MUSICAL STRUCTURAL ANALYSIS DATABASE BASED ON GTTM

Masatoshi Hamanaka
Kyoto University
masatosh@kuhp.kyoto-u.ac.jp

Keiji Hirata
Future University Hakodate
hirata@fun.ac.jp

Satoshi Tojo
JAIST
tojo@jaist.ac.jp

ABSTRACT

This paper, we present the publication of our analysis data and analyzing tool based on the generative theory of tonal music (GTTM). Musical databases such as score databases, instrument sound databases, and musical pieces with standard MIDI files and annotated data are key to advancements in the field of music information technology. We started implementing the GTTM on a computer in 2004 and ever since have collected and publicized test data by musicologists in a step-by-step manner. In our efforts to further advance the research on musical structure analysis, we are now publicizing 300 pieces of analysis data as well as the analyzer. Experiments showed that for 267 of 300 pieces the analysis results obtained by a new musicologist were almost the same as the original results in the GTTM database and that the other 33 pieces had different interpretations.

1. INTRODUCTION

For over ten years we have been constructing a musical analysis tool based on the generative theory of tonal music (GTTM) [1, 2]. The GTTM, proposed by Lerdahl and Jackendoff, is one in which the abstract structure of a musical piece is acquired from a score [3]. Of the many music analysis theories that have been proposed [4–6], we feel that the GTTM is the most promising in terms of its ability to formalize musical knowledge because it captures aspects of musical phenomena based on the Gestalt occurring in music and then presents these aspects with relatively rigid rules.

The time-span tree and prolongational trees acquired by GTTM analysis can be used for melody morphing, which generates an intermediate melody between two melodies with a systematic order [7]. It can also be used for performance rendering [8–10] and reproducing music [11] and provides a summarization of the music that can be used as a search representation in music retrieval systems [12].

In constructing a musical analyzer, test data from musical databases is very useful for evaluating and improving the performance of the analyzer. The Essen folk song collection is a database for folk-music research that contains score data on 20,000 songs along with phrase segmentation information and also provides software for processing the data [13]. The Répertoire International des Sources Musicales (RISM), an international, non-profit organization with the aim of comprehensively documenting extant musical sources around the world, provides an online catalogue containing over 850,000 records, mostly for music manuscripts [14]. The Variations3 project provides online access to streaming audio and scanned score images for the music community with a flexible access control framework [15], and the Real World Computing (RWC) Music Database is a copyright-cleared music database that contains the audio signals and corresponding standard MIDI files for 315 musical pieces [16,17]. The Digital Archive of Finnish Folk Tunes provides 8613 finish folk song midi files with annotated meta data and Matlab data matrix encoded by midi toolbox [18]. The Codaich contains 20,849 MP3 recordings, from 1941 artists, with high-quality annotations [19], and the Latin Music Database contains 3,227 MP3 files from different music genres [20].

When we first started constructing the GTTM analyzer, however, there was not much data that included both a score and the results of analysis by musicologists. This was due to the following reasons:

- There were no computer tools for GTTM analysis.

Only a few paper-based analyses of GTTM data had been done because a data-saving format for computer analysis had not yet been defined. We therefore defined an XML-based format for analyzing GTTM results and developed a manual editor for the editing.

- Editing the tree was difficult.

Musicologists using the manual editor to acquire analysis results need to perform a large number of manual operations. This is because the time-span and prolongational trees acquired by GTTM analysis are binary trees, and the number of combinations of tree structures in a score analysis increases exponentially with the number of notes. We therefore developed an automatic analyzer based on the GTTM.

- There was a lack of musicologists.

Only a few hundred musicologists can analyze scores by using the GTTM. In order to encourage musicologists to co-operate with expanding the GTTM database, we publicized our analysis tool and analysis data based on the GTTM.

- The music analysis was ambiguous.

A piece of music generally has more than one interpretation, and dealing with such ambiguity is a major problem when constructing a music analysis database. We performed experiments to compare the different analysis results obtained by different musicologists.

We started implementing our GTTM analyzer on a computer in 2004, immediately began collecting test data produced by musicologists, and in 2009 started publicizing the GTTM database and analysis system. We started the GTTM database with 100 pairs of scores and time-span trees comprising and then added the prolongational trees and chord progression data. At present, we have 300 data sets that are being used for researching music structural analysis [1]. The tool we use for analyzing has changed from its original form. We originally constructed a standalone application for the GTTM-based analysis system, but when we started having problems with bugs in the automatic analyzer, we changed the application to a client-server system.

In experiments we compared the analysis results of two different musicologists, one of whom was the one who provided the initial analysis data in the GTTM database. For 267 of 300 pieces of music the two results were the same, but the other 33 pieces had different interpretations. Calculating the coincidence of the time-spans in those 33 pieces revealed that 233 of the 2310 time-spans did not match.

This rest of this paper is organized as follows. In section 2 we describe the database design policy and data sets, in section 3 we explain our GTTM analysis tool, in section 4 we present the experimental results, and in section 5 we conclude with a brief summary.

2. GTTM DATABASE

The GTTM is composed of four modules, each of which assigns a separate structural description to a listener's under-standing of a piece of music. Their output is a grouping structure, a metrical structure, a time-span tree, and a prolongational tree (Fig. 1).

The grouping structure is intended to formalize the intuitive belief that tonal music is organized into groups comprising subgroups. The metrical structure describes the rhythmical hierarchy of the piece by identifying the position of strong beats at the levels of a quarter note, half note, one measure, two measures, four measures,

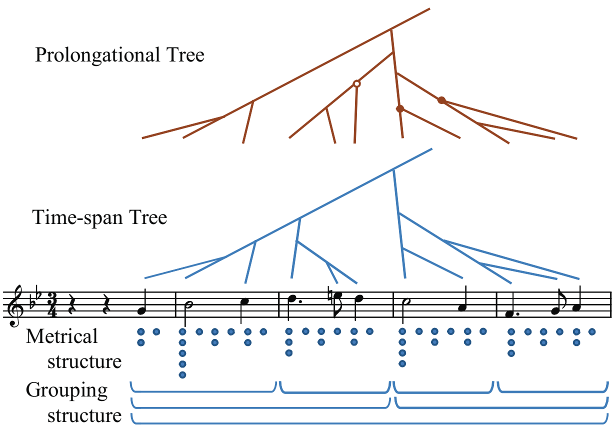

Figure 1. Grouping structure, metrical structure, time-span tree, and prolongational tree.

and so on. The time-span tree is a binary tree and is a hierarchical structure describing the relative structural importance of notes that differentiate the essential parts of the melody from the ornamentation. The prolongational tree is a binary tree that expresses the structure of tension and relaxation in a piece of music.

2.1 Design policy of analysis database

As at this stage several rules in the theory allow only monophony, we restrict the target analysis data to monophonic music in the GTTM database.

2.1.1 Ambiguity in music analysis

We have to consider two types of ambiguity in music analysis. One involves human understanding of music and tolerates subjective interpretation, while the latter concerns the representation of music theory and is caused by the incompleteness of a formal theory like the GTTM. We therefore assume because of the former type of ambiguity that there is more than one correct result.

2.1.2 XML-based data structure

We use an XML format for all analysis data. MusicXML [22] was chosen as a primary input format because it provides a common 'interlingua' for music notation, analysis, retrieval, and other applications. We designed GroupingXML, MetricalXML, TimespanXML, and ProlongationalXML as the export formats for our analyzer. We also designed HarmonicXML to express the chord progression. The XML format is suitable for expressing the hierarchical grouping structures, metrical structures, time-span trees, and prolongational trees.

2.2 Data sets in GTTM database

The database should contain a variety of different musical pieces, and when constructing it we cut 8-bar-long pieces from whole pieces of music because the time required for analyzing and editing would be too long if whole pieces were analyzed.

2.2.1 Score data

We collected 300 8-bar-long monophonic classical music pieces that include notes, rests, slurs, accents, and articulations entered manually with music notation software called Finale [22]. We exported the MusicXML by using a plugin called Dolet. The 300 whole pieces and the eight bars were selected by a musicologist.

2.2.2 Analysis data

We asked a musicology expert to manually analyze the score data faithfully with regard to the GTTM, using the manual editor in the GTTM analysis tool to assist in editing the grouping structure, metrical structure, time-span tree, and prolongational tree. She also analyzed the chord progression. Three other experts crosschecked these manually produced results.

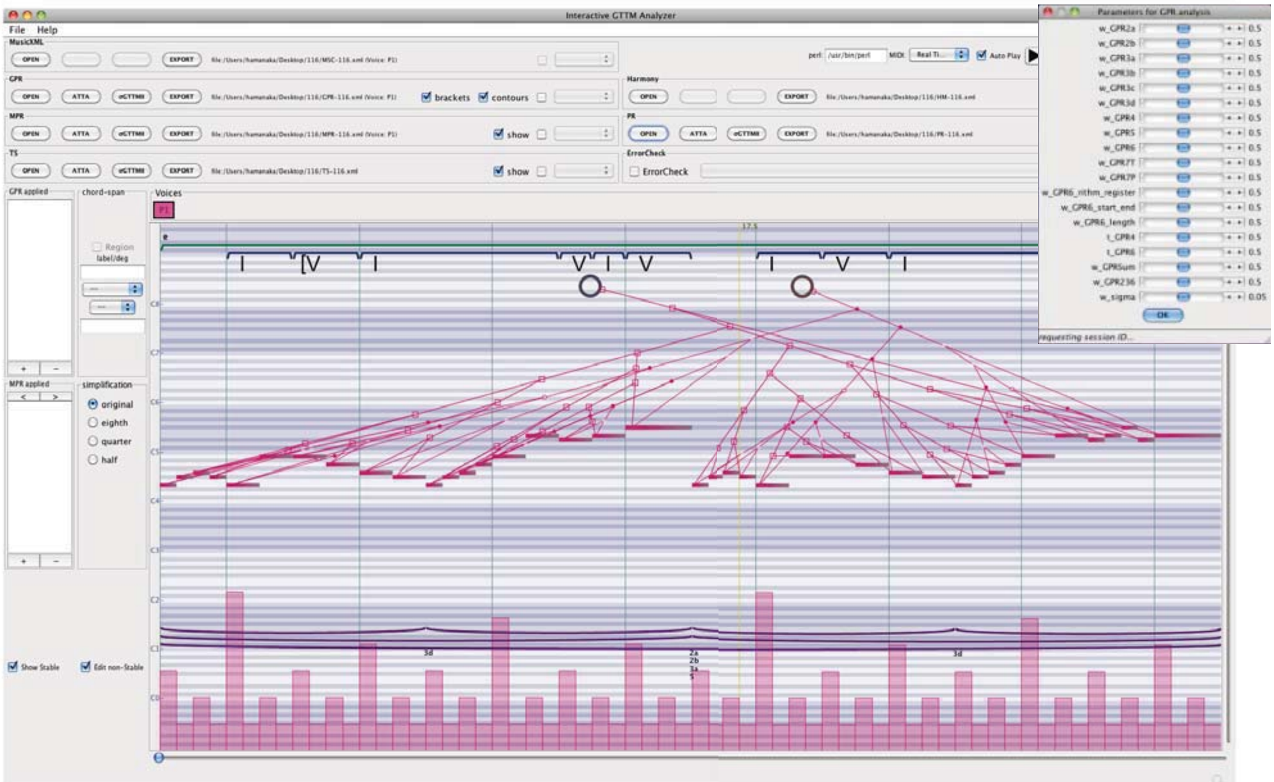

Figure 2. Interactive GTTM analyzer.

3. INTERACTIVE GTTM ANALYZER

Our GTTM analysis tool, called the Interactive GTTM analyzer, consists of automatic analyzers and an editor that can be used to edit the analysis results manually (Fig. 2). The graphic user interface of the tool was constructed in Java, making it usable on multiple platforms. However, some functions of the manual editor work only on MacOSX, which must use the MacOSX API.

3.1 Automatic analyzer for GTTM

We have constructed four types of GTTM analyzers: ATTA, FATTA, σGTTM, and σGTTMII [2, 23–25]. The Interactive GTTM analyzer can use either the ATTA or the σGTTMII, and there is a trade-off relationship between the automation of the analysis process and the variation of the analysis results (Fig. 3).

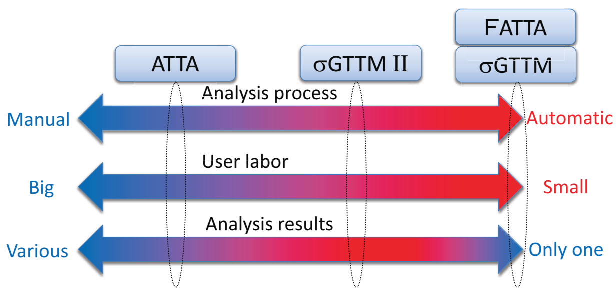

Figure 3. Trade-off between automation of analysis process and variation of analysis results.

3.1.1 ATTA: Automatic Time-Span Tree Analyzer

We extended the original theory of GTTM with a full externalization and parameterization and proposed a machine-executable extension of the GTTM called exGTTM [2]. The externalization includes introducing an algorithm to generate a hierarchical structure of the time-span tree in a mixed top-down and bottom-up manner and the parameterization includes introducing a parameter for controlling the priorities of rules to avoid conflict among the rules as well as parameters for controlling the shape of the hierarchical time-span tree. We implemented the exGTTM on a computer called the ATTA, which can output multiple analysis results by configuring the parameters.

3.1.2 FATTA: Full Automatic Time-Span Tree Analyzer

Although the ATTA has adjustable parameters for controlling the weight or priority of each rule, these parameters have to be set manually. This takes a long time because finding the optimal values of the settings themselves takes a long time. The FATTA can automatically estimate the optimal parameters by introducing a feedback loop from higher-level structures to lower-level structures on the basis of the stability of the time-span tree [23]. The FATTA can output only one analysis result without manual configuration. However, our experimental results showed that the performance of the FATTA is not good enough for grouping structure or time-span tree analyses.

3.1.3 σGTTM

We have developed σGTTM, a system that can detect the local grouping boundaries in GTTM analysis, by combining GTTM with statistical learning [24]. The σGTTM system statistically learns the priority of the GTTM rules from 100 sets of score and grouping structure data analyzed by a musicologist and does this by using a decision tree. Its performance, however, is not good enough because it can construct only one decision tree from 100 data sets and cannot output multiple results.

3.1.4 σGTTM II

The σGTTM II system assumes that a piece of music has multiple interpretations and thus it constructs multiple decision trees (each corresponding to an interpretation) by iteratively clustering the training data and training the decision trees. Experimental results showed that the σGTTM II system outperformed both the ATTA and σGTTM systems [25].

3.2 Manual editor for the GTTM

In some cases the GTTM analyzer may produce an acceptable result that reflects the user's interpretation, but in other cases it may not. A user who wants to change the analysis result according to his or her interpretation can use the GTTM manual editor. This editor has numerous functions that can load and save the analysis results, call the ATTA or σGTTM II analyzer, record the editing history, undo the editing, and autocorrect incorrect structures.

3.3 Implementation on client-server system

Our analyzer is updated frequently, and sometimes it is a little difficult for users to download an updated program. We therefore implement our Interactive GTTM analyzer on a client-server system. The graphic user interface on the client side runs as a Web application written in Java, while the analyzer on the server side runs as a program written in Perl. This enables us to update the analyzer frequently while allowing users to access the most recent version automatically.

4. EXPERIMENTAL RESULTS

GTTM analysis of a piece of music can produce multiple results because the interpretation of a piece of music is not unique. We compared the different analysis results obtained by different musicologists.

4.1 Condition of experiment

A new musicologist who had not been involved in the construction of the GTTM database was asked to manually analyze the 300 scores in the database faithfully with regard to the GTTM. We provided only the 8-bar-long monophonic pieces of music to the musicologist, but she could refer the original score as needed. When analyzing pieces of music, she could not see the analysis results already in GTTM database. She was told to take however much time she needed, and the time needed for analyzing one song ranged from fifteen minutes to six hours.

4.2 Analysis results

Experiments showed that the analysis results for 267 of 300 pieces were the same as the original results in the GTTM database. The remaining 33 pieces had different interpretations, so we added the 33 new analysis results to the GTTM database after they were cross-checked by three other experts.

For those 33 pieces with different interpretations, we found the grouping structure in the database to be the same as the grouping structure obtained by the new musicologist. And for all 33 pieces, in the time-span tree the root branch and branches directly connected to the root branch in the database were the same as the ones in the new musicologist's results.

We also calculated the coincidence of time-spans in both sets of results for those 33 pieces. A time-span tree is a binary tree and each branch of a time-span tree has a time-span. In the ramification of two branches, there is a primary (salient) time-span and secondary (nonsalient) time-span in a parent time-span (Fig. 4). Two time-spans match when the start and end times of the primary and secondary time-spans are the same. We found that 233 of the 2310 time-spans in those 33 pieces of music did not match.

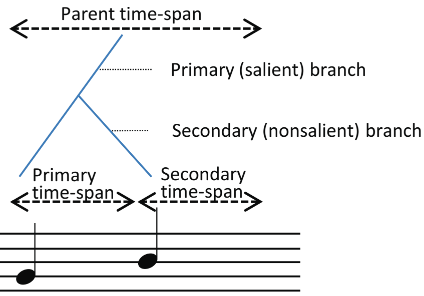

Figure 4. Parent and primary and secondary time-spans.

4.3 An example of analysis

"Fuga C dur" composed by Johann Pachelbel had the most unmatched time-spans when the analysis results in the GTTM database (Fig. 5a) were compared with the analysis results by the new musicologist (Fig. 5b). From another musicologist we got the following comments about different analysis results for this piece of music.

(a) Analysis result in GTTM database
In the analysis result (a), note 2 was interpreted as the start of the subject of the fuga. Note 3 is more salient than note 2 because note 2 is a non-chord tone. Note 5 is the most salient note in the time-span tree of first bar because notes 4 to 7 are a fifth chord and note 5 is a tonic of the chord. The reason that note 2 was interpreted as

the start of the subject of the fuga is uncertain, but a musicologist who is familiar with music before the Baroque era should be able to see that note 2 is the start of the subject of the fuga.

(b) Analysis result by the musicologist

The analysis result (b) was a more simple interpretation than (a) that note 1 is the start of the subject of the fuga. However, it is curious that the trees of second and third beats of the third bar are separated, because both are the fifth chord.

The musicologist who made this comment said that it is difficult to analyze a monophonic piece of music from the contrapuntal piece of music without seeing other parts. Chord information is necessary for GTTM analysis, and a musicologist who is using only a monophonic piece of music has to imagine other parts. This imagining results in multiple interpretations.

5. CONCLUSION

We described the publication of our Interactive GTTM analyzer and the GTTM database. The analyzer and database can be downloaded from the following website:

http://www.gttm.jp/

The GTTM database has the analysis data for the three hundred monophonic music pieces. Actually, the manual editor in our Interactive GTTM analyzer enables one to deal with polyphonic pieces. Although the analyzer itself works only on monophonic pieces, a user can analyze polyphonic pieces by using the analyzers's manual editor to divide polyphonic pieces into monophonic parts. We also attempted to extend the GTTM framework to enable the analysis of polyphonic pieces [23]. We plan to publicize a hundred pairs of polyphonic score and musicologists' analysis results.

Although the 300 pieces in the current GTTM database are only 8 bars long, we also plan to analyse whole pieces of music by using the analyzer's slide bar for zooming piano roll scores and GTTM structures.

6. REFERENCES

[1] M. Hamanaka, K. Hirata, and S. Tojo: "Time-Span Tree Analyzer for Polyphonic Music," *10th International Symposium on Computer Music Multidisciplinary Research* (CMMR2013), October 2013.

[2] M. Hamanaka, K. Hirata, and S. Tojo: "ATTA: Automatic Time-span Tree Analyzer based on Extended GTTM," *Proceedings of the 6th International Conference on Music Information Retrieval* (ISMIR2005), pp. 358–365, September 2005.

[3] F. Lerdahl and R. Jackendoff: *A Generative Theory of Tonal Music*. MIT Press, Cambridge, 1983.

[4] G. Cooper and L. Meyer: *The Rhythmic Structure of Music*. University of Chicago Press, 1960.

[5] E. Narmour: *The Analysis and Cognition of Basic Melodic Structure*. University of Chicago Press, 1990.

[6] D. Temperley: *The Congnition of Basic Musical Structures*. MIT Press, Cambridge, 2001.

[7] M. Hamanaka, K. Hirata, and S. Tojo: Melody morphing method based on GTTM, *Proceedings of the 2008 International Computer Music Conference* (ICMC2008), pp. 155–158, 2008.

[8] N. Todd: "A Model of Expressive Timing in Tonal Music," *Musical Perception*, 3:1, 33–58, 1985.

[9] G. Widmer: "Understanding and Learning Musical Expression," *Proceedings of 1993 International Computer Music Conference* (ICMC1993), pp. 268–275, 1993.

[10] K. Hirata, and R. Hiraga: "Ha-Hi-Hun plays Chopin's Etude," *Working Notes of IJCAI-03 Workshop on Methods for Automatic Music Performance and their Applications in a Public Rendering Contest*, pp. 72–73, 2003.

[11] K. Hirata and S. Matsuda: "Annotated Music for Retrieval, Reproduction, and Sharing," *Proceedings of 2004 International Computer Music Conference* (ICMC2004), pp. 584–587, 2004.

[12] K. Hirata and S. Matsuda: "Interactive Music Summarization based on Generative Theory of Tonal Music," *Journal of New Music Research*, 32:2, 165–177, 2003.

[13] H. Schaffrath: The Essen associative code: A code for folksong analysis. In E. Selfridge-Field (Ed.), *Beyond MIDI: The Handbook of Musical Codes*, Chapter 24, pp. 343–361. MIT Press, Cambridge, 1997.

[14] RISM: International inventory of musical sources. In Series A/II Music manuscripts after 1600, K. G. Saur Verlag, 1997.

[15] J. Riley, C. Hunter, C. Colvard, and A. Berry: Definition of a FRBR-based Metadata Model for the Indiana University Variations3 Project , http://www.dlib.indiana.edu/projects/variations3/docs/v3FRBRreport.pdf, 2007.

[16] M. Goto: Development of the RWC Music Database, *Proceedings of the 18th International Congress on Acoustics* (ICA 2004), pp. I-553–556, April 2004.

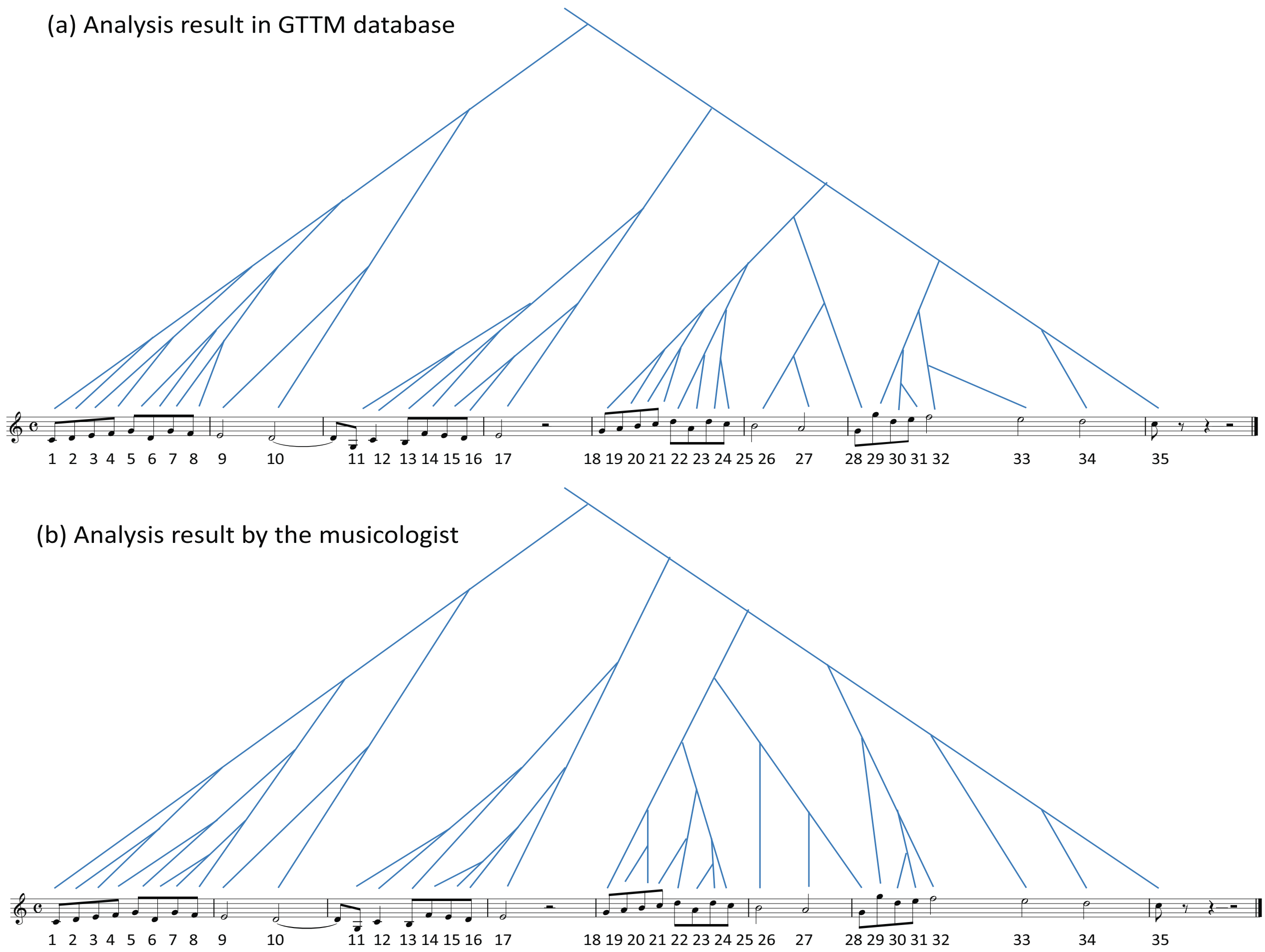

Figure 5. Time-span trees of "Fuga C dur" composed by Johann Pachelbel.

[17] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka: RWC Music Database: Popular, Classical, and Jazz Music Databases, *Proceedings of the 3rd International Conference on Music Information Retrieval* (ISMIR 2002), pp. 287–288, October 2002.

[18] T. Eerola and, P. Toiviainen: The Digital Archive of Finnish Folk Tunes, University of Jyvaskyla, Available online at: http://wwwjyufi/musica/sks, 2004.

[19] C. McKay, D. McEnnis, and I. Fujinaga: A large publicly accessible prototype audio database for music research, *Proceedings of the 7th International Conference on Music Information Retrieval* (ISMIR 2006), pp. 160–164, October 2006.

[20] N. Carlos, L. Alessandro, and A. Celso: The Latin Music Database, *Proceedings of the 9th International Conference on Music Information Retrieval* (ISMIR2008), pp. 451–456, September 2008.

[21] E. Acotto: Toward a formalization of musical relevance, in B. Kokinov, A. Karmiloff-Smith, and J. Nersessian (Eds.), *European Perspectives on Cognitive Science*, New Bulgarian University Press, 2011.

[22] MakeMusic Inc.: Finale, Available online at: http://www.finalemusic.com/, 2014.

[23] M. Hamanaka, K. Hirata, and S. Tojo: FATTA: Full Automatic Time-span Tree Analyzer, *Proceedings of the 2007 International Computer Music Conference* (ICMC2007), Vol. 1, pp. 153–156, August 2007.

[24] Y. Miura, M. Hamanaka, K. Hirata, and S. Tojo: Use of Decision Tree to Detect GTTM Group Boundaries, *Proceedings of the 2009 International Computer Music Conference* (ICMC2009), pp. 125–128, August 2009.

[25] K. Kanamori and M. Hamanaka: Method to Detect GTTM Local Grouping Boundaries based on Clustering and Statistical Learning, *Proceedings of 2014 International Computer Music Conference* (ICMC2014) *joint with the 11th Sound & Music Computing Conference* (SMC2014), September 2014 (accepted).

[26] M. Hamanaka, K. Hirata, and S. Tojo: Time-Span Tree Analyzer for Polyphonic Music, *10th International Symposium on Computer Music Multidisciplinary Research* (CMMR2013), October 2013.