

StochGran on OSX

Mara Helmuth

College-Conservatory of Music Center for Computer Music, University of Cincinnati
email: mara.helmuth@uc.edu

Abstract

StochGran, the Cmix-based granular synthesis composition application, has been ported to Macintosh OSX Cocoa with OpenGL graphics and RTCmix 3.1.0 synthesis. A redesigned interface is easier to use, with various enhancements to streamline the composer's actions to produce sound. Graphical representation of the probability distributions as well, as the previously implemented shape of change curves, now allows graphical editing of the extreme distributions. Graphical control of algorithmic compositional parameters allows the composer to specify preferred parameters, yet keep enough randomness in the process to allow unexpected and interesting results. The demonstration will include sounds and piece excerpts by UC-CCM students and myself.

1 Introduction

With the proliferation of options for granular synthesis composition tools and music synthesis languages such as MAX MSP and Supercollider, I was originally uncertain about the need to port StochGran to current platforms. However, StochGran has several strengths which I have not found easily duplicatable in other software. First, the number of degrees of freedom in the underlying Cmix programs, sgran and stgran, is quite high. This is extremely difficult to match in a real time music language, as so many parameters must change independently and simultaneously to create sophisticated timbral gestures. Real time control of all of these parameters becomes a serious issue. Second, graphical control of the shape of change functions is not easily reproducible in environments such as Supercollider (2.x). Finally, free, downloadable public domain software with modifiable source code allows musicians to share code and develop quickly without the constraints of commercial contracts. This approach is attractive philosophically, and artistically freeing.

2 Background

Earlier work by Iannis Xenakis, Curtis Roads and Barry Truax was mentioned in previous papers. I first programmed StochGran for stochastic granular synthesis on NeXTstep with Objective C. The Cmix

synthesis instrument on which the application was based had 36 parameters for each sound event and used probability distributions to control grain parameters. [Helmuth 1991] Later I ported the application to SGI Irix 6.2 with Rapidapp generated Motif interface, and C++. [Helmuth 1998] Sampling was added, to process input soundfiles with an analogous stochastic windowing. The sampling instrument had 50 parameters per event, as input sound controls were added. Algorithmic options were added, but there was not much flexibility in this area. The program has been in extensive use by myself, students at UC-CCM and some others. John Gibson has included versions of the underlying instruments "sgran" and "stgran" in RTcmix 3.1.0, which runs on Linux and OSX. Because the OSX development environment is closely related to the original NeXTstep, Objective C is a clean and powerful language, and OSX Aqua is beautiful, I was eager to create an OSX StochGran.

3 Design Improvements

A new graphical mode of displaying the probability distributions enables the composer to predict better the range of sound an event can contain, and to edit it more easily, than the previous slider manipulation method. Also, a simplification of certain GUI features make the process of running the program less confusing for beginners.

3.1 Streamlining the Process

The file saving method involved writing a Cmix scorefile to disk, writing the soundfile, and then playing the sound with a button. These three steps are now compressed into one: hitting the play button. There might be a slight delay due to the soundfile being written to disk, so the composer is informed of the state of the program through the status field, which may display "Waiting for input" or "Playing." I eliminated some of the rarely used features on tabs behind the main tab, for advanced waveform control. Synthesis or sampling is now controlled on one display window, for clarity. Better error checking is implemented with more alert panels. Graphical control of the shape of change functions operates similarly to previous versions. The sound examples for the CDROM are made with the following curves, and different distribution extremes.

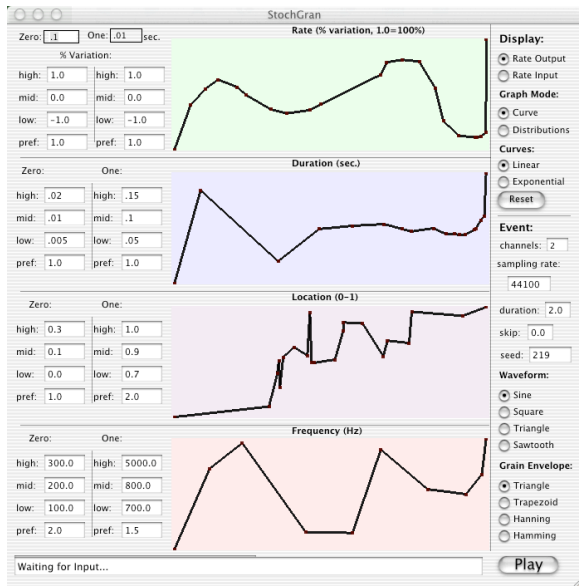


Figure 1. StochGran main window, with four shape of change functions displayed. 1- toggles input/output rate. 2- duration. 3- location (if sampling, amplitude). 4- frequency (if sampling, transposition). The fields on the left specify probability distribution range and preference values. On the right are envelope, waveform and display choices.

4 Algorithmic Composition

I was interested in expanding the algorithmic capabilities of StochGran, which had become the part of the program I most frequently used. In the IRIX StochGran, the C subprograms sgscore and stscore generated a specified number of Cmix scorefiles from a very wide range of possibilities. Usually one sifts through many files to find a few good sounds. In this version of StochGran one can now specify the preferred values and amount of preference for each parameter graphically. Consequently one has much better control over what types of sounds are produced, and can work more efficiently. A single color-coded display indicates the range limits, preferred value and amount of preference, for “zero” distributions in blue, and for “one” distributions for red. In this case as well as with the previously mentioned graphical representations of probability distributions, clarity enhances both learning the application as well as more spontaneous use by experienced composers. While controlling only some of the parameters graphically may be seen as a limitation, it would be unwieldy to control each one of the 200 values needed to generate the 50 parameters for sampling. A balance between clarity and flexibility is maintained.

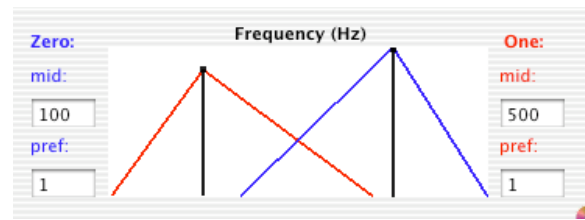


Figure 2. Editable graph for the frequency parameter on the Algorithms window. The events generated will move between the red and blue distributions as controlled by randomly generated shape of change functions. The black lines indicated the preferred point in the distribution, and the height of the top peak indicates the amount of preference for that point.

5 Future Research

A future enhancement, which should be implemented by summer 2002, is better drawing of the probability distributions (fig. 2), which indicate a more realistic representation of grain densities.

A Linux version of StochGran is also planned.

In a related project, I hope to implement real time control with the already written real time granulation instruments, with a hardware controller device and substantially different control interface.

References

- Helmuth, M. 1991. “Patchmix and StochGran: Two Graphical Interfaces.” *Conference*. International Computer Music Association, pp. 563-566.
- Helmuth, M. 1998. “StochGran on Silicon Graphics IRIX.” *Conference*. International Computer Music Association, pp. 251-252, and CDROM.