaussian noise with SNR = 0 dB.*

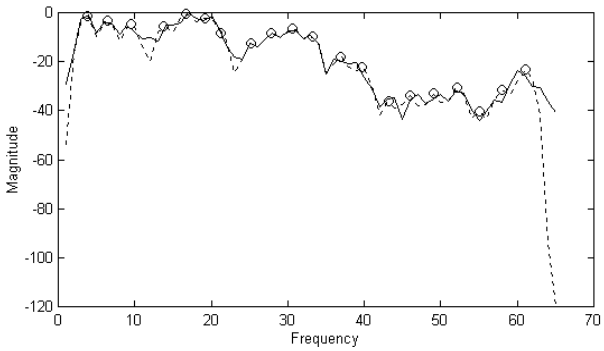

Fig. 9: *Real speech signal, with an analysis window 2.5 fundamental periods long.*

5. REFERENCES

[1] R. Mc Aulay and Th. Quatieri, “Speech analysis/synthesis based on a sinusoidal representation”, IEEE Trans. On Acoust., Speech, and Signal Proc., vol. ASSP-34, August 1986.

[2] X. Serra and J. O. Smith. 1990, ``Spectral Modeling Synthesis: A Sound Analysis/Synthesis System Based on a Deterministic plus Stochastic Decomposition.", Computer Music Journal, 14(4):12-24

[3] S. Levine, J.O. Smith III, “A Sines+Transients+Noise Audio Representation for Data Compression and Time/Pitch-Scale Modifications”, 105th Audio Engineering Society Convention, San Francisco 1998.

[4] S. Levine, T. Verma, J.O. Smith III, “Multiresolution Sinusoidal Modeling for Wideband Audio With Modifications”, ICASSP 1998, Seattle.

[5] R. L. Streit and R. Barrett, ``Frequency line tracking using hidden Markov models,", IEEE Transactions ASSP, vol. 38, no. 4, pp. 586-589, 1990.

[6] Ph. Depalle, G. Garcia and X. Rodet, “Tracking of partials for additive sound synthesis using Hidden Markov Models”, Proc. of IEEE-ICASSP 1993, Minneapolis.

[7] Ph. Depalle and L. Tromp, “An Improved Additive Analysis Method Using Parametric Modelling of the Short-Time Fourier Transform”, ICMC 1996, Hong-Kong, August 1996.

[8] B. Widrow and M. Lehr, “30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation”, Proc. of the IEEE, Vol 78, No. 9, September 1990.

[9] S. Haykin, “Neural Networks, a Comprehensive Foundation”, Prentice Hall 1999.