An Analysis/Synthesis System of Audio Signal with Utilization of an SN Model

Martin TURI NAGY, Gregor ROZINAJ

Dept. of Telecommunications, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava, Slovak Republic

turi@ktl.elf.stuba.sk, gregor@ktl.elf.stuba.sk

Abstract. *An SN (sinusoids plus noise) model is a spectral model, in which the periodic components of the sound are represented by sinusoids with time-varying frequencies, amplitudes and phases. The remaining non-periodic components are represented by a filtered noise. The sinusoidal model utilizes physical properties of musical instruments and the noise model utilizes the human inability to perceive the exact spectral shape or the phase of stochastic signals.*

SN modeling can be applied in a compression, transformation, separation of sounds, etc. The designed system is based on methods used in the SN modeling. We have proposed a model that achieves good results in audio perception.

Although many systems do not save phases of the sinusoids, they are important for better modelling of transients, for the computation of residual and last but not least for stereo signals, too. One of the fundamental properties of the proposed system is the ability of the signal reconstruction not only from the amplitude but from the phase point of view, as well.

Keywords

Sinusoidal, analysis, synthesis, peak detection, peak interpolation, peak continuation, parameter estimation, residual modeling.

1. Introduction

In most cases, the PCM representation of the signal is not suitable for sound analysis. The PCM represents levels of changing of an acoustic pressure, that impacts to a human ear. In general, we can use spectral modelling, or another representation to transform the audio signal to a more applicable form. One of these representations is the SN model. The sinusoidal part represents the periodic components of the audio signal while the noise part represents the stochastic components of the audio signal. SN modelling is also widely used in a speech coding.

There are other models that are similar to the SN model. One of them is vocoding [1]. Vocoders (voice coders) were used for the first time in speech coding. They utilize the STFT (short-time Fourier transform). The input signal is represented as multiple parallel channels. Each of them describes the audio signal as an individual frequency band. Vocoders can be used to pitch-shift or time-scale the audio signal (with SN models this can be done too). The most important disadvantage of vocoders is the insufficient frequency resolution of the filter bank.

Other similar systems to the SN model used in speech processing are based on HNM (harmonic plus noise) model [16]. HNM assumes the speech signal is composed of a harmonic and a stochastic part. The harmonic part describes the quasiperiodic components of the speech signal while the noise part describes the nonperiodic components. The HNM model allows prosodic modifications (which include both time and pitch change), too.

Another common model for speech coding is MBE (multi-band excitation) [1]. The speech is considered as a product of an excitation spectrum and a spectral shape of the voice tract. The excitation spectrum is modelled as the combination of a harmonic and a random part. The spectrum is divided into subbands and in each subband the voiced/unvoiced decision is made. Then the subband is modelled like if it contained only harmonic or only stochastic components.

This paper is divided into following sections. First, the SN model is described. Then, our proposed system is depicted. After that, the system performance and examples are shown.

2. SN Model

In the past, sinusoidal modelling was used in the speech compression and in the audio analysis/transformation/synthesis. In the computer processing of the audio signals, the sinusoids alone were not considered as a sufficient model for the modelling of a wideband audio. Serra [2] was the first who came with an improvement – the residual noise model that models the non-sinusoidal part of the signal as a time-varying noise source. These systems are called sinusoids plus noise systems.

Sounds that are produced by musical instruments or other systems can be modelled as a sum of the deterministic and the stochastic part, or in other words, as a set of sinusoids plus the noise residual [3]. Sinusoidal components

are produced by a vibrating system and they are usually harmonic. The residual contains the energy produced by an excitation mechanism and by other components that are not results of periodic vibration.

In the standard SN model the deterministic part is represented as a sum of sinusoidal trajectories with time-varying parameters. The trajectory is a sinusoidal component with time-varying frequencies, amplitudes and phases. It appears in a time-frequency spectrogram as a trajectory. The stochastic part is represented by the residual. The whole signal can be written as

$$x(t) = \sum_{i=1}^{N} a_i(t)\cos(\theta_i(t)) + r(t), \tag{1}$$

where α_i and θ_i are amplitude and phase of the sinusoid and $r(t)$ is the noise residual, which is represented by the stochastic model. We assume that sinusoids are locally stable. This means, that their amplitudes are not changing too fast and that phases are locally linear. The whole signal is modelled either with a utilization of the sinusoidal or the stochastic model. The residual $r(t)$ contains all the components that are not represented by the sinusoidal model (undetected sinusoids, too).

The human perception is sensitive neither to details of the sound spectral shape, nor to the phase of non-periodic signals. In assumption that the residual contains only stochastic components, it can be represented by a filtered white noise. The instant amplitude and phase are not saved, but they are modelled by the time-varying filter of the spectral shape, or by short-time energies of fixed frequency bands, i.e. Bark bands, as used in our system.

In general, SN systems can be used for the sound synthesis, content-based audio processing, sound analysis processing and for audio coding. After the original sound is parametrized by the SN model, pitch-shifting and time-scaling can be applied to the sound. The synthesized signal can be analyzed further, or the analysis can be performed directly to the parameters. Moreover, the original signal can be compressed through a utilization of the parameters, or in speech signals, the prosody can be modified. Consequently, SN modelling can be used for an automatic transcription or a separation of sound sources. There are also other specific applications, for example sound databases, sound editing, etc.

The design of SN model is shown in Fig. 1. First, the input signal is analyzed, to take time-varying frequencies, amplitudes and phases. Then, the sinusoids are synthesized and subtracted from the original signal to take the residual. Stochastic analysis is applied to the residual. Afterward short-time energies of Bark bands are computed. The stochastic signal has to be synthesized and added to the synthesized sinusoids to obtain the whole signal.

The most complicated part of the system is the sinusoidal analysis. The input signal has to be divided into overlapped and windowed frames. Then the short-time spectrum of the frame is obtained, with an utilization of STFT. The spectrum is then analyzed and peaks are detected. After that, the parameters of the peaks are estimated (frequency, amplitude and phase).

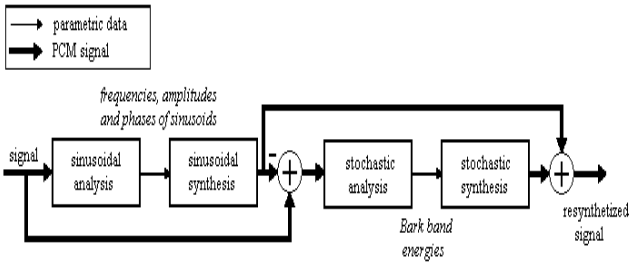

Fig. 1. Design of SN model.

In the sinusoidal synthesis, the parameters are connected into interframe trajectories. The peak continuation algorithm tries to find reasonable peak from the next frame for each trajectory. The obtained trajectory contains all the information needed for the synthesis. The trajectory parameters are interpolated to avoid discontinuities in the phase and frequency.

We can obtain the residual in the time domain by subtracting the synthetized sinusoids from the original signal. This residual can be represented by the filtered white noise. Because the human ear is not sensitive to variations of energy inside the Bark bands for quasi-stationary signals [15], the exact spectral shape is not needed. The only information needed are the short-time energies inside the Bark bands. In psychoacoustic experiments, it has been shown that the sound is analyzed in the hearing system by a bank of filters. Bandwidth of these filters is called critical bandwidth. There are 25 critical bands between 0 and 20 kHz, which are not linearly spaced, called also Bark bands. The Fig. 2 enables a comparison of the Bark-scale with the frequency scale.

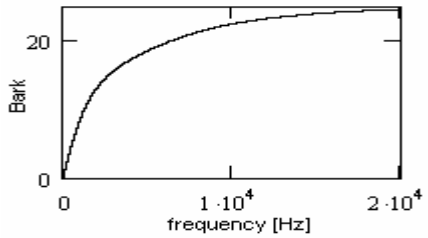

Fig. 2. Relations between Bark-scale and frequency scale.

In the noise synthesis, the complex spectrum is created and random phases for amplitudes (obtained from energies) are generated. Adjacent frames are combined with an utilization of the overlap-add synthesis.

3. Existing Systems

The first sinusoidal model was proposed by McAulay and Quatieri [10] for speech coding and by Smith and Serra for the representation of musical signals [7]. There are many improved implementations based on these systems, for example LEMUR [12], SNDAN [13], etc.

We have focused on two systems, implemented by Virtanen [6] and by Levine [8]. The systems are well-documented and they are similar to our proposed system.

Virtanen proposed the modular system that can use more algorithms for the peak detection, peak interpolation, parameter estimation and peak continuation. Virtanen's system uses a fixed peak detection algorithm (local maxima are taken above fixed threshold of the amplitude spectrum), quadratic interpolation [7], parameters are estimated with the utilization of STFT and the peak continuation uses a parameter derivatives algorithm [6]. The system can use a multiresolution modelling. The bandwidth is divided into three parts. In the first part frequencies between 20 Hz and 200 Hz are detected using an 86 ms long window (3792 samples), in the second part frequencies between 200 Hz and 5 kHz are detected using a 46 ms long window (2028 samples), and in the third part frequencies between 5 kHz and 10 kHz are detected using the same length of the window as in the second part, but with slightly different parameters. All the windows are half-overlapped. In his design, Virtanen does not use the method of an interpolative analysis in the system, because of computational demands. In the synthesis, amplitudes are linearly interpolated and phases are reconstructed by cubic polynomial interpolation. The noise analysis/synthesis uses the Bark-band modelling of the stochastic signal.

Levine's system includes the transient model, which is not implemented in our system, so the sound qualities are not directly comparable. The bandwidth up to 5 kHz is modelled with sinusoids, the rest up to 16 kHz is modelled by noise. Levine uses a multi-resolution sinusoidal modelling. The bandwidth is divided into three parts. In the first part frequencies between 0 and 1250 Hz are detected using a 46 ms long window (1024 samples), in the second part frequencies between 1250 Hz and 2500 Hz are detected using a 23 ms long window (512 samples) and in the third part frequencies between 2500 Hz and 5000 Hz are detected using an 11.5 ms long window (256 samples). All three types of the windows are overlapped with a 50% overlap. Before the signal is analyzed, it is filtered by the octave-spaced, twice oversampled filterbank. A parameter estimation is done by an F-test algorithm [11] (the method employs a set of orthogonal windows called discrete prolate spheroidal sequences; to treat bias and smoothing problems, an estimate of the spectrum is calculated as a weighted average of several data windows). The trajectory continuation algorithm takes a peak which is the nearest in frequency and amplitude, and at the same time does not cross the threshold. In the synthesis, amplitudes are linearly interpolated and phases are reconstructed with a cubic polynomial interpolation. The noise analysis/synthesis uses the Bark-band modelling of the stochastic signal (analysis windows are 3 ms long (128 samples).

4. Design of the System

Our system was proposed for audio signals saved in *wav* format, with sampling frequency 44 100 Hz. The system was implemented in ANSI C in Linux and uses libraries *sndfile* [4] and *gsl* [5].

Most of the systems do not save the phase information, but our system does. Since we use the method of iterative analysis [14] – this means that we subtract the detected sinusoid from the original signal in the time domain, we need the exact phase and amplitude information. Phase information is needed for the computation of the residual and for better modelling of transients. Although this system was primarily proposed for mono audio signals, it could model stereo channels separately, too. For stereo signals, the phase perception is significant. Hence, phase information is not discarded, as in majority of other systems.

To model the signal in the bandwidth below 8 kHz (most of the systems use 5 kHz threshold), we use multi-resolution sinusoidal modelling [8]. With this method, the bandwidth is divided into more parts, and for each sub-bandwidth a different window length is used. Because of the good time-frequency resolution, we are looking for higher frequencies in smaller frames. This method reduces a pre-echo and improves quality of the signal for a polyphonic audio.

The human ear perceives frequencies between 20 Hz and 16 kHz (some authors mention 20 kHz as the upper border). In our system, the sinusoids represent a bandwidth up to 8 kHz. This approach results in a reasonable computational reduction and an acceptable quality. For most audio signals, it is not improper to model frequencies between 8 kHz and 16 kHz with the noise.

4.1 Sinusoidal Analysis Block

Here, the input signal is divided into frames. Each frame is multiplied by a windowing function. Then, STFT is computed. On its magnitude, the prominent spectral peaks are detected. To obtain better frequency estimation, a quadratic interpolation method is used [7]. Then, the amplitude and phase of the sinusoidal component is estimated (this method will be described later). With the help of these parameters the sinusoid is generated and it is subtracted from the original signal in the time domain. Spectral peaks are detected until their magnitudes are above the threshold. Then, the analysis continues on the next frame.

We wanted to separate sinusoidal components, which are in a 20 Hz distance. According to the sampling frequency the frame length had to be 2205 samples. This seems to be an applicable time-frequency compromise. In the dependency on the windowing function there have to be at least two periods in the frame (so we can detect the sinusoid with the minimum frequency 40 Hz). We use frames with the half-overlap.

In the bandwith below 8 kHz, we model the signal with a utilization of the multi-resolution sinusoidal modelling. We detect spectral peaks in three types of frames (Fig.3). In the longest frame we detect frequencies from 0 to 2 kHz. In the frame with the half-length we detect fre-

quencies between 2 kHz and 4 kHz. In the frame with the quarter-length we detect frequencies from 4 kHz to 8 kHz. Considering that we have divided initial frame to quarters, we have adjusted the length of the frame to 2208 samples.

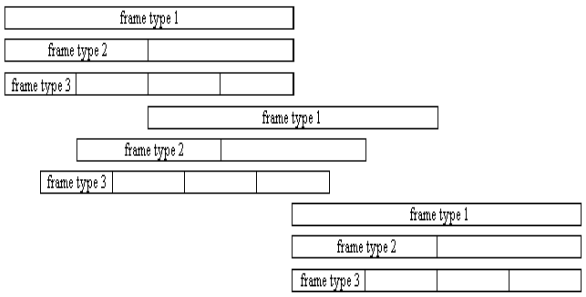

Fig. 3. Overlap of frames.

We have worked with various windows (rectangular, triangular, Hanning, Hamming, Blackman) and we have found out, that the best results were obtained with the rectangular window. This fact has been caused by the following iterative analysis of the residual described in [14]. Since we subtract the sinusoid from the original signal in the time domain, we are not annoyed by the spreading of the spectra, on the contrary the parameters estimation is better. After the detection of parameters we generate the sinusoid and we subtract it from the original signal in the time domain. Then we compute STFT again. As the whole sinusoid is subtracted, there is no need to reduce its sidelobes in the spectrum. We are able to use rectangular windows and we get better frequency estimations and audible improvements in the synthesized signal. When computing STFT, we use the zero-padding. For each frame type there is another zero-padding factor. To obtain the equal frequency resolution, we fill up each frame with zeros to the length of 4416 samples. The frequency resolution is then approximately 10 Hz. Although it will not improve the detection of closely spaced sinusoids, it refines the spectrum and enhances the estimation of the frequency with quadratic interpolation. We detect peaks on a magnitude in dB. For each bandwidth, we have to look for peaks in different range of DFT samples (Tab. 1).

Frequency bandwidth	Frame length (samples)	Zero-padding (samples)	Zero-padding factor	Range of detected DFT samples
0-2kHz	2208	4416	2	0-200
2kHz-4kHz	1104	4416	4	201-400
4kHz-8kHz	552	4416	8	401-801

Tab. 1. Frame lengths, their zero-padding and ranges of DFT samples, where we detect spectral peaks.

One of the reasons why we subtract the sinusoids in the time domain is the method used for amplitude and phase estimation. This method gives much better results than the estimation with an interpolation of nearby DFT samples, but is more computational expensive.

We can write equations for DFT as

$$X(k) = \sum_{n=0}^{N-1} x(n).e^{-j\frac{2\pi}{N}kn} = \sum_{n=0}^{N-1} x(n).\left[\cos\left(\frac{2\pi}{N}kn\right) - j.\sin\left(\frac{2\pi}{N}kn\right)\right] \tag{2}$$

If $X(k) = a_k + j.b_k$ then we can write

$$X(k) = a_k + j.b_k = \sum_{n=0}^{N-1} x(n).\cos\left(\frac{2\pi}{N}kn\right) - j.\sum_{n=0}^{N-1} x(n).\sin\left(\frac{2\pi}{N}kn\right). \tag{3}$$

From this

$$a_k = \sum_{n=0}^{N-1} x(n).\cos\left(\frac{2\pi}{N}kn\right)$$

$$b_k = -\sum_{n=0}^{N-1} x(n).\sin\left(\frac{2\pi}{N}kn\right) \tag{4}$$

Then we obtain the amplitude and the phase from relations

$$A = \sqrt{a_k^2 + b_k^2} \tag{5}$$

$$\varphi = arctg\left(\frac{b_k}{a_k}\right)$$

When we compute a_k and b_k we appoint k for the ratio that determines position of our real detected frequency. Then

$$k = \frac{f}{\frac{f_s}{N}} = \frac{f.N}{f_s}, \tag{6}$$

where N is the length of DFT, f is our detected frequency and f_s is the sampling frequency. An advantage of this method consists in better results of the amplitude and phase computation, but its disadvantage is that it is computationally more expensive.

As we mentioned earlier, we stop the detection of peaks in the actual frame, when the magnitudes of peaks are below our threshold. We use various thresholds for each type of frame, because, when we detect higher frequencies, there is a higher probability that we can detect noise peaks. For the frames with the length of 2208 samples the threshold is 0.5, for the frames with the length of 1104 samples the threshold is 1.0 and for the frames with the length 552 samples the threshold is 2.0. If the threshold was surpassed, we continue with detection on a next frame.

4.2 Sinusoidal Synthesis Block

Here, detected peaks are connected to trajectories and then they are synthesized. Detected peaks are sorted by a frequency and for each trajectory we are looking for the peak, which is closest to the trajectory's frequency [7]. If the distance between the trajectory and the peak is bigger than the permitted value, the sinusoidal peak is not connec-

ted to the trajectory. If the distance is smaller, the sinusoidal peak is connected. If the trajectory did not find its continuation, it will be "turned-off". We do not solve conflicts between trajectories, we just use greedy algorithm. We do not need to solve conflicts, if we have strict criteria for the continuation of trajectories, which will repress most of the conflicts. We also introduced a hysteresis to computation of distances between peaks and trajectories. If the trajectory is "turned-off", the distance is multiplied by 2.0, if the trajectory is "turned-on", the distance is multiplied by 2.5. If the trajectory was "turned-off" for five frames, it will "die" (we will cast it out of our trajectories stack).

After the computation of trajectories for the given frame, synthesis follows. Because the frames were half-overlapped in analysis, the parameters are overlapping too. The synthesis frames are not overlapped, because we use the interpolation of parameters. Therefore we use synthesis frames with half the length of frames from analysis (Tab. 2).

Frequency bandwidth	Frame length (samples)
0-2kHz	1104
2kHz-4kHz	552
4kHz-8kHz	276

Tab. 2. Frame lengths for synthesis.

In the synthesis, we use the linear interpolation of amplitudes and cubic interpolation of frequencies and phases, as described in [6]. The analysis and synthesis frames are compared in Fig. 4.

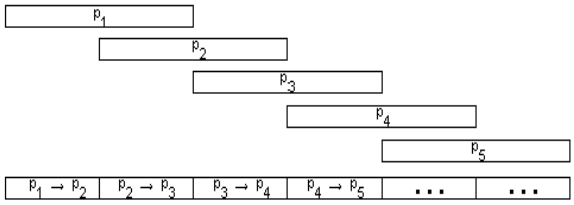

Fig. 4. Overlap of parameters in analysis and synthesis. In synthesis, parameters are interpolated.

4.3 Noise Analysis Block

After the sinusoidal analysis and synthesis the noise analysis follows. In this block, the residual is computed. This residual is analyzed and Bark band energies are computed. We use additive and residual modelling concurrently, because we model the whole residual – remaining frequencies in the bandwidth up to 8 kHz and all the frequencies between 8 kHz and 16 kHz (that have not been modelled yet). Human auditory perception cannot differentiate the change of energy inside the certain frequency bands (Bark bands) for noise-like stationary signals. The bandwidth up to 16 kHz is divided into 25 Bark bands, in which we estimate short-time energy inside the Bark's band [8].

We divide the residual signal into windowed frames with the length of 552 samples (we use Hanning windows). After STFT, we compute the power spectrum from the complex spectrum. From that we compute the corresponding energy for each Bark band.

4.4 Noise Synthesis Block

In the noise synthesis, we create the magnitude of the residual signal. A random phase is added to each component and IFFT is computed from the complex spectrum.

Our experiments have shown that it is reasonable to slightly amplify the magnitude (for instance multiplying the magnitude with the coefficient 1.5). Windowing function is applied to the resulting signal and the signal is added to synthesized sinusoids with the utilization of the overlap-add synthesis [9]. In the overlap-add synthesis, we use frames with the same length as in the noise analysis. These frames are half-overlapped and Hanning windows are applied to them. Fig.5 shows how we add noise frames to synthesized sinusoids.

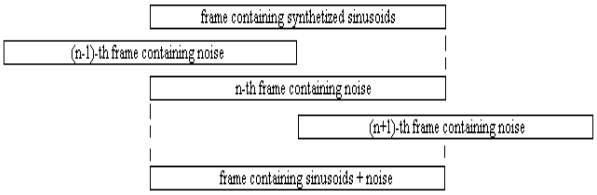

Fig. 5. Adding noise frames to synthetized sinusoids.

5. System Performance

In this section, an example of the performance of our system is shown. The input signal is a sinusoid with frequency of 110 Hz. The white noise with maximum amplitude 0.2 has been added to the sinusoid. This signal has been saved in 16-bit mono wav format. The waveform of the original signal, sinusoidal synthesized signal and the residual are shown in Fig.6.

Fig.7 shows the behaviour of the residual and synthesized noise in the time and frequency domain.

6. Comparison to Other SN Systems

Although there is a large number of SN systems, most of them are not freely available. Our system has been compared to two other systems (proposed by Virtanen and Levine) by a subjective method (listening to the synthesized signals). Both systems use the multi-resolution sinusoidal modelling. Our system uses similar algorithms as these two systems, so it is natural to make a comparison to them.

Virtanen's system uses longer windows (3792, 2028 and 2028 samples) than ours (2208, 1104 and 552 samples), but there is not an audible difference between the frequency detection. We model a smaller bandwidth (up to 8 kHz) than Virtanen (up to 10 kHz). The perceptual difference between the synthetized signals results in the richer sound in our system, but sometimes we have audible chirps

in the higher frequencies. It is caused by our more inferior peak continuation algorithm.

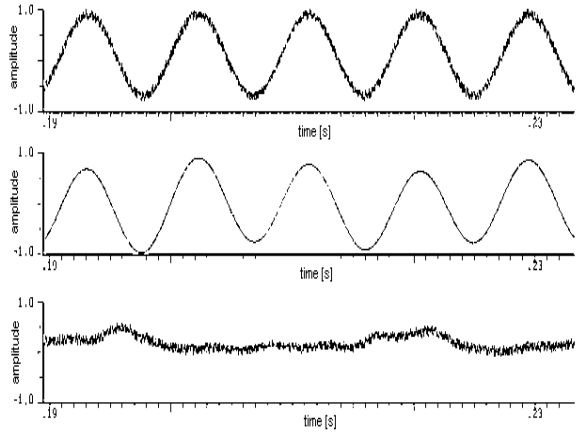

Fig. 6. The original signal, synthesized signal and residual.

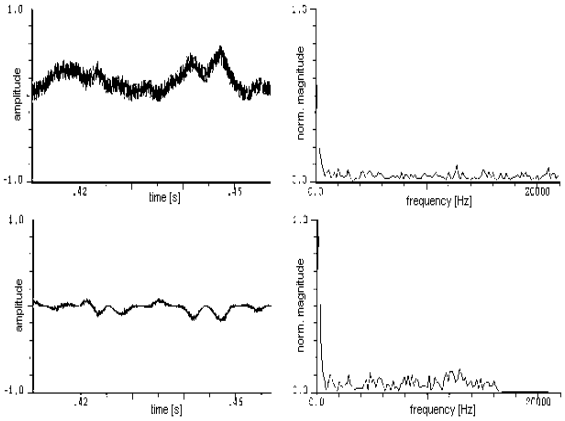

Fig. 7. The the residual and synthetized noise in both frequency and time domain.

It is impossible to directly compare our system to the system proposed by Levine, because we do not use the transient model, as a result the sound quality of transients in our system is worse. Levine uses shorter windows (1024, 512 and 256 samples) than we do. He uses F-test to estimate the peak, we use the combination of spectra interpolation (we add zero-padding in STFT) and quadratic interpolation. He gets better time resolution than we do, and we get better frequency resolution (the parameters of our system were empirically adjusted to get a balanced time-frequency compromise). In addition, we use a wider bandwidth than Levine, and it is audible in the resynthesized signal.

Tab. 4 shows an example of results through which our synthetized signal can be compared briefly with Levine's and Virtanen's. The results were obtained by utilization of objective quality measure methods recommended by ITU-R BS.1387 – ODG (Objective Difference Grade), DI (Distortion Index) and EHS (Harmonic Structure of Error). ODG is a measure of quality comparable to the Subjective Difference Grade (SDG), which is calculated as the difference between the quality rating of the reference and the test signal. The SDG and ODG have a range of [-4;0] where –4 stands for very annoying difference and 0 stands for imperceptible difference between reference and test signal. DI is a quality measure based on the calculated Model Output Variables(MOVs). It is calculated with a trained neural network. EHS is a measure how tonal the noise signal is. The calculation is based on the autocorrelation of the error spectrum. It is impossible to compare the systems directly, because the authors have not published their functional systems, only the original and synthetized signals. We have computed the quality measure of our and the other synthesized signals. The signals are published on Internet [17].

	Our system	Virtanen	Levine
Models used	***SN***	**SN**	**STN**
Sinusoidal modeling bandwidth	***up to 8 kHz***	**up to 10 kHz**	**up to 5 kHz**
Type of noise modeling	***additive + residual modeling***	**additive + residual modeling**	**additive modeling**
Bandwidths for multi-resolution sinusoidal modeling	***20 Hz – 2 kHz*** ***2 kHz – 4 kHz*** ***4 kHz – 8 kHz***	**20 Hz – 200 Hz** **200 Hz – 5 kHz** **5 kHz – 10 kHz**	**40 Hz – 1.25 kHz** **1.25 kHz – 2.5 kHz** **2.5 kHz – 5 kHz**
Analysis windows used	***half-overlapped rectangular windows with lengths of 2208, 1104 and 552 samples***	**half-overlapped Hamming windows with lengths of 3792, 2028 and 2028 samples**	**half-overlapped Hamming windows with lengths of 1024, 512 and 256 samples**
Peak detection	***fixed***	**fixed**	**F-test**
Peak interpolation	***quadratic***	**quadratic**	**quadratic**
Parameter estimation	***iterative analysis of the residual***	**STFT**	**STFT**
Peak continuation	***based on frequency criterion***	**based on parameter derivatives**	**based on frequency + amplitude criterion**
Sinusoidal synthesis	***based on cubic polynomial interpolation***	**based on cubic polynomial interpolation**	**based on cubic polynomial interpolation**
Phase saving	***yes***	**no**	**no**

Tab. 3. Comparison between Virtanen's, Levine's and our system.

As we mentioned earlier, it is not possible to compare directly our system to the system proposed by Levine, since we do not use the transient model. Our system's quality measures and Virtanen's are comparable. In comparison with both systems mentioned, our residual contains more energy than the other two systems (this is the reason why we get worse results of EHS). If we adjusted our thresholds lower, we would be able to take the remaining energy, but it is not needed, because by now we get comparative, in

some cases even better, audible results than the other two systems. In addition, we do not discard phases, so the system could be used for stereo signals, too.

Author of the system	Name of the signal	ODG		DI		EHS	
		Our	Other	Our	Other	Our	Other
Levine	*Mozart - Figaro*	-3.55	-3.08	-2.17	-1.30	0.72	0.39
Virtanen	*Tuomasi Nurmio – Kova Luu*	-3.57	-3.54	-2.21	-2.15	0.64	0.28

Tab. 4. Results of objective quality measure methods.

7. Conclusion

In this work we have focused on the analysis and synthesis of the audio signals with the utilization of the SN model. We have designed the analysis/synthesis system based on SN modelling. Use of the system has been verified by testing on real and synthetic signals.

In designing the system, we have concentrated to characteristics, which allowed us to analyze the signal with the sufficient accuracy, and we have tried to avoid the high computational cost. It is especially a question of a time-frequency compromise and the matter of chosen methods of sinusoidal detection, because the sinusoidal analysis takes a serious part in analysis and synthesis process. Our system has been designed to be able to work with polyphonic audio signals, too. For this purpose we have used the multiresolution sinusoidal modelling. We have used the iterative analysis of the residual signal in the system. With the help of this method, we were able to dramatically reduce the detection of the sinusoid sidelobes.

The experiments [17] have shown that the sound quality is comparable to the other SN systems, although the systems that use transient model (STN systems) obtain better quality measure results. Our residual contains more energy for reasons of higher threshold and for the windows not optimized for time localization but for good frequency estimation. On the other hand, we do not discard the phases and our phase estimation algorithm leads to better analysis and synthesis of stereo signals.

Acknowledgements

This work has been supported by the Grant Agency of the Slovak Republic, grant No. VEGA 1/0146/03 and VTP 1003/2003.

References

[1] SPANIAS, A. Speech coding: A tutorial review. In *Proceedings of the IEEE*. 1994, vol. 82, no. 10. p. 1541–1582.

[2] SERRA, X. A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition. *Ph.D. thesis.* Stanford University, 1989.

[3] SERRA, X. Musical Sound Modeling with Sinusoids plus Noise. *Musical signal processing*. 1997, Roads C.& Pope S. & Picialli G. & De Poli G., Swets & Zeitlinger Publishers.

[4] GALASSI, M. et al. The GNU Scientific Library. *http://sources.redhat.com/gsl/,* November 2003.

[5] CASTRO LOPO, E. libsndfile. *http://www.zip.com.au/~erikd/libsndfile/#Download*, November 2003.

[6] VIRTANEN, V. Audio signal modeling with sinusoids plus noise. *MSc Thesis*, Tampere University of Technology, August 2000.

[7] SMITH, J.O., SERRA,X. PARSHL: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation. In *Proc. of the International Computer Music Conference*, 1987.

[8] LEVINE, S., SMITH, J.O. A Sines+Transients+Noise Audio Representation for Data Compression and Time/PitchScale Modifications. *105th Audio Engineering Society Convention*, San Francisco, 1998.

[9] GOODWIN, M. Adaptive Signal Models: Theory, Algorithms, and Audio Applications. *Ph.D. thesis*, University of California, Berkeley, 1997.

[10] McACULAY, R.J., QUATIERI, T.F. Speech Analysis/Synthesis Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech, And Signal Processing*, August 1986, vol. 34(4).

[11] THOMSON, D. J. Spectrum Estimation and Harmonic Analysis. In *Proceedings of the IEEE*, 1982, 70(9).

[12] FITZ, K. LEMUR. *http://www.cerlsoundgroup.org/Lemur/*, April 2004.

[13] BEAUCHAMP, J. SNDAN. *http://ems.music.uiuc.edu/~beaucham/software/sndan/*, April 2004.

[14] VIRTANEN, T. Accurate Sinusoidal Model Analysis and Parameter Reduction by Fusion of Components. *AES 110th convention*, Amsterdam, Netherlands, May 2001.

[15] ZWICKER, E., FASTL, H. *Psychoacoustics: Facts and Models*. Berlin Heidelberg: Springer-Verlag, 1990.

[16] STYLIANOU, Y. Applying the Harmonic plus Noise Model in Concatenative Speech Synthesis. *IEEE Transactions on Acoustics, Speech, and Signal Processing,* January 2001, vol. 9, no. 1, p. 21-29.

[17] TURI NAGY M., ROZINAJ G. *http://www.ktl.elf.stuba.sk/projects/audio/sn/index.htm*, November 2004.

About Authors...

Martin TURI NAGY was born in 1980 (Bratislava). In 2004 he received MSc. degree from the Slovak University of Technology in Bratislava, Slovak Republic. Nowadays he continues studying as a postgradual student.

Gregor ROZINAJ received MSc. degree from the Slovak University of Technology in Bratislava, Slovak Republic in 1981 and PhD at the same university in 1990. Now he works as an Associate Professor at the Department of Telecommunications, Slovak University of Technology in Bratislava.