CURVEPAINTER: A NEW COMPOSITIONAL TOOL

Colby N. Leider
Department of Music, Princeton University
Princeton NJ  08544
http://music.princeton.edu/~colby
colby@music.princeton.edu

ABSTRACT
A new compositional tool for drawing control curves and function tables for software synthesis languages is described. The program, curvePainter, allows users to graphically create and modify curves using a variety of techniques. Furthermore, the program can generate curves from spectral features of sound files for use with other compositional programs.

BACKGROUND
With the advent of software synthesis languages came the notion of control functions or control curves—a paradigm that enabled composing within a continuum. Xenakis’ seminal book Formalized Music (1971) and other writings (e.g., 1986), the UPIC system (1977), the work of Estrada (e.g., Estrada 1994) and the gen functions of Music-N-style languages all illustrate applications of composing with control curves in a continuous space. Furthermore, recent programs like Cecelia (Piche and Burton 1998), CurveControl (Bencina 1995), and StochGran (Helmuth 1998) supplement the notion of graphical generation of control functions with robust interfaces to a target software synthesis environment.

MOTIVATION AND OVERVIEW
Although most software synthesis languages provide facilities for generating control functions—some of them graphically—many do not offer a robust means of generating, editing, and graphically viewing multiple control curves simultaneously for arbitrary parameters. curvePainter is a new graphical compositional tool that generates control functions and score files for use with software synthesis languages. curvePainter is not bound to any particular synthesis language, synthesis method, instrument, or platform.

The program does not produce sound itself; rather, it creates files of numbers for use in environments like SuperCollider, Max/MSP, cmix, OpenMusic, and Csound. It allows the user to create and view up to twenty-four independent vectors of floating-point numbers (“curves”) using a variety of techniques. Additionally, curves may be extracted from various characteristics of sound files, such as the spectral centroid, pitch, amplitude envelope, and spectral peaks. Graphical interfaces to the legacy gen functions are also provided.

Many processing functions are supplied for shaping individual curves, including filters, a compander, and a phase shifter. Furthermore, individual curves may be used to warp the shape of other curves. For example, one curve may “morph” into another to create a third curve, or several curves may be averaged over time to create another curve.

Because the program makes no assumptions about the target environment, files generated with curvePainter can be used to control any parameter in any kind of synthesis model. Curves might be used to control the time-varying amplitudes of harmonics in an additive synthesis instrument; the density, duration, and frequency of grains in a granular synthesis instrument; the trajectories of point sources in a spatialization instrument; or the breath pressure, embouchure, pitch, and vibrato of a flute physical model. The curves may also of course be used to create deterministic or stochastic score files.

Figure 1 illustrates the main window of curvePainter. The program draws each of up to 24 curves in separate colors.
The Generate Menu
Users may begin generating curves using the basic tools provided under the Generate menu. The primary curve generation techniques currently available are Gaussian, Impulse, Ramp, Random, Sinc, Sinusoid, Square, Triangle, Average, Arbitrary Function, and FM. When the user selects one of these from the Generate menu, a window of options is immediately presented. Most of the functions simply require data such as frequency, amplitude, phase, etc., to generate a new curve, as required by that particular function. The Average method, however, is unique in that other curves must be currently active. It simply creates a curve whose values represent the arithmetic mean of the curves selected to average. Also, users may enter a mathematical formula in two dimensions to generate a curve by selecting Arbitrary Function.

The curves generated using these techniques are by nature often quite simple. They serve as source material to create more complex curves via warping and modification in the transform menu.

The Transform Menu
Once basic curves have been created, they may be modified using various techniques. The Transform menu, shown in Figure 2, allows users to directly operate on previously drawn curves.

The DSP Menu
curvePainter can extract curves from spectral features of sound files, such as the spectral centroid, pitch, amplitude envelope (derived from the frame-by-frame power spectral density), and spectral peaks. When a user opens a sound file, a spectrogram immediately appears, along with a time-domain representation. The user then may select an operation to perform from the DSP menu that will extract a particular feature as a curve. Once the spectral feature has been turned into a curve, it may be modified and used just as any other curve.

The Graph Menu
The Graph menu features controls for the physical appearance of the curves. The user may toggle a grid overlay and choose between logarithmic and linear axes. Controls are also provided for choosing the colors of the graph.

The File Menu
The File menu allows users to save sets of curves for future modification. The Export function presents the user with various options for creating ASCII text files of floating-point numbers derived from the curves. Controls over the layout of the text file and the sampling frequency of the curves are also available.

Future Work
curvePainter was developed in Tcl/Tk and C and uses Kåre Sjölander’s Snack sound extension and the Tcl/Tk BLT extension. It runs under Linux and...
can also run under Irix and Windows 98/NT. A version for MacOS X is planned. Future work on the program might include a World Wide Web-based version.

A freehand drawing tool should be added to allow users to intuitively create arbitrarily shaped curves. Currently, users are bound to the built-in curve generation and transformation techniques, although it would be fairly straightforward to write plug-ins that provide new functionality.

The program currently provides a “Show Statistics” feature, available under the Edit menu, that calculates the average deviation, skew, kurtosis, maximum value, minimum value, mean, median, standard deviation, and variance of the selected curve. The window is shown in Figure 3 below.

<table>
<thead>
<tr>
<th><strong>Average Deviation:</strong></th>
<th>1.273763956</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Skew:</strong></td>
<td>1.193721604</td>
</tr>
<tr>
<td><strong>Kurtosis:</strong></td>
<td>-1.5037131846</td>
</tr>
<tr>
<td><strong>Maximum Value:</strong></td>
<td>3.9959976015</td>
</tr>
<tr>
<td><strong>Minimum Value:</strong></td>
<td>2.6417595901e-06</td>
</tr>
<tr>
<td><strong>Mean:</strong></td>
<td>2.0018138349</td>
</tr>
<tr>
<td><strong>Median Value:</strong></td>
<td>2.0031447319</td>
</tr>
<tr>
<td><strong>Standard Deviation:</strong></td>
<td>1.415302302</td>
</tr>
<tr>
<td><strong>Variance:</strong></td>
<td>2.0033072872</td>
</tr>
</tbody>
</table>

**Figure 3:** The Statistics window.

A helpful feature of the program would be the ability to warp curves so that they exhibit specified statistical values. For example, the user might want to increase the standard deviation of a curve’s values over time. Although there is of course no single method of uniquely changing a curve’s statistics, selected options might be computed and displayed.

**WORKS CITED**


