Spatialized real-time additive resynthesis in Max/MSP using Spat and Sigmund

Kelian (Mike) Li | kli11@gmu.edu | www.likelian.net/projects
School of Music| George Mason University

ABSTRACT

This research attempts to integrate both real-time additive resynthesis and spatialization in the time domain. The partials are spatialized randomly but controlled by the users. The output signal can be either rendered to speakers or headphones. This project is developed in a visual programming environment, which is easy to access and modify by non-coders.

INTRODUCTION

Additive synthesis is one of the oldest sound synthesis methods developed. It creates sound by summing elementary waveforms (primarily sinusoids). Based on *discrete time Fourier transform* (DTFT), any sound can be analyzed into a set of sinusoids. Thus, an oscillator bank can regenerate the analyzed sound from given data.

Spatial elements have been part of music composition since ancient times. In 1973, Michael Gerzon proposed Ambisonics [1], a matrix scheme, that allows spatial recording, encoding, decoding, and playback.

A new method that integrates sound synthesis and spatialization is needed for more delicate music creation purposes.

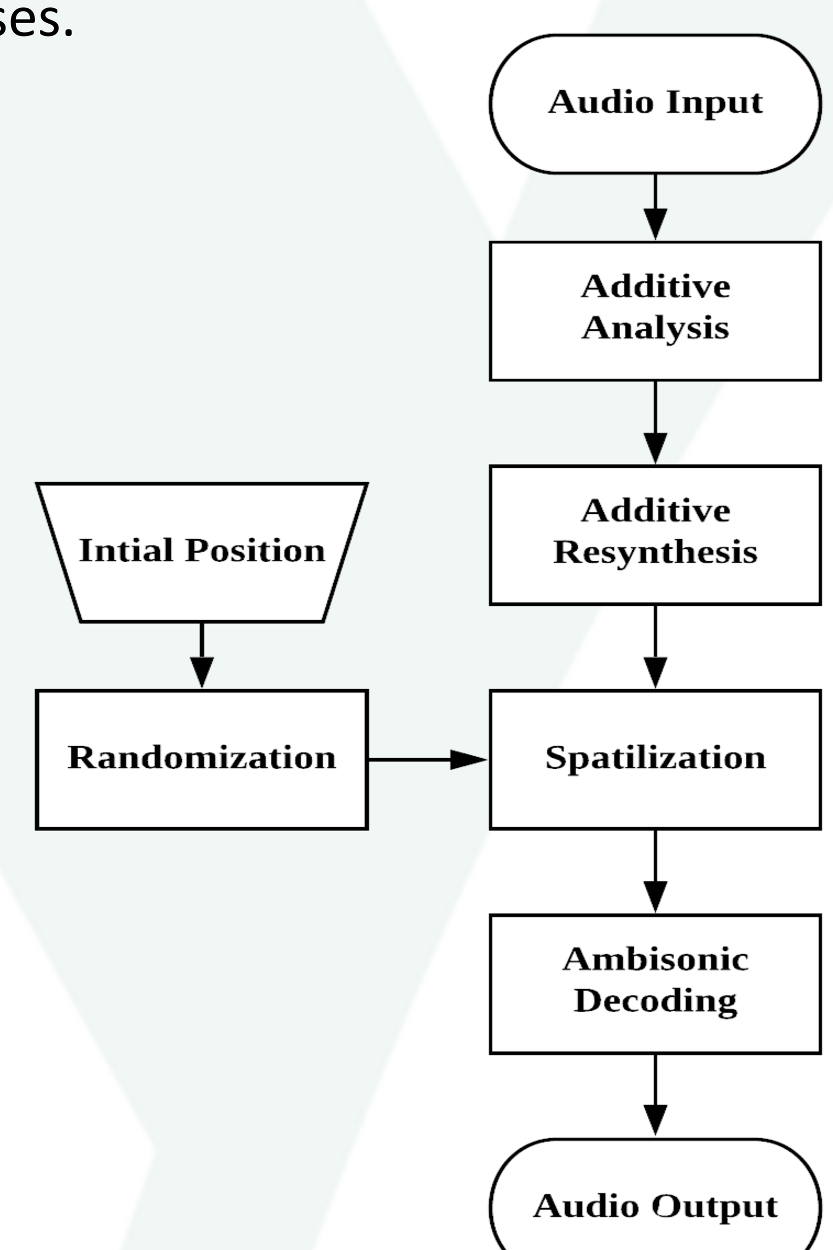

Figure 1 Overview

METHOD

- **Additive Analysis and Resynthesis**

Sigmund~ [2] is an object in Max/MSP that analyzes audio signals in real-time tosinusoidal components. After the analysis, a Max external Odot [3] is introduced to organize the data. It mainly uses a C-like programming syntax in code-boxes to process Open Sound Control (OSC) [4] messages.

In order to resynthesize the input audio, the *migrator* algorithm [4] is employed to smoothly approximate the analyzed sound in real time. it updates one oscillator at a time with a new frequency. Each oscillator only produces one static frequency for a fixed amount of time through an envelope.

Figure 2 Randomized positions

- **Sound Spatialization**

Spat~ 5 software package [5] is adapted for sound spatialization purposes. The first spat5.oper object determines the center spatial location. The route object and the o.pack object together gather *azimuth*, *elevation,* and *distance.* To control and display the randomness amount of 3 parameters in the 2-dimensional screen, a nodes object and a multislider are applied.

After this OSC bundle is sent to another spat5.oper object, more detailed data like orientation, diffusion, and filters are added, and a new OSC bundle is created for the spatial processing in spat5.spat~ object.

DISCUSSION/CONCLUSIONS

- **Limitations**

The primary limitation of this patch is the high demand on CPU power. Processing 128 sources in Spat resulted in significant distortion and audible delays. Hence, resolution was reduced down to 35 voices. As a result, this patch is unable to simulate real-world recorded samples properly and deceive human hearing. One possible solution is to employ two or more computers and divide the Max portion and the MSP portion to a different computer.

- **Future Works**
  - i. Controlling mechanism
  - ii. Other 3D geometrical data
  - iii. Randomization based on partials' amplitudes
  - iv. Localization
  - v. Walsh function synthesis
- **Conclusion**

A real-time additive resynthesis patch that is dedicated for spatialization in the time domain has been presented. It successfully spatializes partials randomly in space, and users have control over the amount of randomness. Users can also change the initial center position of the sound sources. The trade-off of the spatialization is increased computational cost.

ACKNOWLEDGEMENTS

Thanks to Jeffrey Lubow and Dr. John MacCallum at CNMAT, Dr. Jesse Guessford and Dr. Robert Gillam at George Mason University, and Fernando Lopez-Lezcano and Dr. Matthew Wright at CCRMA.

REFERENCES:

[1] Gerzon, Michael A. "Periphony: With-height sound reproduction." *Journal of the Audio Engineering Society* 21, no. 1 (1973): 2-10.
[2] Puckette, Miller. Cort Lippe and Ted Apel. "Sigmund~ Sinusoidal Analysis and Pitch Tracking". Version .05 for Mac Max/Msp 5. http://crca.ucsd.edu/~tapel/software.html
[3] Freed, Adrian, John MacCallum, and Andrew Schmeder. "Dynamic, Instance-based, object-oriented programming in Max/MSP using open sound control message delegation." In *ICMC*. 2011.
[4] Wright, Matthew, and Adrian Freed. "Open SoundControl: A New Protocol for Communicating with Sound Synthesizers." In *ICMC*. 1997.
[5] Carpentier, Thibaut. "A new implementation of Spat in Max." In *15th Sound and Music Computing Conference (SMC2018)*, pp. 184-191. 2018.