5 Interactivity and live computer music

SERGI JORDÀ

'When asked what musical instrument they play, few computer musicians respond spontaneously with "I play the computer." Why not?'(Wessel and Wright 2002). Actually, most computer music performers still seem shyly reluctant to consider the computer as a *regular* musical instrument, but nonetheless, the computer is finally reaching the point of feeling as much at home on stage as a saxophone or an electric guitar. This assimilation was boosted over the last ten years with the arrival of affordable machines powerful enough for realtime audio processing and of versatile audio software such as Max/MSP, Pure Data or SuperCollider, but live computer music is far from being a novelty; computer-based realtime music systems started in the late 1960s and early 1970s and non-digital live electronic music goes back as far as the nineteenth century.

An ambitious goal for any new instrument is the potential to create a *new kind of music*. In that sense, baroque music cannot be imagined without the advances of sixteenth- and seventeenth-century luthiers, rock could not exist without the electric guitar, and jazz or hip-hop, without the redefinitions of the saxophone and the turntable. This chapter explores the potential of computers as musical instruments and analyses what it is that makes them so especially different from previous instruments, unveiling the novel possibilities as well as the drawbacks they entail.

The computer as a sound-producing device

While acoustic instruments inhabit bounded sound spaces, especially constrained in terms of timbre, tessitura and physical mechanism, computers are theoretically capable of producing any audible sound, either from scratch (through sound synthesis techniques) or by sampling existing sounds and altering them further through processing. For many musicians, this ability to explore an infinite sonic universe, an aspect that will be explored in chapter 11, constitutes the first and most obvious advantage of the computer over traditional instruments. For our purposes, another essential distinction between the computer and acoustic instruments lies in their control mechanisms, i.e. in the way they are played.

In traditional instrumental playing, every nuance, every small control variation or modulation (e.g. a vibrato or a tremolo) has to be addressed physically by the performer (although this level of control is almost automatic and unconscious in a trained musician). In digital instruments, all parameters can indeed be varied without restriction, continuously or abruptly, but moreover, the performer no longer needs to control directly all these aspects of the production of sound, being able instead to direct and supervise the computer processes that control these details. A related automation mechanism is already present in several electric instruments; applying a mechanical vibrato by means of a Leslie speaker in a Hammond organ or of a variable-speed motor in a vibraphone is indeed much less demanding than keeping a wobbling finger on a violin string! In the case of digital instruments, these automation processes cease to be restricted to simple oscillations and can grow in number and complexity. As a result of the potential intricacy of the ongoing processes, which can be under the instrument's sole control or a responsibility shared by the instrument and the performer, performing music with 'intelligent devices' tends towards an *interactive* dialogue between instrument and instrumentalist.

Interaction

'Interaction' involves the existence of a mutual or reciprocal action or influence between two or more systems. Few of our daily activities do not involve some kind of interaction, as we humans constantly interact with other humans as well as with many artefacts. Driving a car or swinging on a rocking chair are two examples of interactive activities with mechanical devices, since they both involve a two-way communication channel or a feedback loop. Because human communication is the paradigmatic example of interactive communication, complex systems that sense and react to human behaviour through the detection of aspects such as direct physical manipulation, body movement, or changes in the human physiological or psychological states, are often called *interactive*, although it should be pointed out that when cognition is left out of the equation, reaction replaces interaction (Rafaeli 1988). In merely reactive systems, output messages are only related to the immediately previous input message, which makes these systems fully predictable. A light switch will remain a (reactive) light switch independently of the amount of sophisticated sensors it may hide.

Music, on the other hand, has always been a highly interactive activity. Musicians interact with their instrument, with other musicians, with dancers or with the audience, so from this perspective, the very idea of 'interactive music', a term often employed for referring to the music resulting from the dialogue between a musician and a computer, seems self-evident. Ironically enough, interactivity actually suffered due to technological advances attained in the nineteenth and twentieth centuries. Recorded music eliminated the feedback dialogue between the audience and the live musicians, turning music performance into a one-way communication. Later, as a result of multitrack recording, even the dialogue between different players was eliminated. We will here discover how, with their added 'intelligence', digital computers are finally prepared for paying technology's 'musical interaction debts'. We will learn how computer-based instruments can surpass the sound and note levels, flirt with composition and respond to performers in complex, not always entirely predictable ways, even acting not only as instruments, but almost as performers, composers or conductors (Chadabe 1984).

First steps in interactive computer music

In 1957 Max Mathews had programmed the first sounds ever generated by a digital computer, but computers were not yet ready for realtime, so groundbreaking approaches undertaken in the 1950s and 1960s had to remain in the analogue domain. In the late 1950s, the American inventor and composer Raymond Scott[1] constructed electromechanical musical sequencers and *instantaneous music composition performance machines* that could harmonise a melody or provide rhythm accompaniments in realtime (Chusid and Garland 1994). Scott, also a pioneer in approaching electronic music from a pop perspective, was an important influence for Robert Moog, who several years later would construct his popular voltage-controlled analogue synthesisers.

But not everyone could afford Moog's invention. In the mid-1960s, Gordon Mumma and David Berhman, two young American composers distinctly influenced by John Cage, started building analogue circuits capable of producing or processing sound in response to external acoustic stimuli, predating the Machine-Listening interactive systems that will be further discussed in chapter 10. In Mumma's composition *Hornpipe* (1967), an analogue device analyses and amplifies the resonances of the hall in which the performer is playing the French horn (Mumma 1975). Mumma can indeed be considered among the first composers who integrated electronic principles in the realtime operation of the musical structures of the compositions, an approach he termed as 'Cybersonic' and which he applied in many of his pieces.

By the late 1960s, computers still lacked the computing power needed for realtime synthesis (i.e. crunching several hundred thousand operations per second), but they were indeed able to receive and send lower bandwidth information from/to the real (analogue) world. They could receive and digitise voltages (coming from sliders, buttons, sensors, etc.) and they could also send voltages back. This is what Max Mathews achieved again in 1969 with the GROOVE synthesiser, in which he decided to connect a computer to a voltage-controlled analogue synthesiser for managing from the former any parameter of the latter. In his first experiments, Mathews adopted a player-piano approach, storing in the computer the sequence of pitches to be played, thus leaving for the performer the control of timbre, dynamics and of the overall tempo of the piece (Mathews 1991).

Although Mathews' chosen approach can be considered conservative, he had set the basis for a new type of musical interaction. By storing in the computer the instructions that were to be sent to the synthesiser, he showed that new digital instruments could have memory; and when an instrument is capable of knowing which note comes next, some extra knowledge may enable it to take decisions on the fly, even taking into account the performer's gestures. In this context, each note emitted or each action achieved by this hypothetical instrument could thus be the result of a combined decision made by both the player and the instrument. The *musical computer* thus becomes the *philosophical instrument* endowed with memory *and* with sensibility, described by Denis Diderot in the eighteenth century: 'The philosophical instrument is sensitive; it is at the same time the musician and the instrument . . . Imagine the Harpsichord having sensitivity and memory, and tell me if it would not play back by itself the airs that you have performed upon its keys. We are instruments endowed with sensibility and memory' (Diderot 1951, p. 880).

Interactive music: the computer as a semi-autonomous device

The composer and software programmer Laurie Spiegel worked with Max Mathews' GROOVE in the early 1970s and started developing interactive programs for it. In 1973 she coined the term *intelligent instruments* and in 1985 she developed *Music Mouse*, a musical software for the Macintosh, the Commodore Amiga and the Atari ST (Spiegel 1987). The program, which was played by moving the mouse through an onscreen grid, enabled non-experts to perform as if they were accomplished musicians, turning for the first time a computer into a musical instrument that anyone could play. Yet, *Music Mouse* was also a powerful instrument, suited not only for amateurs but also for professionals, as the recordings of Spiegel's compositions and improvisations performed with it testify.

Along with Spiegel, in the early 1970s several other musicians and researchers began to work from similar premises. Joel Chadabe had already started experimenting with automated and random-controlled Moog synthesisers in live performances in 1967. Ten years later, in his piece *Solo*, he connected two Theremin antennas to a Synclavier (one of the first commercially available digital synthesisers) and controlled though the antennas the tempo and the timbre of several simultaneous but independent sequences. Moving his hands in the air allowed him to 'conduct' an improvising orchestra. As a result of these experiments, he coined the term 'interactive composing'. Several years later, Chadabe would found the company Intelligent Music, which in 1986 released the programs *M* and *Jam Factory* (Zicarelli 1987), which together with the aforementioned *Music Mouse*, constituted the first commercial *interactive music* software.

Salvatore Martirano started building electronic music instruments in the mid-1960s at the University of Illinois. Finished in 1972, the *SalMar Construction* included 291 touch-sensitive switches, connected through a massive patch bay to a combinatorial logic unit, which could affect in real time four concurrently running programs that drove the four voices of a custom analogue synthesiser. The fact that the four processes ran in parallel, concurrently modifying parameters such as pitch, loudness, timbre and envelope, made it impossible to fully predict the system's behaviour. In Martirano's own words, 'performing with the system was too complex to analyze . . . Control was an illusion. But I was in the loop. I was trading swaps with the logic' (Walker *et al.* 1992).

By the mid-1970s small microcomputers such as the Kim-1 or the Intel 8008 started to be affordable, opening the ground to a new generation of musicians and experimentalists that started performing with computers with little means and outside of the more academic circles. The *League of Automatic Composers*, the first microcomputer and network band in history, formed in California's Bay Area by John Bischoff, Rich Gold, Jim Horton (later joined by Tim Perkis), constitutes a perfect example of this trend. Each member of the group owned a Kim-1 microcomputer with a sound output, and each one programmed his own computer with programs that were able to produce music by themselves, but also to interchange data with their colleagues' computers, thus creating a network of mutually *listening* computers. 'When the elements of the network are not connected the music sounds like three completely independent processes, but when they are interconnected the music seems to present a "mindlike" aspect' (Bischoff *et al.* 1978). Influenced by the *League* approach, other musicians such as the trombonist, composer and improviser George Lewis began to experiment with computer music improvisation, establishing duets between his trombone and an autonomous computer. This and related approaches, in which the computer behaves more as a performance partner than as an instrument, also known as *machine musicianship*, will be further discussed in chapter 10.

A great technological and democratic breakthrough took place in the mid-1980s, when the MIDI standardisation finally allowed a musician to effortlessly connect every synthesiser with every computer (International MIDI Association 1983; Loy 1985). The MIDI standard coincided with the definitive popularisation of microcomputers, initially 8-bit machines (e.g. Sinclair ZX81, Apple II, Commodore VIC20 or C64) and 16-bit since 1984, with the release of the Apple Macintosh soon to be followed by the Commodore Amiga and the Atari ST. The combination of MIDI with affordable microcomputers finally triggered the commercial music software market. Among the release of many musical applications, most of them MIDI sequencers, there was also room for more experimental and idiosyncratic programs that could fall into the 'interactive music' category, such as the aforementioned *Music Mouse, M, Jam Factory, upBeat* or the interactive sequencers developed by the company Dr. T, which were conceived both as studio and realtime interaction tools predating *Ableton Live* by fifteen years.[2]

The aforesaid applications depict different strategies towards the realtime creation of music using computers, but the flexibility and versatility offered by computers is better explored using a powerful programming language that allows the development of one's own personal ideas. Although computer music programming languages are studied in depth in chapter 4, we cannot omit here some of the software environments that since the late 1990s, and in combination with the increasing power of personal computers, have allowed the definitive bloom of realtime computer music. Certainly, easy-to-use data-flow graphical programming languages such as *Pure Data* (Puckette 1996) or *Max/MSP* (Puckette 2002) or customisable audio programming languages such as *SuperCollider* (McCartney 2002), which musically speaking allow almost anyone to do almost anything, are partly responsible for the popular and definitive acceptance of the laptop as a musical instrument.

Live computer music: what can be done?

In the sleeve notes of their 1989 CD, John Bischoff and Tim Perkis note that 'for us, composing a piece of music is like building a new instrument, an instrument whose behaviour makes up the performance. We act at once as performer, composer and instrument builder, in some ways working more like sculptors than traditional musicians' (Bischoff and Perkis 1989). It is hard to provide a structured vision that covers the multiplicity of all current interactive music possibilities. It seems clear, to start, that such systems permit creative approaches that are sometimes closer to music activities other than the ones we conventionally understand by 'playing an instrument'. As a counterpart to 'unlimited' power, the work of many live computer musicians encompasses a broad range of activities, often more related to composition and to instrument design and building, than to performance or musical practice.

In 1990, Jeff Pressing listed some of these possibilities with the help of traditional music-making metaphors. Practical examples and applications have swelled, but his list, 'playing a musical instrument, conducting an orchestra, playing together (ensemble) with a machine, acting as a one-man band' (Pressing 1990), is still quite pertinent today. For the interactive music pioneer Joel Chadabe, performing with these systems is like 'sailing a boat on a windy day and through stormy seas'. The performer is not in control of everything; some external, unpredictable forces, no matter what their real origin or strength are, affect the system, and the output is the result of this permanent struggle. Whether surprise and dialogue is encouraged through randomness, by ungraspable complexity or by the machine's embedded knowledge, independently of the degree of unpredictability they possess, at their best, these new instruments often shift the centre of the performer's attention from the lower-level details to the higher-level processes that produce these details. The musician performs control strategies instead of performing data and the instrument leans towards more intricate responses to performer stimuli. This situation tends to surpass the note-to-note and the 'one gesture–one acoustic event' playing paradigms present in all traditional instruments, thus allowing musicians to work at different musical levels and forcing them to take higher-level (i.e. more compositional) decisions on-the-fly (Jordà 2002).

The concept of 'note', the structural backbone of Western music, becomes an option rather than a necessity, now surrounded by (macrostructural) form on one side, and (microstructural) sound on the other. As a matter of fact, the inclusion of audio synthesis and processing capabilities into the aforementioned programming environments such as *Pure Data*, *Max/MSP* or *SuperCollider* since the late 1990s is undeniably the single most important occurrence in live computer music for the last two decades. At first glance, virtual or software synthesisers may not seem a radically new concept, but they exhibit essential differences from their hardware counterpart, the venerable keyboard synthesiser. They first provide unlimited scope for freedom of sonic expression, experimentation and imagination. More importantly, they finally allow the seamless integration of macrostructural and microstructural music creation strategies, a blend that does not come without problems:

> What kinds of interfaces to sound might we want for this model? What can we make? Will we be able to move our hands through the space around us, shaping the sounds as we listen, roughing them up here and smoothing them there, and pushing and pulling areas of sonic fabric up, down, toward and away, together and apart? What might be the variables with which we interact? In what dimensions might we move. (Spiegel 2000)

Or as a second observer notes:

> Certain gestures can manipulate sound synthesis directly, while others articulate notelike events, while other actions direct the structural progress of the piece. Applying these concepts to computer realization gives rise to new possibilities that blur the boundaries separating event articulation and structural modification. A single articulation can elicit not just a note, but an entire structural block. Variations in the articulation could very well modify the timbre with which that structure is played, or could introduce slight variations on the structure itself or could modify which structure is elicited. The potential combinatorial space of complexity is quite large. (Tanaka 2000, p. 390)

Although the computer does effectively bridge the gap between musical thought and physical instrumental ability, the above quotations remind us that live music performance cannot be separated from control and gesture. Having focused on the output of musical computers, in the next section we shall turn to the input side, studying gestural controllers.

Gestural controllers

Acoustic instruments consist of an excitation source that can oscillate in different ways under the control of the performer(s), and a resonating system that couples the vibrations of the oscillator to the surrounding air. Where in most non-keyboard acoustic instruments, the separation between the control interface and the sound generating subsystems is fuzzy and unclear, digital musical instruments can always be easily divided into a gestural controller (or input device) that takes the control information from the performer(s), and a sound generator that plays the role of the excitation source. The controller component can typically be a simple computer mouse, a computer keyboard, a MIDI keyboard or a MIDI fader box, but with the use of sensors and appropriate analogue to digital converters, any control signal coming from the outside (i.e. the performer, but also the audience or the environment – as in the case of interactive installations) can be converted into control messages understandable by the digital system. Changes in motion, pressure, velocity, light, gravity, skin conductivity or muscle tension, almost anything, can now become a 'music controller'.

In 1919 the Russian physicist Lev Termen invented the Theremin, the first electronic musical instrument to be played without being touched (Theremin 1996). With its two metal antennas, around which the performer moves the hands controlling respectively pitch and loudness, it constitutes one of the earlier and more paradigmatic attempts of alternative musical control. It should be pointed out, however, that unlike new digital systems, in which any input parameter coming from the controller can be arbitrarily assigned to any sound or musical parameter, the Theremin is a *real* instrument; its sound is actually the direct result of the electromagnetic field variations caused by the proximity of the hands around the antennas. If acoustic instruments are built upon the laws of mechanics, the Theremin behaviour is determined by the laws of electromagnetism. Digital instruments, on their side, are only limited by the imagination and know how of their constructors: a substantial distinction with both positive and negative consequences.

The burgeoning of alternative music controllers started two decades ago with the advent of MIDI, which standardised the separation between input (control) and output (sound) electronic music devices. In an attempt to classify music controllers, Wanderley (2001) distinguishes between (a) instrument-like controllers, (b) extended controllers and (c) alternative controllers. In the first category, not only keyboards, but virtually all traditional instruments (such as saxophones, trumpets, guitars, violins, drums, xylophones or accordions) have been reconceived as MIDI controllers. Although their control capabilities are always reduced when compared to their acoustic ancestors, they offer as a counterpart the possibility to ride an expanded, unlimited sonic palette. The second group, *extended controllers*, includes traditional instruments (which sometimes can be even played 'unplugged'), which with the add-on of extra sensors afford additional playing nuances or techniques and thus supplementary sound or music control possibilities (Machover 1992). Although several extended controllers have been constructed to measure (e.g. for virtuosi such as Yo-Yo Ma or Wynton Marsalis) none of them is being played on a regular basis; none of them has managed to 'dethrone' their original instrumental role model.

The two aforementioned categories profit from known playing techniques and thus may address a potentially higher number of instrumentalists. Until recently, and with honourable exceptions like the quasi-epic work of Donald Buchla (Rich 1991), all commercially available controllers, mainly *midified* versions of traditional instruments, have remained mostly imitative and conservative. Yet traditional performance techniques may not constitute the best strategy to confront the new music-making paradigms discussed in the previous sections.

When it comes to the third category, the jumble of alternative controllers not easily includable in any previous grouping, it is difficult to provide a taxonomy that facilitates a quick overview. Joseph Paradiso, one of the main experts in sensing and field technologies for music controllers' design, classifies controllers according to how they are used or worn. Some of the categories he proposes are *batons, non-contact gesture sensing* (which includes Theremins, ultrasound, computer vision, etc.) and *wearables* (which includes gloves and biosignal detectors) (Paradiso 1997). Among this plethora of alternative controllers, the Radio Baton, a crossover between a conductor's baton and a percussion instrument, conceived again by Max Mathews in 1987 and based on detection principles analogous to that of the Theremin, is one of the more popular (Mathews 1991). As a sign of the growing interest in this field, the annual conference *New Interfaces for Musical Expression* (NIME), which started in 2001 as a fifteen-person workshop, now gathers annually more than two hundred researchers, luthiers and musicians from all over the world to share their knowledge and late-breaking work on new musical interface design. The yearly proceedings[3] constitute the ultimate and most up-to-date source of information on this topic. A more concise but well documented overview of non-imitative controllers that covers dozens of control surfaces, gloves, non-contact devices, wearable or bioelectrical devices can be found in Cutler, Robair and Bean (2000).

Even if the universe of gestural controllers may appear at first glance a quite exclusive club, it definitely is not. Analogue-to-MIDI interfaces designed for converting analogue input values into any type of MIDI message provide straightforward means for opening the computer to the external world, and allow the connection of up to sixteen or thirty-two suitable sensors. Cheaper (and slightly less straightforward) options are offered by easily programmable micro-controllers (such as the Basic Stamp), which permit constructing a custom analogue-to-MIDI interface for less than 50 Euros. Sensors for measuring all kind of physical parameters are also readily available and can be connected to any of these devices, enabling virtually any kind of physical gesture or external parameter to be tracked and digitised into a computer. Moreover, many cheap and widely available control devices meant for the general market, such as joysticks or graphic tablets, can become interesting music controllers. The joystick is a wonderful controller: it has two, three or more degrees of freedom and several buttons and triggers which allow assigning different parameter combinations. Data gloves were originally developed for virtual reality environments, and while professional models are still expensive, several cheap toy-like versions sporadically hit the market such as the *Mattel Power Glove* (1989) or the *P5Glove* (2003). Graphic tablets, which offer great resolution along the X–Y axes plus pressure and angle sensibility, have also proven application for music. To conclude, several software toolkits exist that provide unified, consistent and straightforward connection and mapping of these easily affordable input devices (joysticks, tablets, game pads, etc.) into the more popular programming environments (e.g. Steiner 2005). The stage is definitely set for experimentation with new controllers.

Music controllers can preserve traditional playing modes, permitting us to blow, strike, pluck, rub or bow our 'computers'; new traditionalists in turn, may prefer to continue clicking, double-clicking, typing, pointing, sliding, twirling or dragging and dropping them. The decision is up to everyone. With the appropriate sensors, new digital instruments can also be caressed, squeezed, kissed, licked, danced, hummed or sung. They can even disappear or dematerialise while responding to our movements, our muscle tension or our facial expressions.

With the flexibility offered by MIDI, any controller can certainly be combined with any sound- and music-producing device. Still, each choice is critical. As pointed out by Joel Ryan, improviser, leading researcher in the NIME field and technical director of the Dutch laboratory STEIM, 'a horizontal slider, a rotary knob, a sensor that measures the pressure under one finger, an accelerometer which can measure tilt and respond to rapid movements, a sonar or an infrared system that can detect the distance between two points, each have their idiosyncratic properties' (Ryan 1991). Any input device can become a good or a bad choice depending on the context, the parameter to control, or the performer who will be using it. Just as the automotive engineer chooses a steering wheel over left/right incrementing buttons, 'we should not hand a musician a butterfly net when a pitchfork is required' (Puckette and Settel 1993). The challenge remains how to integrate and transform this apparatus into coherently designed, meaningful musical experiences with emotional depth. It is in fact extremely hard to design highly sophisticated control interfaces without a profound prior knowledge of how the sound or music generators will proceed; a parallel design process will surely be more enriching than buying the ultimate controller for plugging into any custom software.

A fruitful example of this suggested 'holistic instrumental design approach' can be found in the work of composer/violinist Dan Trueman who for almost a decade has pursued the deconstruction and reinvention of the electronic violin. Trueman's research has not been limited to adding sensors to a violin bow or to the fingerboard; in search of the 'real' acoustic instrumental feel, he has even designed special spherical and hemispherical speakers that better simulate the complex sound-radiation patterns of acoustic instruments. His BoSSa, an array of spherical speakers which can be excited and played with a special sensor bow, constitutes the perfect integration of both research lines (Trueman and Cook 1999; Trueman 2006).

The future of digital musical instruments?

We have postulated that by themselves, controllers are not musical instruments. From the same perspective, the computer is too generic to be properly considered one; many musical instruments can be conceived based on a digital computer, each with completely different idiosyncrasies. In this context, a new standard digital instrument has yet to arrive. In effect, not only has no recent electronic instrument managed to reach the (limited) popularity of the Theremin or the Ondes Martenot, invented in 1919 and 1928 respectively; the latest instrument that may argue to have attained classic status, is not digital, not even electronic. Since it started being played in a radically unorthodox and unexpected manner in the early 1980s, thus becoming a *genuine* musical instrument, the turntable has developed its own musical culture, techniques and virtuosi (Shapiro 2002; Hansen 2002). The fact that so many digital turntable simulators already exist, some of them even quite successful commercially, gives us as many clues to the health of the turntable, as it does to the sterility of new instrument design.

New standards may not be essential for the creation of new music; perhaps even the concept of a *musical instrument* is just an old romantic burden that would be better left aside, but somehow it seems that some unrestrained potential is being eschewed in this lawless anything-goes territory. New digital instruments conceived holistically and not as a conglomerate of several interchangeable components are scarce; even worse, in most cases they are only performed by their creators. This situation complicates any progression in the field, both from the design and from the performance perspective. It is not only that electronic music controllers evolve so rapidly that it is rare for a musician to work long enough with one to develop virtuosic technique; it is that every new incarnation seems to come out of the blue. 'A growing number of researchers/composers/performers work with gestural controllers but to my astonishment I hardly see a consistent development of systematic thought on the interpretation of gesture into music, and the notion of musical feed-back into gesture' (Michel Waisvisz, from Wanderley and Battier 2000).

The Dutch improviser (and founder of the aforementioned STEIM lab) Michel Waisvisz, can be considered indeed as one of the very few new instruments virtuosi. Since the early 1980s, Waisvisz has been performing with his self-designed *Hands*, a pair of ergonomically shaped plates fitted with

Figure 5.1 Michel Waisvisz performs with The Hands (photo: Carla van Tijn)

sensors, potentiometers and switches, strapped under the hands of the performer, which are meant to be played in a sort of 'air accordion' manner. Nicolas Collins, with his Trombone-propelled electronics, Laetitia Sonami with the Lady's Gloves, or Atau Tanaka, who has turned a medical electromyograph designed for evaluating the physiological properties and the activity of the muscles into an instrument of his own (Tanaka 2000), are among the few professional performers who like Waisvisz, use new idiosyncratic devices as their main musical instruments.

Interfaces for multithreaded and shared control

'The exploration of post-digital sound spaces, and with it laptop performance, is a dialog conducted with mice, sliders, buttons and the metaphors of business computing . . . Many post-digital composers would be hesitant to freeze this dialogic property through the design and use of a hardware controller' (Turner 2003). Sad but true? When the improviser and software developer Emile Tobenfeld was asked, in 1992, about the desirable features a software instrument for computer-assisted free improvisation should include, he listed: (i) precise control of the timbre; (ii) gestural control of previously composed musical processes; (iii) simultaneous control of multiple instrumental processes; (iv) the ability to start a process, and relinquish control of it, allowing the process to continue while other processes are started; (v) the ability to regain control of an ongoing process; (vi) visual feedback from the computer screen (Tobenfeld 1992). With these ideas in mind we could probably deduce that Turner is wrong in at least one point: it is not the mouse that is so appreciated, but the *monitoring screen* instead. Many laptopists favour indeed the use of MIDI fader boxes for easily controlling the sliders of their *Max/MSP* patches, and the availability of this type of commercial device has increased in parallel with the availability of realtime audio software.

However, most of the music controllers being developed do not pursue Tobenfeld's proposed multithreaded and shared control approach, prolonging the traditional instrument paradigm instead. Trying perhaps to exorcise forty years of tape music, researchers in the field of new musical interfaces tend to conceive new musical instruments highly inspired by traditional ones, most often designed to be 'worn' and played all the time, and offering continuous, synchronous and precise control over a few dimensions. An intimate, sensitive and not necessarily highly dimensional interface of this kind (i.e. more like a violin bow, a mouthpiece or a joystick, than like a piano) will be ideally suited for direct microcontrol (i.e. sound, timbre, articulation). However, for macrostructural, indirect or higher-level control, a non-wearable interface distributed in space and allowing intermittent access (i.e. more like a piano or a drum) should be undeniably preferred (Jordà 2005). Moreover, not many new instruments profit from the display capabilities of digital computers, whereas in the musical performance approach we are discussing, given that the performer tends to frequently delegate and shift control to the instrument, all affordable ways for monitoring processes and activities are especially welcome. Visual feedback becomes thus a significant asset for allowing this type of instrument to dynamically 'communicate' the states and the behaviours of their musical processes. Visual feedback could partially solve another relevant problem of laptop performance, such as the perception difficulties and the lack of understanding these types of performances provoke in the audience (Turner 2003), which could be synthesised as 'how could we readily distinguish an artist performing with powerful software like *SuperCollider* or *PD* from someone checking their e-mail whilst DJ-ing with *iTunes*?' (Collins 2003).

A promising line of research in tune with the aforementioned problems is the application of tangible interfaces to realtime music performance. Tangible User Interfaces (TUIs) combine control and representation within a physical artefact (Ullmer and Ishii 2001). In table-based tangible interfaces, digital information becomes graspable with the direct manipulation of simple objects which are available on a table surface. This is attained by combining augmented reality techniques that allow the tracking of control objects on the table surface, with visualisation techniques that convert the table into a flat screening surface. Not unlike the tables full of toys and sounding gadgets of many free music improvisers, a table with these characteristics favours multi-parametric and shared control, interaction and exploration and even multi-user collaboration. Moreover, the seamless integration of visual feedback and physical control, which eliminates the indirection component present in a conventional screen + pointer system, allows a more natural, intuitive and rich interaction.

In recent years, researchers have developed a variety of tabletop tangible musical controllers, such as *SmallFish, the Jam-O-Drum* (Blaine and Perkis 2000), the *Audiopad* (Patten *et al.* 2006), or the *reacTable* (Jordà *et al.* 2005). In the *reacTable* several musicians can share the control of the instrument by caressing, rotating and moving physical artefacts on a luminous table, constructing different audio topologies in a kind of tangible modular synthesiser or graspable flow-controlled programming language. According to its creators, the reacTable has been designed for installations and casual users as well as for professionals in concert, as it combines immediate and intuitive access in a relaxed and immersive way, with the flexibility and the power of digital sound design algorithms, resulting in endless improvement possibilities and mastership. This claim seems especially relevant if we consider the speed at which technology and fashion shift in our current twenty-first century. Proselytism will surely not be attained by promising ten years of sacrifice. If we aspire for a new instrument to be played by more than two people it will have to capture the musicians' imagination from the start. Which brings us to the next and final section of this chapter: where is the frontier between the 'serious' musical instrument and the 'sound toy'?

Control, virtuosity, intimacy and expressiveness

The instruments we are discussing inhabit a continuum that ranges from the *absolutely passive* conventional instrument (in which the performer is in charge of every smallest detail), to the *fully autonomous* (i.e. human independent) performing machine. They could offer indeed the possibility – the

CD player would constitute an extreme example – to be played by pushing a 'powerful' button that would execute a whole precomposed musical work. With a CD player one can faultlessly play music of extreme complexity with absolutely no effort or training. Satirical as it may sound, this tricky illusion is used indeed in many current interactive sound installations, which seeking to guarantee a complex or predefined musical output, do not give to their interactors more than a couple of bits to play with. We are not criticising here the successful commercial crossover of interactive musical entertainment into mainstream culture, which in the recent years has brought a proliferation of musical game experiences such as *Karaoke Revolution*, *Guitar Hero*, Toshio Iwai's *Electroplankton* for the Nintendo DS, or Sony's Eye Toy rhythm action games (Blaine 2006). Computer-aided interactive music systems can have many applications, each of them perfectly licit and with their own place in the market. However, when faked or useless interactivity happens to be the blot of most contemporary interactive arts, we have to be cautious in order not to trivialise musical creation. *Good* new instruments should learn from their traditional ancestors and not impose *their* music on the performers. A good instrument should not be allowed, for example, to produce *only* good music. A good instrument should also be able to produce 'terribly bad' music, either at the player's will or at the player's misuse.[4] Only if these conditions are sufficiently fulfilled, will an instrument allow its performers to *play* music and not only to *play with* music.

A related control shortage, albeit for opposed reasons, can also be found in the work of many laptop music performers who embrace a bottom-up compositional style favoured by the flexible exploration encouraged in environments such as *Max/MSP* or *Pure Data*. Multithreaded musical processes proliferate in which the responsibility is often left almost entirely to the computer; control is forgotten or delimited to scarce macro control; musical contrasts are sandpapered, the music's inertia augments and the concepts of 'instrument' and of 'performance dexterity' vanish.

It should thus be the luthier-performer's responsibility to establish a balance between different musical levels so that accuracy and fine control on chosen levels do not happen at the expense of leaving important processes unattended. Conscientious instrument designers will better profit out of new digital instruments' possibilities if they do not overlook the essential (albeit implicit and hidden) control features that have made acoustic instruments good enough to resist the acid test of time. We have repeatedly pointed out how interactive music systems emphasise the dialogue between the performer and the instrument, often producing unexpected results, be they non-linearity, randomness or ungraspable complexity. Are these features incompatible with the idea of absolute control, mastery and confidence popularly associated with virtuoso performance? We should not forget that non-linear behaviours are not exclusive to digital interactive music systems. They can be found, for example, in the vocalised and overblown tenor saxophone style or in the use of feedback in the electric guitar. Musicians explore and learn to control these additional degrees of freedom, producing the very intense kinetic performance styles upon which much free jazz and rock music is based. If non-linearity is at first intuitively seen as a source of potential lack of control, it can also mean higher-order and more powerful control. Distinct virtuosity paradigms definitely coexist: whereas the classical virtuoso, with her infinite precision and love for details, may appear closer to the goldsmith, the new digital instruments virtuoso, not unlike the jazz one, could be compared to the bullfighter for their abilities to deal with the unexpected. Confidence is definitely a rare quality in digital instruments (they are computers after all), but a performer needs to know and trust the instrument in order to be able to push it to the extremes and to experiment without fear. Only when a certain level of confidence is reached will performers feel a sense of intimacy with the instrument that will help them in finding their own voice and developing their expressiveness.

Interactive musical instruments can do more than merely transmit human expressiveness like passive channels. They can also be responsible for provoking and instigating in the performer new ideas or feelings to express. When these and related issues are fully understood, we shall hopefully discover computer-based instruments which will not sound as if they were always playing the same piece (although they will be also capable of playing one piece repeatedly, with infinite subtleties and variations, sounding always fresh and innovative). These instruments shall be flexible enough to permit full improvisation without any prior preparation and finally, versatile enough to possess their own defined identities which will even allow the development of personal styles among their performers.

Selected discography

Bahn, C. 2000. *The Rig: Solo Improvisations for Sensor Bass and Live Electronics.* Electronic Music Foundation, EMF CD 030

Behrman, D. 1977. *On the Other Ocean; Figure in a Clearing.* Lovely Music LO 141

Bischoff, J. and Perkis, T. 1989. *Artificial Horizon.* Artifact Recordings, AR102

Brown, C. 1996. *Duets.* Artifact Recordings, AR115

Casserley, L. 1999. *Labyrinths.* Sargasso SG28030

Chadabe, J. 1981. *Rhythms.* Lovely Music VR 1301

Fenn O'Berg 2004. *The Return of Fenn O'Berg.* Mego 54

FMOL Trio. 2002. *The Köln Concert.* Hazard Records 028

Fuzzybunny 2000. *Fuzzybunny.* Sonore

Le Caine, H. 1999. *Compositions & Demonstrations 1946–1974.* Electronic Music Foundation 15

Lewis, G. 1993. *Voyager.* Avant 014

M. I. M. E. O. (Music in Movement Electronic Orchestra) 1998. *Electric Chair + Table.* Grob

Mumma, G. 2002. *Live Electronic Music.* Tzadik TZ 7074

Pair A'Dice 2001. *Near Vhana.* Ninth World Music 15

Parker, E. 1997. *Evan Parker's Electro-Acoustic Ensemble: Toward the Margins.* ECM 453514

Ryan, J. 1994. *Enfolded Strings Inial Mix.* Inial

Scott, R. 2000. *Manhattan Research, Inc.: New Plastic Sounds and Electronic Abstractions.* Basta 90782

Spiegel, L. 1991. *Unseen Worlds.* Scarlet

— 2001. *Obsolete Systems.* Electronic Music Foundation EM119

The Hub 1989. *Computer Network Music.* Artifact 1002

V. A. 2000. *Clicks & Cuts.* Mille Plateaux 8079