

From Piano to Computer
by Jean-Claude Risset

Initially published in: Proceedings of the
ICMC 1990, pp. 15-19; <http://hdl.handle.net/2027/spo.bbp2372.1990.001>

Original PDF with new page layout.

Comment by Cort Lippe

The musical, scientific, and technical brilliance of this work cannot be underestimated. The interactive nature of the studies are examples of human/machine interaction in a highly inventive process-oriented compositional environment.

ABSTRACT. In this presentation, I shall mostly demonstrate what I believe to be the first implementation of a live interaction between a pianist and a computer, accomplished in collaboration with Scott Van Duyne. The computer reacts in real time to the pianist's performance and adds a part on the same acoustic piano. This part depends upon what the pianist plays and how he plays it. The acoustic piano is a Yamaha Disklavier, equipped with MIDI input and output. In order to explore and demonstrate various kinds of interaction, I composed a *Duet for one pianist* comprising eight sections, each of which explores and demonstrates an elementary process of interaction.

From piano to computer to piano: this title corresponds to the flow of events in the live interaction between a pianist and a computer which I shall describe and demonstrate. The pianist plays. MIDI information on his playing is transmitted to a computer. The computer program determines additional musical notes which depend upon what the pianist plays and how he plays: the information about these notes are transmitted to the piano, which plays them. I shall first describe the implementation, its difficulties, the problems which had to be solved. Then I shall describe the processes of interaction explored in my *Duet for one pianist*.

But the title also corresponds to my own itinerary. I began playing the piano and composing for piano-as well as for other instruments. Then I spent much time exploring the musical resources of the computer: here I was helped by my experience as a performer, since it is the responsibility of the composer to introduce performance nuance in his synthetic music. My recent activity with piano-computer interaction, which I was fortunate to start as a composer in residence in M.I.T., put the piano again at the center of my preoccupations.

One may also venture to find other justifications for this title. According to Max Mathews and John Pierce, the idea of using the computer to compose sounds occurred to them while they were attending a piano concert - which, fortunately, did not absorb their attention completely. Later, James Tenney used the computer to produce piano-roll like printouts which he then punched by hands following the printed spots - he could then have them played by player pianos used by Colin Nancarrow. I myself synthesized poor computer copies of piano sounds, I wrote a piece, *Distances*, for piano and computer sounds, and I tried to mimic on the piano the endless descents I had synthesized for *Little Boy*. John Chowning currently writes a piece for pianists playing on MIDI keyboards, taking advantage of the virtuosity to control digital sounds in an atypical and perhaps perverted way. After all, isn't the piano the basic *digital* instrument?

*This presentation reports mostly on work done in 1989 at Music and Cognition, Media Laboratory, M.I.T., while the author was Composer in residence thanks to a grant of the Massachusetts Council of the Arts. The author acknowledges the dedicated and highly competent assistance of Scott Van Duyne, whose work was most helpful and inspiring.

Live interaction on an acoustic piano with MIDI input and output

In 1989, while I was Composer in residence at Music and Cognition, Media Laboratory, M.I.T., thanks to a grant of the Massachusetts Council of the Arts, I worked on the implementation of a live interaction between piano and computer. This implementation was accomplished with the collaboration of Scott Van Duyne. I must pay tribute here to Scott's dedication, competence and musicality, thanks to which the delicate problems we met were promptly solved in an elegant manner.

Of course, there was a tradition there, thanks to Barry Vercoe's work on the "synthetic performer". Barry developed programs to enable the computer to follow a performer playing an instrument such as the flute or the violin, and to add its own part in proper synchrony with the performance. This process was used by Philippe Manoury in his pieces *Jupiter* and *Pluto*, both for instruments and 4X digital processor. Barry Vercoe had his programs control a Yamaha Disklavier to produce a synthetic accompaniment to a live violinist on an acoustic piano.

The Yamaha Disklavier is an acoustic upright piano which looks like an ordinary piano, but which is also equipped with mechanisms allowing it to play by itself. On this piano, each key can be played from the keyboard, but it can also be activated by electrical signals: these signals trigger motors which actually depress or release the keys. Each key also sends out information as to when and how loud it is played. The information to and from the piano is in the MIDI format, used for synthesizers.

The Disklavier allows the pianist to record a sequence of notes: however the recorded sequence is fixed, as in a player piano. In fact, impressive demonstrations had been realized by Mike Hawley on a Bösendorfer Imperial piano equipped with recording and playback devices: however no modification could not be performed in real time. My project was to use the MIDI input and MIDI output to set up a genuine interaction between the pianist and the computer part, such that the part played by the computer could be changed in various ways by what the pianist plays and how he plays. This could be accomplished by a Macintosh computer receiving the latter information and sending back the appropriate MIDI signals to play the additional part: the programming determines in what way the computer part depends upon what the pianist plays.

The desired flexibility in the type of interaction can only be achieved through careful programming. We were privileged to use MAX, a powerful program written by Miller Puckette for making computer music. The program is specially designed to set up real-time interactions, and it is endowed with MIDI input and output capabilities. It includes a "patcher" which permits the user to specify its goal by designing graphical "patches" which specify an appropriate combination of elementary modules.

Problems of live interaction: MIDI feedback

In trying to implement such live interaction between piano and computer, we immediately encountered problems which did not occur when implementing the same kind of live interaction between computer and MIDI keyboards and synthesizers. The worst problem was a MIDI feedback phenomenon. The Disklavier sends out a MIDI message when any note is played on the piano, regardless of whether it originated from the pianist or from a MIDI command

sent by the computer. Thus, trying to echo a note played by the pianist with a note symmetrical with respect to a certain pitch, one would get an oscillation between the two notes. To remedy this problem, Scott Van Duyne built in MAX a "feedback filter" identifying and disregarding the feedback MIDI messages. For each incoming MIDI message, such a filter has to look up tables where the MIDI commands sent by the computer are stored to ascertain whether this message corresponds to a note which had been triggered by the computer. This way, MIDI feedback could be avoided.

Rhythmic skewing

On the Disklavier, the response delay to MIDI note-on commands varies with intensity - an inescapable feature of the acoustic piano, since the intensity is determined by the velocity of the hammer. This time can vary from about 50 milliseconds for loud sounds to about 200 milliseconds for soft sounds. It causes rhythmic skewing: loud notes sound more promptly than soft notes, producing a bizarre warping of the intended musical articulation.

Scott Van Duyne solved this problem by building in MAX an "anti-skewing module" compensating for the variations of the response delay. First, the values of the delay as a function of the velocity were obtained experimentally using a timer module available in MAX. This information could be looked up to determine the transit time corresponding to a given velocity. Then the unevenness in timing could be compensated by subtracting the transit time from a preset delay time to which everything is being aligned, and the out message is placed in a delay line for output at the appropriate time.

This requirement of a preset delay time of about 200 ms is a necessary evil when one requires proper unskewing. It is however necessary to permit a good reproduction of a recorded sequence of notes. It also takes care of a related problem, namely the shortening of notes.

Controlling a process oriented composition

In addition to solving the above problems, Scott Van Duyne developed additional tools in MAX to help developing a composition that would explore and demonstrate certain elementary processes of live interaction. Among the tools developed were pitch trigger with velocity and tempo detectors: these could modify the processes and make them dependant of performance features; sequence editing modules; and "listener/playback" devices which could produce multiple canons from the material played by the pianist, taking advantage of the capacity of MAX to control different voices at different tempos.

Duet for one pianist : a piano piece with various processes of live interaction between computer and performer

I have taken advantage of this live interaction between piano and computer to compose a *Duet for one pianist*, a piano piece with various processes of live interaction between computer and performer. This duet is made up as a set of eight sketches, each of which explores and demonstrates a specific elementary mode of live interaction. Here again, I acknowledge invaluable help from Scott Van Duyne.

The scores were not written out and subsequently implemented in MAX. Instead processes were developed in MAX which would modify and respond to MIDI input from the piano; and the score gradually developed through experimentation with these processes.

The processes implemented in the eight sketches will be demonstrated. They can be briefly described as follows.

Doubles. The pianist plays alone, then on the repeat the computer adds ornaments. These are specified in the patch or prerecorded: they are called when the pianist plays certain notes; their tempo can be influenced by the tempo of the pianist.

Mirrors. Each key played by the pianist is echoed by a key stroke, symmetrical with respect to a certain pitch - a process used in Webern's second *Variation opus 27*, quoted at the beginning (and also at the end with time reversal). The symmetry center and the response delays are changed during the piece to vary the effects.

Extensions. To the arpeggios played by the pianist, the computer adds additional notes transposed in pitch.

Fractals. To each note played, the computer adds five notes spaced approximately - but not exactly - one octave apart. Thus the pitch patterns played by the pianist are distorted in strange ways: an octave jump is heard as a semitone descent.

Stretch. Pitches are added, as in *Extensions*, but the intervals are not merely transposed: they are stretched by a factor ranging between 1.3 and 2.7. This stretches the chords as well as the melodies played by the pianist.

Resonances. At the beginning and the end, the computer plays long sustained chords. In the middle section, the pianist plays mute chords: the strings are set in resonance by the sequences played by the computer.

Up Down. Quasi-octave arpeggios are triggered by the pianist, whose few notes can thus generate many notes. The tempo of the arpeggios is set first by the tempo of certain patterns played by the pianist; later by the pitch he plays; then by the loudness.

Metronomes. This begins by a short canon: the computer echoes the pianist on transposed pitches and at different tempos. It later plays simultaneously different sequences at different tempos. Then it repeats the same pitches, but again at different metronomic tempos, either preset or set by the pianist.

In its present version, the piece requires a Disklavier Yamaha piano or a similar instrument with MIDI input and output, and a Macintosh II computer with a MIDI interface and at least 2 Megabytes of internal memory. The piece has proven to be easily "portable" - it has been performed on both the upright and the Grand version of the Disklavier.

This piece is only a beginning: many more processes of interaction can be developed and exploited. Of course, there is a lot of research to be done on the correlates of performance expressivity: this has already been started, in particular by Johan Sundberg and Carol Palmer. In particular, I did not touch a field which Robert Rowe has tackled, and which is very promising: exploring possibilities of improvisation in collaboration with intelligent programs. At least, my Duet for one pianist demonstrates that the piano can also be an *intelligent instrument*, as Max Mathews says. I must say that it is very rewarding to play: one really feels the computer can be a partner. I certainly find it interesting to extend the possibilities of the computer to the control of acoustic instruments, which I previously enjoyed mixing with computer-synthesized sounds (e.g. in *Dialogues*) or processing by computer (e.g. in *Sud*).

Parting remarks

In this as in many other applications of the computer to music, the challenge is to take advantage of our synergy with the computer to develop novel, valuable and refined possibilities, to use the computer as a tool of personalization and refinement rather than as a depersonalizing machine. I believe this is a challenge to mankind for centuries to come, and not only in the field of music: but computer music must continue to lead the way here and to demonstrate that our interaction with the computer could - should - be helpful and harmonious.