Comments on "Assisted lead sheet composition using FlowComposer"

François Pachet[1], Alexandre Papadopoulos[1], and Pierre Roy[1]

CTRL, Spotify francois@spotify.com

Abstract. We comment on the motivation of this paper, presented in the Application track at CP 2016. We argue that solving real world, ambitious problems is an effective way to identify new, interesting concepts, possibly basic ones. We take the example of this paper to challenge the traditional separation between theoretical and applied research in computer science.

Keywords: Markov chains · Applications · Music · global constraints

1 From Claude Benard to Leslie Lamport

The main reason we have been focusing, for a great number of years, on so-called industrial problems is because we are convinced that working on such problems is the key to identifying basic, sometimes fundamental problems. This approach is not new and can be seen as an instance of the scientific method of observation first advocated by Claude Bernard in [6]: It is only by making experiments and observing the real world that we can find hidden truths about it. In the case of Claude Bernard, his daily observations of rabbits in many situations triggered his curiosity until he discovered animal glycogenesis, among many other important phenomena.

In computer science, however, the distinction between basic and applied research remains tenaciously strong, in spite of several attempts by famous researchers to challenge it. For instance, Turing award winner Leslie Lamport [11] wrote:

> I find it remarkable that, 20 years after Dijkstra first posed the mutual exclusion problem, no one had thought of trying to find solutions that were fast in the absence of contention. This illustrates why I like working in industry: the most interesting theoretical problems come from implementing real systems.

By chance or by choice, our work is deeply rooted in these views, and we want to comment our position by taking the paper "Assisted lead sheet composition using FlowComposer", presented at CP 2016 in Toulouse (France), as a representative example.

2 The *NValue* constraint

The origin of this paper can be traced back to a project we (first and last co-authors of this paper) conducted in 1999 at Sony Computer Science Laboratory to generate playlists automatically. At that time, the word "playlist" was not commonplace, so we talked about "music programs" [18].

The intuition we had was not completely clear, but we believed there was something deep in the idea that good playlists, i.e. playlists that would keep a listener's attention, should satisfy various contradictory criteria, dealing both with continuity and variety. In order to make experiments, we needed to implement an engine and constraint satisfaction seemed the most natural way to go. However, a number of properties on sequences we wanted to express could not be formulated by global constraints available at the time. In particular, we needed:

1. A way to express cardinality on *values* of attributes. This is how the *NValue* constraint was introduced in [18]. We introduced the definition and proposed basic filtering mechanisms for it. The constraint was subsequently studied by several researchers in the CP community, who proposed complete filtering methods [7], decompositions [8]. 20 years later, *NValue* is still a constraint studied in the community (see e.g. recent papers such as [10]).
2. A way to express "continuity of timbre between successive songs". This is how research on timbre similarity started, with then Ph.D. student Jean-Julien Aucouturier. This study led to Jean-Julien's thesis [2]. Along the way, an interesting result was found by observing that some songs behaved as similarity "hubs" [1, 3]. The same story happened here: the notion of hubs was subsequently studied by the Music Information Retrieval community, and then the machine-learning community, and many explanations have been provided since about the origins of hubs, and ways to circumvent them [26, 9].

In both cases, the *NValue* constraint and "hubness", we had identified relevant new concepts by looking at novel applications (playlists and audio similarity). We were not interested so much in studying the concepts themselves, and fortunately the research community did it, probably far better than what we would have done ourselves. We are, however, proud of these "discoveries" and the way they were handled by the respective research communities. In a metaphoric sense, these new concepts were found following the empirical method of Claude Bernard, *stricto sensu*: create objects of interest on which we can graft intuitions because of our expertise in the domain (here, music), observe and manipulate them, make hypothesis, and repeat until everything is clear.

3 The roots of Flow Composer

The paper "Assisted lead sheet composition using FlowComposer" finds its origin in a similar context to the preceding examples. This work originated in the

desire to build interactive music systems that could "learn" from examples. The Continuator project [17] had successfully shown that it was possible to learn musical streams in real time, coming from a digital (MIDI) synthesizer. Many studies were performed with the Continuator (in the classroom in particular [27]).

The next natural step was to apply similar ideas to musical composition. However, going from unstructured improvisation to compositions required the algorithms to cope with various constraints. This is how the Flow Machines project originated (funded by an ERC advanced grant): the quest to design machine learning algorithms for music and text, that could satisfy all sorts of constraints, imposed either by the genre or the user. Flow Composer, the system described in the paper presented at CP, is one result of this project. The basic idea was to reformulate Markov chains as constraint satisfaction problems, so that additional constraints could be expressed naturally without modifying the core algorithm (the main claimed advantage of constraint programming as opposed to traditional problem solving).

It took several steps to come up with a working system that could be used by a musician: first we got results about formulating Markov chains as constraints [15], adding positional constraints to Markov chains [16] and adding meter to Markov chains [28]. We later realised that our algorithm was in fact similar to belief propagation [23], and that meter was regular, and produced a novel implementation of metrical Markov sequences [21]. Many other results were obtained in this vein (i.e. adding global constraints to a Markov chain), such as $1/f$ [19] or controlling the maximum order [22]. This line of works generated a substantial amount of theoretical and algorithmic results [20], as well as fruitful collaborations with other CP researchers. Jean-Charles Régin and Guillaume Perez, in particular, reformulated a number of our algorithms in the framework of Multi-valued Decision Diagrams (MDD), yielding substantial gains in efficiency [25, 24].

In order to use these generators, we designed the interface Flow Composer described in the paper. The basic idea was to let users enter arbitrary chunks of songs (melody or chords) and let the generator fill in the blanks, a process referred to now as *inpainting*. This process took several iterations, ranging from the development of a javascript library for music web editing [12] to studies about the impact of feedback on composition with these tools [13]. Thanks to this interface, Flow Composer was intensively used by musicians, in particular by SKYGGE (real name Benoit Carré)[1] to compose and produce what turned out to be the first mainstream music album composed with AI: Hello World[2]. The project is now continuing at Sony[3], where other interactive editors have been proposed in the same vein [5]. Surely, many more will come and music inpainting will become commonplace.

[1] https://open.spotify.com/artist/4aGSoPUP1v4qh7RfBlvgbR/about
[2] https://www.helloworldalbum.net/
[3] https://www.sonycsl.co.jp/tokyo/2811/

4 After the paper

Three years after the publication of this paper, what remains the most important for us in these developments is not the interface we built *per se* and its followers, but the fact that we showed that such editors could be used productively to compose high quality, original music. In that respect, the utmost validation of this work lies in the following:

1. The overwhelming media reception of the album composed with it[4], including some outstanding reviews[5]
2. The enthusiasm of all the musicians who participated in this projet. 15 songs were composed and signed by various artists (including Stromae), and this is in itself a validation, as musicians would not sign songs they are not proud of
3. The overall success of the album on streaming platforms (a total of 11 million streams as of October 2019), with Magic Man (5.5 million streams) and Hello Shadow (2.7 million) as main hits
4. The positive reception by music critics. For instance, Avdeef writes[4]:
   > SKYGGE's Hello World is a product of these new forms of production and consumption, and functions as a pivot moment in the understanding and value of human-computer collaborations. The album is aptly named, as it alludes to the first words any new programmer uses when learning to code, as well as serving as an introduction to new AI-human collaborative practices. Hello, World, welcome to the new era of popular music.

   Similarly, emphasising the difference between interactive AI-assisted song composition, which Flow Composer pioneered, and fully automatic composition, Miller writes [14]:
   > On the one hand, we have François Pachet's Flow Machines, loaded with software to produce sumptuous original melodies, including a well-reviewed album. On the other, researchers at Google use artificial neural networks to produce music unaided. But at the moment their music tends to lose momentum after only a minute or so.

5 Conclusion

This paper was presented at the Application track of CP 2016. It is indeed an applied paper as it describes a system that has been used successfully to generate original music. In this note, however, we emphasise the fact that this paper is an offshoot of a more general and ambitious quest that transcends the dichotomy between "applied" and "basic" research. It is only because we had high-level, ambitious and ill-defined goals (essentially, show how AI could impact music composition) that this work was conducted, and it is only in the light of these goals that it made any sense.

[4] https://www.helloworldalbum.net/press/

[5] e.g. "AI and humans collaborated to produce this hauntingly catchy pop music" by Melissa Avdeef, or "Is this the world's first good robot album?" by Alex Marshall

References

1. Aucouturier, J., Pachet, F.: Improving timbre similarity : How high's the sky ? J. Negative Results in Speech and Audio Sciences (2004)
2. Aucouturier, J.J.: Ten Experiments on the Modelling of Polyphonic Timbre. Ph.D. thesis, University of Paris 6, Paris, France (May 2006), http://www.jj-aucouturier.info/papers/PHD-2006.pdf
3. Aucouturier, J.J., Pachet, F.: A scale-free distribution of false positives for a large class of audio similarity measures. Pattern Recognition **41**, 272–284 (2008)
4. Avdeef, M.: Artificial intelligence and popular music: Skygge, flow machines, and the audio uncanny valley. Arts **8**(130) (2019)
5. Bazin, T., Hadjeres, G.: NONOTO: A model-agnostic web interface for interactive music composition by inpainting. CoRR **abs/1907.10380** (2019)
6. Bernard, C.: An Introduction to the Study of Experimental Medicine. Dover Books on Biology) (1957)
7. Bessiere, C., Hebrard, E., Hnich, B., Kiziltan, Z., Walsh, T.: Filtering algorithms for the nvalueconstraint. Constraints **11**(4), 271–293 (2006). https://doi.org/10.1007/s10601-006-9001-9, https://doi.org/10.1007/s10601-006-9001-9
8. Bessiere, C., Katsirelos, G., Narodytska, N., Quimper, C., Walsh, T.: Decomposition of the nvalue constraint. In: CP. Lecture Notes in Computer Science, vol. 6308, pp. 114–128. Springer (2010)
9. Feldbauer, R., Flexer, A.: A comprehensive empirical comparison of hubness reduction in high-dimensional spaces. Knowl. Inf. Syst. **59**(1), 137–166 (2019)
10. Hebrard, E.: Reasoning about np-complete constraints. In: IJCAI. pp. 5672–5676. ijcai.org (2018)
11. Lamport, L.: My writings. https://lamport.azurewebsites.net/pubs/pubs.html, accessed: 2019-09-30
12. Martín, D., Neullas, T., Pachet, F.: Leadsheetjs: A javascript library for online lead sheet editing. In: Proceedings of the first TENOR Conference (05 2015)
13. Martín, D., Pachet, F., Frantz, B.: The Creative Process in Lead Sheet Composition: Perspectives from Multiple Domains, pp. 275–296. Palgrave Macmilan (08 2018)
14. Miller, A.: Creativity and ai: the next step. Scientific American (October 2019)
15. Pachet, F., Roy, P.: Markov constraints: steerable generation of Markov sequences. Constraints **16**(2) (2011)
16. Pachet, F., Roy, P., Barbieri, G.: Finite-length Markov processes with constraints. In: IJCAI. pp. 635–642 (2011)
17. Pachet, F.: The continuator: Musical interaction with style. In: ICMA (ed.) Proceedings of ICMC. pp. 211–218. ICMA, Goteborg, Sweden (September 2002), best paper award
18. Pachet, F., Roy, P.: Automatic generation of music programs. In: CP. Lecture Notes in Computer Science, vol. 1713, pp. 331–345. Springer (1999)
19. Pachet, F., Roy, P., Papadopoulos, A., Sakellariou, J.: Generating 1/f noise sequences as constraint satisfaction: The voss constraint. In: IJCAI. pp. 2482–2488. AAAI Press (2015)
20. Pachet, F., Roy, P., Papadopoulos, A., Briot, J.P.: Constrained Markov Sequence Generation Applications to Music and Text. Springer (2019)
21. Papadopoulos, A., Pachet, F., Roy, P., Sakellariou, J.: Exact sampling for regular and markov constraints with belief propagation. In: CP. Lecture Notes in Computer Science, vol. 9255, pp. 341–350. Springer (2015)

22. Papadopoulos, A., Roy, P., Pachet, F.: Avoiding Plagiarism in Markov Sequence Generation. In: AAAI. AAAI Press (2014)
23. Pearl, J.: Reverend bayes on inference engines: A distributed hierarchical approach. In: AAAI. pp. 133–136. AAAI Press (1982)
24. Perez, G., Rappazzo, B., Gomes, C.P.: Extending the capacity of 1 / f noise generation. In: CP. Lecture Notes in Computer Science, vol. 11008, pp. 601–610. Springer (2018)
25. Perez, G., Régin, J.: Mdds: Sampling and probability constraints. In: CP. Lecture Notes in Computer Science, vol. 10416, pp. 226–242. Springer (2017)
26. Radovanović, M., Nanopoulos, A., Ivanović, M.: Hubs in space: Popular nearest neighbors in high-dimensional data. Journal of Machine Learning Research **11**(Sep), 2487–2531 (2010)
27. Rowe, V., Triantafyllaki, A., Pachet, F.: Children's Creative Music-Making with Reflexive Interactive Technology: Adventures in improvising and composing. Routledge Research in Education, Routeledge (2016)
28. Roy, P., Pachet, F.: Enforcing Meter in Finite-Length Markov Sequences. In: desJardins, M., Littman, M.L. (eds.) AAAI. AAAI Press (2013)