On the Music of Emergent Behavior: What Can Evolutionary Computation Bring to the Musician?

Eduardo Reck Miranda

ABSTRACT

In this article, the author focuses on issues concerning musical composition practices in which emergent behavior is used to generate musical material, musical form or both. The author gives special attention to the potential of cellular automata and adaptive imitation games for music-making. The article begins by presenting two case-study systems, followed by an assessment of their role in the composition of a number of pieces. It then continues with a discussion in which the author suggests that adaptive imitation games may hold the key to fostering more effective links between evolutionary computation paradigms and creative musical processes.

In this article, I forgo the discussion of whether or not computers can compose music, seeing it as no longer relevant: computers *can* compose, if programmed appropriately. Perhaps one of the greatest achievements of artificial intelligence (AI) to date lies in the construction of machines that can compose music of high quality, such as the Experiments in Musical Intelligence (EMI) system [1]. However, these AI systems are either hard-wired to compose in a certain style or able to learn how to imitate a style by looking at patterns in a group of training examples; they do not create new musical styles, so to speak. Conversely, issues of whether computers can create new kinds of music are much more difficult to study because, in order to test this idea, the computer should neither be embedded with particular models at the outset nor learn from carefully selected examples. Furthermore, it is difficult to judge what the computer creates in such circumstances, because the results normally sound very strange to us; they tend to lack the cultural references that we normally rely on when appreciating music.

One plausible approach to these problems is to program the computer with abstract models that embody our understanding of the dynamics of certain compositional processes. Since the invention of the computer, many composers have tried out mathematical models thought to embody musical composition processes, such as combinatorial systems [2], stochastic models [3] and fractals [4]. Some of these trials have produced interesting music and much has been learned about using mathematical formalisms and computer models in composition. The potential of evolutionary computation is, therefore, a natural progression for computer music research. In this article, I introduce a discussion on the potential of cellular automata and adaptive games for the composition of truly "new" music (albeit music that has the potential to make sense to our ears). The article begins with a brief introduction to CAMUS (Cellular Automata MUSic) and Chaosynth, two cellular automata–based music systems of my own design, followed by an assessment of their role in the composition of a number of musical pieces.

CASE STUDIES

CAMUS uses two simultaneous cellular automata (CA) to generate musical forms: the Game of Life (Fig. 1) and Demon Cyclic Space. Due to limitations of space, I will briefly introduce only the role of the Game of Life in the generative process [5].

Propagating Cellular Automata Musical Forms

The Game of Life is a two-dimensional CA that attempts to model a colony of simple virtual organisms. In theory, the automaton is defined on an infinite square lattice. For practical purposes, however, it is normally defined as consisting of a finite $m \times n$ array of cells, each of which can be in one of two possible states: alive, represented by the number one, or dead, represented by the number zero. On the computer screen, living cells are colored as black and dead cells are colored as white (see Fig. 1). The state of the cells as time progresses is determined by the state of the eight nearest neighboring cells. There are essentially four rules that determine the fate of the cells at the next tick of the clock:

1. **Birth:** A cell that is dead at time t becomes alive at time $t + 1$ if exactly three of its neighbors are alive at time t
2. **Death by overcrowding**: A cell that is alive at time t will die at time $t + 1$ if four or more of its neighbors are alive at time t
3. **Death by exposure**: A cell that is alive at time t will die at time $t + 1$ if it has one or zero live neighbors at time t
4. **Survival**: A cell that is alive at time t will remain alive at time $t + 1$ only if it has either two or three live neighbors at time t

Whilst the environment, represented as E, is defined as the number of living neighbors that surround a particular live cell, a fertility coefficient, represented as F, is defined as the number of living neighbors that surround a particular dead cell. Note that both the environment and fertility vary from cell to cell and indeed from time to time as the automaton evolves. In this case, the life of a currently living cell is preserved whenever $2 \leqslant E \leqslant 3$, and a currently dead cell will be reborn whenever $F = 3$. Clearly, a number of alternative rules can be set. The general form for such rules is (E_{min}, E_{max}, F_{min} and F_{max}) where $E_{min} \leqslant E \leqslant E_{max}$ and $F_{min} \leqslant F \leqslant F_{max}$.

Eduardo Reck Miranda (composer and research scientist), School of Computing, University of Plymouth, Drake Circus, Plymouth PL4 8AA, U.K. E-mail: <E.Miranda@plymouth.ac.uk>.

An earlier version of this paper was presented at the 2000 Genetic and Evolutionary Computation Conference (GECCO-2000) in Las Vegas, Nevada, 8–12 July 2000. The conference proceedings: Darrell Whitley, David Goldberg, Erick Cantu-Paz, Lee Spector, Ian Parmee and Hans-Georg Beyer, eds., *Proceedings of the 2000 Genetic and Evolutionary Computation Conference* (San Francisco, CA: Morgan Kaufmann, 2000).

© 2003 ISAST

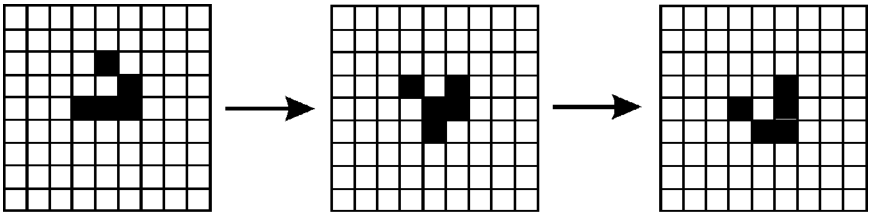


Fig. 1. Game of Life cellular automaton in action. The rules of the automaton are applied simultaneously to all cells on the grid at the tick t of an imaginary clock. (© Eduardo Reck Miranda)

The CAMUS implementation of the Game of Life algorithm enables the user to design rules beyond Conway's original rule. Rules other than (2, 3, 3, 3) may exist, but not all of them produce interesting emergent behavior.

CAMUS uses a Cartesian model in order to represent a triplet of notes, that is, an ordered set of three notes. In this context, a triplet is an ordered set of three notes that may or may not sound simultaneously. These three notes are defined in terms of the distances between them—or intervals, in music jargon. The horizontal coordinate of the model represents the first interval of the triplet, and the vertical coordinate represents its second interval (see Fig. 2a).

To begin the musical generation process, the grid of cells is set up with an initial random configuration and allowed to run. When the Game of Life automaton arrives at a live cell, its coordinates are taken to estimate the triplet from a given lowest reference note; in the case of Figure 1a, the given reference note is *F1*. For example, the cell at the position (5, 5) in Figure 2b is alive (that is, its color is black) and will thus generate a triplet of notes. The coordinates (5, 5) describe the intervals of the triplet: a fundamental note is given, then the next note will be at 5 semitones above the fundamental and the last note 10 semitones above the fundamental. Although cell updates occur at each time step in parallel, CAMUS plays the live cells column by column, from top to bottom. Each of these musical cells has its own timing, but the notes within a cell can be of different lengths and can be triggered at different times. Once the triplet of notes for each cell has been determined, the states of the neighboring cells in the Game of Life are used to calculate a timing template, according to a set of temporal codes [6].

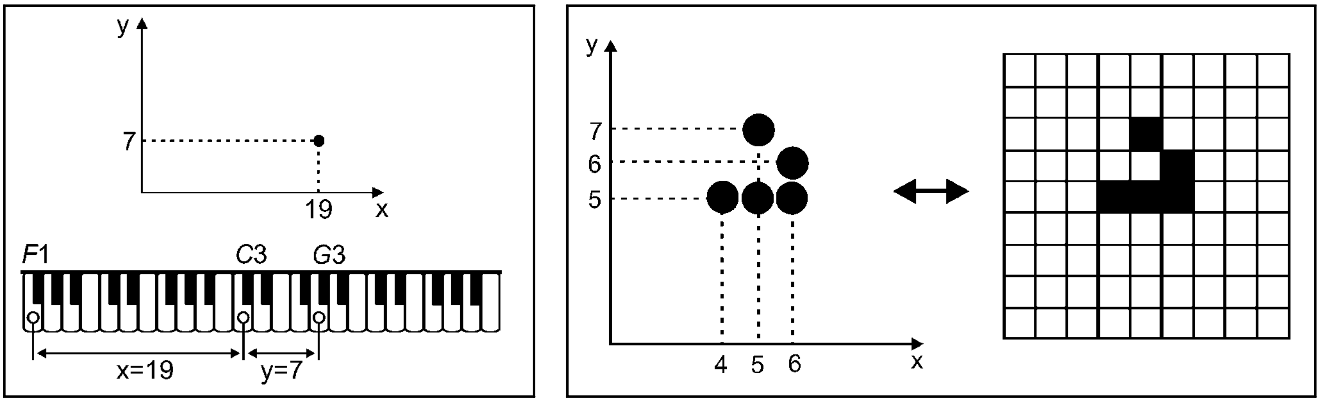
Emergent Sound Synthesis

Chaosynth is essentially a granular synthesizer [7]. Granular synthesis works by generating a rapid succession of very short sound bursts (for example, 35 ms long) called granules that together form larger sound events. The results tend to exhibit a great sense of movement and sound flow. This synthesis technique can be metaphorically compared with the functioning of a motion picture in which an impression of continuous movement is produced by displaying a sequence of slightly different images at a rate above the scanning capability of the eye. So far, most of these systems have used stochasticity to control the production of the granules; for example, to control the waveform and the duration of the individual granules. Chaosynth uses a different method: that of cellular automata. The automaton used in Chaosynth tends to evolve from an initial random distribution of cells in the grid towards an oscillatory cycle of patterns (see Fig. 3) [8]. The behavior of the CA resembles the way in which most of the natural sounds produced by some acoustic instruments evolve: they tend to converge from a wide distribution of their partials (for example, noise) to oscillatory patterns; an example would be a sustained tone.

Each sound granule produced by Chaosynth is composed of several spectral components. Each component is a waveform produced by a digital oscillator (for example, a lookup sampling table containing one cycle of a waveform), which needs two parameters to function: frequency and amplitude. The automaton controls the frequency and duration values of each granule (the amplitude values are set up via another procedure). The values (i.e. the colors) of the cells are keyed to frequencies and oscillators are keyed to a number of cells (see Fig. 4). The frequencies of the components of a granule at time t are established by the arithmetic mean of the frequencies associated with the values of the cells associated with the respective oscillators (Fig. 5). Suppose, for example, that each oscillator is associated with nine cells and that at a certain time t, three cells correspond to 110 Hz, two to 220 Hz and the other four to 880 Hz. In this case, the mean frequency value for this oscillator at time t will be 476.66 Hz.

The user can also specify the dimension of the grid, the number of oscillators, the allocation of cells to oscillators, the allocation of frequencies to CA values and various other CA-related parameters. The duration of a whole sound

Fig. 2. (a, left) CAMUS uses a Cartesian model in order to represent a triplet of notes. Given a fundamental note (e.g. *F1*), the x-coordinate establishes the distance to the second note and the y-coordinate establishes the distance between the second and the third notes of the triplet. These distances are calculated in terms of musical semi-tones. (b, right) Each frame of the Game of Life automaton produces a number of triplets of notes. A frame here is a matrix of cells at a certain time. For example, a sequence of three frames is shown in Fig. 1. (© Eduardo Reck Miranda)



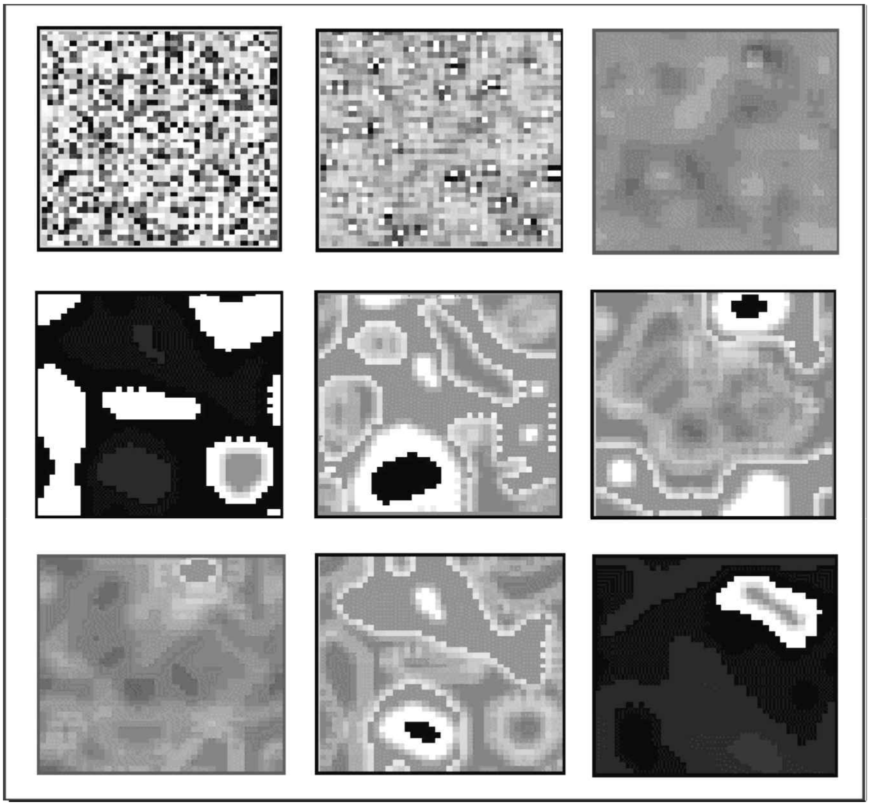
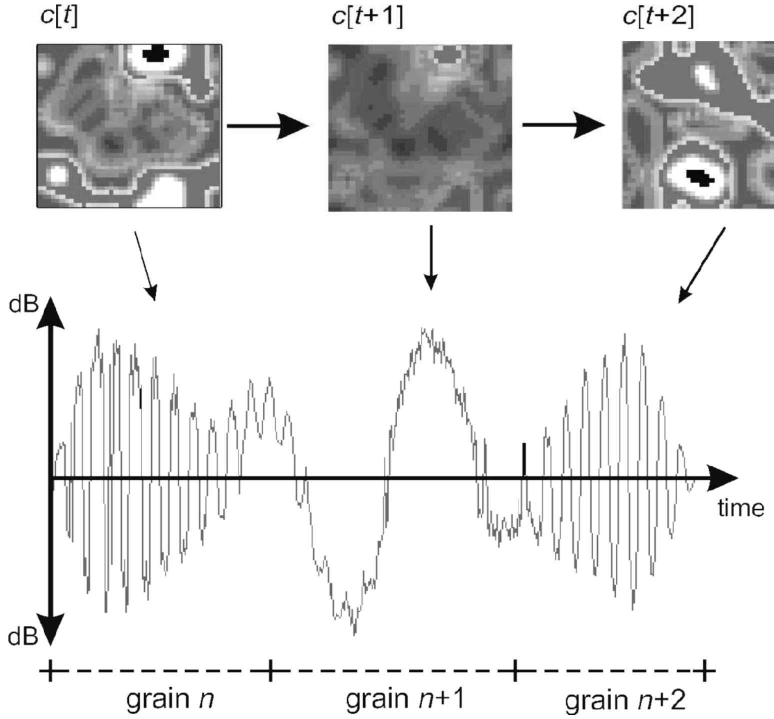


Fig. 3. Chaosynth's automaton tends to evolve from an initial random distribution of cells in the grid (top frame on the left side) towards an oscillatory cycle of patterns. (© Eduardo Reck Miranda)

Fig. 4. Each frame of the automaton, or configuration *c* of the cell at a certain time *t*, *c*[*t*], defines the spectral content of a sound grain. (© Eduardo Reck Miranda)



event is determined by the number of CA iterations and the duration of the granules; for example, 100 iterations of 35-ms granules results in a sound event of 3.5 sec in duration.

Brief Assessment

Despite the arbitrariness of the musical engine of CAMUS, I have come to conclude, through observing that CAMUS can produce interesting musical sequences, that cellular automata are appropriate for generating musical material. Indeed, I have composed a number of pieces using CAMUS-generated material—for example, *Entre o Absurdo e o Mistério* [9], for chamber orchestra, and the second movement of the string quartet *Wee Batucada Scotica* [10]. Chaosynth also has proved to be a successful system in the sense that it is able to synthesize a large number of unusual sounds that are normally not found in the real acoustic world, but that nonetheless are pleasing to the ear [11]. The results of the CA experiments are very encouraging, as they are good evidence that both musical sounds and abstract musical forms might indeed share similar organizational principles with cellular automata. I would like to make it clear, however, that none of the pieces cited above were *entirely* automatically generated by the computer. The programs produced raw but high-level materials that were *manually* arranged for the final composition. Nevertheless, the computer-generated material was of good quality and, as far as computer music is concerned, this is a great achievement. Without CAMUS and Chaosynth, these pieces would probably never have been composed.

DISCUSSION AND CONCLUSION

In general, I found that Chaosynth produced more interesting results than CAMUS did. This might be due to the very nature of the phenomena in question. The inner structures of sounds seem more amenable to CA modeling than do large musical structures. As music is primarily a cultural phenomenon, in the case of CAMUS I suspect that we would need to add generative models that take into account the dynamics of social formation and cultural evolution. In that case, we should look for modeling paradigms where phenomena (in our case, musical processes and forms) can emerge autonomously. We have strong evidence that the adaptive games paradigm might shed some light on this problem.

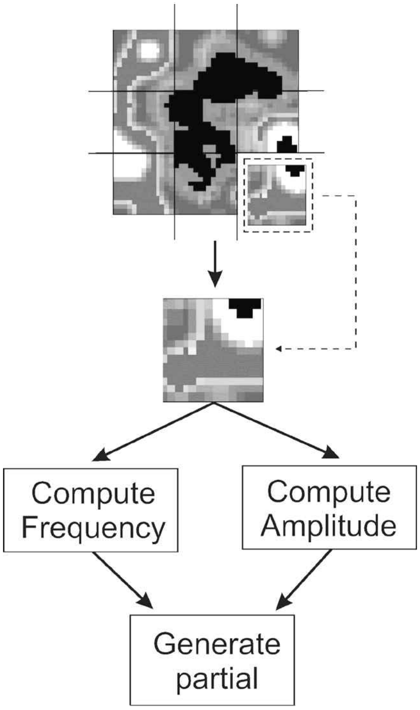


Fig. 5. One oscillator is allocated to each sub-grid of the automaton, and each oscillator generates one of the partials that will compose a grain. The frequency and amplitude for these oscillators are calculated by averaging the values that are associated with the colors of the cells. These associations are user-specified. (© Eduardo Reck Miranda)

I have recently implemented two experiments whereby I have successfully simulated the emergence of sounds and melodic forms in a virtual community of agents, by means of adaptive imitation games [12].

The agents are furnished with a speech synthesizer, an artificial ear and a memory mechanism. To begin with, the agents should not have any repertoire in their memory. The objective of the simulation is to let them build their own repertoire of sounds and/or short melodies by imitating each other. The only constraints of the model are imposed by the physiology of their speech and hearing apparatus and of course by the rules of the interactions. The interactions occur as follows: two agents are selected for a round; one plays the role of a producer and the other plays the role of a listener. In the case of melodies, the producer sings a melody either selected from its repertoire or produced from scratch if there is nothing in the repertoire. The imitator in turn compares this melody with the ones it already knows. The imitator selects from its own repertoire the melody that sounds most similar to the melody that it heard, and then sings it. The producer compares the imitator's attempt with the melody that it had produced originally. If they are similar, then the interaction is a success; otherwise it is a failure. This result is then communicated to the imitator by means of non-verbal feedback. Immediately after the interaction both agents update their memories according to the result of the interaction. In short, if the interaction was successful, then both agents increase a success counter for the melody that was used by both agents. If the interaction was a failure, then the imitator tries to shift the failed melody closer to the melody it heard, hoping that next time it will result in a better imitation. Sometimes melodies that have not scored successfully for a long time are deleted from the repertoire. In other cases, melodies that are too close to each other in the repertoire are merged. Despite the simplicity of the model, the results of the simulations so far are quite impressive. After a few thousand interactions, involving a few dozen agents, we observed the emergence of a coherent repertoire of sounds and melodies that were shared by all agents. We would say that in this case the agents reached a cultural agreement. I am currently scaling up this experiment in order to investigate whether such emergent phenomena also produce coherent results in situations involving high-level musical forms.

Should such experiments with adaptive games corroborate my hypothesis that we can improve algorithmic composition systems considerably by including mechanisms that take into account the dynamics of cultural evolution and social interaction, then a new generation of greatly improved intelligent composing systems may begin to appear over the next few years.

References and Notes

1. D. Cope, *Computers and Musical Style* (Oxford, U.K.: Oxford Univ. Press, 1991).

2. C. Dodge and T. Jerse, *Computer Music: Synthesis, Composition and Performance* (New York: Schirmer Books, 1985).

3. I. Xenakis, *Formalized Music* (Bloomington, IN: Indiana Univ. Press, 1971).

4. D. Worral, "Studies in Metamusical Methods for Sound Image and Composition," in *Organised Sound* **1**, No. 3 (1996).

5. For more information on CAMUS and the CA algorithms, see K. McAlpine, E.R. Miranda and S. Hoggar, "Making Music with Algorithms: A Case-Study System," in *Computer Music Journal* **23,** No. 2 (1999) and E.R. Miranda, "Music Composition Using Cellular Automata," *Languages of Design* **2** (1994).

6. See McAlpine et al. [5]; Miranda [5].

7. E.R. Miranda, *Computer Sound Synthesis for the Electronic Musician* (Oxford, U.K.: Focal Press, 1998).

8. For more details about this CA algorithm, see E.R. Miranda, "Granular Synthesis of Sounds by Means of a Cellular Automaton," in *Leonardo* **28,** No. 4, 297–300 (1995); E.R. Miranda, "On the Origins and Evolution of Music in Virtual Worlds," in P. Bentley and D. Corne, eds., *Creative Evolutionary Systems* (San Francisco: Morgan Kaufmann, 2001).

9. E.R. Miranda, *Entre o Absurdo e o Mistério* (Porto Alegre: Goldberg Edições Musicais, 2001) (full score including parts).

10. E.R. Miranda, *Wee Batucada Scotica* (Porto Alegre: Goldberg Edições Musicais, 1998) (full score including parts).

11. *Olivine Trees,* an electroacoustic piece composed using Chaosynth, was awarded the bronze medal at the International Luigi Russolo Electroacoustic Music Competition in Italy, 1998 (the recordings of all these pieces are available by request).

12. For up-to-date reports on these experiments, see E.R. Miranda, "Emergent Sound Repertoires in Virtual Societies," in Computer Music Journal **26**, No. 2 (2002) and E.R. Miranda, "Mimetic Development of Intonation," in Proceedings of the 2nd International Conference on Music and Artificial Intelligence, Springer-Verlag—Lecture Notes on Artificial Intelligence (2002).

Glossary

adaptive games—this term relates to systems that evolve and adapt over time. Psychology, computer science, economics, biology, and neuroscience depend upon a deeper understanding of the mechanisms that govern adaptive systems. A common feature of these systems is that organized behavior emerges from the interactions of many simple parts. Ants organize to build colonies, neurons organize to produce adaptive human behavior, and businesses organize to create economies.

cellular automata—a regular spatial lattice of cells, each of which can have any one of a finite number of states. These cells normally are numerical variables, but in theory they could represent anything that can be computed. The state of all cells in the lattice is updated simultaneously, and the state of the entire lattice advances in discrete time steps. The state of each cell in the lattice is updated according to a local rule, which often depends on the state of the cell and its neighbors at the previous time step.

combinatorial systems—systems based upon combinatorial mathematics, a branch of mathematics studying the enumeration, combination and permutation of sets of elements and the mathematical relations that characterize these properties.

emergent behavior—group behavior resulting from the interaction of the group's many elements. The behavior of the individual elements is much simpler than the behavior that emerges from their interactions.

granular synthesizer—a sound synthesizer that uses the granular synthesis techniques. Granular synthesis works by generating a rapid succession of very short sound bursts called granules (e.g. 35 milliseconds long) that together form larger sound events. This technique can be metaphorically compared to the functioning of a motion picture, in which an impression of continuous movement is produced by displaying a sequence of slightly different images at a rate above the scanning capability of the eye.

stochastic models—models relating to, or governed by, probability. The behavior of a probabilistic model cannot be predicted exactly, but the probability of certain behaviors is known. Such systems are usually simulated using a pseudo-random number generator, where one of a sequence of numbers is generated by some algorithm so as to have an even distribution over some range of values and minimal correlation between successive values.

Eduardo Reck Miranda graduated in computing at Universidade do Vale do Rio dos Sinos, São Leopoldo, Brazil, and went on to study music, with a focus on composition, at Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil. He has an MSc in Music Technology from the University of York, U.K., and a Ph.D. in Music from the University of Edinburgh, U.K. Having previously held academic and research positions in Scotland and France, he is currently a Reader in Artificial Intelligence and Music at the School of Computing of the University of Plymouth, U.K. Miranda has published a number of books and research papers in major international journals, and his compositions have been performed worldwide. He is a member of the editorial board of Leonardo Music Journal, Organised Sound and Contemporary Music Review.

FORTHCOMING

Leonardo Music Journal Volume 13

Groove, Pit and Wave—Recording, Transmission and Music
Publication Date: December 2003

Despite Thomas Edison's assumption that the gramophone was nothing more than a sonic autograph album, suitable only for playing back the speeches of famous people, over the last 100 years recording has radically transformed the composition, dissemination and consumption of music. Similarly, the business-like dots & dashes of Morse and Marconi have evolved into a music-laden web of radio masts, dishes, satellites, cables and servers. Sound is encoded in grooves on vinyl, particles on tape and pits in plastic; it travels as acoustic pressure, electromagnetic waves and pulses of light.

The rise of the DJ in the last two decades has signaled the arrival of the medium as the instrument—the crowning achievement of a generation for whom tapping the remote control is as instinctive as tapping two sticks together. Turntables, CD players, radios, tape recorders (and their digital emulations) are *played,* not merely heard; scratching, groove noise, CD glitches, tape hiss and radio interference are the sound of music, not sound effects. John Cage's 1960 "Cartridge Music" has yet to enter the charts, but its sounds are growing more familiar.

For *Leonardo Music Journal* Vol. 13 we consider the role of recording and/or transmission in the creation, performance and distribution of music.

For more information, visit <http://mitpress.mit.edu/Leonardo/lmj>.