POLITECNICO
MILANO 1863

SASP Project: A simple Granular synth

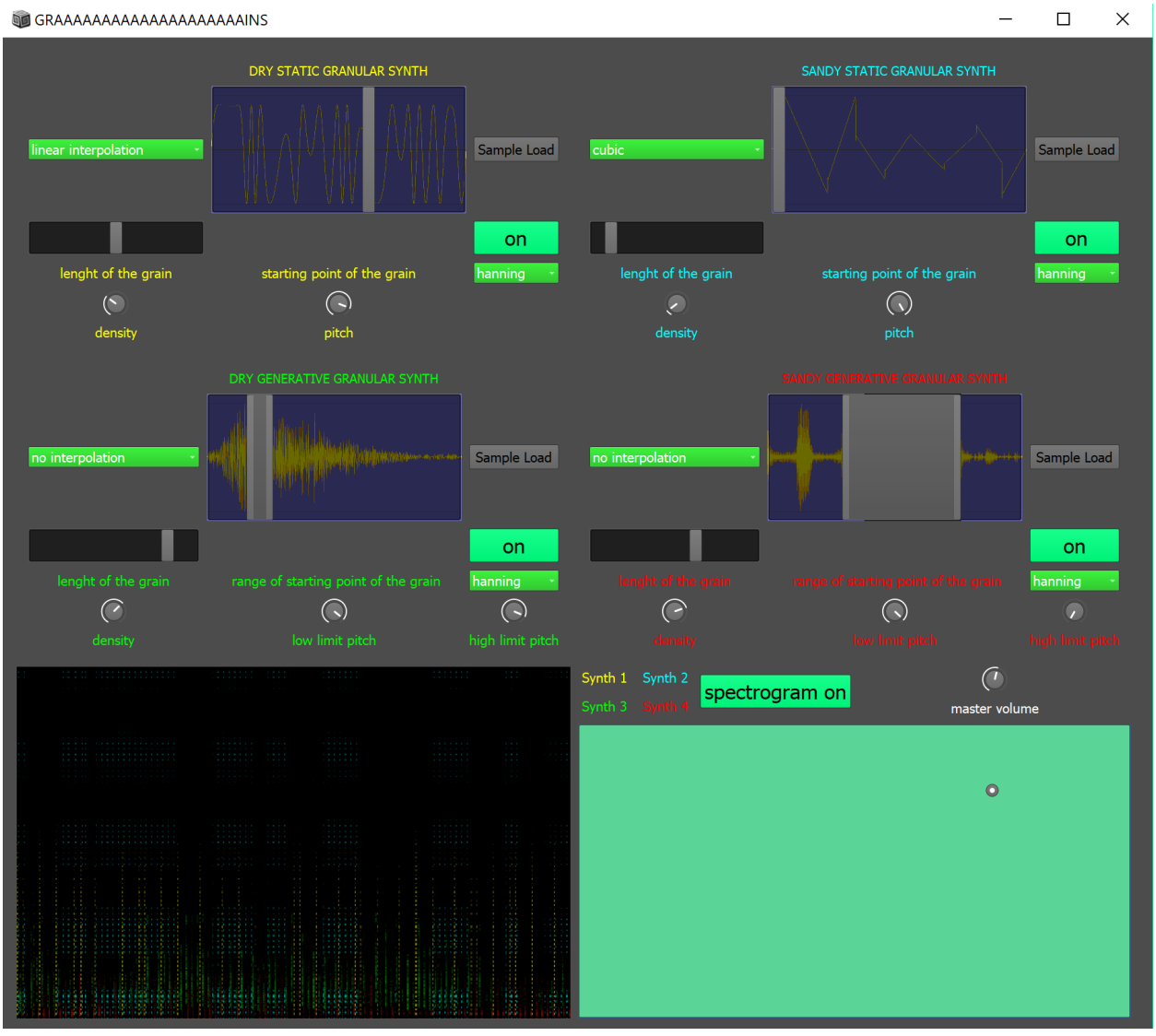
COMPUTER SCIENCE AND ENGINEERING - MUSICAL ACOUSTIC

Andrea Molgora

2017/2018

Introduction

This project is created with the intention of providing to new users/students a repsonsive, easy to use GUI with intuitive parameters, leading on a better understanding of the granular synthesis technique.
The synth has 4 voices, which means there are 4 independent granulizers at work in parallel, each with its peculiar characteristics which will be analyzed later. So every voice has its local parameters: the only global parameters are a master volume and a mixer implemented in a 2-dimensional slider with color feedback.
There is also implemented via FFT analysis a modified spectrogram which can be turned on/off and can show how the grains are positioned in a time/frequency graph, with colors corresponding to voices in such a way to better understand how each voice affect various frequencies.



Synth 1

Listing shows the SuperCollider script for the first synth.

```
SynthDef(\granularDry,{

    arg sndbuf, trig=10, dur=0.05, startRate=0.5, startPos=0.0, freqPos=0.1,
       interp=1, freqPan=3, gate=1, envbufnum=-1, amp=1, master =1;

    var sig, env;

    sig = GrainBuf.ar(2, //stereo signal
         trigger: Impulse.kr(trig),
         dur: dur,
         sndbuf: sndbuf,
         rate: startRate,
         pos:  startPos,
         interp: interp,
         envbufnum: envbufnum,
         pan: LFNoise1.kr(freqPan));

    env = EnvGen.kr(Env.asr, gate, doneAction:2);
    sig = sig*env;
    Out.ar(0,sig*amp*master);
    Out.ar(10,sig)
})
```

This synth is a typical synchronous granular synth with static parameters.
It's the simplest voice of all fours, since it receives a static duration of the grain between 5ms and 100ms, a static starting point of the grain, and a static playback rate which defines the pitch. Moreover, through the Impulse UGen we set a trigger for the grains with a static time-step. It is also possible to set an interpolation for the pitch shifting and choosing an envelope function for the grain between Hanning, square or triangular window.
The output level is controlled by the mixer and the master volume knob, and an additional audio bus is used to perform the FFT analysis. The output is panned left and right through a LFO which randomly returns a value to update the panning.

Synth 2

Listing shows the SuperCollider script for the second synth.

```
SynthDef(\granularSandy,{

    arg sndbuf, trig=10, dur=0.05, endRate=2, startPos=0.0, freqPos=0.1,
       interp=1, freqPan=3, gate=1, envbufnum=-1, amp=1, master =1;

    var sig, env;

    sig = GrainBuf.ar(2,
         trigger: Blip.kr(trig),
         dur: dur,
         sndbuf: sndbuf,
         rate: startRate,
         pos:  startPos,
         interp: interp,
         envbufnum: envbufnum,
         pan: LFNoise1.kr(freqPan));

    env = EnvGen.kr(Env.asr, gate, doneAction:2);
    sig = sig*env;
    Out.ar(0,sig*amp*master);
    Out.ar(12,sig)
})
```

The second synth is an asynchronous granular synth with static parameters.
It's similar to the first one, as it too receives a static duration of the grain, a static starting point of the grain, and a static playback rate which defines the pitch. However, the trigger for the grains is the Blip UGen, which brings us to a more random density distributions of the grains. Like the first one, it is also possible to set an interpolation for the pitch shifting and choosing an envelope function for the grain between Hanning, square or triangular window.
The output level is controlled by the mixer and the master volume knob, and an additional audio bus is used to perform the FFT analysis.

Synth 3

Listing shows the SuperCollider script for the third synth.

```
SynthDef(\granularDryLFO,{

    arg sndbuf, trig=10, dur=0.05, startRate=0.5, endRate=2, startPos=0.0,
       endPos=1.0, freqPos=0.1, interp=1, freqPan=3, gate=1, envbufnum=-1,
       amp=1, master =1;

    var sig, env;

    sig = GrainBuf.ar(2,
         trigger: Impulse.kr(trig),
         dur: dur,
         sndbuf: sndbuf,
         rate: LFNoise1.kr.range(startRate,endRate),
         pos:  LFNoise2.kr(freqPos).range(startPos,endPos),
         interp: interp,
         envbufnum: envbufnum,
         pan: LFNoise1.kr(freqPan));

    env = EnvGen.kr(Env.asr, gate, doneAction:2);
    sig = sig*env;
    Out.ar(0,sig*amp*master);
    Out.ar(14,sig)
})
```

The third synth is a synchronous granular synth with dynamic parameters.
It only receives a static duration of the grain. A dynamic starting point of the grain and a dynamic playback rate are chosen between two endpoints defined by the user, and the value is polled respectively by the LFO random UGens LFNoise1 and LFNoise2. The trigger for the grains is the Impulse UGen, giving us a static density distributions of the grains. Like the first one, it is also possible to set an interpolation for the pitch shifting and choosing an envelope function for the grain between Hanning, square or triangular window.
The output level is controlled by the mixer and the master volume knob, and an additional audio bus is used to perform the FFT analysis.

Synth 4

Listing shows the SuperCollider script for the fourth synth.

```
SynthDef(\granularSandyLFO,{

    arg sndbuf, trig=10, dur=0.05, startRate=0.5, endRate=2, startPos=0.0,
       endPos=1.0, freqPos=0.1, interp=1, freqPan=3, gate=1, envbufnum=-1,
       amp=1, master =1;

    var sig, env;

    sig = GrainBuf.ar(2,
         trigger: Blip.kr(trig),
         dur: dur,
         sndbuf: sndbuf,
         rate: LFNoise1.kr.range(startRate,endRate),
         pos:  LFNoise2.kr(freqPos).range(startPos,endPos),
         interp: interp,
         envbufnum: envbufnum,
         pan: LFNoise1.kr(freqPan));

    env = EnvGen.kr(Env.asr, gate, doneAction:2);
    sig = sig*env;
    Out.ar(0,sig*amp*master);
    Out.ar(16,sig)
})
```

The fourth synth is an asynchronous granular synth with dynamic parameters.
Like the third one, it only receives a static duration of the grain. A dynamic starting point of the grain and a dynamic playback rate are chosen between two endpoints defined by the user, and the value is polled respectively by the LFO random UGens LFNoise1 and LFNoise2. The trigger for the grains is the Blip UGen, giving us a non-static density distributions of the grains. Like the first one, it is also possible to set an interpolation for the pitch shifting and choosing an envelope function for the grain between Hanning, square or triangular window.
The output level is controlled by the mixer and the master volume knob, and an additional audio bus is used to perform the FFT analysis.