

Evochord

Gordon Monro
www.gommog.com

[From *Generative Arts Practice: A Creativity & Cognition Symposium*, ed. Ernest Edmonds, Paul Brown and Dave Burraston, Creativity and Cognition Studio Press, University of Technology, Sydney, 2005, pp. 140-143.]

Evochord is an installation in which a genetic algorithm tries to evolve a harmonious chord. Sound and visual output are generated in real time. The visual output consists of coloured shapes, one for each note of the chord. The pitches of the notes in the chord are not tied to any scale, but are freely variable.

The overall process is controlled by a variable mutation rate, which changes from high to low and back to high in a cycle lasting about one hour. The mutation rate is indicated visually by movement of the shapes on the screen. Each hour-long cycle is different, because of the random nature of the mutations.

In the *Evochord* installation a genetic algorithm tries to evolve a harmonious chord. The installation is intended primarily as an artistic experience and secondarily as a partial visualisation and sonification of a genetic algorithm.

1. The chords

Each chord has 19 notes. The same waveform is used for every note; it contains the first five harmonics, with amplitudes respectively 1.0 (the fundamental), 0.5, 0.3333, 0.25 and 0.2. This spectrum means that a single note has a relatively "solid" timbre.

The pitches of the fundamentals range from 50 Hz to 3200 Hz, a range of six octaves. Internally, pitch is measured in semitones above the lowest frequency, so the range is from 0 to 72 semitones. The notes are not tied to any particular scale, having a pitch resolution of approximately 1/900 of a semitone (0.11 cents).

The 19 notes of a chord are represented on the screen by a hexagonal array of coloured shapes (themselves distorted hexagons). The colour of a shape indicates pitch, red for low pitches, green for intermediate and blue or bluish-purple for high. The vertices of the shapes move according to the mutation rate, moving faster and further for higher mutation rates.

The sound output is in stereo, with the panning (how far left or right) of a note in the chord determined by the horizontal position of the shape on the screen. Apart from this, the position of the shapes on the screen

has no significance.

2. The measure of consonance or harmoniousness

In fact we work with dissonance: the higher the dissonance the lower the consonance, and *vice versa*. The dissonance of a chord is measured by a version of a formula given by William Sethares (Sethares 1993; Sethares 2005). The formula used in (Sethares 2005) for the dissonance of two sine tones of frequencies f_1 and f_2 Hz, with $f_1 \leq f_2$, is equivalent to the following.

Set $q = \frac{f_2 - f_1}{0.0207f_1 + 18.96}$. The dissonance is $e^{-0.8424q} - e^{-1.38q}$.

The dissonance between two complex tones is calculated by breaking the complex tones down into sine waves (partials) and then summing the dissonance between every pair of partials (one partial of a pair belonging to the first note, the other belonging to the second). In this sum the dissonance of a pair of partials is weighted by the product of the amplitude of the two partials. It should be noted that although two sinewaves in unison have zero dissonance, a complex tone will have nonzero dissonance according to this formula when paired with itself. This is because two different partials have some degree of dissonance when paired together.

For this work the dissonance between two complex tones was modified by a penalty factor, depending on the pitch of the lower of the two tones. The penalty factor was proportional to the pitch of the lower tone as measured in semitones, ranging from 1.0 for a pitch of 0 semitones to 6.0 for a pitch of 72 semitones. This modification was made for artistic reasons, as discussed below.

The total dissonance for a chord is obtained by summing the pairwise dissonance for every pair of notes in the chord.

3. The genetic algorithm

The genotype of a chord consists of $19 \times 16 = 304$ bits; each note in the chord is represented by a 16-bit quantity. Thus there are 65536 possible pitches for each note, distributed over a range of 72 semitones. This gives the pitch resolution of approximately 1/900 of a semitone mentioned above. Each 16-bit quantity corresponds to a specific fixed position on the display.

Initially a population of 100 chords is generated randomly. The genetic cycle proceeds as follows:

Breeding: Pairs of chords are chosen at random. One-point crossover is used, with a crossover point chosen at random; two children are

generated from the two parents.

Mutation: There are now 200 organisms, as the parents are retained. All 200 have bitwise mutation applied; the mutation rate is discussed below.

Culling: The population is divided at random into pairs, and the fitter chord of each pair is retained (tournaments of size 2). The population size is thus brought back to 100.

The retaining of the parents means that this algorithm is elitist, except that mutation is applied after breeding to the entire populations, parents included.

The mutation rate: The bitwise mutation rate varies cyclically over a cycle length of 1000 generations, from 0.5 to 0.0001 and back again. Let x be a quantity which increases linearly over the first half of a cycle from 0.0 at the start of the cycle to 1.0 halfway through. The mutation rate is given by a smoothly varying function of x , which has the following effect.

Table 1 – The varying mutation rate.

x	mut. rate	x	mut. rate
0.0	0.5	0.6	0.00017
0.2	0.0078	0.8	0.00011
0.4	0.00063	1.0	0.0001

The mutation rate is varied symmetrically in the second half of the cycle.

4. The display

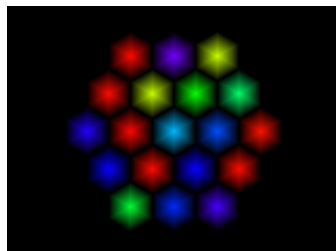
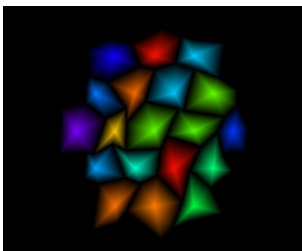


Figure 1 - The display at high mutation rate (left) and at a low mutation rate (right) Each generation takes four seconds. At any one time only the fittest

(least dissonant) chord is selected for display. If the previous chord was different, glissandi occur over four seconds as the old chord transitions to the new. The colours of the corresponding shapes likewise change over four seconds.

The movement of the vertices depends on the quantity x defined above. The movement is vigorous for $x = 0$; at $x = 1$ the shapes are static hexagons. The movement is not tied to the 4-second generation time. Cubic interpolation is used to get a smooth movement of vertices.

5. Results

At first the the chord is continually changing violently; as the mutation rate drops the fittest individual may remain the same for quite long periods, or subtle changes may occur. Beating patterns at various extended listening. As the mutation rate starts to increase in the second half of the cycle, the chord generally remains stable for a considerable time, but eventually is disrupted. The dissonance formula fits well enough with actual perception so that the chords the computer rates as more consonant generally sound that way as well.

The penalty for higher-frequency notes in the dissonance formula has the effect that the chords usually contain a good spread of notes from high to low. In arbitrary units, the dissonance at the start of a cycle is around 100; the least dissonant chord found during a cycle usually has dissonance around 40 to 50. This is not optimal, as the solution with all notes in the chord having the highest frequency (3200 Hz) has dissonance 0.05. Evidently a cycle does not contain enough generations with a low to moderate mutation rate for this solution to be found. The modified dissonance formula will slow down convergence to this solution, but should not block it entirely.

As noted above, *Evochord* is intended primarily as an artistic experience; the parameters used were chosen with this in mind. The intended effect is that different "interesting" chords will evolve each cycle.

References

Sethares, W. (1993) Local Consonance and the Relationship Between Timbre and Scale, *Journal of the Acoustical Society of America*, Vol 94, Issue 3 (September 1993), pp. 1218-1228.

Sethares, W. (2005) Some Useful Computer Programs.
<http://eceserv0.ece.wisc.edu/~sethares/comprog.html> (modified 7 Jan 2005; last viewed 14 Aug 05).