Music Box: An Algorithm for Producing Visual Music

Lindsay Grace
Armstrong Institute for Interactive Media Studies
School of Fine Arts
Miami University
Oxford, Ohio, U.S.A.
LGrace@MUOhio.edu

This material is published in the IEEE proceedings of the International Conference on Advances in Computer Human-Interaction (ACHI, St Marteen, 2010)

http://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=4782475&isYear=2010

***Abstract*— This research proposes a method for producing music via visual composition in a computer-game like environment. This is accomplished through the development of artificial intelligence software that applies the visual rules of standard emergent behaviors to the algorithmic arrangement of musical tones. This research presents the proposed system, defining the algorithm and demonstrating its implementation.**

Keywords-User Interfaces, Music, computer graphics, computer games

I. Introduction

As the ubiquity of interactive media increases and the cost of implementing such technologies decreases, new opportunities in creative explorations develop. One such opportunity is the application of new systems of artistic creation. This paper describes the implementation of a popular emergent behavior algorithm to the generation of real time music. The result is a software system that marries the engaging qualities of a video game with a distinctly original music composition experience. This paper is structured in three parts, an overview of Related Word, the specifics of the System Design, and the details of the final Implementation.

Related Work

In 1987, Craig Reynolds proposed a now widely adopted algorithm for imitating artificial life. The seminal paper, entitled Flocks, Herds, and Schools: A Distributed Behavioral Model [1] described a highly effective algorithm for simulating emergent behavior in real time animation. The research provided here applies Reynolds's visual algorithm to the production of sound. In short, it takes what Reynolds accomplished in simulated visual aesthetics, and offers its compliment in simulated acoustical aesthetics.

The concept of using the patterns of visual flocking to produce musical tones is not entirely new. Most recently the 2007 artistic work, Flock by Jason Freeman [2], demonstrates the potential of coupling real time spatial data to music direction. In this case, the music is not generated by the patterns of flocking; it is instead a translation of participant location into a musical score. Each participant moves, is tracked, and their movement indicates musical direction employed by instrumentalists.

The work of Golan Levin also offers some overlap in the area of tying real time, interactive visual experiences with the production of sound. Levin's internationally awarded Audiovisual Environment Suite [3] provides five interactive systems which allow people to create and perform abstract animation and synthetic sound in real time [4]. Each of the five works uses user interaction to produce music, but none uses emergent behavior principals specifically. Instead, the production of sound is directly linked to a translation of user behaviors. If the user moves, or directs the movement of the agent within Levin's environment the music is directly effected.

Applying emergent behavior to the production of music changes the character of these interactions in a way the aforementioned works have note. Instead of providing direct agency, the user of an emergent musical system is simply wafting the musical composition. The system defined and implemented in this research is not a dependent on user interaction, but merely affords for it. Like traditional emergent behavior, the musical system has the ability to run without interaction, but merely affords it.

The work of Golan Levin and Jason Freeman do not exploit the existing natural harmony of flocking, but instead seek to derive compositional cues from the behavior of their participants. Even when the composition of music or visual performance is not participant driven, it is rarely entrusted to the patterns of simulation. Michael Lew's work in designing an instrument for cinema editing as a live performance [4] approaches the performance characteristics this project pursues, but it continues the tradition of depending on the user input to propel production. The simple system outlined in this research works instead toward autonomous music generation.

From an art-historical perspective, Alvin Lucier's I Am Sitting in a Room [5] is a noteworthy reference from which to begin an understanding of the Music Box system. In this piece and composition the composer records himself narrating a text, plays the recording back into the room, and re-records it. Each successive recording accentuates frequencies until the narrated text ceases to be intelligible, and instead becomes a set of harmonies and tones.

I Am Sitting in a Room is at its heart an algorithmic musical composition. It is a real world application of recursion and exploitation of the properties of sound when technology is allowed to intervene in arrangement. The beauty of the work is produced when technologies' inability to duplicate the natural world is exposed. Likewise, Music Box is about allowing the technology of artificial intelligence software to intervene in the composition of music. It is not an attempt at imitating the natural world, as a bird call seeks to imitate the bird. Instead, it is an exploitation of the set of algorithmic rules for the presentation of visual information to make music. Where Alvin Lucier's compositional score relied on text, Music Box relies on Craig Reynolds' steering behaviors for the simulation of intelligent life simulations.

This coupling of audio and visual is also an inversion of the research and artistic experimentation on synesthetic color organs conducted by previous artists and scientists as early as 1893. Further reading on Telemann and Rameau for example, will illuminate the qualities of the clavecins oculaire [6]. Most importantly, where the tradition of color organ research is largely dependent on an understanding of the physical characteristics of light and sounds waves, Music Box is investigating visual behavior and its relationship to musical behavior. Just as music is said to dance, bounce, or travel, Music Box looks to create an auditory experience from the visual behavior of the simulation.

Although these examples vary widely in discipline, their foci are fairly united. Each is a scientifically informed experiment into the character of music creation.

II. System Design

A. *Visual Algorithm*

The algorithm presented here is the result of several iterations of designing an experience that was both functional and enjoyable. The results were refined after several qualitative evaluations from both public performance and academic review.

The key goals in defining an enjoyable experience were remaining faithful to emergent behavior in the production of sound, avoiding monotony, affording real time interaction, and preserving fundamental qualities of harmonious sound.

The algorithm employs Reynolds' key rules for steering behaviors. These are separation (V1), alignment (V2), and cohesion (V3) to produce the visual simulation. These rules, illustrated in figure 1, dictate the movement of each individual element in the simulation, calculating three-dimensional vectors for the direction and rate of movement.

The fundamental guide for the Reynolds algorithm is the derivation of emergent behavior through balancing these three steering behaviors. An implementation of the algorithm calculates a minimum of three vectors for each agent in the simulation. Each vector calculation reflects a single logical goal. The separation vector reflects the individual agent's goal to avoid crowding other agents. The alignment vector reflects the agent's goal to steer in the same direction as other agents in its group. The cohesion vector applies the agent's goal to move toward the average position of other agents in its group. When these three vector calculations are averaged, each individual agent mimics the basic behavior of flocking birds or a school of fish. To increase the quality of the simulation, additional logical goals can be applied and calculated as vectors. Additional goals include avoidance and territorial bounds.

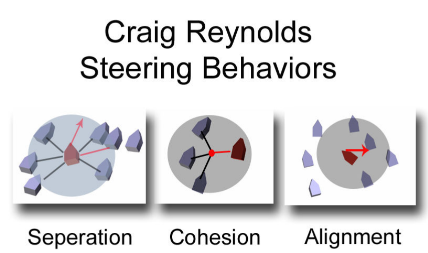

Figure 1: Reynolds Steering Behavior model in 3D

For this implementation, goal setting (V4), territorial bounds (V5), and perching (a simple timer) were added. To create a predator and prey situation, each agent was provided either a predator or prey designation. The predators were given the goal (V4) of the nearest proximal prey.

In total, 5 vectors are calculated for each element, during each step of the simulation. For prey, a sixth vector for avoidance (V6) is added to the final movement calculation. Avoidance was calculated as the inverse of goal setting. To disrupt overly unified behavior, each individual element is also assigned an individual deviation multiplier (DM). When all vectors are calculated and weighted via the deviation multiplier, the final velocity is calculated. Final velocity is the sum of all vectors and the individual element's current velocity.

Movement for prey is thus calculated:

V1 = V1 * DM
V2 = V2 * DM
V3 = V3 * DM
V4 = V4 * DM
V5 = V5 * DM
*V6 = V6 * DM*

Velocity = current velocity + V1+V2+V3+V4+V5+V6

The analogical basis of prey-predator envelops the simulation in an engaging mini narrative, as intention is further emphasized by watching prey narrowly escape their predators. It also allowed the simulation to avoid a sense of randomness that may result from emergent behavior simulations in which individual elements lack goal setting. By providing the prey-predator simulation, the system can run its course independent of stimuli outside the simulation. In essence, it remains autonomous.

B. *Musical Algorithm*

To create music, each element in the simulation is assigned the same, single note audio tone. Predators receive a harmonic tone (e.g. a flute or voice), prey receive a rhythmic tone (e.g. a drum or bass). Each tone is given a randomly assigned frequency pitch at the start of the simulation. Although the system loads very few sounds, the diversity of sound is achieved initially by changes in initial frequency.

When the simulation begins, all elements repeat their tone indefinitely. Predator tone pitch is individually calculated as a factor of the element's spatial velocity in three dimensions. The faster the visual element moves, the higher its pitch climbs. All predator pitches are capped at 44,000 hertz. Prey pitch is not changed during the simulation, to provide a constant rhythmic basis.

The basic formula for musical tone generation in the system is:

Pitch = (Velocity X + Velocity Y + Velocity Z) + Initial Frequency

The aforementioned calculations are made every rendered frame, or a minimum of 30 times per second. Each tone is repeated at the end of its iteration. If, for example, a tone is two seconds long it will be repeated every two seconds.

All elements are also subject to a framing parameter and standard attenuation simulation. In the framing parameter, all non-visible objects are omitted from the audible experience. The attenuation simulation employs standard 3D spatial audio cues, making elements nearest the camera's end of the viewing frustum more prominent than those further from it. These additions emphasize the relationship between the visual performance and the music generated. They also support fairly intuitive user interaction.

III. IMPLEMENTATION

A. *Software Design*

The software system is executed in real-time and implemented using Microsoft's DirectX technology, programmed against the Blitz3D computer game engine. It requires a minimum of a 2GHZ AMD processor, with 2MB RAM. The system is managed before and during run time. Before run time a user can edit the simulation's text-formatted configuration file to change the quality of the music tones being performed and their interplay. Factors include the number of prey and predators, the tones within the performance, the visual range of the simulation, the tempo of the performance, and the weight of each vector on the final visual simulation. Tempo is managed by changing the play length of each tone assigned to the elements in the simulation.

During the simulation, users can manage a variety of factors that change the quality of both the visual performance and the music performed. Run-time control is managed through the computer keyboard or through a 10-button joypad as described in table 1 and figure 2.

Table 1. Run-time user composition control

Figure 2. Diagram of run-time control scheme

For demonstration, a single source file in .wav format was used for each predator element. Prey tones were generated from a list of 10 .wav formatted files. The specific audio file chosen is either randomly selected at the start of each simulation or assigned by the user through pre-simulation configuration.

B. Demonstration

The final demonstration of the algorithm was developed as an interactive performance. For analogical clarity, the prototype version of the systems was developed as a set of musical notations. Each quarter note, eight and sixteenth note in the notation represent a predator, and each bass clef represents prey. The demonstration begins when the musical notations ascend from two dimensions to three. The notes resound as they race around the clefs, seeking them out, but never actually catching them.

The demonstration uses a single violin pluck as the base tone for predators. Prey is provided a variety of acoustic rhythmic taps from a variety of drums or a repeated melody performed on the bass. These prey sounds include a five note melody performed on the viola, a tonal African drum tap in 3 tones, and a 5 tone bass scale.

Joystick	Keyboard
Analog Stick: Navigate	Arrows control camera direction
Joystick 1:Mount camera to prey Joystick 2:Decrease adherence to flocking rules Joystick 3:Increase adherence to flocking rules Joystick 4: Follow clef Joystick 5: Delete one note Joystick 6: Add one note Joystick 7: Zoom out Joystick 8: Zoom in Joystick 10 / Start – Restart	[ESC] Ends program [A] / [Z] Zoom in/out [+] / [-] Increase/Decrease Rule Behavior [S] / [X]: Decrease, increase lighting [W] Wire frame toggle

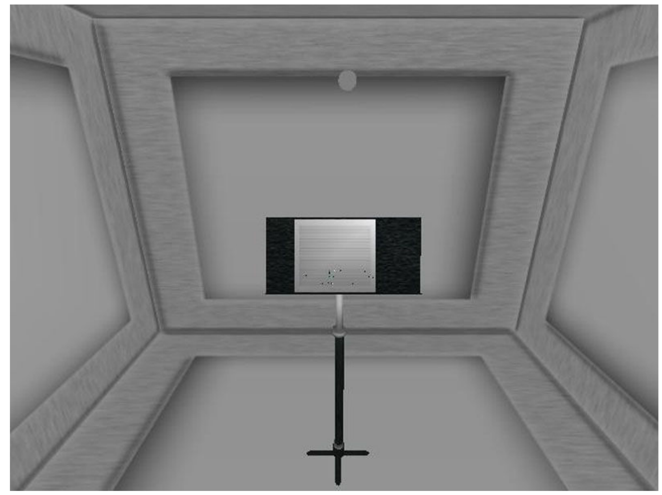

Figure 3. Music Box performance start state

As in figure 3, when the system is started, the user is presented with a 3D render of a traditional music stand and musical sheet music. Users may choose to navigate the space to emphasize specific sounds or change the character of the music created. As shown in figure 4, users may freely navigate, lock their camera on a specific element, or allow the camera to ride an element as it travels.

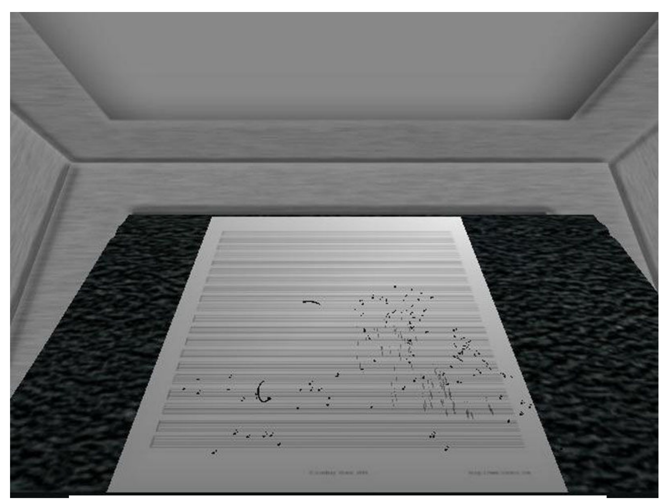

Figure 4. Music Box performance

The simulation continues, resounding with musical tones indefinitely, shown in figure 5. Users may complicate the performance by removing all prey during run time or otherwise changing the balance and distribution of musical elements.

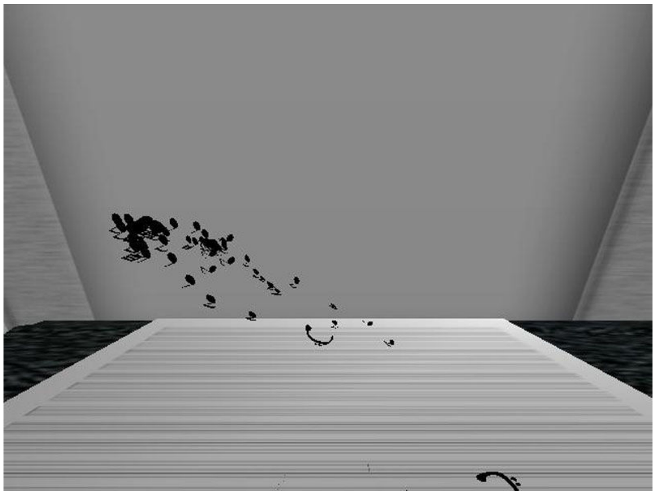

Figure 5. Music Box performance

This work has been demonstrated at several venues to audiences who have responded positively to the novelty of the system. Most recently it was displayed at the creative exhibition of the Fifth International Conference on Advances in Computer Entertainment Technology in Athens, Greece. It was also on display at the University of Illinois-Chicago. It is typically displayed with 100 predators, and 5 prey, but as previously stated, the systems is designed for reconfiguration.

IV. Conclusion and Future Work

This project seeks to offer a model for the application of an emergent behavior to the production of musical tones. This is an exploration in the potential of video game technology to produce distinct musical qualities. As with most aesthetic investigations, the evaluation of this algorithm is subjective. As such, it is the investigators objective to simply offer this algorithm as a point of departure for future work. This research represents an algorithm and demonstration that further a more than century old curiosity about the relationship of visual stimuli and corresponding aural sensation.

Acknowledgment

The researcher would like to thank the University of Illinois Chicago's Electronic Visualization laboratory, who encouraged this exploration. The Miami University Armstrong Institute for Interactive Media Studies deserves special acknowledgement for its support and appreciation of interdisciplinary practice in new media technologies.

Reference

[1] Reynolds, C. W. 1999. Steering behaviors for autonomous characters, in Proceedings of the Game Developer's Conference, pp. 763–782

[2] Freeman. J. "Collaborative creation, live performance, and flock." Leonardo Music Journal Vol. 18, 2008.

[3] Levin, G.: Painterly Interfaces for Audiovisual Performance, Master Thesis, Massachusetts Institute of Technology, 2000

[4] Lew, M. Live cinema: designing an instrument for cinema editing as a live performance, Proceedings of the 2004 conference on New interfaces for musical expression, p.144-149, June 03-05, 2004, Hamamatsu, Shizuoka, Japan

[5] Lucier, A. I am sitting in a room. Lovely Music, Ltd., 1990. CD

[6] A. Dulic and K. Hame. "Visual music instrument." International Journal of Arts and Technology 2009 - Vol. 2, No.1/2 pp. 22 - 39

For more information please visit:
http://musicbox.mindtoggle.com