Music Generation Using Bayesian Networks*

Tetsuro Kitahara[1]

College of Humanities and Sciences, Nihon University
3-25-40, Sakurajosui, Stagaya-ku, Tokyo 156-8550, Japan
kitahara@chs.nihon-u.ac.jp, http://www.kthrlab.jp/

Abstract. Music generation has recently become popular as an application of machine learning. To generate polyphonic music, one must consider both *simultaneity* (the vertical consistency) and *sequentiality* (the horizontal consistency). Bayesian networks are suitable to model both simultaneity and sequentiality simultaneously. Here, we present music generation models based on Bayesian networks applied to chord voicing, four-part harmonization, and real-time chord prediction.

1 Introduction

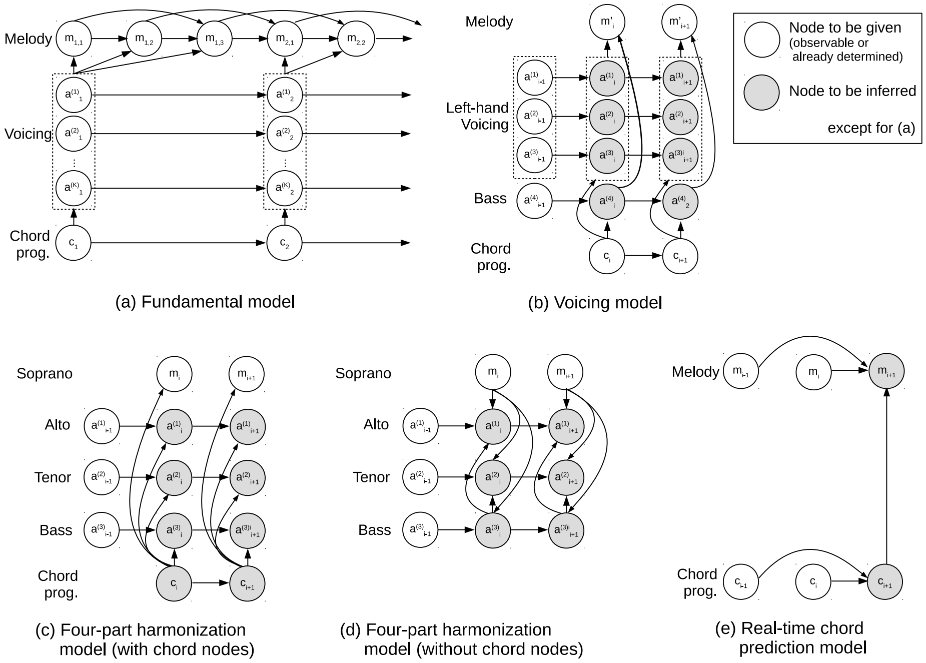
Music is widely known as an application domain of machine learning. However, in the beginning of the 21st century, recognition/analysis tasks were actively studied, such as music transcription and genre classification. But recently, the number of studies devoted to music generation has been increasing (e.g., [1]).

When generating polyphonic music, one must consider two-directional consistencies: *simultaneity* (i.e., the vertical or pitch-axis consistency) and *sequentiality* (i.e., the horizontal or time-axis consistency). Our team has investigated music generation models considering both simultaneity and sequentiality using Bayesian networks [2–4]. Here, we present our models applied to chord voicing [2], four-part harmonization [3], and real-time chord prediction [4].

2 Assumed music structure and fundamental model

Suppose that a chord progression $C = [c_1, c_2, \cdots, c_N]$ (c_i: chord symbol) exists in a piece of music. Each chord c_i (e.g., Am) is played with a particular voicing $(a_i^{(1)}, a_i^{(2)}, \cdots, a_i^{(K)})$ ($a_i^{(k)}$: note name (a.k.a. pitch class)) (e.g., (C, E, A)). As noted in Introduction, a set of simultaneous notes $(a_i^{(1)}, a_i^{(2)}, \cdots, a_i^{(K)})$ should be harmonically consistent with one other, and each sequence $A^{(k)} = [a_1^{(k)}, a_2^{(k)}, \cdots, a_N^k]$ should be temporally smooth. At the same time, a melody $M = [m_{1,1}, m_{1,2}, \cdots, m_{2,1}, \cdots]$ exists, where $m_{i,j}$ represents the note name of the j-th note in the i-th chord region. The sequences of chords, voicings, and melody notes are considered to have temporal dependencies within each sequence

* This work was supported by JSPS KAKENHI Grant Numbers 16K16180, 16H01744, 16KT0136, and 17H00749.



(a) Fundamental model

(b) Voicing model

(c) Four-part harmonization model (with chord nodes)

(d) Four-part harmonization model (without chord nodes)

(e) Real-time chord prediction model

Fig. 1. Fundamental model and models specialized to each task

but also depends on one another, as shown in Figure 1 (a). In fact, this fundamental model is difficult to construct because of variations in the number of melody notes within each chord region. We therefore simplify the model based on restrictions to music structures designed for each music generation task.

3 Chord voicing

Chord voicing refers to estimating voicings $(A^{(1)}, A^{(2)}, \cdots, A^{(K)})$ according to a given chord progression C and melody M. Here we assume $K = 4$ for simplicity. To resolve the difficulty due to variations in the number of melody notes within each chord region, we use a different melody node $m'_i = (r_{i,0}, \cdots, r_{i,11})$ $(0 \leq r_{i,p} \leq 1)$ that represents the relative length of the appearance of each note name. For example, $m'_i = (0.5, 0, 0.25, 0, 0.25, 0, \cdots, 0)$ is given for a melody [E, D, C, C] (with equal duration). The simplified model is shown in Figure 1 (b).

This model is applied sequentially from the beginning to the end of a given piece. Given c_i, m'_i, and $(a^{(1)}_{i-1}, \cdots, a^{(K)}_{i-1})$, the i-th chord voicing $(a^{(1)}_i, \cdots, a^{(K)}_i)$ as well as its next voicing $(a^{(1)}_{i+1}, \cdots, a^{(K)}_{i+1})$ is estimated because each voicing should be smoothly connected to the next voicing. $(a^{(1)}_{i+1}, \cdots, a^{(K)}_{i+1})$ will be overriden at the next step because this step is repeated for each increment of i.

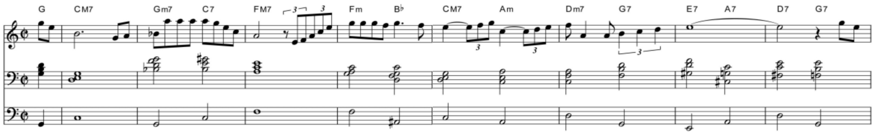


Fig. 2. An example of voicing (excerpted)

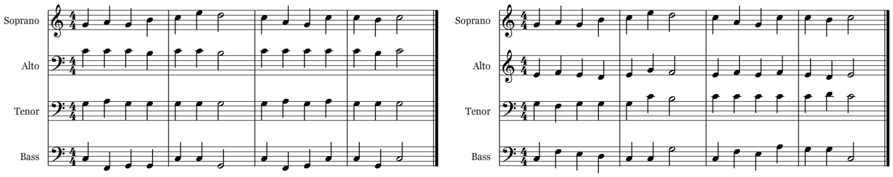


Fig. 3. Example of harmonization (Left: model with chord nodes, Right: model withuot chord nodes)

An example of chord voicing is shown in Figure 2. The model has been trained with 30 jazz pieces arranged for the electronic organ. Listening tests conducted by music experts revealed that 94.7% of the chord voicings were acceptable.

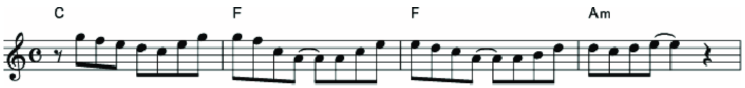
4 Four-part harmonization

Here, we focus on harmonization. Unlike voicing, a sequence of chord symbols is not given—it has to be estimated. For simplicity, we adopt the "one chord for one melody note" assumption. Based on this assumption, the Bayesian network can be simplified to that shown in Figure 1 (c). Here we assume $K = 3$. This problem is called four-part harmonization because the harmony consists of four voices (i.e., soprano, alto, tenor, and bass). Furthermore, we constructed a Bayesian network in which the chord nodes are removed (Figure 1 (d)) because the chord symbols are sometimes too ambiguous.

Figure 3 shows an example of harmonization using these two models. Our objective quantitative evaluation reveals that the model shown in Figure 1 (d) generates more temporally smooth harmonies than the model shown in Figure 1 (c) even though harmonizations with the former model tend to contain slightly more dissonant sounds.

5 Real-time chord prediction

Finally, we apply our Bayesian network to real-time chord prediction. Music experts can often precisely predict the next chord by listening to the current chord, even if they are not familiar with the piece being played. This ability derives from the fact that chord progressions have strong temporal dependencies; experts have learned these dependencies based on their musical experience. They



Melody prediction			Chord inference	
C (.23)	E (.15)	D (.15)	F (.42)	C (.16)
C (.30)	**E** (.26)	G (.18)	F (.45)	Em (.16)
D (.25)	G (.25)	C (.16)	F (.41)	Dm (.16)
C (.45)	G (.16)	B (.13)	F (.48)	Em (.16)
G (.21)	**E** (.17)	B (.13)	F (.34)	Em (.18)
C (.47)	**G** (.19)	F (.08)	F (.50)	Dm (.15)
C (.34)	B (.34)	D (.07)	F (.39)	Em (.22)
A (.28)	E (.20)	G (.17)	F (.30)	Em (.22)
F (.16)	E (.16)	A (.13)	F (.32)	Em (.18)
E (.33)	**A** (.23)	F (.13)	F (.33)	Em (.20)
D (.22)	G (.22)	C (.20)	F (.30)	C (.22)
E (.43)	**C** (.25)	G (.11)	F (.29)	C (.24)
A (.33)	G (.19)	**E** (.13)	F (.31)	Em (.21)
C (.47)	G (.19)	F (.08)	F (.29)	C (.25)
C (.47)	G (.19)	F (.08)	C (.30)	F (.19)
C (.56)	G (.10)	E (.10)	C (.32)	F (.20)
A (.27)	D (.17)	E (.15)	Am (.22)	F (.21)
D (.22)	G (.22)	C (.19)	Am (.21)	C (.20)
E (.43)	C (.25)	G (.11)	Am (1.0)	
C (.38)	Others (.06)		C (.28)	F (.19)
B (.32)	C (.14)	E (.14)	Am (.22)	C (.17)
C (.51)	G (.11)	D (.09)	Dm (.23)	Am (.14)
G (.21)	E (.17)	B (.13)	Dm (.16)	Am (.15)
C (.40)	A (.30)	**E**,G (.07)	Dm (.22)	Am (.14)
C (.36)	E (.14)	D,G (.11)	Dm (.21)	Am (.15)

Fig. 4. Example of real-time chord prediction results

are therefore able to play an accompaniment to a melody that they are listening to for the first time. The goal here is to achieve a computer system that plays such an accompaniment.

Real-time chord prediction can also be achieved through a simplified version of the fundamental model shown in Figure 1 (a). For simplicity, we estimate only chord symbols, we determine the voicings through a separately designed rule. The model used here is shown in Figure 1 (e). Given a new melody note, its next note is predicted. At the same time, the most likely next chord is inferred based on the current chord and the predicted next note.

An example of chord prediction is shown in Figure 4. This figure shows that the model appropriately predicts chord progression.

6 Conclusion

We have presented Bayesian network models that achieve different music generation tasks: chord voicing, four-part harmonization, and real-time chord prediction. Bayesian networks are flexible models that are suitable to construct a unified music generation model. In the future, we will apply our model to other types of music generation tasks.

References

1. G. Harjeres and F. Pachet: DeepBach: A Steerable Model for Bach Chorales Generation, arXiv:1612.01010 [cs.AI], 2016.
2. T. Kitahara, M. Katsura, H. Katayose, and N. Nagata: Computational Model for Automatic Chord Voicing based on Bayesian Network, *ICMPC*, pp.395–398, 2008.
3. S. Suzuki and T. Kitahara: Four-part Harmonization Using Bayesian Networks: Pros and Cons of Introducing Chord Nodes, *J. New Mus. Res.*, 43, 3, pp.331–353, 2014.
4. T. Kitahara, N. Totani, R. Tokuami, and H. Katayose: BayesianBand: Jam Session System based on Mutual Prediction by User and System, *ICEC*, pp.179–184, 2009.