AIIS: An Intelligent Improvisational System

ABSTRACT

The modern use of electronic sound in live performance (whether by instrument or composed processing) has continued to give rise to new explorations in its implementation. With the construction of Aiis, we sought to build an interactive performance system which allows for musical improvisation between a live performer and a computer generated sound world based on feedback between these components. The system's micro and macro decisions generate a programmable musical "personality" derived from probabilistic measures in reaction to audio input. The system's flexibility allows factors to be modified in order to make wholly new and original musical personalities.

1. INTRODUCTION

1.1 Musical Influences

The genre of electroacoustic music allows for the exploration and augmentation of sonic worlds which transcend the limitations of acoustic instruments. A key component of the genre is the ability to create musical structures which draw fundamental features from digital processes such as algorithmic composition, mathematical modeling, and interactive systems [1, 2].

Contemporary performers are also making use of electronic sounds in genres from Free Improvisation to Noise. From playing through effects in order to accentuate or drastically modify their sonic palette, to using electronic instruments in improvisation, this marriage of instrumentalist with electronic music source offers countless approaches. The results of these collaborations, to name only a few, can be seen in the music of Peter Evans Quintet, Evan Parker's work with the Electroacoustic Ensemble, or the recent pairing of Merzbow, Mats Gustafsson, Thurston Moore and Balzs Pandi.

1.2 Intelligent Systems

Against this backdrop, our initial research centered around a few guiding questions. Firstly, can the ultimate hallmarks of humanity, those being creativity and free decision making, be digitally replicated convincingly enough to power collaborations with a live performer? Secondly, through analysis, can a specific performer's musical personality, otherwise thought of as their reaction to stimuli, be conceived as a collection of probabilities and thereby replicated?

In the 1980's, Rodney Brooks put forth a new foundation for artificial intelligence known as subsumption architecture. The foundation was a representation of the then revolutionary, reactive paradigm. Until then, necessary and sufficient properties of machine intelligence were thought to consist of three crucial components: sensing, planning, and acting. It was during the planning stage that a machine would analyze sensor data to generate symbolic representations of objects in the world in order to react accordingly [3]. The reactive paradigm eliminates this intermediate phase and simplifies the acting stage. In lieu of the planning stage, the reactive paradigm implements substantially more sensing and reacting behaviors which work together in a non-hierarchical fashion to generate intelligent behavior.

Aiis is the first instantiation of an interactive system which attempts to address our questions by implementing the reactive paradigm architecture. It seeks to provide a continuously creative system which is able to interact with a performer by responding to stimuli and creating its own content. Included in its programming are features which react to stimuli in an improvisatory and musical fashion.

2. BIG PICTURE

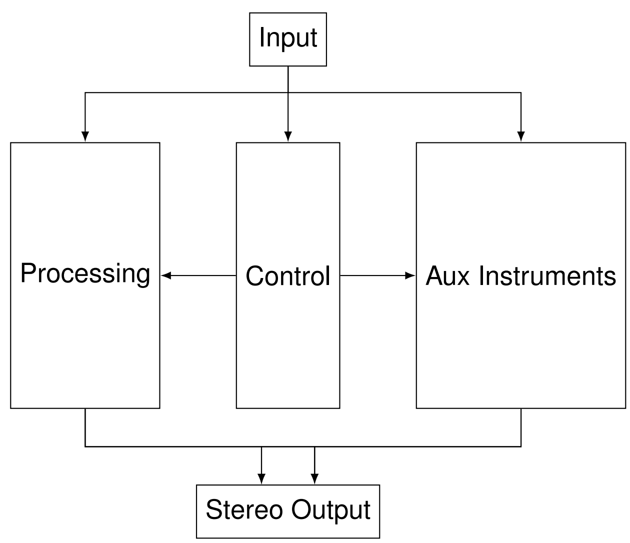


Figure 1. Over arching flow of the machine. Input from a performer is sent to the processing, control, and auxillary instrument modules to be manipulated. The stereo output is a sum of the outputs from processing module and the auxillary instruments

Inspired by Merzbow, Don Dietrich, and Philip White, *Aiis* generates a type of spectral painting through signal processing and analyzes live audio as input data to control parameters.

Audio is routed to three separate internal modules - *Processing*, *Control*, and *Auxiliary Instruments* - each of which contains sets of rules which manipulate or analyze the present waveform. The *Processing* module sets the foundations of the sonic environment by manipulating input signal. This is augmented by the *Auxiliary* module which houses two additional processes that provide further musical underpinnings. Meanwhile, signal is analyzed in the *Control* module which develops decision making tools utilized in shaping the sonic environment of the *Processing* module as well as large scale musical form. In this sense, *Control* is the paintbrush of our spectral painting deciding when, where, and how the palette of colors will be managed to produce musical results.

2.1 Processing

In order to create a sound world which is larger and more diverse than the range of an acoustic instrument, live audio is routed to four separate channels within the *Processing* module, each of which manages a specific frequency range. These channels extend the range to values forty half steps below the lowest naturally occurring note and twenty half steps above the highest for a total of almost eight octaves possible on a two and half octave instrument. Channel three operates in the same range as the performer while channels one through four operate from the lowest to highest frequency ranges respectively.

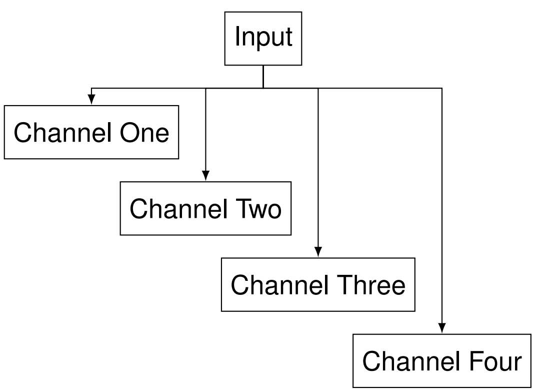


Figure 2. Abstracted view of the processing subspace. Input from the performer is sent to four distinct channels to be processed for the sonic environment of the piece.

Though independent of each other, the channels all contain an identical signal flow beginning with a pitch shifter and terminating with a stereo panner as illustrated in Figure 3. By processing the pitch shifted signal, the channels gain an extremely versatile potential for generating frequency range specific audio despite their identical chains of effects. Each effect also contains anywhere from one to four parameters which can be set independently by *Control* thus further expanding the channel's versatility.

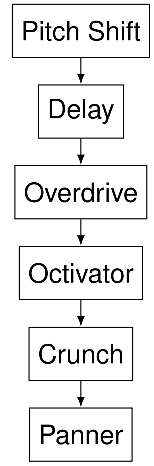


Figure 3. Chain of effects through which the performer's audio is processed.

2.2 Aux Instruments

After testing and listening to early versions of this system, it seemed the overall texture (even with the variety of effects made available via the *Processing* module) still needed a boost of saturation in order to begin approximating our Noise music muses. The purpose of the *Auxiliary Instruments* module is to create this soft bedding on top of which the rest of the sonic environment of the piece develops. This is achieved with the addition of two instruments: *Sub-sound* and *Glisser*. The former is overdriven white noise modulated through a combination of delay line, low-pass filter, and hi-pass filter. The latter, embedded within each sub-channel of the *Processing* module, is tapped after channel pitch shifting and performs granular synthesis on a section of captured signal. It will then glissando to a subsequent capture via waveform and pitch to create a soft, flowing, tonal contrast to the saturated colors of its partner.

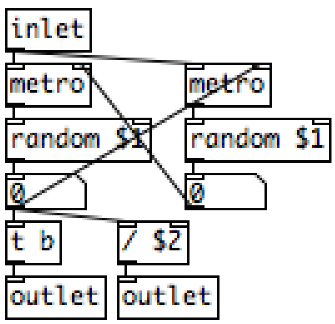
2.3 Control

The *Control* module establishes the mechanisms for control of a number of parameters within the system. It accomplishes this goal on three distinct levels. In its most basic function, referred to below as Non Reactive controls (NR), it creates a perpetually original audio output by randomly altering effect parameters in the channels of the *Processing* module. On the Small Scale Reactive level (SSR), analyses of the performer's audio overtly influence a smaller more select group of factors within the *Processing* module's effect parameters. Finally, based on analysis of audio, Control effects another set of parameters we will refer to as Large Scale Reactive (LSR) which are intended to aid in the creation of macro structures, musical form, and time.

2.3.1 Non Reactive Controls

The lowest level of controls operates on the smallest objects: the effect parameters. At this level, there is no reaction from the system to the performer, rather, attributes of the nonrepetitive nature of the output are constantly generated. This is where the sound world gains its richness and depth.

We quickly found that while random processes can easily accomplish this goal, implementing a simple random operator would only succeed in altering the parameters at a periodic rate. For example, an effect can be turned on and off randomly, but would make that decision every 5 seconds. The solution at which we arrived, named ”double random”, was a simple tool which generates random numbers at random intervals of time less than a given parameter. This tool, with



;

Figure 4. *Double Random* became an important to tool to randomly affect parameters at random time intervals. The time intervals were constrained by an upper bound specified by the user. Typically, we used 8 seconds.

an upper boundary of eight seconds, ended up being used exclusively in the control of every effect parameter found in the *Processing* module. As a result, effect parameters are manipulated randomly within a specific numerical range and this operation takes place at random time intervals between zero and eight seconds.

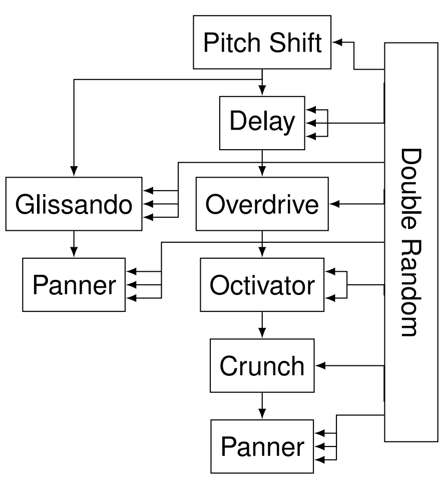


Figure 5. Non reactive control flow on a micro level. The effect parameters, indicated by the arrows on the right side of each, are controlled randomly through the ”double random” object in Figure 4

2.3.2 *Small Scale Reactive Controls*

The next level up from NR controls are the SSR controls. SSR analyzes amplitude and note rate from audio input in order to effect parameters in the *Processing* module. For example, when a performer increases in volume the system senses this and may instigate a ”freeze” on the channel’s delay effect (essentially turning off the NR controls managing parameters). SSR controls output parameters as indicated in Figure 6

Perhaps due to their reactive nature, but certainly as departures from the normative functioning of the piece, these controls are more pronounced within the generated sonic environment of the machine. As a result performers are more aware of these developments in what can be imagined as the middle ground of musical thought and will react accordingly. In this way we can think of the SSR controls as generating musical phrases. Though important in construction of the musical world moment to moment, these controls are not as crucial to the macro level musical structures managed by the LSR controls.

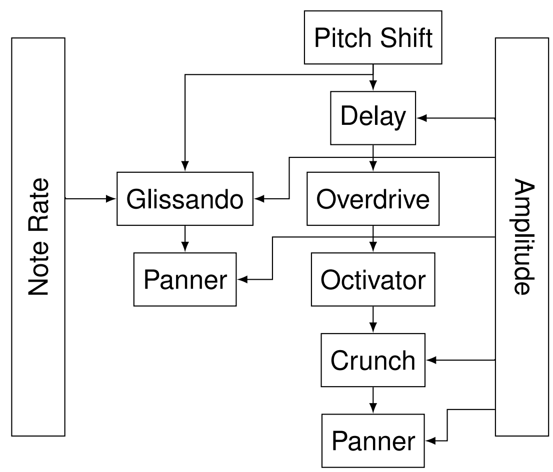


Figure 6. Small scale reactive controls. Certain effect parameters are controlled by the amplitude of the performer, while some are controlled by the note rate.

2.3.3 *Large Scale Reactive Cotrols*

LSR controls were implemented to enable the system to generate musical structures in collaboration with a performer. There are a few behaviors that an improvisor or performer exhibit which we felt were necessary to replicate on the macro level. First, we enable the system to stay or ”freeze” in a given state for a elongated period of time. Secondly, since a live performer does not always react instantaneously to their partner, we implemented a variable reaction time to input. Lastly, we created a sensitivity to musical structure vis a vis saturation and complexity over time. These controls offer a variety of long term, clearly audible choices concerning the structure of a performance.

The system’s analysis of the rate of notes controls the turning on and off of the NR control of all channel delays and pitch shifts essentially ”freezing” the output in a given state.

Unlike the SSR controls, the large scale note rate reaction will freeze or re-instantiate the delay for every channel simultaneously. The same process is incorporated in regards to pitch shifting; every channel yields to this control. Note rate also controls the "reaction" parameter. This parameter was created to introduce a human like reaction time to input from the performer. This parameter uses a bell shaped, gaussian distribution (with a variable mean and pre-programmed variance) to represent an expected reaction time and the intuitive experimentation of an individual which may deviate from this norm. The variable mean is controlled by the note rate of the performer which allows the computer to react, in general, more rapidly if the performer is playing faster and more slowly if the performer is playing longer notes.

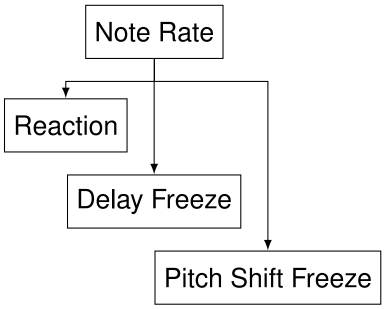


Figure 7. Large scale reactive control. The performers note rate will affect the reaction time of the computer as well as whether or not to freeze the pitch shift and delay for all channels.

At its core, the system is intended to react to the performer and generate sounds that would contribute to the musical world, which would then be interpreted and reacted to by the performer. This generates an improvisational feedback circuit. If the circuit breaks, if one or the other entities refuses to listen and react, we may lose a sense of musical structure.

Amplitude controls the most perceptible macro events generated by the machine: entire channels of the *Processing* module turning on and off. This parameter contains four separate levels within itself, each one dependent on input amplitude. If the performer is at his or her softest level, the machine will only turn one channel on at a time. Likewise, at the second softest volume, the machine will turn either one or two channels on at a time. The same idea applies to the third level, but at the fourth level (the performers absolute loudest level), all channels will be opened.

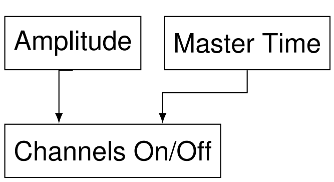


Figure 8. Large scale reactive control from performer's amplitude and *Master Time*. The amplitude will decide how many channels to turn on at a given time, while the *Master Time* decides when to do this.

The final addition to this piece was made after months of trials. In performing with the system, the lack of musical pulse became apparent. This inhibited the ability to develop structure within the piece and provided an uncanny valley for musical action. In almost every musical piece, except those which feature an abstraction of time, a pulse can be distinguished and followed by both the performers and the audience. Whereas this pulse is commonly represented by bars and measures, in free improvisation it often takes the form of long term macro beats or "breaths".

"Master Time" was implemented to portray these large scale musical pulses within the overarching structure of the piece. While the amplitude of the performer influences how many *Processing* channels are present, "Master Time" controls when the channels do so. The result of this is heard in the system's ability to maintain large scale musical sections punctuated by macro periodicity despite the random nature of the aural world developed by the NR and SSR controls. After experimenting with various time intervals, the current version sets "Master Time" was set to five seconds.

3. PERSONALITY

Up to this point, we have discussed each level of control and how they react to the performer, but another step was implemented to make controls contribute to the "personality" of the system. The term personality (or predilection), in this sense, means the ability to make decisions with respect to the sonic environment or large scale musical structures and the probability with which those decisions are made. If this personality were absent, each of the effect parameters and controls described above would mirror a one-to-one mapping (e.g., if the performer is loud, the delays will freeze). When considering replication of a given personality, if we consider a performer's predilection to be the probability of their reaction to a certain stimulus, then we must also implement the opposite reaction as a type of musical "creativity".

Effect	Off	50% Mix	100% Mix
Delay	10	45	45
Overdrive	10	45	45
Octavator	72	2	16
Panner	50	0	50

Table 1. The percent chance for the effect to turn on at 50% mix or 100% mix, or for the effect to turn off. These probability measures account for the machine's creativity. Every effect parameter is decided on by probabilities.

Probability measures were therefore introduced in order to account for this notion of predilection vs. experimentation. Table 1 illustrates examples of how the percentage of an effect in the audio mix is controlled by a probability as well as the "double random" object. Each time "double random" outputs a bang, the machine filters that bang as to whether it will turn off the effect or turn on the effect at 50% or 100% mix with unique probabilities. From the table, it is observed that the octivater has a very low chance of turning on while the delay and overdrive have a much higher chance of doing so. This idea of implementing creativity as a combination of probabilistic measures is also applied to SSR and LSR controls.

```
if (amplitude > baseline_amplitdue + 15)
      output 1 with probability .2;
else if (amplitude < baseline_amplitude - 5)
      output 0 with probability .03;
```

Figure 9. Psuedocode example to assign probabilities to small and large scale reactive controls.

4. RESULTS

The reactive and non-reactive nature of the controls, coupled with the probabilistic personality measures, creates a improvisational tool which enables musical experimentation and ever evolving structure. The different levels of control allow for the production and maintenance of the sonic environment, as well as larger structural gestures. If a performer is not constantly aware of the aural world generated by the machine, the resulting music lacks congruence in its textural qualities. If the performer refuses to acknowledge the machine?s larger scale structural changes, then the musical form will become similarly unconvincing.

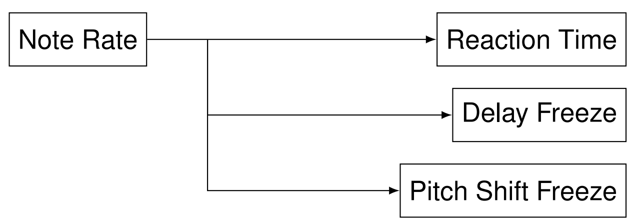


Figure 10. Reactive paradigm for the note rate sensors. The machine reacts to the note rate sensors in each of the three behaviors on the right. Each behavior is independent of the other.

The implementation of subsumption architecture and the reactive paradigm largely contributes to the system's success in generating musical form. On the left side of Figure 10, the sensing mechanism is the performer's note rate while the right side illustrates each of the possible reactions. These behaviors maintain the machine/performer circuitry that is crucial to generating music. Because each is independent of each other, there is no hierarchy of decisions, our system is made free to interpret and react to the performer.

Similarly, Figure 11 illustrates the reactive paradigm as it pertains to the amplitude sensors of the system. Together, the behaviors exhibited in each of these figures underlie the musical decisions conveyed by our system.

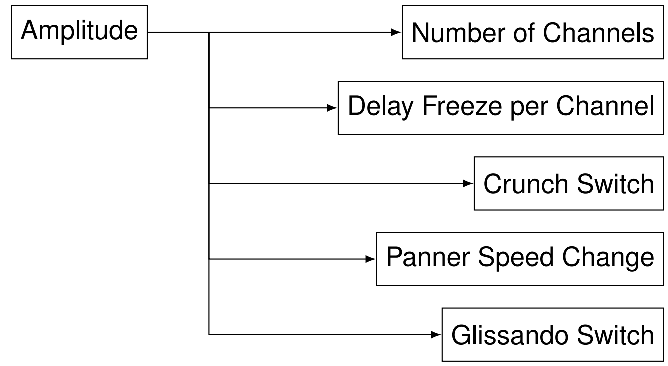


Figure 11. Reactive paradigm for the amplitude sensors. The machine reacts to the amplitude sensors in each of the five behaviors on the right. Each behavior is independent of the other.

5. FUTURE

In the future, we plan on extending the capabilities of the machine to be able to control graphics as well as analyze the motion of a performer. A piece is currently under development that will allow a dancer, musician, and the machine to improvise and interact with each other through the use of movement, sound and graphics. We are also in the process of building more machines with different ”personalities” based on sets of processing rules and controls. Though still using the reactive paradigm we hope to add and manipulate the processing subspace and the control subspace in order to broaden the versatility of the machine. Lastly, it is foreseeable that we will be able to create machines that can mimic specific performer's probability measures through the analysis of recordings and compositions.

6. REFERENCES

[1] G. Lewis, “Too Many Notes: Computers, Complexity and Culture in Voyager,” *Leonardo Music Journal*, vol. 10, pp. 33–39, 2000.

[2] R. Rowe, *Interactive Music Systems: Machine Listening and Composing*. MIT Press, 1993.

[3] R. Murphy, *Introduction to AI Robotics*. MIT Press, 2000.