HAL
open science

Appropriating Music Computing Practices Through Human-AI Collaboration

Hugo Scurto, Frédéric Bevilacqua

To cite this version:

Hugo Scurto, Frédéric Bevilacqua. Appropriating Music Computing Practices Through Human-AI Collaboration. Journées d'Informatique Musicale (JIM 2018), May 2018, Amiens, France. hal-01791504

HAL Id: hal-01791504

https://hal.science/hal-01791504v1

Submitted on 14 May 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

APPROPRIATING MUSIC COMPUTING PRACTICES THROUGH HUMAN-AI COLLABORATION

Hugo Scurto
Ircam - Centre Pompidou
STMS IRCAM–CNRS–SU
Hugo.Scurto@ircam.fr

Frédéric Bevilacqua
Ircam - Centre Pompidou
STMS IRCAM–CNRS–SU
Frederic.Bevilacqua@ircam.fr

ABSTRACT

This paper presents a prototypical computational framework for music computing appropriation. Putting Human-Computer Interaction at the center of the issue, it proposes to use Artificial Intelligence (AI) to assist humans in their first uses of music computing systems—namely, their exploration. We first review how interaction design may be central to exploration and appropriation of music computing systems, highlighting guidelines and potential directions to improve it. We then present our framework proposal, centered on exploration, detailing both its interactive workflow and the AI model at stake. We finally illustrate the pedagogical potential of our framework in two musical applications that we implemented, and discuss future research toward understanding how AI could be used by humans as collaborators for self-expression and appropriation of music computing practices.

1. INTRODUCTION

The introduction of the computer in music has brought a wealth of novel practices around sound and music. Cutting-edge technologies have been developed for sound synthesis, processing, analysis, and control, enabling the emergence of new music works, practices, notations, and performances.

However, such new technologies remain hard for people to appropriate. While music computing is now taught in many music institutions, it still suffers from its apparent complexity. As a consequence, many musicians stay attached to their classical practices and resign to appropriate music computing, while many musicians-to-be never take a chance to explore music computing systems. This is paradoxical as computing has become ubiquitous in the last ten years.

Can we think of a computer tool that would facilitate exploration and appropriation of music computing systems? In this paper we propose a computational framework for appropriating music computing. We first argue that the field of Human-Computer Interaction (HCI) may offer promising approaches to support appropriation by improving interface accessibility. We then propose to use Artificial Intelligence (AI) to provide users with human-centred interactions during their exploration of music computing tools. We finally illustrate our framework's potential with two example musical applications, and discuss future work to be done to better understand how AI could be used as collaborators by humans in their own appropriation of music computing and expression of musical ideas.

2. THE ROLE OF INTERACTION DESIGN IN MUSIC COMPUTING APPROPRIATION

We focus on a particular use case of music computing appropriation. The use case refers to the situation where a user makes use of a music computing system on his or her own, outside educational institutions. In this situation, appropriation issues arise during the first uses of the music computing system.

2.1. Appropriation in Music Computing

2.1.1. Gathering Information on a System

A first option to start using a system consists in gathering information on it—in a passive learning setup. Information on a system can be found in various media, from the most straightforward (*e.g.*, a text or video tutorial) to the most technical (*e.g.*, a research paper), as well as through online discussion (*e.g.*, an Internet forum). These activities are often time-consuming: users first have to find relevant information (which can be hard for obsolescent systems), then to filter it (*i.e.*, find what is useful for a specific goal), supposed that they have a specific goal in mind. Overall, time spent on passively learning hinders users to interact with sound and music directly, which might drive them away from experimenting with the system.

2.1.2. Experimenting with an Interface

A second option consists in starting interacting with the system's interface from scratch—in an active learning setup. Interacting with the interface implies trying many different actions directly to understand the functioning of the system (in a trial-and-error fashion), and eventually to achieve a specific goal. In our case of music computing, these activities are crucial as it is important for users to actively control sound so as to strengthen action-perception loops [15]. Yet, it is also possible that users get discouraged in interacting with the interface if they get too much

Hugo Scurto, Frédéric Bevilacqua, *Appropriating Music Computing Practices Through Human-AI Collaboration*, Journées d'Informatique Musicale (JIM 2018), Amiens, France, éd. L. Bigo, M. Giraud, R. Groult, F. Levé, pp. 115–120, 2018. CC BY-SA 4.0.

Figure 1. Example of interface in a music computing system (here, the u-he Bazille VST [1]).

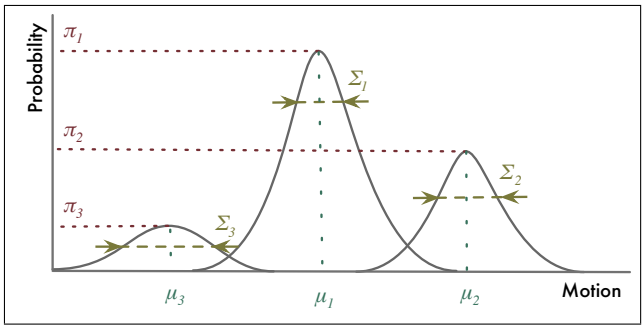

Figure 2. Example of model in a music computing system (here, a Gaussian Mixture Model for gesture-sound mapping [2] [13]).

error during their trials. Notions of appropriation [23] and novice to expert transition [7] are thus crucial in the design of interactions at stake in a given interface. Several works in the field of Human-Computer Interaction (HCI) have given guidelines for designing interactions in creative interfaces that facilitates appropriation [19]. We believe these could be applied to the design of interactions in music computing systems.

2.2. Human-Computer Interaction in Music Computing Systems

2.2.1. User Interface

We identify two drawbacks of current music computing systems in the context of appropriation. First, most music computing systems' interfaces can look quite intimidating for completely novice users. Some of them directly derive from their analog ancestors (*e.g.*, sound synthesis engines, see Figure 1, or spatialization tools [4]): they are thus designed for expert users, not to facilitate interaction for novice users. There do have been attempts to improve interface accessibility of these systems [6, 5, 20]. Yet, these interface simplifications are often done to the detriment of the system's abilities: after having appropriated some tasks, users become limited by the interface's lack of sophistication.

2.2.2. Models and Representations

The second drawback is that music computing systems often rely on complex models and representations that are not directly linked to sound or music. For example, improvisational systems [1] or gesture following systems [3] require users to have specific knowledge on models at stake (see Figure 2) to understand how parameters relate to system customization. Similarly, musical environemnts such as Max or PureData require to learn new representations as well as programming to start interacting. This drawback is common to both music novices and experts—who might know a certain amount of musical parameters, but do not know how they relate to new mathematical parameters—, and constitutes a major issue for music pedagogy—as such music computing systems have been shown promising for instrument learning [8] and embodied practice [2]. We believe redesigning interaction with these systems for improving appropriation could spread these pedagogical benefits among more people.

2.3. Interactive Machine Learning

2.3.1. Overview

In the last decade, the field of Interactive Machine Learning has enabled more accessibility and appropriation of music computing systems by implementing human-centred interactions. At the crossroads between HCI and AI, it studies human interaction with algorithms, and integrates human actions in the design of algorithms themselves. Its applications has enabled the emergence of new music practices [11], as well as new computational frameworks [13]. As an example, the "mapping-by-demonstration" framework developed by Françoise [12] allows users to build custom gesture-sound mappings by directly demonstrating examples of gestures while listening to sounds, thus improving accessibility compared to previously-cited systems.

2.3.2. Pedagogical applications

From a pedagogical point of view, the potential of interactive machine learning systems has been identified, yet little exploited. For gestural control of sound, they have been cited as allowing "learners to experience components of higher-level creativity and social interaction even before developing the prerequisite sensorimotor skills or academic knowledge" [18]. Interestingly, novel application domains, such as music therapy and musical expression for people with disabilities, have also emerged [21]. We believe extending interactive machine learning approaches to other music computing systems could constitute an opportunity to widen the reach of more music computing practices to more people.

[1] http://www.u-he.com/cms/bazille

[2] https://github.com/Ircam-RnD/xmm

3. CO-EXPLORATION: A FRAMEWORK PROPOSAL FOR AI-ASSISTED INTERACTION

We now present a framework proposal that focuses on assisting human exploration of music computing systems using AI. We first motivate the framework, then describe both its interactive workflow and AI modelling.

3.1. Motivation

3.1.1. Why assisting exploration?

Exploration is the early phase of learning during which a human iteratively acts on an interface and receives feedback information, allowing him or her to gradually grasp the system's functioning and qualities. As discussed previously, it is a crucial phase in appropriation regarding learning and skill development, as good or bad initial experience will determine the future degree of motivation and involvement of a learner term for a given task [7]. By aiming at a framework that assists this exploration phase, our wish is to lower the threshold for learners to directly interact with the system and sense its abilities, paving the way for further understandings of how the system actually works. Moreover, we argue that exploration may be a specific case of embodied interaction [15], in which expression plays a key role [16].

3.1.2. How to assist exploration?

To assist human exploration, we find it relevant to use the metaphor of transmission of knowledge between humans. Consider a human that has an idea but does not know how to convert it in a concrete realization. Usually, the human will ask assistance to a second human to realize this conversion—we call it the assistant. Iterative interaction between the two humans takes place, during which the assistant takes actions on the system and the human gives feedback on it—until converging to a final design. Our idea is to have an AI agent take the role of the assistant: AI acts on the system, upon which the human gives feedback. AI thus explores design possibilities in collaboration with the human, letting the human focus solely on aligning their conceptual space with the perceptual space offered by the AI—postponing the sensorimotor and/or academic learning phase to a later phase. Such co-adaptation phenomenon between the human and the machine has been shown as a useful mechanism for reducing the human's cognitive overload [17].

3.2. Workflow Definition

We propose to formalize the interactive workflow of such a framework—we call it "co-exploration" (see Figure 3).

3.2.1. Design Through Co-Exploration

Co-exploration stands for collaborative human-AI exploration of a given music computing system. The human explores the expressive abilities of the system (progressively learning interesting abilities in the system), while the AI explores the aesthetic preferences of the human (progressively learning which system parameters are relevant for the human). Design through co-exploration encapsulates the possibility for a (possibly novice) human to create a musical artifact from a (possibly unknown) computer system by collaborating with an AI agent in the exploration of design possibilities. A typical scenario would imply the agent generate an initial random solution to the human, who would progressively shape it through her preferences.

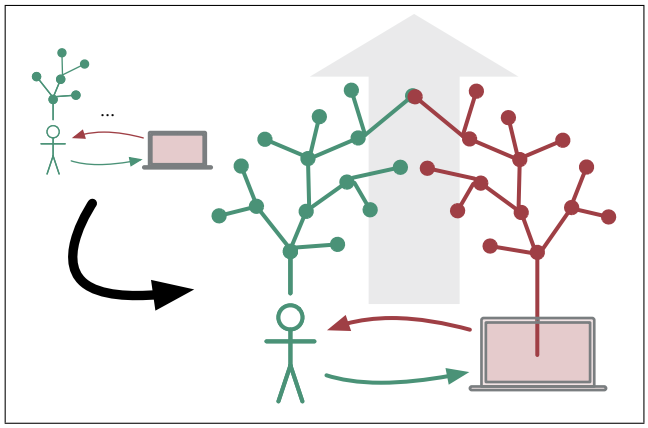

Figure 3. Co-exploration framework. In a standard situation (behind), a human explores a system by iteratively acting on it. In co-exploration (front), an AI agent explores a system in parallel to the human.

3.2.2. Feedback-mediated interaction

As we saw it, human interaction with the music computing system is mediated by an AI. Concretely, this means that the human does not interact with the system's interface: an AI agent does it instead. Instead, the human focuses on giving evaluative feedback on the AI agent's actions, judging the system's output on a perceptual level. Potentially, this feedback could be of any type—be it text-based, demonstration-based, or physiological.

3.2.3. Reinforcement feedback

We propose to investigate reinforcement feedback, which can be positive or negative. Advantages are threefold. First, it could encapsulate several kinds of feedback in one unique format, such as general advice (*e.g.* "this is good", "this is bad"), implicit knowledge (*e.g.* "do it more like this", "don't go that way"), as well as explicit specification (*e.g.* "this is exactly what I want", "never show me this again"). Second, it could be expressed relatively easily (compared to text-based feedback, which forces users to create a concrete verbalization of what they want). Third, it could give a sense of agency to the human (compared to physiological feedback, which most humans do not control). Overall, we hypothesize that communicating such high-level feedback could facilitate musical exploration of a system compared to specifying its low-level parameters.

3.3. AI Model

We propose to investigate the interactive use of a specific category of AI algorithms, called reinforcement learning, which applications in music computing have been few and far between [10, 9].

3.3.1. Reinforcement learning

Reinforcement learning defines a formal framework for the interaction between a learning agent and an environment in terms of states, actions, and rewards [22]. At time t, an agent senses its environment through an observation called state S_t (typically, a vector of discrete parameters), and on that basis takes an action A_t on it (typically, a set of discrete modifications on these parameters). At time $t+1$, in response to its action, the agent receives a reward R_{t+1} from the environment, as well as a new state S_{t+1}. From this information, the agent iterates interaction, progressively learning how to optimize interaction with the environment so as to maximize the total amount of reward it receives over the long run.

3.3.2. Learning from exploration

Reinforcement learning algorithms differ from supervised (and unsupervised) learning algorithms. For the latters, learning typically occurs offline on the basis of a static training dataset, which is a set of labeled (or unlabeled) examples we would like the system to generalize behaviour from. In reinforcement learning, the agent learns online by directly interacting with its environment. As a result, a reinforcement learning agent must always balance between exploration and exploitation to improve its learning – exploration meaning trying new actions to discover which ones yield the most reward, and exploitation meaning choosing the best actions in terms of reward at the time of computation.

3.3.3. Requirements for human interaction

In the case of co-exploration, we must add another element to the formal framework defined above. We propose that a human would be responsible for giving reward to the learning agent, as a consequence of the agent's action toward the environment's state (see Figure 4). Our hypothesis is that the numerical reward could constitute a feedback channel from the human to the AI agent (*e.g.* telling the agent that its action has been good or bad from the human's subjective point of view).

Interactive agent teaching have been investigated in previous research, leading to the creation of efficient learning agents [14]. Our research differs from these works in the sense that it focuses on interactive agent teaching from the human point of view (how it is "efficient" for the human, not necessarily for the agent), and that the tasks to be learned in creative activities may have different properties than those in goal-oriented activities (such as learning how to play Tetris).

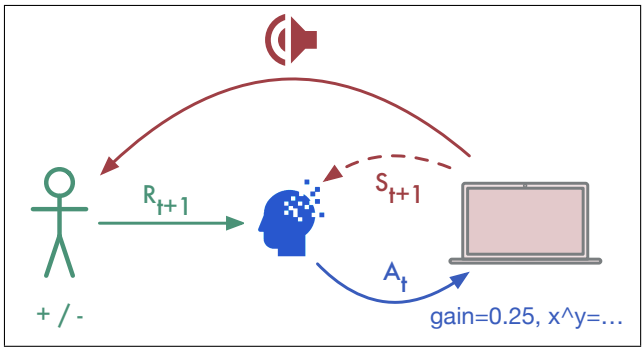

Figure 4. Co-exploration workflow. Interaction between the human and the music computing system is mediated by an AI agent (in blue). At time t, the agent acts directly on the system's parameters. At time $t+1$, the system generates a new state (for example, a sound), that is subjectively evaluated by the human through a reinforcement feedback. By iterating the loop, the agent learns how to co-explore.

4. EXAMPLE MUSICAL APPLICATIONS

We implemented a first prototype of our co-exploration agent—we call them "co-explorers"—and applied it within two existing music computing systems.

4.1. Implementation

We are currently implementing coax, a software library for collaborative human-AI exploration. Its architecture will allow users to choose among a variety of reinforcement learning agents, and to modify some high-level parameters that control their exploration. Agents can be connected to any kind of interactive music system that sends and receives OSC messages. The current prototype implements the Sarsa learning algorithm [22] upon a set of Python classes. Its interface consists in two buttons: one for indicating positive feedback, one for negative—we will discuss further improvements in next section. We implemented our two applications in the Max/MSP environment.

4.2. First application: VST Exploration

4.2.1. Motivation

The first application connects an AI agent to a VST (*u-he Bazille*, see Figure 1). VSTs are software interfaces that allows to create sounds by combining various synthesis algorithms and audio processings. Their interface typically consists of knobs and faders, directly linked to low-level parameters of synthesis and processing. While these music computing systems are widely used and effective among expert composers, their interface may constitute a huge barrier for novice users to launch into exploring and understanding their functioning, preventing them from appropriating digital sound creation.

4.2.2. Description

The workflow consists of users first listening to a random sound generated by the VST, give positive or negative feedback on it depending of their subjective evaluation of the sound, then listen to a new sound that was synthesized taking into account previous feedback. Formally, the environment's state consists of a vector of twelve VST parameters; the agent's actions consists of moving one of these parameters up or down. The resulting collaboration consists in the user focusing on developing its listening abilities and tastes toward the VST, while the AI agent learns how to convert the user's tastes in terms of VST parametrization. A preliminary evaluation led with four expert computer musicians experimenting with our system with three different VSTs confirmed our initial conjectures on how useful co-exploration could be for appropriating a VST—we should discuss such results in a future publication.

4.3. Second application: Mapping Exploration

4.3.1. Motivation

The second application connects an AI agent to a gesture-sound mapping (using the *XMM* library, see Figure 2). Mappings are complex functions that link gestural devices to sound synthesizers. They allow performers to focus on their physical gestures in their musical practice and embodied engagement with sound. However, they typically imply a phase of parametrization, which require knowledge on mathematical modelling and programming. This can hinder novice users as well as expert musicians to create their own gestural controller and develop self-expression.

4.3.2. Description

The workflow consists of users first experimenting with a random gesture-sound mapping, give positive or negative feedback on it depending of their subjective evaluation of the mapping, then experiment with a new mapping that was generated taking into account previous feedback. Formally, the environment's state consists of a vector of twelve mapping parameters; the agent's actions consists of moving one of these parameters up or down. The resulting collaboration consists in the user focusing on developing its sense of corporeal engagement with sound, while the AI agent learns how to convert it in terms of mapping parametrization. Ultimately, as users give feedback on mappings that they experiment with, the AI agent would be able to generate mappings that either suit them (exploitation), or aim at surprising them (exploration).

5. CONCLUSION AND FUTURE WORK

In this paper we have formulated a scientific hypothesis on how human-AI collaboration could support music computing appropriation. Part of future work will consist of refining the design of learning algorithms, as to optimize human-AI exploration. Also, a general interface for co-exploration might be designed. We will keep on testing our framework with other music computing systems as well to spread the range of musical activities covered—for example, connecting co-explorers with sequential music content (such as notes or events).

An important part of future work will consist of evaluating our framework with expert users, *i.e.* users that work around music at a professional level. We are currently leading and building a set of case studies that aim at better understanding how AI agents could assist these users in their exploration of music computing practices. Specifically, our aims are (1) assessing whether co-exploration might be useful for them compared to their standard exploration strategies, and (2) studying how co-exploration might be useful on a longer term along their creative practice. These two-scale studies would allow us to better quantify the pedagogical benefits our framework might offer.

Finally, we will lead real-world applications with novice users, *i.e.* users that are neither musicians, nor computer specialists. The goal of these applications will rely on observing how co-exploration might constitute an entry point for these novice users to music computing practices. Based on previous works [18], we envision that our interactive machine learning workflow may allow novices to experience creative and social aspects of music before devoting time to learning and understanding technical aspects. Overall, we will be able to better understand how AI could be used by humans as collaborators for self-expression and appropriation of novel music computing practices.

6. ACKNOWLEDGEMENTS

We thank Baptiste Caramiaux, Jules Françoise and Benjamin Matuszewski for useful discussions and suggestions.

7. REFERENCES

[1] G. Assayag, G. Bloch, M. Chemillier, A. Cont, and S. Dubnov. Omax brothers: a dynamic topology of agents for improvization learning. In *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*, pages 125–132. ACM, 2006.

[2] F. Bevilacqua, F. Guédy, N. Schnell, E. Fléty, and N. Leroy. Wireless sensor interface and gesture-follower for music pedagogy. In *Proceedings of the 7th International Conference on New Interfaces for Musical Expression (NIME)*, pages 124–129. ACM, 2007.

[3] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana. Continuous realtime gesture following and recognition. In *International gesture workshop*, pages 73–84. Springer, 2009.

[4] T. Carpentier, M. Noisternig, and O. Warusfel. Twenty years of ircam spat: looking back, looking forward. In *41st International Computer Music Conference (ICMC)*, pages 270–277, 2015.

[5] M. Cartwright and B. Pardo. Synthassist: Querying an audio synthesizer by vocal imitation. In *Proceedings of the 14th International Conference on New Interfaces for Musical Expression (NIME)*, 2014.

[6] M. Cartwright, B. Pardo, and J. Reiss. Mixploration: Rethinking the audio mixer interface. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, pages 365–370. ACM, 2014.

[7] A. Cockburn, C. Gutwin, J. Scarr, and S. Malacria. Supporting novice to expert transitions in user interfaces. *ACM Computing Surveys (CSUR)*, 47(2):31, 2015.

[8] A. Cont. Antescofo: Anticipatory synchronization and control of interactive parameters in computer music. In *International Computer Music Conference (ICMC)*, pages 33–40, 2008.

[9] N. Derbinsky and G. Essl. Exploring reinforcement learning for mobile percussive collaboration. In *Proceedings of the 2012 International Conference on New Interfaces for Musical Expression (NIME)*, 2012.

[10] R. Fiebrink and B. Caramiaux. The machine learning algorithm as creative musical tool. *arXiv preprint arXiv:1611.00379*, 2016.

[11] R. Fiebrink, D. Trueman, N. C. Britt, M. Nagai, K. Kaczmarek, M. Early, M. Daniel, A. Hege, and P. R. Cook. Toward understanding human-computer interaction in composing the instrument. In *Proceedings of the 2010 International Computer Music Conference (ICMC)*, 2010.

[12] J. Françoise. *Motion-sound mapping by demonstration*. PhD thesis, Université Pierre et Marie Curie, 2015.

[13] J. Françoise, N. Schnell, R. Borghesi, and F. Bevilacqua. Probabilistic models for designing motion and sound relationships. In *Proceedings of the 2014 International Conference on New Interfaces for Musical Expression (NIME)*, pages 287–292, 2014.

[14] W. B. Knox and P. Stone. Interactively shaping agents via human reinforcement: The tamer framework. In *Proceedings of the fifth international conference on Knowledge capture*, pages 9–16. ACM, 2009.

[15] M. Leman. *Embodied music cognition and mediation technology*. MIT Press, 2008.

[16] M. Leman. *The expressive moment: How interaction (with music) shapes human empowerment*. MIT press, 2016.

[17] W. E. Mackay. Responding to cognitive overload: Co-adaptation between users and technology. *Intellectica*, 30(1):177–193, 2000.

[18] D. Morris and R. Fiebrink. Using machine learning to support pedagogy in the arts. In *Personal and ubiquitous computing*, pages 1631–1635. Springer, 2013.

[19] M. Resnick, B. Myers, K. Nakakoji, B. Shneiderman, R. Pausch, T. Selker, and M. Eisenberg. Design principles for tools to support creative thinking. In *National Science Foundation Workshop on Creativity Support Tools. Washington DC.*, 2005.

[20] D. Schwarz, G. Beller, B. Verbrugghe, and S. Britton. Real-time corpus-based concatenative synthesis with catart. In *9th International Conference on Digital Audio Effects (DAFx)*, pages 279–282, 2006.

[21] H. Scurto, R. Fiebrink, et al. Grab-and-play mapping: Creative machine learning approaches for musical inclusion and exploration. In *Proceedings of the 2016 International Computer Music Conference*, 2016.

[22] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 2011.

[23] V. Zappi and A. McPherson. Dimensionality and appropriation in digital musical instrument design. In *Proceedings of the 2014 International Conference on New Interfaces for Musical Expression (NIME)*, pages 455–460, 2014.