Anticipation in collaborative music performance using fuzzy systems: a case study (Extended abstract)

Oscar Thörn[1], Peter Fögel[2], Peter Knudsen[2], Luis de Miranda[3], and Alessandro Saffiotti[1]

[1] AASS Cognitive Robotic Systems Laboratory
School of Science and Technology, "Orebro University, Sweden
oscar.thorn@oru.edu, alessandro.saffiotti@oru.edu
[2] Music Höskolan, Örebro University, Sweden
peter.fogel@oru.edu, peter.knudsen@oru.edu
[3] School of Humanities, Education and Social Sciences, Örebro University, Sweden
luis.de-miranda@oru.se

1 Introduction

The creation and performance of music has inspired AI researchers since the very early times of artificial intelligence [5], and there is today a rich literature of computational approaches to music [7], including AI systems for music composition and improvisation. As pointed out by Thom [10], however, these systems rarely focus on the spontanous interaction between the human and the artificial musicians. We claim that such interaction demands a combination of reactivity and anticipation, that is, the ability to act now based on a predictive model of the companion player [8]. This paper reports our initial steps in the generation of collaborative human-machine music performance, as a special case of the more general problem of anticipation and creative processes in mixed human-robot, or *anthrobotic* systems [2].

2 Methodology

We focus on the collaborative execution between a human musician and a robotic performer. We assume that the latter is capable of autonomous artistic execution, whose modalities are controlled by a fixed set of parameters. We addresse the problem of controlling these parameters to obtain a harmounious performance.

Figure 1 illustrates the concept used in this paper. A human musician plays freely, and an AI system controls the parameters of a robotic performer. We use "robotic" in a broad sense to mean any agent that generates physical actions: this could be a dancing robot, a virtual drummer, or a sound processing agent that spatializes in the hall the music produced by the human. In the case addressed in this paper, the musician is a jazz pianist and the robot performer is an off-the-shelf Strike 2 virtual drummer.[4] The drums parameters controlled by the AI

[4] AIR Music Technology, https://www.airmusictech.com/product/strike-2/.

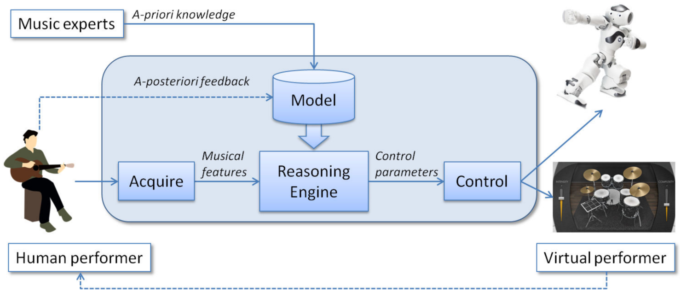

Fig. 1. The proposed methodology to control a virtual music partner

system include patterns, intensity, complexity, fills, instruments used, and enter or exit sequences. Acquisition is done through a MIDI interface and a feature extraction algorithm; controlling the drums parameters is done through MIDI. Upon discussions with the musicians, we found out that part of the knowledge of how the drummer's parameters depend on the pianist's play is conscious, and the musicians can easily expressed it in terms of approximate rules using vague linguistic terms, like:

> **If** the rhythmic complexity on the lower part of the keyboard is high,
> **then** the rhythmic complexity of the drums should increase.

Fuzzy logic offer suitable tools to encode this type of knowledge, and therefore we use it in our system. Note that, while rule-based systems have often been used for music composition and improvisation [9], the use of fuzzy logic in this field is much less explored [6].

3 System Design

The core of our system is a multiple-input multiple-output Fuzzy Inference System (FIS) [3], which implements the "Reasoning engine" block in Figure 1. The system runs at a fixed clock cycle, and it resembles the structure of a classical fuzzy controller. It takes as input a set of music parameters extracted in real time that describe the human execution, and it produces as output a set of control parameters for the virtual drummer. Differently from most conventional fuzzy controllers, the rules' conditions are not simply conjunctions of positive literals, but general formulas in (fuzzy) propositional logic. This gives us greater expressive power in representing the musician's knowledge. Because of this, the system has been implemented from scratch rather than relying on existing toolboxes.

Below we briefly list the mail elements of the overall system. A more detailed technical description can be found in the full version of this paper [11].

Input features. The interface between the software and the piano is implemented using the Python MIDO library. MIDI uses ports as interfaces between producers and consumers of MIDI messages and the system will continually poll the input port for MIDI messages. Form this the system extracts features both explicit and implicit. The features extracted include: velocity $v(t)$, rhythmic density $d(t)$, time since last note $T(t)$ and beat couter $b(t)$.

Temporal filters. Some of the knowledge expressed by the musicians implicitly refers to a temporal aspect, e.g., considering the average or the derivative of the intensity. These aspects could be captured in the feature extraction part by adding ad-hoc temporal filters. We opted instead for using a second FIS to extract relevant temporal features. This is a recurrent fuzzy system [1] that takes as input the current features at time t plus its own output at time $t-1$. This solution allows us to better capture the specific knowledge of the musician. The extracted temporal features include: $\bar{v}(t)$ (average velocity), $\bar{d}(t)$ (average density), $\Delta_v(t)$ (velocity slope), $\Delta_d(t)$ (density slope), and $\delta(t)$ (step change).

Anticipation. We have encoded a simple predictive model in the above temporal FIS to infer a coming climax or anti-climax from a change in intensity and complexity. The main FIS includes anticipatory rules that react to these forecasted features, e.g., anticipate a climax by starting a drums fill-in; or anticipate an anti-climax by muting the kick first, and then the snare once the change occurs.

Output parameters. The Strike 2 virtual drummer allows us to control its behaviour and settings by sending MIDI messages. Currently our software controls the intensity and complexity of the drummer as well as starting, stopping and changing the pattern (e.g., verse, bridge, chorus, fills, intros and outros) and muting individual parts of the kit.

Fuzzy inference. The main inferece system is a FIS based on the above input and output variables. It usese the usual fuzzify-inference-defuzzify pipeline [3]. A detailed description is in the full version of this paper [11].

4 Development and Testing

System development. The system has been implemented using Python 3.6.8 and the MIDO library (1.2.9). We used Strike 2 (2.0.7) as virtual drummer. The input comes from a MIDI piano, or from a MIDI file for debugging purposes.

Knowledge elicitation. The project includes people from computer science, music performance, audio engineering and philosophy. This highly inter-disciplinary nature requiree a careful process for the conceptual and practical development. Throughout the project, participants have kept journals on their thoughts, and various interaction means have been used — discussions, workshops, shared documents, examples of piano performance, and system demos. In the initial phases, piano recordings were analyzed by the performer himself through a process of open coding, where different features of the playing were identified and described; e.g. "phrase with high intensity", "build up in velocity", etc. These indications then provided a basis for identifying the relevant musical parameters and fuzzy rules in the AI system. Interestingly, the interaction has led to cross fertilization and mutual enrichment of the participants. For example, the need to describe music performance in logical terms led to the development of a new analytical perspective on how, when and why different styles are being chosen and used. On the other hand, the fuzzy models had to be enriched to meet the complexity of human musical performance, e.g., to change the feeling of intensity in the music using density of notes, change of notes registries, sustain pedal, or dynamics.

Testing. The project was done from the start in a tight loop between the musicians and the software developers. To allow this, we have first developed a simple but fully usable system, and then modified the system and the model incrementally in collaboration with the musicians. At the time of this writing, the system has not been evaluated by an external audience yet. This will happen in three public concerts, two on May 28 and one on June 12: we hope to report the results at the SAIS workshop.

5 Next Steps

So far we have used a pure knowledge-based approach. This allowed us to go through an open, modular and incremental development loop together with the music experts. We next plan to integrate this approach with a data-driven approach, e.g., to complete and/or adapt the rules as done, e.g., in [4].

References

1. Jürgen Adamy and Roland Kempf. Regularity and chaos in recurrent fuzzy systems. *Fuzzy Sets and Systems*, 140(2):259–284, 2003.
2. L De Miranda, M Rovatsos, and S Ramamoorthy. We, Anthrobot. In *Proc of Robo-Philosophy: What social robots can and should do:*, pages 48–59, 2016.
3. D Driankov. A reminder on fuzzy logic. In D Driankov and A Saffiotti, editors, *Fuzzy Logic Techniques for Auton Vehicle Navigation*, chapter 2. Springer, 2001.
4. A Friberg, R Bresin, and J Sundberg. Overview of the kth rule system for musical performance. *Advances in Cognitive Psychology*, 2(2-3):145–161, 2006.
5. Lejaren A Hiller Jr and Leonard M Isaacson. Musical composition with a high-speed digital computer. *J. of the Audio Engineering Society*, 6(3):154–160, 1958.
6. Chien-Hung Liu and Chuan-Kang Ting. Computational intelligence in music composition: A survey. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 1(1):2–15, 2017.
7. Bhavya Mor, Sunita Garhwal, and Ajay Kumar. A systematic literature review on computational musicology. *Archives of Computational Methods in Engineering*, pages 1–15, 2019.
8. Robert Rosen. *Anticipatory systems*, pages 313–370. Springer, 2012.
9. K Tatar and P Pasquier. Musical agents: A typology and state of the art towards musical metacreation. *Journal of New Music Research*, 48(1):56–105, 2019.
10. Belinda Thom. Interactive improvisational music companionship: A user-modeling approach. *User Modeling and User-Adapted Interaction*, 13(1-2):133–177, 2003.
11. O Thörn, P Fögel, P Knudsen, L de Miranda, and A Saffiotti. Anticipation in collaborative music performance using fuzzy systems: a case study. arXiv:submit/2688295, 2019.