The Overtone Fiddle: an Actuated Acoustic Instrument

Overholt, Daniel

Published in:
Proceedings of 11th International Conference on New Interfaces for Musical Expression

Publication date:
2011

Document Version
Accepted author manuscript, peer reviewed version

The Overtone Fiddle: an Actuated Acoustic Instrument

Dan Overholt
Department of Architecture, Design
and Media Technology
Aalborg University, Denmark
Niels Jernes Vej 14, 3-107
dano@create.aau.dk

ABSTRACT

The Overtone Fiddle is a new violin-family instrument that incorporates electronic sensors, integrated DSP, and physical actuation of the acoustic body. An embedded tactile sound transducer creates extra vibrations in the body of the Overtone Fiddle, allowing performer control and sensation via both traditional violin techniques, as well as extended playing techniques that incorporate shared man/machine control of the resulting sound. A magnetic pickup system is mounted to the end of the fiddle's fingerboard in order to detect the signals from the vibrating strings, deliberately not capturing vibrations from the full body of the instrument. This focused sensing approach allows less restrained use of DSP-generated feedback signals, as there is very little direct leakage from the actuator embedded in the body of the instrument back to the pickup.

Keywords

Actuated Musical Instruments, Hybrid Instruments, Active Acoustics, Electronic Violin

1. INTRODUCTION

The Overtone Fiddle follows upon the development of the author's prior Overtone Violin [6], with a change of focus towards another area of investigation. Whereas the Overtone Violin is entirely electronic (there is no use of a resonating acoustic body), the Overtone Fiddle described here integrates a full acoustic chamber. It receives resonant stimulation directly from both the strings on the instrument, and from an internally mounted tactile sound transducer, which is controlled via DSP running on an attached iPod Touch®. The physical design of the instrument accommodates this by incorporating space to mount the iPod locally, as can be seen in Figure 1.

The instrument is essentially a 'pochette' [13] type of violin design, with a standard violin length and a 2" external width. Luthier Don Rickert, of Adventurous Muse [9] made plans and constructed the instrument requested for this project. The internal cavity is 1.75" wide in order to accommodate a specific tactile sound transducer. The overhanging top and back, in addition to adding to the vibrating surface, and thus, sonority of the instrument, also provide mounting surfaces for external components such as batteries, pickup preamps, and so forth. The entire back of the instrument is made of maple, and screwed onto the instrument sound chamber (main body) in order to allow easier access to internal elements and wiring.

NIME'11, 30 May–1 June 2011, Oslo, Norway.

2. SENSOR & ACTUATOR DESIGN

The Overtone Fiddle uses a tightly focused sensing approach to capture string vibrations – several designs were attempted, only the chosen method is described herein.

Figure 1. The Overtone Fiddle – first prototype.

2.1 Magnetic Pickup

While an optical pickup system similar to that used with the Overtone Violin could have been designed for the Overtone Fiddle as well, it was deemed unnecessary, as a commercially available magnetic pickup system was found that captures the movements of the strings themselves directly. This system is called REBO [8], and it functions in the same manner as an electromagnetic guitar pickup. It is mounted to the end of the instrument's fingerboard (see Figure 2, left).

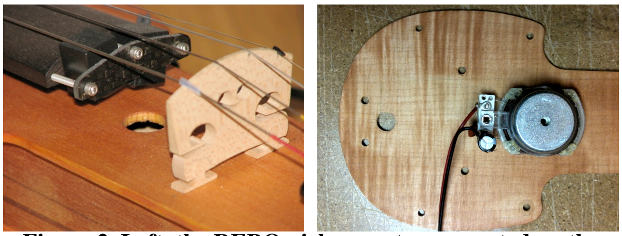

Figure 2. Left, the REBO pickup system mounted on the Overtone Fiddle, and right, the internally mounted tactile sound transducer (not shown here, a similar transducer is located in the instrument's second acoustic body).

2.2 Tactile Sound Transducers

Signals can be injected directly into the main acoustic body of the instrument via a voice-coil type of tactile sound transducer (see Figure 2, right), as well as into an optional second acoustic body that hangs below the instrument (see Figure 3). This lower resonating body is made of very thin carbon fiber and balsa wood – materials that would not be strong enough to support the full string tension of strings on the main body – thus allowing extremely efficient transfer of acoustic energy from another embedded tactile transducer, to the structural elements of the box. The box itself is quite a simple design for this prototype. It was designed to dimensions allowing it to function as a proper Helmholtz-resonator (internal shape and porting), in order to maximize the volume of the resulting sound. As such, it is actually capable of producing louder tones than the main body of the instrument.

Figure 3. The Overtone Fiddle with carbon fiber / balsa wood second acoustic body mounted underneath.

Designed as a 5.6" x 5.6" x 1.2" box, this second body has a total internal volume of roughly 37.6 cubic inches. To relate this to the spherical shape of a traditional Helmholtz resonator, solving equation 1 below provides the radius of a sphere with an equivalent internal volume. Then, to arrive at the proper size of the soundhole, this radius is divided by four, resulting in a prototype design with a 0.519” radius circular soundhole.

$$R = \sqrt[3]{\frac{V}{4/3\ PI}}$$

Equation 1. Solving for the radius of an equivalent spherical Helmholtz resonator, given the internal volume of the prototype second body.

The second body of the Overtone Fiddle is also driven with DSP-generated feedback signals, usually based on the sound from the strings (indeed, any audio signal the performer desires is possible, if suitably programmed). Many types of responsive software can be programmed to run on the iPod touch mounted on the fiddle. Sounds and effects can be responsive to any motions sensed by the accelerometers and gyroscopes in the iPod Touch, with parameters controlled by both real-time analysis of incoming sound from the strings, and gestural movements of the performer.

The tactile transducers inside both the top and bottom resonating bodies are driven independently by a 11.1volt Li-Ion battery-powered Class-T stereo audio amplifier. As mentioned, the main body of the instrument is designed to accommodate this, by providing space for mounting the battery, amplifier, and associated circuitry. This makes the entire instrument self-contained (not including the bow and its corresponding electronic circuit).

2.3 Bow Design

The bow used with the Overtone Fiddle is custom made by the E.W. Incredibow company, from a simple carbon fiber rod that is lighter (almost half the weight of a wood bow) and longer than a normal violin bow. This is helpful in order to accommodate the added mass of a small battery-powered sensor circuit based on the CUI32 [5], along with a wireless 802.11g radio module [11], and an absolute orientation sensor [2]. A simple BASIC-language program was written on the CUI32 using StickOS [12], which is the default operating system for the CUI32. It receives the stream of orientation vectors from the sensor, and translates them into Open Sound Control (OSC) protocol, in order to send them to the iPod Touch. The orientation sensor reports the direction in which the bow is pointing using Euler angles or quaternions, sending updates at 300Hz. This is accomplished by its sensor fusion algorithm, which combines data from internal accelerometers, gyroscopes, and magnetometers.

The WiFly module is configured to broadcast its own ‘AdHoc’ 802.11 base station, which is then chosen as the network to join in the ‘WiFi settings’ of the iPod Touch. This allows the CUI32 to send UDP and/or OSC and communicate easily with the iPod Touch. One of the strengths of this approach is that the orientation of the bow can be compared to the violin/iPod’s orientation (as determined using the iPod’s internal sensors) and differences in these measurements can be used to control various parameters of real-time effects processes. The mapping of such controls to real-time parameter updates is of course a major task given to the composer / performer / programmer of the system.

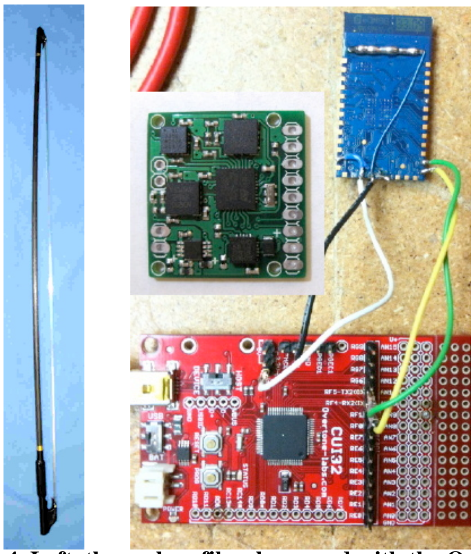

Figure 4. Left, the carbon fiber bow used with the Overtone Fiddle, and right, the electronics components before being attached to the bow (red CUI32 bottom, blue 802.11g radio module top, and green absolute orientation sensor middle).

3. MUSICAL PROGRAMMABILITY

Software used with the Overtone Fiddle can be written in a variety of applications, such as SuperCollider [4], PureData (using libpd [3] or RjDj [10]), or the MoMu the Mobile Music Toolkit [1], all of which can run in real-time on the iPod Touch. In general, the author tends to use SuperCollider, but has experimented extensively with others as well. The instrument can thereby incorporate all of the flexibility of modern digital signal processing techniques, for example, altering the timbral structure of the sound being produced in response to player input.

In order to use the sensor data from the bow with any of these iPod Touch applications, the data must be formatted into either OSC as mentioned earlier, or another network format that the iPod application can receive via the WiFly module. For an example here, Figure 5 shows BASIC code that captures sensor values on the 16 analog input pins on the CUI32, and sends them as UDP to a custom “scene” in RjDj – really just a PureData patch, which is shown in Figure 6. In this case, the absolute orientation sensor was not used, as this was a simple test to verify connectivity. Nonetheless, there are many different types of analog sensors that can be used with the 16 analog input pins on the CUI32. Therefore, such a setup will almost surely be useful in the future.

To explain the BASIC code shown in Figure 5, it can be seen that on line 10 the 2nd UART (serial port of the CUI32) is initialized. This UART is connected to the WiFly module via two wires: one for transmitting, and one for receiving. Lines 20 through 160 are declaring variables “a … p” that correspond to the 16 analog input pins on the CUI32. These are configured as ‘debounced’, which causes a simple 3-point running average filter to be executed on the incoming sensor values, in order to smooth out any glitches. Lines 170-210 create a connection between the iPod Touch and the CUI32 (the iPod must already have joined the WiFly's AdHoc network).

```
10  configure uart 2 for 115200 baud 8 data no parity
20  dim a as pin an0 for analog input debounced
30  dim b as pin an1 for analog input debounced
40  dim c as pin an2 for analog input debounced
50  dim d as pin an3 for analog input debounced
60  dim e as pin an4 for analog input debounced
70  dim g as pin an6 for analog input debounced
80  dim h as pin an7 for analog input debounced
90  dim i as pin an8 for analog input debounced
100 dim j as pin an9 for analog input debounced
110 dim k as pin an10 for analog input debounced
120 dim l as pin an11 for analog input debounced
130 dim m as pin an12 for analog input debounced
140 dim n as pin an13 for analog input debounced
150 dim o as pin an14 for analog input debounced
160 dim p as pin an15 for analog input debounced

170 sleep 100 ms  rem -- wait for WiFly to boot
180 print "$$$";  rem -- tell WiFly to enter config mode
190 sleep 300 ms  rem -- wait for it to enter config mode
200 print "open 169.254.1.3 2000"  rem -- connect to iPod Touch
210 sleep 100 ms  rem -- wait for connection to establish

220 configure timer 0 for 10 ms  rem -- interrupt update rate 100Hz
230 on timer 0 do print "A",a,b,c,d,e,"0",g,h,i,j,k,l,m,n,o,p,";"
240 while 1 do  rem -- no need to do anything in the main loop because
250 endwhile  rem -- everything is handled by the interrupt routine
```

Figure 5. StickOS BASIC program that sends the CUI32's analog sensor inputs to the [netreceive] object in the RjDj app on the iPod (running a corresponding RjDj "scene", which is actually the PureData patch shown in Figure 6).

Finally, line 220 enables an internal timer in the CUI32 (functionally similar to the [metro] object in PureData), and configures it to cause events every 10 milliseconds. Every time one of these events happens, line 230 sends the actual sensor values to RjDj, which is running the PureData patch seen in Figure 6. The list of sensor values is always preceeded with a capital "A", and appended with a semicolon. In PureData, the semicolon is needed by the [netreceive] object to signify the end of a packet, and the capital "A" is used as an identifier to signify the beginning of the packet. The top [match] object in Figure 6 always checks for the capital "A" (number 65 in ASCII-code) so that synchronization is maintained.

The same functionality can be achieved with other iPod applications with a few modifications to the code. For example, SuperCollider requires the use of OSC-format strings in order to receive network data, so the addition of proper OSC syntax (string identifiers and 4-byte boundaries) is added to the BASIC code in order to use it with SuperCollider running on the iPod Touch. For the sake of brevity, an example of such is not included here.

The CUI32 circuit board was designed by the author as an improved version of the CREATE USB Interface (CUI) [7], and is sold by SparkFun electronics and other online retailers.

4. NEW PERFORMANCE TECHNIQUES

Since any DSP algorithms in use can be controlled through gestural interaction using both the motion sensors in the iPod (accelerometers, gyroscopes, etc.) as well as the electronics on the bow, the system promotes new performance techniques for interactions above and beyond those supported by traditional acoustic instruments. For instance, in initial improvisational performances by the author, the timbre of the Overtone Fiddle changes is made to change dramatically when rapid movements are performed.

The multitouch screen surface on the iPod is also useful in certain musical contexts. While it clearly cannot be used while simultaneously bowing anything other than open strings, there can nonetheless be sections of a performance allowing the performer enough time to access the screen. For example, a DSP algorithm can sustain a note beyond the time that the performer actually bowed it, thus allowing modifications to the timbre thereafter through interaction with the multitouch screen. Simple parameter changes can of course be executed in between notes as well, etc.

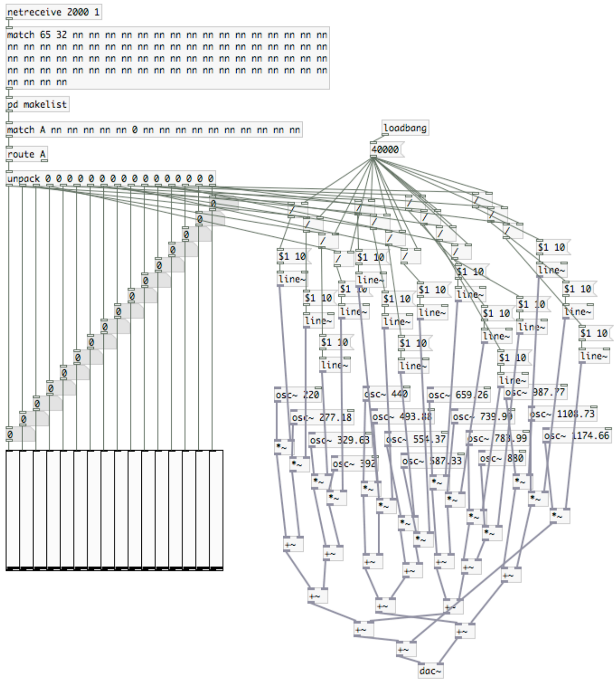

Figure 6. PureData patch used in RjDj to receive real-time analog sensor data from the CUI32. On the left side, sliders visually represent incoming sensor values, and on the right side, a simple additive synthesizer used for testing that generates sounds in response to the sensor data.

4.1 New Sonic Possibilities

DSP algorithms are used to adjust the body vibrations of the acoustic part of the instrument, actively changing the harmonics heard in the musical tone quality. Consequently, the acoustic properties of the Overtone Fiddle are adjustable, rather than being permanently defined by the woodworking skills of a luthier. In other words, the internal actuator can cause the wooden body to exhibit new dynamic behaviors, and it is this methodology that distinguishes the Overtone Fiddle from prior instruments such as the Chameleon Guitar [14].

While the sound quality of a traditional acoustic instrument is fixed by its physical design, actuated musical instruments can malleably simulate even physically impossible acoustic properties, such as a violin with hundreds or thousands of sympathetic strings; this would be akin to extreme versions of instruments like the Norwegian Hardanger Fiddle, or the Viola d'Amore from the baroque period. For the performer and the audience, however, the perception is that the sound being produced by an actuated acoustic musical instrument such as the Overtone Fiddle is somehow physical – the resulting music is created through gestures on the instrument held close to the performer, who can use the technology to produce interesting timbres that might never have been heard before. It is an important consideration for the performer, that the Overtone Fiddle is not connected by cables to a computer, nor to any remote loudspeakers.

5. MUSICAL OBJECTIVES

The main objective of the development of the Overtone Fiddle is to pursue a long-term research project that is focused on the development of new acoustic musical instruments – at first, in the bowed string family and then expanding to others. The addition of technological components to acoustic instruments is used in order to extend these existing instrument types with new expressive and performative possibilities. The project also aims to explore the potentials that these instruments have in both new compositions and new methods of performance.

It is the author's hope that the development of such new instruments will help revive the evolution of more traditional musical instruments today, through a combination of some of today's most advanced technologies with traditional instrument making and musical skill and practice. In this sense, traditional acoustic instruments are already seen as advanced technologies in their own right, because they have been refined over many years of development. The goal is to add new dimensions and expressive possibilities to the capabilities of traditional acoustic instruments, and to explore these in contemporary music and performance. The research project should not be construed as an attempt to improve the basic acoustic instrument designs themselves, but seen as an extension of said instruments' expressive and performative range.

Because the acoustic properties of the Overtone Fiddle can be changed through mathematical processes in real time, it allows artists to make radical changes to their sound. This gives an opportunity to create music that explores new sound worlds, yet sill follows in the traditional musical training to a certain degree. Composers and performers can make use of the instrument's programmability by means of sound synthesis, sound effects and generative algorithms, all of which can be configured to respond to input from the musician and allowing an almost infinite number of different instrument interaction methods.

6. CONCLUSION AND FUTURE WORK

The development of the Overtone Fiddle offers both technical and artistic challenges that the author enjoys embracing. It has been shown that the first prototype of the instrument is capable of many new musical interactions – future versions of the fiddle, as well as other bowed string instruments are currently in the works. While cables or even wireless audio transceivers could be used to enable the Overtone Fiddle to connect to a laptop for more powerful signal processing than is possible with an iPod Touch, keeping the instrument compact and self-contained is a high priority for this project. Nonetheless, initial experiments were done using a laptop, and OSC-sending remote-control apps such as TouchOSC or Fantastick.

Future prototypes of actuated bowed string instruments may incorporate more traditional violin bodies, instead of the 'pochette' type of design. They may also involve a new concept developed in this research project, that of a "bridgeless" violin that was tested in the process of building this first prototype. In this case, an extended fingerboard is used, with the wide end culminating in a raised nut where the bridge would normally be. The strings are then entirely supported by the fingerboard (never touching the body of the instrument). The instrument body is hung from a bracing system running underneath (that also supports the iPod and electronics), in order to separate the fingerboard vibrations from the actuated body. With this setup, a secondary instrument body is not used.

The author can be seen playing the prototype Overtone Fiddle in a video titled "An Evening of Actuated Instruments" together with Edgar Berdahl on the Haptic Drum and Robert Hamilton on the Feedback Resonance Guitar in an improvisational setting. This video is located online at the Actuated Musical Instruments Guild website: http://actuatedinstruments.com/.

7. ACKNOWLEDGEMENTS

I would very much like to thank Chris Chafe, Max Mathews, Edgar Berdahl, Rich Testardi, Don Rickert, and my wife Anne-Marie Hansen for all of their help with this project. The support they have given me has been a great help in many ways.

I also wish to acknowledge Lars Beer Nielsen at the Innovation Center Denmark, and Keith Devlin at Stanford University's Human Sciences & Technologies Advanced Research Institute (H-STAR) for their financial support, providing the opportunity to spend 2 months at Stanford's Center for Computer Research in Music and Acoustics (CCRMA) working on this research.

8. REFERENCES

[1] Bryan, N. J., Herrera, J., Oh, J., Wang, G. Momu: A mobile music toolkit. *In Proceedings of the. International Conference on New Interfaces for. Musical Expression (NIME)*, Sydney, Australia, 2010.

[2] CH Robotics, http://www.chrobotics.com/ accessed January 29, 2011.

[3] LibPd, http://gitorious.org/pdlib/ accessed January 29, 2011.

[4] McCartney, J. Rethinking the Computer Music Language: SuperCollider. *Computer Music Journal, 26(4), 61-68.* 2002

[5] Overholt, D. CUI32 microcontroller board, http://code.google.com/p/cui32/ accessed January 29, 2011.

[6] Overholt, D. The Overtone Violin: a New Computer Music Instrument. *Proceedings of the International Computer Music Conference*, *ICMC 2005,* (Barcelona, Spain, 5-9 September 2005).

[7] Overholt, D. Musical interaction design with the CREATE USB Interface: Teaching HCI with CUIs instead of GUIs. *Proc. of the 2006 International Computer Music Conference*, New Orleans, 2006.

[8] REBO, http://www.uli-boesking.de/rebo/ accessed January 29, 2011.

[9] Rickert, D. http://www.adventurousmuse.com/ accessed January 29, 2011.

[10] RjDj, http://www.rjdj.me/ accessed January 29, 2011.

[11] Roving Networks (WiFly GSX module), http://rovingnetworks.com/ accessed January 29, 2011.

[12] Testardi, R. (StickOS), http://cpustick.com/stickos.htm accessed January 29, 2011.

[13] Pochette (kit violin), http://en.wikipedia.org/wiki/Kit_violin accessed January 29, 2011

[14] Zoran A. and P. Maes, Considering Virtual and Physical Aspects in Acoustic Guitar Design, *Proceedings of New Instruments for Musical Expression (NIME) Conference*, Genova, Italy, June 5-7, 2008.