SOUND PONG: AN INTERACTIVE EXPLORATION OF SONIC SPACE

Jon Bellona *Jeremy Schropp*

University of Oregon
Intermedia Music Technology
1225 University of Oregon
Eugene, OR 97403

ABSTRACT

Sound Pong is an electronic ensemble composition for four performers using four Wii-motes and four pairs of Rec-Specs. The eight-channel work takes an historical look at the gaming experience through the use of modern controllers set inside a classic 8-bit aesthetic. Similar to the early video game *Pong*, or Robert Rauschenberg's *Open Score*, an object is hit between players from in and around a dictated space. The sound field outlines the audience space and, by placing the performers within this space, helps to fuel audience interaction. A game-like interface projects onto the front wall, fusing both audience and performer spaces together, while simultaneously augmenting the audience's interactive sensory experience.

1. INTRODUCTION

Since the establishment of multi-channel audio, composers and artists have looked at controlling the location of sounds in space. Pierre Henry developed the *pupitre d'espace* for the dissemination of sounds via induction coils, and Karlheiz Stockhausen used a rotating amplifier to distribute sounds for the performance of *Gesang der Jünglinge*.[1] In Rauschenberg's *Open Score*, performers volleyed sounds in space by using FM radio signals emitted from transmitters set in tennis rackets.[2] Rauschenberg's work applied existing metaphors and pre-established rules; however, the outcome of the work was a collaboration between technology and performers, existing with an element of indeterminate chance.

The use of electronics and the computer for "aesthetical expression" appears in early works by Frieder Nake and Manfred Mohr, both of whom utilize the computer to draw repetitive patterns that following a set of parameters with a degree of randomness.[3] The development of the GROOVE system at Bell Labs in 1968, extended continuous control of musical parameters to human touch, and Maxwell Ghent's *Phosphones* (1971) brought performers, music, and technology together.[1] The development of *Tennis For Two* (1958), and later the iconic *Pong* (1972), displayed visuals moving throughout virtual space in real-time, controlled by the interactions of users.[4] The video game propels ideas about controlling visuals over time through user input. Since the release of *Pong,* the gaming experience has evolved from single player arcade games to the home gaming console to online play with millions of others [5]. The re-appropriation of devices and video games has become a popular theme among artists, in particular Mary Flanagan, Joseph DeLappe, and Cory Arcangel. Arcangel regularly hacks old NES games for installation art.[6] The Nintendo Wii gaming system, introduced in 2006, bases a majority of its interactive games on simulating sports' gestures. Re-appropriating the Wii-mote as a musical controller for *Sound Pong* emblematically enforces the gaming experience and the use of 8-bit culture, for which the Nintendo (NES) first became popular.[7] *Sound Pong* combines early sound distribution ideas with the interactive nature of performance through the visual aesthetic of classic computer games. This compositional approach merges various disciplines and cultures into a cohesive work that encourages audience participation and simultaneously challenges traditional musical conventions.

Underlying musical structures frame *Sound Pong*; however, like Rauschenberg's *Open Score*, the performance is a "formal dance improvisation."[8] *Sound Pong* shifts away from traditional concert hall music through both performance art concepts and popular cultural metaphors. The performance challenges conventions of concert performance by using gestures of sports and the language of gaming experience to help build its structure. Players and observers engage in a real-time interactive work that employs a simple visual confirmation display to augment the sound experience.

2. CONCEPT & PROGRAMMING CHOICES

Our initial concept was to translate a virtual, moving ball into a controller of sound distribution. Looking into programming solutions, we found a simple bouncing ball example via Processing[1], which could serve as an algorithmic model for our virtual ball.[9] Using a moving ball as a sound controller was also a performance concept, where multiple performers could play with and pass

[1]Processing is an open-source programming language and programming environment used in multiple disciplines including education, animation, and interactive installations. More can be found online at http://processing.org

sounds around inside a given space. In thinking about volleying sound, we looked to Nintendo Wii-motes as a performance instrument since the controller enforced our performance ideas, and the ease of routing Wii-mote data to the computer via OSCulator[2], the data stability of the Wii-motes, and the wireless connection of multiples Wii-motes to a single computer, created a stable performance solution.

We decided early on to use Kyma[3] for our audio processing, and in order to manage the number of software programs used in the composition, we chose to use Max/MSP[4] for our data hub, as we could manipulate both Wii-mote data and the virtual ball code within one application. Because we collected Wii-mote data via OSCulator, we decided to also use OSCulator to send data inside Max to/from Kyma with 'udpsend' and 'udpreceive' objects.

3. MAPPING DATA

3.1. Program Translation and Basic Control

First, we translated the bouncing ball Processing code to Max. Using 'pictslider', 'value', and 'if' objects, we were able to visualize the code in a short amount of time. Bouncing a ball within a confined space inside Max also gave us vector information, (x,y) coordinates, which we could send to Kyma for controlling the distribution of sounds. Only after basic control of the ball motion and sound distribution was established, did we begin mapping Wii-mote data as controls over the virtual ball.

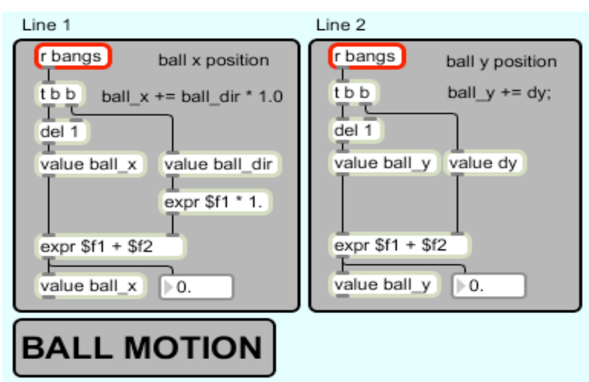

Figure 1. First two lines of Processing code translation inside Max/MSP.

3.2. Wii-mote Data

There were two types of specific information that we wanted from the Wii-motes: button triggers and continuous accelerometer data. We mapped button triggers as controlling the direction of the virtual ball (i.e. "hitting" the ball), and as triggers of sound events. Button triggers helped us to realize the piece through selection of sound banks and activating section changes.

Wii-mote accelerometer data provided support of performance gestures. We first measured the speed of arm swings as a performer 'hit' the ball. We mapped the speed measurement onto the velocity of the ball, whose position was subsequently mapped onto a sound's location in space. For example, if a performer swung quickly, the ball would move faster across the performance screen, and the related sound would subsequently move faster across the space. There was always a direct correlation between action, animated motion, and sound distribution.

3.3. Kyma

Kyma supported our two major needs: triggered sound events and continuous panning control within a multi-channel environment. As noted earlier, vectored coordinates of our virtual ball controlled the panning location of sound, which was accomplished using Open Sound Control (OSC)[5] messages and mapped to the Angle parameter inside a 'MultiChannel' Kyma sound object. To further increase the auditory experience of sounds moving throughout space, we algorithmically simulated a Doppler effect, placing the effect on all moving sounds.[6] All scaling of data was done inside of Max before sending this information over to Kyma. Sound events were triggered with MIDI, and unique sounds were assigned to each performer.

---

[2]OSCulator is a software application that connects hardware devices with software using various communication protocols, including Bluetooth and OSC. More can be found online at http://osculator.net

[3]Kyma is a sound design environment that supports real-time sound manipulation and multi-channel panning control. More can be found online at http://symbolicsound.com

[4]Max/MSP/Jitter is a graphical programming environment for controlling music, media, and video. More can be found online at http://cycling74.com

[5]Open Sound Control (OSC) is a stable, 32-bit protocol used for interconnecting hardware controller devices to the computer, as well as software on one or more computers using local networks. More can be found online at http://opensoundcontrol.org/

[6]It should be noted that we used Kyma Sound Library's Doppler shift effect, which uses three 'DelayWithFeedback' sound objects. We did control the hot variable, !Pan, through the location of our moving ball via OSC messages.

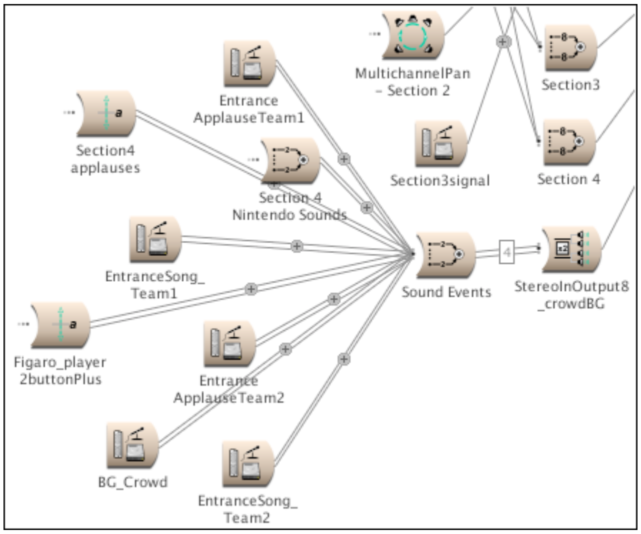

Figure 2. Kyma sound objects used in *Sound Pong*.

4. COMPOSITION PROCESS

4.1. Concept to Composition

Because our initial concept, a virtual ball controlling sound distribution, was paramount to the execution of the piece, we waited until we had a working algorithmic model before delving into the composition. From our working model of a virtual ball, to continuous control over sound distribution in our space, we moved forward with the generation of sound materials and the development of a performance that played with the language of video games and audience participation.

4.2. Sound Material

There are two types of sounds inside *Sound Pong*: sound events controlled by section changes, and performance sounds controlled by the performers. Sounds associated with events include opening music, incidental crowd noise, and 8-bit designed sounds triggered for emphasis of both musical and non-musical events.

Outside of sounds and music associated with particular events in *Sound Pong*, there are a total of seventeen performance sounds. These performance sounds consist of four banks of four sounds each, and each bank has a unique theme. Each player has control over four sounds, one in each bank of sounds. Only one bank of sounds may be accessed at any given time in order to keep related timbres together. The seventeenth performance sound is reserved for the second section of the piece, a tennis ball sound, which playfully acknowledges the aesthetic framework and historical precedence from which the piece is derived.

4.3. Scoring *Sound Pong*

The final compositional structure manifested itself through system trials, conceptual discussions, and practices with the performers. The compositional process became a collaborative effort as the technology, performers, and our ideas about performance all informed our programmatic and compositional decisions.

5. MUSICAL STRUCTURES

While *Sound Pong* enables improvisation and leaves the outcome of the piece open-ended, there is an underlying musical structure to the work. The piece is broken down into four sections, each with a distinct musical objective. The exposition of all sound material comprises the first section, where time is given for each player to reveal the sounds in each of his/her sound bank. Player One has control over the timing between the exposition of each sound bank, triggered by the Wii-mote Left and Right buttons.

The second section is a short, humorous section meant to engage the audience and recognize the inherent nature of the video game aesthetic. The virtual ball becomes an on-screen tennis ball and the performance sound changes to a realistic, tennis ball sound. The four players exchange volleys as if warming up before a real match.

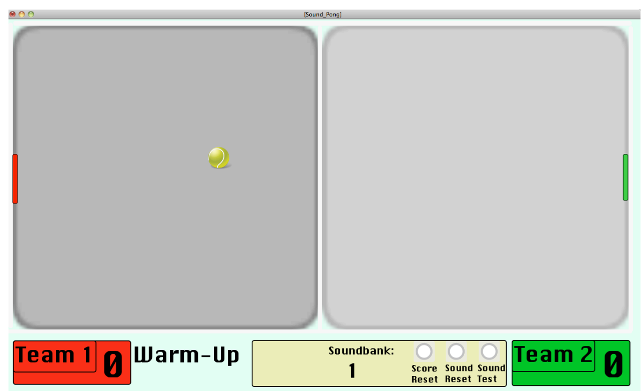

Figure 3. The performance screen projection, displaying section two.

The third section serves as a development of action, where sound banks continually change, the virtual ball is sped up to simulate tension and rise of action, and the performers are free to move about the space, disregarding normal boundaries established by volleyed sports. There is no competition here, only quick shifts in timbre, panning, and movement by performers. The third section climaxes with a triggered sound event by Player Two, a held operatic note, and cut short by an 8-bit musical phrase, which signals the introduction of the fourth section, the competitive match.

The competitive match is an indeterminate section, where the first team to get to seven points wins and also ends the piece. Not only does an 8-bit sound signal the start of the game, but the on-screen display flashes “Game On.” In order to propel musical action within the fourth section, sounds are sped up after pre-determined amounts of time. Thus, scoreless action results in faster moving sounds and motions by the performers. The final match point, the concluding event of the work, is emphasized with both an 8-bit sound and a theatrical performance shift, where all four performers arrive at center stage for a final bout. The rapid action of sound and performers here brings the entire piece to its ultimate climax; not only does the outcome of this event determine the winner, but also triggers the conclusion of the piece.

Figure 4. Performers during the final match point.

Composing theatrical moments into the piece helped to accentuate the moments of rising action and demarcate section changes. By building in moments throughout the piece, for instance, the team entrances during the introduction, the dramatic end to the development section, and the center stage battle of the match point, the audience could more easily follow the performance. Both sound and visual cues were used to designate section changes, thus providing clear demarcations for all performers too.

6. AUDIENCE PARTICIPATION

Because performers and audience members were placed inside the sound field together, preconceived notions about performer-audience relationships subsided. The social and, at times, competitive nature surrounding video games enhanced ideas about engaging audience members. We made several choices to increase the sensory experience of *Sound Pong* through audience participation.

Performers were encouraged to entertain, talk, and mix with the crowd during the performance. Since performers were directly hitting sounds through, over, and around the audience who also resided inside the sound field, the piece was a shared, interactive experience.

Performers were divided into two teams identifiable by color, and corresponding colored cards were distributed randomly inside concert programs. By making the final section an indeterminate game and assigning audience members a specific team, audience members were encouraged to rout and cheer for ‘their’ team, building off ideas of social acceptance and belonging.

Figure 5. Audience members cheering with scorecards.

Audience members were given multiple outlets for engaging with the work. Performers moved in and around the audience. Auditory sounds moved through the entire space. A visual display of the virtual ball reinforced game aesthetics and the movement of sounds. Musical cues and an on-screen display designated section changes, informing both the performers and audience members.

Composing a non-traditional concert piece came with its challenges. For instance, non-traditional concert works and performance contexts lack an historical framework that would typically inform audience etiquette. We placed fellow composers within the audience to help suggest and reinforce “appropriate” audience behavior, easing the psychological transition of participating as an audience member inside a concert hall setting.

7. OTHER CHALLENGES

As a result of our choice to model video game aesthetics, another challenge lay in designing a system that acted similar to common and familiar cultural perceptions of video games. While the vocabulary of video games helped us dictate what should and should not be included within the system, ironing out the functions of what users expect from a system proved challenging. For instance, any one player could rapidly press the B button in order to play defense for their team. As soon as the ball crossed the court line into the opposing team’s territory, the ball could immediately be returned, and it would be almost impossible to score. Coding in a trigger delay function to help the offensive team, as well as the progression of the musical movement, was crucial. Coding behaviors, like the example above, supported the performance structure and aesthetics. For example, we programmed a natural acceleration algorithm, where the ball's motions would increase over time if no one team scored. While initially coded to ensure that a team would score and the piece could progress, the acceleration algorithm also augmented and enhanced the performance, increasing dramatic tension the longer both teams went scoreless.

8. PROGRAMMING FOR PERFORMANCE

We learned valuable lessons by working with performers of data-driven instruments and the instruments' associated technology. This section is devoted to sharing our observations and insights, in the hopes of furthering dialogue about the development of performance practice using data-driven instruments.

8.1. Tuning Instruments

Similar to an acoustic performance that tunes its instruments before playing, we incorporated a digital performance tuning practice. Since the performers of *Sound Pong* could not bow or blow on their instruments to ensure proper tuning, we included two data confirmation tests, one visual and one auditory, which could be run before each rehearsal and performance. We tested our instruments with a confirmation of data entering the computer and with a confirmation of a triggered sound, before initial stage entrance, in order to ensure a smooth start for each and every performance.

8.2. Section Leaders

Determining a section leader per team and a leader for the entire piece was important to the flow of the work. Specific control functions were assigned to each section leader, allowing us to relinquish compositional control over section changes, sound-bank selection, and triggering of climactic events. This organizational decision enabled additional players to focus solely on their performance. Moreover, by having each team leader control the performance structure functions, *Sound Pong* could potentially be realized by only two performers.

8.3. Performance Instructions & Order

Knowing that technology can falter, having a set performance structure helped us minimize the variables between rehearsals and the performance hall. A visual set of instructions ensured a consistent and familiar set-up practice, and a performance screen provided visual cues to both the performers and audience. These performance instructions serve as our documentation, so that anyone could re-stage *Sound Pong* and ensure an easy set-up.

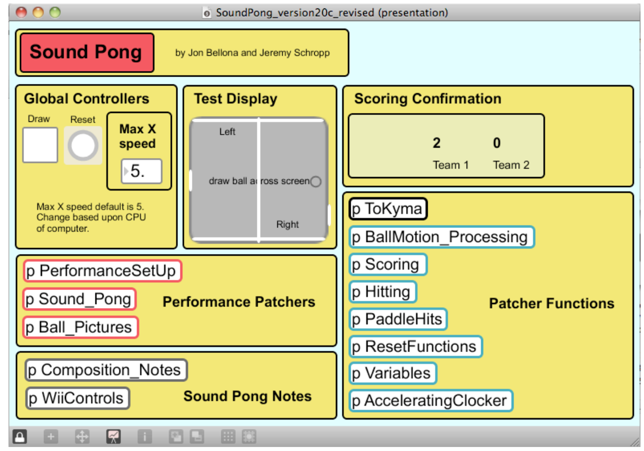

Figure 6. Opening *Sound Pong* patch, with set up instructions and necessary performance modules collected together.

8.4. Computer Technician

Composers are often their own technicians, so much so that we often tend to overlook the vital importance of this role. Setting up local area networks for Bluetooth messages; cueing section changes during a performance; even plugging in USB devices; all are technical factors related to music involving technology. Performances that involve multiple performers and potentially multiple computers may call for an established technician role. In *Sound Pong,* while both composers could adequately handle the computer setup and execution of the piece, we designated one composer as the computer technician. This role helped us to formally distribute responsibility during the rehearsals and performances, which allowed the other composer to focus on additional performance needs.

8.5. Modular Programming... Cleanly

While most who work with Max/MSP revel in the use of the presentation mode to sweep our patch cord messes under the carpet, working with human performers doesn't necessarily allow for this luxury of messy code. Since composition ideas for *Sound Pong* came through live performer rehearsals, the ability to rapidly modify functions and performance mappings during rehearsals was a must. By isolating functions into separate modules and simultaneously keeping our underlying Max patch well-commented and as clean as possible, we maximized our time with performers and accelerated our experimentation with new ideas. Both of these factors increased the time spent working with the human performance details of *Sound Pong* while concurrently increasing the trust performers had with the technology.

9. CONCLUSION

We judged the success of the performance more upon user experience than the technical execution of the piece. Since the nature of video games is less about technical virtuosity than playful competition and teamwork, we attempted to make the work naturally playful for both the performers and the audience. By layering the performer and audience space atop each other, the resulting environment allowed the audience to participate from within. The music serves the function of the game, the game environment, and helps to engage the audience in a new and interesting way.

In order to prompt further discourse as applied to the use of Nintendo Wii-motes in ensemble performance, we have created and published a mapping interface for Wii-motes online.[10] The open-source Max patch handles up to four Nintendo Wii-motes at any given time. The interface allows anyone to quickly and efficiently map Wii-mote data in a variety of creative applications, while also serving as a potential performance module. In addition, we published the *Sound Pong* source patches online for future deconstruction and discussion.[11]

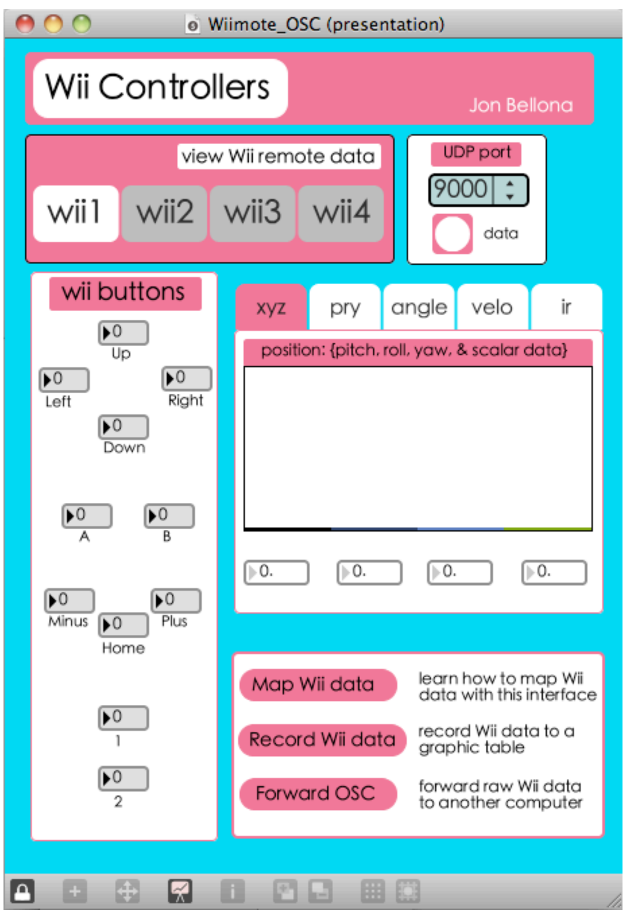

Figure 7. Our open-source Wii-mote Max interface available online, created for easy data mapping with Wiis.

10. REFERENCES

[1] Chadabe, J. *Electric Sound***.** New Jersey: Prentice Hall 1997.

[2] Rauschenberg, R. *Open Score by Robert Rauschenberg, 9 Evenings: Theatre and Engineering*. New York: Experiments in Art and Technology and ARTPIX, 2007.

[3] Reas, C., McWilliams, C., LUST. *Form+Code*. New York: Princeton Architectural Press, 2010.

[4] Hunter. W. “From ‘Pong’ to ‘Pac-man’,” in DesignBoom. http://www.designboom.com/eng/education/pong.html . Last accessed March 6, 2012.

[5] Takahashi, D. “Activision Blizzard earnings: Call of Duty Elite snares 7M subscribers,” in Venture Beat. http://venturebeat.com/2012/02/09/activision-blizzard-earnings-call-of-duty-elite-snares-7-million-subscribers/. Last accessed March 7, 2012.

[6] Arcangel, C. *Artist Web Site*. http://www.coryarcangel.com/things-i-made/ishotandywarhol/. Last accessed March 8, 2012.

[7] Sheff, D. *Game Over*. New York: Random House 1993.

[8] Kuo, M. “Open Score (review),” in *TDR: The Drama Review*, MIT Press, Vol. 52 no. 4 (Winter 2008), pp. 194-196.

[9] Reas, C. “Bounce,” in Processing. http://processing.org/learning/topics/bounce.html. Last accessed March 10, 2012.

[10] Bellona, J. *Wii Controllers: GUI for Max/MSP*. http://deecerecords.com/projects#wiimote. Last accessed March 12, 2012.

[11] Bellona, J., Schropp, J. *Sound Pong* source patches. http://deecerecords.com/music#soundpong. Last accessed March 12, 2012.