JamiOki-PureJoy: A Game Engine and Instrument for Electronically-Mediated Musical Improvisation

Benjamin Vigoda
MIT Media Laboratory
20 Ames Street
Cambridge, MA, USA
ben@benvigoda.com

David Merrill
MIT Media Laboratory
20 Ames Street
Cambridge, MA, USA
dmerrill@media.mit.edu

ABSTRACT

JamiOki-PureJoy is a novel electronically mediated musical performance system. PureJoy is a musical instrument; A highly flexible looper, sampler, effects processor and sound manipulation interface based on Pure Data, with input from a joystick controller and headset microphone. PureJoy allows the player to essentially sculpt their voice with their hands. JamiOki is an engine for running group-player musical game pieces. JamiOki helps each player by 'whispering instructions' in their ear. Players track and control their progress through the game using a graphical display and a touch-sensitive footpad. JamiOki is an architecture for bringing groups of players together to express themselves musically in a way that is both spontaneous and formally satisfying. The flexibility of the PureJoy instrument offers to JamiOki the ability for any player to play any requested role in the music at any time. The musical structure provided by JamiOki helps PureJoy players create more complex pieces of music on the fly with spontaneous sounds, silences, themes, recapitulation, tight transitions, structural hierarchy, interesting interactions, and even friendly competition. As a combined system JamiOki-PureJoy is exciting and fun to play.

Keywords

JamiOki, PureJoy, collaborative performance, structured improvisation, electronically-mediated performance, found sound

1. INTRODUCTION

It is not easy to bring people together to create impromptu yet 'good' music. Most people do not know how to play a musical instrument at the level that allows them to fluently express musical ideas spontaneously in a group setting. Furthermore, without adequate rehearsal even highly experienced musicians may find it difficult to play compelling music. Only the most virtuosic of classically trained musicians can even attempt to achieve intense emotional involvement and expression while simultaneously sight-reading a piece for the first time in a performance situation. Only the greatest free jazz musicians can come together on stage without knowing one another or having played together before to improvise a piece that has interesting structure at multiple time scales, tight transitions, structured silences, harmonies, counterpoint, complex dynamics, and other hallmarks of thoughtful composition and prior preparation. The JamiOki-PureJoy system attempts to overcome these challenges and to catalyze people of all levels of musical experience to come together for the first time and within a short amount of time feel comfortable playing highly expressive and interesting music together.

Figure 1: JamiOki-PureJoy players on stage at SIGGRAPH 2006. Each player receives visual instructions by an individualized score at his feet, and audio feedback by his headset.

1.1 PureJoy summary

PureJoy is a multi-person musical instrument that supports on-the-fly sampling, looping, layering, effects, and expressive manipulation of sound. The instrument has been used mostly with players' voice as the source material, but it can sample and manipulate other found sounds as well, supporting real-time collaborative musique concrète. A headset microphone provides audio input, the control interface is a commercially available USB joystick/gamepad, and the behavior is written in Pure Data [5]. Playing a PureJoy instrument, even a person with modest singing ability can create a

NIME07, June 7-9, 2007 New York, NY, USA

broad range of expressive sounds and rhythms. Experienced players can extend the power of their voice with real-time multitrack layering, looping, scratching, pitch shifting, volume control, and effects processing. Conceptually, PureJoy is a signal processing megaphone with looping and layering capabilities and an ergonomic control interface; The player's voice enters and is sculpted by the hands.

1.2 JamiOki summary

JamiOki helps people play a piece of music without requiring them to read sheet music. By giving musical suggestions or instructions through headphones, each player has their own individualized score to follow. Call and response is a basic musical pattern, and JamiOki offers private calls to players, eliciting public responses. The JamiOki system encodes the form of the musical game/composition/structured improvisation in a data structure based on a Gantt chart. This data structure can represent highly complex formal structures involving tasks, dependencies between tasks, types of transitions, and assignment of tasks to sets of players. Players can signal completion of a task using the USB footpad. On a graphical display, players can view who has what assignments, what tasks they are currently involved in, and how their parts relate to the parts of other players.

1.3 PureJoy and JamiOki Together

PureJoy and JamiOki work together to create a complete system for structured musical improvisation. Playing PureJoy alone, we found that people tend to follow a relatively simplistic musical arc. JamiOki injects musical structure into a PureJoy jam. In return, a non-conventional and highly flexible instrument like PureJoy is well-suited to the open-ended improvisational instructions in JamiOki.

2. PUREJOY

Expressivity is a central and enduring goal for new musical instruments and systems, yet the human voice itself is an incredibly expressive instrument. Unfortunately, many would-be singers are uncomfortable with their own voice, and they find it foreign and unpleasant to hear a recording of themselves. It is an inexpensive and nimble audio sampler / looper / manipulation device that flexibly expands the capabilities of the human voice, transforming it in a way that lends comfort to players. PureJoy can also be used with non-vocal sounds, making it a tool for real-time musique concrète. Built to support multi-player use, PureJoy facilitates group-based jamming and performance.

2.1 PureJoy: Motivation and Related Work

Michel Waisvisz performs on an instrument of his own design known as 'The Hands' [22], supporting live recording, playback and two-handed gestural manipulation of voice and pre-recorded samples. Waisvisz has performed with 'The Hands' for 20 years, and his interface is largely unchanged. This deep experience with his own invented instrument has allowed Waisvisz to develop an extreme virtuosity, a rare accomplishment among novel instrument builders. Learning from Waisvisz, we have been playing and performing with PureJoy for two years, and continue to play frequently. Custom-built physical interfaces to software tools like Pure Data, Max/MSP and SuperCollider have grown in popularity in recent years [15] [17].

Although prototyping new musical interfaces with sensors and microcontrollers offers flexibility in interaction affordances, we chose to leverage industrial-scale manufacturing and use a commercial joystick [4] rather than a custom-built interface. This allows others to easily implement the instrument, and in case of mechanical failure we always have a spare instrument backstage.

PureJoy is a collaborative instrument that implements the three properties of shared collective musical control described by Jordà [12]: user-number flexibility (we have performed with up to 6 performers together), user-role flexibility (each performer can simultaneously be responsible for several distinct functional parts with distinct rhythms and timbres in an ongoing performance or jam session), and interdependencies (a shared loop length creates a common 'canvas' that players add parts to; Any player can change the loop length at any time, affecting all other players). As such, PureJoy is an instantiation of 'musical performance groupware' as proposed by Wessel [23]. Other explorations in collaborative, computer-mediated musical interaction include The League of Automatic Composers (discussed in [12]), Jordà's reacTable [11] and Paradiso et al.'s Musical Trinkets [18]. Blaine and Fels [7] discuss a number of challenges that face designers of collaborative musical experiences, including ease-of-learning, type of control, level of cross-modal interaction, and support of virtuosity. A full discussion of how PureJoy fits into Blaine and Fels' classification scheme is beyond the scope of this paper, but one element we have been trying to improve is the tension between ease-of-learning and support for virtuosity, a topic also addressed by Wessel and Wright [24]. We believe that PureJoy supports virtuosity, but it is a bit too complex for walk-up-and-play installations, and we are now experimenting with an interactive tutorial and pre-loading of sound samples to mitigate this problem.

2.2 Description of PureJoy Usage

The player's interface to PureJoy is a USB joystick and a headset microphone. Four of the joystick buttons are dedicated 'track' buttons, each representing a single sound sample that is either pre-loaded or recorded on the fly. The other buttons are function buttons. Pressed in combination with the track buttons, the function buttons control sample recording, loop enabling and disabling, placement and clearing of sample 'trigger points' around a loop, volume adjustment, sound-effect modulation and continuous sound manipulation.

The PureJoy instrument utilizes a single loop that is shared among the players. Each player can record and manipulate their own separate samples, but sounds created by a single player are audible to the others. A new sample recorded by a player overwrites any previously recorded sample in the given track. The new sample is automatically inserted into the loop at the phase when it was originally recorded, so samples recorded in sync with sounds on other tracks will remain in sync, and will be overlaid each time the loop comes around. In the current implementation, samples of up to 10 minutes can be recorded, but if a sound is longer than the loop length and is currently looping it will be truncated each time it is re-triggered. A player can also deliberately trigger playback of a stored sample at any time, and can instantly silence playback from any or all channels. Furthermore, any number of 'trigger points' can be set around the loop for

Figure 2: The joystick interface used for the PureJoy instrument. A PureJoy player also wears a headset/mic that allows for personalized auditory instructions, and vocal or other sonic input.

a given sample, where the sample will be played each time the loop comes around. A player can clear all trigger points for any sample, or alternately can disable looping behavior for a given sample, without clearing the trigger points. Looping behavior on such a sample can then be re-enabled, with trigger points intact. Any player can set the length of the shared loop at any time, which can be a potentially disruptive action, but creates a compelling inter-dependency between players. A new sample can be recorded while simultaneously setting the loop-length, making the creation of a foundational bassline or percussion track a common initial step of a performance.

PureJoy also supports continuous scrubbing and pitch-shifting of stored samples. The scrubbing manipulation seeks continuously through the sample similarly to how a DJ 'scratches' a vinyl record. The pitch-shift manipulation has a four-octave range. Scrubbing and pitch-shifting make use of the left-right and up-down degrees of freedom of the joystick, and they take precedence over any existing looping and trigger points in use.

Finally, reverberation or other effects can be applied to the incoming audio signal, making it possible to record a sample with an effect applied, or to sing directly through the PureJoy with the effect on one's voice.

A drawback of using an off-the-shelf joystick as our musical interface is that it doesn't have as many subtle inherent affordances that might produce usage intuitions as an acoustic instrument [16], or perhaps a hand-built interface. Also, we have have utilized every available button, and have no extras available to easily expand PureJoy's set of affordances. However, as discussed in section 2.1 we gain robustness and the ability to replicate the interface easily. Another limitation of our current implementation is that even with effects applied, the output still sound unmistakably like the player's voice. In the future, we plan to implement more aggressive effects that can completely distort and disguise the player's voice.

3. JAMIOKI

In recent times we have experienced a loss of our shared musical tradition. Put simply, there are very few songs that everyone knows. Jazz musicians are expected to know 'real book' tunes, and most people can at least hum along with the Beatles, but universally known songs are rare. JamiOki seeks to help people come together to make music spontaneously. The goal is to enable a group of strangers to meet and (without significant rehearsal) play impromptu music together that is both emotionally spontaneous and formally satisfying. Like Karaoke, JamiOki is intended to be engaging for people with a range of musical abilities, but JamiOki is not limited to singing the words of songs. The system is built on a data structure and playback engine that is intended to be capable of coding essentially any musical game or composition.

3.1 JamiOki: Motivation and Related Work

3.1.1 Game Pieces

JamiOki is closely related to 'game pieces' such as those created by composers such as Anthony Braxton, Cornelius Cardew, and John Zorn [3], as well as earlier theater improvisation games that were pioneered by Viola Spolin [20] and later by Del Close [10] among others. Musical game pieces emerged simultaneously from the both the free-jazz and twentieth-century classical traditions. The form is alive and well today, and Vigoda has collected and invented a large number of representative examples [1].

If music is viewed as a communications system, game pieces specify the protocol layer, guiding how and when the musicians interact, but generally not providing information about what sounds they should play. In a musical game piece, players follow simple rules to invent their own music together. The purpose of the rules is to restrict the interaction of a group of people in order to create musical structure, while still allowing enough freedom so that people can be creative and spontaneous, and stretch themselves within the boundaries of the game. Game pieces therefore provide a way to compose the structure of an improvisation.

3.1.2 Graphical Scores

Graphical scores use abstract visual forms as musical notation that can be freely interpreted by musicians. They often bear only a passing resemblance to conventional music notation (staff lines, notes, etc.), but still offer visual information for a musician to utilize to create music.

Early visual scores predate computers and include paper scores for electronic music such as those by Stockhausen (Kontakte and Studie II), Yuasa (Toward the Midnight Sun), and Xenakis's Mycenae-alpha. Computerized graphical scores include a system by Eric Lindemann called ANIMAL (ANIMAted Language)[14] that 'combined elements of a graphic user interface management system (GUIMS) and a graphic programming language (GPL).' Like JamiOki this system offers the ability to connect visual forms to musical events. Unlike ANIMAL however, in JamiOki the visualization provides players with information about their status within the musical structure.

A more recent electronic graphical score was created by Hugo Solís García, *Improvisatory Music and Painting Interface (IMPI)* [19]. IMPI allows one person, a conductor/painter, to create abstract imagery in real-time and serve

it over a network to the players with individual screens, as instruction and inspiration for their musical output. Going beyond IMPI, JamiOki facilitates inter-musician interactions, in addition to assigning player actions.

3.1.3 *Improvisational Musical Video Games*

Blaine and Perkis' Jam-O-Drum [9] is based on the metaphor of a drum circle, and helps players synchronize their drumming with one another. Their call-and-response system addresses the 'chaotic interaction and lack of direction' that they observed in early testing, and 'your turn indicators' were employed to 'cue players when to play and when to listen.' Jam-O-World [8] expanded on Jamo-O-Drum, including a game called CircleMaze that used 'two and four bar phrases designed to be recombinant in 256 interlocking configurations.' In contrast, JamiOki-PureJoy allows players complete freedom to create new sounds with their voices, and JamiOki's composition interface, game data structure, and playback engine makes it relatively easy to compose with greater structural complexity than just call-and-response, while still avoiding chaos.

FreQuency and Amplitude are video games created by Harmonix that enable players to create remixes of songs, including different melodies or beat lines, changes in tempo, and modulation of the sound of an instrument. In multiplayer competitive mode, one player plays a musical riff for the other and then the other player must attempt to play it back. In contrast, JamiOki is essentially non-competitive. As in Karaoke, the stakes are simply the risk of embarrassment, and the joy of listening. Furthermore, JamiOki does not use pre-existing musical material other than an optional background metronome beat track.

3.1.4 *Interfaces for Improvisational Grammar*

Wessel used ideas about musical grammar to design a real-time collaborative performance system [23]. His parser would attempt to identify sections of in the music being improvised by the players, so that they could be pulled out and manipulated by other players. Wessel cites Lerdahl and Jackendoff as inspiration.

3.2 Detailed Description

JamiOki is a system for guiding musical collaboration that leverages a data structure for expressing game pieces. The JamiOki system includes a composition design flow for creating games and an audio-visual 'browser' for conducting a group of players through a game piece in real-time.

3.2.1 *Development of JamiOki*

Games for Song is a collection of musical game pieces for groups of children, musicians, or friends [1]. JamiOki replaces the human facilitator for these games with computer mediation. To create the JamiOki system, it was first necessary to create a data structure that could encode any musical game.

There is a long history of research on underlying structures in music. In his lectures 'The Unanswered Question: Six Talks at Harvard,' Leonard Bernstein opened the possibility that Chomsky's theories on linguistics could be applied to finding an underlying universal grammar for music [6]. Lerdahl and Jackendoff took Bernstein's notion seriously and observed recursive 'tree' structures in a large number of transcriptions from Western classical music in their book, 'A Generative Theory of Tonal Music' [13].

Figure 3: A multitrack recording of a circle game

Figure 4: A multitrack recording of a tree-structured game

The first version of the JamiOki system used a recursive tree data structure based on Lerdahl and Jackendoff [21]. This data structure encodes the assumption that if activity A is supporting activity B, then B must begin after A begins, and end before A ends. The listener hears B as a 'sub- routine' of A. If all dependencies in a JamiOki composition obey this type of structure, then the game forms a grammatical tree as described by Lerdahl and Jackendoff.

It proved impossible to represent circle games such as *Floor Head Hum* [1] using a tree-based data structure. Joe Rothermich (one of our group [2]) suggested that Gantt charts might circumvent this incompatibility. Gantt charts are a tool used in industry to organize group projects and other timeline based collaboration. In a Gantt chart representation of a JamiOki game, tasks are musical activity, resources are players, and arrows between tasks are temporal dependencies.

3.2.2 *JamiOki Composition Mode*

To compose with JamiOki, we start by recording a rough edit of a composition using a multitrack sound editor. Figure 3 and figure 4 illustrate this first step in the game design flow.

The rough game is then encoded as a Gantt chart, using available editing software. Circle games are easily represented in Gantt form, as illustrated in figure 5.

Unfortunately, a naive representation of a tree structured game, causes an 'illegal dependency loop' in most Gantt edi-

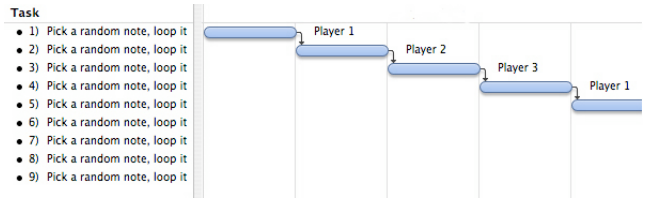

Figure 5: Creating a Gantt chart for a JamiOki circle game

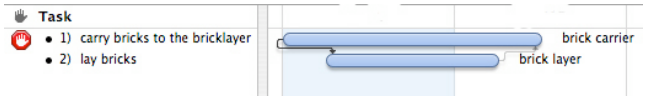

Figure 6: Creating a Gantt chart for a JamiOki tree game

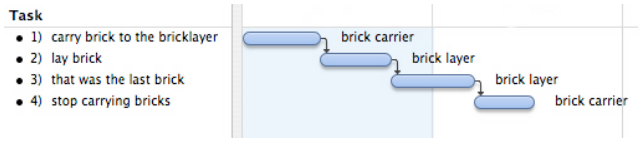

Figure 7: Creating a legal Gantt chart for a JamiOki tree game

tors. In this example, laying bricks (think saxophone solo) is a subroutine of carrying bricks to the bricklayer (think bass line), because the brick carrying activity is necessary to support the brick laying activity. One possible way around this violation is to choose an alternate encoding such as the one illustrated in figure 7. It is also possible in some editors to simply ignore the violation and to export XML to be used in the JamiOki play mode.

Cobra is perhaps John Zorn's most complex game piece. It incorporates nearly all of the elements from all of his preceding game piece compositions. If we ignore the Gantt dependency loop violation caused by tree structured games, and augment Gantt charts to include different kinds of dependencies, such as "slow fade transition" and "abrupt transition," then Gantt charts are capable of encoding Cobra. In general, it seems that extended Gantt charts are capable of encoding essentially any possible musical piece.

Our early compositions so far have tended to sound more like loop-based vocal music rather than classical counterpoint or theme and variations. We have not yet composed and played enough to know whether or not we can readily achieve the density of temporal variation that is possible with conventional musical notation.

3.2.3 JamiOki Play Mode

In play mode, JamiOki acts as a personal music coach or conductor, helping every player in a group create music and jam with the other players. The JamiOki system gives musical instructions over headphones, and also visually displays instructions to players using a projection or screen.

A JamiOki task can be very specific such as, *'now, record and loop this: < audiosample >'* while displaying a few bars of conventional musical notation. Tasks can be more open-ended, such as *'play quarter notes, from this scale < audiosample >,'* or even *'play whatever you like as long as it is percussive.'* JamiOki can also provide an instruction for how to interact with another player or players such as, *'follow the rhythmic lead of player B.'* These interaction instructions can also run the gamut from very specific to very open-ended.

Players also share a multi-button pressure-sensitive footpad which lets them cue transitions that can signal that the player is ready to begin or end a musical activity.

4. PERFORMANCES AND EXHIBITIONS

An early version of the JamiOki-PureJoy system was presented at the Mobius ArtRages Annual Art Party in November 2005, in which six musicians sat in a ring around a projection of tibetan mandalas. The mandalas were displayed as a graphical score, approaching and receding from players to indicate when they should play, and arrows would grow between the mandalas to indicate leader-follower relationships. Merrill played the PureJoy, but other electronic and acoustic instruments were played as well.

For a performance at SIGGRAPH 2006, our visual display was updated to provide a few bars of conventional musical notation to each player, as well as instructions for dynamics and other musical events. This performance marked our first use of a tree-based data structure, based on the work of Lerdahl and Jackendoff.

At the MIT Music Library 'Library Music' exhibition in January 2007, conventional musical notation was eliminated in lieu of auditory call-and-response, and it was the first time a large number of non-musicians played the system extensively. We debuted our Gantt chart composition design flow for games here, creating JamiOki composition mode. JamiOki-PureJoy is currently installed in a large kiosk for continuous public use in the main lobby of the MIT Stata Center. More information, photos and video is available at inventmusic.org [2].

4.1 Conclusions: PureJoy and Jamioki Together

It is not easy to bring people together to play spontaneous yet 'good' music. JamiOki-PureJoy is a novel electronically mediated musical performance system. It pursues the possibility that with the help of a computational architecture, a group of strangers can meet and (without rehearsal) play impromptu music together that is both emotionally spontaneous and formally satisfying.

PureJoy is a musical instrument that uses the human voice or other spontaneously recorded sound as its source material. With a joystick as its physical interface, it allows a player to essentially sculpt their voice with their hands.

JamiOki is an engine for running multi-player musical game pieces. By giving musical suggestions and instructions through headphones and a graphical presentation, each player has their own individualized score.

Without JamiOki, novice players of PureJoy typically record samples into every track, then play with the scratching and pitch bending effects. They rarely create spontaneous silences or sharp musical transitions. JamiOki adds compelling musical structure to PureJoy, so that an impromptu musical 'jam' lasts longer and is more satisfying for the players.

Without PureJoy, novice singers sometimes balk at a JamiOki instruction like 'sing a melody,' feeling that their naked voice is not musical enough. With its layering and effects processing, PureJoy entices users to feel comfortable with the vocal and non-vocal sounds they create. Advanced JamiOki games can require musicians to switch instruments frequently, playing multiple parts simultaneously, which is made possible with PureJoy. Together PureJoy and JamiOki create a cohesive musical video game.

5. FUTURE WORK

In the future we plan to investigate additional integration of JamiOki and PureJoy, enabling JamiOki games to respond to what is being played on PureJoy, for example to advance the structure automatically or to give players feedback about their participation. We may also investigate allowing PureJoy to be muted or otherwise controlled by JamiOki, for example to encourage player cooperation. Further integration of PureJoy status information into the JamiOki visual display may also be useful, and generally we hope to integrate form and function more fully in the visual display. A high priority is to transform the composition design flow into a real-time collaborative activity.

Technical improvements aside, we plan to extend our testing of the system to reach a wider community. This will include installing the system in more locations across the MIT campus, giving workshops with a capella groups, working with artists in residence at MIT as well as professional vocalists and 'beatboxers,' and corresponding with readers of this paper or our blog at inventmusic.org [2].

6. ACKNOWLEDGEMENTS

We thank the inventmusic group for their enthusiastic participation in our technological and musical experiments. Special appreciation to Shawn Hershey, Joe Rothermich and Erik Nugent for their tireless implementation, brainstorming and improvisational testing sessions. We also appreciate John Zorn for thinking with us and directing Cobra with our group of instrument builders at MIT. This work is funded (in part) by the MIT Council for the Arts, the MIT Office of the Arts, MIT Student Life Programs and the Things that Think and Digital Life consortia at the MIT Media Lab. Finally, we thank the librarians at the Lewis Music Library, Pattie Maes, Tod Machover, Barry Vercoe and Joe Paradiso for their support of PureJoy.

7. REFERENCES

[1] `http://www.benvigoda.com`.

[2] Invent music blog. `http://inventmusic.org/`.

[3] John zorn's game pieces. `http://www.omnology.com/zorn05.html`.

[4] Microsoft sidewinder devices make pc games more fun. `http://www.microsoft.com/presspass/press/1999/may99/devicepr.mspx`.

[5] Pd (aka pure data). `http://puredata.org`.

[6] L. Bernstein. *The Unanswered Question: Six Talks at Harvard*. Harvard University Press, 1976.

[7] T. Blaine and S. Fels. Contexts of collaborative musical experiences. *Proceedings of the 2003 conference on New Interfaces for Musical Expression (NIME'03)*, 2003.

[8] T. Blaine and C. Forlines. Jam-o-world: evolution of the jam-o-drum multi-player musical controller into the jam-o-whirl gaming interface. *Proceedings of the 2002 conference on New Interfaces for Musical Expression (NIME'02)*, 2002.

[9] T. Blaine and T. Perkis. The jam-o-drum interactive music system: a study in interaction design. *Proceedings of the conference on Designing interactive systems: processes, practices, methods, and techniques*, 2000.

[10] C. Halpern, D. Close, and K. Johnson. *Truth in Comedy: The Manual of Improvisation*. Meriwether Pub, 1993.

[11] S. Jordà. Sonigraphical instruments: from fmol to the reactable. *Proceedings of the 2003 conference on New Interfaces for Musical Expression (NIME'03)*, 2003.

[12] S. Jordà. Multi-user instruments: models, examples and promises. *Proceedings of the 2005 conference on New Interfaces for Musical Expression (NIME'05)*, 2005.

[13] F. Lerdahl and R. Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983.

[14] E. Lindemann. Animala rapid prototyping environment for computer music systems. *Computer Music Journal*, 15(3):78–100, 1991.

[15] D. Merrill and J. Paradiso. Personalization, expressivity, and learnability of an implicit mapping strategy for physical interfaces. *In the Extended Abstracts of the Conference on Human Factors in Computing Systems (CHI'05)*, 2005.

[16] D. Merrill and H. Raffle. The sound of touch. *In the extended abstracts of CHI 2007*, 2007.

[17] D. Overholt. The overtone violin: A new computer music instrument. *Proceedings of the 2005 International Computer Music Conference*, 2005.

[18] J. Paradiso, K. Hsiao, and A. Benbasat. Tangible music interfaces using passive magnetic tags. *Proceedings of the 2001 conference on New Interfaces for Musical Expression (NIME'01)*, 2001.

[19] H. Solıs. Improvisatory music and painting interface. Masters thesis, Massachusetts Institute of Technology, 2004.

[20] V. Spolin. *Improvisation for the Theater: A Handbook of Teaching and Directing Techniques, Including a New Appendix*. Northwestern University Press, 1999.

[21] B. Vigoda. Mandala: A visual grammar for ensemble musical improvisation. *International Conference on Computer Graphics and Interactive Techniques, ACM SIGGRAPH Art gallery*, 2006.

[22] M. Waisvisz. The hands, a set of remote midi-controllers. *Proceedings of the International Computer Music Conference (ICMC'85)*, pages 313–318, 1985.

[23] D. Wessel. Improvisation with highly interactive real-time performance systems. *Proceedings of the International Computer Music Conference, Montreal, Computer Music Association*, 1991.

[24] D. Wessel and M. Wright. Problems and prospects for intimate musical control of computers. *Proceedings of the 2001 conference on New Interfaces for Musical Expression (NIME'01)*, 2001.