The inaugural International Conference on Music Communication Science 5-7 December 2007, Sydney, Australia
http://marcs.uws.edu.au/links/ICoMusic

A spatialized additive synthesizer

Charles Verron[(1)(2)], Mitsuko Aramaki[(3)], Richard Kronland-Martinet[(2)],Gregory Pallone[(1)]
[(1)] Orange Lab France Telecom R&D, 22307 Lannion, France
[(2)] CNRS Laboratoire de Mécanique et d'Acoustique, 13402 Marseille, France
[(3)] CNRS Institut de Neurosciences Cognitives de la Méditerranée, 13402 Marseille, France

ABSTRACT

In virtual auditory environments, sound generation is typically based on a two-stage approach: synthesizing a monophonic signal, implicitly equivalent to a point source, and simulating the acoustic space. The directivity, spatial distribution and position of the source can be simulated thanks to signal processing applied to the monophonic sound. A one-stage synthesis/spatialization approach, taking into account both timbre and spatial attributes of the source as low-level parameters, would achieve a better computational efficiency essential for real-time audio synthesis in interactive environments. Such approach involves a careful examination of sound synthesis and spatialization techniques to reveal how they can be connected together. This paper concentrates on the sinusoidal sound model and 3D positional audio rendering methods. We present a real-time algorithm that combines Inverse Fast Fourier Transform (FFT-1) synthesis and directional encoding to generate sounds whose sinusoidal components can be independently positioned in space. In addition to the traditional frequency-amplitude-phase parameter set, partials positions are used to drive the synthesis engine. Audio rendering can be achieved on a multispeaker setup, or in binaural over headphones, depending on the available reproduction system.

1. SINUSOIDAL MODELING

In [MAQ86] McAulay and Quatieri present a complete system for speech coding based on sinusoidal analysis/synthesis. The model they describe consists of representing a sound as a sum of time varying sinusoidal components. Amplitude, phase and frequency of partials are extracted from the Short-Time Fourier Transform (STFT) of the original sound, and used as parameters to drive the synthesis engine. The sinusoidal representation is also adapted to a broad range of audio signals, such as musical or environmental sounds. For example sounds produced by vibrating solids are well simulated by sums of damped sinusoids [Cook02]. The partial's frequencies and damping factors can be estimated from the analysis of natural sounds [AKM06] or from analytical solutions of the mechanical model expressed from the object shape and material. Similar techniques are used for modeling harmonic sounds produced by many musical instruments. Sinusoidal analysis/resynthesis allows many effects like filtering, pitch-shifting and time-stretching in computer music. The SDIF Sound Description Interchange Format [SW00] has been created to store, play and manipulate sinusoidal representations. The model has also been significantly extended to take noisy components into account, leading to a 'sinusoid+noise' representation [SS90].

2. FREQUENCY-DOMAIN SINUSOIDAL SYNTHESIS

Sinewaves synthesis can be done either in time or frequency domain. In [RD92] Depalle and Rodet propose a complete frequency-domain additive synthesizer. Compared to time-domain sinewave synthesis, their method allows to drastically reduce computational complexity, without affecting perceived sound quality. This algorithm is the kernel of the 3D synthesis engine presented in part 4. It consists of creating Short Time Spectra (STS) by successively adding a spectral motif, with specific amplitude and phase, at desired frequency for each partial. STS are inverse fast Fourier transformed (IFFT) and overlap-added (OLA) to obtain the synthetic sound. Let *U(f)* be the STS and *W(f)* the spectral motif. Amplitude, frequency and phase parameters are noted $[A_m,F_m,P_m]$. *U(f)* is given by:

$$U(f) = \sum_{m=1}^{M} A_m e^{jP_m} W(f - F_m)$$

where *M* is the total number of partials. The sampled version *U[k]* of this spectrum is calculated and IFFT-OLA. This procedure is illustrated in figure 1, for *M* disjoint partials.

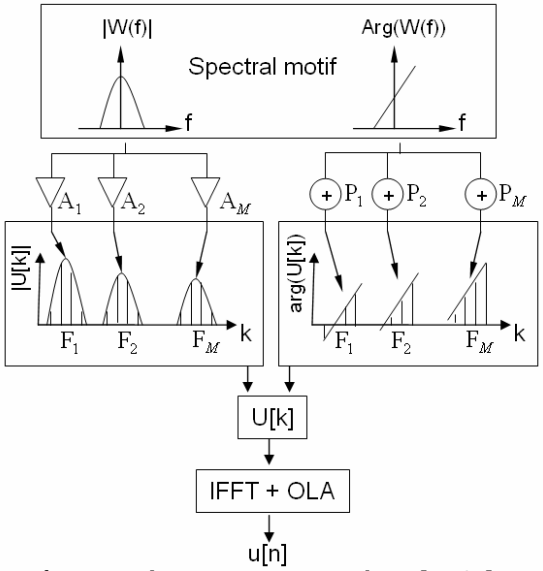

Figure 1: frequency-domain sinewaves synthesis [RD92]

The spectral motif is the complex spectrum of a time-domain window *w(t)*. The crucial reduction factor on the algorithm's computational complexity relies on the choice of this window. If window's energy is sufficiently concentrated in a narrow low

frequency band, then the complex spectrum can be truncated without losing much information. It results in reducing the number of complex multiplications and additions necessary to add each partial, because adding the narrow-band spectral motif involves modifications of the sampled STS *U[k]* only for a few frequency bins. However, to obtain the perfect reconstruction of a constant amplitude sinewave, *w[n]* (sampled version of *w(t)*) should also satisfy the condition:

$$\sum_{i=-\infty}^{+\infty} w[iL+n] = 1 \quad \forall n$$

where *L* is the synthesis stride. This constraint is compatible with a narrow-band spectral motif only when the stride is small compared to the window size. A 75% overlap can be used but it does not optimize the algorithm efficiency. Rodet and Depalle propose to use a second window after the IFFT that allows to reduce the overlap to 50% [RD92]. The drawback, important for our application, is the difficulty to filter the STS in the frequency domain before the IFFT. Consequently we choose to keep the simple approach of a single window with a small synthesis stride. We use a "digital prolate spheroidal" window [VBM96] weighted to satisfy the perfect reconstruction constraint with 75% overlap between blocks. The resulting spectral motif is sharp so that synthesizing sinusoidal blocks (of any size) only requires 7 points per partial when building the STS.

3. 3D SOUND SOURCE POSITIONING

For versatility, a 3D sound synthesizer should be independent of the available reproduction setup and compatible with most of existing positional audio methods and their directional encoding/decoding scheme. This part quickly reviews techniques to position a monophonic point source in a virtual 3D space. A more detailed overview of 3D audio encoding and rendering can be found for example in [JLP99]. The source, placed in the spherical coordinate system shown in figure 2, is assumed to radiate a plane wave in the listener direction.

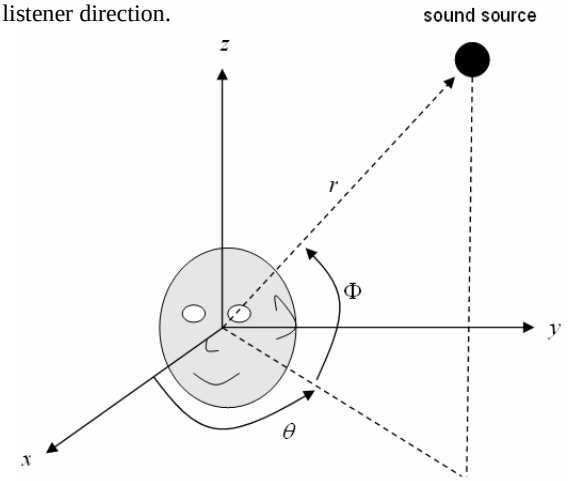

Figure 2: egocentric spherical coordinate system

Among 3D positional audio methods, some aim at simulating the localization cues at the ear canal entrance (binaural techniques), other at reproducing the sound field properties at the "sweet spot" (Discrete panning techniques [Sch71] [Pul97], Ambisonics [MM95]) or in a relatively extended area (Higher Order Ambisonics, Wave Field Synthesis [DNM03]). Binaural techniques are mainly dedicated to headphone reproduction but can be extended to multispeaker configurations by means of crosstalk cancellation. Other methods require multichannel playback systems. Apart from Wave Field Synthesis, which uses time differences to create virtual sources, all positioning methods can be implemented by applying a vector of gain factors to the original monophonic sound. This is the key point to connect directional encoding to the FFT-1 synthesis algorithm described in part 2.

First-order Ambisonics

We only present first-order Ambisonics (B-format) equations that can be found in [MM95]. Let *u[n]* be the monophonic sound to be spatialized. B-format encoded signals are:

$$X[n] = \cos(\theta)\cos(\Phi)u[n] \qquad Y[n] = \sin(\theta)\cos(\Phi)u[n]$$

$$Z[n] = \sin(\Phi)u[n] \qquad W[n] = 0.707u[n]$$

According to the playback system characteristics (number and placement of speakers) a decoded multichannel signal is calculated. For a 2D square system, the 4-channel sound would be:

$$LF[n] = W[n] + 0.707(X[n] + Y[n])$$

$$RF[n] = W[n] + 0.707(X[n] - Y[n])$$

$$LB[n] = W[n] + 0.707(-X[n] + Y[n])$$

$$RB[n] = W[n] + 0.707(-X[n] - Y[n])$$

Each decoded channel is obtained by matrixing the encoded components (*Z[n]* does not appear in the equations because it is a 2D playback system example). Consequently each decoded channel is the original signal *u[n]* weighted by a gain factor.

Discrete panning techniques

Discrete panning techniques consist in selectively feed the loudspeakers closest to the virtual source position in the playback system. The set of gain factors applied to each channel can be viewed as the directional encoding. No directional decoding is performed since encoded signals directly feed the loudspeakers. Directional gains can be calculated using amplitude or intensity panning laws. The commonly used Vector Base Amplitude Panning is a generalization of amplitude panning for 3-dimensionnal loudspeaker distributions [Pul97]. The sphere surrounding the listener is divided into loudspeaker triplets. The monophonic signal feeds only the triplet comprising the attended source location, with appropriate gain factors. Virtual source sharpness and location accuracy vary according to the number of loudspeakers and their distribution in space. A virtual spreading of the sound source is also possible by feeding loudspeakers located around the original triplet.

Multichannel binaural synthesis

Binaural synthesis consists in filtering a monophonic signal by the HRTF measured at the desired position for reproduction over headphones. Alternatively, the HRTF set can be decomposed into a

set of filters and frequency independent directional functions. In [JWP06] the authors propose several implementations based on this decomposition, compatible with Ambisonics and discrete panning. The approach is illustrated on figure 3. It consists in:

- creating a directionally encoded multichannel signal for each virtual source, by applying a set of gain factors $\{G_1,\ldots,G_C\}$ (eventually given by ambisonic or discrete panning functions).
- mixing all sources in the encoded domain, channel by channel.
- post-filtering the resulting multichannel signal with a unique set of HRTF and down-mixing to 2 channels.

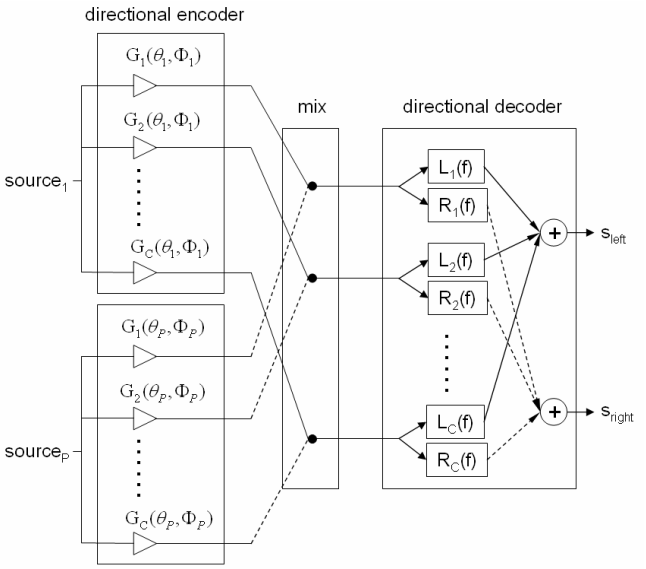

Figure 3: binaural synthesis by multichannel panning and HRTF post-filtering.

4. 3D FFT-1 SOUND SYNTHESIS

The algorithm presented here is an original combination of previously described techniques. It uses FFT-1 sinusoidal synthesis and directional encoding to efficiently generate spatialized, time-varying sinewaves. The principle is to generate an encoded multichannel STS $\{U_1[k],U_2[k],\ldots,U_C[k]\}$ that intrinsically contains partials position. As with positional audio systems, perceived partial's location accuracy increases with the number of channels *C*. The construction of each channel consists in successively adding the spectral motif, as described in part 2, except that an additional directional gain $G_c(\theta_m,\Phi_m)$ is introduced to encode partial's position. The continuous spectrum for channel *c* with *M* partials is expressed as:

$$U_c(f)=\sum_{m=1}^{M} G_c(\theta_m,\Phi_m)A_m e^{iP_m} W(f-F_m)$$

Directional gain factors can be calculated either by discrete panning or ambisonic techniques. They can be computed for any speaker configuration, ensuring the synthesizer can be used with any playback system. The multichannel STS magnitude is illustrated on figure 4, for *M* disjoint partials.

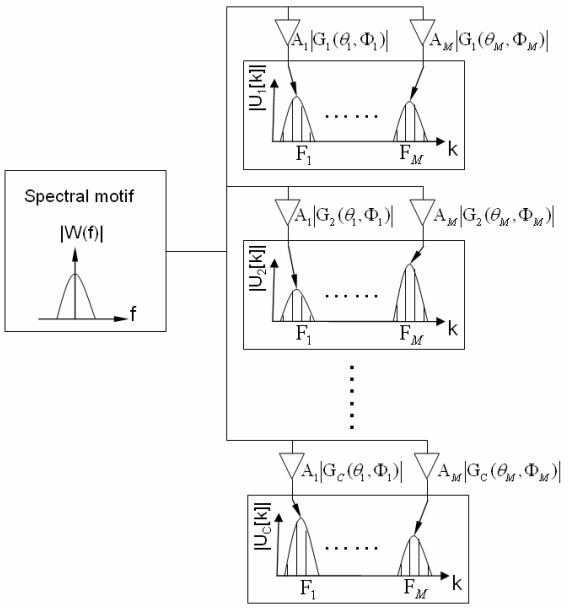

Figure 4: directionally encoded multichannel STS magnitude

Two rendering algorithms can be used according to the reproduction system, as shown on figure 5. For a multichannel playback system, the directionally encoded multichannel STS is inverse Fourier transformed to feed the loudspeaker without any additional processing (except the regular matrixing when $G_c(\theta_m,\Phi_m)$ are provided by an ambisonic encoder). For reproduction over headphones, multichannel binaural is used, HRTF post-filtering being applied to each channel in the frequency domain.

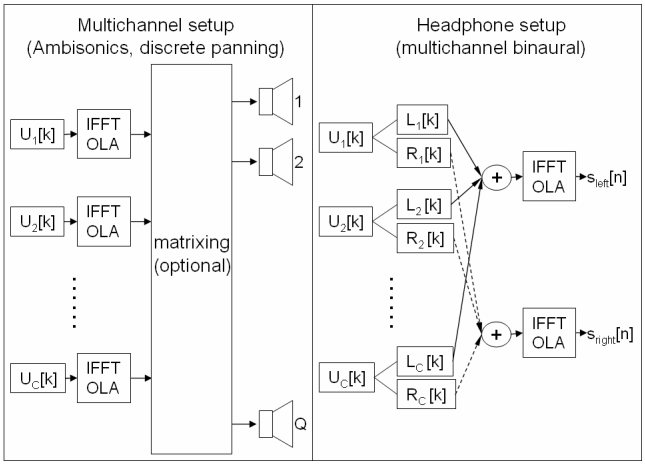

Figure 5: rendering algorithm (multispeaker or headphone setup)

The two-stage approach (sound synthesis then spatialization) applies directional gains on time-domain signals. If we assume partials to be pre-synthesized and accessible as separated waveforms, *C* multiplications per sample are still necessary for each of them, to encode the position. The one-stage approach applies

directional gains $G_c(\theta_m,\Phi_m)$ directly to partial's amplitude parameter A_m. This way CK/L complex multiplications per sample are necessary to synthesize one partial and encode its position (K is the size of the spectral motif). Typically we choose a block size N=512, L=N/4 (75% overlap) and K=7. Consequently the number of multiplications required for each directionally encoded partial is approximately 6 times lower with the one-stage approach than with the two-stage approach (considering a complex multiplication equivalent to 3 real ones). For multispeaker playback systems, the drawback is that our algorithm requires C extra IFFT. In the binaural case however, the one-stage approach is particularly interesting. It requires only 2 IFFT and makes it possible to apply the HRTF filtering in the frequency domain without any FFT calculation. The algorithm has been implemented using VBAP directional functions and 10 channels in 2D. Binaural sound examples are available at www.lma.cnrs-mrs.fr/~kronland/spatsynth/index.html. Hundreds of partials can be synthesized and distributed in space in real time. Partials can also be given several positions simultaneously. It gives new control possibilities to generate diffuse sources in virtual auditory environment, by spreading source's sinusoidal components in space (illustrated on figure 6). Perceptual effects of this technique remain to be evaluated. A comparison with source width rendering by decorrelation methods [JWP06] [PB04] has also to be done in the near future.

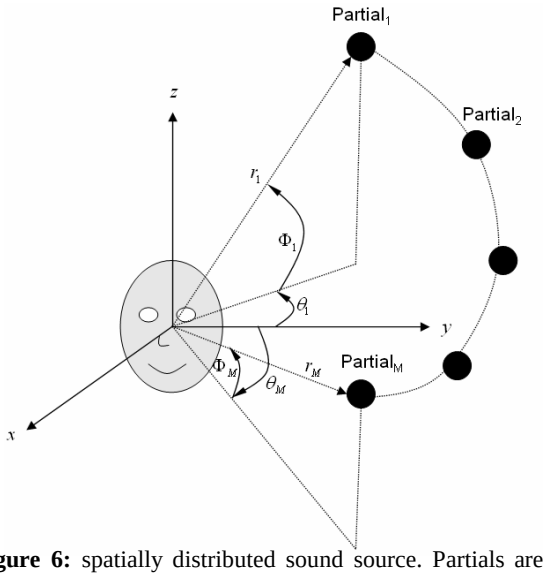

Figure 6: spatially distributed sound source. Partials are spread around the listener

5. CONCLUSION

A spatialized additive synthesizer has been described. Our algorithm results in efficient real-time synthesis of multiple sound sources positioned around the listener. Sinusoidal components can be individually distributed in space to simulate diffuse sources. Musical effects may be achieved by assigning independent trajectories to each partial. The synthesized sound is adapted to any listening configuration, from headphones to arbitrary mutichannel loudspeaker setup. Further investigations will be carried out to integrate noise in the synthesizer, to make it fully compatible with the 'sinusoid+noise' model. Perceptual effects of sinewaves spatial dispersion also need to be fully investigated for different types of sounds.

6. ACKNOWLEDGEMENTS

The authors thank Philippe Depalle for his help on sinusoidal modeling and frequency-domain sound synthesis.

7. REFERENCES

1. [MAQ86] R. J. McAulay and T. F. Quatieri. "Speech analysis/synthesis system based on a sinusoidal representation", IEEE Trans. ASSP, 34(4), August 1986.

2. [Cook02] P. R. Cook, "Real Sound Synthesis for Interactive Applications", A. K Peters Ltd., Natick, Massachusetts, 2002.

3. [AKM06] A. Aramaki and R. Kronland-Martinet, "Analysis-Synthesis of Impact Sounds by Real-Time Dynamic Filtering", IEEE Trans. ASSP, 14(2), March 2006.

4. [SW00] D. Schwarz, M. Wright, "Extensions and Applications of the SDIF Sound Description Interchange Format", Proc. ICMC, Berlin, August 2000.

5. [SS90] X. Serra and J. O. Smith. "Spectral modeling synthesis: A sound analisys/synthesis system based on a deterministic plus stochastic decomposition", Comp. Music. J., 14(4), 1990.

6. [RD92] X. Rodet and P. Depalle, "Spectral envelopes and inverse FFT synthesis", Proc. of the 93rd AES Conv., 1992.

7. [VBM96] T. Verma, S. Bilbao, T. H. Y. Meng, "The Digital Spheroidal Prolate Window", IEEE Proc. ASSP Conf., 1996.

8. [JLP99] J.-M. Jot, V. Larcher, J.-M. Pernaux, "A Comparative Study of 3-D Audio Encoding and Rendering Techniques", Proc. 16th Int. Conf. AES, March 1999.

9. [Sch71] J. M. Chowning, "The Simulation of Moving Sound Sources", JAES, 19(1), 1971.

10. [Pul97] V. Pulkki, "Virtual Sound Source Positioning Using Vector Base Ampliude Panning", JAES, 45(6) 1997.

11. [MM95] D. G. Malham and A. Myatt, "3-D Sound Spatialization Using Ambisonic Techniques", Computer Music Journal, 19(4), 1995.

12. [DNM03] J. Daniel, R. Nicol, S. Moreau, "Further Investigations of High Order Ambisonics and Wavefield Synthesis for Holophonic Sound Imaging", Proc. of the 114th AES Conv., March 2003.

13. [JWP06] J.-M. Jot, M. Walsh, A. Philp, "Binaural Simulation of Complex Acoustic Scene for Interactive Audio", Proc. of the 121th AES Conv., Oct. 2006.

14. [PB04] G. Potard, I. Burnett, "Decorrelation Techniques for the Rendering of Apparent Sound Source Width in 3D Audio Displays", Proc. Int. Conf. on Digital Audio Effects, 2004.