Design Issues in Interaction Modeling for Free Improvisation

William Hsu
Department of Computer Science
San Francisco State University
1600 Holloway Avenue
San Francisco CA 94132 USA
hsu@tlaloc.sfsu.edu

ABSTRACT

In previous publications (see for example [2] and [3]), we described an interactive music system, designed to improvise with saxophonist John Butcher; our system analyzes timbral and gestural features in real-time, and uses this information to guide response generation. This paper overviews our recent work with the system's *interaction management component* (IMC). We explore several options for characterizing improvisation at a higher level, and managing decisions for interactive performance in a rich timbral environment. We developed a simple, efficient framework using a small number of features suggested by recent work in mood modeling in music. We describe and evaluate the first version of the IMC, which was used in performance at the Live Algorithms for Music (LAM) conference in December 2006. We touch on developments on the system since LAM, and discuss future plans to address perceived shortcomings in responsiveness, and the ability of the system to make long-term adaptations.

Keywords

Interactive music systems, timbral analysis, free improvisation.

1. INTRODUCTION

Most improvisation systems today work mostly with MIDI data; see, for example, [1] and the references in [2]. We have been developing a system, in collaboration with saxophonist John Butcher, that tracks both timbral and gestural information in performance. [2] focused on timbral feature extraction; we described how the system worked with characteristics such as noisiness, harmonicity of partials, amplitude envelope flutter, presence of multiphonics, and sharp attacks. An ensemble of improvising agents accesses a variety of performance information in its interactions with Butcher. Each agent "plays" a virtual instrument capable of a wide range of timbral and gestural variation, such as a waveguide bass clarinet and filtered noise, and responds to specific timbral or gestural features. A human operator has the option of shaping some aspects of the high-level behavior of the ensemble, in a manner similar to Butch Morris' conductions. [3] described tracking timbral contours over musical gestures, and referencing these contours for response generation. Hence, the gestural repertoire of the system is inherently able to adjust to evolution in Butcher's performance. The system was not designed to be fully autonomous, but it is able to make simple high-level decisions in performance without human interference. This version of the system was used in duets with Butcher at ZKM (Karlsruhe) in May 2006; check http://userwww.sfsu.edu/~whsu/Timbre for audio clips.

Our recent work with the system has been on improving the *interaction management component*, which coordinates the higher level behavior of the improvising agents. We would like to have the option of increasing the autonomy of the system, while maintaining reasonable musicality. Our goal was to develop a simple framework for performance behavior, that incorporated a few perceptually significant gestural/timbral features. We would use this framework to organize our low-level measurements and features, and form high-level characterizations of Butcher's (or another human improviser's) performance. The same framework would be applied to coordinate the behavior of our improvising agents.

The framework should be efficient to compute and use a relatively small amount of storage, since our entire interactive music system has to run on a machine with average computing resources by today's standards. Using the framework, the system should adjust relatively quickly to a human improviser in performance, without the need for an extended pre-performance training phase. The system should be responsive to short-term events, but able to adapt its higher-level behavior to longer term changes in the human improviser's performance.

We will first overview related work on improvisation modeling and mood modeling. Using some ideas from music content classification, we develop a simple framework for describing a solo performance. Then we will describe a first version of the interaction management component, which was used with the system in two short performances at the Live Algorithms for Music (LAM) conference in December 2006 in London. We will evaluate the results of the LAM performances, and describe briefly work on the system since then, including future plans to address perceived shortcomings of the earlier system.

2. RELATED WORK

Most previous work in interaction modeling for jazz and improvisation has focused on analysis of MIDI note events. George Lewis' Voyager [1] extracts a variety of statistics from input from a pitch-to-MIDI converter, to coordinate the generation of complex responses to a human improviser and independent behavior from its internal processes.

Walker [4] used results from conversation analysis to construct a system for small group jazz improvisation. As is appropriate for

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Nime'07, June 7-9, 2007, New York, NY, USA.
Copyright remains with the author(s).

this context, the interaction was largely based on well-defined roles (solo vs. comping, trading fours, etc). Only MIDI information was used.

Dannenberg et al. [5] built a system that improvised with a trumpet player; neural networks were used in a training phase to recognize styles such as "frantic", "lyrical" and "syncopated". A pitch-to-MIDI converter captured data from the trumpet for use by the system. Thom [6] also used machine learning techniques to model melodic generation in contexts such as jazz.

Roberto Morales' GRI [7] combines pitch with information from sensors that capture a flautist's physical gestures. GRI uses neural-network based learning to form a characterization of an improviser's behavior during a learning phase; during performance, it tries to predict the improviser's future behavior.

In [8], Collins describes an improvisation simulation for human guitarist and four artificial performers. The emphasis is on event onset, pitch and other gestural information, with the artificial performers listening closely, responding to and possibly referencing the gestures from other performers. Some parameters such as "shyness", "sloppiness" and "keenness", which are not commonly described in other systems, are used to characterize the high-level behavior of the improvising agents. While these parameters directly describe how improvisers might listen and react in a performance, it is very difficult to make such estimates from a human's performance.

While there has been little work in improvisation modeling/simulation that addresses the role of timbre, timbral analysis is essential in recent research in music content classification. For example, [9] uses timbral features to detect and track mood in traditional classical music. Since we apply some of their ideas to our framework for free improvisation, we will discuss them in greater detail in Section 3.

3. PERFORMANCE MODE DESCRIPTORS

In [9], Lu et al. described the design of a system for tracking the mood or emotive content of a piece of music. Thayer's two-dimensional model of mood [10] was used to classify the mood of a music clip into one of four classes. Three primary sets of (over 15) features were found to be useful for mood detection: intensity (essentially amplitude/loudness), timbre, and rhythm. A Gaussian mixture model is then used to map regions of the feature space into one of the four mood classes.

While free improvisers often avoid references to traditional notions of mood and affect, we felt that Lu's three feature sets provided a succinct approach for characterizing some of the perceptually important aspects of an improvised performance. We decided to adapt their approach for the interaction management component in our system.

There are relatively tight constraints on our system in terms of resources and real-time performance. Hence, for our simple framework, we chose one representative feature from each of Lu's three feature sets. Lu's intensity feature set comprises various amplitude-based measurements; our choice is an overall loudness estimate.

The timbre feature set included *brightness*, *spectral flux*, *sub-band contrast* etc.; we settled on an auditory roughness estimate based on [11], not in Lu's original feature set, for our representative in this group. Roughness is estimated by extracting partials from the audio; for each pair of partials, its contribution to the roughness measure is computed. Finally, the roughness contributions of all the pairs are summed to form the overall roughness estimate. We have worked with roughness extensively in our system (see [3]), and it has proved to be a useful feature in timbrally rich saxophone tones. Moreover, [12] indicates that roughness variation seems to be correlated with tension-release patterns in music.

The third feature set of rhythm comprises *rhythm strength, rhythm regularity,* and *tempo*. Rhythm strength and regularity are much less prominent features for free improvisation than for traditional classical music; the obvious candidate from this set appears to be tempo (essentially event density).

While our system already provides usable estimates for loudness and roughness, the measurement of rhythmic features for saxophone performance turned out to be more challenging than expected. As is well-known, it is in general difficult to determine note onsets in legato saxophone lines with pitch inflections. This is further complicated by the use of extended techniques to produce complex timbres. (For a good overview of the problems of and approaches to onset detection, see [8].) We attempt to segment saxophone gestures into approximate note events by combining amplitude-based measurements and information from a pitch detector. We look for regions of stable pitch, rated by the pitch detector with a high quality factor, and mark transitions between them as note transitions. While this works reasonably well in many situations, it still fails with some types of material. For example, consider a clip about 30 seconds into Track 4 of Evan Parker's Conic Sections solo CD; the pitch inflections are too fast and extreme. (One might argue that note onsets are simply not well-defined in such material, and even a human would do no better at identifying note transitions.) We also attempted a measure of rhythmic regularity, but were again hindered by onset detection difficulties.

With our three selected feature representatives, we have a feature vector comprising intensity (soft/loud), timbre (smooth/rough), and tempo (slow/fast). In addition, we observed that the roughness estimate requires rather large windows (> 100 ms) to compute; a fast run of notes will often appear to be "rough", because of note transitions and instability within the analysis window. Hence, we refine our high-level description such that only clips classified as slow can have a valid roughness descriptor; for fast clips, roughness is always undefined. Using our simple framework, we classify a performance clip into one of seven modes: silence, slow/soft/smooth, slow/soft/rough, slow/loud/smooth, slow/loud/rough, fast/soft, fast/loud. It is also straightforward to use these descriptors to guide the behavior of improvising agents.

4. INTERACTION MANAGEMENT COMPONENT (IMC)

4.1 Component Design

Figure 1 is a simplified block diagram of our system, showing how the new interaction management component (IMC) communicates with the timbral measurement components and the improvising agents. A window of audio data (from the saxophonist or other human improviser) is analyzed to extract intensity, roughness and tempo estimates. One of the pre-defined seven performance modes is determined for each event window.

Each improvising agent consults the current performance mode description when deciding on future actions. The simple decision logic depends on parameters such as: the likelihood of a response, average response duration, average idle period before next consultation etc. These parameters are similar to some in the improvisation simulation of [8], and can be changed during a performance to adjust agent behavior. One of the seven performance modes is chosen to guide the response; for example, an agent may choose to *support* (respond in the same mode as the human), or *oppose* (respond in a mode with contrasting parameters).

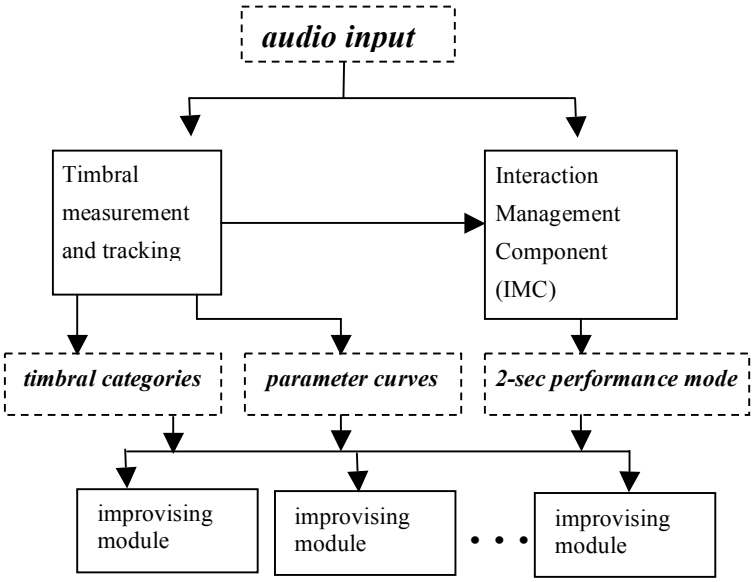

Figure 1: High-level system organization

In [9], Lu's system looks for audio segments of at least 16 seconds of "constant mood", in compositions primarily from the classical and romantic periods. Performance modes in free improvisation tend to show much greater flux. Measurements of selected tracks from John Butcher's solo recordings show that there are relatively few segments that maintain the same performance mode for more than eight seconds; many segments of constant mode are under four seconds. (We should point out that different improvisers obviously have different gestural preferences, and measurements for solo performances will almost certainly differ for small-group improvisations.) For our initial design,, we report the performance mode over a 2-second window.

Figure 2 shows in more detail the operation of the IMC. The audio input stream is divided into 20 ms windows. An average amplitude, pitch estimate and pitch estimate quality is computed for each window, and sent to the IMC. In addition, the IMC receives from the timbral analysis components the 2-second average of the roughness estimate of the input audio. As described in Section 3, note onsets are estimated using a combination of amplitude and pitch estimates and transitions; the number of discrete notes identified over the last two seconds determines whether the human improviser is playing in a fast or slow mode. The amplitude measurement is also used to characterize the performance as loud or soft, or silence; the roughness measurement is used in the rough/smooth characterization.

Each improvising agent has an internal set of characteristics that guides its behavior. These characteristics include its likeliness to perform, average duration of a response, the waiting time between responses, and whether it prefers to respond in a supporting or contrasting mode. During a performance, each agent monitors the current performance mode of the human improviser. Its instantaneous behavior is guided by a combination of its internal characteristics and the performance mode of the human.

For the time being, we have decided not to explore machine-learning techniques. While such techniques are highly attractive for well-known reasons, a training phase is often required before proper operation. In many improvisation contexts, one does not have the luxury of an extended rehearsal before the actual performance. Also, many free improvisers play in a huge variety of modalities rather than relatively well-defined styles; we are dubious that a limited training period would be sufficient to effectively capture a useful proportion of an improviser's behavior. A detailed evaluation of the effectiveness of machine-learning techniques for characterizing free improvisation might be an interesting future project.

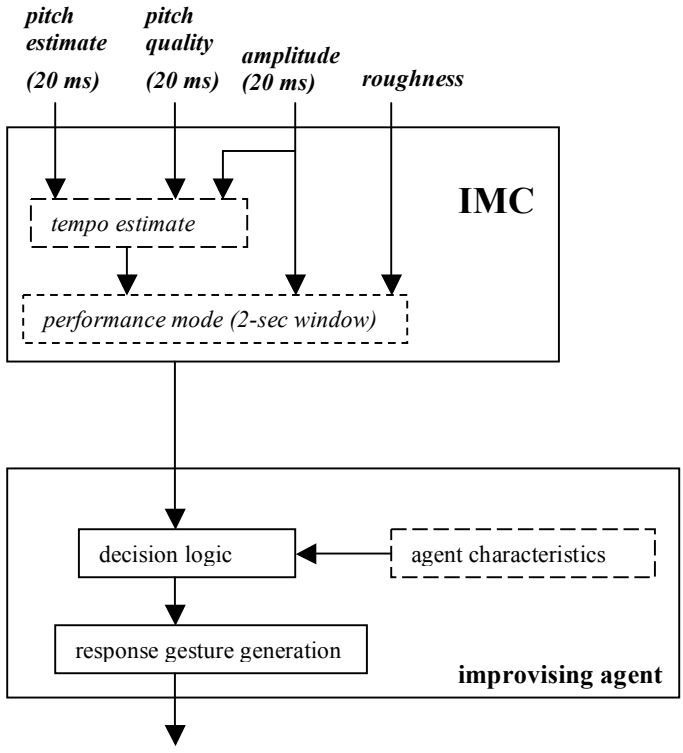

Figure 2: Interaction Modeling Component Operations

4.2 Evaluation at LAM 2006

Our system with the initial version of the IMC made its first public appearances at the Live Algorithms for Music conference (see http://homepage.mac.com/oobop/lam/events.htm) in

December 2006. It performed in a short segment with Evan Parker, and a longer segment with John Butcher. (We are working to edit and upload excerpts of the performances; for details please check http://userwww.sfsu.edu/~whsu/Timbre.)

In the improvisation with Evan Parker, as anticipated in Section 3, the IMC had difficulties analyzing accurately some of his saxophone gestures, which often had very fast runs of legato notes with extreme pitch inflections. However, there was usually enough ebb and flow in Parker's playing for the IMC to work with. Since John Butcher's performance worked more with timbral variation with more stable pitch information, it was easier for the system to get good estimates of his gestural and timbral characteristics. In both performances, reasonable results were obtained for longer periods with much less human intervention, when compared with earlier versions of the system,

While the system did produce some musically interesting results, we felt that there were also segments where a "soft gray" quality of interaction dominated. There were interesting sonic events and combinations, but the sense of connection between events could be stronger. After moving to an interaction framework based primarily on high-level performance event windows, the system seemed to lose some of the responsiveness it used to have, when mostly low-level measurements and features were referenced. Also, we felt that we could improve the adaptability of the system to long-term changes in the human improviser's behavior, and possibly reduce further intervention by a human operator.

5. RECENT DEVELOPMENTS AND FUTURE DIRECTIONS

Based on the observations from Section 4, we embarked on a significant redesign of the IMC. One major goal was to improve the responsiveness of the system in terms of timing of response generation and connection between sonic events; our approach was to integrate better the use of low-level measurements, and identify sets of potential *trigger events* that affect fine-grain timing of gestures. Another goal was to enhance the ability of the system to make long-term adaptations in the choice of performance mode and response materials; we built an *adaptive performance map* for matching the human's performance characteristics with the system's response behavior, and used a simple mechanism for inferring "desirable" matches. Limited space prevents us from describing this work here; a future publication will detail these developments.

Since LAM 2006, we have not yet been able to test the new IMC in a live situation with John Butcher or another saxophonist. We have however run tests using recordings; some clips can be found at http://userwww.sfsu.edu/~whsu/Timbre. The saxophone solo recording is in the right channel, with a single agent "playing" filtered noise in the left channel. The generated gestures are fairly simple for demo purposes and have not gone through the finetuning that is usual before a performance. Also, there is no "feedback" from the agent performance into the saxophone performance, so a major part of the interactive experience is missing.

Our work with the interaction management component has greatly increased the autonomy of our system. The system is now better able to coordinate both high and low level information, and seems capable of some musically interesting behavior with less human intervention. Future directions include improving note-segmentation for tempo estimation, and exploring machine learning techniques for interaction management.

6. REFERENCES

[1] Lewis, G., Too Many Notes: Computers, Complexity and Culture in *Voyager*. In *Leonardo Music Journal*, Vol. 10, 2000.

[2] Hsu, W., Using timbre in a computer-based improvisation system. In *Proceedings of the ICMC* (Barcelona, Spain, Sept. 5-9, 2005).

[3] Hsu, W., Managing Gesture and Timbre for Analysis and Instrument Control in an Interactive Environment. In *Proceedings of New Interfaces for Musical Expression* (Paris, France, June 4-8, 2006).

[4] Walker W., A Computer Participant in Musical Improvisation. In *Proceedings of 1997 Human Factors in Computing Systems (CHI '97)*, 123-130.

[5] Dannenberg, R., Thom, B., Watson, D., A Machine Learning Approach to Musical Style Recognition. In *Proceedings of the ICMC* (Thessaloniki, Greece, 1997).

[6] Thom, B., Machine Learning Techniques for Real-time Improvisational Solo Trading. In *Proceedings of the ICMC* (Havana, Cuba, 2001).

[7] Morales R. et al., Combining audio and gestures for a real-time improviser. In *Proceedings of the ICMC* (Barcelona, Spain, Sept. 5-9, 2005).

[8] Collins, N., *Towards Autonomous Agents for Live Computer Music: Realtime Machine Listening and Interactive Music Systems.* Ph.D. Thesis, University of Cambridge, Cambridge, UK, 2006.

[9] Lu, L. et al., Automatic Mood Detection and Tracking of Music Audio Signals. *IEEE Transactions on Audio, Speech, and Language Processing,* Vol. 14, No. 1 (Jan. 2006), 5-18.

[10] Thayer, R., *The Biopsychology of Mood and Arousal.* Oxford University Press, Oxford, U.K., 1989.

[11] Vassilakis, P., Auditory roughness estimation of complex spectra – roughness degrees and dissonance ratings of harmonic intervals revisited. *Journal of Acoustical Society of America, 110(5/2).*

[12] Vassilakis, P., An improvisation on the Middle-Eastern mijwiz: auditory roughness profiles and tension/release patterns. *Journal of Acoustical Society of America 117(4/2).*