Sharing the Same Sound: Reflecting on Interactions between a Live Coder and a Violinist

Francesco Ardan Dal Rì
DISI - Department of Information Engineering and Computer Science, University of Trento
Via Sommarive 9, 38123;
Conservatory of Music F. A. Bonporti
Via S. Giovanni Bosco 4, 38122
Trento, Italy
ardan.exp@gmail.com

Francesca Zanghellini
Conservatory of Music C. Monteverdi
Piazza Domenicani 19, 39100
Bolzano, Italy
fzanghellini@unibz.it

Raul Masu
Institute of Music, Science and Engineering. King Mongkut's Institute of Technology Ladkrabang
Chalongkrung Road, Ladkrabang, 10520
Bangkok, Thailand
raul@raulmasu.org

ABSTRACT

This paper introduces a performative ecosystem created with the aim of promoting a joint expression between a live coder and an instrumentalist. The live coding environment is based on TidalCycles[1], controlling a sample machine implemented in SuperCollider[2]. The instrumentalist can record short samples of his/her playing in different buffers, which the live coder can then process. The ecosystem was intensively used by the first and the second author of this paper (respectively live coder and violinist) to develop a performance. At the end of this paper, we provide a number of reflections on the entanglement of the different roles and agencies that emerged during the rehearsals.

Author Keywords

Live coding, collaboration, agency, performance ecology, interactions

CCS Concepts

•Applied computing → Sound and music computing; Performing arts;

1. INTRODUCTION

Among the various collaborative strategies developed within the field of computer music, live coding has become one of the available and investigated practices. While a large majority of collective live coding performance experiences focuses on laptop ensembles (e.g. [11]), networked ensembles (e.g. [53]), or collaborative live coding environments (e.g. [51, 50]), some of them also explore musical co-creation with acoustic musical instruments (e.g. [40, 33, 4]). However, in these performances, the instrumentalist and the live coder generally act autonomously, and every musician is primarily responsible for his/her own sound.

In this paper, we present a novel interactive ecosystem in which each performer (in our case a live coder and a violinist) can act, in different ways and with different tools, on the other's musical actions. The system was developed by the first two authors of this paper, who tested it through a series of rehearsals, with the idea of developing a permanent musical project. After each rehearsal, they collected comments regarding their performative strategies and the musical roles they assumed. Then, these comments have been clustered and discussed, providing a structured reflection on affordances and shared agency that emerge in such an interactive collaborative system.

2. BACKGROUND

As this paper investigates the relationship between a live coder and a violinist, the idea of performance ecosystems provides us with a general lens to look at our case. From this general perspective, we funnel our references first to collaborative music systems, and finally, systems that integrate live coding with instrumental practices.

2.1 Performance Ecosystems

In the last few decades, the idea of a *performance ecology* or *ecosystem* has emerged to scrutinize the complex network of relations among musicians, artifacts and instruments, and the environment [60, 17]. For istance Gurevich and Trevino discussed the "relationships between composers, performers, and listeners as a part of a system" [17]. Recently, the ARCAA framework [34] has been proposed to analyze performance, as a specific ecology of human actors and artifacts - see [5, 2, 27] for a reference on artifact ecology - and suggests scrutinizing roles (of persons), contexts, and activities [34]. The framework was used in many studies (e.g. [37, 35]). In our analysis, we will use this model to reflect on the role which emerged.

Performance ecologies are also relevant to reflect on cognitive musicking processes. As humans, we tend to offload part of our cognitive processes in the tools that we use [8], developing an embodied relationship with the surrounding

[1]https://tidalcycles.org/
[2]https://supercollider.github.io/

Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Copyright remains with the author(s).

NIME'23, 31 May–2 June, 2023, Mexico City, Mexico.

environment which grounds our processes in our experience [59]. Consequently, the tools that surround us in our experience of music-making have an impact on the process itself, music instruments being an increased encapsulation of musical theory [30].

Such a relation has an impact on agency, which is distributed among people and objects in performances [58]. For instance, Melbye [43] discusses how the musical agency of his practice is shared between him and his FAAB (an augmented self-resonating double bass), accounting for this relation as an ecology of human and non-human actors. In this context, the term *actor* assumes a Latourian meaning: "anything that does modify a state of affairs by making a difference is an actor – or, if it has no figuration yet, an actant". [22]. In relation to distributed agency, Stapleton has recently argued in favour of the value of ambiguity as a resource to nurture multiple ideas in relation to one system [57].

2.2 Collaborative Music Systems

Within performance ecosystems, this paper particularly focuses on collaboration among the musicians involved. An early example is represented by the electro-acoustic piece *Microphonie*, composed by Stockhausen in 1965, where one performer strikes a big tam-tam with different objects, a second performer moves a microphone to capture the sound in different positions in relation to the surface of the tam-tam, and finally, a third musician manipulates the sound by playing with an analogue bandpass filter [44].

In the computer music domain, the most evident examples of collaboration are probably offered by laptop ensembles (e.g. [7, 49]). Laptop ensembles often keep a traditional form of music collaboration, where each musician is responsible for his/her own part. However, electronic and computer music allows for a form of collaboration where multiple persons can manipulate the same sound stream.

Examples span from the notorious *Reactable*, [20] where multiple performers can collectively act on music parameters by interacting with tangible objects [21], to the *Music Room* [45], an installation where a couple of persons manipulate an algorithmically generated music stream by freely moving in a room. In the scope of music and sound technology for dance, a few papers have recently investigated the individual perspective of the various actors (i.e. dancers, choreographers, musicians, technicians) [36, 12].

In the area of live coding, a variety of systems and studies specifically rely on collaboration. Many examples focused on collaboration on web browsers [52, 39], networked performances [25, 49, 24], or learning environments [61, 3].

2.3 Live Coding and Traditional Instruments

Although the literature relating to live coding practices in collaborative environments is extremely rich, as shown at the end of 2.2, the number of examples relating to such practices together with instrumentalists is sensibly smaller. It is possible to subdivide these experiences into three main categories.

First, we have examples where the coding and the instrumental practice are performed by the same person. In this case, live coding systems are used as a technological compendium or as a companion to instrumental improvisation. For example, Hoogland uses his live coding language Mercury [19] along with his acoustic drum performance [4], Gorelik live codes a Disklavier while interacting with it [15], MrReason loops his guitar playing with TidalCycles [47], Alexandra Cárdenas uses an electric guitar to trigger an autonomous live coding system [10].

Secondly, we have examples where live coding systems are used to generate real-time notation for instrumentalists. In this case, the live coder acts as an orchestrator/conductor, controlling the development of the performance. This category includes systems such as Pitchcircle3D [18], which displays a graphical representation of the circle of fifths, SGLC [26], generating real-time notation for musicians using short commands, and CMN (Code Music Notation) used to create live notation for a marimba player [29].

Finally, we have examples where live coding systems are used to produce sound alongside other instrumentalists in improvisational contexts. In this case, the live coder takes the role of an actual musician, and the sound output is the direct result of the sum of individual actions of everyone involved playing their own instruments, as in any traditional ensemble. Duo performances are generally more frequent (e.g. [28, 56, 40]), but we also have reports of systems used in small ensembles [33] up to improvisations with entire orchestras [16]. The different nature of the practices involved opens up a series of reflections. For instance, discussing collaboration with a percussionist, McLean reflected on the importance of speeding up music creation and showcased how TidalCycles was effective in supporting his workflow [40].

Whilst many examples that integrate instrumentalists and live coders into a joint musical expression exist, the vast majority of those examples are actual performances and not academic reflections. As such, our contribution is grounded on a spreading practice, and we hope it can broaden the related debate.

3. THE SYSTEM

The structure of the system was implemented as follows.

The core sound engine, developed in SuperCollider, is based on five one-second length buffers and five SynthDefs: one simply passing the violin signal to the speakers, one for recording samples in the buffers, the remaining three for playing back the samples in different ways.

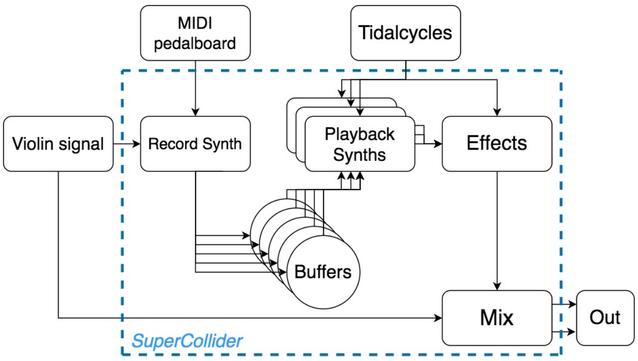

Figure 1: Block scheme of the proposed system.

The recording Synth takes the incoming signal, applies an ASR envelope on it to avoid digital clicks, and records it into one of the five buffers according to the footswitch pressed on a MIDI pedalboard. The playback Synths can access and play specific buffers by their index, declared in the live coding environment via a required argument. Being the ecosystem based on the TidalCycles live coding library, they have been implemented according to the typical structure of a SynthDef for TidalCycles[3], and can therefore

[3]`http://tidalcycles.org/docs/configuration/adding\_synthesizers/`

receive additional control arguments (e.g. volume, reading speed, start/end point, etc.), as well as processing the sample through a variety of effects. The first Synth is used for transposing the sample at various rates and intervals, the second is based on a granular UGen which allows to create clouds of grains, and the third uses the samples as a modulator signal for an FM engine.

During the performance, the instrumentalist can record a short, single sample on a specific buffer by pressing the corresponding footswitch on a MIDI pedalboard. At this point, the sample became available for the live coder to be used, manipulated, and organized into patterns. At any moment, the instrumentalist can also decide to overwrite the content of an already filled buffer, consequently varying the musical result generated by the live coding library. An overall pipeline of the system is presented in Figure 1.

Following initial rehearsals, the need for visual support emerged, allowing the live coder to see which buffers were recorded/overdubbed, and the instrumentalist to see in real-time which buffers were playing. Therefore, a visualization system has been implemented in atom-p5.js[4], which allows plotting background images directly in the text editor used for live coding (Atom[5]). An example can be seen in Figure 2. The smaller circles at the bottom right represent the content of the five buffers: initially empty, they are filled as soon as a sample is recorded by the instrumentalist, and their colour changes whenever the buffer is overdubbed. The larger, coloured circles in the centre of the screen represent the samples that are being recalled, and they light up whenever the corresponding sample, or part of it, is triggered by the live coding system. Their stay on screen has a fixed duration (they gradually fade into the background) and is not representative of their actual duration in time. In line with the general praxis of showing the screen during a live coding performance [23], the visuals are projected together with the code, and therefore can also be seen by the instrumentalist.

Further fine-tuning was performed during the intensive week of rehearsals, according to minor issues and ideas that emerged.

The code is open and available at `https://github.com/return-nihil/Live_Coding/tree/main/Duo`

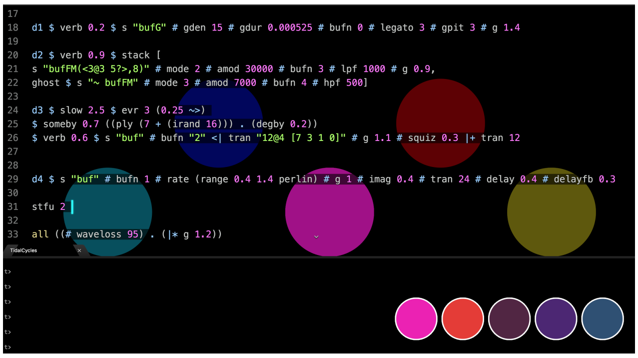

Figure 2: A screenshot of the live coder's interface.

4. THE SYSTEM IN USE: METHODOLOGY

The first two authors of the paper (respectively live coder and violinist) used the system for creating a musical piece with the aim of performing in public concerts[6].

First, the two musicians used the ecosystem in daily sessions (Figure 3) for a week of intensive initial exploration. Afterwards, they continued to rehearse for approximately one month with a more sparse schedule (one or two sessions per week, with a total of 15 sessions). Throughout this period, the two musicians underwent an autoethnographic process, using the technique as it has been adapted to HCI research [48].

At the end of each session, the two musicians noted their main feelings and a short description of their experience with the ecosystem in a diary, including the main elements that emerged and the roles/activities they developed at each step. The diary was used to ground a reflection-on-action [54], which is a research task that "takes place after the activity and enables the exploration of what happened and why in order to develop questions, ideas, and examples about the activities and practices in focus." [55]. To initiate such a reflexivity process, we analyzed the diary using thematic analysis [6]. We progressively encoded and clustered the logs, and we identified the main characteristics of the two performers' experiences, which we report in the next section.

5. TWO MUSICIANS PERSPECTIVES

Whilst presenting here the perspectives of the two performers, we integrate direct quotes - in *italics* - from the diaries we kept throughout this time. As in [13], we decided to speak of the two instrumentalists in third person. Despite the fact that we primarily collected data from two authors' biographical perspectives, the final text was worked collectively by the three authors of this paper.

5.1 The Live Coder's Perspective

From the live coder's point of view, it emerged that he considered the performances as substantially different from his previous live coding experiences.

Firstly, he perceived his **reaction times** as drastically slower with respect to the instrumentalist's. Indeed, the live coder often found himself in the condition of "*having to chase on the different ideas and changes of musical situation proposed*". Despite TidalCycles enabling a quite fast declaration of musical structures, he immediately felt the need to readjust his system implementing several high-level shortcuts, allowing him faster typing. This difficulty was further accentuated by the interactive aspect of the ecosystem, and situations often arose in which he was "*coding with a specific musical idea in mind, but in the meantime, the violinist overwrote the sample I was trying to process*". This involved deeper reasoning about **performative strategies**, requiring the live coder to deal with complex structures in a build-up manner, or, in a less deterministic fashion, to take this possibility into account as an aleatoric aspect of his performative actions.

Another aspect perceived as different is related to the **musical material**. In his usual performances, the live coder used a series of very different samples/synths, to which he attributed specific musical functions with which he found himself comfortable (e.g. short/percussive samples → rhythm, monophonic synths → melodies, etc.). Having in this case only five samples, moreover very similar to each other in terms of timbre, and without the possibility of knowing them in advance, the musical functions must be thought out and built on the fly. While returning a "*very coherent*"

[4]https://github.com/ndr-brt/atom-p5js
[5]https://github.com/atom/atom

[6]A video from a performance is visible at `https://youtu.be/DeXHBE2wFgg`

musical result, this adds a further level of complexity, which has led to a modification of the musical choices implemented by the live coder. Specifically, his attention has shifted from rhythmic creation and structuring to the application of a greater number of effects on the samples themselves, in order to "*variate their timbral nature [...] and place them more clearly in the musical functional space*".

For these reasons, regarding the **roles** assumed during the performances, he also perceived himself as an orchestrator, as his musical thought could not always be linked to a precise compositional idea, but rather to the extemporaneous combination of single samples with specific, already developed structures.

From the point of view of the relationship with the instrumentalist, on the other hand, the live coder noticed that **the code** produced at the end of the rehearsals **was much sparser** than usual, not only in terms of the actual amount of lines but also in terms of complexity. Not having to cover the whole musical space all by himself, we can argue that the reduced musical density is largely due to sharing it with another musician. However, other aspects to consider have emerged: specifically, the live coder has often perceived the need to simplify the structures in order to better "*clarify the roles of each within the performance*", as well as to leave some solo spaces to the violinist, allowing her to "*play and interact freely with the system as if it were some kind of creative looper*".

Finally, we report some reflections of the live coder regarding the **visual component**, which was perceived as particularly useful, since a large part of the cognitive process was in fact offloaded to the visual support. Specifically, being able to see which buffers have already been filled or overwritten during the performance helped him to immediately understand whether a certain musical change was due to "*random variations declared in my code or to the violinist's actions*". Furthermore, the possibility of seeing which buffer was playing in a given moment allowed the live coder to speed up his acting: "*Given the extemporaneous nature of the samples, which are not only new every time, but can even change over time, it was difficult for me to immediately identify the connection on which I wanted to act. Being able to connect the auditory stimulus to the visual one, I could identify them much easier*".

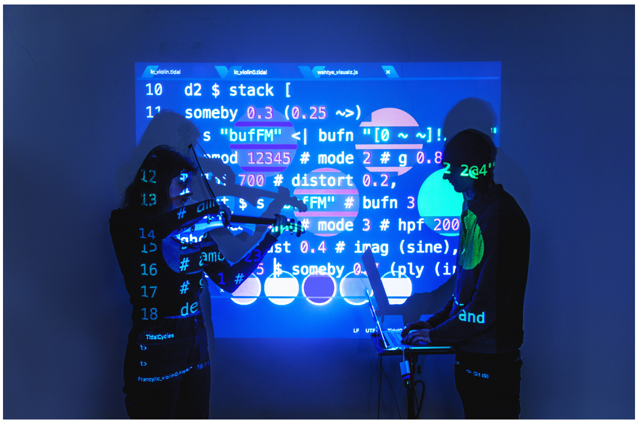

Figure 3: The two musicians during a rehearsal.

5.2 The Violinist's Perspective

From the violinist's point of view, it was possible to distinguish two different roles during the performance: one in which she focused on influencing the structures arranged by the live coder, acting as a "*composer*", and one in which she improvised on the background created by the live coder, assuming the role of "*soloist/co-improviser*".

As a **composer**, two different performative moments emerged: one in which she provided the live coder with samples, in order for him to build musical structures, and another in which she overwrote her recordings, directly acting on the timbral result produced by the coded sequences. In the first part of the performance, she found it efficient to personally organize the allocation of samples. With five slots available, she generally divided them into two main parts: buffers 1-2 were filled with low-pitch notes, and buffers 3-4-5 with higher notes or timbral effects (tremolo, pizzicato chords, fluted notes, etc). In this way, she allowed the live coder to "*start to easily build up some simple structures to be used as background*". After these musical structures began to evolve, she usually started replacing what was previously recorded, thus increasing the degree of functional complexity. In this phase, three main strategies emerged: the five samples were managed as "*harmonic fields*" (single notes belonging to the same scale/mode), as "*timbre variations*" (same note but played with different techniques), or as "*effects*" (like tremolo, trill, scratchy bow noise, col legno notes, etc). This procedure was considered musically effective, mainly because "*the timbral changes often highlighted the musical structures produced by the live coder*".

As a **soloist**, she often perceived the coded counterpart as structural scaffolding upon which she could improvise freely. Indeed, she refers to these structures as "*background*". Nevertheless, the violinist also pointed out the need for balance between lyric/melodic parts in the foreground and structural/rhythmic parts where she cooperates with the live coder in building patterns. Another element often emerged is constant attention to musical density: she usually found it difficult to manage rests and silences, and she felt the need to play a lot, but this resulted in "*very dense musical pieces with an overall linear development*", almost filled with continuous musical gestured from the violinist herself.

In terms of the **relationship with the live coder**, the violinist also perceived substantial differences in reaction times with respect to the live coder. In fact, the interplay did not seem convincing when she propose a repetitive pattern, waiting for the live coder to reproduce it, because he usually takes too much time to reply. However, when this procedure was reversed, the interplay worked much better: when the live coder evaluated a pattern, the violinist found it "*easy to repropose*" with her instrument. This slow response was perceived as a limitation by the violinist, which constrained her in "*adapting to the live coder's performative speed*".

Aspects related to the **shared agency** with the live coder were also highlighted. In particular, she primarily focused on the melodic/harmonic and timbral features of the musical pieces, while delegating most rhythmical and structural parts to the live coder. In this sense, she felt comfortable following the structural changes proposed by her counterpart, due to the faster instrumental reaction times, and perceived these roles as musically effective. Although "*obliged in somehow following the coding*", the violinist also contributed to the musical creation by imitating the coded structures, "*musically modify and enrich the patterns and creating interesting musical dialogues*". Further reflecting on the interplay, the violinist developed improvisational strategies in order to musically clarify "*who is creating what*" during the performances, consequently enhancing the imitation game.

Finally, the **visual part** has been positively evaluated and helped her to cope with a few issues that emerged during the first rehearsals. In fact, the violinist found it sometimes difficult to recall by hearing what samples she previously recorded in a specific buffer, due to the eventual processing applied by the live coder. By visually identifying the sample played in real-time, she could therefore act on them purposefully. A minor problem solved by the visual part was related to mechanical mistakes in pressing the MIDI footswitches while performing: "*sometimes, I was not sure to have actually recorded something*". The violinist also suggested some possible strategies for conducting the performance and communicating with the live coder using a series of visual messages (e.g. "*repetitively pressing a footswitch*" means "*do something with this sample*").

6. DISCUSSIONS

6.1 Roles in the Ecology

We look at the role that emerges during the live interaction. We acknowledge that many musical strategies are mediated by discussion occurring during the breaks. However, we decided to look only at this level of engagement because we aim at reflecting on the various roles that emerge in the entangled live relationship with the system [14], and facilitate a zoom-in by limiting the complexity of the analysis. To this end, we used a simplified version of the ARCAA framework [34]. As, in this discussion, we primarily look at the interactions occurring during the performance, we decided to remove the context layer and use the framework to visualise only the different roles and actions that emerged - Figure 4.

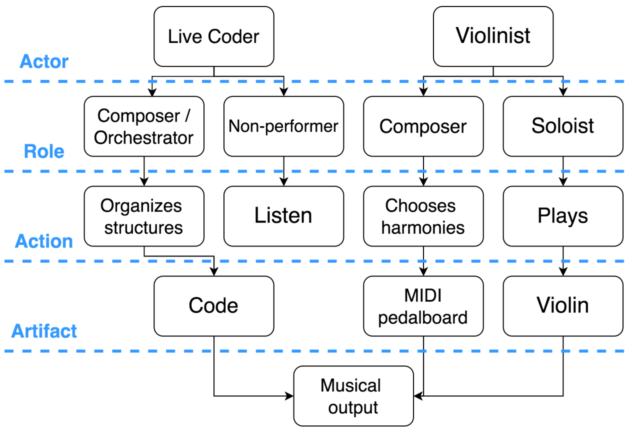

Figure 4: ARCAA framework applied to this study.

The violinist's creative act is separated into two main sets of musical activities which reflect the traditional melody-and-accompaniment subdivision. Given her background in Western classical music, she often refers to moments of interaction with the ecosystem as "*composing*" and "*creating harmonic fields*", while at other times she constructs more elaborate melodic lines, like a soloist playing over an accompaniment. As has already been pointed out in other similar studies (e.g. [38]), also in this case it emerges how the personal background influences the relationship with technology in determining the affordances of the interactive system. In line with the idea that affordances are learned in social contexts [9], the violinist's cognitive musicking process relies on long-learned practices which co-determined the affordances within the ecosystem.

The live coder practice was greatly affected by the interactive environment. Indeed, a substantial difference in reaction times emerged: while the violinist could almost immediately switch the musical situation, it takes much longer for the live coder to adapt to musical changes. This aspect, formalized by McLean as "idea-to-code latency" [41], is a well-known factor in live coding and is generally embraced as an intrinsic characteristic of the practice, without being perceived as a limitation. In this case, however, this factor influenced the live coder to the point that he identified moments of *non-performing*, in which he let the instrumentalist interact autonomously with the system, without operating any changes to the code. Furthermore, it emerged that his performative role was more similar to that of an orchestrator, since his musical material depended exclusively on what was proposed by the instrumentalist, and therefore, could not be known from the outset. Rather than developing complex musical structures, he concentrated more on defining spaces of rhythmic and interval possibilities, within which he organized specific sounds provided by the violinist, similar to what happens in other cases of collaboration between electronic instruments and classical ones [38].

6.2 Performative Strategies and Shared Agency

Due to the intrinsic different characteristics of the two practices involved, in 5 we have seen how different performative strategies accordingly emerged. Overall, while the violinist perceived the system as a context she needed to adapt to, the live coder mostly worked to overcome different challenges, both technical ("*trying to keep pace with the instrumentalist*") and musical ("*not knowing the samples in advance*").

While looking at their relations not in terms of actions and processes but in terms of creation and agency, both performers had to negotiate part of their agency with the system and with each other, allowing both moments to listen to the result and decide how to intervene, without having to act constantly. Therefore, the musical processes were offloaded not only on the "tool" (the system itself) as widely discussed in the literature (e.g. [58]) but also in a space of possible musical actions co-determined by the action of the other performer. Indeed, in line with what was discussed in [30], it was also evident how the dual interactions of this ecosystem had a strong impact on their musical processes, which changed by adapting to the interactive environment. For instance, the combination of the violin with the system determined a new encapsulation of musical theory for the instrumentalist, who started to think much more harmonically, or a more bottom-up approach for the live coder, who proceeded to assemble the musical structures in smaller steps. However, these processes are mutually influencing each other and, in terms of agency, the music creation is shared.

While observing the different actions in these performances, as we did in 5, it is useful to look at the interaction from a design and cognitive standpoint. From the overall perspective of agency of music creation, it is more complex to discriminate between composing and performing, and the agency of the various human and non-human actors. As such, if we look at ecology from the perspective of agency, the distinction between roles and actions becomes very blurred. This is in line with Latour's Actor Network Theory [22], which proposed that every component in a system, either human or non-human, can have agency, as discussed in various papers such as [58, 57]. In our case, this is particularly relevant as the sonic output is mutated by all the actors, combining the personal experience of the two musicians with the classical tradition inscribed in the violin [32] and an explicit type of formalization as live coding is [31].

6.3 Collaboration

Finally, since this paper offers new strategies for possible collaboration between live coders and instrumentalists, we wish to propose some considerations regarding the collaborative aspect of such an interactive system. Although improvisational situations were highlighted in which both performers each acted on their own instrument (similar to [33, 40]), the ecosystem favoured moments of other forms of interaction. Indeed, while sharing many similarities to a live sampling system in which musical material is sampled during the performance itself (e.g. [1]), with the on-top addition of a certain degree of autonomy provided by a live coding library (e.g. [42]), the ecosystem presented here offers a novel perspective of collaboration. Specifically, some decisive choices within the musical creation that are generally made by the same actor are in our case delegated to the other: the one who organizes the musical material (the live coder) can neither know the material a priori nor choose when to change it, while the one who supplies it (the violinist) can't decide how and when it will be used. As seen in 6.2, this peculiarity further blurred the distinction between different musical roles during the same performance, allowing the development of co-mediated performative strategies, and consequently determining new spaces of co-agency. As such, we propose that our ecosystem offers novel ways of computer-mediated music collaboration. In this frame, two types of interaction coexist: 1) the HCI perspective on interactions between humans and computer, and 2) a musical perspective of interaction among musicians. In our study, the two perspectives are overlapped in a complex ecology of shared agency.

7. CONCLUSIONS

In this paper, we presented an interactive ecosystem in which a live coder and an instrumentalist can cooperate in creating improvised music while sharing agency both between themselves and the machine. By discussing the two musician's perspectives regarding their experience while performing, we highlighted the creative and expressive potential of an ecosystem that pushes musicians towards a collaborative paradigm.

Although a similar interactive ecosystem is not an absolute novelty in the field of computer music and live coding, to the best of our knowledge this practice still remains uncommon and poorly documented. It is our hope that this work will inspire other live coders and other instrumentalists to create hybrid performances centred on interaction and shared musical creation.

8. ACKNOWLEDGMENTS

We would like to thank the TOPLAP Italia community for providing support and suggestions, and Luca Porcelluzzi for technical help in filming and documenting our rehearsals.

9. ETHICAL STANDARDS

This paper complies with the ethical standard of the NIME conference [46]. This work does not present any conflict of interest. The participants are the authors of the paper, as such, they agree on the treatment of their data. The software and libraries used are open-source, the code has been released, and can be freely adapted for anyone to use it, without the explicit need of buying specific hardware or operating complex mods on traditional musical instruments.

10. REFERENCES

[1] J. Aveyard and D. Wilkinson. Third city 2017: Improvisational roles in performances using live sampling. *Open Cultural Studies*, 2(1):562–573, 2018.

[2] M. Bettega, R. Masu, and M. Teli. "it's like a gps community tool": Tactics to foster digital commons through artifact ecology. In *Designing Interactive Systems Conference 2021*, DIS '21, page 1710–1725, New York, NY, USA, 2021. Association for Computing Machinery.

[3] A. Blackwell, A. McLean, J. Noble, and J. Rohrhuber. Collaboration and learning through live coding (dagstuhl seminar 13382). In *Dagstuhl Reports*, volume 3. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2014.

[4] A. F. Blackwell, E. Cocker, G. Cox, A. McLean, and T. Magnusson. *Live coding: a user's manual*. MIT Press, 2022.

[5] S. Bødker and C. N. Klokmose. Dynamics in artifact ecologies. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design*, pages 448–457, 2012.

[6] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[7] I. Bukvic, T. Martin, E. Standley, and M. Matthews. Introducing l2ork: Linux laptop orchestra. In *NIME*, pages 170–173, 2010.

[8] A. Clark and D. Chalmers. The extended mind. *analysis*, 58(1):7–19, 1998.

[9] A. Costall. Socializing affordances, 1995.

[10] A. Cárdenas. Feedforward - piece for electric guitar and autonomous live coding. *Performance recording available at: https://www.youtube.com/watch?v=leZeH7Hx3YQ&t=458s*, 2021.

[11] A. de Campo. 6.7 republic: Collaborative live coding 2003–2013. *Collaboration and learning through live coding*, page 152, 2014.

[12] C. Erdem, K. H. Schia, and A. R. Jensenius. Vrengt: A shared body-machine instrument for music-dance performance. In M. Queiroz and A. X. Sedó, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 186–191, Porto Alegre, Brazil, June 2019. UFRGS.

[13] C. Erdem, K. H. Schia, and A. R. Jensenius. Vrengt: a shared body-machine instrument for music-dance performance. *arXiv preprint arXiv:2010.03779*, 2020.

[14] C. Frauenberger. Entanglement hci the next wave? *ACM Transactions on Computer-Human Interaction (TOCHI)*, 27(1):1–27, 2019.

[15] D. Gorelick. Fantasie no. 0 in c minor for piano and computer. *Performance recording available at: https://www.youtube.com/watch?v=Ru4Ukst8YLo*, 2022.

[16] A. J. H. Goulart and M. D. Antar. Live coding the computer as part of a free improvisation orchestra of acoustic instruments, 2015.

[17] M. Gurevich and J. Treviño. Expression and its discontents: toward an ecology of musical creation. In *Proceedings of the 7th international conference on New interfaces for musical expression*, pages 106–111, 2007.

[18] T. Hall. Pitchcircle3d: A case study in live notation for interactive music performance. 2016.

[19] T. Hoogland. Mercury: a live coding environment focussed on quick expression for composing,

performing and communicating. In *Proceedings of the Fourth International Conference on Live Coding*, pages 353–364, 2019.

[20] S. Jordà, M. Kaltenbrunner, G. Geiger, and R. Bencina. The reactable. In *ICMC*, 2005.

[21] M. Kaltenbrunner, S. Jorda, G. Geiger, and M. Alonso. The reactable*: A collaborative musical instrument. In *15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'06)*, pages 406–411. IEEE, 2006.

[22] B. Latour. *Reassembling the social: An introduction to actor-network-theory*. Oup Oxford, 2007.

[23] S. W. Lee. Show them my screen: Mirroring a laptop screen as an expressive and communicative means in computer music. In *NIME*, pages 443–448, 2019.

[24] S. W. Lee and G. Essl. Communication, control, and state sharing in networked collaborative live coding. *Ann Arbor*, 1001:48109–2121, 2014.

[25] S. W. Lee and G. Essl. Models and opportunities for networked live coding. In *Live Coding and Collaboration Symposium*, volume 1001, pages 48109–2121, 2014.

[26] S. W. Lee and J. Freeman. Real-time music notation in mixed laptop–acoustic ensembles. *Computer Music Journal*, 37(4):24–36, 2013.

[27] P. Lyle, H. Korsgaard, and S. Bødker. What's in an ecology? a review of artifact, communicative, device and information ecologies. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*, NordiCHI '20, New York, NY, USA, 2020. Association for Computing Machinery.

[28] C. Lyons and S. Kettley. Chris lyons livecoding in supercollider with saxophonist steve kettley at click clack. *Performance recording available at: https://www.youtube.com/watch?v=9Ma2T4HiwN8*, 2019.

[29] T. Magnusson. Cmn (code music notation). *Project available at: https://github.com/thormagnusson/cmn.*

[30] T. Magnusson. Of epistemic tools: Musical instruments as cognitive extensions. *Organised Sound*, 14(2):168–176, 2009.

[31] T. Magnusson. Algorithms as scores: Coding live music. *Leonardo Music Journal*, 21:19–23, 2011.

[32] T. Magnusson. *Sonic writing: technologies of material, symbolic, and signal inscriptions.* Bloomsbury Publishing USA, 2019.

[33] T. Magnusson, A. Eldridge, and C. Kiefer. Instrumental investigations at emute lab. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 509–513, 2020.

[34] R. Masu, M. Bettega, N. N. Correia, T. Romão, and F. Morreale. Arcaa: A framework to analyse the artefact ecology in computer music performance. New York, NY, USA, 2019. Association for Computing Machinery.

[35] R. Masu, M. Bettega, N. N. Correia, and T. Romao. Investigating performance ecologies using screen scores: a case study. *accepted for publication at Personal and Ubiquitous Computing*, 2023.

[36] R. Masu, N. N. Correia, S. Jurgens, J. Feitsch, and T. Romão. Designing interactive sonic artefacts for dance performance: An ecological approach. In *Proceedings of the 15th International Conference on Audio Mostly*, AM '20, page 122–129, New York, NY, USA, 2020. Association for Computing Machinery.

[37] R. Masu, N. N. Correia, and T. Romao. Nime scores: a systematic review of how scores have shaped performance ecologies in nime. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Shanghai, China, June 2021.

[38] R. Masu, N. N. Correia, and T. Romão. Technology-mediated musical connections: the ecology of a screen-score performance. In *Proceedings of the 16th International Audio Mostly Conference*, pages 109–116, 2021.

[39] C. McKinney. Quick live coding collaboration in the web browser. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 379–382, London, United Kingdom, June 2014. Goldsmiths, University of London.

[40] A. McLean. Reflections on live coding collaboration. In *Proceedings of the third conference on computation, communication, aesthetics and x*, pages 213–220, 2015.

[41] A. McLean and G. Wiggins. Patterns of movement in live languages. 2009.

[42] A. McLean and G. Wiggins. Tidal–pattern language for the live coding of music. In *Proceedings of the 7th sound and music computing conference*, pages 331–334, 2010.

[43] A. P. Melbye. Resistance, mastery, agency: Improvising with the feedback-actuated augmented bass. *Organised Sound*, 26(1):19–30, 2021.

[44] R. P. Morgan. Stockhausen's writings on music. *The Musical Quarterly*, 61(1):1–16, 1975.

[45] F. Morreale, A. De Angeli, R. Masu, P. Rota, and N. Conci. Collaborative creativity: The music room. *Personal and Ubiquitous Computing*, 18(5):1187–1199, 2014.

[46] F. Morreale, N. Gold, C. Chevalier, and R. Masu. NIME Principles & Code of Practice on Ethical Research, Jan. 2023.

[47] mrreason. Live coding and guitar looping live performance. *Performance recording available at: https://www.youtube.com/watch?v=OwOAOLRbnKQ*, 2021.

[48] C. Neustaedter and P. Sengers. Autobiographical design in hci research: designing and learning through use-it-yourself. In *Proceedings of the Designing Interactive Systems Conference*, pages 514–523, 2012.

[49] D. Ogborn. Live coding in a scalable, participatory laptop orchestra. *Computer Music Journal*, 38(1):17–30, 2014.

[50] D. Ogborn, J. Beverley, L. N. del Angel, E. Tsabary, and A. McLean. Estuary: Browser-based collaborative projectional live coding of musical patterns. In *International Conference on Live Coding (ICLC)*, volume 2017, 2017.

[51] D. Ogborn, E. Tsabary, I. Jarvis, A. Cárdenas, and A. McLean. Extramuros: making music in a browser-based, language-neutral collaborative live coding environment. In *Proceedings of the first international conference on live coding*, pages 163–69, 2015.

[52] D. Ogborn, E. Tsabary, I. Jarvis, A. Cárdenas, and A. McLean. Extramuros: making music in a browser-based, language-neutral collaborative live coding environment. In *Proceedings of the first international conference on live coding*, pages 163–69,

2015.
[53] J. Rohrhuber, A. de Campo, R. Wieser, J.-K. Van Kampen, E. Ho, and H. Hölzl. Purloined letters and distributed persons. In *Music in the Global Village Conference (Budapest)*, 2007.
[54] D. A. Schön. *The reflective practitioner: How professionals think in action.* Routledge, 2017.
[55] J. Simonsen and T. Robertson. *Routledge international handbook of participatory design*, volume 711. Routledge New York, 2013.
[56] O. South. Thoughts on live coding as a session musician. *Project and recordings available at: https://toplap.org/2019/12/11/thoughts-on-live-coding-as-a-session-musician-3-of-3/*, 2019.
[57] P. Stapleton and T. Davis. Ambiguous devices: Improvisation, agency, touch and feedthrough in distributed music performance. *Organised Sound*, 26(1):52–64, 2021.
[58] P. Stapleton, S. Waters, N. Ward, and O. Green. Distributed agency in performance. In *Proc. of International Conference on Live Interfaces*, 2016.
[59] F. J. Varela, E. Thompson, and E. Rosch. *The embodied mind, revised edition: Cognitive science and human experience.* MIT press, 2017.
[60] S. Waters. Performance ecosystems: Ecological approaches to musical interaction. *EMS: Electroacoustic Music Studies Network*, pages 1–20, 2007.
[61] A. Xambó, J. Freeman, B. Magerko, and P. Shah. Challenges and new directions for collaborative live coding in the classroom. In *International Conference of Live Interfaces (ICLI 2016). Brighton, UK*, 2016.