

TOWARDS COMPUTABLE PROCEDURES FOR DERIVING TREE STRUCTURES IN MUSIC: CONTEXT DEPENDENCY IN GTTM AND SCHENKERIAN THEORY

Alan Marsden

Lancaster University, UK
a.marsden@lancaster.ac.uk

Keiji Hirata

Future University Hakodate, Japan
hirata@fun.ac.jp

Satoshi Tojo

Japan Advanced Institute of
Science and Technology
tojo@jaist.ac.jp

ABSTRACT

This paper addresses some issues arising from theories which represent musical structure in trees. The leaves of a tree represent the notes found in the score of a piece of music, while the branches represent the manner in which these notes are an elaboration of simpler underlying structures. The idea of multi-levelled elaboration is a central feature of the *Generative Theory of Tonal Music* (GTTM) of Lerdahl and Jackendoff, and is found also in Schenkerian theory and some other theoretical accounts of musical structure. In previous work we have developed computable procedures for deriving these tree structures from scores, with limited success. In this paper we examine issues arising from these theories, and some of the reasons limiting our previous success. We concentrate in particular on the issue of context dependency, and consider strategies for dealing with this. We stress the need to be explicit about data structures and algorithms to derive those structures. We conjecture that an expectation-based parser with look-ahead is likely to be most successful.

1. BACKGROUND

It is common to regard the structure of a piece of music as in some way hierarchical. Heinrich Schenker [1] was not the first to propose the idea that a piece of music contains different levels of elaboration or reduction, but his influence has been so great as to mean that ‘Schenkerian’ is almost a synonym for hierarchical in music theory. The later *Generative Theory of Tonal Music* (GTTM), by Lerdahl & Jackendoff [2], explains musical structure as explicitly hierarchical and tree-structured, borrowing concepts from formal linguistics.

More recent work also uses trees to represent musical structure. One well known, and again explicitly linguistic-inspired, example is Steedman’s chord grammar [3, 4] which represents the structure of a complex chord sequence as a tree showing the derivation of the sequence from a simple model such as a twelve-bar blues. Rizo has used trees as a basis for a model of melodic similarity [5].

Recent theories such as those of Steedman and Rizo are defined in formal terms, allowing an analysis to be systematically derived from a sequence of chords or notes. Schenker was writing before the birth of formal cognitive science, and did not express his theory in this kind of systematic fashion. (Indeed, it is clear that it was never his intention to take music theory in this direction—that is the result of appropriations by scholars of later generations [6].) Lerdahl & Jackendoff, on the other hand, were writing on the explicit basis of theories of linguistic grammar and also took account of some of the early work in musical computing (e.g., [7]). Their theory is accordingly expressed in more formal terms, but still without the degree of precision required for derivation of analyses from a score without expert musical knowledge. Lerdahl & Jackendoff are quite explicit about this: ‘our theory cannot provide a computable procedure for determining musical analyses’ [2, p. 55]. Yet earlier they ‘conceive of [their] theory as being in principle testable by the usual scientific standards’ [p. 5]. A theory is only testable if it can make precise predictions, and the only logical resolution of these two statements is that Lerdahl & Jackendoff considered that suitable extension of the theory would produce a computable procedure for determining musical analyses.

Over the past decade, the authors have developed computer software to derive analyses in accordance with Schenkerian theory [8] and GTTM [9]. The results have been only partially successful. The ATTA software of Hamanaka, Hirata and Tojo requires the user to adjust parameters in order to arrive at acceptable analyses in accordance with GTTM. Marsden’s Schenkerian analysis software can only make analyses for short extracts of music, and the results only partially match those of experts.

In view of this limited success, and the enduring popularity of the idea of reduction in musical computing, we believe it is worth stepping back to reconsider some of the fundamental issues concerning the derivation of tree structures in music. In particular, we aim to consider some of the details around context dependency which complicate the formulation of an effective computable procedure to automatically derive trees from the notes in the score of a piece of music.

Copyright: © 2013 Alan Marsden et al. This is an open-access article distributed under the terms of the [Creative Commons Attribution License 3.0 Unported](http://creativecommons.org/licenses/by/3.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

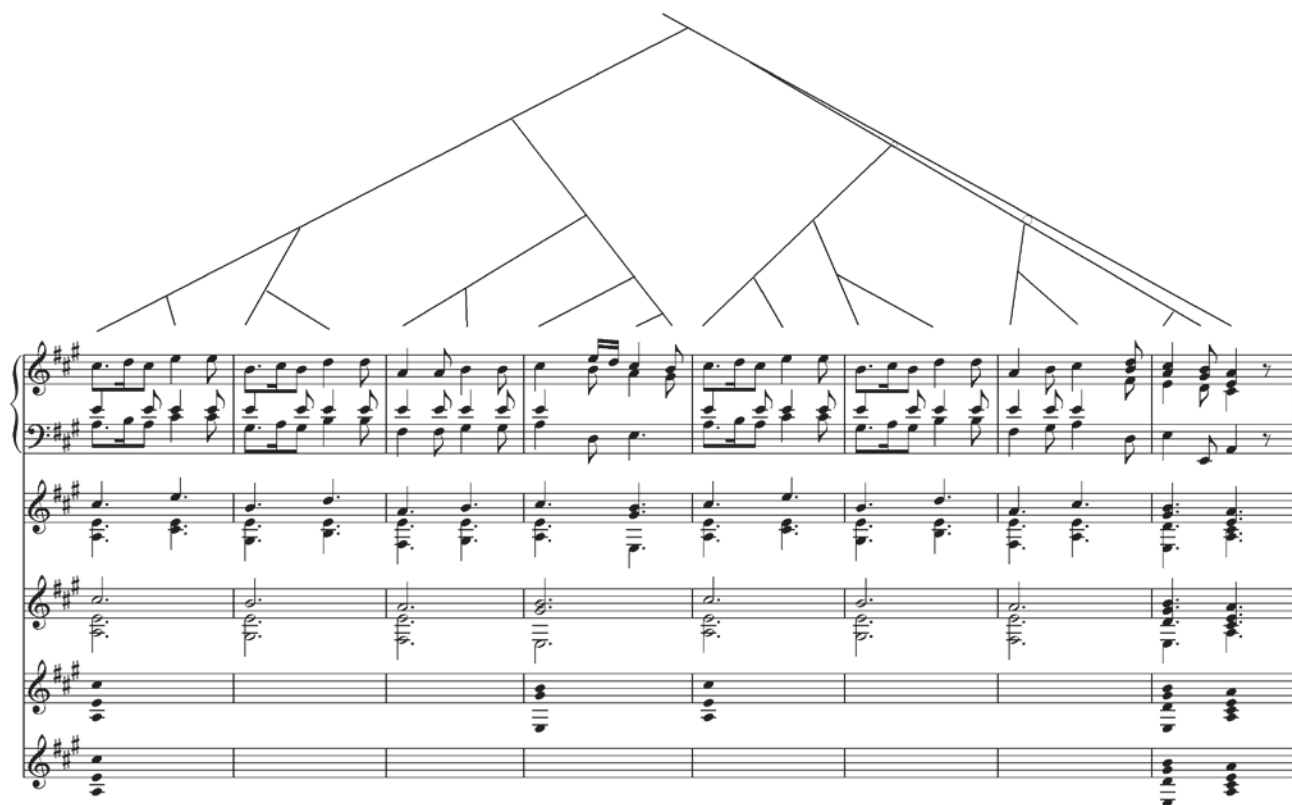


Figure 1. Analysis of the theme from the first movement of Mozart's piano sonata in A major, K.331.

2. TREE REPRESENTATIONS

Formally, a tree is a connected graph of nodes and arcs in which there are no cycles. Some additional properties are considered to be essential when representing music, however. Firstly, arcs have a direction, connecting ‘parents’ to ‘children’. The parent is the ‘reduction’ of the children, and the children the ‘elaboration’ of the parent. Secondly, child nodes in a musical-structure tree have an explicit order: ‘left’ children occur before ‘right’ children. In principle there is no restriction to the number of children a parent may have, and it is not uncommon for trees representing musical structure to have parents with three or more children. The simplest trees, however, have no more than two children per parent (called *binary trees*), and it is common to restrict discussion and the definition of procedures to this case. Nothing is lost by this, because it is always possible to convert any finite tree to a binary tree which represents exactly the same information, and to convert it back again to the original tree

In GTTM, Time-span reduction and prolongational reduction are explicitly represented in trees. They are often binary, but the theory does allow cases of parents with more than two children. Schenkerian analyses are notated in music-notation-like ‘graphs’ using noteheads and slurs rather than trees, but parent-child relations can be derived from these, and equivalent tree structures generated [10].

Figure 1 shows Lerdahl & Jackendoff’s time-span reduction of the first eight bars of the theme from the first movement of Mozart’s piano sonata in A major, K.331. (For clarity, not all of the lowest levels of branching are shown in the figure.) Schenker’s analysis of this theme is somewhat different, but for present purposes we can

point out that a Schenkerian analysis would look rather like the notation in the four lower staves, but with the vertical order reversed and the addition of slurs joining notes into groups more or less in accordance with the grouping shown by the corresponding level of branching in the tree.

2.1 Trees and Cognition

Schenker believed that his analyses showed the ‘background and middleground’ of a piece of music, which were the ‘indispensable prerequisites to a musical work of art’ [1, p. 3–4]. Background and middleground were, for Schenker, the genesis of a work not literally in the sense of the sequence of events (real or mental) which led to its composition, but in a metaphysical sense, constituting something of the reality of the piece.

Lerdahl & Jackendoff state the goal of theory to be a ‘formal description of the musical intuitions of a listener who is experienced in a musical idiom’ [2, p.1] and later make clear that they are concerned with ‘the final state of his understanding’ and not the mental processes which lead to this state.

Both Schenker and Lerdahl & Jackendoff, therefore, consider their reductions to correspond to a cognitive conception of the piece, but neither is directly concerned with how that conception is created in the act of listening.

2.2 Musical Grammar

On the other hand, both Schenker and Lerdahl & Jackendoff aim to show a systematic relationship between the notes of the score and the reduction. Schenker writes of musical ‘laws’, though he does not explicate



Figure 2. Recomposition of Figure 1 to place a copy of bar 4 at the beginning.

them in a precise fashion. (Instead, one is given the strong impression that only geniuses have true understanding of the laws!) Lerdahl & Jackendoff, by contrast, give a large set of rules to relate notes to analyses. However, as frequently pointed out, some of these are irregular rules in that they state only a ‘preferred’ relation between notes and structures, which need not hold if other rules imply a different relation. How conflicts between preference rules are to be resolved is not specified in the theory.

It is our conviction that music theories such as GTTM and Schenkerian theory form a useful ground for building computational systems which are capable of automatically deriving the structure of a piece of music from the notes in the score. Furthermore, we believe that such derivation of structure is essential for some effective musical processing in such tasks as finding similarity or segmenting pieces of music. In the following, we examine some difficulties in employing GTTM and Schenkerian theory as such a basis for computational structure-finding systems.

3. CONTEXT DEPENDENCY

Schenker was clear that the *Ursatz* (the simplest structure at the top level of every great piece of music) governed every aspect of the structure of a piece of music. The details of how a passage is reduced, therefore, depend in part on where that passage comes in the *Ursatz*. A fundamental problem of Schenkerian analysis is that one cannot know what the *Ursatz* is, and how it relates to the details of the piece, until one has analysed the structure: one needs to know the context to properly analyse the structure, but one cannot know what the context is before the structure is analysed. For example, it is common to find the same passage of music analysed in two different ways in a Schenkerian analysis according to where it comes in the piece. The typical case is for the melody of a passage to be analysed as an elaboration of the third or fifth degree of the scale when it occurs early in the piece, but for the same melody to be analysed as the descending 3-2-1 or 5-4-3-2-1 of the *Ursatz* late in the piece.

Marsden’s Schenkerian analysis software [8] overcomes the difficulty of not knowing the location of the

Ursatz in advance by effectively generating many possible analyses of the structure, then selecting those which contain an *Ursatz*, and finally selecting the one which appears best. This is an extremely costly procedure in computational terms, and cannot form the basis of a practical system.

GTTM similarly contains many instances of context-dependency. The preference rules for ‘cadential retention’ and ‘structural beginning’, TSRPR 7 and 8, cause reductions to depend on the grouping structure not just for the time-span where the reduction takes place, but for the enclosing time-span(s) also. In other words knowledge of higher-level structure is required before the lower level structure can be determined. In this case the structures are in different components of GTTM (grouping and time-span reduction), but a preference rule for grouping structure, GPR 7, completes the circle by stating that grouping structures are preferred which result in more stable time-span reductions.

Lerdahl & Jackendoff point out the importance of such context dependency in discussion of bar 4 (measure 4) of the Mozart theme in Figure 1 [2, p. 118–120, 134–35, 167]. Out of context, the opening chord of the bar would be considered the ‘head’ to which the entire bar reduces, because it is on the strong beat (TSRPR 1) and because it is closer to the tonic (TSRPR 2; in fact it is the tonic!). The correct reduction, which makes sense of the phrase, is instead to take the dominant chord at the end of the bar as the head, as indicated in Figure 1.

3.1 Recomposition of Contexts

Figure 2 illustrates the context dependency in the reduction of bar 4 of the Mozart theme. The theme has been rewritten to start with a copy of bar 4 and a new bar 2 to retain the overall pattern of descending sequence in the first two bars. Here it is clear that the pattern of notes in bar 4 is reduced in different ways according to the context, as shown by the tree structure in Figure 2. A Schenkerian graph of Figure 2 would also show a difference between the reduction of bar 4 in Figure 2 and the copy of it in bar 1 of that figure. In the reduction of bar 4 a slur which has its beginning earlier in the piece would end on



Figure 3. Recomposition of Figure 1 to make five-bar phrases.



Figure 4. Recomposition of Figure 3 to echo cadence.



Figure 5. Recomposition of Figure 3 to recreate four-bar phrase.



Figure 6. Recomposition of Figure 2 to prevent cadence at bar 4.

the melody note B while in bar 1 a slur would begin on the first C sharp and end somewhere beyond the end of bar 1. (In making this claim, we follow the procedure of music theorists who rely on their own intuition about the structure of a piece of music, tested by repeated listening and introspection. We furthermore assume that other listeners will have the same intuitions as ours. We judge that for our present purposes the cost of proper scientific listening tests is not warranted, but we would be interested to hear if other listeners do not share our intuitions.)

In the case of Figure 2, the new context for the pattern of bar 4 is evident from the fact that it occurs at the beginning. However, it is not only this which can cause a different reduction of this bar. Figure 3 shows a different recomposition of the Mozart theme to make the phrases five bars long. Here bar 4 is reduced differently because a new bar follows which takes the role of cadence. (Some might prefer the reduction of bar 4 to be connected to the branch from bar 5 rather than the one from bar 3, but this does not change the assignment of the tonic chord at the beginning of bar 4 as head for that bar rather than the dominant.)

From Figure 3, one might conclude that so long as the pattern of bar 4 does not occur at the end of a phrase, it should be reduced to tonic harmony, but this is contradicted by the example in Figure 4, which replaces the new cadential bar 5 by a copy of bar 4. Here the new bar 5 sounds like an echo of the cadence in bar 4.

Figures 5 and 6 illustrate the affect of other contexts for bar 4. Figure 5 illustrates that the possibility of splitting ten bars into two phrases of five bars each does not necessarily lead to a tree structure congruent with a division into two phrases of five bars. Here the new bar 5, while having the same outline of I-V and C sharp to B in the melody, groups with the beginning of the next phrase, partly by virtue of the similarity of rhythm. In Figure 6, which recomposes the music in Figure 2, bar 4 is prevented from acting as a cadence not by the insertion of a stronger cadence (as in Figure 3), but by a continuation which causes it to sound once again like a beginning.

To illustrate the significance of the difference in these structural analyses, imagine a software system designed to separate music into segments, and to report the degree to which a segment will sound finished or unfinished.

Such a system should segment Figures 1 and 2 into bars 1–4 and 5–8, Figures 3 and 4 into bars 1–5 and 6–10, Figure 5 into bars 1–4 and 6–10, and Figure 6 into bars 1–3 and 4–7. It should report that the first segments made up of bars 1–4 or 1–5 will sound finished but less final than the second segments 5–8 or 6–10, and that the first segment in Figure 6, bars 1–3, will not sound finished. All of these could be concluded directly from the graphs by taking the highest-level branching to indicate the segmentation, the presence of a retained cadence to indicate strong finality, and the presence of right-branching on the right-most branch (as would be the case in bar 3 of Figure 6) to indicate sounding unfinished.

3.2 Strategies for Context Dependency

3.2.1 Separation of bottom-up and top-down

GTTM includes two kinds of tree: time-span reduction and prolongational reduction. Time-span reduction is characterised as concerning ‘relative stability within rhythmic units’, and prolongational reduction ‘relative stability expressed in terms of continuity and progression’ [2, p. 123]. What this means precisely is not entirely clear, especially since rhythmic units are partially defined by time-span reduction in view of the interdependence between time-span reduction and grouping. Furthermore, the concepts of ‘cadential retention’ and ‘structural beginning’ clearly concern continuity and progression to some degree.

Another distinction between time-span reduction and prolongational reduction, not explicitly stated by Lerdahl & Jackendoff but clearly implied in their presentation, is that time-span reductions are made mostly bottom-up while prolongational reductions are made top-down. Perhaps a strategy to deal with context dependency is to make this bottom-up/top-down distinction absolute and revise time-span reduction to disregard top-down rules such as cadential retention and structural beginnings. This would reduce the pattern in bar 4 of the examples above always using right-branching and yielding tonic harmony as the head. A top-down process like prolongational reduction would then modify the tree to reflect context dependencies, for example replacing right branching by left branching at cadences.

Marsden’s Schenkerian-analysis software [8] also operates in a two-step bottom-up then top-down process. It uses a version of the CYK parsing algorithm which fills a table with information about possible parses in a bottom-up process (the Schenkerian-analysis software also collects information about possible *Ursatz* membership) and then uses this information to build a parse top-down.

3.2.2 Expectation-based parsing

Most of our listening to music is to pieces we have heard before, or if not, at least to pieces similar to others we have heard before. Perhaps a reduction mechanism can take two inputs: the notes of the score, and a sequence of expectations based on the last time the piece was heard

and the structure derived from that hearing, whether expressed in the fashion of GTTM as a prolongational and time-span tree, or in the manner of Schenkerian theory, or some other manner (e.g., expectation expressed in a numerical value [11]). If the piece has not been heard before, expectations can be generated on the basis of memories of similar pieces or a style [12]. Even if one’s memory is not sufficiently accurate to expect what the next note will be, a trace of a previously derived tree structure might remain, or melodic expectation might be generated based on a familiarity with a style. Thus, on arriving at bar 4 in the Mozart theme, the listener will expect that the next bar will be a return to the opening, so bar 4 must function as a cadence.

Indeed in every case in the examples given above, the correct parsing of a structural unit is not clear until the next unit has begun. It would seem that parsing takes place after the event rather than while the music is being heard, but it is not clear how long the delay is. (Clearly limits to short-time and working memory will have an impact on this.) Possibly the delay is long enough for a rough parse to be made for an entire phrase before the detail of a reduction is completed.

Top-down information is known to influence visual object recognition, and experimental evidence suggests that high-speed processing of low-spatial-frequency information is instrumental in this process [13]. Perhaps similar low-bandwidth information or approximations in listening to music provide the same kind of top-down control. For example, it is possible that the listener rapidly extracts the main harmonies from a passage, and uses these to generate an outline tree to capture the I-V, I-V-I structure of the Mozart theme. This outline tree is then filled in with the rest of the detail of the reduction.

3.2.3 Category labels

Similar phenomena of context dependency occur in language, but we are not aware of any musical examples which have the force of ‘garden-path sentences’ which require the reader or hearer to undo an existing parse and re-parse the sentence for it to make sense. [14] shows how a Definite Clause Grammar can be used to parse the sentence ‘That man that whistles tunes pianos.’ The word ‘whistles’ is initially parsed as a transitive word with ‘tunes’ taken to be a noun and its object. The occurrence of the word ‘pianos’, however, causes the parsing to backtrack and then take ‘whistles’ to be intransitive and ‘tunes’ as a verb. Techniques therefore exist in natural language processing to cope with similar context dependency, but they cannot be naively transferred to music. For example, [15] reports that a backtracking parser, without additional features to guide it towards the correct parse, frequently failed to find a correct parse for jazz chord sequences within a reasonable time.

A characteristic of linguistic grammars is that they associate labels, such as ‘noun phrase’, with internal nodes of a parse tree. This allows for more efficient and reliable parsing because it disambiguates words or sequences of

words which can have different functions, such as the word ‘tunes’ in the garden-path sentence mentioned above. In the incorrect parsing it is categorised as a noun, whereas it should be a verb.

Perhaps the use of category labels in musical reduction trees could similarly disambiguate cases which behave differently in different contexts. Each putative head in a reduction, for example, might have one of the categories *b*, *m*, or *c* attached, for ‘beginning’, ‘middle’ or ‘close’. The pattern in bar 4, then, could produce the alternative reductions $I(b)$ and $V(c)$, where *I* and *V* stand as shorthand for the tonic at the beginning of the bar and the dominant at the end respectively. The grammar for categories could then be as follows:

- $b \rightarrow b m$
- $c \rightarrow m c$
- $b b \rightarrow b c b$
- $c \rightarrow b c$
- $m c \rightarrow m c c$ [only when the second *c* is a copy of the first]
- $b \rightarrow b b$ [only if no other parsing is possible]

The bars of the original theme (Figure 1) would be initially categorised as *b*, *m*, *m*, *b/c*, *b*, *m*, *m*, *c*. The alternative reduction for bar 4 with the label *c* will be selected in parsing because there is no rule to accept *m b* and selecting *c* allows the non-preferred sequence *b b* to be avoided.

The grammar also leads to the correct reductions for the other examples to be selected. Figure 6, for example, has initial categories *b/c*, *m*, *m*, *b/c*, *m*, *m*, *c*. The reduction with category *b* will be chosen for bar 4 because there is no following *b* which would allow *c* here to be absorbed by the third rule, and this bar is not copied by the final *c*.

Grammars using categories have been applied to music (e.g., [4]) but even for chord sequences, which are simpler than collections of notes, this alone does not lead to a successful analysis system. [14] shows greater success in analysing chord sequences when category labels assigned in a probabilistic fashion so that the label most likely to lead to a correct parse is used first, or at least early in backtracking.

3.3 A Tension-Relaxation Grammar

Category labels in language function not only to indicate syntactic position but also function. We suggest that in music this function might relate to the commonly used concepts of tension and relaxation. Lerdahl & Jackendoff relate prolongational reduction to the sense of tension and relaxation in a piece of music [2, 16]. Schenker does not use the same language, but his metaphor of a piece of music as a living organisms is not so far from these ideas. For him pieces grow and exhibit intention, cause and effect. A grammar of tension and relaxation, if such a thing is possible, could provide a basis for category labels, for expectation, and for outline parsing. The grammar might look something like this (where *S* is a complete ‘sentence’, *T* tension and *R* relaxation):

- $S \rightarrow S S$
- $S \rightarrow T R$

$$T \rightarrow T S$$

$$R \rightarrow S R$$

Other rules would indicate how tension and relaxation were related to, for example, harmonies:

$$T \rightarrow I V$$

$$R \rightarrow V I$$

$$R \rightarrow IV V I$$

$$R \rightarrow ii V I$$

etc.

Tension and relaxation could be derived from rhythmic characteristics of the music also, or from dynamics and timing. The function of the grammar is precisely to take information from whatever source seems useful and use it to guide derivation of structure and meaning from the music. Studies of performers’ bodily movements while playing have shown that even these convey information about tension and structure to an audience (e.g., [17]). Even bodily movements might therefore provide input to the tension-relaxation grammar.

4. CONCLUSIONS

This discussion has focused on the general principles of tree structures in music and their derivation. Research which draws directly and only from music theory is unlikely to progress further than the authors’ earlier work because Schenkerian theory, GTTM and the like are not expressed with the degree of precision required for computational implementation. They also lack empirical validation. Further progress will depend on derivation from examples (preferably large sets of them) and other empirical data.

Data from listening experiments is costly to obtain, and the structural intuitions which Schenker and Lerdahl & Jackendoff believed their theories revealed do not correspond to overt measurable behaviours. Experiments which test the match of reductions to tunes ask listeners to perform an unfamiliar task without any clear relation to other musical behaviours [18, 19]. The results are therefore of dubious validity. In our view a more solid basis for empirical data relevant to tree structures in music comes from four sources:

1. *Existing analyses by musical experts.* There are not many published examples of analyses according to GTTM, but the second and third authors have a test set of melodies analysed by experts. A sizeable quantity of published Schenkerian analyses exist in journals and textbooks.
2. *Variations.* In many cases, a theme and variation share a common underlying structure. What a theme and set of variations has in common therefore provides information about the proper tree-structure representation of the theme or variation.
3. *Music similarity data.* In the same way, melodic similarity, on which a quantity of data is emerging from MIR research, provides suggestive information about underlying tree structures. Similar pieces of music often share similar structures.

4. *Operational effectiveness in music processing.* Music-processing tasks often require structural information. (Examples include performance rendering, segmentation, and summarisation.) We conjecture that embedding systems for deriving tree structures from pieces of music within software to perform such tasks will provide empirical validation of the structures derived: if the task is performed well, the structure-derivation is likely to be correct.

We do not wish to discount the value of sophisticated music theory to music computing. On the contrary we believe that it has much to offer but that successful employment of ideas from music theory will also require the application of concepts and procedures from modern computational science. In particular, we believe that employment of the ideas of category labels, expectation, look-ahead and initial tracing of an over-arching structure of tension and relaxation will be useful for future progress.

It is common in computing to separate data structures from algorithms, and we suspect that music theory would benefit from a similar separation. Both GTTM and Schenkerian theory, in their textual expositions, describe the data structures in which musical structure is embodied (despite the fact that many of the rules of GTTM are expressed in a quasi-procedural fashion, using formulations such as ‘prefer a reduction which ...’). As pointed out above, the algorithmic part—how to derive the structures from a score—is not made explicit but remains implicit in the theorists’ examples. Computational linguistics, by contrast, makes a clear distinction between grammars and the parsers which use grammars, employing processes such as backtracking, decomposition/recomposition, and expectation, as we have seen. We believe that advances in the theory of musical structure will depend on similar clarity about data structures and explicit algorithms. It is our conjecture that in the case of deriving tree structures from musical scores, some kind of expectation-based parser, coupled with a look-ahead buffer, is most likely to be successful.

Acknowledgments

The authors are grateful to the anonymous reviewers’ valuable comments that improved the manuscript. This research was supported by a Short-Term Invitation Fellowship in December 2012 funded by the Japan Society for the Promotion of Science.

5. REFERENCES

- [1] H. Schenker, *Der freie Satz*. Universal Edition, 1935. Published in English as *Free Composition*, translated and edited by E. Oster, Longman, 1979.
- [2] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*. MIT Press, 1983.
- [3] M. Steedman, “A generative grammar for jazz chord sequences,” *Music Perception*, vol. 2, no. 1, pp. 52–77, 1984.
- [4] M. Steedman, “The blues and the abstract truth: music and mental models,” in A. Garnham & J. Oakhill (eds.), *Mental Models in Cognitive Science*. Psychology Press, pp. 305–318, 1996.
- [5] D. Rizo, Symbolic music comparison with tree data structures, PhD thesis, University of Alicante, 2010.
- [6] N. Cook, *The Schenker Project: Culture, Race, and Music Theory in Fin-de-siècle Vienna*. Oxford University Press, 2007.
- [7] J. Tenney and L. Polansky, “Temporal gestalt perception in music,” *Journal of Music Theory*, vol. 24, no. 2, 205–241.
- [8] A. Marsden, “Schenkerian analysis by computer: a proof of concept,” in *Journal of New Music Research*, vol. 39, no. 3, pp. 269–289, 2010.
- [9] M. Hamanaka, K. Hirata, and S. Tojo, “Implementing ‘A Generative Theory of Tonal Music’,” *Journal of New Music Research*, vol. 35, no. 4, pp. 249–277, 2006.
- [10] A. Marsden, “Generative Structural Representation of Tonal Music,” *Journal of New Music Research*, vol. 34, no. 4, pp. 409–428, 2005.
- [11] M.T. Pearce and G.A. Wiggins, “Expectation in melody: The influence of context and learning,” *Music Perception*, vol. 23, no. 5, pp. 377–405, 2006.
- [12] W.E. Caplin, *Classical Form: A Theory of Formal Functions for the Music of Haydn, Mozart, and Beethoven*, Oxford University Press, 1998.
- [13] M. Bar, K.S. Kassam, A.S. Ghuman, J. Boshyan, A.M. Schmidt, A.M. Dale, M.S. Hamalainen, K. Marinkovic, D.L. Schacter, B.R. Rosen and E. Halgren, “Top-down facilitation of visual recognition,” *Proceedings of the National Academy of Science*, vol. 103, no. 2, pp. 449–454, 2006.
- [14] F.C.N. Pereira and D.H.D. Warren, “Definite Clause Grammars for Language Analysis—A Survey of the Formalism and a Comparison with Augmented Transition Networks,” *Artificial Intelligence* vol. 13, pp. 231–278, 1980.
- [15] M. Granroth-Wilding and M. Steedman, “Statistical parsing for harmonic analysis of jazz chord sequences,” *Proc. International Computer Music Conference (ICMC)*, Ljubljana, 2012, pp. 478–485.
- [16] F. Lerdahl, *Tonal Pitch Space*, Oxford University Press, 2001.
- [17] M.M. Wanderley, B.W. Vines, N. Middleton, C. McKay, W. Hatch, “The musical significance of clarinetists’ ancillary gestures: An exploration of the

field,” *Journal of New Music Research*, vol. 34, no. 1, pp. 97–113, 2005.

- [18] Y. Oura and G. Hatano, “Identifying melodies from reduced pitch patterns,” *Psychologica Belgica*, vol. 31, no. 2, pp. 217–237, 1991.
- [19] N. Dibben, “The cognitive reality of hierarchic structure in tonal and atonal music,” *Music Perception*, vol. 12, no. 1, pp. 1–25, 1994.