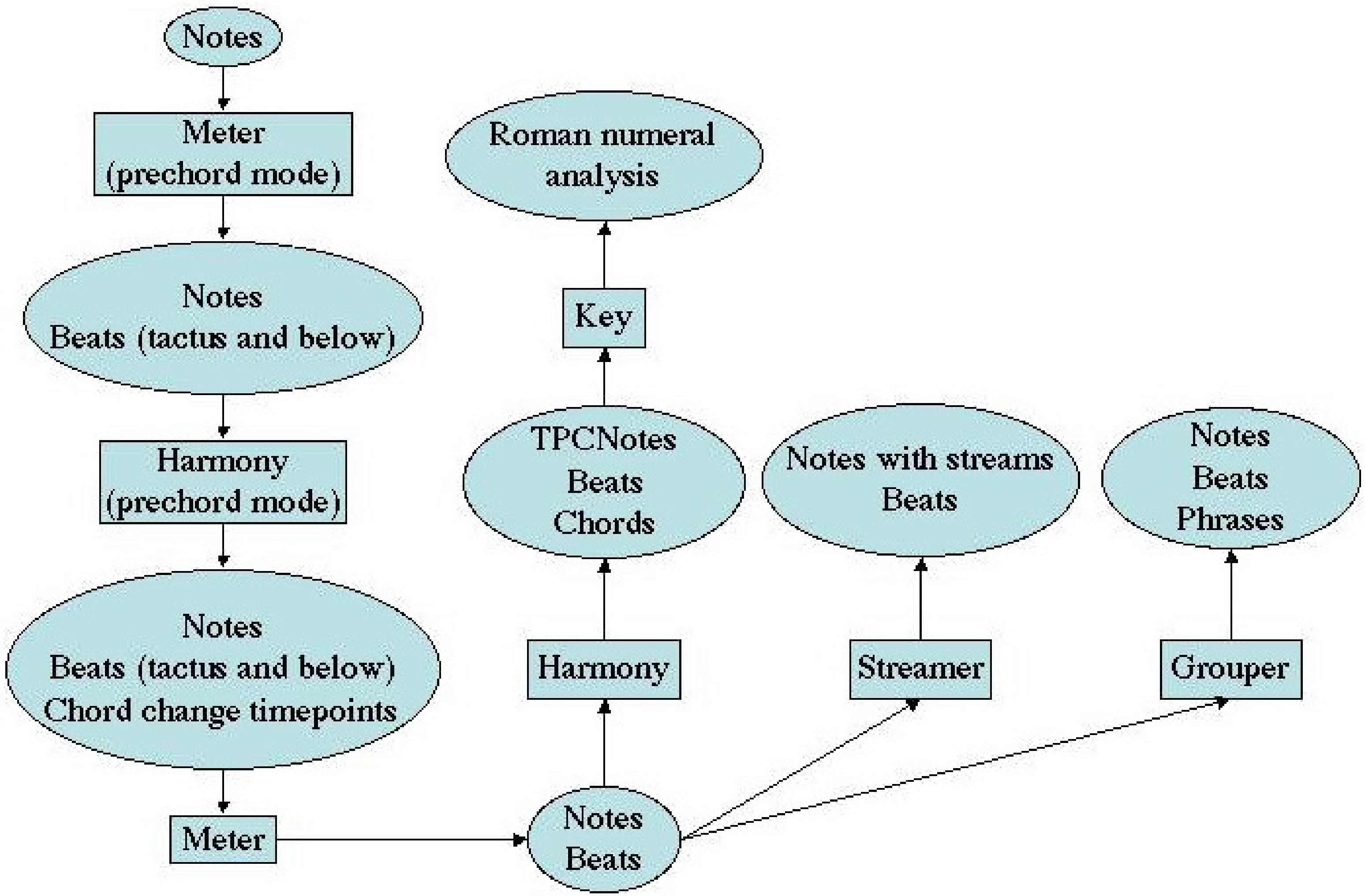
David Temperley's *The Cognition of Basic Musical Structures* (2001, MIT Press)

David Meredith
Department of Computing,
City University, London.
dave@titanmusic.com
www.titanmusic.com

Science and Music Seminar, Faculty of Music, University of Cambridge

Tuesday, 18 March 2003.

1. Overview of Temperley's theory



1. Overview of Temperley's theory

1. In this seminar we're going to focus on issues arising from Temperley's (2001) book, *The Cognition of Basic Musical Structures*.
2. I imagine that you all have at least some familiarity with Temperley's work and many of you may have interesting things to say about it, so I'm only going to speak for about 15 minutes or so and then we can either have an open discussion or we can systematically go around the table so that everyone has a chance to have their say.
3. I think it might be useful to begin with a 'bird's eye' view of the complete model—something which is missing from both Temperley's book and my review.
4. As we probably all know, Temperley's theory consists of six 'preference rule systems' consisting of well-formedness rules and preference rules of the type used in Lerdahl and Jackendoff's (1983) *Generative Theory of Tonal Music*.
5. Each of these preference rule systems takes a representation of a musical passage as input and generates as output a structure that is supposed to correctly describe certain aspects of how the musical passage is interpreted by an expert listener.
6. Temperley provides models of six aspects of musical structure:
   (a) metrical structure;
   (b) phrasing;
   (c) contrapuntal structure (voice-leading, streaming);
   (d) pitch-spelling;
   (e) harmonic structure; and
   (f) key structure.
7. With some help from Daniel Sleator, Temperley has implemented these six preference rule systems in five computer programs, the pitch-spelling and harmonic structure theories being combined into one program.
8. This diagram here shows the recommended way of using these five programs together to generate a complete analysis.
9. The input to the system is in the form of a 'note-list' or piano-roll which gives the onset time, offset time and MIDI pitch number of each note.
10. You have to generate a metrical structure before you can run any of the other programs on the input.
11. However, harmonic structure can have quite a strong effect on metrical structure, so Temperley (2001, p. 46–47) suggests that the metrical structure be generated by a two-pass method.

12. This involves first running the meter program on the input note-list in what Temperley calls ‘prechord’ mode. This generates an output that contains the input notelist together with a description of the metrical structure at the tactus level and below.

13. This is then given as input to the Harmony program, again running in ‘prechord’ mode, which generates an output containing the notelist and beatlist given to it as input together with a list of ‘Prechord’ statements that simply indicate where the chord changes occur.

14. The output of the Harmony program can then be fed back into the Meter program, running in full mode this time, which uses the knowledge of where chord changes occur to generate a more robust prediction of the complete metrical structure.

15. The output of the Meter program containing the notelist and the beatlist for five levels of the metrical structure can then be used as input to the Harmony, Streamer and Grouper programs, as shown here.

16. When it’s given as input to the Harmony program, an output is generated which contains,

    (a) TPCNote statements giving the tonal pitch class of each note (the TPC being the pitch-name of a note with the octave designation omitted); and

    (b) Chord statements that give the root (as a TPC) for each of the smallest metrical segments in the passage.

17. This output can then be given as input to the Key program which can be used to compute a roman numeral analysis.

18. The output of the Meter program can also be given as input to the Streamer program which tells us the stream or voice to which each note belongs.

19. Finally, the output of the Meter program can be given as input to the Grouper program which predicts where the phrase boundaries will be perceived to be.

20. The Grouper program can only accept monophonic input.

21. However, if you use the Streamer program to generate a number of monophonic parts, each of these parts could be given as input to the Grouper program which could then be used to compute a separate phrase structure for each part.

2. Using the theory to model listening, composition, performance and style

- Temperley and Sleator's programs scan the music from left to right, using dynamic programming to find the analysis that best satisfies the preference rules.
- *Ambiguity:* Two or more best analyses at a given point in the music.
- *Revision:* The best analysis at some point in the music does not form part of the best analysis at some later point.
- *Expectation:* The most expected events are those that will lead to an analysis that best satisfies the preference rules.
- *Style:* A passage is in the style defined by a set of preference rules if the analysis that best satisfies the preference rules achieves a score that is not too high (boring) and not too low (incomprehensible).
- *Composition:* Choices guided by goal to produce piece that optimally satisfies preference rules.
- *Performance:* Temporal and dynamic expression geared towards conveying structure in accordance with analysis that best satisfies the preference rules.

2. Using the theory to model listening, composition, performance and style

1. It's well known that Lerdahl and Jackendoff's GTTM models the final state of the listener's understanding. However, Temperley goes a long way towards showing how the preference rule approach can be used to model certain aspects of the real-time listening process—in particular, ambiguity, revision and expectation.
2. The search for an analysis that best satisfies some given set of preference rules is an example of what is called an *optimisation problem* in computer science.
3. In their computer implementations, Temperley and Sleator solve this problem using the technique of *dynamic programming*.
4. Each of the programs scans the music to be analysed from the beginning to the end, keeping track of a set of 'best-so-far' analyses as it proceeds.
5. Temperley proposes that ambiguity arises when there are two or more best analyses at a given point in the music.
6. He also points out that the analysis at a given point in the music that best satisfies the preference rules might not form part of the analysis that best satisfies the rules at some later point in the music.
7. This seems to model the situation where a listener revises his or her interpretation of something heard earlier in the light of something heard later on.
8. Temperley also proposes that while one is listening to a piece, one always most expects those events that best satisfy the preference rules.
9. When combined with the dynamic programming technique, the preference rule approach therefore seems to be able to explain certain diachronic aspects of the listening process.
10. Temperley also shows how the preference rule approach can be used to model aspects of musical style, composition and performance.
11. When a program implementing one of the preference rule systems is used to analyse a passage, it returns not only the 'most preferred' analysis but also a numerical score indicating how well this analysis satisfies the preference rules (Temperley, 2001, p. 293).
12. If the most preferred analysis of a passage has a very low score, this indicates that the passage cannot be interpreted in terms of the preference rule system.
13. On the other hand, a passage that achieves a very high score when analysed by one of Temperley's models, would, in general, be unacceptably boring.
14. Temperley proposes that between these extremes lies a (fuzzy) 'range of acceptability' within which pieces are considered to be in the 'style' implicitly defined by the preference rule system.

15. Temperley also proposes that preference rules can be viewed as constraints on compositional practice, the composer's choices being guided by the goal to produce a piece that 'optimally' satisfies the preference rules governing the musical style in which the composer is working (Temperley, 2001, p. 298).

16. He also suggests ways in which preference rules can be used to explain performance practice.

17. For example, he suggests that performers slow down at the ends of phrases because this lengthens the gap in which the phrase boundary should be heard without disrupting the metric structure, thus satisfying the first phrase-structure preference rule (the Proximity rule) (Lerdahl and Jackendoff's (1983) GPR 2) while also satisfying satisfying the third metrical preference rule, the Regularity Rule.

3. Possible topics for further discussion

1. Is it appropriate to use MIDI as input to a theory of music cognition?
2. Is it appropriate to test a computational model of music cognition by comparing the output of the model with notated scores?
3. Are motivic analysis, harmonic and tonal analysis simply ‘a search for a parsimonious encoding of musical input’ (Temperley, 2001, p. 238)?
4. Is dynamic programming a feasible model of music perception?
5. Is it valid to describe a cognitive process at the computational level ‘without worrying about how it may be instantiated neurologically’ (Temperley, 2001, p. 5)?

3. Possible topics for further discussion

1. Finally, I just thought I'd suggest some possible topics for further discussion just in case the conversation runs dry.
2. First of all, I suggest in my review that using what is essentially MIDI files as input to a theory of music cognition may be problematic because a MIDI file does not seem to be a good representation of the information that is available to the listener at any point in the auditory pathway while he or she is listening to a passage of music.
3. Second, Temperley evaluates his models by comparing their output with the information in printed scores and published analyses. Personally, I think this is a very laudable thing to do and I believe that a correctly notated score of a passage of Western tonal music can represent quite well certain aspects of how an expert listener interprets the passage.
4. However, there exists a vocal contingent that abhors the use of scores in this way so some of you may disagree with me about that.
5. Third, Temperley (2001, p. 238) suggests that motivic analysis, harmonic and tonal analysis are all simply 'a search for a parsimonious encoding of the musical input'. This raises the larger question of whether perception is really simply a matter of data reduction or compression, that is, parsimonious encoding. I'd be interested to hear what you think about that.
6. Fourth, Temperley (2001, p. 19) cautiously suggests that dynamic programming might well be a feasible model of how the brain (or, at least, the mind) solves optimisation problems such as those involved when listening to a passage of music. Personally, I don't know enough about either dynamic programming or how the brain works to be able to formulate a proper opinion. However, some of you may know enough about both.
7. My fifth suggested topic arises from Temperley's (2001, p. 5) claim that it is quite valid to describe a cognitive process at the computational level 'without worrying about how it might be instantiated neurologically'.
8. Now, this might have been just an unfortunate slip of the pen, but it seems to me that over the past 15 years or so neurologists have learnt a great deal about precisely how the brain works. There has also been quite a lot of effort put into constructing detailed computational models of the auditory system that reasonably successfully account for certain aspects of music perception and cognition such as grouping and auditory scene analysis.
9. Now, it has almost become a cliché for those who model the mind to justify paying no attention to the brain on the grounds that so little is known about brain function that almost any model of the mind is feasible. It seems to me that the time for doing this may well now be past.

References

Lerdahl, F. and Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. MIT Press, Cambridge, MA.

Temperley, D. (2001). *The Cognition of Basic Musical Structures*. MIT Press, Cambridge, MA.