

Evolving Structure in Liquid Music

J. J. Ventrella

Jeffrey@Ventrella.com

Summary. A software application called “Musical Gene Pool” is described. It was designed to evolve non-linear music from an initial random soup of sounds, which play continuously. Most evolutionary music systems to date require the user to select for musical aspects in a piecemeal fashion, whereas this system is experienced as continuous music throughout the entire process, as follows: a human listener gives fitness rewards after sounds (organisms) emerge from the gene pool, take turns playing, and return back to the pool. Organisms start out unicellular (one sound), but as the listener selectively rewards random sequences deemed more musical than others, some organisms join up to form larger, multicellular organisms – which become phrases or extended musical gestures. Genetic operators of splitting, death, replication, and mutation occur in the gene pool among rewarded organisms. This results in gradual evolution of structure as the music continues to play. This emerges in response to the listener’s own internal emerging musical language, based on accumulated musical memory. This structure is *liquid* – continually able to flow and rearrange to allow serendipity. While there is a limit to organism length (duration of phrases), it is proposed that the interactive scheme could be adjusted to evolve increasingly larger organisms, and hence, longer musical passages. These would essentially be mobile chunks of linear music with self-similarity in their structures – revealing the histories of their evolution.

1.1 Introduction

Music “...is a fluid reality. The only thing that primitive polyphony, classical counterpoint, tonal harmony, twelve-tone serial music, and electronic music have in common is the principle of giving form to noise in accordance with changing syntactic structure.” Jaques Attali, *Noise*, p. 10.

Definitions of music that acknowledge the act of performing and listening are *phenomenological*, as contrasted with descriptions of music as “object” – as a piece of structure, possessing meaning in and of itself [1]. In this chapter, we take a phenomenological stance and elevate the role of listening. In fact, we

make the act of listening *participatory* – whereby it becomes the catalyst for the generation of the music itself. Attali [2] describes music as “giving form to noise.” Here is described a music generation system where form emerges from noise through interaction: the listener and an evolvable population of sounds become collaborative agents, changing over time in response to each other.

The *Musical Gene Pool* is an interactive software application which generates fluid music with no explicit beginning, middle, or end. It is seeded with a primordial soup of random sounds, and, given occasional feedback from a listener, evolves to become increasingly structured. Once the amount of structure in the gene pool has reached the highest possible level, continued listener feedback can still change the *quality* of that structure. If feedback is discontinued, the music keeps playing, maintaining the same quality.

While this experimental musical form currently requires the listener to sit at a computer and respond via mouse or keyboard clicks, the preferred usage is for the listener (or *listeners*) to be remote from the computer, possibly sitting in a comfortable chair or walking around doing some activity, with some way to provide occasional feedback using a clicker. Given sufficient population size, musical operators, and sufficient time for listening and evolving, it could produce rich and varied music. Music which harnesses implicit representations of the aesthetics of the listeners who have interacted with it.

The resulting populations in the gene pool could be stored in digital files, and “run” (as in running a software simulation). These files would be extremely small compared with the size of a piece of music stored as a typical MIDI file, where the linear ordering of events maps directly to time. The advent of evolutionary music provides opportunities for non-linear forms of music storage, as proposed by Brown [3]. One can describe the Musical Gene Pool files as compressed forms of linear music: when considering the amount of repetition, theme, and variation in most music, the Musical Gene Pool stores the *seeds* of this variation in a compact form. Each time it is run, the variation unfolds, and it is never twice the same.

1.1.1 Evolutionary Music

Composers and scientists alike have taken a recent interest in evolutionary music composition. There are now conferences and publications dedicated to evolutionary music and art [4]. Burton and Vladimirova [5] provide an introduction to techniques, such as the genetic algorithm (GA) [6] and their applications to music generation. Miranda and Biles [7] provide more recent coverage of progress in evolutionary music. Some of these systems are based on the idea of an *artificial composer* – and seek to represent essential aspects of composition in some encoded form, either for automatic music generation or as a way to better understand the act of composition [8] [9]. Some systems simulate virtual ecologies in which sonic communication evolves for survival purposes and have musical value [10] [11]. Other systems acknowledge the

human composer/listener as still far superior to any artificial agent. The *interactive evolution* approach uses variations on the genetic algorithm, which includes a human to provide the fitness values for a genetic algorithm. That is the approach used here.

1.1.2 Schoenberg, Cage, Reich, Eno

In advancing the art of evolutionary music, we must acknowledge the pioneers who have shaped the modern musical landscape and expanded our vocabulary of what music is or can be. Schoenberg, originated dodecaphonic music: the 12-tone row and techniques for manipulating these musical seeds. While atonal music may not sink deep into the human psyche (for many people), it provides an intellectual basis for appreciation of the algorithmic nature of musical variation. John Cage opened our ears to the music of the natural world, to randomness, and to the importance of silence. Steve Reich’s minimalist phase compositions place the listener in a timeless dimension where gradual change in texture becomes a primary musical perception. Brian Eno enriched contemporary pop music with technical innovations, notably tape loops generating ambient soundscapes that never repeat the same patterns. He also has helped to make popular the idea of Generative Music [12]. These composers are of course just a few of the many pioneers, but their innovations have relevance to this project.

Perhaps most important is the stance that John Cage encouraged us to take when thinking about music, as “purposeless play” – and so as with the Musical Gene Pool, the listener is encouraged to flow with the random chance that is at the bottom of the pool, and to suspend expectations and desires of how the music “should” come about. Taking this stance can have the best results, acknowledging that the listener and the gene pool are equal collaborators in the creation of the music. Using a bottom-up attitude to music generation invites serendipity, which is the creative engine of the Musical Gene Pool.

1.2 Description

The Musical Gene Pool model consists of three basic elements:

1. a population of musical organisms (the gene pool)
2. a mechanism for playing the sounds of the organisms (the *ear*)
3. listener feedback in the form of a single binary signal

1.2.1 Organisms

The gene pool consists of hundreds of organisms. An organism is a collection of one or more cells; thus, organisms can be unicellular or multicellular. Unicellular organisms specify one sound, while multicellular organisms specify a

series of sounds played over time, as in a musical phrase. Figure 1.1 illustrates a multicellular organism made of six cells, numbered 0 through 5. The lines connecting the cells represent the delays between cells as they make sounds. Notice that cell 0 in this example has a line extending to the right. This indicates the duration of a delay *before* it plays. Also notice that cell 3 has zero delay, and so it overlaps with cell 2: the two cells produce simultaneous sounds played together. Having two cells play simultaneously creates harmony. Any number of contiguous cells can have *zero* delay, to create chords.

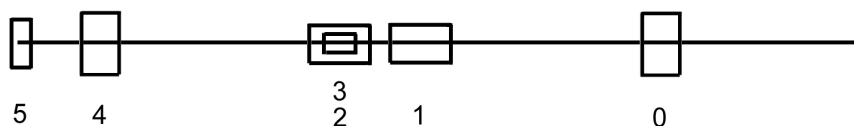


Fig. 1.1. A six-celled organism

Each cell has five attributes, the first four of which correspond to standard MIDI. These are as follows:

1. **Pitch:** the “note” of the sound. This ranges from P_{min} to P_{max} .
2. **Volume:** (velocity) ranges from V_{min} to V_{max} .
3. **Instrument:** ranges from 0 to I . This corresponds to the General MIDI standard mapping of instruments (such as “grand piano,” “cello,” etc.)
4. **Duration:** how long the sound is played. This ranges from D_{min} to D_{max} , in units of milliseconds. Note: this value does not apply to “one-shot” sounds, or sounds that have short sustain and decay, such as *xylophone* or *bell*.
5. **Delay:** how long the cell waits after the last cell played before it plays (indicated by the lengths of the lines in Fig. 1.1). This can be any value in the set D (0, 1, 2, 3, 4, 8), which is multiplied by the constant D_s to map it to milliseconds. D_s affects the tempo of the music. There are no explicit denotations of the concept of “half note,” “quarter note,” etc.

The exact values used in the experiment are given at the end of this section.

Random organisms are continually chosen from the population and lined up end-to-end to form a continuous stream of cells (sounds) which are played in sequence. After each organism has had its turn playing its cell’s sounds, it returns back into the population. We shall now precisely define the “gene pool” as this population. It *does not* include those organisms which have left the pool and are part of the stream.

1.2.2 An Explanation by Way of the Visual Interface

The process shall be explained by way of the visual interface and the behavior of the application, which was designed to make the process intuitive. This is

shown in Fig. 1.2. The gene pool is represented as the region at the bottom. A stream of organisms can be seen emerging from the pool at the bottom left, and the white arrow shows the initial direction that organisms move in after leaving the pool and taking their positions along the stream. The cells pass through the sound-playing threshold at top (the ear), and then descend back into the pool. While playing a sound, organisms move from left to right, passing through a sound-playing threshold; thus, the cells are actually heard in order from right to left. And so cell 0 (the “head cell”) plays first, as indicated in Fig. 1.1.

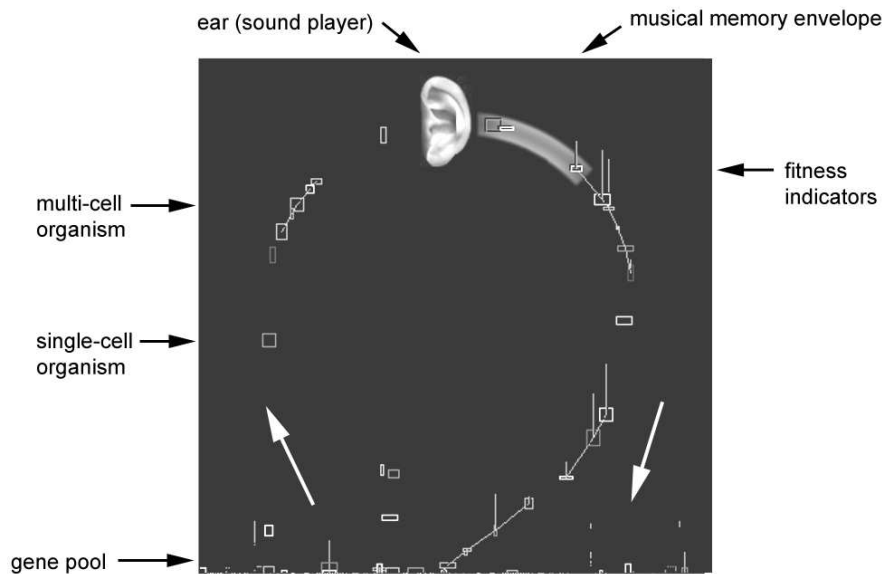


Fig. 1.2. The visual interface for the Musical Gene Pool

The stream of organisms follows a circular path which rotates clockwise at a rate of 0.0666 Hz (one rotation every 15 seconds). This rate affects the *tempo* of the music, along with the cell delay scalar D_s . Cells are visualized as rectangles. The height of the rectangle is determined by pitch, and the width is determined by duration. The color of the rectangle is specified using hue, saturation, and luminance values, as follows: hue is determined by instrument, saturation is determined by pitch, and luminance is determined by volume. This mapping, while somewhat arbitrary, does visualize all of the attributes of a cell, and ensures a substantial visual variety. This visual interface provides clarification and motivation for the listener/composer using the application. But here is an important point: the visual display is entirely unnecessary for this system to work. All that is really needed are the listener’s ears and brain,

and a way to provide reward. In fact, in preliminary tests with subjects, the visuals were found to be a distraction to pure musical judgment, as discussed below.

1.2.3 Listener Feedback

Many interactive evolutionary music systems developed so far use a rather piecemeal approach to evolving music: the user chooses an individual (a sound, musical fragment, etc.), listens to it, rates it, chooses another, listens, rates it, etc. – and then updates the generation to create a new population. The regimen is not unlike the standard GA, but with a human inserted into the selection stage. The Musical Gene Pool was designed to recognize that the human must be able to experience this music holistically in order to make meaningful choices. And so it provides musical continuity throughout the entire evolutionary process.

As the organisms stream through the ear, and create continuous sounds, the listener can respond to sounds recently heard by sending a binary signal (either by pressing the mouse button in the window or pressing the spacebar on the keyboard). A duration of $M = 1800$ milliseconds (just under two seconds) after a sound passes through the ear is called the “musical memory envelope,” illustrated in Fig. 1.2 as the region to the right of the ear. The listener is asked to reward any sounds or sound sequences that are interesting, expressive, curious, or in any way worthy of encouragement for further experimentation. When the user issues a reward, the cells that lie in the musical memory envelope receive *fitness values*, which are visualized in Fig. 1.2 as vertical lines, or *fitness indicators*.

The envelope carries the most strength in the middle and the least strength at the beginning and end, as shown in Fig. 1.3.

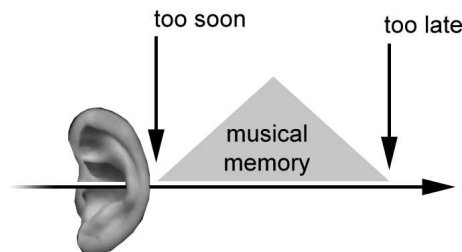


Fig. 1.3. Musical memory envelope

The strength of a cell’s reward is determined by where the cell lies in the envelope. Cells that have just played and lie at the leftmost region of the envelope receive a smaller fitness value, under the assumption that they are too recent to contribute to the holistic musical experience that the listener is

responding to. Cells that lie at the rightmost end of the envelope also receive a small fitness value, under the assumption that too much time has passed for this sound to have contributed to the holistic musical experience. If the sound sequences are especially desirable, and remain so for extended time, the listener can hold down on the key continuously. This is indicated by the long black rectangle in Fig. 1.4, resulting in a *plateau* of maximum fitness values. Notice the vertical lines at the bottom, which show the fitness values given to the cells (sounds) at the top.

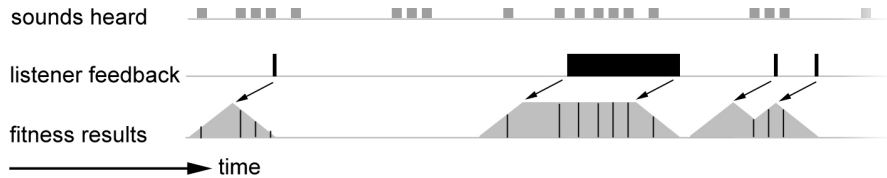


Fig. 1.4. Fitness values applied to cells

1.2.4 But Just How Long Is Musical Memory?

In reality, musical memory covers many arbitrary lengths of time, ranging from the momentary thrill of a trill to the lifelong memory of a tune. All ranges of memory contribute to one's reaction to music. Admittedly, the Musical Gene Pool only accounts for the reactions to musical experience that can occur within a few seconds. This may seem like a gross oversimplification. However, there are some subtle aspects to how listener behavior changes over time as the character of the music takes shape. As the gene pool becomes more self-similar, the listener begins to respond not just to immediate sound sequences, but also to how these sounds fit within the emerging quality of the music as a whole. The listener's internal representation of the emerging musical language becomes a factor in judgment; this is discussed in more detail later in the chapter.

The preferred duration of the musical memory envelope may vary with listeners, as well as over time while listening. Since there is no such thing as a *correct* duration, this should be made into an adjustable parameter (in a subsequent version of the software) which the listener can change at any time.

1.2.5 Genetic Operators

Besides receiving fitness values as a result of the listener-rewarding cells just heard, there is another important operation: cells automatically join up with neighboring cells occurring in the envelope (if they are not already part of a common organism). This causes all cells in the envelope to merge into the same organism. This is the *join* operator, and it is one of the genetic operators

used in this system. The reason for using the join operator is that the listener has presumably responded to the ordering of these sounds, and not merely to the sounds themselves. It is meant to correspond with music experience of the sounds in combination: as a phrase, gesture, motif, or rhythmic expression. An organism can grow to up to $C = 16$ cells only, and so in the case when a join would cause an organism to become larger, the join operation is not performed.

The fitness of an organism is measured as the sum of all its cells' fitness values. After an organism with positive fitness has returned to the gene pool it has a certain number of time steps F in which it is "fertile," meaning it can either *split* or *replicate* (with chance of mutation after replication). The chance of either of these operations occurring is proportional to its fitness: the higher the fitness, the higher the chance of one of these operations occurring. An overview of the genetic operators is shown in Fig. 1.5. The *join* operation occurs outside of the gene pool and is the direct result of listener interaction. The *split* and *replicate* operations are performed while in the gene pool, and so they happen unknown to the listener. But their effects become apparent over time, after affected organisms begin emerging from the gene pool and make themselves known. These two operators are described in detail below.

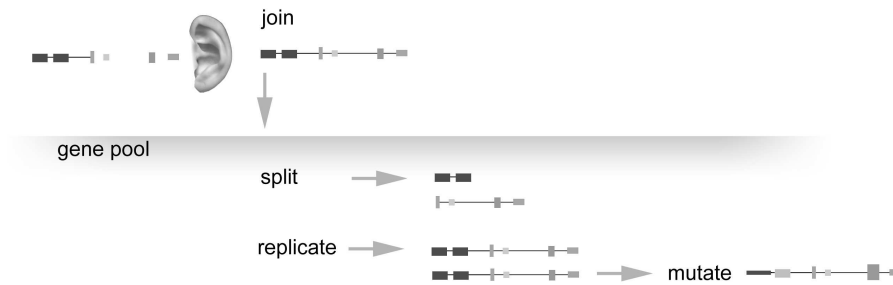


Fig. 1.5. Genetic operators

If an organism is in the gene pool, and if it has positive fitness, and if it is fertile, then there is

1. a random chance S of splitting:
The organism is severed at some split locus cell, and becomes two organisms. The split locus is determined by randomly choosing a cell with a relatively low fitness value.
2. a random chance R of replication:
A copy of the organism is made and added to the population.
MUTATIONS: As soon as this organism is created, it has a random chance of any of the following mutations occurring

Pitch shift Pm (the pitch values of all cells shift equally by a random value ranging from -12 to 12)

Instrument Homogenization Ihm (a cell with relatively high fitness is chosen and its instrument value is randomly copied to a percentage $H = 0.5$ of the organism's cells).

Instrument Shift Ism (the instrument value of all cells are shifted uniformly by a random amount in the range -5 to 5).

Octave Normalization Om (the average pitch value of all cells is determined. Then, any cell whose pitch has a difference with this average of more than 6 is shifted towards the average by a value of 12. This reduces large jumps in pitch, and encourages more melodic sequences)

Cell Harmonization Hm (a random cell is chosen and doubled. It is then given a delay of zero, and its pitch is shifted by a random amount ranging from -12 to 12)

Arpeggiate Am (if any cell has a delay of zero, its delay is changed to a random delay value within the set D)

Zap Zm (a random attribute of a random cell is changed to a random value)

Scramble Sm (the ordering of the cells is randomly shuffled)

Harmonize and *arpeggiate* are complimentary: *harmonize* tends to make organisms denser, while *arpeggiate* thins them out.

S and R (rates of splitting and replicating) are effectively lower, due to the nature of the algorithm. The algorithm chooses a random organism from the entire population as a candidate to apply the operator – but if that organism is not fertile, has zero fitness, or is not in the gene pool, then the operator is not applied.

Other Mutations

Towsey, et al [9] describe twenty-one “melodic features” used to manipulate melodies in a genetic algorithm-based system for automated music composition. Other evolutionary music systems employ similar melodic mutations. Currently only a handful of musical mutations are applied in the Musical Gene Pool, as described above, but others have been tried, and many more are intended to be added in subsequent versions, such as

1. *delay homogenization* (taking the delay values from some cells and copying them to other cells. This has been tried, but it is problematic as it tends to result in too many evenly-spaced delays)
2. *overlap* (taking two organisms and combining them – side-to-side, like earthworms mating – so that they overlap in time, thus creating a single, more dense organism with higher likelihood of harmonization and combined instrument sounds).
3. *stutter* (taking an arbitrary piece of an organism and generating a new organism by attaching copies of this piece end-to-end)
4. *octave shift* (shifting all pitch values up or down by a value of 12)

5. *retrograde* (reversal in time, as in dodecaphony)
6. *inversion* (reversal in pitch, as in dodecaphony)
7. *retrograde plus inversion* (as in dodecaphony)
8. *pitch compression/expansion* (scaling the range of pitch values)
9. *general homogenization* (taking an arbitrary attribute, such as volume, and making all the cells' attributes take on the same value, chosen from a high-fitness cell. This has been tried, but it is problematic, as it tends to result in organisms with too much order and symmetry.)

These are just a few of the possible mutations which have been explored, or are candidates for exploration. It was found that as more mutations get added to the mix, the chance of these mutations being applied must be lowered, so as to preserve the original form and not dissolve existing structure. This is a common balancing act in many genetic-algorithm-based systems.

1.2.6 Genotype Representation

Note that the genetic operators *join* and *split* are performed at the *cell level*, meaning that a *cell* is analogous to a *gene*. And so the *genotype* of an organism would be defined as the array of cells. This is a high-level genotype representation. When a join or split occurs, it occurs between the *cells*, considered as hard units. The reason for this level of representation is to allow these genetic operators to map more easily to musical meaning. The mutations on the other hand have a more granular effect, and can change a single attribute of a cell, although here too, these mutations are still performed in recognition of the cellular organization of the genotype.

1.2.7 Extended-Time Crossover

A key ingredient in the standard genetic algorithm technique is *crossover* – the recombination of genetic material between two parent genotypes, usually to create offspring genotypes. One may ask why crossover is not used in this scheme. But in fact, an effect quite similar to crossover does occur as a result of organisms splitting and rejoining over extended time. The important difference is that this is not *sexual* reproduction in the strict sense that two parent genotypes combine genes to produce offspring genotypes. Instead, organisms reproduce *asexually* (via the replicate operator). But the fact that they can split, and rejoin with other organisms later, gives rise to the kind of experimental mixing that we normally attribute to sex. This special brand of extended-time crossover is deemed more applicable to the Musical Gene Pool, as it allows listener discretion to be a direct factor in recombination.

1.2.8 Managing Population Count

A bit of explanation concerning software implementation is in order as far as representing the population of organisms in the gene pool. An array of size P is

used to store the organisms. Each organism in the array is itself represented as an array containing the values that determine its cells' attributes. Array sizes can easily be adjusted to accommodate different usage scenarios. The genetic operators of joining, replicating, and splitting affect the population array. They are not applied to any of the organisms which have left the gene pool and are part of the stream. First we explain the join operator: when organism b joins onto the end of organism a , organism a grows in length, taking copies of the cells from organism b ; and then the original organism b dies. This marks the array index of organism b as *dead*, and decreases the population size by 1. Replicating and splitting, on the other hand, increase the population size by 1 each. In this case, if the population size is below the maximum P , then the first dead index in the array (resulting from a previous join operation) is chosen to store the newly created organism. If the population is already maxed out (no indices are marked dead), then a relatively low-fit organism is chosen to die, leaving room for the new organism. The relatively low-fit organism is chosen by "tournament selection" (two organisms are randomly selected, and the one with the lowest fitness is taken).

Note that even though the organisms in the population are kept in a linear, ordered array, the ordering is irrelevant to user interaction and the way the music evolves (because organisms emerge from the gene pool randomly).

At the very beginning, population size is P (the maximum size), but it quickly decreases as soon as the listener starts rewarding cells and they begin joining to form multicellular organisms. Also at first there are more of them dying than replicating or splitting. But soon after, as the gene pool starts brewing, the population size increases. In a mature gene pool it will approach and remain at or near the maximum P . Population dynamics are of course sensitive to the specific tuning of the rates of replication and splitting, in combination with the behavior of the listener, which affects the frequency of join operations.

1.2.9 Using MIDI Channels to Manage Instrumentation

The Musical Gene Pool uses the Java MIDI package. Many modern computers come with an on-board MIDI synthesizer with 16 channels. A common usage is to assign a unique instrument to each channel (or to just use the default instruments), giving a palette of 16 instruments which can play in parallel, so they can be heard overlapping in time. In order for an arbitrary number of instruments to be specified (more than 16), a technique is used in this system to approximate the effect of arbitrarily many channels. This is accomplished by cycling through the channels, incrementing a *channel count* each time a sound is played, and reassigning the instrument of that channel through a *program change event*. (Channel 10 is skipped: on many standard MIDI synthesizers, this channel is reserved for percussion sounds, where the pitch parameter is used for percussion sounds).

1.2.10 Parameters

Table 1.1 shows the complete list of all parameter values that have been chosen. These parameters were chosen to optimize the listener experience in the context of the Java applet described here. These values would be adjusted in the case of adapting this experiment to form a large, collaborative sound ecology (as described below).

Table 1.1. Parameters

Symbol	Description	Value	Units
t	Time step duration	1/30	second
P	Maximum population size	200	count
C	Max cells per organism	16	count
D_s	Delay scalar	120	milliseconds
D_{min}	Minimum cell sound duration	10	milliseconds
D_{max}	Maximum cell sound duration	700	milliseconds
P_{min}	Minimum cell sound pitch	30	general MIDI
P_{max}	Maximum cell sound pitch	90	general MIDI
V_{min}	Minimum cell sound volume	30	general MIDI
V_{max}	Maximum cell sound volume	127	general MIDI
I	Number of cell sound instruments	64	general MIDI
D	The set of cell delay values	0, 1, 2, 3, 4, 8	delay units
S	Split Rate	0.3	chance per time step
R	Replication Rate	0.3	chance per time step
P_m	Pitch Shift Mutation Rate	0.1	chance per time step
I_{sm}	Instrument Shift Mutation Rate	0.05	chance per time step
O_m	Octave Normalize Mutation Rate	0.2	chance per time step
I_{hm}	Inst. Homogenize Mutation Rate	0.5	chance per time step
H_m	Harmonize Mutation Rate	0.05	chance per time step
A_m	Arpeggiate Mutation Rate	0.01	chance per time step
Z_m	Zap Mutation Rate	0.05	chance per time step
S_m	Scramble Mutation Rate	0.05	chance per time step
M	Musical Memory Envelope	1800	milliseconds
H	Inst. Homogenize Percentage	0.5	chance per cell
F	Fertility Duration	500	time steps

1.3 How Musical Structure Evolves

As the term “gene pool” indicates, the organisms are free-floating: they have no predetermined positions within the population, and so they can line up in any sequence. When unicellular organisms join to form multicellular organisms they begin to form some degree of structure in terms of cell positioning, and thus in sounds occurring over time. There is an analogy to the states of matter:

a population of unicellular organisms is similar to a gas. A population with larger organisms might be seen as a liquid – having some structure, but still able to “flow” with listener response. This is illustrated in Fig. 1.6.

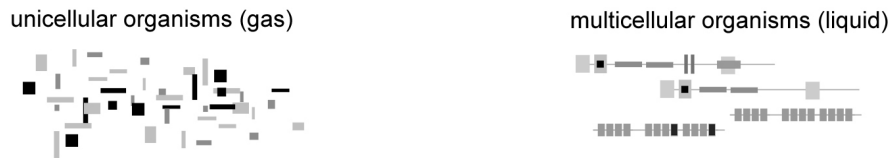


Fig. 1.6. Musical gas and musical liquid

1.3.1 Listening is Composing

How can composing be reduced to a simple binary act, in the form of a mouse click or keyboard strike? This may seem offensive to accomplished composers, but it is of course not meant to replace, or in any way compare to the act of composing, which requires top-down design, and involves great skill. On the contrary, this is a form of bottom-up emergence, which begins with chaos, and gradually acquires order. The nature of that order depends of course on the tastes of the listener, and relatively little skill is required (although it helps).

1.3.2 Preliminary Tests

Preliminary, informal tests were done on about six subjects, each using the Musical Gene Pool for some duration ranging from about five minutes to about 30 minutes. At the very beginning, when all organisms are unicellular, and all attributes are randomly distributed evenly over the full range, the listener is asked to simply respond to sounds and sound sequences that have some vague musical interest, so as to get things rolling. In the early stages, it has been observed that the listener will generally respond to timbre, since there are so many instrument variations, and virtually no melodic or rhythmic structure to speak of. But over time, the mutation causing instrument homogenization tends to decrease the variety of timbre, and the listener begins to respond more to instances of melodic or rhythmic structure. This is when things start to get interesting.

1.3.3 Attraction to a Tonal Center

Dodecaphonic music sought to free musical expression from the tyranny of tonality; it was one of the most significant experiments in modern music. But we have reason to believe that human ears and brains have a predisposition to

melodies and harmonies which have a hierarchical structure. Theories of tonality are continually being developed, giving support to this claim [13]. The logic and hierarchy of traditional tonality is easily mapped to the overtone series, which occurs in nature [14]. The natural favoring of tonal centers is evident in the behavior of the listener of the Musical Gene Pool, based on observation. This may be enhanced since the listener is concentrating on finding musical sense, i.e., seeking order. However, there may be a purely mathematical reason for a tonal center to emerge initially; it may simply be a function of the gene pool becoming more homogeneous: a specific pitch may become more common (or perhaps more prominent based on the instrument or volume). Even without the listener gravitating towards a tonal center, this pitch may propagate in the gene pool. It is suspected that any slight increase of an arbitrary pitch will cause the listener to hear tonality (usually this pitch is heard initially as a tonic or a fifth). Once this feedback cycle begins, the tonality increases, and the listener becomes more responsive to the subtle aspects of the emerging key. Tonal center, or any musical feature for that matter, cannot be attributed only to the listener. It results from listener gene pool feedback.

1.3.4 Attraction to Repetition

The human brain is not only good at detecting sequences of patterns, it is actually wired to do this, and on multiple layers of the neocortex [15]. And so we are sensitive to the slightest hint of repetition. If an organism has replicated and if by chance the two copies pass through the ear one right after the other, then this causes a repetition of the sound sequence over a short duration of time. Even if each organism in itself is not musically meaningful, the repetition is experienced as structure for the listener: it becomes a pattern, and this is quickly perceived. Much like the arbitrary convergence of a tonal center, some forms of musical structure may emerge simply as a result of the listener responding to repetition. Familiarity with this pattern reinforces its musical value, and it causes a feedback loop.

1.3.5 Context

An organism that has little musical meaning at one point in time for the listener may become potent at another stage based on its context. For instance, it may follow a different organism which sets the musical stage for new meaning. Also, by the time the organism comes around again, the listener may have entered into a different modality of listening, perhaps focusing on a new tonal center or a new rhythm. In either case, *context* – whether within the sequence of organisms or within the listener’s mind – can change the meaning of an organism. In Darwinian terms, a change in its ecological niche may change its fitness.

1.3.6 What Kind of Music Emerges?

By now the reader may be wondering what kind of music emerges. To begin, here is what *does not* emerge: (1) specific, familiar melodies in their entirety (although snippets of familiar tunes may emerge). (2) Clearly-defined chordal structure based on traditional Western harmony (although, again, moments of chordal logic can emerge). (3) Strongly metered rhythm. What *can* emerge could be described as alternating moments of impressionistic, expressionistic, and minimalist passages, which are sometimes mediocre, and at other times quite magical. The listener can push the music towards a strongly rhythmic, staccato style, or towards a slower, softer style. Given enough effort by the listener, distinct keys can be fished out, and distinct key changes can occur as well (thanks to the pitch-shift mutation). These key changes, along with changes in instrumentation, as varying strains of organisms take their turns passing through the ear, create the *impressionistic* effect. This tends to diminish over time as the gene pool becomes more regular. A *minimalistic* effect comes about if the listener encourages a single strain of organism to populate the pool, at the expense of others. The listener can also focus on breeding a certain melodic motif, with a characteristic instrumentation and rhythm. Trying to describe the music in words cannot do it justice, and so the reader is encouraged to run the Java applet [16].

1.3.7 From Chaos to Order, Stopping at Complexity

Music offers a great illustration of complexity. Utter randomness has no informational value. The same goes for total order. For instance, a large orchestra in which every musician is rapidly blasting out random notes will result in a wash of undifferentiated cacophony – not much information content, and little musical meaning. At the opposite extreme, if all the musicians repeatedly played the same note over and over again with no variation, again there would be little information content or musical meaning. Somewhere between these extremes lies music. The bifurcation diagram [17] in Fig. 1.7 illustrates this notion.

Predictability and regularity serve as the background against which unpredictability or novelty (surprise) serves as the informational aspect of a message. The Musical Gene Pool begins at the random end of the scale. As soon as order begins to emerge (such as a replicated pattern, or the homogenization of instruments in an organism), the listener is able to begin using it as background: it takes on an environmental purpose in the musical landscape, and the surprises are read as foreground, i.e., as having musical content.

Styles in music around the world range from orderly to highly chaotic – from the simplest Gregorian chants to the free jazz of Ornette Coleman. But the Musical Gene Pool does not move in this direction. It is also the opposite of some improvisational schemes, which begin with a simple groove or motif and build increasing complexity, syncopation, and variation. Instead,

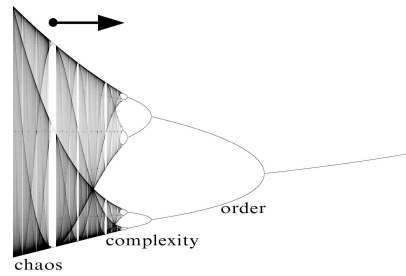


Fig. 1.7. Musical Gene Pool trajectory

the Musical Gene Pool begins with randomness, and gently introduces bits of order here and there, which the listener is invited to respond to. It is not possible to reach the exquisite mix of regularity and surprise of Beethoven, much less the degree of expressiveness. It would be cause for celebration if even brief Beethoven moments did emerge. The primary goal is emergence from total randomness to some degree of complexity, as the listener responds to pieces of emerging order. The path from chaos towards order, and stopping at complexity (and then breeding different flavors of complexity), is what the Musical Gene Pool achieves.

1.3.8 Blind Mode – the Purest Form

To use the application in the purest way, one should turn off the computer monitor, and allow only audition to determine the response, so as not to be distracted by the visuals. As an example of visual distraction, it is possible that the listener, when becoming accustomed to a family of organisms containing a large yellow square “causing” a desirable sound, will begin to respond to organisms with large yellow squares right as they pass through the ear – before even hearing them! Indeed, human vision and audition are tightly linked, especially when sound and visual stimuli occur in synchrony, creating a sense of cause and effect. And since humans are highly visual, it is possible that the active listener may become an active “watcher,” without realizing it. Subjects who have tried it out have commented how very different the experience is with eyes closed.

Running this in “blind mode” would of course be less entertaining and less educational than watching flowing strings of colorful shapes meander on the screen. On the other hand, imagine yourself sitting in a comfortable chair with your eyes closed, and with a clicker in hand. You allow yourself a half hour, and all you have to do is press the clicker when you like what you hear. The chances are you will start to hear more of what you like. And perhaps you will come to like it so much that you will just drop the clicker, sit back, and enjoy.

1.4 Scaling Up

The Musical Gene Pool is considered as a first step, perhaps a prototype, for an extensive sound ecology with a large, slowly evolving population of large organisms, and a collaborative listening community. A large database of organisms could be stored on a server and made accessible online such that multiple listeners could contribute to its evolution over the span of many months. Perhaps listeners would be able to save the state of local gene pools, and then upload them to public server pools from which others could download and mix them into their own local pools.

There is an analogy to the real “musical gene pool” – the one that exists out there in the world: it is the sum total of all music throughout history, with all its variations, cultural influences, memetics, hybridizations, etc.). If the Musical Gene Pool could be made malleable and large enough, and if it could be exposed to the global Internet population, it might begin to mirror some aspects of the global musical gene pool.

This system would include the following:

Longer Organisms The maximum size of an organism is currently 16 cells, which can harness a small representation of listener aesthetics, and generate musical phrases. But ideally, the act of joining organisms should extend over larger time spans, enabling longer musical passages.

Larger Populations A population of 200 is sufficient for experimenting with a single listener, and it takes about a half hour of continuous use to reach a saturation point of musical structure. But after listening to an evolved gene pool for longer than about five minutes, the self-similarity caused by organisms and their kin replaying may be too repetitive for some ears. On the other hand, imagine a population of tens of thousands of large organisms continually evolving under the direction of many listeners. This musical ecosystem would be varied and complex indeed. This might have to be built up incrementally by adding smaller (human-sized) pre-evolved pools. A randomly seeded population which is extremely large would take prohibitively long to generate any musical satisfaction for a listener. But, a large pool which has already been seeded with organisms that were pre-evolved by many listeners would be more approachable. And it could hold immense musical variety.

More Mutations As explained above, there are many forms of musical variation that have been described in algorithmic form and used in similar evolutionary systems. Having a larger set of such mutations would expand the repertoire of possibilities for the listener and encourage more kinds of musical structure to emerge.

1.4.1 Super-organisms

Earlier we considered a population of unicellular organisms as a gas, and a population of multicellular organisms as a liquid (more structured, yet still able to flow with user interaction). Let us take this analogy to the extreme: consider what would happen if organisms were allowed to join up indefinitely, to grow in size, and to eventually freeze up into a single super-organism. What we would end up with is a single linear piece of music – a solid. This solid piece of music would no longer be interactive, but it would be an interesting artifact indeed – it would reveal the history of its own evolution in its structure, as indicated in Fig. 1.8. This organism would be made from pieces of smaller, older organisms, which themselves would be made from pieces of yet smaller, older organisms. And since there is repetition at all levels, this structure would probably exhibit a degree of fractal self-similarity.

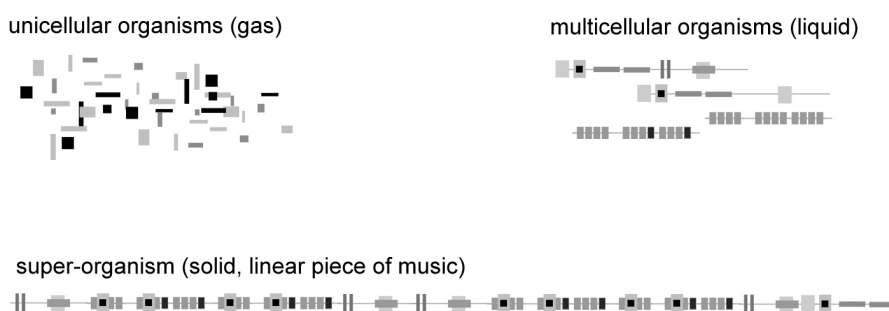


Fig. 1.8. Emergence of self-similarity in a super-organism

Does this mean that the Musical Gene Pool could evolve structure along the entire spectrum from gas to liquid to solid, resulting in one piece of linear music? Certainly not with the current interactive scheme, which is inherently liquid. The splitting operator is currently set at a constant random rate, and the maximum size of an organism is 16 cells. These two factors limit how large organisms can potentially grow. In order to grow increasingly larger organisms, there would have to be no limit to size, and the split operation would have to decrease over time as evolution progresses, until, at the very end, no organisms can split, and a single organism results, filling up the entire space of the population. But even if these changes were made, it would be difficult to evolve a super-organism due to the nature of listener feedback.

Consider a hypothetical scenario in which the listener is evolving a population of thousands of organisms: they are allowed to grow indefinitely, and the split operator has decayed to zero. Imagine that the listener has reduced the number of organisms down to only two. Now, if the goal were to end up with one super-organism, then the listener would have to wait for whatever organism that is playing to finish – for many minutes (perhaps even an hour,

if the delays are long). And just when this organism leaves the ear and the other organism enters, the listener would have to issue a single reward (just in time!) in order to join them. This is not an ideal way to finish the journey.

Even though the system is not set up for this, it is hard to resist the idea of generating a super-organism with deep fractal self-similarity which stands as the evolutionary culmination of many hours of listening and rewarding. Exactly how to accomplish this is uncertain, and in fact it may be contradictory or paradoxical. If the Musical Gene Pool is inherently liquid, then once the organisms have reached a considerably large size, the idea of bottom-up emergence no longer holds, and it must give way to a form of top-down design. In this case the listener is best advised to switch to an editing interface where the remaining large organisms could be spliced together by hand.

1.5 Conclusions

The Musical Gene Pool is an experiment in blurring the distinction between listening and composing. The population of musical organisms is considered to be a collaborator in the process of emergent music. Perhaps one might argue that the Musical Gene Pool system, with its carefully tweaked operators, is more of a composer than the human participant. However, nothing could happen unless the two come together and interact. It is best to think of the human listener and the gene pool as two agents in an improvisation – a dynamic ecosystem. And while this interaction flows over time, it is intriguing to observe how musical memory accumulates in the listener’s mind, and how this changes the way the listener guides the gene pool. This tool may not produce music worthy of Carnegie Hall. But it is hoped that these experiments open our ears and mind more to the psychology of listening and composing, and why it is that some sound combinations, and not others, have musical meaning.

1.6 Acknowledgements

I would like to thank Gary Walker, who taught me how to play guitar and the basics of music theory when I was ten years old, and who recently contributed ideas for this project. Also, thanks to Eddie Elliot, an artistic and technical collaborator of many years, who taught me Java, MIDI programming, and contributed ideas for this project.

References

1. Sansom, M.J.: Understanding musical meaning: Interpretative phenomenological analysis and improvisation. In: British Forum for Ethnomusicology, Annual

- Conference – Music and Dance Performance: Cross-Cultural Approaches, SOAS. (2005)
2. Attali, J.: *Noise: The Political Economy of Music*. Volume 16 of *Theory and History of Literature*. University of Minnesota Press (1985)
 3. Brown, A.R.: Opportunities for evolutionary music composition. In: *Australasian Computer Music Conference*. (2002) 27–34
 4. Raidl, G.R., Cagnoni, S., Branke, J., Corne, D., Drechsler, R., Jin, Y., Johnson, C.G., Machado, P., Marchiori, E., Rothlauf, F., Smith, G.D., Squillero, G.: *Applications Of Evolutionary Computing: Evoworkshops 2004: Evobio, Evocomnet, Evohot, Evoiasp, Evomusart, and Evostoc, Coimbra, Portugal, April 5-7, 2004, Proceedings (Lecture Notes in Computer Science)*. Springer (2004)
 5. Burton, A.R., Vladimirova, T.R.: Generation of musical sequences with genetic techniques. *Computer Music Journal* **23** (1999) 59–73
 6. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional (1989)
 7. Miranda, E.R., Biles, J.A.: *Evolutionary Computer Music*. Springer, Secaucus, NJ, USA (2007)
 8. Todd, P.M., Werner, G.: Frankensteinian approaches to evolutionary music composition. In Griffith, N., Todd, P.M., eds.: *Musical Networks: Parallel Distributed Perception and Performance*. MIT Press/Bradford Books, Cambridge, MA (1999) 313–339
 9. Towsey, M., Brown, A., Wright, S., Diederich, J.: Towards melodic extension using genetic algorithms. *Educational Technology and Society* **4** (2001) 54–65
 10. Dahlstedt, P.: Living melodies: Coevolution of sonic communication. *Leonardo* **34** (2001) 243–248
 11. Werner, G., Todd, P.: Too many love songs: Sexual selection and the evolution of communication. In Husbands, P., Harvey, I., eds.: *Fourth European Conference on Artificial Life (ECAL97)*, Cambridge, MA, MIT Press (1997) 434–443
 12. Eno, B.: Generative music. Transcript of a talk. in *Motion Magazine* (1996)
 13. Lerdahl, F.: *Tonal Pitch Space*. Oxford University Press (2001)
 14. Fink, R.: *On the Origin of Music – An Integrated Overview of the Origin and Evolution of Music*. Greenwich (2003)
 15. Hawkins, J., Blakeslee, S.: *On Intelligence*. Times Books (2004)
 16. Ventrella, J.: Java applet for the musical gene pool. Online at: <http://www.ventrella.com/EvoMusic/EvoMusic.html> (2006)
 17. Gleick, J.: *Chaos – Making a New Science*. Penguin Books (1987)