Towards Intelligent Music Production: A Sample-based Approach

Luca Giuliani, Allegra De Filippo and Andrea Borghesi

Department of Computer Science and Engineering, University of Bologna

Abstract

Technological advances have always played a central role in shaping the production of popular music. Over the past few years, music generation systems started to attract considerable interest within the academic community, although the proposed prototypes rarely managed to emerge and be adopted by producers in their professional workflows. We argue that a major cause of that is the inherent complexity of integrating those systems into well-established music production pipelines, especially given that most of them are designed with the intent of replacing human creativity rather than assisting it. To this end, we discuss our proposal for a novel approach for Intelligent Music Production based on samples arrangement. Such a tools could offer several potential benefits in enhancing human creativity, as they provide the opportunity to keep human artists in the creative loop as well as to reduce computational costs and hardware requirements, making music production more accessible. As a first step towards this direction, we eventually present MusiComb, a prototype for sample-based music generation. Alongside, we report how this relatively simple system has demonstrated its ability to produce realistic tracks in few seconds while adhering to user-defined constraints.

Keywords

Intelligent Music Production, Music Generation, Generative AI, Human-in-the-Loop

1. Introduction

The creation of musical artifacts has been always intertwined with the trajectory of technological evolution. Much like how the invention of the electric guitar gave rise to a whole set of new musical genres, the development of novel electronic and digital assets has consistently shaped the musical landscape. On top of that, a growing body of evidence suggests that we are entering a new phase of musical creativity where composers and producers could benefit from the innovative opportunities presented by novel Artificial Intelligence (AI) systems. This approach to music creation, which has been referred to as Intelligent Music Production (IMP) [1], might represent a strong paradigm shift where AI technologies are adopted to empower the creative capabilities of music producers towards new era of innovation and artistic expression.

In recent years, the remarkable effectiveness of AI led to its spread across several artistic domains, with the most prominent examples being synthetic image [2] and natural language generation [3, 4]. Nonetheless, in the audio realm only a limited number of AI-based products have achieved industrial levels of advancement [1], although research on the subject has been

CREAI 2023 – Workshop on Artificial Intelligence and Creativity, Nov.06 – Dec.09, 2023, Rome, Italy
luca.giuliani13@unibo.it (L. Giuliani)
0000-0002-9120-6949 (L. Giuliani); 0000-0002-1954-7271 (A. De Filippo); 0000-0002-2298-2944 (A. Borghesi)

gaining significant traction within the academic community [5]. In fact, researchers in the field are currently exploring various approaches to delve into the domain of synthetic music generation, some of which encompass the creation of entire tracks from scratch, while some other could be seen as collaborative AI companions aimed to assist human artists rather than superseding them [6]. Specifically, the prevailing trend in synthetic music generation have notably emphasized sub-symbolic methodologies, with the employment of large Deep Learning (DL) models in a wide series of tasks ranging from single- to multi-track composition, and dealing with either symbolic music notation or raw audio format [7, 8]. Conversely, prior to the emergence of those models in the early 2010s, a substantial portion of generators relied on symbolic and rule-based approaches, whose effectiveness was strongly related to the well-studied structured nature of music composition [9]. Nevertheless, such a widespread adoption of deep learning models comes with several drawbacks, including a limited degree of user control, a lack of comprehensive global structural coherence, and the inherent challenge of real-time generation due to their high computational demands and specialized hardware prerequisites [10].

In this paper, we start by examining the role of artificial intelligence in the current music production environment. Our purpose is to pinpoint the limitations that have hindered academic research on the subject to reach the industrial field. Next, we discuss our proposal for the development of more seamlessly integrated tools for Intelligent Music Production, which could serve the purpose of being integrated more easily into already established composition and production workflows. To this end, we present *MusiComb*, a music generation system that we previously introduced in [11]. MusiComb is designed to craft new musical tracks by properly arranging a set of samples under user-defined constraints, and whose empirical evaluation showed that promising outcomes can be obtained at a very low computational cost. This brings about several advantages, including reduced execution time and, decreased hardware requirements, and perhaps most significantly a closer alignment with modern music production pipelines, which would allow practitioner to more easily integrate such systems in their workflows as well as offering them the agency to intervene in the final composition once it has been generated.

The rest of the paper is structured as follows. Section 2 provides an overview of modern music production, highlighting the motivations and advantages of incorporating a sample-based music generation system into the creative process. In Section 3, we review the current state of the art for music generation systems, with a major focus on those designed to handle rules and constraints as well as considering the active participation of a human rather than attempting to replace them. Section 4 describes the architecture of MusiComb and presents some results of experimental works conducted with it. Finally, in Section 5 we conclude our discussion and examine some potential future directions.

2. Music Production Background

Contemporary pop and electronic music production has been extensively influenced by the adoption of digital devices. It was mainly due to the development of Digital Audio Workstations (DAWs) that computers have been rapidly transformed as unavoidable tools for music producers in the past twenty years [12]. DAWs are software used to manage the (digital) music production pipeline. They empower music producers to record, manipulate, and blend together multiple musical tracks, ultimately culminating in the creation of a unified audio waveform. Alongside that, the workflow of music industry professionals has gradually shifted towards a massive use of *sample libraries* [13], namely large pools of pre-recorded music fragments which are manually imported in the DAW and lately processed, overlaid, and eventually arranged throughout time.

Avdeef [14] marks a clear distinction between the traditional approach to music composition and the contemporary methodology employed in the production of modern pop songs. In fact, classical composers often lean on well-defined melodic, harmonic, and rhythmic patterns typically applied at a local scale; on the contrary, modern music producers make an extensive use of pre-recorded samples, which are carefully curated and arranged to construct the final musical composition. Rodgers [15] traces the origin of sampling back to the tradition of *musique concrète*, a particular style of contemporary music that emerged in the mid-twentieth century. Additionally, the author defines sampling as "a postmodern process of musical appropriation and pastiche", and reports how the process of gathering and manipulating samples is one of the most most central – and thus, time-consuming – steps of modern music production. Under this lens, music production aligns with the concept of "novel linkage" introduced by Carnovalini [16], namely the creation of novel material starting from something that already existed.

It is evident that the music industry has not been profoundly influenced by artificial intelligence as its visual and textual counterparts [17], and especially that very few works have tackled algorithmic composition through sample-based approaches yet. Notable exceptions include the works of Anderson [18] and Aucouturier [19], but the vast majority of research papers are mainly focused on generating music from scratch – either in symbolic notation or raw audio format – without taking into account neither rules nor human intervention [5]. Nonetheless, we believe that a framework designed to directly work at sample level could offer music producers several advantages, such as: (1) the possibility to edit the generated output, as it results from the process of samples concatenation onto a two-dimensional grid; (2) the inherent flexibility of the system, as it is virtually able to handle any music genre provided that samples are drawn from a sufficiently large pool of matching ones; and (3) the similarity of this methodology to the most common pipelines adopted by professionals in the field.

On a final note, it is worth noting that an Intelligent Music Production system operating at sample level could also offer additional benefits that extend beyond the creative process. These advantages encompass both ethical considerations related to the perceived ownership of the artists, as well as practical aspects concerning trustworthiness and real-time inference. Indeed, by dealing with samples rather than individual notes – or, even worse, raw audio processing –, the computational workload imposed on the machine is significantly reduced when compared to current end-to-end neural-based models. This efficiency would not only lead to a faster and more cost-effective inference, with benefits on the widespread of the technology and its potential application in a live setting, but also opens the door for more intensive pipelines where several traditional or AI-based audio processing units are chained in order to reach an innovative range of creative options for composers and producers. For a more in-depth discussion of these aspects, we refer the reader to our original contribution [11].

3. Related Works

The long tradition of computer-aided music generation stems from a series of sketches on the Analytical Engine by Ada Lovelace, suggesting that machines could generate musical tracks of any degree of complexity and extent provided that the fundamental relations between sounds were correctly encoded [20]. Clearly, such a stringent logical and mathematical approach may seem to restrictive for contemporary music genres, but nonetheless it better resonates with the mindset of classically trained composers [21]. Hence, it comes as no surprise that many pioneering works in the realm of algorithmic composition such as the "Iliac Suite" [22] and CHORAL [23] primarily harnessed rule-based symbolic frameworks. The explosion of large deep learning models in the last decade has pushed forward the state of the art, with particular regards to the generation of Bach chorales by systems like BachBot [24] and DeepBach [25], nevertheless the stylistic specificities of these systems make them unemployable in modern music production pipelines, albeit very interesting from a computational perspective.

Despite the fast pace at which research on the topic is progressing, both by [1] and [5] report that very few AI-based applications are at disposal for practitioners. Among the main reasons, one is the inherent lack of support for human intervention in neural-based end-to-end models. In particular, those generating raw audio are still prone to introducing noticeable artifacts that systematically impede their suitability for professional environments. Moreover, most of the systems are mainly designed to replace human creativity rather than assisting it, and even those like Generative Audio Workstations (GAWs) who try to include generative AI within music production workflows, are usually offered as standalone services, thus preventing their direct embedding into preexisting software. Among the exceptions, LambDAW [26] is a novel GAW developed on top of the commercial workstation Reaper, which was designed with the main purpose of allowing seamless integration of programming operations within the workstation itself. As regards instead other tools for composition and execution, we mention systems like GEDMAS [18] and Musical Mosaicing [12, 19, 27], as well as other kinds of co-creative applications such as Reflexive Looper [28] and Flow Machines [29].

Finally, [30] proposes an innovative framework consisting of two sequential steps. At first, a sub-symbolic model is trained to create short samples with appropriate musical metadata; then, the generated samples serve as building blocks for the subsequent stage, where they are combined together in order to create the final track. The authors also contribuite by releasing a public dataset for the task, which they further utilize to develop the first phase of the framework. Starting from their work, we further extended it in [11] in order to tackle the second step using both the machine- and the human-generated samples that were already present in the dataset.

4. MusiComb

MusiComb [11] is an AI-based music generation system designed to solve the task of *combinatorial music generation* (ComMU), which was first proposed and tackled in [30]. As depicted in Figure 1, the pipeline of MusiComb is structured around three main steps, i.e.:

1. Users choose the shared metadata of samples, i.e. genre, time, progression, etc...
2. A subset of matching samples is either queried from a database or generated by an AI.

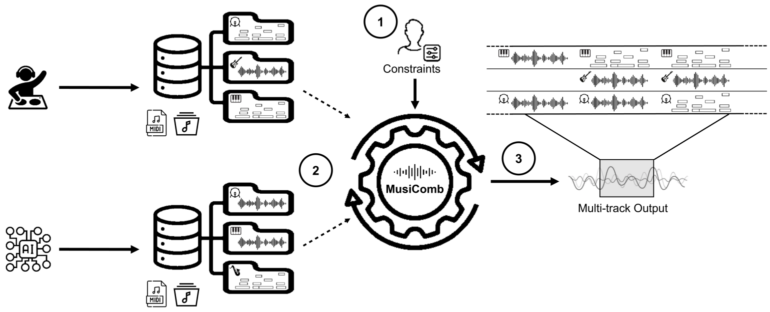

Figure 1: The three steps of our music generation pipeline: (1) the user selects genre, bpm, and other properties of the output track; (2) this metadata is used to query a set of matching samples which can be either in musical notation or raw audio, and either human- or machine-generated; and (3) the retrieved samples are combined together via a constraint program and returned as a multi-track output.

3. The retrieved samples are arranged together using a Constraint Programming approach.

The metadata selected by the user in step (1) is used to to query the subset of matching samples in step (2). Since during the querying process the focus remains solely on the metadata attributes, with no consideration for the internal structure of the samples, they can be either in symbolic music notation or in raw audio format; likewise, they can be either human- or machine-generated. Finally, once the pool is correctly retrieved, we model step (3) as a *job-shop problem* [31] in order to obtain the final output.

In the constraint program, each sample represents a task while machines are represented by *track roles*. Track roles are part of the available metadata in the ComMU dataset, but different roles could be potentially adopted depending on their availability. Instead of posing a limitation to our approach, this highlights its inherent ability to handle different genres as well as different sample libraries at the minimum cost of minor adjustments in the model specification. Finally, in order to obtain a solution, we attach a *demand* value to each of the track roles as follows:

$$\text{demand(sample)} = \begin{cases} \texttt{\#main} & \text{if role(sample)} \in \{\text{main melody, riff}\} \\ \texttt{\#side} & \text{if role(sample)} \in \{\text{sub melody, accompaniment}\} \\ \texttt{\#back} & \text{if role(sample)} \in \{\text{bass, pad}\} \end{cases}$$

These values represents the “cost” of each track role and are intended to measure the “importance” of each sample; then, by defining a total *capacity* of the model, we can adopt these values to pose a constraint on the number of tracks that are allowed to play together. Eventually, the solution of the job-shop problem is obtained by minimizing the total time of the track, and it is returned as a series of samples positions inside a two-dimensional grid (see Figure 2). Although we are not interested in the track being as short as possible, the minimization of the total time allows us to discard degenerate solutions consisting in samples arranged sequentially. For additional details on the proposed pipeline, we refer the reader to the original MusiComb paper [11].

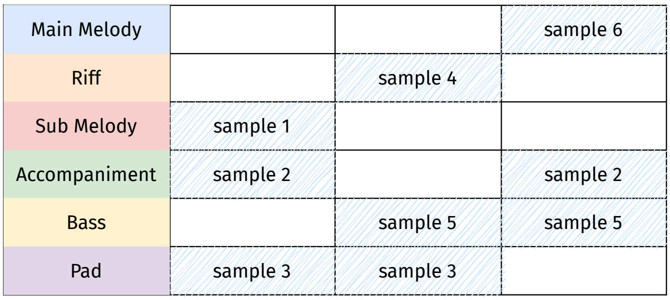

Figure 2: We arrange music samples in a bidimensional grid where the horizontal axis represents time while the vertical one represents their track role.

4.1. Results

In order to assess the capabilities of our approach, we adopted MusiComb to generate music tracks either by querying samples in the ComMU dataset ("ComMU") or by directly generating them using the pre-trained transformer model proposed in [30] ("Generated"). During the course of our experiments, we fixed the total capacity to 6 and adopted the following values for demands: `#main` = 3, `#side` = 2, and `#back` = 1. These values were selected after a preliminary evaluation, and follow the known musical rationale according to which more prominent track roles should be paired with more background-like ones, although we recall that their definition is a custom design choice which can change the outputs of the model.

Table 1 reports the metadata used in each test as well as the execution times needed for samples retrieval and for the solution of the CP problem. Since the ComMU dataset comprises "New Age" and "Cinematic" samples only, we generated tracks according to these two genres while choosing the other metadata from the available ones. The obtained tracks – in symbolic notation format – were then converted to audio leveraging the GarageBand software[1] and can be listened to at the following link: https://soundcloud.com/musicomb.

The generated outputs prove how MusiComb succeeds at replicating the style of the original samples and adhering to the accompanying metadata. his outcome stems from a deliberate design choice when modeling the task, where the primary role of the machine is confined to the skillful arrangement of samples. Furthermore, we also reckon how the low computational demands required by MusiComb could allow for its adoption in live settings, since the generation of the output tracks is fast enough to be potentially used for real-time inference. This capability can be particularly advantageous when the samples are sourced from the dataset, albeit with a potential trade-off in compositional creativity compared to utilizing the neural generative model for sample generation.

5. Conclusion

We presented a position paper on the role of artificial intelligence within the context of modern music production. We began by highlighting a prevailing trend in Intelligent Music Production,

[1] https://www.apple.com/mac/garageband/, version 10.4.8

Test	BPM	Progression	Genre	Key	Measures	Execution Time ComMU	Execution Time Generated
1	130	Am-F-C-G-Am-F-C-G	New Age	Am	8	3s	14s
2	80	Am-Gmaj7-Fmaj7-G-Cmaj7-Dm7-Am-A#maj7-E+-Am	Cinematic	Am	8	3s	15s
3	120	C-F-Am-G	Cinematic	Cmaj	8	3s	16s
4	100	F-G-Em-Am-F-G-Em-Am	New Age	Cmaj	4	4s	13s

Table 1
Metadata used for each generation test and respective execution times.

which tends to prioritize the development of tools aimed to supplant human creativity rather than complement and enhance it. This approach has posed significant barriers to the adoption of AI technologies by music professionals, and we believe that a viable solution lies in the development of AI-based systems designed to handle music in the same way as modern producers do.

Even though some exceptions exist, these are either in a prototype state or are not designed in a way that they could be easily embedded in well-established music production software. For this reason, we claimed that the development of sample-based approaches for music generation systems would bring several benefits to the field. These benefits encompass, but are not limited to, the inherent guarantees on human intervention, the ability of the system to be genre-agnostic given that a sufficiently large pool of matching samples is provided, and most of all the possibility to integrate such systems within already established production enviroments and pipelines such as DAWs and sample libraries.

Eventually, we showed the feasibility of the proposed approach by presenting MusiComb, a sample-based music generation system which was previously introduced in [11]. MusiComb is designed to craft new musical tracks by retrieving and arranging a set of given samples which match a series of user-defined metadata, and experimental evaluations proved its capacility to return promising results within a small amount of time and using readily accessible, cost-effective hardware resources. We believe this to be a strong hint that the development of AI models capable of working with samples, rather than generating individual notes or directly manipulating raw audio, holds substantial potential to bring notable advancements to the field of Intelligent Music Production, both from the researchers' and the professionals' sides.

In conclusion, we acknowledge that MusiComb is in its early stages and that there are still numerous features to be developed. Nonetheless, would our experiments continue to validate the efficacy of sample-based approaches in music generation, we aspire to develop MusiComb as a software to be integrated within established music production tools like Digital Audio Workstations (DAWs) in order to conduct comprehensive qualitative and quantitative analyses on user satisfaction and human-machine interaction among a community of music practitioners.

5.1. Future Works

That of MusiComb is just a preliminary study that, albeit interesting, could not be effectively integrated within a music production workflow yet. Additional steps need to be done to bring it to a professional level and to test its capabilities in a real production environment.

Among all, in our forthcoming research endeavors we would like to explore how the system would reach to both a diverse pool of samples and an alternative neural generative model beyond Hyun et al.'s Transformer. Similarly, we aim to explore the possibility of combining more samples together on the *capacity* dimension, in order to allow us to study the scalability of the approach, considering that an increased number of tracks corresponds to a more complex Constraint Programming (CP) model, potentially entailing higher computational demands.

As observed, MusiComb's real-time capabilities were validated by experimental results, although the need of a relatively high-end GPU is essential when it comes to neural generation rather than sample retrieval, However, it is important to note that the sample generation phase is not mandatory in our approach, as samples can theoretically be obtained from existing datasets, hence this phase could be executed offline without real-time constraints, reducing the need of top-tier computational resources. Finally, as regards live setting scenarios, the rigidity of CP models seems to be prohibitive, as tracks are generated once and for all rather then in subsequent steps. For this reason, one of our future direction involves the development of a new type of arragement problem which deals with samples in a concatenative way, so that a virtually infinite and seamless flow of samples could allow musicians to perform and improvise along with the system. Again, especially in this scenario, fast inference is to be considered as a strong priority, hence extensive tests on the model scalability must be performed.

In conclusion, if our further experiments validate the potential of sample-based approaches for music generation, we aim to develop MusiComb as a standalone application, in order to facilitate the use of such system within professional settings. Integration within well-established music production software such as Digital Audio Workstations would enable us to conduct comprehensive qualitative and quantitative analyses on user satisfaction and human-machine interaction among a community of music practitioners. This step would mark a significant leap towards realizing the practical utility and impact of our approach in the field of music composition and production.

Acknowledgments

This work has been supported by the project TAILOR (funded by European Union's Horizon 2020 research and innovation programme, GA No. 952215)[2].

References

[1] D. Moffat, M. B. Sandler, Approaches in intelligent music production, Arts 8 (2019) 125. URL: https://doi.org/10.3390%2Farts8040125. doi:10.3390/arts8040125.

[2] J. Oppenlaender, The creativity of text-to-image generation, in: Proceedings of the 25th International Academic Mindtrek Conference, 2022, pp. 192–202.

[3] G. Franceschelli, M. Musolesi, On the creativity of large language models, arXiv preprint arXiv:2304.00008 (2023).

[4] I. van Heerden, A. Bas, Ai as author–bridging the gap between machine learning and literary theory, Journal of Artificial Intelligence Research 71 (2021) 175–189.

[2]Discaimer: This paper reflects only the authors' views. The European Commission is not responsible for any use that may be made of the information it contains.

[5] M. Civit, J. Civit-Masot, F. Cuadrado, M. J. Escalona, A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends, Expert Systems with Applications 209 (2022) 118190. URL: https://doi.org/10.1016%2Fj.eswa.2022.118190. doi:10.1016/j.eswa.2022.118190.
[6] H. Zulić, et al., How ai can change/improve/influence music composition, performance and education: three case studies, INSAM Journal of Contemporary Music, Art and Technology 1 (2019) 100–114.
[7] J.-P. Briot, From artificial neural networks to deep learning for music generation: history, concepts and trends, Neural Computing and Applications 33 (2021) 39–65.
[8] J.-P. Briot, G. Hadjeres, F.-D. Pachet, Deep learning techniques for music generation, volume 1, Springer, 2020.
[9] O. Laske, Composition theory: An enrichment of music theory, Journal of New Music Research 18 (1989) 45–59.
[10] S. Dadman, B. A. Bremdal, B. Bang, R. Dalmo, Toward interactive music generation: A position paper, IEEE Access 10 (2022) 125679–125695.
[11] L. Giuliani, F. Ballerini, A. De Filippo, A. Borghesi, Musicomb: a sample-based approach to music generation through constraints, in: 2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, 2023.
[12] A. Zils, F. Pachet, Musical mosaicing, in: Digital Audio Effects (DAFx), volume 2, 2001, p. 135.
[13] C. Nardi, Library music: technology, copyright and authorship, Current Issues in Music Research: Copyright, Power and Transnational Musical Processes. Lisboa: Edições Colibri (2012) 73–83.
[14] M. Avdeeff, Artificial intelligence & popular music: SKYGGE, flow machines, and the audio uncanny valley, Arts 8 (2019) 130. URL: https://doi.org/10.3390%2Farts8040130. doi:10.3390/arts8040130.
[15] T. Rodgers, On the process and aesthetics of sampling in electronic music production*, in: Electronica, Dance and Club Music, Routledge, 2017, pp. 89–96. URL: https://doi.org/10.4324%2F9781315094588-6. doi:10.4324/9781315094588-6.
[16] F. Carnovalini, A. Rodà, Computational creativity and music generation systems: An introduction to the state of the art, Frontiers in Artificial Intelligence 3 (2020). URL: https://doi.org/10.3389%2Ffrai.2020.00014. doi:10.3389/frai.2020.00014.
[17] G. Bromham, How can academic practice inform mix-craft, Mixing music, Perspective on Music Production (2016) 245–256.
[18] C. Anderson, A. Eigenfeldt, P. Pasquier, The generative electronic dance music algorithmic system (GEDMAS), Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment 9 (2021) 5–8. URL: https://doi.org/10.1609%2Faiide.v9i5.12649. doi:10.1609/aiide.v9i5.12649.
[19] J.-J. Aucouturier, F. Pachet, Jamming with plunderphonics: Interactive concatenative synthesis of music, Journal of New Music Research 35 (2006) 35–50. URL: https://doi.org/10.1080%2F09298210600696790. doi:10.1080/09298210600696790.
[20] L. F. Menabrea, A. K. C. of Lovelace, Sketch of the Analytical Engine Invented by Charles Babbage, Esq, Richard and John E. Taylor, 1843.
[21] T. Anders, Compositions Created with Constraint Programming, Oxford University

Press, 2018. URL: https://doi.org/10.1093%2Foxfordhb%2F9780190226992.013.5. doi:10.1093/oxfordhb/9780190226992.013.5.

[22] l. a. hiller, jr., l. m. isaacson, musical composition with a high-speed digital computer, journal of the audio engineering society 6 (1958) 154–160.

[23] K. Ebcioğlu, An expert system for harmonizing four-part chorales, Computer Music Journal 12 (1988) 43. URL: https://doi.org/10.2307%2F3680335. doi:10.2307/3680335.

[24] F. T. Liang, M. Gotham, M. Johnson, J. Shotton, Automatic stylistic composition of bach chorales with deep lstm, in: International Society for Music Information Retrieval Conference, 2017.

[25] G. Hadjeres, F. Pachet, F. Nielsen, Deepbach: a steerable model for bach chorales generation, in: International Conference on Machine Learning, PMLR, 2017, pp. 1362–1371.

[26] LambDAW: Towards a Generative Audio Workstation, Zenodo, 2023. URL: https://doi.org/10.5281/zenodo.7842002. doi:10.5281/zenodo.7842002.

[27] J.-J. Aucouturier, F. Pachet, Ringomatic: A real-time interactive drummer using constraint-satisfaction and drum sound descriptors., in: ISMIR, 2005, pp. 412–419.

[28] M. Marchini, F. Pachet, B. Carré, Rethinking reflexive looper for structured pop music., in: NIME, 2017, pp. 139–144.

[29] F. Pachet, P. Roy, B. Carré, Assisted music creation with flow machines: towards new categories of new, Handbook of Artificial Intelligence for Music: Foundations, Advanced Approaches, and Developments for Creativity (2021) 485–520.

[30] L. Hyun, T. Kim, H. Kang, M. Ki, H. Hwang, K. Park, S. Han, S. J. Kim, ComMU: Dataset for combinatorial music generation, in: Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2022.

[31] J. Błażewicz, W. Domschke, E. Pesch, The job shop scheduling problem: Conventional and new solution techniques, European Journal of Operational Research 93 (1996) 1–33. URL: https://doi.org/10.1016%2F0377-2217%2895%2900362-2. doi:10.1016/0377-2217(95)00362-2.