Chapter 14. Meeting 14, Approaches: Granular and Concatenative Synthesis

14.1. Announcements

- Musical Design Report 3 due 6 April
- Sonic system draft due: 27 April
- Next couple of weeks: need to meet with me to talk about sonic system projects
- Quiz on Thursday

14.2. Musical Design Report 3

- Must be primarily built with rendered digital audio, such as output from Csound, PD, or related tools
- Density, and contrasts in density, must be a significant compositional parameter
- Must feature granular, concatenative, or sound montage synthesis techniques in some manner
- Should have at least one transition between disparate material that is a gradual morph, fade, or dove-tail
- Can be composed with athenaCL, athenaCL and other tools, or other tools alone
- Mixing audio obtained from PD and/or athenaCL/Csound in Audacity or a DAW is highly recommended.

14.3. Listening: Vaggione

- Listening: Horacio Vaggione, *24 Variations*, 2002

14.4. Reading: Roads, Introduction to Granular Synthesis

- Roads, C. 1988. "Introduction to Granular Synthesis." *Computer Music Journal* 12(2): 11-13.
- What are some common duration ranges and grains per second used in granular synthesis?
- Gabor's quanta

- Xenakis's screens and books of screens

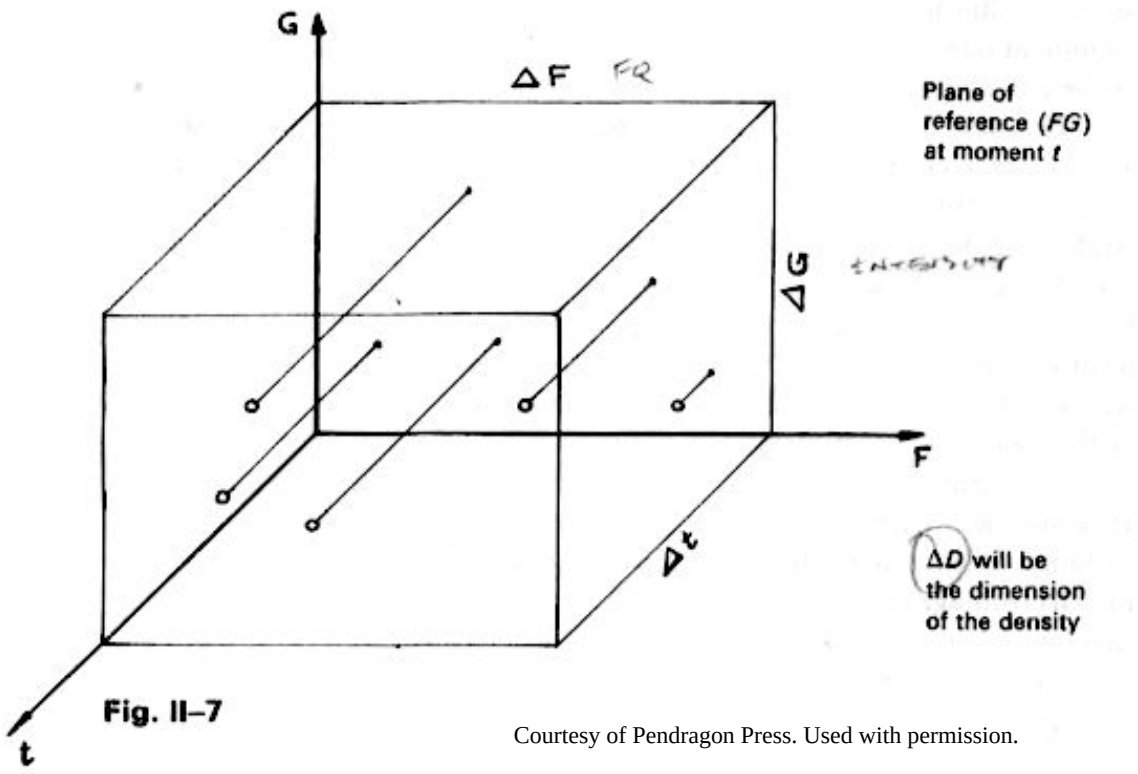


Fig. II–7

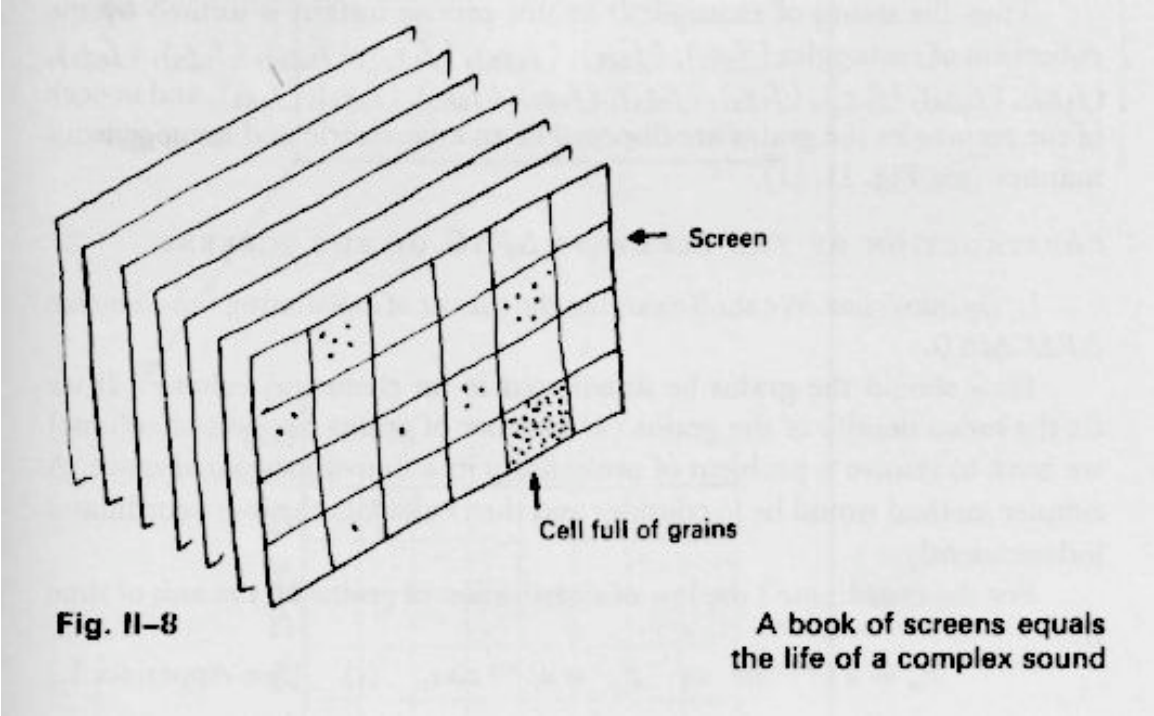


Fig. II–8

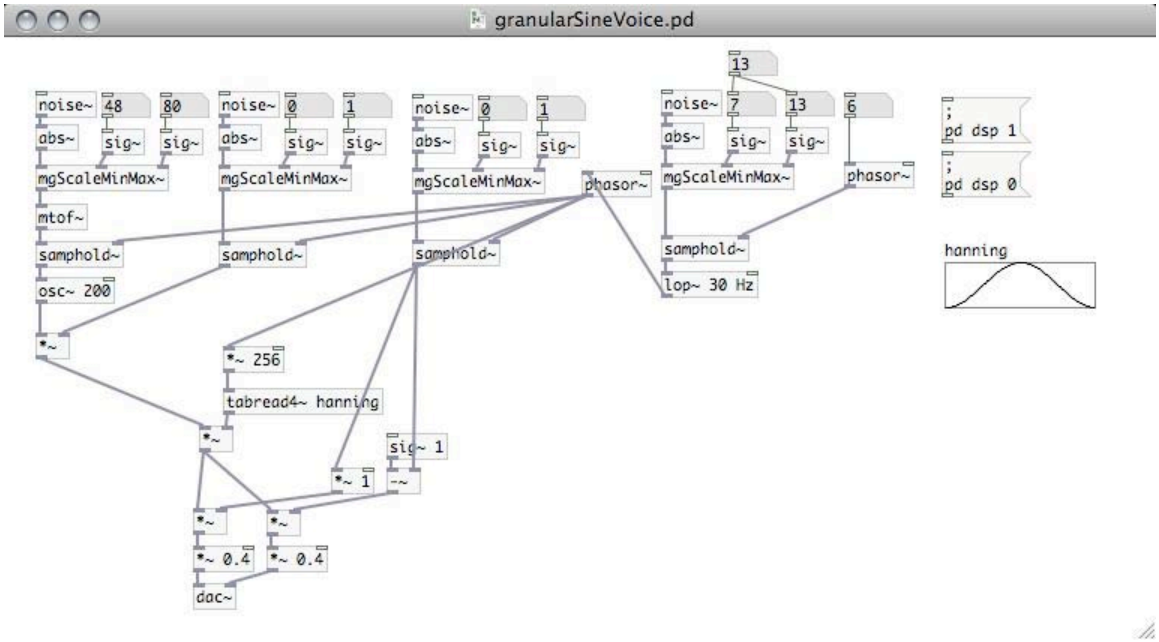
A book of screens equals the life of a complex sound

- What were some parameters that Roads employs in his implementation?

- What are some other applications of granular synthesis?
- What are the visual or animation analogues of granular synthesis?
- Is granular synthesis algorithmic composition?

14.5. Simple Sine Grains in PD

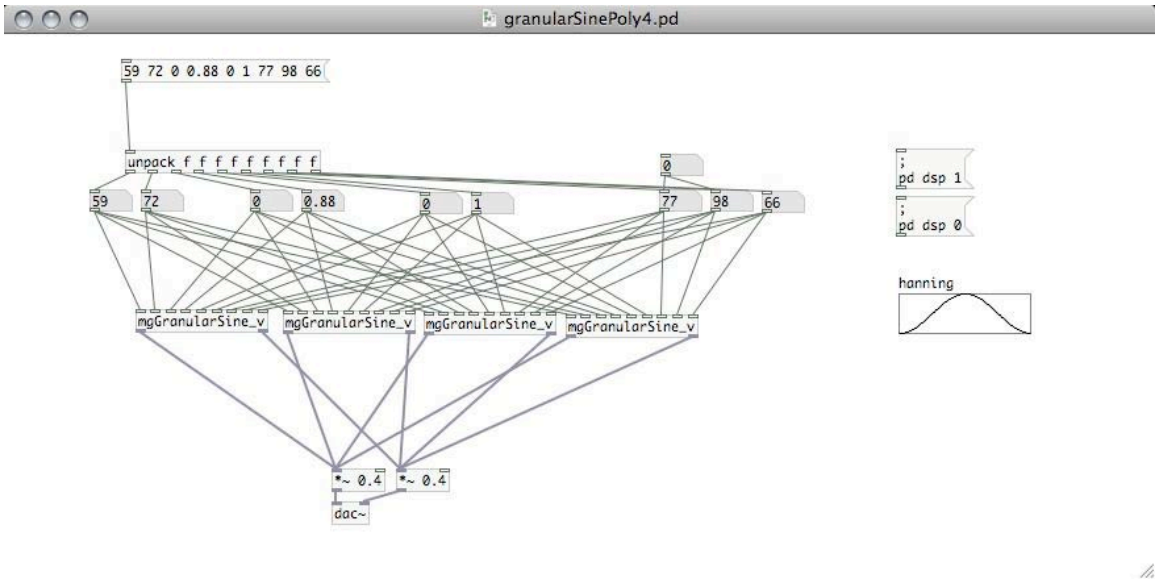
- Enveloped sine tines
- A [phasor~] is used to trigger multiple [samphold~] processes that grab parameter values once per event envelope
- Random parameter values are generated by [noise~] scaled between minimum and maximum values with [mgScaleMinMax~]
- Event envelopes are provided by the hanning array and read with [tabread4~]
- martingale/pd/demo/granularSineVoice.pd



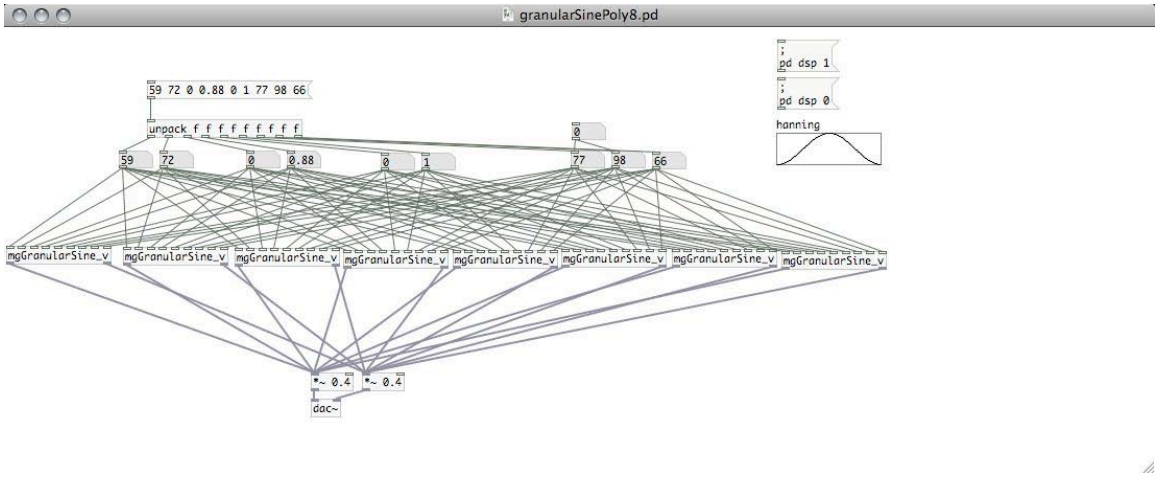
14.6. Polyphonic Sine Grains in PD

- Numerous instances of [mgGranularSine_v.pd] can be controlled together to produce multiple streams of grains
- Due to use of random parameter ranges, each voice will be independent

- martingale/pd/demo/granularSinePoly4.pd

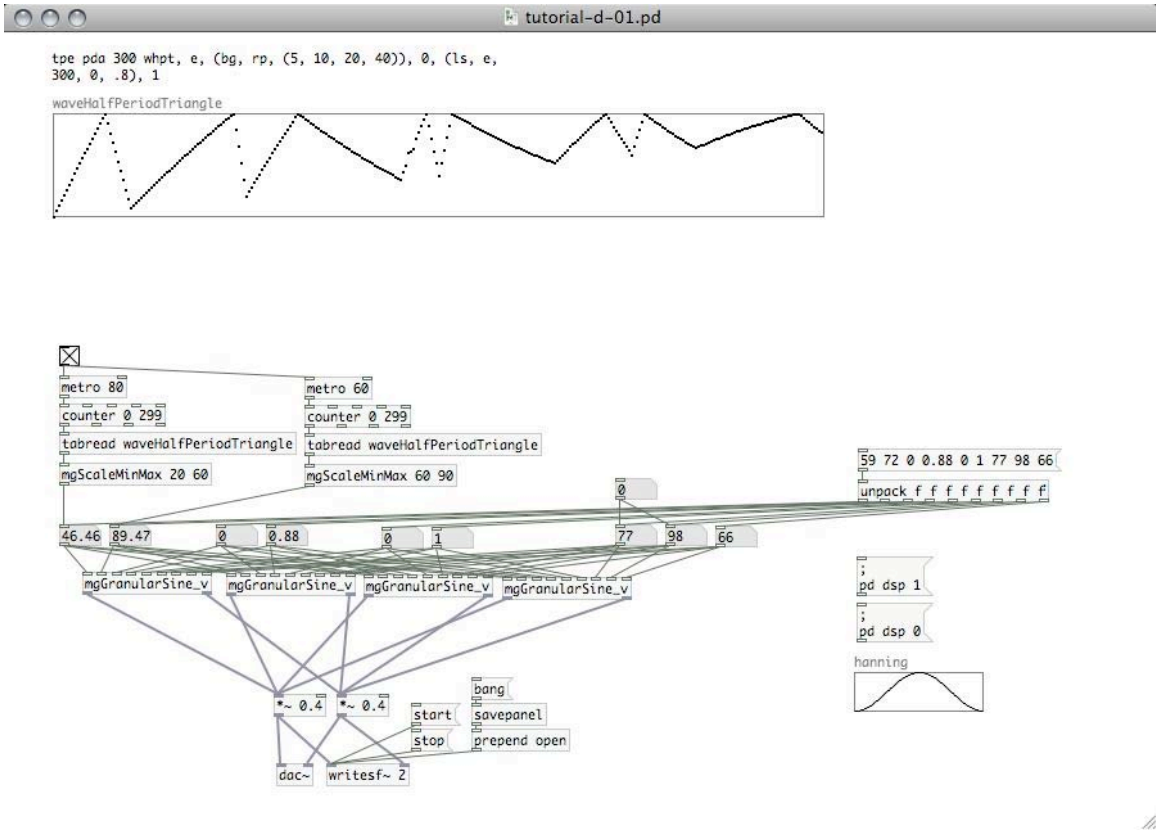


- martingale/pd/demo/granularSinePoly8.pd

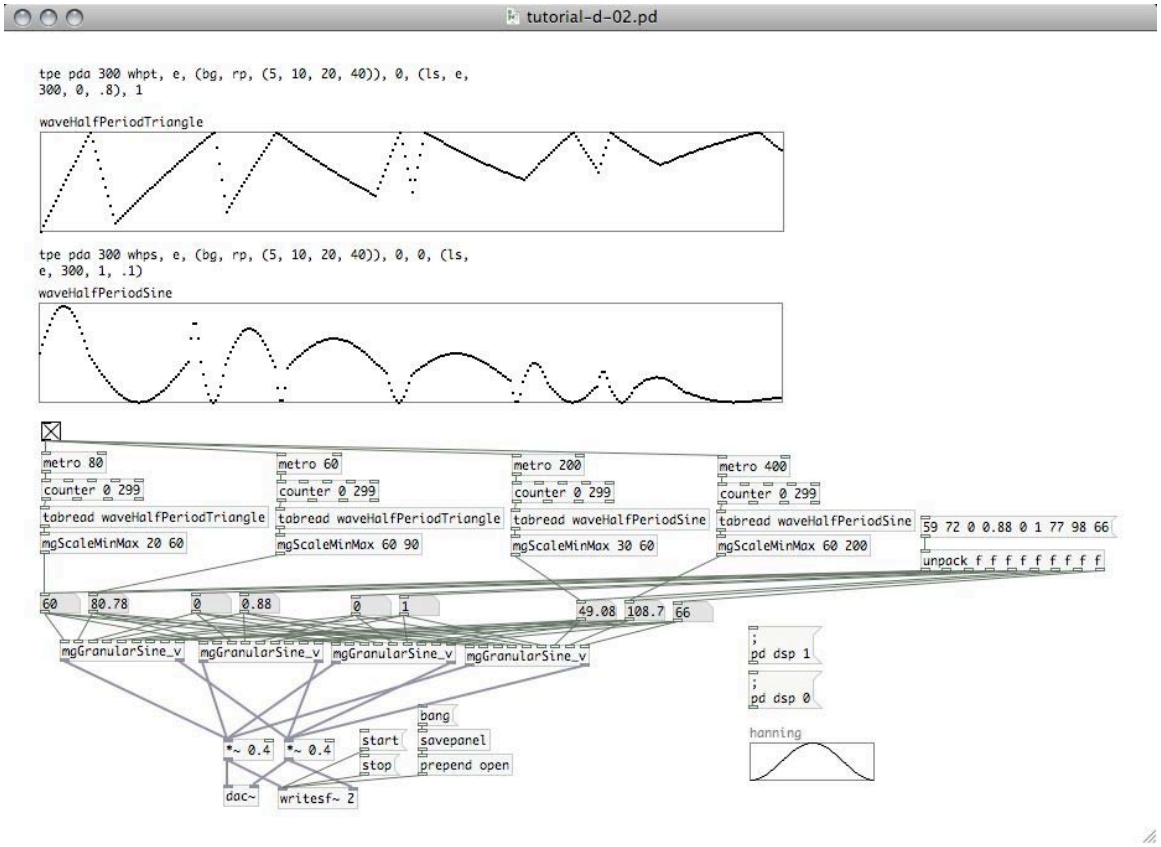


14.7. Large-Scale Parameter Behavior of Polyphonic Sine Grains in PD

- Use the TPe (TextureParameter Export) command with the PureDataArray format to create array structures
  - tpe pda 300 whpt,e,(bg,rp,(5,10,20,40)),0,(ls,e,300,0,.8),1
  - tpe pda 300 whps,e,(bg,rp,(5,10,20,40)),0,0,(ls,e,300,1,.1)
- Reading parameter values from multiple [tabread] at different rates



- Reading parameter values from multiple [tabread] and multiple tables at different rates



14.8. Polyphonic Sine Grains in athenaCL: LineGroove

- Can approach granular synthesis by using extremely small durations and/or fast tempi
- Command sequence using TM LineGroove:
  - emo cn
  - tmo LineGroove
  - tin a 4
  - *set a event time between 60 and 120 ms*

    tie r cs,(ru,.060,.120)
  - *smooth envelope shapes*

    tie x0 c,.1; tie x1 c,.5
  - *set field with a tendency mask converging on a single pitch after 15 seconds*

    tie f ru,(ls,t,15,-24,0),(ls,t,15,24,0)

- *set random panning*

  tie n ru,0,1

- *create a few more instances*

  ticp a b c d e f

- eln; elr; elh

14.9. Polyphonic Sine Grains in athenaCL: DroneArticulate

- TM DroneArticulate realizes each component of the path as a separate lines, writing an independent voice for each pitch one at a time for the entire duration
- Command sequence using TM DroneArticulate:
  - emo cn
  - tmo DroneArticulate
  - *a very large pitch collection made from a Xenakis sieve*

    pin a 5@2|7@6,c1,c9

  - tin a 4
  - *set a event time between 60 and 120 ms*

    tie r cs,(ru,.060,.120)

  - *smooth envelope shapes*

    tie x0 c,.1; tie x1 c,.5

  - *set random panning*

    tie n ru,0,1

  - *reduce amplitudes*

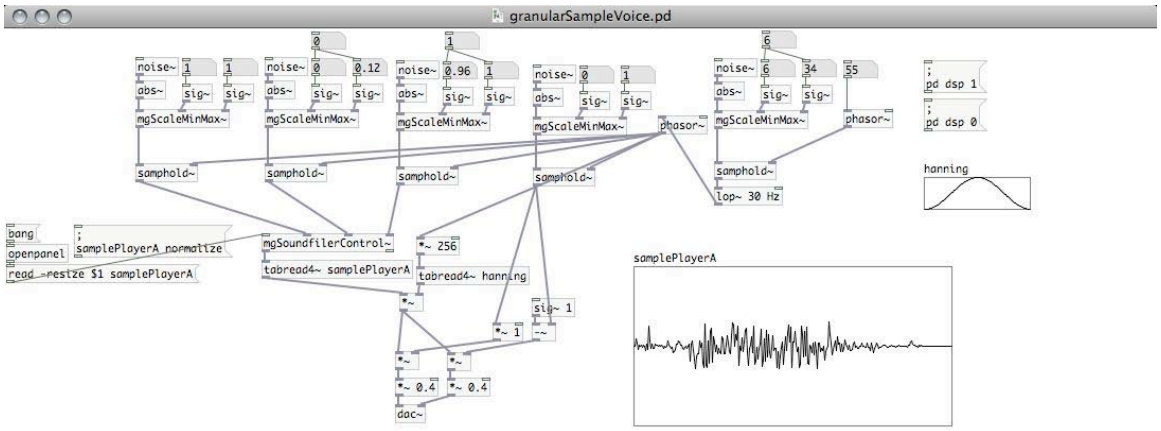
    tie a ru,.6,.8

  - eln; elr; elh

14.10. Simple Sample Grains in PD

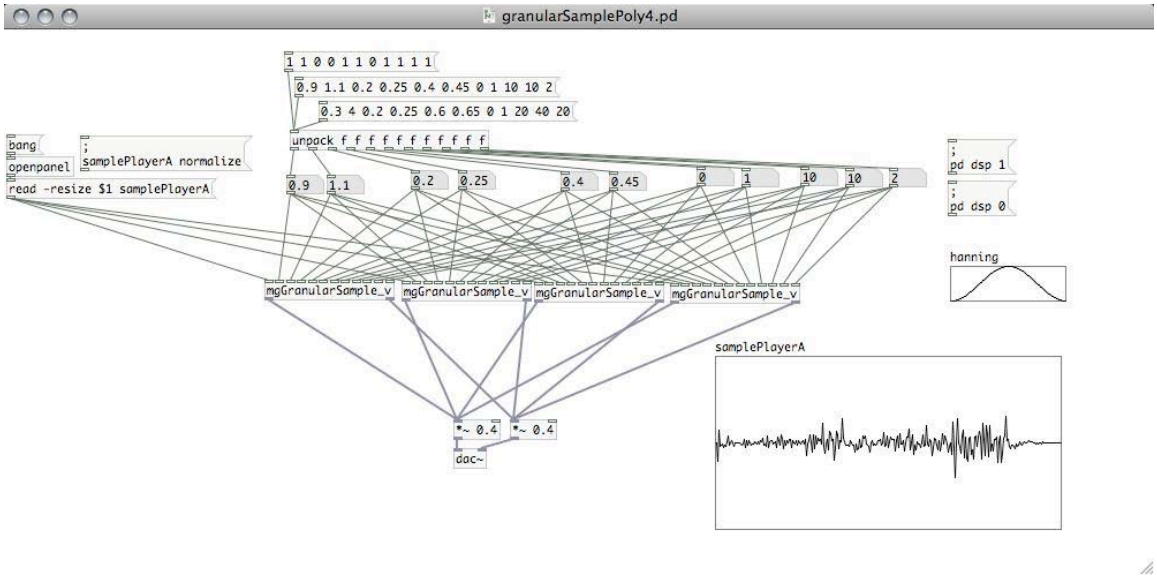
- Enveloped sampled audio files

- Press [bang] to trigger [openpanel] to select an audio file
- Parameters: playback speed min/max, start time min/max (within unit interval), end time min/max, pan min/max, phasor fq min/max, control phasor
- martingale/pd/demo/granularSampleVoice.pd



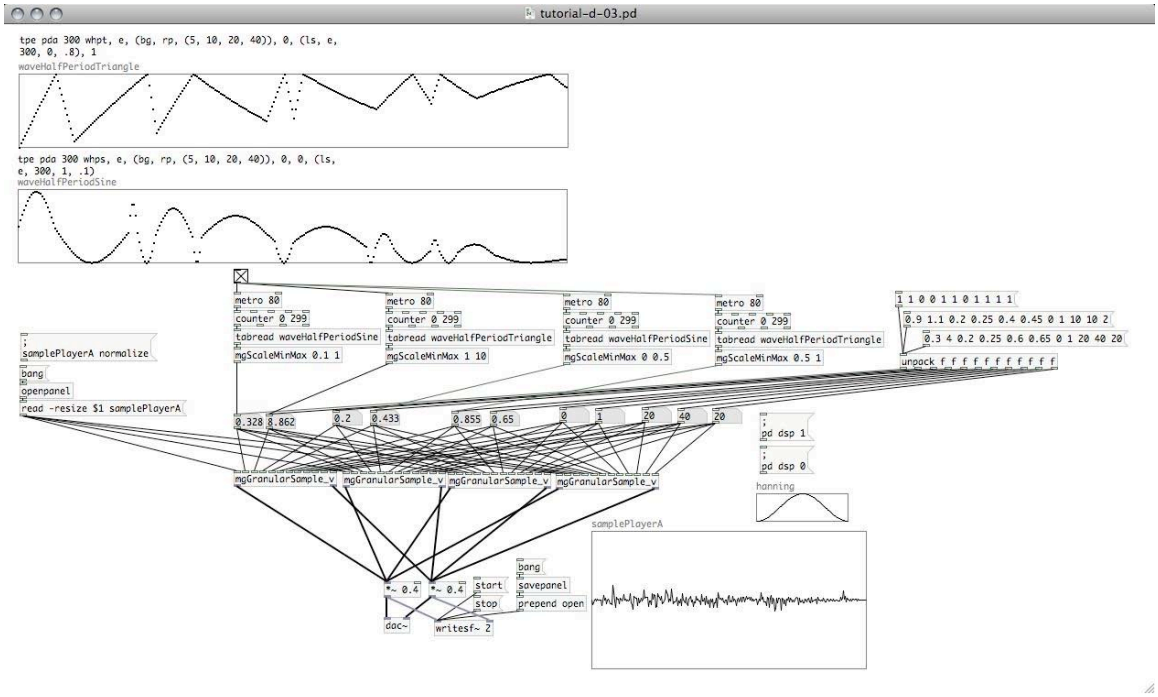
14.11. Polyphonic Sample Grains in PD

- Numerous instances of [mgGranularSample_v.pd] can be controlled together to produce multiple streams of grains
- martingale/pd/demo/granularSamplePoly4.pd



14.12. Large-Scale Parameter Behavior of Polyphonic Sample Grains in PD

- Use the TPe (TextureParameter Export) command with the PureDataArray format to create array structures
  - tpe pda 300 whpt,e,(bg,rp,(5,10,20,40)),0,(ls,e,300,0,.8),1
  - tpe pda 300 whps,e,(bg,rp,(5,10,20,40)),0,0,(ls,e,300,1,.1)
- Reading parameter values from multiple [tabread] and multiple tables at different rates



14.13. Polyphonic Sample Grains in athenaCL from a Single Audio File: LineGroove

- Read segments from an audio file by specifying the audio file (with the ConstantFile PO) and a start time
- Command sequence:
  - emo cn
  - tmo LineGroove
  - *instrument 32 is a fixed playback rate sample player*

tin a 32

- *set a file path to an audio file*

  tie x6 cf,/Volumes/xdisc/_sync/_x/src/martingale/martingale/audio/32673.aif

- *set a event time between 60 and 120 ms*

  tie r cs,(ru,.060,.120)

- *smooth envelope shapes*

  tie x0 c,.01; tie x1 c,.5

- *start position within audio file in seconds*

  tie x5 ru,0,10

- *set random panning*

  tie n ru,0,1

- *create a few more instances*

  ticp a b c d e f

- eln; elr; elh

14.14. Polyphonic Sample Grains in athenaCL from a Multiple Audio Files: LineGroove

- Read segments from an audio file by specifying the audio file (with the DirectorySelect PO) and a start time

- Command sequence:

  - emo cn

  - tmo LineGroove

  - *instrument 32 is a fixed playback rate sample player*

    tin a 32

  - *set a file path to an directory, a file extension, and a selection method*

    tie x6 ds,/Volumes/xdisc/_sync/_x/src/martingale/martingale/audio,.aif,rp

  - *set a event time between 60 and 120 ms*

tie r cs,(ru,.060,.120)

- *smooth envelope shapes*

  tie x0 c,.01; tie x1 c,.5

- *start position within audio file in seconds*

  tie x5 ru,0,10

- *set random panning*

  tie n ru,0,1

- *control a variety of amplitudes*

  tie a ru,.2,.4

- *create a few more instances*

  ticp a b c

- eln; elr; elh

14.15. Polyphonic Sample Grains in athenaCL from Multiple Audio Files: TimeFill

- Use TimeFill to create dynamic changes in the density of sampled files
- Command sequence:
  - emo cn
  - tmo TimeFill
  - *instrument 32 is a fixed playback rate sample player*

    tin a 32

  - *set a file path to an directory, a file extension, and a selection method*

    tie x6 ds,/Volumes/xdisc/_sync/_x/src/martingale/martingale/audio,.aif,rp

  - *set a event time between 60 and 120 ms*

    tie r cs,(ru,.030,.090)

  - *smooth envelope shapes*

tie x0 c,.01; tie x1 c,.5

- *start position within audio file in seconds*

  tie x5 ru,0,10

- *set random panning*

  tie n ru,0,1

- *control a variety of amplitudes*

  tie a ru,.1,.2

- *set number of events*

  tie s3 1000

- *start position within texture normalized within unit interval*

  tie d0 rb,.3,.3,0,1

- eln; elr; elh

14.16. Reading: Sturm, Adaptive Concatenative Sound Synthesis

- Sturm, B. L. 2006. "Adaptive Concatenative Sound Synthesis and Its Application to Micromontage Composition." *Computer Music Journal* 30(4): 46-66.

- Sound examples

  http://www.mat.ucsb.edu/~b.sturm/CMJ2006/MATConcat.html

- What are some practical applications of concatenative sound synthesis?

- How is adaptive concatenative sound synthesis a type of analysis and resynthesis, similar to Markov analysis and generation?

- What are some common sonic features used to select source audio?

- Is concatenative sound synthesis algorithmic composition?

21M.380 Music and Technology: Algorithmic and Generative Music
Spring 2010