Composing Structured Music Generation Processes with Creative Agents

Jérôme Nika[1] and Jean Bresson[1,2]

[1] STMS Lab, IRCAM-CNRS-Sorbonne Université
[2] Ableton

Abstract. We present a framework for interactive generation of musical sequences combining generative agents and computer-assisted composition tools. The proposed model enables guiding corpus-based generative processes through explicit specifications of temporal structures, or using another layer of generative models trained on a corpus of structures. This paper highlights the foundation and extensions of this approach, details its implementation in the OpenMusic/OM♯ visual programming environments, and presents some applications in musical productions.

1 Structuring music generation with specifications

A branch of research on generative musical systems today aims at developing new creative tools and practice through the composition of high-level abstract specifications and behaviors, as opposed to designing autonomous music-generating agents. A detailed study of related works is beyond the scope of this paper, but interested readers are encouraged to consult for instance (Tatar & Pasquier, 2019; Herremans & Chew, 2019; Wang, Hsu, & Dubnov, 2016; Eigenfeldt, Bown, Brown, & Gifford, 2016; Marchini, Pachet, & Carré, 2017; Joslyn, Zhuang, & Hua, 2018; Louboutin, 2019; Collins & Laney, 2017) for further review.

The DYCI2 project[3] has contributed to this research with models and architectures of generative agents specialized in a wide range of musical situations, from instant reactivity to long-term planning. These models combine machine learning and generative processes with reactive listening modules to propose free, planned as well as reactive approaches to corpus-based generation (Nika, Déguernel, Chemla, Vincent, & Assayag, 2017).

The corresponding generative strategies are all built upon a *musical memory*: a model learned on a segmented and labelled musical sequence providing a graph connecting repeated sub-patterns in the sequence of labels. This graph provides a map of the memory's structure, motives and variations, and can be walked through following different strategies to generate new sequences reproducing its hidden internal logic (Assayag et al., 2006).

A *scenario* was introduced to guide the generation process (Nika, Chemillier, & Assayag, 2017). This symbolic sequence is defined on the same alphabet as the

[3] Collaborative research and development project funded by the French National Research Agency: `http://repmus.ircam.fr/dyci2/`.

labels annotating the memory (*e.g.* chord labels, chunked audio descriptor values, or any user-defined labelled items). As summarized in Figure 1, the concatenative generation process outputs an optimal sequence made up of subsequences of the memory, and whose sequence of labels matches the scenario (Nika, 2016).

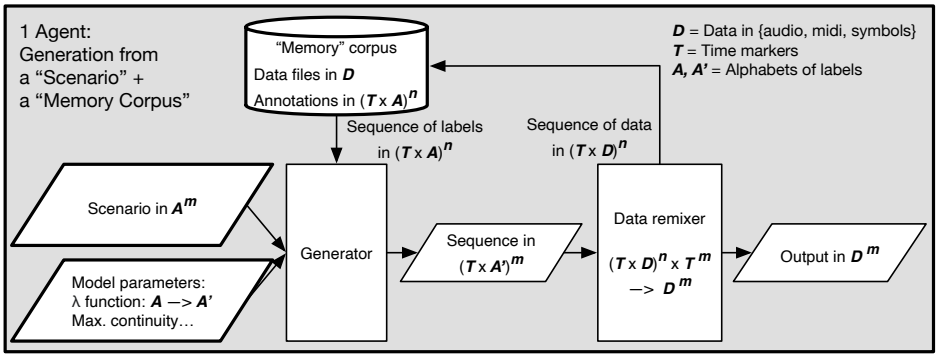

Fig. 1. Architecture of a DYCI2 generative agent.

The Python library implementing DYCI2 models[4] can be addressed through a simple API callable from Python or C bindings, or through OSC messages (Wright, 2005) via UDP. It was interfaced with a library of Max objects dedicated to real-time interaction.[5] In this paper we focus on higher-level compositional applications and present a new client interface implemented as a library for the OpenMusic and OM♯ computer-assisted composition environments (Bresson, Agon, & Assayag, 2011; Bresson, Bouche, Carpentier, Schwarz, & Garcia, 2017). Video examples of applications in artistic productions (see Section 5) are presented in the appendix section (Online references: supporting media).

2 A grounding musical use-case

Lullaby Experience (2019)[6] is an immersive experience imagined by composer Pascal Dusapin, at the crossroads of concert, installation, theater and opera. Surrounded by swarms of singing and whispering voices, the audience evolves within a oneiric scenography to meet a clarinetist, a violinist, a clown, a ballerina... as well as other musicians of Ensemble Modern or other characters imagined by director Claus Guth. The composer needed to generate "singing clouds": musical masses intertwining large numbers of lullabies sung a capella.

[4] DYCI2 Python library: `https://github.com/DYCI2/DYCI2-Python-Library`
[5] DYCI2 Max library: `https://github.com/DYCI2/Dyci2Lib`
[6] *Lullaby Experience* was created in 2019 at the Frankfurter Positionen festival, then presented at the Centquatre in Paris, as part of the Manifeste festival. Computer music design and spatialization: Thierry Coduys. Computer music collaboration: Jérôme Nika. Video report: `https://medias.ircam.fr/embed/media/xb22ae8`.

A database of sung lullabies recordings was collected in a participatory way via a smartphone application. Lullabies sent by participants as audio files from all over the world were to become the *musical memory* of the generative agents. Our technical objective was then to design processes which would navigate through this pool of songs recorded in different languages, with very heterogeneous characteristics and qualities, in order to create these computer-generated "singing clouds" taking the form of polyphonic choirs to be rendered on stage through a spatial sound diffusion system of 64 loudspeakers.

The challenge was therefore to preserve the identity of this heterogeneous memory and to provide strong high-level compositional inputs allowing to specify *scenarios*: the temporal evolutions of these sound masses that had to be sometimes dense, sometimes fragile, static or dynamic, melodic or rhythmic, etc. This led us to rethink the grounds and context of our music generation agents, and to embed them into higher-level computer-assisted composition models.

3 Control of generative agents in OpenMusic/OM#

OpenMusic/OM♯ are visual programming languages providing a graphical interface on Common Lisp, and an extended set of musical structures and editors.[7] The OM-DYCI2 library for OpenMusic/OM♯[8] comprises a C binding to the DYCI2 Python API, and a second-layer binding to Common Lisp (Siguret, 2018). At a higher level, the library allows instantiating and initializing DYCI2 generators as graphical programming objects, parameterizing these generators and querying for generated outputs using scenario specifications.

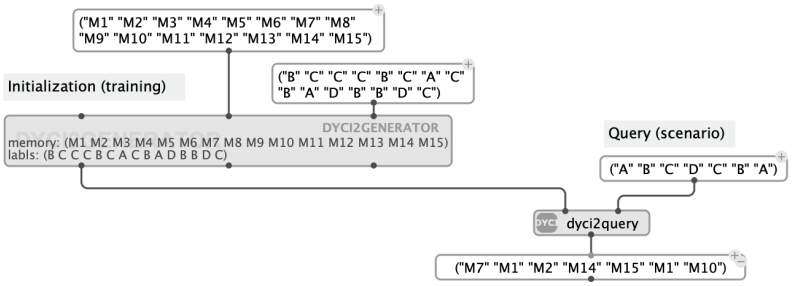

Fig. 2. A simple OM-DYCI2 process in OM♯. Left: Initializing a DYCI2-GENERATOR from a memory defined by a list of contents such as MIDI data, time markers in an audio file, etc. (M1, M2, ...) and a corresponding list of labels (A, B, C, D). Right: Querying the DYCI2-GENERATOR with a scenario. Dataflow is top-down, triggered by evaluation requests at the bottom of the patch.

A DYCI2-GENERATOR is initialized with a simple couple of lists defining the memory of the agent (see Figure 2): a list of contents and a list of labels, deriving

[7] OM♯ is a project derived from OpenMusic. The examples in this paper are in OM♯.
[8] OM-DYCI2 library: `https://github.com/DYCI2/om-dyci2/`

respectively from the segmentation and labelling of the corpus (*Memory corpus* in Figure 1). The contents can be MIDI data, time markers indicating audio segments in an audio corpus, or any other symbolic data depending on the corpus. Other attributes of the generator can be parameterized, such as the *continuity* (maximum length of sub-sequences that can be contiguously retrieved from the memory), or the tolerance to the repetition of the same subsequences.

The *queries* sent to the generator are scenarios defined as simple lists: targeted sequences of labels that the output has to match. The navigation in the model is non-deterministic, which means that two executions of the model with the same parameters will usually be different, while matching the same scenario. Several successive outputs can therefore be used alone, as variations of the same scenario, or as different voices of a same structured polyphony.

Figure 3 shows the complete workflow when using an audio memory to generate new audio sequences. The generator's training data is therefore extracted and formatted from a segmented and annotated audio file. In this example, the contents in the memory are pairs of time markers (in milliseconds), and labels represent corresponding harmonic information. The sequence returned by `dyci2query` is a new sequence of time markers that are used to remix the source material and generate new audio.

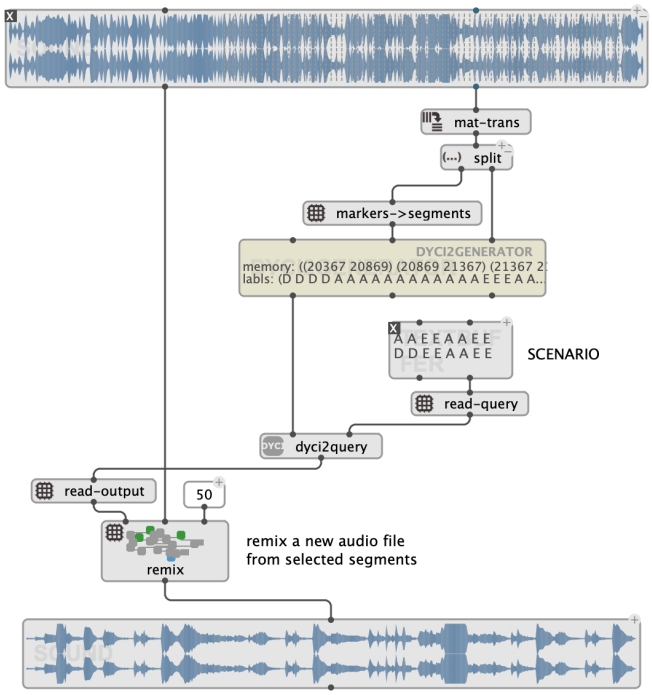

Fig. 3. Generation of audio: `dyci2query` generates a sequence of segments following the given scenario in prepared (segmented/annotated) audio material. The corresponding audio segments in the memory are remixed to produce a new audio file.

The corpus used as memory in this process may be made of larger numbers of audio files, and segmentation and/or labelling data can be stored and loaded as separate files.[9] The computer-assisted composition environment provides tools for MIDI or audio processing, which can be used to prepare such corpora, generate scenarios, or process and transform output sequences (e.g. for remixing, time-stretching, cross-fading, etc. – see Figure 3).

4 Meta-composition: scripting and chaining agents

4.1 Pre-processing labels and using targets as scenarios

The visual programming environment provides a framework to process and transform the labelling information that is used to instantiate the generator's memory and to format the scenarios. If labelling is derived from audio analysis, for instance, it is possible to filter and consider some specific subsets of parameters as drivers for the training and/or generations. Transformation of labels also enables different kinds of normalization of parameter spaces (e.g. reduction of pitch modulo octave, or basic clustering).

Such pre-processing of memory labels (*lambda function* in Figure 1) can have a major impact on the results: changes in the database representation can lead to deep transformations in the temporal model that is learned from it. The generative processes exploit similarities provided by the model, and the cartography of these similarities can radically change depending on which parameters are observed and how they are represented. A discussion about the duality between the accuracy of the memory description and the richness of the temporal model seen from the perspective of the *music information dynamics* (Dubnov, Assayag, & Cont, 2011) is developed in (Surges & Dubnov, 2013), which uses similar models.

The scenario sent as a query to the generator can be a manually defined sequence of labels as seen previously, or be extracted from an other musical material that one would like to use as a target. This process enables to apply the texture of the memory corpus to the structure of the target in an approach similar to *audio mosaicing* (Lazier & Cook, 2003). [10]

4.2 Chaining agents to generate scenarios

In order to increase the level of abstraction in the meta-composition process, a second agent can be introduced beforehand to generate the scenario itself (Figure 4). The training data (memory) of this other agent is a set of sequences of labels, from which new symbolic sequences are produced by "free" generation runs to be used as scenarios to query the first agent. These "free" runs can be compared to Markov random walks tailored for musical contents: they take advantage of the model trained on the symbolic corpus, so that the result be

[9] The DYCI2 library does not provide inbuilt audio anaylsis functionality, and leaves it up to the user or to the host environment to prepare and import their own markers and analysis data.

[10] Online media 1 and 2 (appendix) illustrate this process with the generation of percussion tracks from a voice track, and the generation of several hip-hop tracks.

both new and consistent with the training data (Assayag & Bloch, 2007). The scenarios are then unknown to the user, and the meta-composition paradigm changes radically: the user no longer explicitly defines a temporal evolution, but provides a corpus of sequences that will serve as "inspirations" to articulate the narrative of the generated music.

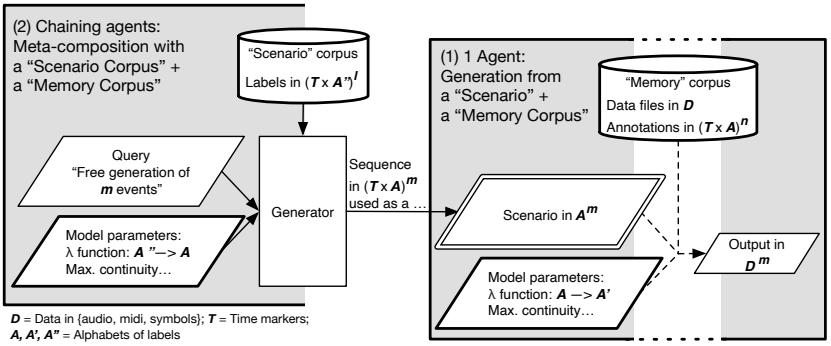

Fig. 4. Navigation through a "Memory Corpus" guided by (1) explicit scenarios manually inputted or (2) scenarios generated by another agent.

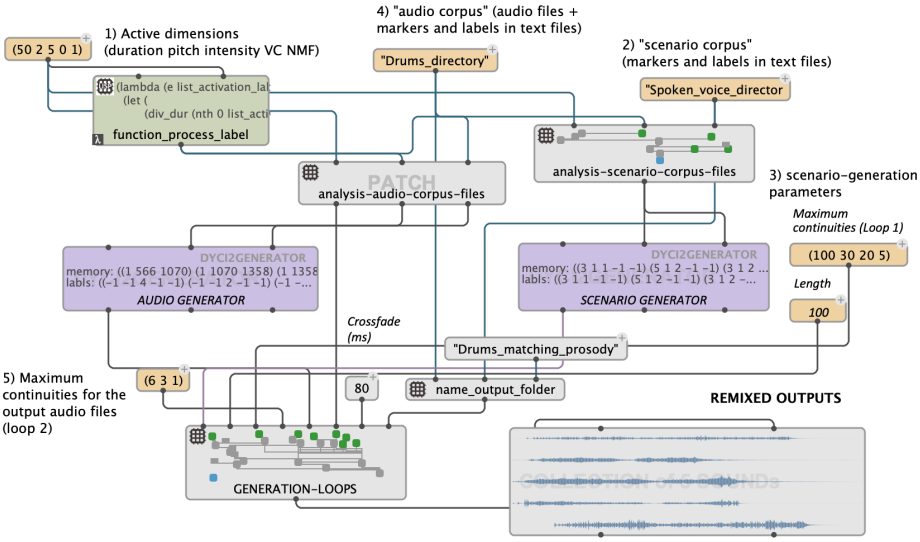

Fig. 5. OM-DYCI2: Chaining agents to generate variations on scenarios and audio outputs matching these scenarios.

On Figure 5, a DYCI2-GENERATOR is trained on a corpus of drums, and uses specific subsets of analysis parameters to instantiate its memory. Several values for the "continuity" parameter are applied in a batch-generation process. On

the other hand, the scenario specification is replaced by a second agent trained on a symbolic corpus built from spoken voice analysis data, accompanied by its own list of maximum continuities, that will also generate as many different scenarios. This example therefore generates several groups of several variations of tracks, driven by variations of the same scenario inspiration.This generation paradigm is the one that best takes advantage of the possibilities of OM-DYCI2. This example and others are further illustrated in the following section by some related artistic productions.

5 Applications

***Lullaby Experience*, by Pascal Dusapin.** In order to implement the *Lullaby Experience* project (see Section 2), the recordings of the database were segmented using tailored onset detection algorithms (Röbel, 2003) and analyzed to obtain labels on 5 dimensions: duration, fundamental frequency (Camacho & Harris, 2008), perceived intensity, voice casting (Obin & Roebel, 2016), and NMF decomposition (Bouvier, Obin, Liuni, & Roebel, 2016). Trained using these analysis data, the agents were thus provided with musical memories embedding a map of the natural and contextual similarities at the syllable level. The agents were therefore able to generate "singing clouds" instantiating melodic, harmonic, rhythmic and/or timbral scenarios defined with various levels of abstraction. Some of these "clouds" were created manually, for example by entering scenarios corresponding to intentions such as "several layers of voices with the same pitch entering successively to make a chord appear". But almost all of them were generated by specifying only "scenario corpora" and "audio corpora" and applying the agent chaining mechanism (see Section 4.2). Depending on the heterogeneity of the chosen corpora the system could generate choirs centered around a melody, or conversely with more complex temporal evolutions – *e.g.* from loops to homorhythmic choirs through textures (see Online media 3).

***Silver Lake Studies*, by Steve Lehman.** Created at the Onassis Cultural Center in Athens in 2019, *Silver Lake Studies* is the first presentation of a long-term collaboration with composer and saxophone player Steve Lehman, that will lead to ambitious productions with French "Orchestre National de Jazz" in 2022. This project built with the DYCI2 library explores the creation of generative processes that can adapt to constantly evolving metrics, the development of real-time spectral enrichment of the saxophone, and "spectral chord changes" as the basis for melodic improvisations. The orchestral masses from the contemporary classical repertoire meet voices from experimental hip-hop (see Online media 4).

C'est pour ça is a dialogue betweeen saxophonist, improviser and composer Rémi Fox and DYCI2 agents. In an upcoming album, laureate of the Dicream grant from the French National Film Board (CNC), the improvised pieces use generative instruments based on the OM-DYCI2 library. Live and tape electronics are generated by a hybrid control from both members of the duo to blur the roles of "acoustic musician" and "electronic musician" (see Online media 5, 6).

C'est pour quoi is a sound installation / listening device produced by Le Fresnoy – Studio National des Arts Contemporain. It is dedicated to the diffusion of pieces resulting from the interaction between human musicians and generative processes. The music results from the interaction between computer agents embedding musical memories and the stimuli provided by musicians, Steve Lehman and Rémi Fox, both mentioned above. The output is broadcasted by a mixed device combining collective listening (speakers, quadraphony) and intimate listening with open headphones that the public can put on or take off to reveal or hide a part of the music (see Online media 7).

***Misurgia Sisitlallan*, by Vir Andres Hera.** This video installation, also produced by Le Fresnoy – Centre National des Arts Contemporains, is inspired from Kircher's "Misurgia Universalis" (1650), a cosmic organ showing the creation of the universe, and from Juana Inès de la Cruz (1689) who materialized, through her poems, the confrontation of cultures. The narrative of the installation mixes anthropological and scientific considerations and travels through the microscopic and macroscopic scales with views of meteorites, lava, and pollen. OM-DYCI2 was used for the sound creation in order to hybridize the languages to generate a halo of voices forming a polyphony sung in Nahuatl, French, Fon, English, Spanish and Haitian Creole (see Online media 8).

***Voi[e,x,s]*, by Marta Gentilucci.** The project *Voi[e,x,s]* is at once a record of a urban site's acoustic heritage, and a series of public performances of live and electro-acoustic compositions by Marta Gentilucci. These pieces use OM-DYCI2 to mix sound field recordings collected before this site's recent redevelopment (creation May 2021).[11]

6 Conclusion

The environment we presented, associating DYCI2 and OM(♯), enables scripting guided music generation processes using layered parameterizable specification levels. This framework makes it possible to easily create large numbers of variations around the same explicit or underlying structures that can be used individually or simultaneously as polyphonic masses.

The compositional practices allowed by this association of generative models and computer-assisted composition could be qualified in a metaphorical way as the generation of musical material composed at the scale of the *narrative*; where the compositional gesture remains fundamental while at a high level of abstraction.

Acknowledgements This work is made with the support of the French National Research Agency and the European Research Council, in the framework of the projects "MERCI: Mixed Musical Reality with Creative Instruments" (ANR-19-CE33-0010) and "REACH: Raising Co-creativity in Cyber-Human Musicianship" (ERC-2019-ADG).

[11] *Voi[e,x,s]* – more information: `https://theatrum-mundi.org/project/voiexs/`

References

Assayag, G., & Bloch, G. (2007). Navigating the oracle: A heuristic approach. In *International computer music conference'07* (pp. 405–412).

Assayag, G., Bloch, G., Chemillier, M., Cont, A., Dubnov, S., et al. (2006). Omax brothers: a dynamic yopology of agents for improvization learning. In *Proceedings of the 1st acm workshop on audio and music computing multimedia* (pp. 125–132).

Bouvier, D., Obin, N., Liuni, M., & Roebel, A. (2016). A source/filter model with adaptive constraints for nmf-based speech separation. In *2016 ieee international conference on acoustics, speech and signal processing (icassp)* (p. 131-135). doi: 10.1109/ICASSP.2016.7471651

Bresson, J., Agon, C., & Assayag, G. (2011). OpenMusic. Visual Programming Environment for Music Composition, Analysis and Research. In *ACM MultiMedia'11 (OpenSource Software Competition).* Scottsdale, USA.

Bresson, J., Bouche, D., Carpentier, T., Schwarz, D., & Garcia, J. (2017). Next-generation Computer-aided Composition Environment: A New Implementation of OpenMusic. In *Proceedings of the International Computer Music Conference (ICMC).* Shanghai, China.

Camacho, A., & Harris, J. G. (2008). A sawtooth waveform inspired pitch estimator for speech and music. *The Journal of the Acoustical Society of America*, *124*(3), 1638–1652.

Collins, T., & Laney, R. (2017). Computer-generated stylistic compositions with long-term repetitive and phrasal structure. *Journal of Creative Music Systems*, *1*(2).

Dubnov, S., Assayag, G., & Cont, A. (2011). Audio oracle analysis of musical information rate. In *2011 ieee fifth international conference on semantic computing* (pp. 567–571).

Eigenfeldt, A., Bown, O., Brown, A. R., & Gifford, T. (2016). Flexible generation of musical form: beyond mere generation. In *Proceedings of the seventh international conference on computational creativity* (pp. 264–271).

Herremans, D., & Chew, E. (2019). MorpheuS: Generating Structured Music with Constrained Patterns and Tension. *IEEE Transactions on Affective Computing*, *10*(4), 510-523. doi: 10.1109/TAFFC.2017.2737984

Joslyn, K., Zhuang, N., & Hua, K. A. (2018). Deep segment hash learning for music generation. *arXiv preprint arXiv:1805.12176*.

Lazier, A., & Cook, P. (2003). Mosievius: Feature driven interactive audio mosaicing. In *Digital audio effects (dafx).*

Louboutin, C. (2019). *Multi-scale and multi-dimensional modelling of music structure using polytopic graphs* (Unpublished doctoral dissertation). Université Rennes 1.

Marchini, M., Pachet, F., & Carré, B. (2017). Rethinking reflexive looper for structured pop music. In *Nime* (pp. 139–144).

Nika, J. (2016). *Guiding human-computer music improvisation: introducing authoring and control with temporal scenarios* (Unpublished doctoral dissertation). Paris 6.

Nika, J., Chemillier, M., & Assayag, G. (2017). Improtek: introducing scenarios into human-computer music improvisation. *Computers in Entertainment (CIE)*, *14*(2), 1–27.

Nika, J., Déguernel, K., Chemla, A., Vincent, E., & Assayag, G. (2017). Dyci2 agents: merging the" free"," reactive", and" scenario-based" music generation paradigms. In *International computer music conference.*

Obin, N., & Roebel, A. (2016). Similarity search of acted voices for automatic voice casting. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *24*(9), 1642-1651. doi: 10.1109/TASLP.2016.2580302

Röbel, A. (2003). A new approach to transient processing in the phase vocoder. In *6th international conference on digital audio effects (dafx)* (pp. 344–349).

Siguret, V. (2018). *Composition musicale assistée par ordinateur : Dyci2 & OpenMusic.* (Ircam – ENS Lyon, Intership report)

Surges, G., & Dubnov, S. (2013). Feature selection and composition using pyoracle. In *Proceedings of the aaai conference on artificial intelligence and interactive digital entertainment* (Vol. 9).

Tatar, K., & Pasquier, P. (2019). Musical agents: A typology and state of the art towards musical metacreation. *Journal of New Music Research*, *48*(1), 56–105.

Wang, C.-I., Hsu, J., & Dubnov, S. (2016). Machine improvisation with variable markov oracle: Toward guided and structured improvisation. *Computers in Entertainment (CIE)*, *14*(3), 1–18.

Wright, M. (2005). Open Sound Control: an enabling technology for musical networking. *Organised Sound*, *10*(3).

Online references: supporting media

1 Generation from a sound target using OM-DYCI2, example #1

Creating a drum track with a credible articulation following the prosody of a spoken voice: `https://www.youtube.com/watch?v=y6ZglbxNJdw`. The system was given drum files as "memory corpus", and an analysis of a spoken voice file as scenario (parameters: loudness with a rather fine definition and pitch classes).

2 Generation from sound targets using OM-DYCI2, examples #2

Generating complex hip-hop tracks using various sound sources as memories activated by target drum patterns as scenarios: `https://www.youtube.com/playlist?list=PL-C_JLZNFAGe8IqgWeW2YRw1kNMlFPj85`.

3 Generating "singing clouds" for *Lullaby Experience*

Examples of stereo reductions of elementary outputs generated for *Lullaby Experience* (Pascal Dusapin) : `https://www.youtube.com/watch?v=4Z7TSMAt8N8`.

4 *Silver Lake Studies* with Steve Lehman

https://www.youtube.com/watch?v=nmSzgpMBDWg. 1min50s - 4min26s: Drum tracks generated from the vocal energy of the rappers of the group Antipop Consortium (being themselves remixed using the chaining mechanism (see Section 4.2); 3min45s - end: pitches of the voices are used to generate orchestral masses; 4min30s - 8min: orchestral masses activated by drums; 10min42s - 12min42: orchestral masses activated by synthesizers.

5 *C'est pour ça* live in Onassis Cultural Center, Athens

https://www.youtube.com/watch?v=mG5sdFxD6qA. In this live performance, the voices accompanied by drums that enter progressively from 8m45s were generated from a patch similar to the one presented in Figure 5.

6 *C'est pour ça* - Playlist of tracks using OM-DYCI2

In the first track, the evolving rhythm is generated by the association of a "scenario corpus" of very simple hip-hop drums loops, and an "audio corpus" of saxophone improvisations using keys sounds only: https://soundcloud.com/jerome-nika/sets/cest-pour-ca-examples/s-lIlLuelEzG9.

7 *C'est pour quoi* - Sound installation at Le Fresnoy

Stereo simulations of *C'est pour quoi*, a sound installation / listening device for music generated with generative agents. https://youtube.com/playlist?list=PL-C_JLZNFAGcjnDw9xKPaTPxtxMmG9Jra. Production: Le Fresnoy - Studio National des Arts Contemporains, first presentation during the "Panorama 22" exhibition, october 2020.

8 *Misurgia Sisitlallan* by Vir Andres Hera

Video teasers and photos of the art installation *Misurgia Sisitlallan* by Vir Andres Hera. https://virandreshera.com/misurgia-sisitlallan/. Production: Le Fresnoy - Studio National des Arts Contemporains, first presentation during the "Panorama 22" exhibition, october 2020.

Fig. 6. Scenographies for *Lullaby Experience* (P. Dusapin); *C'est pour quoi* (J. Nika, S. Lehman, R. Fox); *Misurgia Sisitlallan* (V.A. Hera), (Credits: Quentin Chevrier, Fernando Colin Roque, Vir Andres Hera).