



Realtime Musical Composition System for Automatic Driving Vehicles

Yoichi Nagashima

► To cite this version:

Yoichi Nagashima. Realtime Musical Composition System for Automatic Driving Vehicles. 17th International Conference on Entertainment Computing (ICEC), Sep 2018, Poznan, Poland. pp.217-222, 10.1007/978-3-319-99426-0_20 . hal-02128642

HAL Id: hal-02128642

<https://inria.hal.science/hal-02128642>

Submitted on 14 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Realtime Musical Composition System for Automatic Driving Vehicles

Yoichi Nagashima¹[0000-0002-2465-1428]

¹ Shizuoka University of Art and Culture, 2-1-1 Chuo, Hamamatsu. Shizuoka, JAPAN

nagasm@suac.ac.jp

Abstract. Automatic driving vehicles (ADV) are drawing attention all over the world. ADV contains many realtime sensors. In the future, people (including the driver) will enjoy BGM without attention to ambient conditions, however, it will be much better than the BGM corresponding to the surrounding situation in real time - rather than being properly chosen from existing music. The author proposes an approach of a "realtime musical composition system for automatic driving vehicles" which generates music in real time without using existing music, so we are free from the copyright issue. The realtime composition system can arrange/modify its generating musical factors/elements with realtime parameters such as sensor information in real time, so it is the best solution for "music in ADV". This paper reports on the first prototype of realtime composition system for ADV - as collaborative research (2015-2017) with Toyota Central R&D Labs.

Keywords: Realtime Composition, Automatic Driving Vehicle, Copyright Free.

1. Introduction

Automatic driving vehicles (ADV) are drawing attention all over the world. ADV contains many realtime sensors : (1) radar sensors and distance sensors for preventing collision, (2) video cameras for "drive recorders" and "monitoring blind spots", (3) GPS receiver, and (4) CAN (Controller Area Network) system with driving data - steering, brake, accelerator, speed, etc.

In the future, passengers in the car will enjoy BGM (BackGround Music) without attention to ambient conditions, however, it will be much better than the BGM corresponding to the surrounding situation in real time. For example, if we are in a car driving through a tunnel and the GPS knows that we will exit the tunnel and come out onto the coast, "BGM making us feel the beach" will be fantastic.

Here, we face a traditional problem, copyright matters. The author proposes an approach of using "realtime musical composition system for automatic driving vehicles" which generates music in real time. This is the same idea of "algorithmic musical composition" in the computer music field and it has a long history [1]. The realtime composition (algorithmic composition) system generates music in real time

without using existing music, so we are free from copyright concerns. Also, the realtime composition system can arrange/modify its generating musical factors/elements with realtime parameters such as sensor information in real time, so it seems to be the best solution for "music in ADV".

This paper reports the first prototype of realtime composition system for ADV in collaboration (2015-2017) with Toyota Central R&D Labs (TytLabs) [2]. Researchers of TytLabs contacted the author one day because they discovered a research report by the author (2005-2006) titled "FMC³ (Free Music Clip for Creative Common)". This was a copyright-free music clip generation system for content creators (only in Japanese) [3].

At that time (2003-2004), "Flash movies" were explosively prevalent all over the world - satire, gags, art, etc. However the music part of these flash movies was "existing music" without copyright management, so they eventually disappeared from the internet. As a longtime composer and researcher in computer music the author worked on projects of "realtime composition" system for web content creators aimed without copyright invalidity. The basic policy of the system is the "importance of heuristics in music", so the output music has "4 measure loop music" with traditional/theoretical chord sequences and realtime musical transposition without "intro" nor "ending". The 55 types of chord sequence patterns are randomly selected, 3 types of music styles (8beat, 16beat, shuffle) [3]. This project with TytLabs inherited and expanded on these FMC³'s ideas.

2. Pre-test and Discussion

Before contacting the author, researchers of TytLabs had conducted experiments by themselves. For example, the output data of laser doppler sensors were assigned to musical sounds, then a piano melody was generated like a chime phrase. For example, the information from a video camera was converted to volume, so the sound expressed the brightness or complexity of the outdoors. However, they thought that was not music, but only sound realtime generated corresponding to the outside world. Then, they contacted the author, entered into a contract with the author's university and TytLabs and the collaborative research project began.

In the first year, one of young researcher's of TytLabs came to the author's lab once/twice per month and studied musical theory and cutting-edge computer music aesthetics/technology/theory, and we had some discussions about this challenging field. However, in the second year, the young researcher who studied music was suddenly transferred to another department, so TytLabs leader requested the author to develop a prototype of realtime composition system that substantially shifted from collaborative research to contract research.

We had a deep discussion about why the past experimental prototypes of TytLabs did not succeed. There was no musical construction - metrical structure, rhythm/beat and tonal structure. Although it had a scale corresponding to the output information from the sensor, it was a list of sounds, not a melody. As requested to develop new prototype, the author took a similar approach to FMC³ that aimed at expanding

musical heuristics. Finally we succeeded in design the rough system block diagram of realtime composition for BGM for automatic driving vehicles (Fig.1).

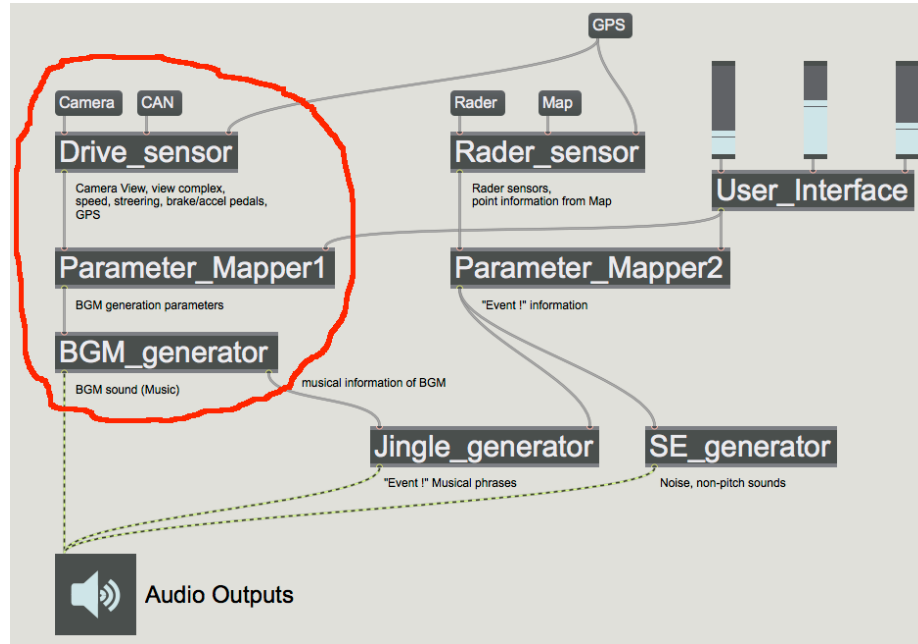


Fig. 1. System block diagram of the experimental prototype system

3. System Description

The system of Fig.1 has roughly two blocks. We call the left block "BGM block" and the right block "momentary event block". The latter is like TytLabs's prototype using doppler radar sensor and Map+GPS, which generates "momentary" sounds - musical (tonal) elements (Jingle generator) and noisy elements (SE generator). The former is an FMC³-like automatic composer using musical theory, know-how and heuristics with live camera and CAN data (driving, speed, etc), and this block (red circled area) is reported on in this paper.

The important point is the connection from "BGM generator" to "Jingle Generator". This connection is the key of the total musical structure - metrical structure, rhythm/beat and tonal structure (chord, scale, tension). If the "Jingle Generator" works without this connection, the BGM and Jingle (small phrases) will collide and the music will be destroyed. The "BGM generator" sends the available chord/scale (information of tonality), so the total music is harmonized.

To develop and test collaboratively, the "BGM generator" was divided into three blocks, because the lab of the author (development) and TytLabs (testing) were distant from each other. The final prototype system will be mounted into a Toyota test car,

but in the development process, the author could not get any "live" sensor data. So, the block "Drive sensor" was changed to a "Drive recorder" and a "Drive Player". At first, TytLabs staff drove the test car and recorded all sensor data into a movie file and a text data file using a "Drive recorder". Next, the author used a "Drive Player" which generates (plays) all sensor data like "live" mode, and developed a realtime composition program using the playback data. Finally, TytLabs staff could drive and test/check the music generator program with the replaced "Drive sensor" live. The "Drive Player" [4] screen shows considerable sensor data and the live camera view. It also contains "skip" buttons to jump to frames anywhere in the drive.

The next important point is dividing "musical generator" function into two blocks. - the upstream "Parameter mapper" block [5] and the downstream "BGM generator" block [4]. In the computer music field it is well-known that the concept "parameter mapping" is the key point in realtime composition and interactive performance [1]. The output from the "Drive sensor" or the "Drive player" are only driving-related physical/technical parameters. However, the "BGM generator" works with many "musical" parameters such as, musical tension, groove feeling, expectation of chord progression, etc. If these different hierarchy parameters are directly connected/transferred, the result will be a fatal failure - there have been a history of successive failures in compute music (in Japan). Thus, this system is divided into two blocks - a parameter "mapping" block from driving parameters to musical parameters, and a music generating only block.

This idea has the merit that the author can develop a "BGM generator" as separated from TytLabs's field test. If the staff of TytLabs wants to change the relationship between the sensor data and the generated music, he can change only the "Parameter mapper" block. Because the "BGM generator" is developed with pure musical theory/heuristics, to arrange/modify/change from the field request will perform badly in the project. Whole system was constructed on "Max7" [6] environment. The detail of the whole system (subpatches) are shown on the web [7].

4. Explanation of Musical Structure

This is the outline the musical structure of the realtime composition algorithm. There are four musical styles - called "8beat", "16beat", "ballad" and "shuffle" - these loop music styles are comfortable for all people. The tempo of BGM is fixed for each musical style, and never changes the while driving. The amateur will consider that if the car speeds up, the BGM will speed up (if the car slows down, the BGM will slow down), but this is very unnatural. The Volkswagen UK presented "Play the Road" system in 2013 [8], this system (not for ADV, but for the driver) also adopted the "constant tempo" concept, as the musical designer of "Play the Road" knew music well.

The BGM does not have an "intro" nor "ending", as the realtime composition works eternally. However, if the car stops at a traffic light the BGM volume decreases (diminuendo) and if the car starts again the BGM volume increases (crescendo) and sometimes the timbre of melody changes, or the musical style changes. The BGM is

constructed as 4 measures loop music with a four chord sequence. From the deep musical consideration (space does not permit description here) [9], the 4-chord sequence is defined by 152 types (Fig.2). For example, the chord sequence #63: [Bb7 - Eb7 - Dm7 - G7] (in the Key: C major) is constructed upon this theoretical reason: [Bb7 - Eb7] is "secondary dominant" sequence, [Eb7 - Dm7] is "sub(contrary) 5th" sequence, [Dm7 - G7] is "two-five" sequence, final [G7] (dominant 7th) wants to go to the tonal center root [C] ("dominant motion", all 4th chord of 152 types are G7 or Db7). 152 chord sequences have musical reasons backed by theory.

Window 1 (Untitled.txt)	Window 2 (Untitled 2.txt)	Window 3 (Untitled 3.txt)	Window 4 (Untitled 4.txt)
1 CM7Am7Dm7G7 24 21 14 7	39 Eb7D7Ab7Db7 3 2 8 1	77 E7Eb7D7G7 4 3 2 7	115 Bb7A7Abm7Db7 10 9 20 1
2 CM7Am7D7G7 24 21 2 7	40 Ebm7D7Ab7Db7 15 2 8 1	78 Em7Eb7D7G7 16 3 2 7	116 Bbm7A7Abm7Db7 22 9 20 1
3 CM7A7Dm7G7 24 9 14 7	41 A7D7Abm7Db7 9 2 20 1	79 Bb7Eb7D7G7 10 3 2 7	117 E7Eb7Abm7Db7 4 3 20 1
4 CM7A7D7G7 24 9 2 7	42 Am7D7Abm7Db7 21 2 20 1	80 Bbm7Eb7D7G7 22 3 2 7	118 Em7Eb7Abm7Db7 16 3 20 1
5 CM7Am7Dm7Db7 24 21 14 1	43 Eb7D7Abm7Db7 3 2 20 1	81 E7A7D7Db7 4 9 2 1	119 Bb7Eb7Abm7Db7 10 3 20 1
6 CM7Am7D7Db7 24 21 2 1	44 Ebm7D7Abm7Db7 15 2 20 1	82 Em7A7D7Db7 16 9 2 1	120 Bbm7Eb7Abm7Db7 22 3 20 1
7 CM7A7Dm7Db7 24 9 14 1	45 A7Ab7Dm7Db7 9 8 14 1	83 Bb7A7D7Db7 10 9 2 1	121 Bb7Ebm7D7G7 10 15 2 7
8 CM7A7D7Db7 24 9 2 1	46 Am7Ab7Dm7Db7 21 8 14 1	84 Bbm7A7D7Db7 22 9 2 1	122 E7Ebm7D7G7 4 15 2 7
9 A7D7Dm7G7 9 2 14 7	47 Eb7Ab7Dm7Db7 3 8 14 1	85 E7Eb7D7Db7 4 3 2 1	123 Bb7Am7D7G7 10 21 2 7
10 Am7D7Dm7G7 21 2 14 7	48 Ebm7Ab7Dm7Db7 15 8 14 1	86 Em7Eb7D7Db7 16 3 2 1	124 E7Am7D7G7 4 21 2 7
11 Eb7D7Dm7G7 3 2 14 7	49 A7Ab7D7Db7 9 8 2 1	87 Bb7Eb7D7Db7 10 3 2 1	125 Bb7Ebm7D7Db7 10 15 2 1
12 Ebm7D7Dm7G7 15 2 14 7	50 Am7Ab7D7Db7 21 8 2 1	88 Bbm7Eb7D7Db7 22 3 2 1	126 E7Ebm7D7Db7 4 15 2 1
13 A7D7Ab7G7 9 2 8 7	51 Eb7Ab7D7Db7 3 8 2 1	89 E7A7Ab7G7 4 9 8 7	127 Bb7Am7D7Db7 10 21 2 1
14 Am7D7Ab7G7 21 2 8 7	52 Ebm7Ab7D7Db7 15 8 2 1	90 Em7A7Ab7G7 16 9 8 7	128 E7Am7D7Db7 4 21 2 1
15 Eb7D7Ab7G7 3 2 8 7	53 A7Ab7Abm7Db7 9 8 20 1	91 Bb7A7Ab7G7 10 9 8 7	129 Bb7Ebm7Ab7G7 10 15 8 7
16 Ebm7D7Ab7G7 15 2 8 7	54 Am7Ab7Abm7Db7 21 8 20 1	92 Bbm7A7Ab7G7 22 9 8 7	130 E7Ebm7Ab7G7 4 15 8 7
17 A7D7Abm7G7 9 2 20 7	55 Eb7Ab7Abm7Db7 3 8 20 1	93 E7Eb7Ab7G7 4 3 8 7	131 Bb7Am7Ab7G7 10 21 8 7
18 Am7D7Abm7G7 21 2 20 7	56 Ebm7Ab7Abm7Db7 15 8 20 1	94 Em7Eb7Ab7G7 16 3 8 7	132 E7Am7Ab7G7 4 21 8 7
19 Eb7D7Abm7G7 3 2 20 7	57 E7A7Dm7G7 4 9 14 7	95 Bb7Eb7Ab7G7 10 3 8 7	133 Bb7Ebm7Ab7Db7 10 15 8 1
20 Ebm7D7Abm7G7 15 2 20 7	58 Em7A7Dm7G7 16 9 14 7	96 Bbm7Eb7Ab7G7 22 3 8 7	134 E7Ebm7Ab7Db7 4 15 8 1
21 A7Ab7Dm7G7 9 8 14 7	59 Bb7A7Dm7G7 10 9 14 7	97 E7A7Ab7Db7 4 9 8 1	135 Bb7Am7Ab7Db7 10 21 8 1
22 Am7Ab7Dm7G7 21 8 14 7	60 Bbm7A7Dm7G7 22 9 14 7	98 Em7A7Ab7Db7 16 9 8 1	136 E7Am7Ab7Db7 4 21 8 1
23 Eb7Ab7Dm7G7 3 8 14 7	61 E7Eb7Dm7G7 4 3 14 7	99 Bb7A7Ab7Db7 10 9 8 1	137 A7Ebm7D7G7 9 15 2 7
24 Ebm7Ab7Dm7G7 15 8 14 7	62 Em7Eb7Dm7G7 16 3 14 7	100 Bbm7A7Ab7Db7 22 9 8 1	138 Eb7Ebm7D7G7 3 15 2 7
25 A7Ab7D7G7 9 8 2 7	63 Bb7Eb7Dm7G7 10 3 14 7	101 E7Eb7Ab7Db7 4 3 8 1	139 A7Am7D7G7 9 21 2 7
26 Am7Ab7D7G7 21 8 2 7	64 Bbm7Eb7Dm7G7 22 3 14 7	102 Em7Eb7Ab7Db7 16 3 8 1	140 Eb7Am7D7G7 3 21 2 7
27 Eb7Ab7D7G7 3 8 2 7	65 E7A7Dm7Db7 4 9 14 1	103 Bb7Eb7Ab7Db7 10 3 8 1	141 A7Ebm7D7Db7 9 15 2 1
28 Ebm7Ab7D7G7 15 8 2 7	66 Em7A7Dm7Db7 16 9 14 1	104 Bbm7Eb7Ab7Db7 22 3 8 1	142 Eb7Ebm7D7Db7 3 15 2 1
29 A7Ab7Abm7G7 9 8 20 7	67 Bb7A7Dm7Db7 10 9 14 1	105 E7A7Abm7G7 4 9 20 7	143 A7Am7D7Db7 9 21 2 1
30 Am7Ab7Abm7G7 21 8 20 7	68 Bbm7A7Dm7Db7 22 9 14 1	106 Em7A7Abm7G7 16 9 20 7	144 Eb7Am7D7Db7 3 21 2 1
31 Eb7Ab7Abm7G7 3 8 20 7	69 E7Eb7Dm7Db7 4 3 14 1	107 Bb7A7Abm7G7 10 9 20 7	145 Eb7Ebm7Ab7G7 3 15 8 7
32 Ebm7Ab7Abm7G7 15 8 20 7	70 Em7Eb7Dm7Db7 16 3 14 1	108 Bbm7A7Abm7G7 22 9 20 7	146 A7Ebm7Ab7G7 9 15 8 7
33 A7D7Dm7Db7 9 2 14 1	71 Bb7Eb7Dm7Db7 10 3 14 1	109 E7Eb7Abm7G7 4 3 20 7	147 Eb7Am7Ab7G7 3 21 8 7
34 Am7D7Dm7Db7 21 2 14 1	72 Bbm7Eb7Dm7Db7 22 3 14 1	110 Em7Eb7Abm7G7 16 3 20 7	148 A7Am7Ab7G7 9 21 8 7
35 Eb7D7Dm7Db7 3 2 14 1	73 E7A7D7G7 4 9 2 7	111 Bb7Eb7Abm7G7 10 3 20 7	149 Eb7Ebm7Ab7Db7 3 15 8 1
36 Ebm7D7Dm7Db7 15 2 14 1	74 Em7A7D7G7 16 9 2 7	112 Bbm7Eb7Abm7G7 22 3 20 7	150 A7Ebm7Ab7Db7 9 15 8 1
37 A7D7Ab7Db7 9 2 8 1	75 Bb7A7D7G7 10 9 2 7	113 E7A7Abm7Db7 4 9 20 1	151 Eb7Am7Ab7Db7 3 21 8 1
38 Am7D7Ab7Db7 21 2 8 1	76 Bbm7A7D7G7 22 9 2 7	114 Em7A7Abm7Db7 16 9 20 1	152 A7Am7Ab7Db7 9 21 8 1

Fig. 2. 152 types of chord sequence in the prototype system

The BGM is constructed by "chord", "bass", "drums" and "melody" parts. In all loops, when the next chord sequence number is selected randomly at the final moment of the previous measure, all musical elements in the next four chords are combined by small-depth randomness (ad lib). For example, the "drums" part is generated corresponding to the musical style, and added with small variations randomly. The "bass" part is generated corresponding to the musical style and chord of each measure (1-4) using root/5th/7th notes of the chord with small variations of the bass phrases. The "chord" part is generated correspondingly to the musical style and chord of each measure, not played at once but played sequentially like "arpeggio" to prevent the sound from overlapping/distorting at the first moment. The "melody" (named for this project only, not a proper melody) part is generated corresponding to the musical style and chord of each measure, like an improvisational "ad lib" phrase with small variations of rhythm randomly. The available notes are - chord notes, available

tension notes in JAZZ musical theory. The appearance probability of tension notes is changed randomly with the musical parameter "musical tension" from the "Parameter mapping" block.

5. Conclusions

After the development and test-run, we had a concluding discussion. The first objective, "musical level of the BGM" was successful, however, "music that makes us feel the surrounding situation" was not satisfactorily created yet. The reported prototype system only generated the BGM-part, so we could not combine the system with "Jingle Generator" and "SE Generator" which generates "momentary" sounds, this is one reason of the dissatisfaction. This means the "program music" in traditional classic music, and is opposite to loop music that we can listen to comfortably. The classical "program music" has exaggerated intros and endings, and the tempo changes by the way. Music is eternal and we faced the essence of music.

The musical know-how gained in this project is effectively utilized in the place of music education for designers. And, the author will demonstrate the system at short paper presentation session in ICEC2018. This first project has finished, and we hope to contribute to the fantastic future of ADV music.

Acknowledgments

The author would like to express his appreciation to Nobushige Fujieda, Yoshiyuki Akai and Hisanori Uda (Toyota Central R&D Labs. Human Science Research Area, Expand-Senses Program).

References

1. Curtis Roads, The Computer Music Tutorial, Mit Press, 1996.
2. Toyota Central R&D Labs, <https://www.tytlabs.com/>
3. Yoichi Nagashima, FMC³, <http://nagasm.org/FMC3/>
4. http://nagasm.org/ASL/Max7_5/fig4/003.jpg
5. http://nagasm.org/ASL/Max7_5/fig4/004.jpg
6. Max. <https://cycling74.com/products/max>
7. http://nagasm.org/ASL/Max7_5/fig4/005.jpg — http://nagasm.org/ASL/Max7_5/fig4/026.jpg
8. "Play the Road". <http://www.youtube.com/watch?v=3flwZ8OpXBY>
9. Yoichi Nagashima, Towards a "realtime musical composing" system for autonomous vehicle, <http://nagasm.org/ASL/paper/SIGMUS201802.pdf> (in Japanese)