

Audio Spray Gun 0.8.1

Richard Garrett

Contents

Contents..... 1

Introduction 2

System requirements..... 2

Getting Started 3

The ASG interface 4

The Parameter Window 5

A Function Generator 6

Control Window 7

Write Window 8

Display Window..... 9

Acknowledgements 9

Contact..... 9

WARNING

Audio Spray Gun can play very large numbers of samples simultaneously, resulting in high sound pressures. Please protect your ears and your speakers by experimenting at low playback levels.

Introduction

Audio Spray Gun (ASG) is an experimental tool for fixed-media composition that simultaneously generates and spatialises large groups of sound-events from a single sound source. This is achieved by randomly creating points constrained by a locus or ‘particle zone’ within a multi-dimensional parameter space. When this locus is transformed over time, the program produces a sequence of events, which can then be rendered to multichannel audio.

While earlier versions of the program were designed to render audio direct to eight-channel surround, version 0.8 has been extended to include three-dimensional spatialisation and the option to store sound-groups as metadata for later modification. For more information on the theory behind this, see my forthcoming paper in the proceedings of ICMC 2015 (UNT, Texas, September 2015).

System requirements

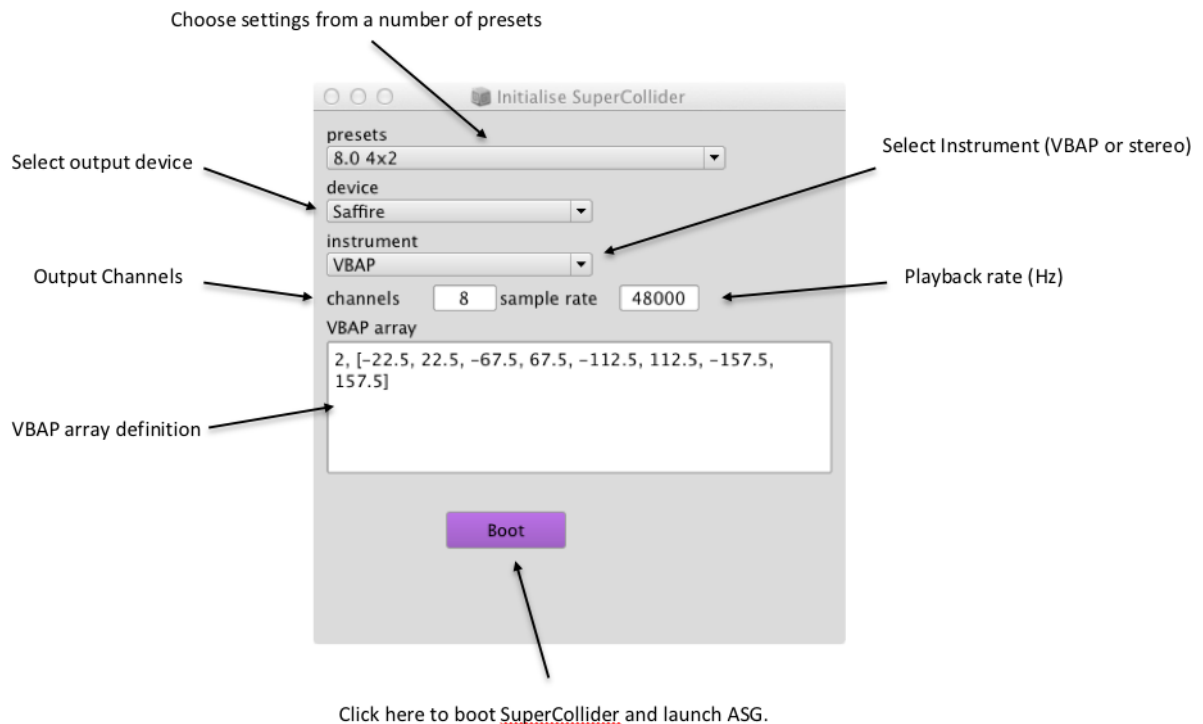
Audio Spray Gun runs in SuperCollider 3.6.3 (SC) and has been tested on an Apple MacBook Pro running OSX 10.7.5. It has not been tested on PC or Linux systems.

ASG also needs Scott Wilson’s VBAP UGens for SuperCollider. These are included in the **sc3-plugins** extension package, downloadable from <https://github.com/supercollider/sc3-plugins>

At present, ASG will not record multichannel interleaved audio files to disc on later versions of SC. This is due to a SC issue that is expected to be resolved in version 3.7.

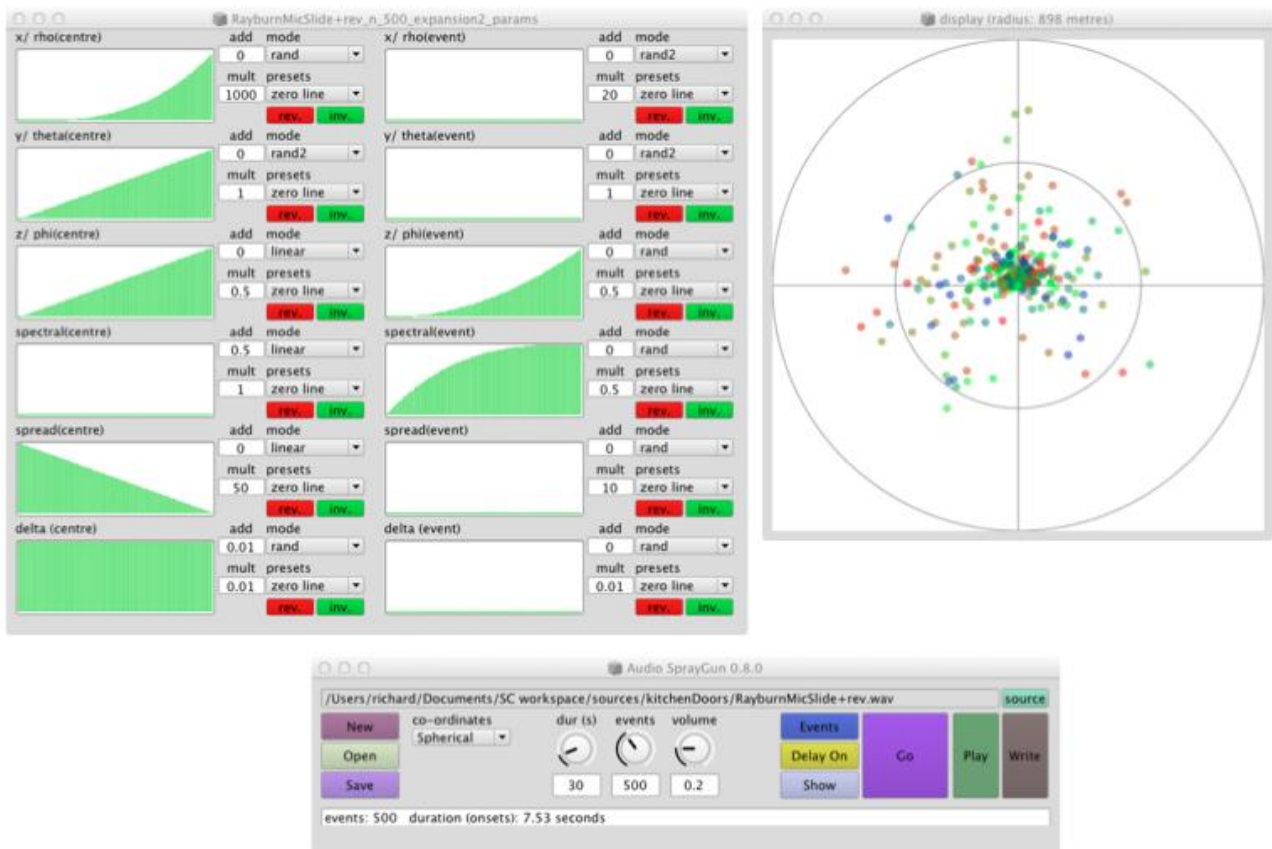
Getting Started

Unpack the .zip file to a convenient location on your hard disc then load the file *ASG081_init.scd* into SuperCollider. This will appear as a text in the IDE. Launch the program by holding down Cmd and clicking the mouse anywhere in the main body of the program. This should launch the initialisation window.



Use these options select output hardware, number of channels, sample rate of the output and so on. There are a number of presets to help you including stereo, 4.0, 5.0, 8.0 (circular) and 8.0 as four stereo pairs. When you have selected the values you want, click on the “Boot” button. This will boot the SC server and launch ASG.

The ASG interface



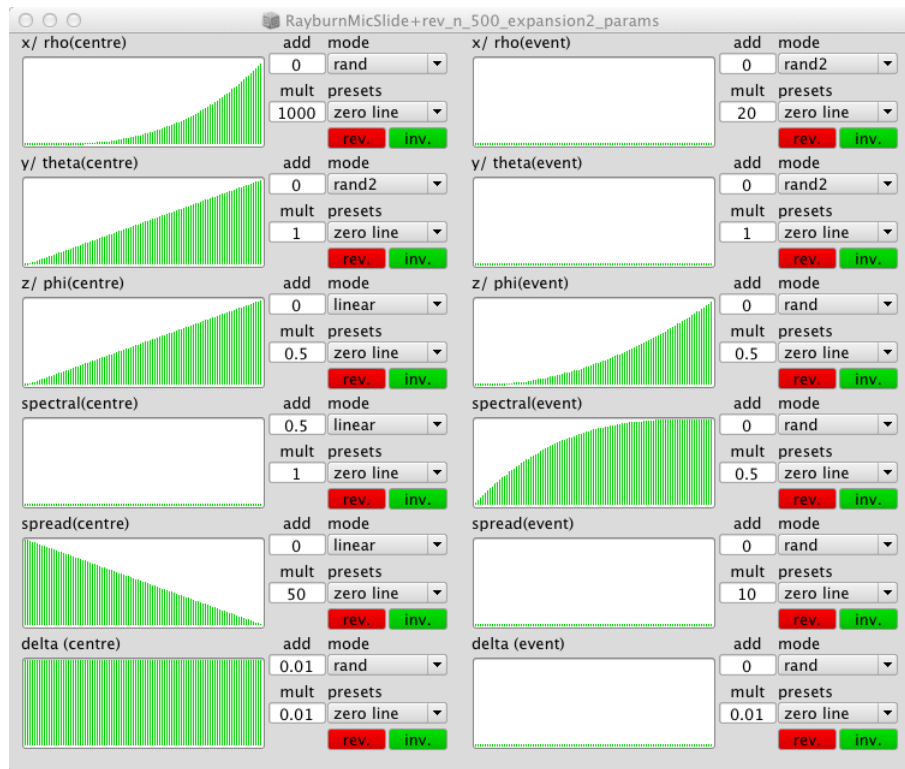
The ASG interface consists of three windows:

the *parameter window* containing twelve function generators that define the sound group;

the *control window* which generates and plays the groups, loads and stores files and so on; and

the *display window* that shows an animation of the group as it evolves.

The Parameter Window

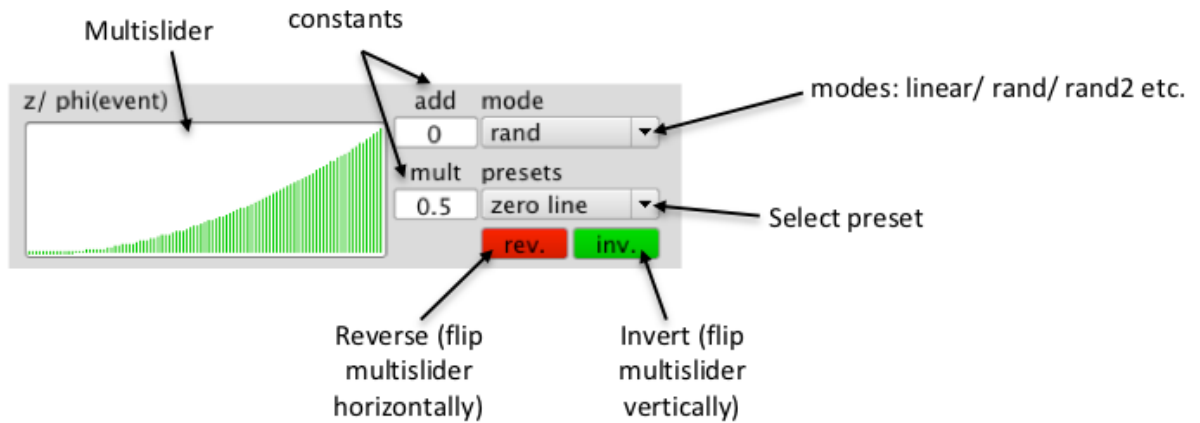


Sound-groups in ASG are generated by the creation of events at random positions within a changing locus in a six-dimensional parameter space. Changes in the position and extent of the locus over the lifetime of the group are governed by the twelve function generators shown in this window. The parameters controlled are as follows:

ρ_{centre}	perceived distance in metres of the locus centre from the listener.	ρ_{event}	distance of the sound event from the locus centre.
θ_{centre}	azimuth of the locus centre with respect to the listener (-1 to +1).	θ_{event}	azimuth of the event w.r.t the locus centre. (-1 to +1).
ϕ_{centre}	elevation of the locus centre w.r.t. the listener (0 to +1).	ϕ_{event}	elevation of the event w.r.t. locus centre (0 to +1).
$\text{spectral}_{\text{centre}}$	resampling value (playback rate) applied to events occurring at the locus centre.	$\text{spectral}_{\text{event}}$	resampling value of the event relative to that of the locus centre.
$\text{spread}_{\text{centre}}$	image spread value at locus centre (VBAP).	$\text{spread}_{\text{event}}$	image spread value of the event relative to that of the locus centre (VBAP).
δ_{centre}	the time interval before the next event (in seconds) when this event occurs at locus centre.	δ_{event}	Added to the time interval before the next event (in seconds) according to event location w.r.t. locus centre.

In this table, spatial values are expressed as spherical co-ordinates (ρ , θ , ϕ) but it is also possible to express them in cylindrical (ρ , θ , z) or Cartesian (x , y , z) co-ordinates.

A Function Generator



Functions generated by the system may be dependant either upon time (as a fraction of the group duration) or upon the number of events played so far (as a fraction of the whole group), according to the setting on the Mode button. Here is the general format of these functions:

$$value(x) = add + f(mult \times env(x))$$

where *add* and *mult* are constants (defined by the two number boxes) and *env(x)* is the height of the curve or envelope shown in the multislider at point *x* (time or number according to mode).

The function *f* can be set to a number of modes using the mode menu e.g.:

linear: $add + (mult \times env(x))$

rand: $add + random(mult \times env(x))$

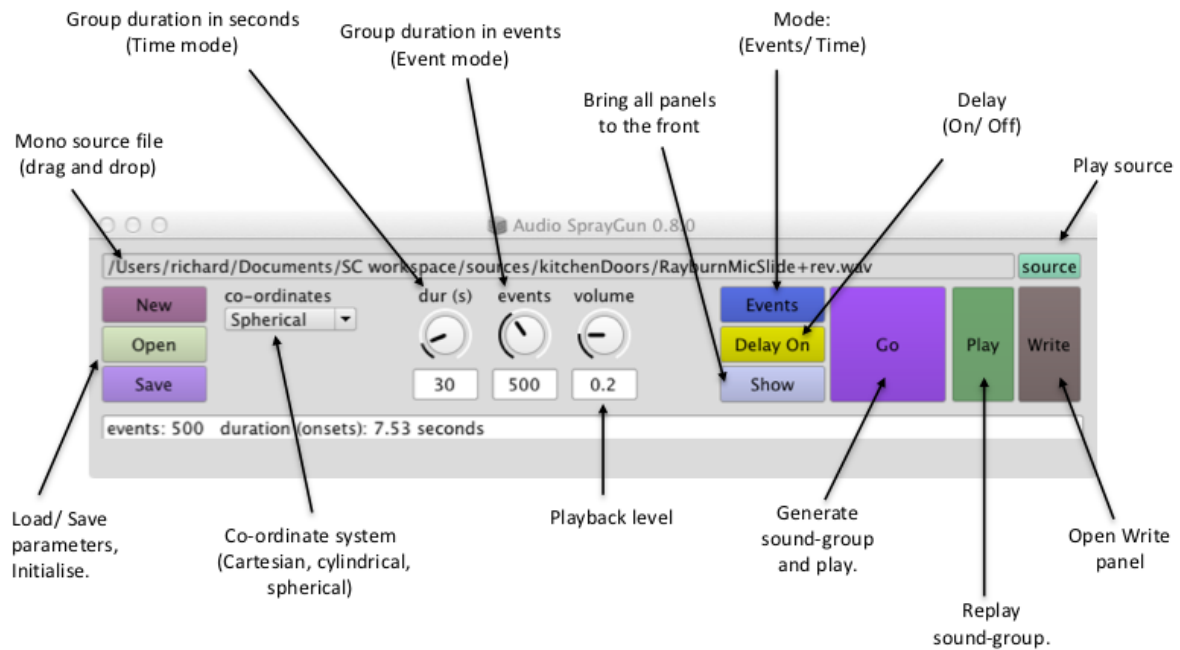
rand2: $add \pm random(mult \times env(x))$

and so on.

The spatial location of the sound is arrived at by first calculating the position of the locus centre (ρ_{centre} , θ_{centre} , ϕ_{centre}) with respect to the listener; then calculating the location of the event with respect to the locus centre (ρ_{event} , θ_{event} , ϕ_{event}) and, from these values, calculating the event location with respect to the listener. The other components are independent of these three and are simply expressed as $\text{spectral}_{\text{centre}} + \text{spectral}_{\text{event}}$, $\text{spread}_{\text{centre}} + \text{spread}_{\text{event}}$ and $\delta_{\text{centre}} + \delta_{\text{event}}$ respectively.

In many cases, the 'centre' functions will be linear and describe a trajectory through the space and the 'event' functions will use random functions to describe some sort of locus expansion or contraction.

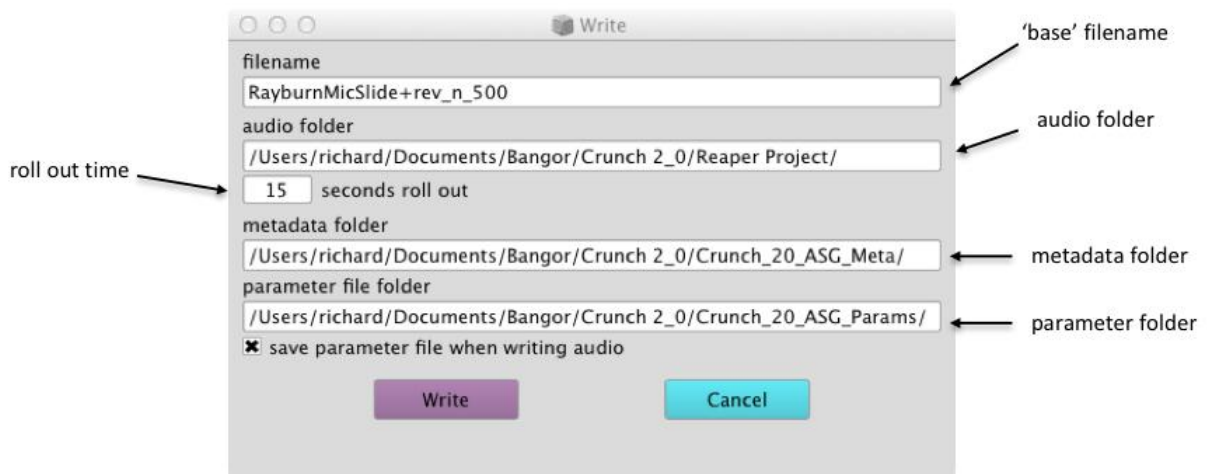
Control Window



The *control window* holds the following controls for program execution and data storage:

Control	Options	Function
(drop box)		Drop source file here.
Source		Press to audition the source file.
New		Set all function generators to their default values
Open		Load a SC archive file containing all parameters for a sound-group definition
Save		Write current sound-group definition to a text file
Co-ordinates	Cartesian/ Cylindrical/ Spherical	Choose co-ordinate system for spatial parameters
Dur (s)		Sets group length as a duration in seconds (sum of delta times) when in Time mode.
Events		Sets group length as a number of events when in Events mode.
Volume		Applies this gain level to each sound-event.
Mode	Time/Events	produces a group of a given duration or with a set number of events.
Delay	Off/ On	When switch on, this applies a delay to each sound event proportional to the distance of the event from the listener. The delay is equivalent to the speed of sound in air of 340 ms^{-1} .
Go		Performs all the calculations to create a sound group from the current parameter set, stores the data as an array and then plays it back. Each time this button is pressed a new group is created.
Play		Replay current group
Write		Open the Write window to save audio, metadata, parameter files.

Write Window



The Write window contains a text field labelled “filename” and three drag and drop fields to define the folders into which files will be written.

When you open the Write window, a ‘base’ filename is generated to describe the sound group. The general format of this name is *<source name>_<mode (n/t)>_<number of events/ duration>*. So, in the example shown, the audio source is *RayburnMicSlide+rev.wav*, the system is in Events mode (signified by the letter *n*) and the number of events is 500. The base filename in this case will be *RayburnMicSlide+rev_n_500*.

When the Write button is pressed two (or three) files are written to disc:

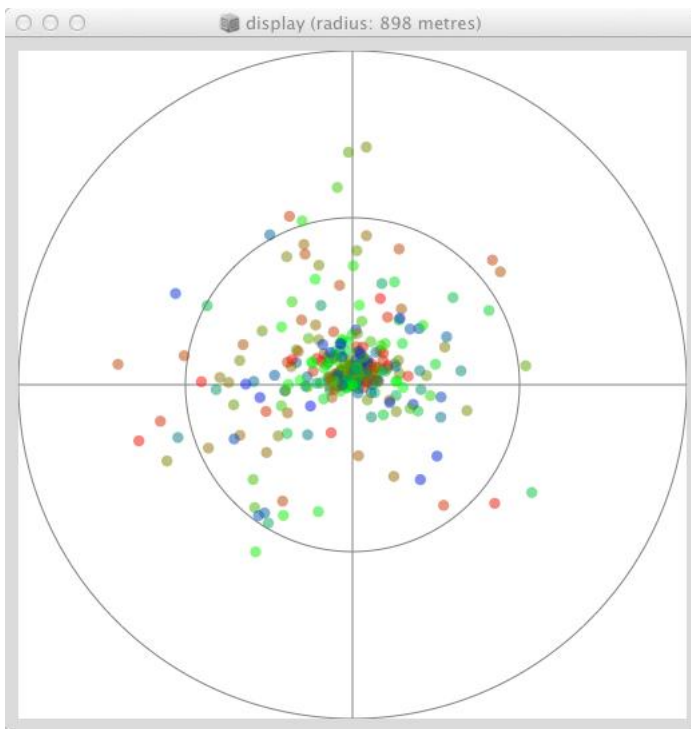
an interleaved multichannel audio file (e.g. *RayburnMicSlide+rev_n_500.aif*) will be stored in the audio folder;

a metadata file containing a list of all the events as points in the parameter space will be stored to the metadata folder in SC archive format as *RayburnMicSlide+rev_n_500_meta*;

and (optionally) the parameters used to generate the group will be saved to the parameter file folder as an SC archive called *RayburnMicSlide+rev_n_500_params*.

The base filename can be edited to suit your requirements.

Display Window



When the Go button is pressed, the *display window* produces an animation, which shows the spatial locations of each sound-event in the group as if looking down on listener at the centre of the circle. Each event is shown as a dot whose colour reflects its resampling co-ordinate using the colours of the rainbow such that red is a low resampling rate and blue is high.

Acknowledgements

Audio Spray Gun is under development as part of my PhD study at Bangor University, Wales with funding from the Arts and Humanities Research Council of the United Kingdom.

Contact

For feedback or advice, please contact richard@sundaydance.co.uk

Richard Garrett, Bangor University UK

Thursday, 10 September 15