INTERVIEW

Composing with Shifting Sand: A Conversation between Ron Kuivila and David Behrman on Electronic Music and the Ephemerality of Technology

Ron Kuivila and David Behrman

ABSTRACT

This dialogue between composers David Behrman and Ron Kuivila centers on the ephemerality of technology and technological innovation in musical practice over the last 40 years. The conversation focuses on a musical lineage that begins with the early, live electronic music of John Cage and D.B. Tudor, the projects of Experiments in Art and Technology (EAT), the work of the Sonic Arts Union and the San Francisco Tape Music Center.

A new generation of artists whose work shared the same sensibility, but using very different kinds of technology, began to emerge in the mid- and late 1970s, when a loose-knit collective of composers and students associated with the Center for Contemporary Music at Mills College (Oakland, California) began to use microcomputers in live electronic music. With the introduction of MIDI control in the early 1980s, many of the people involved in this work began to create shareware software projects, such as Formula, HMSL, and MOXIE, that sought to provide a more stable basis for live electronic music with computers. Since then, the arborescence of activity has made it impossible to establish any clear lineage.

THE AUTHORS

David Behrman has been active as a composer and electronic artist since the 1960s and has created many works for performance as well as sound installations. Most of his work since the late 1970s has involved computer-controlled music systems operating interactively with people who may or may not be musically expert. He designs and writes much of the software for these systems. *Unforeseen Events, My Dear Siegfried, . . . , QSRL, Leapday Night, Interspecies Smalltalk, A Traveller's Dream Journal, Figure in a Clearing* and *On the Other Ocean* are among Behrman's software-based works for soloists and small ensembles that have been performed by a number of musicians since the mid-1970s.

Ron Kuivila became active as a composer using homemade and home-modified electronic instruments in the 1970s. In his early work, he pioneered the use of ultrasound (*In Appreciation*) and sound sampling (*Alphabet*) in live performance. Other pieces have explored compositional algorithms (*Loose Canons*), speech synthesis (*The Linear Predictive Zoo*) and high-voltage phenomena (*Pythagorean Puppet Theatre*). Most recently, his pieces have recalled the sound world of live electronics while exploiting the compositional possibilities of digital signal processing in *fugue states.*

A CONVERSATION

David Behrman: Those of us working with technology are often fascinated by devices or techniques that are (a) new and (b) inexpensive, and a result of this is that works tend to be very involved with things that may appear in the marketplace one decade and be gone the next.

Ron Kuivila: The ephemerality of technology is two-fold—a technology can become unavailable or just horribly banal. It seems to me that there are three ways people avoid this: by getting "under" technology, by working directly with physical principles; by staying "over" technology, by working with abstract principles; or by diving "into" obsolete or banal technologies.

D.B.: Your installation *Parallel Lines* depends for its realization on parallel wires and arcing sparks traveling along them. I imagine this piece would have fascinated people living in the era of Tesla and might have been created by an artist having only Tesla's technology to rely on—at least it comes across that way! The piece seems to me so viscerally and universally "electric" that it is immune to technological obsolescence and will probably continue to appeal to people for a long time into the future.

R.K.: Physical principles do not change, so pieces based on natural phenomena never go out of date (even if they may go out of fashion). Ben Franklin organized a tour for two Italian "electricians" who gave theatrical demonstrations of the wonders of electricity. Two hundred years later, in Italy, I made pieces working with the same phenomena. This is what I mean by working "under" technology.

Ron Kuivila (composer, teacher), Music Department, Wesleyan University, Middletown, CT 06459, U.S.A. E-mail: <rkuivila@wesleyan.edu>.

David Behrman (composer), 10 Beach Street, New York, NY 10013, U.S.A. E-mail: <behrbase@compuserve.com>.

D.B.: At a few moments over the last 40 years, I've had the feeling that a great door had just swung open, inviting artists in to new ways of exploring music possibilities. The first time was in the early 1960s, at the moment when transistors first became available. One day it became possible for the first time to use small, lightweight, inexpensive devices to radically alter or hugely amplify acoustic sounds. John Cage, D.B. Tudor and, a bit later, Gordon Mumma were, for me, the masters from whom I could learn to work with these new possibilities.

R.K.: I think the idea that an electronic configuration can create an identifiable composition while being entirely open to the performer's own choices was very important. In music, abstraction through notation has been the preferred approach since the time of Guido d'Arezzo, the ninth-century monk who devised a precise pitch notation that enabled sight singing of melodies. For Guido, the introduction of notation allowed melodies to be resurrected from the page rather than passed on like fire. Accepting and even championing an unstable relationship between a notation and its resulting sound seems to be an artifact of the age of recording. Pieces of John Cage's such as *Cartridge Music* or the *Variations* series do not describe a fixed temporal architecture; rather they provide some basic tools with which the performer creates a realization. These pieces are more like recipe books for the practice of music. In this context, it was quite natural to begin to consider the time-based behavior of an electronic configuration as the identity of a musical composition. Instead of a recipe, the configuration presents a situation within which the performer is free to act without moment-by-moment directions from the composer. Having defined the situation, the composer can allow the performer free rein without worrying too much about the identity of the piece. This is an example of staying "above" the technology—conceiving of music as a practice rather than a collection of sound objects allows one to adapt to new technological situations and to describe a work "tactically" rather than "literally." You likened this to surfing, where everything done with a surfboard in the surf is a part of surfing. Of course, not everyone is an equally accomplished surfer. Pieces of this form create an entirely new set of demands on the performer. They require a kind of virtuosity of sensibility. Cage's own way of dealing with these new demands was to say that the works of this sort that he composed were really conceived for David Tudor. Cage is quite clear that *Cheap Imitation* marked the end of this approach in his own work.

Often, new technologies at first appear to have some kind of redemptive potential, only to give way ultimately to the tiring familiarity of ubiquity. John Bischoff has described sounds produced by FM synthesis as "beginning with such promise and ending with disappointment." Part of that disappointment is acoustic, part of it is social. The lovely and remarkable works (such as John Chowning's *Stria*) that pioneered the use of frequency modulation as a synthesis technique impart a sense of discovery. As those aural experiences are repeated in later works, their familiarity becomes increasingly tiring and, ultimately, banal. The Web seems to have managed this kind of transition in fewer than two years.

For me, the "live electronic music" that D.B. Tudor and John Cage pioneered in the 1960s remains too indigestible to become banal—that is the power of chance as a disciplinary principle in musical practice.

A very different, but equally effective, approach is to use totally digested technologies, those that are out of date or so commonplace as to be banal. The machines are cheaper, more accessible, easier to manipulate and often carry much greater social resonance than "high-tech" equipment. Tim Perkis's use of a mouse as a virtual guitar pick and a gesturally oriented controller is a wonderful example. Theatrically, his "flamenco mousing" acts as a wonderful antidote to "point and click." His constructive misuse of a Microsoft mouse helps extract sounds out of MIDI synthesizers that have never been heard, and it only costs 40 bucks. It is this kind of invention of quirky technological alternatives that I describe as working "in" the technology. I believe it is in this spirit that D.B. Tudor named his ensemble of associated composers "Composers Inside Electronics."

D.B.: There is a paradox in the legacy of D.B. Tudor: the wonderful quality of his work in electronic music was due in part to his use of quirky, homemade circuitry, the inner workings of which he was slow to divulge to his assistants and colleagues. Yet that quirkiness, which made the music so good, also made it evanescent. It could only exist for a few years before being swept away by the torrent of technological change.

You and I have both experienced how fresh and striking that music was when it was new. And then, an instant later, seemingly—25 years having gone by and Tudor having passed away—we've had the heartbreaking experience of trying to understand his no-longer-working, unlabeled circuitry and of coming to the realization that there was no way to revive that music in a literal sense. The interesting thing is that you, Ron, have figured out a different way to revive it. What have you learned in the course of following this path—writing software to emulate Tudor's hand-wired circuitry and then presenting performances with your students and colleagues?

R.K.: I have made digital simulations of Tudor's phase shifters, ring modulators, filters and envelope followers, both as a practical step in reconstructing some of his work and as an act of friendly heresy (Tudor never liked computers). Digital simulations are also algorithms that, in principle, could be realized with whatever computing resources become available. So they are, in some sense, "notations."

However, in Tudor's work (and in live electronics in general), the instability of the electronics, the absence of presets, and the viewpoint that configuration defines the identity of the composition combine to make it very difficult to distinguish performance from composition. Tudor's approach systematically prevented making any such distinction or any other attempt at rationalization. What is important and what is happenstance? It is impossible to tell, and it might change tomorrow.

My own approach has taken two parallel paths. One has been to actually assemble realizations of some of Tudor's pieces. My goal is not to reconstruct the continuity of his pieces—that is inextricably intertwined with his own unique sensiblity. Instead, it is to reconstruct the "moves"—the set of musical questions—that the pieces create. Tudor's music is to be practiced, not preserved.

The other stream has been to apply my understanding of those questions to the very different set of possibilities created by digital electronics. But computers excel at creating musical preserves (presets and samples) that work perfectly or not at all. So, part of my goal has been to create digital situations that can "fail" musically and gracefully. In my composition *fugue states,* this is done by combining the sound world of live electronics with the digital process of "morphing."

fugue states exists as a set of 32 preset tunings of a digital simulation of a live electronic configuration. The configuration is derived from the processor Gor-

don Mumma designed for EAT's (Experiments in Art and Technology) Pavilion at Expo 1970. (This processor is completely documented in the Dutton paperback, *Pavilion* [1].) Mumma's processor has no fewer than 16 different knobs per channel and eight channels. It is not possible for one person to turn 64 physical knobs at the same time, but it is trivial for a computer to change 64 different values more or less simultaneously; so the 32 preset tunings serve as "fixed stars." The computer can adjust all 64 parameters of the configuration to a weighted average of these fixed stars. The performer performs the piece by "navigating" between these tunings. This is done by setting a "starting point" and a "target point" and smoothly interpolating between the tuning of the two points. The performer can select a new target and "change directions" at any time. So the 32 presets serve as "flavors" as much as "destinations." In principle, it is possible to perform the piece entirely differently each time it is performed. In this way, it attempts to recover both the sound world and the irrationality of live electronics in the digital domain. I see this as practicing the principles underlying Tudor's (and Cage's) music.

But all of this is just one particular train of thought. Microcomputers originally represented a way out of the habits of live electronics rather than a way to revive them. David, your own *Melody Driven Electronics* anticipated the possibilities of microcomputers. With the help of Jim Horton, you created what I believe to have been the first composition that used a microcomputer in concert. This experimentation with microcomputers was taken on communally by composers associated with Mills College, in Oakland, California, as faculty, students or simply as users of their publicly accessible studios. In fact, Jim Horton was a philosophy graduate student who was so inspired by Cage's work that he left the University of Minnesota for Mills College in order to gain access to electronic music instrumentation.

D.B.: Yes, the second time that great inviting door seemed to swing open was around 1976–1978, with the advent of inexpensive microcomputers. You and I were both in the San Francisco Bay Area around that time, at Mills College in Oakland, and we both experienced that moment at first hand. It was in California where this new technology first appeared, and the information about how to handle it was all around us. Looking back on it, that was a Utopian moment: the idea that artists could use this thing called software was fresh and astonishing, and commercialization was a million miles away from our thoughts. It was a moment to dream about new kinds of art.

Some of the music made at that time was very special. There was a Bay Area composers' group of that era, the Computer Network Band, originally formed by John Bischoff, Jim Horton and Rich Gold, which liked to perform concerts in which the participants would wire together a group of computers on a table, turn them all on, and stand back and watch to see what would happen.

Ron, your own contribution to that moment was as the co-author of the music language called Formula, which has attracted a small but intense circle of enthusiastic users over the past 12 years. I am a member of that circle myself; almost all of my work since 1992 is written in Formula. In a sense, all of the works that various composers have made using Formula are collaborations with you and D.B. Anderson (Formula's other co-author) and also, one could say, with Mitch Bradley, whose version of the general-purpose language Forth lies beneath your music language.

There are some things I wanted to ask you about Formula. One is, how do you feel about the collaborations that have resulted from your making a language that artists want to use, and do you think of this as a fundamentally new kind of collaboration?

Another is, how do we deal with the fact that computer users, as though riding a vast wave powered by tens of thousands of new minds, have left behind the 68000 family of processors upon which the language was built? Might Formula be lost, as were the old circuit boards of D.B. Tudor?

R.K.: I believe that Formula and the many other music programming environments (MOXIE, moxc, MAX, etc.) are really a continuation of age-old collaborative processes in music. One difference is that the programming environments do not have the same specific stylistic implications that other kinds of techniques (tone clusters, piano preparation, common practice harmony) have. I certainly expect that, while many of the ideas in Formula are and will be found in other music programming environments, the program itself will eventually disappear.

The problem is that there is no guarantee that a satisfactory replacement will emerge in a timely manner. The problem is much the same as the problem in reconstructing Tudor's work. The culture does not always retain the tools essential to carrying on a certain kind of music. For example, the phonograph cartridges that worked so well as contact microphones have long since disappeared from the market. Other solutions have emerged—for example, the piezoceramic discs used to create the electronic beeps of microwave ovens and alarm clocks can be reconfigured to act as contact microphones—but these solutions will probably also disappear as time passes. There is no single solution to this quandary. One must improvise solutions as the problems appear and try to identify the best underlying strategies.

While I do not think that using Formula implies creating in a specific style, there do seem to be "family resemblances" between pieces made with particular approaches. For example, many pieces of the live electronic era revolve around multiple copies of the same system, while pieces from the microcomputer era involve more differentiated material and roles. The League of Automatic Music Composers, which evolved out of the Computer Network Band in the early 1980s, makes the difference quite apparent—the networking of the computers was identical, but the musical logic inside each system was totally distinct.

D.B.: In the live electronic pieces I composed in the 1960s and early 1970s (e.g., *Runthrough* and *Sinescreen*), it seemed to make sense to double, triple or quadruple the amount of hardware for use in performances. The sound textures were enriched and several performers could play together. The "scores" consisted of general instructions, rather than of specific commands governing moment-to-moment actions. Inevitably a kind of counterpoint would result as the performers pursued their individual paths while listening to one another. In my software-based pieces of the 1980s and 1990s, roles for performers can be changed from moment to moment; layers can be created with the drag of a mouse rather than through months of soldering and drilling; complex or simple structures can be built over time. Some recent pieces, such as *Unforeseen Events,* use a structural element that could only grow out of the new situation in which sensors are combined with interactive software: the "listening" computer goes to a "monolog mode" when the human performers are silent and to a "dialog mode" when it notices that somebody has entered into the scene by

making sounds. Through the past few decades, in which very different techniques and media have succeeded one another, I've always attempted to set up fluid, interesting situations in which performers are exploring rather than following instructions.

R.K.: One source of confusion with pieces based on exploration is, when are they done? When, if ever, can the piece go on without the composer's direct intervention?

D.B.: Interactive music and sound art seem to require a lot more in the way of trials and test runs than works that are fixed in advance of realization. First, there's the stage of getting the machinery to run smoothly, of being sure that it won't crash during a performance. (In a sense, interactive pieces are like new bicycle designs that need to be tested by real people on real dirt roads.) After the machinery has stabilized, it takes time to see how the piece works with, perhaps, several really good performing musicians (who are usually themselves composers). Does the piece seem to engage their imaginations? Does it come out differently when different artists perform it? Are they puzzled or bored by certain parts of the composition and fond of others?

It took me several years, for instance, to iron out the kinks in the piece called *QSRL*, made for Barbara Held [2] in 1994 and developed collaboratively with her since then. (In a sense it is also a collaboration with you, Ron, as it is written in Formula!) I have performed this piece with a number of fine musicians besides Barbara. I knew for a while that something bothered me about one of its 16 sections, but I couldn't figure out for a long time what exactly it was. One day recently it became clear, and I was able to replace the section with a new one that I'm happy with—and now the piece can flutter away.

R.K.: Often sound installations run without the intervention of a performer, but they may still require the composer to set them up in the first place. Given Maryanne Amacher's [3] incredible ear for architectural acoustics, it is hard to imagine one of her pieces being performed without her direct supervision. On the other hand, Laurie Spiegel's computer program Music Mouse [4] manages to be both an instrument and a piece at the same time. It leaves plenty of room for user intervention and invention while maintaining a clear musical identity of its own.

D.B.: With the pieces designed for live performance in concerts, it has been a goal of mine for a long time to provide my friends with portable systems that they can use on their own without my having to be there. I don't feel a tremendous need to be onstage.

References and Notes

1. *Pavilion*, Billy Kluver, Julie Martin and Barbara Rose, eds. (New York: E.P. Dutton, 1972) p. 7.

2. Barbara Held is a flautist and composer currently based in Barcelona. She has performed premieres of works by such composers as David Behrman, Alvin Lucier and Yasunao Tone.

3. Maryanne Amacher's work uses concrete and synthesized sound to create site specific sound environments that feature detailed attention to the acoustical and visual properties of the site. Her current work is prominently featured in the Sombient catalog of CD recordings.

4. Composer and software designer Laurie Spiegel has been composing with interactive computer music systems since the early GROOVE system was developed at Bell Labs by F. Richard Moore. Her program Music Mouse was a pioneering example of a software application that gives non-musicians the ability to construct and shape complex musical textures and compositions.

Manuscript received 27 February 1998.