

Re-coupling: the uBlotar synthesis instrument and the sHowl speaker-feedback controller

Van Stiefel^{*}, Dan Trueman[†], and Perry Cook^{††}

^{*}Faculty of Music, McGill University
van.stiefel@mcgill.ca

[†]Music Department, Princeton University
dan@music.princeton.edu

^{††}Computer Science (and Music), Princeton University
prc@princeton.edu

Abstract

This paper concerns enhancements to the blotar, a digital synthesis instrument based on similarities between physical models of the flute and electric guitar. Digital instrument structure and control parameters are described and related to mapping concerns for a gestural controller—the sHowl, a hand-held speaker used also to produce feedback on an electric guitar signal.

1 Introduction

This paper presents continuing research on a hybrid digital-electro-acoustic instrument that brings together a synthesis instrument, an electric guitar, and a hand-held speaker used to feedback the guitar signal as well as control the parameters of an engaging digital synthesis instrument. Patching a small, hand-held speaker out of the headphone jack of a guitar effects processor produces feedback in the electric guitar signal when the speaker is brought near the instrument’s pick-ups. The mellifluous range of squeals generated by this technique earned it the name “Howl” guitar. The *sHowl* (sensor-Howl) turns this hand-held speaker into a controller with a range of gestures remarkably suited to express the odd beauties of the *blotar*—a digital synthesis instrument based on similarities between the physical modeling of flute and electric guitar timbres.

When the digital synthesis instrument is combined with the “howl” electro-acoustic effect, an expressive dialog ensues between various forms of digital and electro-acoustic feedback. This paper 1) describes design and aesthetic motivations for the *sHowl*; 2) compares aspects of the structure and control parameters of the *blotar* with an improved version of the digital instrument called the

uBlotar; 3) surveys the timbre space of the *uBlotar*; and, 4) describes a process for mapping the control parameters of the *uBlotar* to the *sHowl* (comparing these mappings with earlier ones of *sHowl* to *blotar*) with the aim of demonstrating the increased timbre space and interface possibilities of the enhanced digital instrument. In addition, this paper addresses some general compositional/theoretical issues of concern when mapping hybrid synthesis instruments to gestural controllers.

1.1 “Howl” guitar background

Few instruments can claim the wealth of associations between physical gesture and sound production characterizing electric guitar performance. (It’s difficult to imagine, for example, being an enthralled, informed spectator of an “air-oboe” contest). One of the most characteristic associations between sound and gesture in electric guitar playing results from the production and physical control of feedback as pioneered and mastered by guitarists like Jimi Hendrix. To sustain and modify the feedback, a player must adjust the distance of the pick-ups from the source of amplification. Sometimes turning from the audience at an intense moment in the course of an improvisation, the performer reveals the physical source of amplification as an extension of the instrument, and at the same time, explores the limits of a sonic system/environment by way of a fragile extended instrumental technique.

The Howl guitar, a prototype for the *sHowl* controller, reverses this procedure for producing feedback. Instead of bringing the instrument to the speaker, the speaker is brought to the instrument, both referencing and altering the

gestural modalities of “performing” the effect. Portable speakers are connected to the headphone output of a commercial signal processor, the line6 POD. The feedback is modified by various effects (mainly wah-wah and overdrive) and connected to the main out of the POD. The Howl was an integral timbre in the collaborative composition/improvisation of *maison vague*, a trio based in Essen, Germany (Van Stiefel, guitar, Clark Stiefel keyboards, and Alex Catona, electric cello).

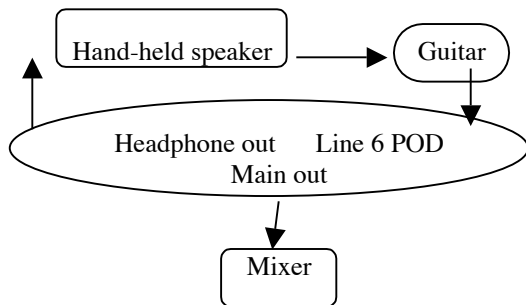


Figure 1. Audio Chain for “Howl” Guitar

The desired musical effect motivating this rigged-up procedure (figure 1) was twofold. The performer can take advantage of digital models of distortion/amplification processes without the ear-splitting volume required to overdrive, distort, and otherwise produce feedback in conventional electric guitar systems. At the same time, digital models of such effects in performance can read as artificial renderings of processes once achieved solely through influencing rather “chaotic” electro-acoustic systems. Manipulating the hand-held speaker to control feedback reintroduces both the visual component of generating feedback as well as an ad hoc element of controlling its output.

In conventional guitar technique, vibrato and string bending are the only techniques available to alter the qualities of a sustained sound. The addition of a foot pedal provides the electric guitarist with added capabilities to modify the sound of a vibrating string. Beyond these, there is little to indicate the physical relationship between the performance act and an audible transformation of sustained sound. The production and control of feedback suggests a potent model for a performer-guitar-computer interface appropriate for controlling continuous musical parameters of computer-generated sounds and effects.



Figure 2. Playing the *sHowl*

2 The Blotar and the uBlotar

2.1 The Blotar

The *blotar* synthesis instrument grew out of the observation that the physical models of the flute and the electric guitar are nearly identical. (M. Karjalainen et al. 1991) (P.Cook 1992) (C.R. Sullivan 1990) The flute model consists of two feedback delay lines, one representing the length of the bore, the other the airflow between to embouchure and the mouthpiece. These are coupled by a non-linearity that roughly models the interaction between the incoming airflow from the player with the oscillating airflow in the bore. This model is known for being one of the most sonically convincing physical models, and also for being quite inspiring in terms of interaction; the control parameters are immediately engaging and challenging.

The electric guitar model also features two feedback delay lines, one representing the string length (which corresponds to the bore length in the flute), and the other, the distance between the guitar and the speaker. These are also coupled by a non-linearity modeling the distortion of an overdriven tube amplifier. In addition to the remarkable fact that these two models are topologically identical, the non-linearity which links the delay lines is also the same for both models ($f(x) = x - x^3$, with a limiting function).

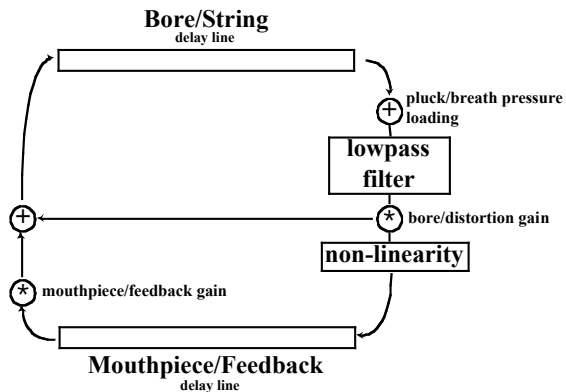


Figure 3. Basic Signal Structure for the Blotar

One difference between the models is how they are excited. Typically, the flute is driven by a continuous input of noise (sometimes varying in pressure, to create a sense of vibrato), while the electric guitar is “plucked” by a short burst of noise (or by loading the “string” delay line in various ways). Another implementation difference lies in the type of low-pass filter used; the flute model typically uses a one-pole filter to simulate the losses in the bore and at the open end, whereas the electric guitar uses the typical one-zero filter from the Karplus-Strong plucked string model and its extensions. (Karplus and Strong 1983)(Jaffe and Smith 1983)

The *blotar* is an implementation of this instrument in Max/MSP and that provides handles to control all of the parameters necessary to move the instrument from more flute-like behaviors to more electric guitar-like. (Trueman and DuBois 2002) While “morphing” between the instruments is challenging (the *blotar* is a remarkably difficult instrument to control and perform), it is possible to create moments where the instrument seems to be having an identity crisis, or, more accurately, the listener may hear features of both the flute and the electric guitar simultaneously. The model includes excitation methods for both the electric guitar and the flute. In addition, the plucking is achieved not through the use of bursts of noise, but by driving the system with the impulse responses from an acoustic guitar (these responses are from the NBody Project, which allows the player to color the pluck with the body-filter of the impulse response (the Body Size parameter scales this impulse). (Cook and Trueman 1999) In practice, this subtle effect is completely overwhelmed by the other features of the model.

An unusual (and physically unmotivated) control parameter, one that turns out to be quite compelling, allows the player to crossfade between the two lowpass filter types; the space between the extremes of this crossfade can be quite dynamic, often completely transforming the instrument from a howling electric guitar to a whimpering

nose-flute. The interesting cross-fade between the two filters yields a more complex filter than either of the two extremes. For example, at the flute position the difference equation is $y(n) = 0.1x(n) + 0.9y(n-1)$ (unity gain one pole at 0.9). In the guitar position the difference equation is $y(n) = 0.5x(n) + 0.5x(n-1)$ (standard Karplus-Strong moving average one-zero filter). However, at the 1/2 way crossfade point between flute and guitar filters, the difference equation is $y(n) = 0.3(x(n) + 0.833x(n-1) + 1.5y(n-1)$ (actually a filter with two zeroes at 0.825 and -0.909 , and one pole at 0.9). This hybrid parametric filter affects both tuning and stability in strange and wonderful ways.

Finally, the original Sullivan electric guitar model provides controls for various details of the guitar model, including a mix of pre- and post- distortion outputs; these were neglected in favor of keeping the model as simple and close to the common ground of the two original models as possible, but are included, among others features, in the *uBlotar* (below). Table 1 lists the various control parameters for the *blotar* and their instrumental motivations.

Parameter	Instrument Model
bore/string delay length (frequency)	both
bore/distortion gain	both
mouthpiece/feedback delay length (freq.)	both
mouthpiece/feedback gain	both
breath pressure	flute
noise gain	flute
pluck amplitude	electric gtr.
pluck position	electric gtr.
body size	electric gtr.
vibrato frequency	flute
vibrato amplitude	flute
One-pole/one-zero cross-fade coefficient	both/neither

Table 1. Control Parameters for the Blotar

2.2 the uBlotar

The *uBlotar* is an elaboration on the *blotar*, taking its basic design and extending it further into the electric guitar domain. The model includes six string/bore models, each with its own lowpass filter (the player can again fade between one-pole and one-zero filter types) and feedback coefficient. Each string includes a DC blocking filter (as does the *blotar*) to remove the DC bias that typically builds up in these kinds of structures, and also a limiting threshold (that the player can control) to prevent the individual strings from blowing up. The outputs from these strings are summed and combined with whatever breath pressure the

player provides and then fed into the “distortion unit” and feedback delay line. Pre- and post- distortion levels can be controlled and mixed at the output.

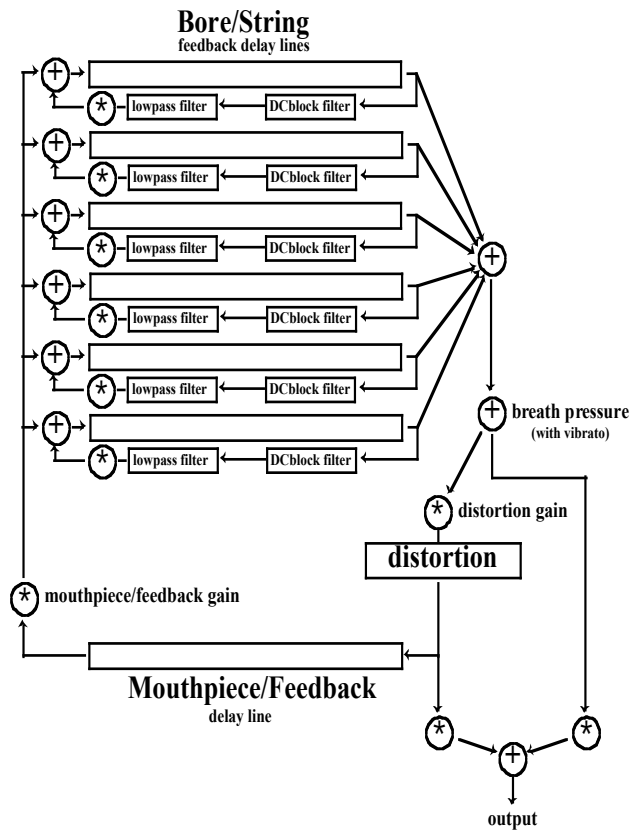


Figure 4. Basic Signal Structure for the uBlotar

The uBlotar also includes various subtle improvements over the blotar, including signal-rate control of delay lengths and other parameters, making smooth adjustments of these parameters possible; such control makes possible the subtle exploration of the various uBlotar timbral spaces. Finally, the plucking implementation of the uBlotar uses a technique from the original Sullivan model; the delay lines are loaded (strings can be plucked individually) with noise, and then passed through a lowpass filter a certain number of times (set by the player) to soften the pluck. Sullivan’s original name for the number of lowpass iterations—squish—is retained in the uBlotar.

Parameter	Instrument Model
bore/string delay lengths (6)	both
bore/distortion gain (6)	both
mouthpiece/feedback delay length (freq.)	both
mouthpiece/feedback gains	both
breath pressure	flute
noise gain	flute
pluck amplitude	electric gtr.
pluck squish	electric gtr.
vibrato frequency	flute
vibrato amplitude	flute
One-pole/one-zero cross-fade coefficient	both/neither
Pre/Post distortion output gain	electric gtr.
Pre distortion gain (drive)	electric gtr.
Limiter threshold	neither

Table 2. Control Parameters for the uBlotar

3 Mapping to the sHowl

The *sHowl* controller is an inexpensive portable speaker outfitted with a dual-axis accelerometer (Analog Devices ADXL202) and a combined linear position and force-sensitive resistor sensor (from Steadlands; www.steadlands.com). Designed for a right-handed guitarist, the linear-FSR is located on the left side of the speaker underneath the four fingers and opposite the thumb. These sensors are wired to a micro-controller (the Basic Stamp IIsx) that produces four MIDI messages corresponding to the sensors (2 from the FSR—position and pressure—and 2 from the accelerometer). These messages are 14-bit, exploiting both the “note” and “velocity” bytes of the standard MIDI “noteon” message; in this way, the full range of the original sensor measurements is preserved. MIDI data is also accessed from the *POD*, particularly from the wah-wah pedal, an effect integral to controlling the electroacoustic effect. The *sHowl* makes the hand-held speaker described above a MIDI controller suitable for controlling projected sounds or additional processes of the electroacoustic signal.

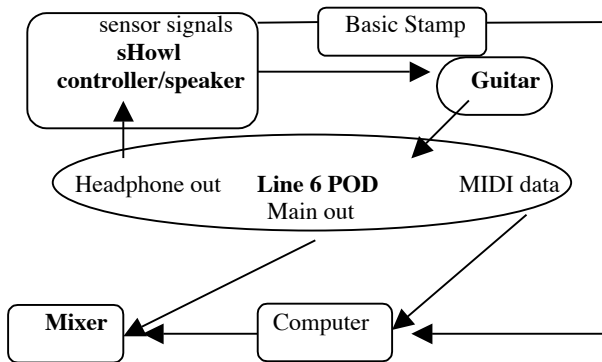


Figure 5. Signal Chain for sHowl controller

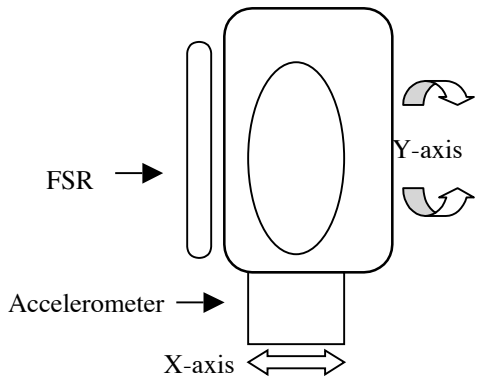


Figure 6. Diagram of sHowl speaker

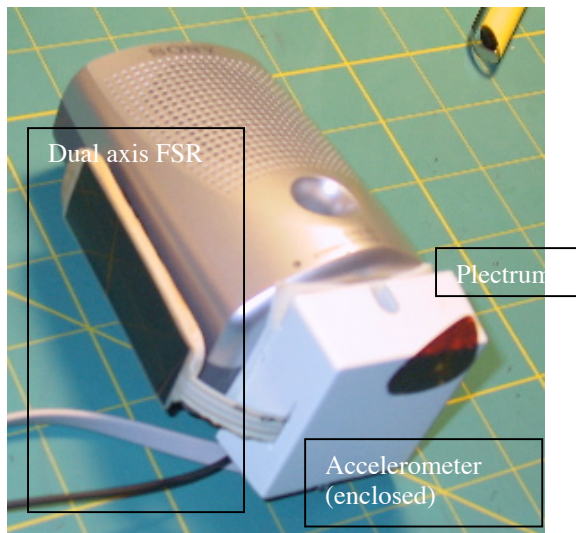


Figure 7. Photo of sHowl

The following two mappings represent some of the possibilities for linking sensors to the control parameters of the *blotar*.

3.1 Mapping *blotar* #1

Figure 5 and 7 show the basic axes and sensors of the speaker-interface. The speaker in a more or less upright (slightly forward-leaning) position is ideal for performing the feedback effect on the guitar signal. Outlined in Table 2, the first mapping for the *blotar* allows for a range of somewhat tamer, mid-to-low-range flute sounds that blend easily with the howl timbre. More extreme *blotar* sounds are produced in positions less conducive to producing feedback with the guitar/speaker system. Adjustments in the wah-wah are mapped onto the filter ratio control of the *blotar*, such that more intense electric-guitar-like *blotar* sounds are matched by higher, more intense howl timbres. In the more quiet range of sounds, *Blotar* parameters are often aligned such that the feedback in the modeled system causes the sound to “build up” over time. Performance of this mapping explores that feature by allowing the speaker, for example, to remain stationary in certain positions while *blotar* parameters feedback and intensify over time—an effect that mimics the guitar-speaker feedback effect. Mapping #1 also allows the player to play the sHowl and blotar effects independently, creating a dialogue between the two.

Parameter	sHowl sensor
bore/string delay length (frequency)	accelerometer Y-axis
bore/distortion coefficient	accelerometer select range of Y-axis
mouthpiece/feedback delay length (freq.)	accelerometer in select range of Y-axis
mouthpiece/feedback coefficient	accelerometer X-axis
breath pressure	accelerometer X-axis
noise gain	none
pluck amplitude	FSR pressure
pluck position	none
body size	none
vibrato frequency	FSR position
vibrato amplitude	pre-set
One-pole/one-zero crossface coefficient	wah-wah pedal

Table 3. Mapping #1 of *blotar* to sHowl sensors

3.2 Mapping *blotar* #2

A second mapping makes more extensive use of the mouthpiece/feedback delay length (expressed as a frequency). The bore/string delay length is triggered by pressure on the FSR, selecting a frequency randomly or from a bank of presets. The X-axis of the accelerometer alters the ratio of the jet/feedback delay length effecting a noisy range of harmonic and wildly inharmonic partials. The Y-axis controls the bore/distortion gain and thus the intensity of the effect. This mapping is effective for slow gestures linked with subtle changes in timbre as well as intense shaking gestures making sounds not unlike a whammy bar on a heavily distorted guitar.

Parameter	sHowl sensor
bore/string delay length (frequency)	FSR pressure triggers preset
bore/distortion coefficient	accelerometer Y-axis
mouthpiece/feedback delay length (freq.)	X-axis controls ratio of fundamental
mouthpiece/feedback coefficient	none
breath pressure	none
noise gain	none
pluck amplitude	none
pluck position	none
body size	none
vibrato frequency	FSR position
vibrato amplitude	FSR pressure
One-pole/one-zero crossface coefficient	wah-wah pedal

Table 4. Mapping #2 of *blotar* to *sHowl* sensors

Both mappings situate pitch as a somewhat unstable feature of the overall timbral quality. While this is largely a compositional choice, it also exhibits/embraces certain limitations in mapping *blotar* sound to *sHowl* sound/gesture. Given the first mapping’s rather quiet, flute-like sounds that effectively morph into louder, more distorted sounds, there is a something of a ritualistic connotation to the palette of timbres available to the first mapping that seems appropriate for introducing the original *Howl* timbre in the course of an improvisation/composition. The first mapping also makes use of the plucked string model, which in practice is a bit noisier than the plucked electric guitar it’s matched with; it resembles a small percussion instrument in this mapping, a woodblock, perhaps. (This timbre is greatly enhanced in the *uBlotar* described below.)

Both mappings explore the opportunity to couple/decouple the mouthpiece delay length with the fundamental frequency. While the first mapping maps the Y-axis to both the fundamental frequency and the mouthpiece/feedback delay length, the two are not coupled by a proportional ratio, as they are in the second mapping. In the first mapping, the mouthpiece/feedback delay length modulates around a stable frequency varied only by the sum of the rate of change in the both X and Y-axes, and is not affected by changes in the fundamental frequency directly. When the fundamental frequency is very high, and not particularly audible, this modulating mouthpiece/feedback delay length provides a muted warbling as the performer shakes the speaker. Contrastingly, the X-axis in the second mapping alters the proportion between the frequencies, which depending on the feedback coefficient, results in a more continuous distorted sound with harmonic partials popping out now and again, like an overblown flute.

3.4 *uBlotar* and *sHowl*

The *uBlotar* is capable of producing a considerable variety of timbres, and appropriately, the structure of its interface suggests a greater variety of control methods than the original *blotar*. As a controller, the *sHowl* has the potential to produce sounds that complement those made by the digital instrument, as it did with the *blotar*. But the sonic and conceptual link between the *sHowl* interface and the original *blotar* was a relatively simple one: the technique for making feedback with the speaker provided a convincing gestural vocabulary for controlling continuous sounds produced by the flute-guitar hybrid. The *uBlotar* brings new advantages and challenges in terms of mapping its control parameters to sensors on the *sHowl*. Perhaps surprisingly, these advantages and challenges have the same origin in the improved design: the enhanced “guitar-ness” of the digital instrument.

The *uBlotar*, as stated, makes greater use of the original Sullivan electric guitar model and thus expands the timbral space of the hybrid digital instrument in a more “plucked excitation” direction. This has the advantage of providing the *sHowl* guitar interface with a range of plucking-percussive sounds denied the guitarist given the original hand-held speaker controller, since the picking hand needs to be holding the speaker to produce continuous feedback sounds. But the role of the speaker interface in producing plucked sounds is less transparent than with continuous sounds. For a mapping to be effective, a just-so relationship between algorithmically controlled parameters and gesturally controlled (both continuous and discrete controls) must be found. For the purposes of this paper, the limitations presented by the *sHowl* controller (and the timbres made independently of the digital instrument)

influence the course of the mapping. Describing a mapping with unique constraints, nevertheless, gives considerable insight into negotiating the control parameters and timbral space of the *uBlotar*.

Moreover, it is interesting to note that accepting such constraints carries aesthetic as well as procedural implications. Researching physical models of sound generation recognizes couplings that are essentially *de-coupled* by the design of a digital instrument (in the form of separable control variables), and which are subsequently *re-coupled* in the process of mapping the variables to a controller (or by otherwise algorithmically relating these variables in a piece of computer music). If using a gestural controller for an instrument like the *uBlotar* remains a priority, one either conceives of an ideal controller specific to the digital instrument or imagines unique mappings for a specific controller. But in either case, *re-couplings* are made in a variety of ways—at a variety of levels—that determine the experience of the sounds in a given context. Naturally, links and associations are made, for instance, through gesture and accompanying sounds as well as through software. While a critical investigation of the aesthetic/compositional implications suggested by these *re-couplings* is beyond the scope of this paper, such concerns provide an impetus to construct a specific framework for interacting with the *uBlotar* and inform the mapping described below.

3.4 Surveying the timbral space of the *uBlotar* and comparing it with the *blotar*

With both the *blotar* and the *uBlotar*, the more sustained feedback/flute sounds can be heard as either distinct from the plucked sounds or as continuations of them. More than likely, a given patch (on both digital instruments) will emphasize a basic timbral exchange in one of two ways: favoring either a single flute-guitar *hybrid* sound or a *dialogue* between flute-like and guitar-like sounds. One of the most crucial enhancements of the *uBlotar* is the inclusion of six bore/string models (as opposed to the *blotar*'s single bore/string). In terms of sonic effect, this complicates this basic timbral exchange. Since the *uBlotar*'s six bore/string models can be loaded with noise individually, there is an array of plucked, "squished," or noisy, flute-like attacks available. When mapping the *uBlotar*, the process of "*re-coupling*" control variables will qualify the timbral space such that the instrument will seem to be, or move between: 1) a strange soloistic timbre; 2) a chorus of like timbres; or 3) an ensemble of different timbres. Given the basic *hybrid-or-dialogue* opposition, the *uBlotar* presents, therefore, a potentially complex timbral space summarized in table X.

Digital Flute-Guitar <i>Hybrid</i>	Digital Flute-Guitar <i>Dialogue</i> <i>of possible ensembles</i>	
A fused flute-guitar timbre, as with the <i>blotar</i> , but one with a greater variety of attack qualities	Sustained noises	Plucked sounds
	Solo	Solo
	Chorus	Chorus
	Ensemble	Ensemble

Table 4. Outline of *uBlotar* basic textures and timbres

3.5 Couple for *sHowl* and *uBlotar*

This timbral space will be traversed no matter what kind of interface one brings to the *uBlotar*. Nevertheless, when using *sHowl* to control the *uBlotar*, as does the piece *Couple*, the electric guitar and feedback timbres enter into the fray as either "one sound among many" or as "particular sounds in dialogue with a few/many." Berlioz's fantasy of the guitar as a miniature orchestra is realized; though, it is doubtful he would regard an ensemble of screeching garbage cans, so natural to the *uBlotar*, as particularly orchestral!

As inferable from the instrument model parameters listed in table 2, continuous control in the form of gestural mapping can affect *either* the sustained sounds (feedback gain/frequency, noise gain, vibrato, etc.) *or* changes in the input of noise (pluck amplitude and "squish," pre-distortion, filter crossfade, etc.), *or both!* Issues brought up earlier influence the emphasis in a particular mapping, as does the "bandwidth" of the performer—the amount of information controllable by a player's range of gestures in a given amount of time. (Cook 2001)

In the mapping strategy outlined below, the six-string guitar model provides an overarching metaphor by suggesting patterns of pitch and timbre modifications. Contrasting pitches, timbres, and articulations are presented in a guitar-like counterpoint; they change suddenly like "block chords" in parallel or contrary motion, or, change one "string" at a time, with oblique motion prevailing. When and how frequencies change, or are repeated, also models guitar technique. If frequencies are to be perceived as discrete and equal-tempered (in this case, they need to sound with fretted guitar notes), tuning the plucked sounds is considerably easier than tuning the sustained sounds. So the mapping outlined below used in *Couple* favors the plucked sound model for producing recognizable notes. (This is also because the guitar-speaker feedback produces an oddly tuned collection of pitches that welcome impossible-to-tune screeches produced by the *uBlotar*).

Figure 8 shows how in terms of gesture and control the availability of contrasting plucked sounds has inspired a simple but effective change in the interface design. Instead of a single pick attached to the speaker, several are attached, with which the performer and listener can associate changes in the plucked timbre.



Figure 8. Photo of “Side, Down, and Bottom” Picks (a fourth pick is at the top of the speaker)

Table 5 lists control parameters of the *uBlotar* with the method of control using the *sHowl* interface. The reference to “pick direction” really refers to the specific pick the player will use to play the electric guitar. Each pick will orient the accelerometer so that the axes are consistently mapped and the player can imagine what timbre the *uBlotar* will be playing when plucking notes on the electric guitar.

<i>uBlotar</i> Parameter	<i>sHowl</i> control and description
bore/string delay lengths (6)	FSR triggers a frequency sent to one of the strings, repeated plucks are triggered algorithmically or by the position sensor underneath speaker-hand fingers
bore/distortion gain (6)	“Sustain” is set algorithmically in both a “block chord” and “one string at a time” manner
mouthpiece/feedback delay length (freq.)	Y-axis switches between a fixed and modulating frequency. “Down” pick = modulating.
mouthpiece/feedback gains	X-axis continuous. “Side” pick position is lowest in feedback gain.
breath pressure	Adjusts slightly depending on rate of change in x-axis
noise gain	none
pluck amplitude	Algorithmic in both a “block chord” and “one string at a time” manner
pluck squish	Algorithmic in both a “block chord” and “one string at a time” manner
vibrato frequency	none
vibrato amplitude	none
One-pole/one-zero cross-fade coefficient	Y-axis switches between filters. “Up” pick
Pre/Post distortion output gain	preset
Pre distortion gain (drive)	preset
Limiter threshold	preset

Table 5. *sHowl* sensors to *uBlotar* control parameters

References

- Cook, P. (1992). “A meta-wind-instrument physical model, and a meta-controller for real-time performance control.” Proc. Int. Computer Music Conference.
- Cook, P. R., and D. Trueman (1999). "Spherical Radiation from Stringed Instruments: Measured, Modeled, and Reproduced," Journal of the Catgut Acoustical Society.
- Cook, P.R. (2001). "Principles for Designing Computer Music Controllers," ACM CHI Workshop in New Interfaces for Musical Expression (NIME), Seattle, Wa.
- Jaffe, D. A. and J. O. Smith (1983). “Extensions of the Karplus-Strong plucked string algorithm,” Computer Music Journal, 7(2):56-69.
- M. Karjalainen, U. K. Laine, T. I. Laakso, and V. Välimäki. (1991). “Transmission-line modeling and real-time synthesis

of string and wind instruments.” Proc. Int. Computer Music Conference.

Karplus, K. and Strong, A. “Digital synthesis of plucked-string and drum timbres,” Computer Music Journal, 7(2):43-55.

Max/MSP: see <http://www.cycling74.com>

Trueman, D., and L. DuBois. “PeRColate,” see <http://music.columbia.edu/PeRColate>

Stiefel, V., D. Trueman, and P. Cook (2004). “Hybridizing the Electric Guitar: through speakers, sensors, and software,” <http://music.princeton.edu/~van>

Sullivan, C.R. (1990). “Extending the Karplus-Strong algorithm to synthesize electric guitar timbres with distortion and feedback,” Computer Music Journal, vol. 14, pp. 26-37.