Version 0.9beta

Scanned Generators

Intro

Scanned synthesis involves a slow dynamic system whose frequencies of vibration are below about 15 hz. The system is directly manipulated by motions of the performer. The vibrations of the system are a function of the initial conditions, the forces applied by the performer, and the dynamics of the system. Examples include slowly vibrating strings, two dimensional surfaces obeying the wave equation, and a waterbed. We have simulated the string and surface models on a computer. Our waterbed model is purely conceptual.

The Scanned Generators for Blue consists of 4 main parts:
1. Scanned Matrix Generator
2. Scanned Matrix Plot
3. Scanned Trajectory Generator
4. Scanned Trajectory Plot

The idea is to generate patterns for use with the Scanned Synthesis opcodes scanu(2) and scans of Csound.
Blue provide a way to run formulas in Csound code using buttons and sliders, creating a Matrices and a Trajectories. In addition, 2 Python scripts make it possible to create a visual representation of these matrices and trajectories.

Scanned Matrix Generator

Used to generate patterns for the Matrix.
Uses the .matrxT extension and format as produced by GEN44.

It starts counting at 1, but the real start value is up to the user. It ends generating values for Lines or Clusterts until matrix size or lower by setting the ‘matrix size’ slider.
In the example below the matrix size is set to 128.

A start value can be set and and end value. Start may be higher, end values can be set to lower values; in that case, sections of the Patterns will be generated.

Matrix size value is printed to the output file of the generated Matrix, for example: string_128.matrxT.
This provides the user the right information so that he/she set the correct value to other tables that are needed: for initial position, damping, masses, centering and displace (=velocity).

The Scanned Matrix Generator provides 3 different ways for generating the X, Y and W coordinates. W = the weight (see GEN44 information)

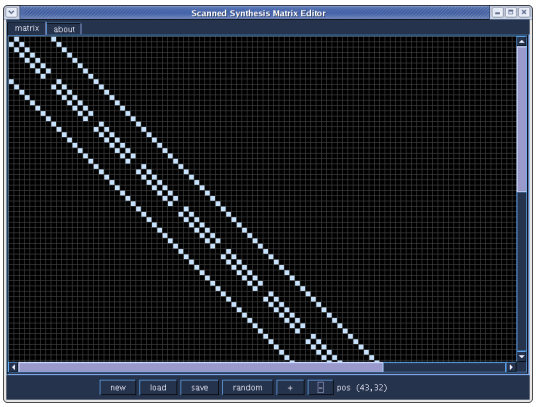
1. Lines & Clusters: Lines are uninterrupted lines (although the lines can contain equal spaces in between the nodes). Clusters are interrupted, or 'sections' of lines.
2. Random Pattern: a defined section of the Matrix can be populated with an adjustable number of nodes.
3. Waves: sines and other waveforms can be used to generate a continuous flow of nodes.

All these generator modules can be mixed to get really complicated matrices.

The 'traditional' Matrix with 128 node points (the .matrxB format) can be found as presets in the Preset list in the Instrument: string, stringcircular, grid, cylinder and torus. See tutorials: https://www.csounds.com/scanned. Here, they are converted 1:1 to the text format .matrxT.

Scanned Matrix Plot

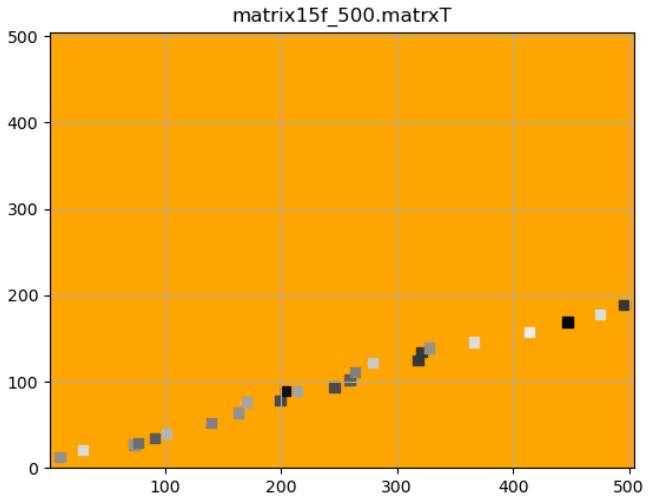
This Python script uses the Matlabplot Python library to plot the mass nodes of a Scanned Synthesis Matrix of type ‘text’.
The presentation is one comparable to the Matrix Editor from Steven Yi.



But there are differences:

- Steven’s Matrix starts at the top left. The Scanned Matrix Plot starts at the bottom left.
- Steven’s Matrix Editor loads and edits binary matrix files (.matrxB) only, the Scanned Matrix MatLabplot reads text matrix files (.matrxT) only.
- The Scanned Matrix Plot can not be edited.

The Blue file has all Presets matrices, like cylinder, string and torus. These .matrxT files should sound exactly like their older brothers in .matrxB format.



In this demonstration, looking at the name of the file: matrix15f_500.matrxT, you may conclude that a Matrix, named matrix15f, contains 500 points. The dots show their coordinates. The dots have a color range from white to black, representing the lowest and the highest weight of that mass.

Scanned Trajectory Generator

For the scans opcode, the table of a Trajectory does not need to have the same size as the other ftables used in scanu/scanu2. Size can be higher, equal or smaller. There can even be empty values (spaces).
A smaller size ftable makes a 'slice' of the masses to scan, and repeats the scan of that portion of the matrix nodes.

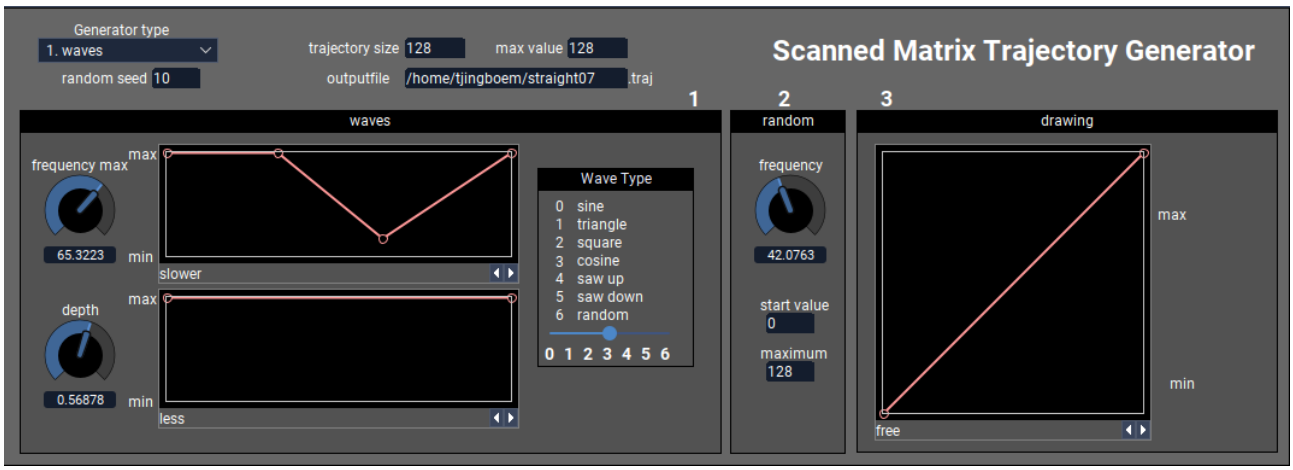
Example of an 'old' Trajectory format in action:
ifntraj ftgen 7, 0, 0, -23, "128-spiral-8,16,128,2,1over2.traj"

(Let Csound determine the size of the elements by using '0'. The negative value for GEN23 (-23) is to avoid normalization.)

The format of the Scanned Trajectory Generator is:
NAME+NUMBER_maxVALUE_NUMBER.traj where:
1. NAME+NUMBER : the name and version number to identify this trajectory
2. _maxNUMBER : the maximum value of the mass node you want to include
3. _NUMBER : length of the trajectory. May be higher, equal or lower than the number of mass nodes in the Matrix.

Example of the format as generated by the Scanned Trajectory Generator :
ifntraj ftgen 7, 0, 0, -23, "spiral01_max128_128.traj"

The value must be between 0 and the max or equal amount of nodes in the Matrix.



Scanned Trajectory Plot

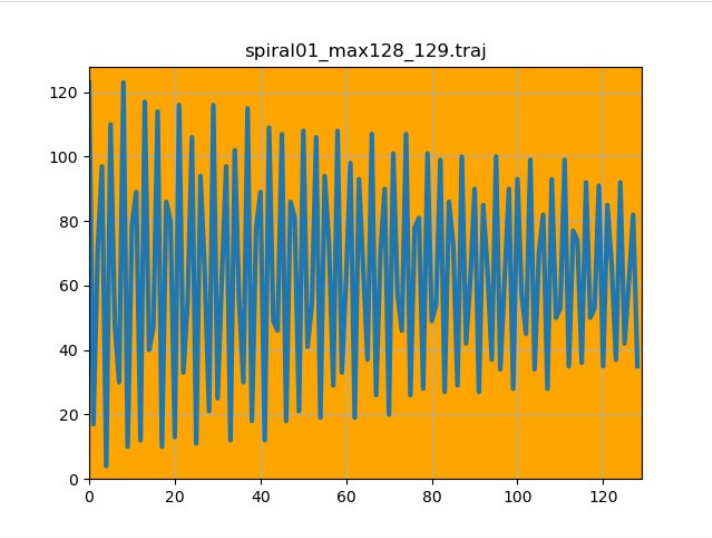
This Python script uses the Matlabplot Python library to plot a Scanned Synthesis Trajectory.
This is a trajectory connecting the value points, that shows how the mass nodes of the scanned synthesis matrix are going to be read.
The Trajectory has a great influence on the sonic result.

When the trajectory would be linear, like going from beginning to end, all mass nodes would be scanned in sequence like a straight line, scanning the Matrix from zero to end.

The Scanned Trajectory Generator is used to generate the Trajectory by the Waves, Random points or Drawing modules manually;
the Scanned Trajectory Plot shows the generated routes reading through the Matrix. It is a visual presentation of this route.

The format of the Trajectory MUST be: NAME+NUMBER_maxVALUE_NUMBER.traj where:
1. NAME+NUMBER : the name and version number to identify this trajectory
2. _maxNUMBER : the maximum value of the mass node you want to include
3. _NUMBER : length of the trajectory. May be higher, equal or lower than the number of mass nodes in the Matrix.

Trajectories that are NOT in this format can not be recognized and shown.



Final Thoughts

2 things are needed to set an existing table in motion: the matrix and the trajectory. Using different matrices results the table to sound in different harmonics. While different trajectories (reading back the table) result in a filter effect on these harmonics: when a Trajectory does not scan the matrix from 1 to matrix size completely, it will scan not all matrix nodes so some moving harmonics will not all be read or less strong. And a fast scan trajectory like the one you see above, produces a FM like distortion.