The Hyper-Flute

Cléo Palacio-Quintin
Flutist-improviser-composer
3501 De Lorimier Avenue
Montréal, Québec, Canada H2K 3X5
(514) 525-3649
cleopq@sympatico.ca

ABSTRACT
The Hyper-Flute is a standard Boehm flute (the model used is a Powell 2100, made in Boston) extended via electronic sensors that link it to a computer, enabling control of digital sound processing parameters while performing. The instrument's electronic extensions are described in some detail, and performance applications are briefly discussed.

Keywords
Digital sound processing, flute, hyper-instrument, interactive music, live electronics, performance, sensors.

1. FROM INSTRUMENTAL GESTURE … … TO DIGITAL PROCESSING

When I decided to play flute with live electronics, the key issue for me, as a performer, was preserving the intimate relationship between my body, my instrument and the sound it produces. I wanted to keep intact the acoustic richness of the flute, and my way of playing it. The computer had to become a virtual extension of the acoustic instrument.

The richness of physical control required for performance with traditional acoustic instruments takes time to learn. I spent more than 15 years honing my instrumental skills on the flute. While playing an acoustic instrument, all performers receive mechanical feedback cues via a variety of physiological and perceptual mechanisms. These haptic sensations include tactile sensation (touch sensitivity of the skin) and proprioceptive or kinaesthetic perception (awareness of one's body state, including position, velocity and forces supplied by the muscles). Of course, aural feedback is also very important, but the tactile sensation of how one is playing tends to prevail over the sonic outcome.

In extending my flute's tonal field with digital sound processing, I wanted to retain the same subtle control over the sound. It was obvious that I would be better off using my already refined instrumental skills to control the sound processing parameters along with the acoustic flute. The key to achieving this was to use kinetic sensors to capture performance information on the flute and then send these data to the computer. Sensors can convert physical energy into electricity; they make the link between the human world and the machine world.

The next section describes in detail the types of sensors used on the Hyper-Flute, how they react, and exactly where they are installed on the instrument. The physical gestures made while playing the flute have direct consequences on all sensor information sent to the computer. While the mapping of these data into meaningful controls for sound processing is one of the most important issues in working with such an interface, this short article focuses more on the physical description of the instrument.

2. SENSORS

Though there is not much space available to add such hardware to a flute because of its complex mechanism of small keys, it was possible to install several sensors in specific, strategic places:

- magnetic field sensors, which detect the position of the G# key and low C# key (controlled by the two little fingers);
- an ultrasound transducer, which monitors the distance between the flute and the computer;
- mercury tilt switches, activated by the tilting and rotation of the flute;
- pressure sensors under the main points of contact between my hands and the flute (i.e. the left hand and both thumbs);
- a light sensor, which reacts to ambient light on the flute; and
- button switches (discrete values: on/off), which can be reached with the thumbs while playing).

These analog sensors send continuous voltage variation data to a *Microlab* interface, which converts them into MIDI (Musical Instrument Digital Interface) data. These data are then redirected to the computer via a standard MIDI port.

The Microlab is an electronic interface that analyzes the voltage variations from various analog sensors (between 0 and 5 volts) and converts this information into standard MIDI data, which can be sent to a computer, synthesizer, sampler or any other MIDI-compatible device. The interface was originally designed and developed by J. Scherpenisse and A.J. van den Broek, working at the Department of Sonology at the Royal Conservatory in The Hague, Netherlands.

Proprioceptive sensors, describing movements or position, continually send data as MIDI Continuous Control Messages.

Two *magnetic field sensors* transmit the exact positions of two keys (G# and low C#, both controlled by the little fingers). The very short key action distance is precisely measured in 95 steps. I can play with the keys to generate different curves for the output, with quite accurate control. This of course affects the acoustic properties of the flute.

The *ultrasound transducer* is used to measure the distance between the flute and the computer. An ultrasonic signal is sent from the computer and received on the flute. By calculating the delay time, the Microlab provides two different scalings of the flute-to-computer distance.

Some movements also provide discrete values. Two *mercury tilt switches* are triggered by the movement of the instrument. Tilting the flute (moving the footjoint up) activates a Note Number message (on-off), and rotating the flute (turning the headjoint outward) activates another Note Number message.

Pressure sensors are considered as isometric, because there is no movement involved, only muscle tension. Three of them are installed on my flute at the main contact points; each sends Continuous Control Messages. A larger one is installed under my left hand, which holds the flute. There is constant contact and pressure variation as I play. A smaller pressure sensor is found at the B key, under the left thumb. While playing, this key moves often, and is sometimes released completely. The third sensor is under the right thumb, which also supports the flute. There is constant variation depending on which fingerings are played and on the instrument's balance. For these three sensors, maximum values are reachable only with extreme pressure, which does not occur in normal playing, but can be used expressively.

A *light sensor* also sends Continuous Control Messages. This photoresistor, which detects variations in ambient light, is positioned at the headjoint, and is designed to be used in conjunction with stage lighting effects.

Other controllers used on the Hyper-Flute are small *button switches,* which send discrete values (Note Number on/off). Two of them (blue) are placed close to the headjoint and are not easily reachable while playing. I mostly use them to change settings between sections or at the beginning of a piece. Four others are placed close to the thumbs, and can be reached while playing.

3. MAPPING & PERFORMANCE

Using the Max-MSP programming environment, different programs, or "patches," are developed and integrated into a complex software interface, which performs the flute's sound processing in real time. This software is entirely controlled by the Hyper-Flute. For each programmed patch, all the MIDI data can be processed and used to control different sound processing parameters. The mapping of the MIDI daya to different parameters can be modified before each performance, or even during a performance. This mapping is a crucial step in the interface between instrumental gesture and digital sound processing.

As explained earlier, each sensor occupies a specific position on the instrument. The way the flute is played and held creates multiple interactive relationships between some of the sensors. For example, diminishing the pressure on one or both thumb sensors immediately increases the pressure under the left hand, and pushing one of the button switches causes a corresponding thumb pressure sensor to lift. This interaction between the various sensors needs to be considered while programming the mapping of the data to the sound processing parameters.

Of course, different sound processing patches require different ways of controlling them. The mapping must be adapted to each specific situation, and a lot of fine-tuning is necessary. Since my sound processing software is in continual development, no definite mapping scheme is in use yet. I am constantly experimenting with different combinations of direct, convergent and divergent mapping, some being more suitable than others for controlling specific sound processing patches. Comprehensive analyses of the data generated by the sensors while playing the instrument would be necessary to find more precise relationships and develop a very good, multiparametric interface (see Hunt and Kirk, 2000, for more details on mapping strategies.)

Musical applications of the Hyper-Flute are infinite. The immediate link between the physicality of playing the flute and the data sent to the computer makes possible a variety of interactions between performer and computer. Besides processing the acoustics of the flute in real time, the patches can also be used to control other types of electronic structures—for example, to trigger sound files or independent sound synthesis algorithms. The design of the patches thus becomes part of the compositional process. The whole concept is very different when a composed piece is being played, as opposed to improvisational performance. Musical examples will be performed during the demonstration of the Hyper-Flute.

4. ACKNOWLEDGMENTS

I extend especial thanks to Bert Bongers and Lex van den Broek, who built parts of my instrument and patiently taught me how to solder tiny cables, sensors, resistors, switches, and so on; to Jonathan Impett, whose Meta-Trumpet and musicianship have been and continue to be a huge source of inspiration; to Paul Berg, for making me discover and plunge into the world of computer music; and to Anne LaBerge, for her fantastic flute tips, contagious energy and musicality.

5. REFERENCES

[1] Bongers, B. Physical Interfaces in the Electronic Arts. In *Trends in Gestural Control of Music* [CD-ROM], M.M. Wanderley and M. Battier, eds., Paris: IRCAM-Centre Pompidou, 2000.

[2] Bongers, B. The Use of Active Tactile and Force Feedback in Timbre Controlling Electronic Instruments. *Proceedings of the 1994 International Computer Music Conference,* Arhus, Denmark, 1994.

[3] Hunt, A. and Kirk, R. Mapping Strategies for Musical Performance. In *Trends in Gestural Control of Music*, op. cit.

[4] Impett, J. A Meta-Trumpet(er). *Proceedings of the 1994 International Computer Music Conference,* op. cit.

[5] Machover, T. and Chung, J. Hyperinstruments: Musically Intelligent and Interactive Performance and Creativity systems. *Proceedings of the 1989 International Computer Music Conference,* San Francisco, CA, 1989.