A Robust Parser-Interpreter for Jazz Chord Sequences

Mark Granroth-Wilding[1] and Mark Steedman[2]
[1]Computer Laboratory, University of Cambridge, UK,
[2]School of Informatics, University of Edinburgh, UK

Abstract

Hierarchical structure similar to that associated with prosody and syntax in language can be identified in the rhythmic and harmonic progressions that underlie Western tonal music. Analysing such musical structure resembles natural language parsing: it requires the derivation of an underlying interpretation from an unstructured sequence of highly ambiguous elements—in the case of music, the notes. The task here is not merely to decide whether the sequence is grammatical, but rather to decide which among a large number of analyses it has. An analysis of this sort is a part of the cognitive processing performed by listeners familiar with a musical idiom, whether musically trained or not.

Our focus is on the analysis of the structure of expectations and resolutions created by harmonic progressions. Building on previous work, we define a theory of tonal harmonic progression, which plays a role analogous to semantics in language. Our parser uses a formal grammar of jazz chord sequences, of a kind widely used for natural language processing (NLP), to map music, in the form of chord sequences used by performers, onto a representation of the structured relationships between chords. It uses statistical modelling techniques used for wide-coverage parsing in NLP to make practical parsing feasible in the face of considerable ambiguity in the grammar. Using machine learning over a small corpus of jazz chord sequences annotated with harmonic analyses, we show that grammar-based musical interpretation using simple statistical parsing models is more accurate than a baseline HMM. The experiment demonstrates that statistical techniques adapted from NLP can be profitably applied to the analysis of harmonic structure.

Keywords: harmony, expectation, grammars, machine learning, cognition.

1 Introduction

Hierarchical structure can be identified in rhythmic patterns of musical melodies and the harmonic progressions that underlie them (Winograd, 1968; Lindblom and Sundberg, 1969; Keiler, 1981; Lerdahl and Jackendoff, 1983; Steedman, 1984; Johnson-Laird, 1991; Pachet, 2000; Chemillier, 2004; Rohrmeier, 2011; Katz and Pesetsky, 2011). Similar structure is found in the prosody and syntax of language, commonly analysed using tree diagrams that divide a passage of speech or text recursively into its constituents, down to the level of individual words. It is reasonable to expect that the techniques used to process natural language might apply to the interpretation of music.

In natural language processing (NLP), analysing the syntactic structure of a sentence is usually a prerequisite to semantic interpretation. The main obstacle to such analysis is the high degree of ambiguity in even moderately long sentences and the search problem it engenders. In music, a similar sort of structural analysis, exhibiting a similar degree of ambiguity, is fundamental to interpretation by a listener (Lerdahl and Jackendoff, 1983; Steedman, 1984; Temperley, 2001). Hierarchical structures have been proposed to characterize the cognitive structures that underlie a listener's processing and recollection of a musical signal (Keiler, 1981; Steedman, 1984; Rohrmeier and Cross, 2009) and an analysis of these structures underlies the raising of harmonic expectation in a listener (Huron, 2006). The same analysis is involved in practical computational tasks such as key identification and score transcription. These tasks in general depend on both a harmonic (tonal) analysis and a rhythmic (metrical) analysis.

Our focus in the present paper is on analysis of the structures of harmonic expectation and resolution that underlie the cognition of harmonic progressions (Huron, 2006). We use the three-dimensional tonal harmonic space first described by Euler (1739) and others, and developed in computational terms, including the distance metric we use, by Longuet-Higgins (1962a,b) and Longuet-Higgins and Steedman (1971). This representation provides the basis for a theory of tonal harmonic progression—that is, a framework in which to analyse the relationships between the chords underlying a passage of music.

The input to the analysis is a sequence of chord symbols of the sort used by jazz performers on lead sheets. An assumption of our approach, in common with many others, is that this serves as a proxy for some intermediate level of representation which features in the process of harmonic analysis of a performance undertaken unconsciously by a listener. We treat this analysis of the tonal relations between chords analogously to the logical semantics of a natural language sentence. By defining a representation of relations in the tonal space in a form similar to that used to represent natural language semantics, we are able to apply techniques from NLP directly to the problem of harmonic analysis.

We define by hand a small formal grammar of jazz chord sequences using a formalism based closely on one widely used for NLP and developed from the version developed for musical purposes by Steedman (1996). We then use statistically-based modelling techniques commonly applied to the task of parsing natural language sentences with such grammars. The parser maps music, in the form of chord sequences, onto its harmonic interpretation expressed as a trajectory through the tonal space. To obtain the parsing model, we use supervised learning over a small corpus of chord sequences of jazz standards taken from lead sheets used by performers. Each is annotated by hand with harmonic analyses that we treat for the purposes of the parsing task as a gold standard.

It is important to be clear that the purpose of the handbuilt grammar is not to capture all and only the sequences in this corpus. It is rather to assign possible harmonic analyses to a much larger, essentially infinite, set of sequences in the same idiom. Although this grammar is small enough to write on a single page (figure 14), it is extremely ambiguous. Like natural language grammars, it allows large numbers of analyses, most of which are semantically ridiculous, for even quite simple examples. The purpose of the parsing model is to assign probabilities to these alternatives expressing their likelihood estimated on the basis of the training sample data, in order to choose the most likely, and even to exclude the least likely entirely, in order to reduce the search space for the correct one.

As in NLP, the parser is evaluated by the degree to which the analyses it chooses for held out unseen sentences correspond to those assigned by human annotators. In this connection, it is important to realize that it is more important that the training data be *consistent* than that it be correct in every detail. If it is consistent, then the parser will be able to recover the interpretations implicit in the annotation, *and any other comparably consistent annotation it is trained on*. However, if the annotation is not internally consistent, then it cannot be modelled.

It is for this reason that certain annotations in the standard parsing corpus for English, the Penn Treebank (Marcus et al. 1993), are tolerated despite being technically incorrect linguistically—for example, the distinction between N-modifiers and NP-modifiers was judged not to be reliably drawn by the annotators, so all nominal modifiers are annotated, often incorrectly, as NP-modifiers. Similarly, one can take linguistic issue over the choice of a "small clause" analysis of object control for the treebank—but no-one really cares, because the same modelling technique would be able to recover the alternative analysis, and the two are interconvertible.

Musical treebanking is the same. Opinions may differ as to whether in a chord sequence C F C, the F is harmonically dependent on the first C, the last C, or both. So long as the notation is both coherent and consistent on such points, it does not much matter which we choose.

Of course, both for NLP and the musical equivalent, we ultimately want an extrinsic evaluation, via objective success on a task in the real world, such as question answering from text or successful improvization. However, intrinsic evaluation on ability to recover the necessary information is a standard first step on the way, as a sanity check and as a benchmark for alternative approaches to prove themselves against. It is an evaluation of this sort that we present here.

The full corpus can be accessed, and its coherence and consistency be assessed, at `http://jazzparser.granroth-wilding.co.uk/`. All musical examples used in the paper can also be heard at that URL.

2 The Relationship of Music and Language

The temptation to believe music and language to be closely related cognitive systems seems irresistible (Sundberg et al., 1991; Rebuschat et al., 2011). Grammarians have noticed strong parallels between language

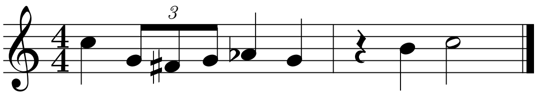


Figure 1: "Shave and a haircut, six bits"

and music at the level of the sound-systems of phonology and prosody (Daniélou, 1968, cf. Lerdahl and Jackendoff, 1983:314-330, Fabb and Halle, 2012). At times, this insight has led to the application of theoretical devices from language to music (Meyer, 1956; Longuet-Higgins, 1962a,b; Lindblom and Sundberg, 1969; Smoliar, 1976; Lerdahl and Jackendoff, 1983; Baroni et al., 1983, 1984; Temperley, 2007). It has been much less clear how to extend this apparently productive generalization to higher levels of structure and interpretation.

For example, what is the "meaning" of the melodic passage in figure 1 (first used in this context by Longuet-Higgins, 1976)? It is rhythmically and melodically well-formed and, in its little way, entirely satisfying as a piece of Western tonal music, in a sense that the first six notes alone would not be. In that sense, we may be tempted to assign a rhythmic and harmonic structure to it, say along the lines suggested by Lerdahl and Jackendoff (1983). The interest of such structures for present purposes lies in the extent to which we can assign them an interpretation.

The notion of musical meaning has very frequently been linked to the idea of the emotions (Cooke, 1959). The most empirically testable claims of this kind have defined emotion in terms of the satisfaction or frustration of musical *expectations* of various kinds (Meyer, 1956; Cooper and Meyer, 1963; Narmour, 1977; Margulis, 2005; Huron, 2006; Pearce and Wiggins, 2006; Lehne et al., 2013).

Most listeners will intuitively divide the tune in figure 1 into two parts, corresponding to the two bars, and sense that the first bar creates an expectation which the second bar satisfies. More specifically, the first bar moves from the tonic or key note C to the fifth or dominant tonality of G, which creates an expectation of a cadential return to the tonic.

The above description can be verified by making the claimed tonal progressions explicit with some chords: C major for the first half bar, establishing the tonic; G major (with the "dominant" seventh note—G^7) in the second half-bar; then a further chord of G^7 followed by C major in the second bar. In the light of this observation, we can claim that an important part of the meaning of the piece as a whole is a statement of the tonic, followed by a "cadential" progression from its dominant back to that tonic.

The notion of musical structure and meaning that we deal with in this paper is confined to such relatively local cadential relations between chords and sequences of chords. We make no claim that these relations extend to higher levels of structure, such as that of sonata form, as sometimes claimed by Schenker (1906) and followers. We suspect that a quite different kind of of rule may apply at these levels, such as the periodic patterns of Simon and Sumner (1968). In this respect, our theory is consistent with the observations of Tillmann and Bigand (2004) concerning the psychological distinction between "local" and "global" structure and interpretation in music.

3 Musical Syntax

The syntax of Western tonal harmony and that of natural language can both be analysed using tree structures, and both have been claimed to feature formally unbounded embedding of structural elements (Winograd, 1968; Keiler, 1981; Lerdahl and Jackendoff, 1983; Steedman, 1984; Rohrmeier, 2011). In harmony, these structures arise as a result of relationships of harmonic expectation and resolution between chords (Huron, 2006). This phenomenon is sometimes referred to as harmonic *tension*, but should not be confused with other notions of tension in music. For example, Lerdahl (2001) appeals to a quite different type of musical tension, which includes notions of harmonic expectation, but has more to do with a perceived sensation of tension in a listener, also due to metre, dissonance (see Johnson-Laird et al., 2012) and other musical factors.

In terms of the formal expressive power of their syntactic grammars, the works above contrast with approaches to harmonic analysis based on Rameau (1722) and Riemann (1893), though in other respects they are closely related. This paper describes a syntactic formalism based on that of Steedman (1996) for wide-coverage analysis of Western tonal harmony, focusing on the analysis of jazz standards, and the application of statistical parsing techniques to the practical problem of automatic computational analysis. Other approaches to computational analysis have been explored using related formalisms (Pachet, 2000; Chew, 2000; Hamanaka et al., 2006; de Haas et al., 2009; Marsden, 2010; Choi, 2011).

3.1 Cadences

The key component of harmonic structure is the *cadence* of the kind implicit in our analysis of figure 1, built from expectation-resolution patterns. Large structures can be analysed as *extended cadences*, made up of successive expectation-resolution patterns chained together[1]. These patterns can be formalized in terms of harmonic *function* (Riemann, 1893).

Cadences come in two varieties. The *authentic cadence* consists of a chord rooted a perfect fifth above its expected resolution. This type of tension chord is referred to as a *dominant* chord, and is the kind implicit in figure 1. The *plagal cadence* consists of a tension chord rooted a perfect fourth above its resolution. This type of tension chord is referred to as a *subdominant* chord. In both cases, the resolution chord is classified as a *tonic* chord. This classification of a particular occurrence of a chord identifies its *function* on that occasion of use, and partly establishes its place in the harmonic structure in relation to the surrounding chords. The same chord type, such as a G major triad, on different occasions of use in the same piece may function variously as a dominant or subdominant tension chord or as a tonic resolution (or both), or as a substitute for such dominants, subdominants or tonics (for example, as a Neapolitan sixth).

An extended cadence occurs when a tension chord resolves by the appropriate interval to a chord that is itself cadential, creating a further tension and subsequently resolving. An example is the D^{7} chord in figure 2, an *extended* dominant. Such a definition is recursive, and extended cadences can accordingly be indefinitely extended. This kind of extension is most common with the authentic cadence. We include in our use of the terms *dominant* and *subdominant* this recursive, or extended, function. Keiler (1981) treats the cadential relation in a similar recursive fashion.

A cadence Dm^{7} G^{7} C has two possible interpretations: it may contain a recursive dominant relation or be an alternative transcription of the common classical form of a perfect cadence F^{6} G^{7} C. However, when the recursion reaches back further, preceded for example by A^{7}, only the former interpretation explains the relation between the seemingly tonally distant tension chord and its eventual resolution (here the cadence from A^{7} eventually resolving to C), even in the case where a Dm^{7} chord would otherwise lead to an ambiguous interpretation.

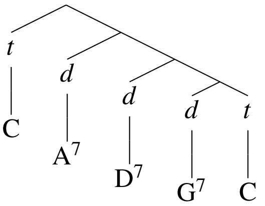


Figure 2: An extended authentic cadence, a typical example of (tail) recursion in music. The A^{7} acts as a dominant resolving to the D^{7}, which in turn resolves by the same relation to G^{7}, which then resolves to the tonic C.

A cadence might not reach its eventual resolution in a tonic chord immediately. An *unresolved* dominant cadence, such as D^{7} G^{7}, creating an expectation of tonic C, may be interrupted by a further cadence, say A^{7} D^{7} G^{7}, creating the same expectation, whereupon *both* cadential expectations will be resolved by the *same* tonic C, as in example (1).

(1) C (D^{7} G^{7}) (A^{7} D^{7} G^{7}) C

We refer to this operation as *coordination* by virtue of its similarity to right-node raising coordination in sentences like *Keats bought and will eat beets* (see section 5.1), in which *beets* satisfies the expectations of both *bought* and *eat*.

Coordinated cadences may themselves be embedded in a coordinated cadence, as in example (2) from *Call Me Irresponsible*, by Jimmy Van Heusen, with coordination of constituents marked by &.

(2) ((D♯o^{7} Em^{7} Am^{7}) & ((E^{7} & (Bϕ^{7} E$^{7\flat 9}$)) A^{7}))

The longer cadence in which example (2) appears includes still further levels of embedding and is shown as a tree structure in figure 3. This embedding process is again mirrored in natural language coordination like *Keats ((may or may not) cook) but (certainly eats) beets* (see section 5.1).

Chords that function as dominants are often partially, though ambiguously, distinguished by the addition of notes other than those of the basic triad. In particular, the "dominant seventh", realized by the addition of the note a (major) tone below the chord's root, enhances the cadential function of a dominant chord and height-

[1] Throughout this paper, the term *cadence* will be used precisely to refer to connected structures of expectation-resolution patterns and not to refer to resolutions at points of particular significance in the global structure of a piece.

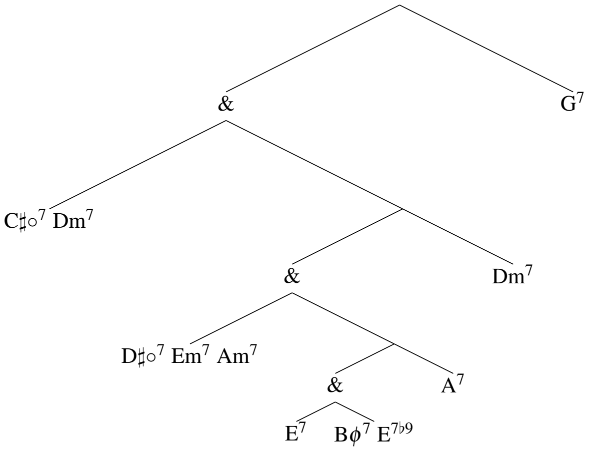


Figure 3: Tree representing the embedded structure of unfinished cadences in *Call Me Irresponsible*. The cadence shown here is in fact further embedded: the eventual resolution to the tonic C is not reached until after another cadence structure, similar to this one.

ens the expectation of the corresponding tonic. However, this note may be omitted from a dominant chord, and conversely the same keyboard note may be used in chords that are not functioning as a dominant, such as a Dm^7 functioning as a substitute for the subdominant of tonic C.

Although the theory of harmonic structure presented here is concerned with modelling the cognitive structures underlying harmonic expectation, it is worth noting that the same notion of recursive structure is required of any theory that accounts for the A^7 chord of example (1) by reference to a local tonality of D whilst maintaining the key of the passage as C (that is, without modulation).

3.2 The Jazz Sublanguage

The typical size and complexity of the cadence structures discussed above varies with musical period and genre. Tonal jazz standards or themes are of particular interest for this form of analysis for several reasons.

First, they tend to feature large extended cadences, often with complex embedding. Second, they contain many well-known *contrafacts*, harmonic variations of a familiar piece, created using a well-established system of harmonic substitutions, embellishments and simplifications. Finally, jazz standards are rarely transcribed as full scores, but are more analytically notated as a melody with accompanying chord sequence. Analysing the harmonic structures underlying chord sequences, rather than streams of notes, avoids some difficult practical issues such as voice leading and performance styles, but still permits discovery of the kind of higher-level structures we are concerned with.

Our study focuses on the analysis of harmonic structure in chord sequences of jazz standards. This is not to say that the approach is not applicable beyond this domain, nor that it is confined to analysing chord sequences. The lexicon of the grammar outlined below, however, is somewhat specific to the genre. The grammar of Rohrmeier (2011), although using a different notation to that proposed here, captures a very similar form of structural analysis (pers. comm.), but aims for broader coverage and has been shown to be applicable to the annotation of a wide range of genres.

To help with understanding many of the examples below, it is worth noting that jazz chord progressions use a rich vocabulary of chord types: sixth chords (C^6), major seventh chords (CM7), half-diminished chords (F♯ϕ^7), and so on, as well as simple major and minor chords. In general, the harmonic function of a chord is not fully determined by its type, but certain types strongly suggest a function. For example, a C^7 is likely, though not certain, to have a dominant function, whilst a C^6 is likely to have a tonic function. The examples below use such suggestive chord types to make more obvious the functional interpretation that is intended. However, the statistical modelling techniques we describe are fully general, and cope with the full ambiguity of interpretation characteristic of real performance.

4 A Model of Tonality

In analysing the roles of pitch in music, it is important to distinguish between *consonance*, the sweetness or harshness of the sound that results from playing two or more notes at the same time, and harmonic interpretation, relevant to the phenomenon that we have already alluded to as tension (or the creation of expectation) and resolution (or its satisfaction). Both of these relations over notes are determined by small whole-number ratios, and are easily confounded. However, they arise in quite different ways, the first from the perceptual apparatus, the second from cognition.

As is common in music theory, we treat these as two theoretically distinct musical dimensions. A complete model of musical cognition would require a theory of both and of their interaction (as discussed, for example, by Krumhansl, 1990 and Johnson-Laird et al., 2012). The present work models the harmonic dimension alone.

4.1 Consonance

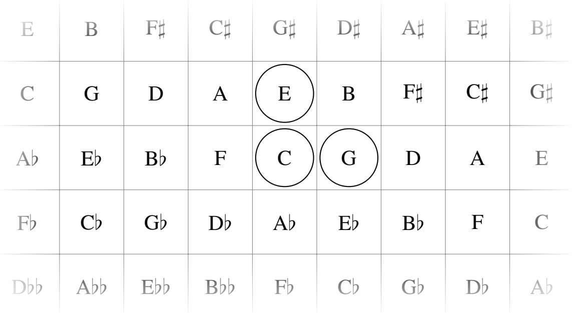
The modern understanding of consonance originates with Helmholtz (1862), who explained the phenomenon in terms of the coincidence and proximity of the secondary overtones and difference tones that arise when simultaneously-sounded notes excite real non-linear physical resonators, including the human ear itself. These tones may include all integer multiples (and, in some cases, dividends) and the fundamental.

4.2 Harmony

The tonal harmonic system also derives from combinations of small integer pitch ratios. However, the harmonic relation is based solely on the first three prime ratios in the harmonic series: ratios of 2, 3 and 5 (commonly known as the octave, perfect fifth and major third). The tuning based on these intervals is known as *just intonation*.

4.2.1 Just Intonation In just intonation, an interval can be represented as a frequency ratio defined as the product $2^x \cdot 3^y \cdot 5^z$, where x, y, z are positive or negative integers. It has been observed since Euler (1739) that the harmonic relation can therefore be visualized as an infinitely extending discrete three-dimensional space with these three prime factors as generators. Since notes separated by octaves are essentially equivalent for tonal purposes, it is convenient to project the space onto the $3,5$ plane. We present this theory as formally developed by Longuet-Higgins (1962a,b) in figure 4.

Longuet-Higgins and Steedman (1971) observed that all musical scales are convex sets of positions, and defined a Manhattan taxi-ride distance metric over this space. According to this metric, it will be observed that the major and minor triads, such as CEG (shown in figure 4) and CE♭G, when plotted in this space are two of the closest possible clusters of three notes. The triad with added major seventh is the single tightest cluster of four notes. The triads and the major seventh chord are therefore stable, raising no strong expectations, of the kind that typically end a piece. Chords like the augmented and diminished chords and the dominant seventh are more spread out. This difference is vital to the induction of harmonic expectation, and its satisfaction.



E	B	F♯	C♯	G♯	D♯	A♯	E♯	B♯
C	G	D	A	E	B	F♯	C♯	G♯
A♭	E♭	B♭	F	C	G	D	A	E
F♭	C♭	G♭	D♭	A♭	E♭	B♭	F	C
D♭♭	A♭♭	E♭♭	B♭♭	F♭	C♭	G♭	D♭	A♭

Figure 4: Part of the space of note-names (adapted from Longuet-Higgins, 1962a,b). Notes are separated by major thirds along the horizontal axis and perfect fifths along the vertical. The space extends infinitely in both dimensions. The circled points form a C major triad.

The space of justly intoned intervals does not include ratios involving higher prime factors. Whilst these ratios are important to the explanation of consonance, they do not play a role in the description of the tonal harmonic system.

4.2.2 Equal Temperament Over several centuries, it was gradually realized that the tonal harmonic space could be approximated, first by slightly mistuning the fifths to equate all the positions that have the same names in figure 4, and then by even further distorting the major thirds, to equate C with B♯, D♭♭, etc. In the system of *equal temperament*, this is done by spacing the 12 tones of the diatonic octave evenly, so that all the semitones are (mis)tuned to the same ratio of $\sqrt[12]{2}$.

Since the eighteenth century most instruments have been tuned according to equal temperament. It has the advantage that all keys and modes can be played on the same instrument without retuning and has permitted the development of musical styles in which pieces may modulate relatively freely between keys. In terms of the tonal space, the result is a projection onto a finite toroidal space of just 12 points, looping in both dimensions. Each point is (potentially, infinitely) tonally ambiguous as to which point in the full justly-intoned space of figure 4 it denotes.

Equal temperament thus obscures the harmonic relations between notes. However, human listeners can *resolve* this tonal ambiguity in context, and invert the projection onto the torus to recover the interpretation of the intervals in the full harmonic space. This is possible because the harmonic intervals that are sufficiently close in justly intoned frequency to be equated on the equally tempered torus *are sufficiently distant in the full* *space for the musical context to disambiguate them.* For example, if the context defines the tonality as G, then an equally tempered note that could in isolation be interpreted as any of C, B♯, D♭♭, etc. (or the identically named points to the left or right) must be interpreted as C, because that is the only harmonic interpretation that is anywhere close to G.

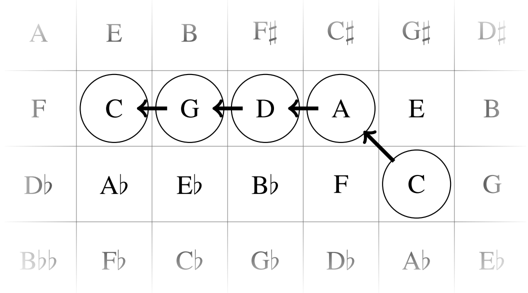
It is important to realize that ambiguous equally tempered music is unconsciously interpreted in terms of the full tonal frequency space of harmonic distinctions, just as a (theoretically, infinitely ambiguous) two-dimensional photograph is interpreted as a three-dimensional scene. It is for this reason that equally tempered B♭ is interpreted in tonal music as related to C by either a dominant seventh (ratio $\frac{8}{9}$) or a minor seventh (ratio $\frac{8}{9}$), but never by an interval related to the seventh harmonic (ratio $\frac{7}{8}$). The equally tempered minor/dominant seventh should therefore never be claimed to approximate a suboctave of the seventh harmonic, as is often alleged (Jeans, 1937; Bernstein, 1976; Tymoczko, 2006). This is not to deny that varieties of music *other* than the tonal might take the seventh harmonic as a primitive ratio, although it is doubtful that such a music could support equal temperament or even a very extensive form of harmony. Experience of the Bohlen-Pierce scale (Mathews and Pierce, 1989) appears to prove the point.

It should be noted that the tonal space used here represents the tonal relations between notes in the tonal harmonic system and is thus unsuited to a model of consonance and dissonance. As noted above, we maintain the distinction on theoretical grounds between these two musical structures and do not modify the space to better accommodate a notion of proximity due to consonance or voice leading (cf. Euler, 1739; Riemann, 1914; Lerdahl, 2001; Tymoczko, 2011) or perceptual responses, which may be the result of a combination of these and other structures (cf. Krumhansl, 1990).

4.3 Domain for Analysis

In our grammar for jazz chord sequences, we take the full tonal space as the semantic domain of harmonic analysis. The harmonic interpretation of a piece is the path through the tonal space traced by the roots of the chords.

If we establish that there is a dominant-tonic expectation-resolution relationship between two chords, we know that the underlying interval between the roots is a perfect fifth, a single step to the left in the space. On the other hand, establishing that a pair of chords stand in a subdominant-tonic relationship dictates a perfect fourth between them, a rightward step. Where no expectation-resolution relationship exists, as between a tonic and the first chord of a cadence that follows it, we assume a movement to the most closely tonally related instance of the chord root.



A	E	B	F♯	C♯	G♯	D♯
F	C	G	D	A	E	B
D♭	A♭	E♭	B♭	F	C	G
B♭♭	F♭	C♭	G♭	D♭	A♭	E♭

Figure 5: A tonal space path for the extended cadence: C A^7 D^7 G^7 C.

Figure 5 shows an example of a tonal space path for an extended cadence. The perfect fifth relationship between the dominants and their resolutions is reflected in the path. The first step on the path is not an expectation-resolution relationship, so proceeds to the closest instance of the A (according to the Manhattan distance metric of Longuet-Higgins and Steedman, 1971). By identifying the syntactic structure of the harmony, that is the recursive structure of expectation-resolution relationships between pairs of chords, we produce the path through the space that this dictates for the chord roots of the progression, including those that have been substituted for.

5 Combinatory Categorial Grammar

Combinatory Categorial Grammar (CCG) is a grammar formalism used for parsing natural language sentences to produce logical representations of their semantics. A short introduction to CCG is given in the next section. For a full introduction to its application to natural language, see Steedman (2000).

5.1 CCG for Language

A CCG grammar includes a lexicon, which associates words with one or more syntactic categories determining structures they may appear in. During parsing, categories assigned to consecutive chords are projected onto

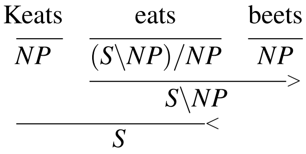


Figure 6: A derivation showing the function application rule in use.

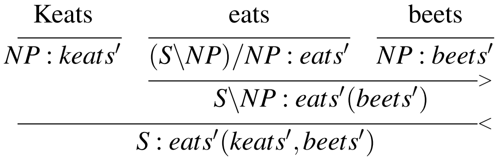

Keats
eats
beets
$NP : keats'$
$(S\backslash NP)/NP : eats'$
$NP : beets'$
$>$
$S\backslash NP : eats'(beets')$
$<$
$S : eats'(keats', beets')$

Figure 7: An example of a derivation with a logical form associated with each category.

constructions and sentence interpretations using a small set of combinatory rules, constrained by the form, or "type", of the categories.

The lexical categories are defined in terms of a small set of atomic syntactic types, including, for instance, S (sentence) and NP (noun phrase). A category's combinatory potential is defined in terms of the atomic types using the $/$ and $\backslash$ operators. Thus, a category X/Y denotes a *function* category that can combine with an *argument* category Y *to its right* to produce a result of category X. Likewise $X\backslash Y$ indicates that a Y is expected to the left.

Categories can combine by grammatical rules of *function application*, defined as follows. The symbols $>$ and $<$ are used to identify the rules in derivations, such as figure 6.

a. $X/Y \quad Y \quad \Rightarrow \quad X \quad (>)$
b. $Y \quad X\backslash Y \quad \Rightarrow \quad X \quad (<)$

Figure 6 shows the use of the function application rule in a simple syntactic derivation. (Note that the term *function* is not related to the concept of harmonic function.)

In order to produce an interpretation for the full sentence from the syntactic derivation, each lexical item also has a semantics, or logical form, and each rule defines how the logical forms of its arguments are combined. The function application rules in their full form are:

a. $X/Y : f \quad Y : x \quad \Rightarrow \quad X : f(x) \quad (>)$
b. $Y : x \quad X\backslash Y : f \quad \Rightarrow \quad X : f(x) \quad (<)$

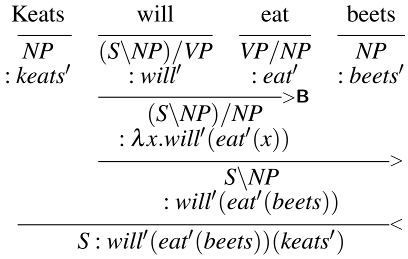


Figure 8: A derivation demonstrating the use of the function composition rule

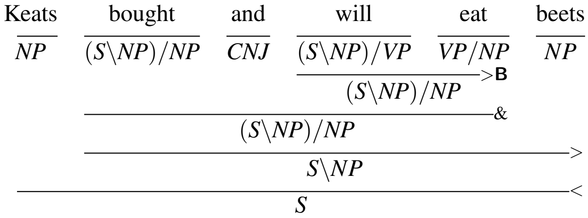


Figure 9: A derivation using a coordination rule to combine two constituents of the same syntactic type, separated by a conjunction (*and*).

Figure 7 shows an example of a derivation with semantics. We use an apostrophe to distinguish the language-independent meaning of a word from its written surface form. Thus, *beets'* refers in logical expressions to the objects denoted, depending on the language, by the words *beets*, *betteraves*, *betor*, etc.

Several other rules allow grammars to capture linguistic phenomena such as coordination and relativization. The only one relevant to the present discussion is *function composition.*

Function composition rules permit complex categories to be combined before their argument is available. The result may then be applied (using function application) to the argument when it is eventually encountered. The final outcome is the same as if only function application had been used, but composition allows this outcome to be produced by a different order of combinations. This is important for, among other things, incremental analysis of a sentence. Figure 8 demonstrates the use of the function composition rule.

Function Composition:

a. $X/Y : f \quad Y/Z : g \quad \Rightarrow \quad X/Z : \lambda x.f(g(x)) \quad (>\mathbf{B})$
b. $X\backslash Y : f \quad Z\backslash X : g \quad \Rightarrow \quad Z/Y : \lambda x.g(f(x)) \quad (<\mathbf{B})$

It should be noted that, although in this particular derivation the analysis of the tensed verb phrase is left-branching, the logical form that it builds is right branching and identical to that in the alternative function application-only derivation, as its semantics requires. (The latter is suggested as an exercise.)

The full range of reasons for treating natural language grammar in this way need not detain us here, but one is to do with the fact that constructions like coordination involving long-range semantic dependencies treat incomplete fragments like *will eat* as typable *constituents* that can be combined with others of the same type in derivations. Figure 9 shows the use of the coordination rule to combine *bought* and *will eat* into a single constituent that can combine with *beets* (the semantics is omitted, but can be inferred from figure 8).

The rest of the paper shows that musical analysis involves similar long-range dependencies, and calls for the same approach.

5.2 CCG for Harmony

For parsing the syntax of harmony, we use a formalism similar to the standard CCG for English. Following Steedman (1996), we use harmonic syntactic categories that define cadential expectation, like G^D/C^T, identifying chords like G^7 as combining with a C chord acting as a tonic to its right. In both cases, categories X/Y can be seen as defining "expectation" of Y.

Lexical categories are assigned to chords. Pairs of adjacent categories are then combined using a small set of rules to build up an interpretation of the whole passage. We use some of the standard combinatory rules given in section 5.1 and some rules specific to harmonic syntax. Each category, lexical or derived, is paired with a semantics, or logical form, representing an interpretation of the chord roots in the tonal space—the harmonic analysis itself. We describe this in detail in the appendix.

In the examples here, we shall omit the logical forms of categories, but it is crucial to bear in mind that each intermediate category produced during a derivation corresponds to a partial harmonic interpretation of the chords and that this is available as the category's logical form. This distinction between the analysis structure and the constituent structures required to build it compositionally from the surface form has been made for natural language semantics (Steedman, 2000) and was first proposed in the present form for harmonic analysis by Steedman (1996). Closely related formalisms for grammatical harmonic analysis have been proposed without making this distinction formally explicit (Keiler, 1981; Rohrmeier, 2011).

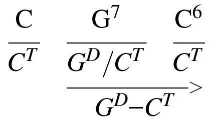


Figure 10: Partial CCG derivation of a simple cadence: a dominant-tonic resolution. The derivation uses the dominant and tonic lexical categories, combined using the function application rule (see section 6.2). The symbol $>$ identifies the rule used.

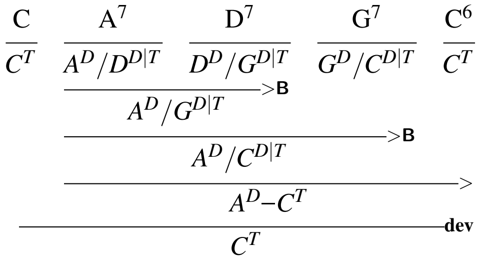


Figure 11: CCG derivation of an extended cadence, using the tonic category and the (extended) dominant category, combined by using the rules of section 6.2. Dominant categories are combined first to interpret the incomplete cadence, which is then combined with its resolution. The symbols $>$, $>_{\mathbf{B}}$ and **dev** identify the rules used.

An atomic category's syntactic type carries information about the tonality at the start and end of the passage it spans. This is the only harmonic information relevant to constraining how it can combine with adjacent categories. Each end has a harmonic root, in the form of an equally tempered pitch class, and a chord function, one of T (tonic), D (dominant) and S (subdominant). For brevity, where the start and end parts of an atomic category are the same, we write just one: a category $C^T{-}C^T$ is abbreviated to C^T. Such a category is used to represent a tonic chord.

A passage beginning on a G chord functioning as dominant, followed by a tonic C receives the syntactic type $G^D{-}C^T$. The two-step cadence D^7 G^7 C would receive the type $D^D{-}C^T$. In the latter example, the type has been derived from interpretations of the C as a tonic chord, the D^7 a secondary dominant and the G^7 a dominant, combining these partial interpretations and their associated harmonic analyses. Typically a sequence of chords specified by equally tempered notes

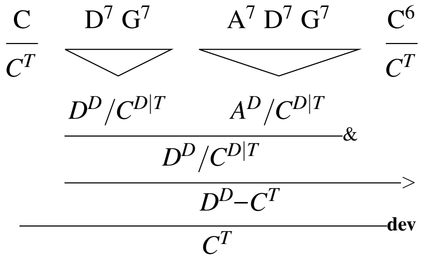


Figure 12: CCG derivation using the coordination rule to combine interpretations of unresolved cadences.

will support many such interpretations, varying in plausibility.

A forward-facing slash category X/Y gives the starting tonality Y expected for the category to its right (its argument) and the starting tonality X that will be used for the result of applying it to such an argument. Such a category is used to reflect the interpretation of a dominant chord, like those in figure 10. A forward slash category can be combined with its immediately following resolution to a tonic chord using the *function application* combinatorial rule, as shown in figure 10.

Extended cadences, using a recursive, or extended, dominant function, such as the one in figure 11, are handled by allowing the dominant category to combine with another dominant, using the *function composition* rule (identified by $>_{\mathbf{B}}$), just as in the linguistic examples in section 5.1. The result is another forward slash category, which is subsequently combined with the resolution. The lexical category used to interpret a dominant chord, the G^D/C^T of figure 10, is now replaced by $G^D/C^{D|T}$, permitting it to combine either with a tonic resolution (as in figure 10) or with another dominant chord (as in figure 11). It should be noticed that this particular derivation of the extended cadence is now left-branching, like the example in section 5.1, even though the analysis it produces (omitted in figure 11) is formalized as a right-branching ("tail-recursive") structure.

Backward slash categories are precisely the reverse of forward slash categories. They specify the end tonality required of the argument (after the slash) and the end tonality that the result will have. They are much less used than / and usually simpler[2].

A combinatory rule resembling the one used for natural language coordination (see section 5.1) allows interpretation of interrupted, or *coordinated*, cadences. A chain of dominant seventh chords, without their final resolution, can be treated as a constituent, thanks to function composition. The *coordination* rule (identified by &) combines two such constituents which expect the same resolution into a single slash category, which also expects this resolution[3]. We demonstrate this in figure 12 (derivations of the constituents are omitted for brevity).

As is typically the case with lexicalized grammar formalisms, much of the work of interpretation is done in the choice of lexical categories for each chord. Chord substitution is handled in this way. For example, jazz musicians may replace a dominant seventh chord by another dominant seventh chord whose root is an augmented fourth lower. This is the *tritone substitution*. Each substitution is handled by adding a new line to the lexical schemata in figure 14, discussed in the next section. An example derivation using a tritone substitution is shown in figure 13.

6 A Grammar for Jazz

6.1 The Lexicon

We are now able to define a jazz chord lexicon in full, shown in figure 14. Each entry is a lexical schema which has a mnemonic label to serve as an identifier, a surface chord class, a syntactic type and a logical form (see the appendix for full details of the notation for logical forms). The surface chord class generalizes over chord roots X. During parsing, a lexical schema may be used to assign a category to a chord, provided the chord falls into the general class of chords represented to the left of the :=. Thus, whilst the harmonic analysis takes the form of a path traced by the chords' roots through the tonal space, the chords' types restrict the categories that may be used to interpret them[4]. The pitch classes within the category itself (right of the :=) are expressed, using roman numerals, relative to the played root of the chord to which the category is assigned (X). An example use of each schema is given in the key of C.

All surface chords are assumed to be in equal temperament. The input therefore does not distinguish between enharmonically equivalent roots, like G♯ and A♭.

[2] This is a result of the fact that the analysis, which represents harmonic expectation, is inevitably forward-looking. The few backward slash categories used handle tonic elaborations and do not contribute any relations to the analysis structure.

[3] This approach to interrupted cadences differs from that most commonly seen in music theory (Piston, 1949) and is advocated by, among others, Keiler (1978) and Rohrmeier (2011).

[4] Chord types have further influence on the interpretation through the statistical models introduced below.

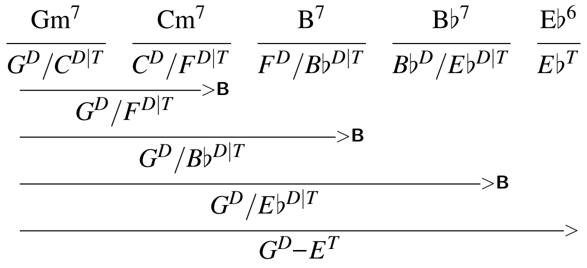


Figure 13: A cadence from *Can't Help Lovin' Dat Man* (in the key of E♭). The B^7 replaces an F^7, an example of the tritone substitution, and receives the same syntactic type that F^7 would have received.

Indeed, this disambiguation is part of the analysis performed during parsing, and may be inferred from the logical form of a full parse. The constraints expressed by the syntactic categories operate prior to this analysis, so cannot make these distinctions. We arbitrarily choose to use flats throughout the lexicon.

The mnemonic label *Ton* is used to identify a simple tonic chord function. The corresponding syntactic category takes on the chord's pitch class. The logical form represents a point in the tonal space which is constrained to be one of those points that are mapped by equal temperament to the root of the surface chord. At this stage, it is meaningless to distinguish between the points in this infinite set: what will be of importance is the root's relation to other points in the path. The logical form is a coordinate in a 4×3 space identifying this set, written $\langle x,y\rangle$. Like the syntactic types, the coordinate in the lexicon implicitly generalizes over the possible roots of the surface chord. For example, if the surface chord has root C, the logical form will become $\langle 0,0\rangle$, whilst if the root is B the logical form is $\langle 1,1\rangle$ (see figure 4).

The mnemonic *Dom* identifies a rule that says a surface chord C^7 can be interpreted with the syntactic type $C^D/F^{D|T}$ and a logical form denoting a leftward movement in the space to its resolution. It can be applied, for example, to a surface chord G^7, giving the syntactic type $G^D/C^{D|T}$. As in the natural language semantics in section 5.1, we use the lambda calculus to express a predicate whose argument is not yet filled. When one of these categories is combined with its resolution, the predicate (*leftonto*/*rightonto*) will be applied to the resolution's logical form. Figure 15 shows this in action to interpret a short extended cadence.

The mnemonic *Dom-tritone* in figure 14 identifies the tritone substitution of a dominant function chord. The syntactic type is identical to that that would have been assigned as a simple dominant interpretation of the substituted chord (that rooted on the tritone). In other words, this entry allows us to interpret a chord D♭^{7} exactly as if it had been a G^7 chord. Most of the entries in the lexicon represent other substitutions and work along similar lines. Those included in the lexicon here constitute a set suitable to interpret a large range of jazz standards, but more could be added to cover a wider range of substitutions or to adapt the grammar to a different domain.

6.2 Combinatory Rules

Most of the work of interpretation is done in the selection of lexical categories. Only four rules are used to build derivations. **Function application** and **function composition** are merely adaptations of their conventional forms to the musical formalism and behave as described in section 5.2. The rules are applied to simultaneously combine the syntactic categories and the logical forms. Each rule has a symbol used to identify its use in derivations.

Function application:

Forward ($>$)

$$X/Y : f \quad Y{-}Z : x \quad \Rightarrow \quad X{-}Z : f(x)$$

Backward ($<$)

$$X{-}Y : x \quad Z\backslash Y : f \quad \Rightarrow \quad X{-}Z : f(x)$$

Function composition:

Forward ($>_{\mathbf{B}}$)

$$X/Y : f \ Y/Z : g \Rightarrow X/Z : \lambda x.f(g(x))$$

Backward ($<_{\mathbf{B}}$)

$$X\backslash Y : g \ Z\backslash X : f \Rightarrow Z\backslash Y : \lambda x.f(g(x))$$

The **coordination** rule combines unresolved cadences to behave as a single unresolved cadence. The two cadences are required to be of the same harmonic

Mnemonic label	Category schema			Example	
				chord	syntactic type
Ton.	X(m) :=	I^T :	$[\langle 0,0\rangle]$	CM7	C^T
Ton-III.	Xm :=	$\flat VI^T$:	$[\langle 0,2\rangle]$	Em	C^T
Ton-bVI.	X :=	III^T :	$[\langle 0,1\rangle]$	A♭M7	C^T
Dom.	X(m)^7 :=	$I^D/IV^{D\vert T}$:	$\lambda x.leftonto(x)$	G^7	$G^D/C^{D\vert T}$
Dom-backdoor.	X(m)^7 :=	$VI^D/II^{D\vert T}$:	$\lambda x.leftonto(x)$	B♭^{7}	$G^D/C^{D\vert T}$
Dom-tritone.	X(m)^7 :=	$\flat V^D/VII^{D\vert T}$:	$\lambda x.leftonto(x)$	D♭^{7}	$G^D/C^{D\vert T}$
Dom-bartok.	X(m)^7 :=	$\flat III^D/\flat VI^{D\vert T}$:	$\lambda x.leftonto(x)$	E^7	$G^D/C^{D\vert T}$
Subdom.	X(m) :=	$I^S/V^{S\vert T}$:	$\lambda x.rightonto(x)$	F	$F^S/C^{S\vert T}$
Subdom-bIII.	X :=	$VI^S/III^{S\vert T}$:	$\lambda x.rightonto(x)$	A♭	$F^S/C^{S\vert T}$
Dim-bVII.	X∘ :=	$IV^D/\flat VII^{D\vert T}$:	$\lambda x.leftonto(x)$	Ddim^7	$G^D/C^{D\vert T}$
Dim-V.	X∘ :=	$II^D/V^{D\vert T}$:	$\lambda x.leftonto(x)$	Fdim^7	$G^D/C^{D\vert T}$
Dim-III.	X∘ :=	$VII^D/III^{D\vert T}$:	$\lambda x.leftonto(x)$	A♭dim^7	$G^D/C^{D\vert T}$
Dim-bII.	X∘ :=	$\flat VI^D/\flat II^{D\vert T}$:	$\lambda x.leftonto(x)$	Bdim^7	$G^D/C^{D\vert T}$
Pass-I.	X∘ :=	I^T/I^T :	$\lambda x.x$	Cdim^7	C^T/C^T
	X∘ :=	I^D/I^D :	$\lambda x.x$	Gdim^7	G^D/G^D
Pass-VI.	X∘ :=	VI^T/VI^T :	$\lambda x.x$	Adim^7	C^T/C^T
	X∘ :=	VI^D/VI^D :	$\lambda x.x$	Edim^7	G^D/G^D
Pass-bV.	X∘ :=	$\flat V^T/\flat V^T$:	$\lambda x.x$	G♭dim^7	C^T/C^T
	X∘ :=	$\flat V^D/\flat V^D$:	$\lambda x.x$	D♭dim^7	G^D/G^D
Pass-bIII.	X∘ :=	$\flat III^T/\flat III^T$:	$\lambda x.x$	E♭dim^7	C^T/C^T
	X∘ :=	$\flat III^D/\flat III^D$:	$\lambda x.x$	B♭dim^7	G^D/G^D
Aug-bII.	X^7 :=	$\flat VI^D/\flat II^{D\vert T}$:	$\lambda x.leftonto(x)$	Baug	$G^D/C^{D\vert T}$
Aug-VI.	X^7 :=	$III^D/VI^{D\vert T}$:	$\lambda x.leftonto(x)$	E♭aug	$G^D/C^{D\vert T}$
Colour-IVf.	X(m) :=	V^T/V^T :	$\lambda x.x$	F	C^T/C^T
Colour-IVb.	X(m) :=	$V^T\backslash V^T$:	$\lambda x.x$	F	$C^T\backslash C^T$
Colour-IIf.	X(m) :=	$\flat VII^T/\flat VII^T$:	$\lambda x.x$	Dm	C^T/C^T
Colour-IIb.	X(m) :=	$\flat VII^T\backslash\flat VII^T$:	$\lambda x.x$	Dm	$C^T\backslash C^T$
Dom-IVm.	Xm :=	II^D/V^T:	$\lambda x.leftonto(x)$	Fm^6	G^D/C^T

Figure 14: The lexicon of the jazz grammar. Each line represents a lexical schema which may be used to interpret a chord. Each schema consists of a class of chord types it may interpret, a syntactic type and a logical form. For each schema, a typical example is given of a chord in the key of C that might receive this interpretation, and the syntactic type of the category it would be assigned.

$$
\dfrac{\dfrac{\dfrac{\text{D}^7}{D^D/G^{D|T} : \lambda x.\mathit{leftonto}(x)} \quad \dfrac{\text{G}^7}{G^D/C^{D|T} : \lambda x.\mathit{leftonto}(x)}}{D^D/C^{D|T} : \lambda x.\mathit{leftonto}(\mathit{leftonto}(x))}{>}\mathbf{B} \quad \dfrac{\text{C}}{C^T : [\langle 0,0\rangle]}}{D^D{-}C^T \ : [\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))]}{>}
$$

Figure 15: CCG derivation, including a representation of the harmonic 'semantics'—the structure of harmonic expectations and resolutions.

functional type—either authentic (dominant function) or plagal (subdominant function). The logical form of the result (not shown in the rule here) is a function that will be applied to the resolution and represents both cadences resolving to the same point. See the appendix for a formal definition.

Coordination (&):

$$X^F/Y \quad Z^F/Y \quad \Rightarrow \quad X^F/Y \qquad\qquad F \in \{D,S\}$$

The trivial **development** rule joins together fully resolved passages, building an interpretation of a whole piece out of its constituent cadences. Its semantics (also found in the appendix) is simply the concatenation of the two constituents. Thus, a piece of music is analysed as a sequence of expectation-resolution structures and no structure is analysed between these fragments (cf. Rohrmeier, 2011).

Development (dev):

$$V{-}W \quad X{-}Y \quad \Rightarrow \quad V{-}Y$$

This is a permissive rule: it permits any two consecutive passages interpreted individually as harmonically stable to be conjoined, regardless of key. Such passages include resolved cadences and individual tonic chords. Since different modulations are treated identically in the semantics and there is no reason to suppose that any remote modulation, however rare, is impermissible, we do not use the syntactic component of the grammar to put any restrictions on modulation. Statistical preferences are captured by statistical parsing models such as the one discussed below.

An example derivation using all four rules is shown in figure 16. It is shown with its semantics in the appendix.

7 Statistical Parsing Models

Just as with natural language parsing, the lexical ambiguity of interpretation of chord sequences, due here largely to the range of substitutions covered by the grammar, prohibits exhaustive parsing to deliver every syntactically well-formed interpretation. Moreover, we need a way to distinguish the most plausible among a huge number of possible interpretations. It is usual in NLP to use statistical models based on a corpus of hand-annotated sentences to rank possible interpretations (*supervised* models). Such techniques can be used to speed up parsing by eliminating apparently improbable interpretations early in the process. Bod (2002), Honingh and Bod (2005), Temperley (2007) and Marsden (2010) have shown that statistical techniques used in NLP can be applied to chord sequence parsing and other tasks for other genres (such as folk songs). This paper shows that such methods can be extended to the present more harmonically demanding musical domain.

7.1 Jazz Corpus

To train our statistical models, we have constructed a small corpus of jazz chord sequences. The sequences are taken from lead sheets standardly used by jazz performers. In the interests of consistent annotation, we excluded certain sequences whose analysis we were uncertain of, either because they included rare ambiguous modal substitutions, or because they seemed to lie outside the rather mainstream jazz idiom we sought to capture (one regretted example was Thelonious Monk's *Epistrophy*). The focus of the present approach is on the rather uncontentious harmonic structures described above. We believe that similar syntactic grammars could be defined for genres incorporating a wider range of harmonic expectations, a belief which is encouraged by the work of Rohrmeier (2011), which applies a very similar grammar to ours to a range of genres.

Every chord has been annotated by a single annotator with a mnemonic code representing a category from the lexicon of the jazz grammar shown in figure 14. Due to the small number of rules used by the grammar, the addition of parentheses surrounding coordinated sequences is sufficient to implicitly define a unique tonal

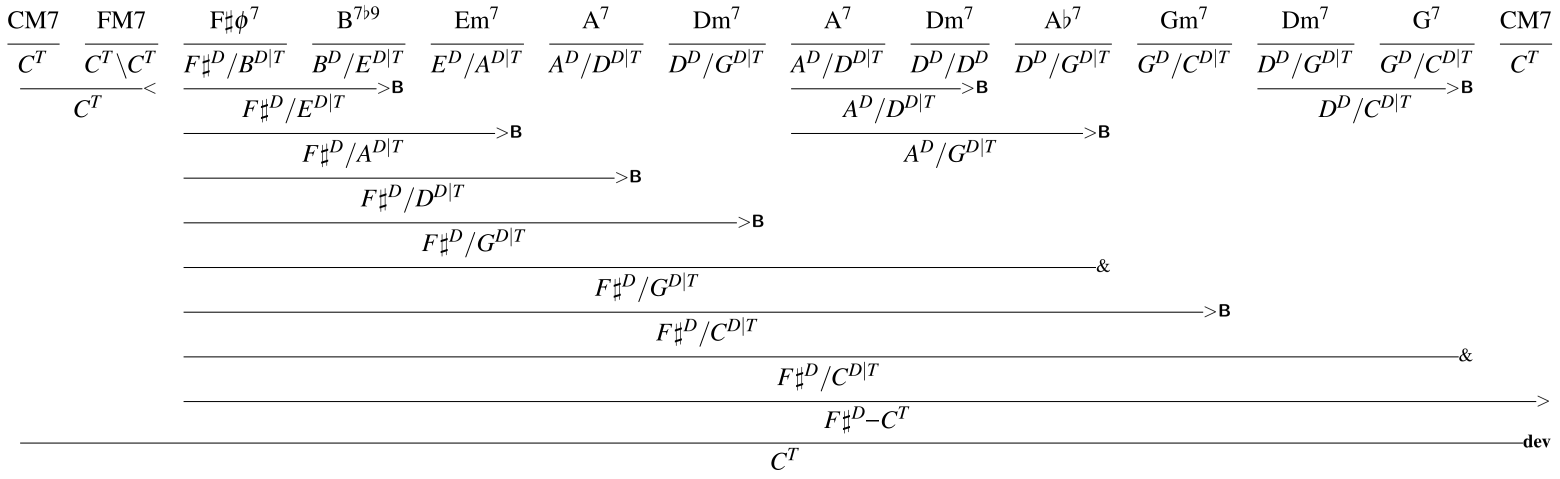


Figure 16: Syntactic derivation of a long extended cadence from *Alice in Wonderland* using all rules. Logical forms are omitted. The same example is shown with its semantics in the appendix.

space analysis of every sequence. We do not claim that these are the only possible analyses in every detail, only that they are musically coherent and consistent. The corpus consists of 76 annotated sequences, totalling roughly 3,000 chords. It is available to download from `http://jazzparser.granroth-wilding.co.uk/`.

7.2 Parsing Models

Hockenmaier and Steedman (2002) adapted the generative probabilistic parsing models of probabilistic context-free grammars (PCFG) to CCG. Using a corpus of parsed sentences, generative probability distributions are estimated for expansions at internal nodes in the derivation tree—steps of the derivation where categories are combined to interpret a larger span. The distributions are used to estimate a probability for any full derivation tree and hence of the corresponding harmonic analysis. If multiple full parses are found, they can be ranked according to the probabilities assigned by the model. In evaluating the parser, we will always choose the single most probable interpretation.

In our experiments, we use a direct adaptation of the model of Hockenmaier and Steedman (2002) to parse chord sequences. We refer to this model as PCCG. During parsing, a probability is assigned compositionally to every derived category using the parsing model. (See Hockenmaier and Steedman, 2002 for details of the model.) A *beam* is applied to every internal node at which multiple possible interpretations are found: all but the most probable derivations are removed, reducing the time the parser needs to spend exploring unpromising partial derivations.

7.3 Adaptive Supertagging

This beam search parsing strategy permits practical parsing of chord sequence inputs despite the high level of lexical ambiguity in the grammar. However, parsing speed can be further increased using another statistical technique from natural language parsing.

Supertagging is a technique, related to part-of-speech tagging, used for parsing with lexicalized grammars like CCG (Srinivas and Joshi, 1994). Probabilistic sequence models, using only statistics about short-distance dependencies, are employed to choose full CCG categories (rather than parts of speech) from the lexicon for each word. In general, the choice of category, representing for us most of the interpretation of a chord, depends on analysis of more distant parts of the sequence, that is on long-distance dependencies. In practice, short distance statistics can usually quite reliably rule out at least the least probable interpretations.

A bad choice of categories could make it impossible to parse the sequence. The *adaptive supertagging* algorithm (Clark and Curran, 2007) allows categories considered less probable by the supertagger to be used if necessary. First, the supertagger assigns a small set of most probable categories to each word and the parser attempts to find a full parse with these categories. If it fails, the supertagger supplies some more, slightly less probable, categories and the parser tries again. This is repeated until the parser succeeds or gives up (for example, after a set number of iterations).

Many types of probabilistic sequence model can be used as a supertagging model. We use a hidden Markov model (HMM) in which states represent categories, decomposed into a choice of lexical schema *Sch*, from the lexicon of table 14, and pitch *SR*. To avoid problems of data sparsity, the transition distribution of the HMM is modelled as a choice of schema, conditioned on the previous schema, and a choice of root pitch, conditioned on the schema and the previous root pitch:

$$P_{tr}(Sch_i, SR_i | Sch_{i-1}, SR_{i-1}) = \\ P(Sch_i | Sch_{i-1}) \times P(\delta(SR_i, SR_{i-1}) | Sch_i) \quad (1)$$

The function $\delta(x,y) \in \{0,\ldots,11\}$ represents the difference between two pitch classes x and y. This has the effect of making the model generalize over keys.

The input given to the supertagger is a sequence of chord labels, of the sort used on jazz lead sheets, such as CM7 and B$\flat^7$. These are decomposed into a pitch class *CR* (C, B$\flat$) and a chord type *CT* (M7, ^{7}). The emission distribution of the HMM is defined to be 0 for all schema roots that do not match the chord root, conditioning the chord type just on the lexical schema of the state.

$$P_{em}(CT_i, CR_i | Sch_i, SR_i) = \\ \begin{cases} P(CT_i | Sch_i) & \textit{if } CR_i = SR_i \\ 0 & \textit{otherwise} \end{cases} \quad (2)$$

The supertagging model is trained using maximum likelihood estimation on the annotated categories from the corpus described above. The limited size of the corpus means that it does not contain enough data to train models much more complex than this. Some initial experiments with higher-order Markov models (n-gram models) suggest that they do not perform any better than

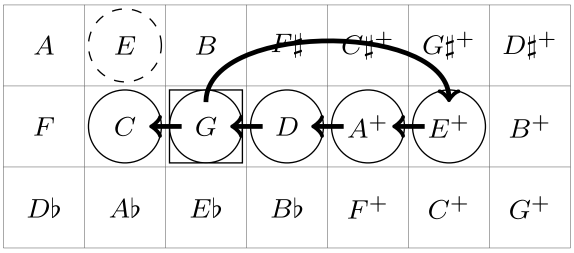


Figure 17: Tonal space analysis for the coordinated cadence G^7 E^7 A^7 Dm^7 G^7 C. The initial G^7 (square) is followed not by the closest point that equal temperament maps to E (dashed), but a more distant one, as required for the two G^{7}s resolve to share their resolution.

the HMM we use here when trained on this small corpus.

We compare a system using the PCCG parsing model to a second using the supertagger with the adaptive supertagging algorithm to narrow down the choice of lexical categories available to the parser. The parser applies a beam just as in PCCG. We call this model ST+PCCG.

Using both models, we allow the parser a fixed amount of time to parse a particular sequence before giving up. We set this time to five hours and with both systems almost all parses finished well within that time.[5]

7.4 Baseline Model

In an attempt to quantify the contribution made by restricting interpretations to those that are both syntactically well-formed under the jazz grammar and likely under the jazz parsing model, we have constructed an alternative baseline model which assigns tonal space interpretations without using the grammar. This baseline uses an HMM very similar to that described above as a supertagger model, which directly assigns a tonal space point to each chord, instead of assigning categories to chords and parsing to derive a tonal space path.

A reasonable first approximation to an analysis can be derived by assuming that no chord substitutions are used and that the tonal space path proceeds by the smallest possible steps, according to the Manhattan distance metric. There are two reasons why deviations from the naive path occur. First, the correct disambiguation of the equal temperament note may not be the point closest to the previous, as happens at points of coordination, where the resolutions of two cadences are constrained to be the same. An example is shown in figure 17. Second, there may be a substitution (like the tritone substitution), meaning that the surface chord's root is not the root of the chord in the analysis.

The HMM's state labels consist of three values. The first, the *substitution coordinate*, *Sub*, denotes the pitch class of the chord root after accounting for substitution, but before projecting from the toroidal space of equal temperament onto the full tonal space. For example, a state with $Sub = G$ could be associated with a D♭ chord to interpret it as a tritone substitution.[6] The second value, the *block coordinate*, *Blk*, is a coordinate that denotes the relationship in the tonal space between the actual point in the analysis and the point nearest to the previous point on the path after accounting for substitution. That is, it accounts for disambiguation of enharmonically equivalent points (e.g. C♯ and D♭). Although an infinite number of block coordinates is possible, in practice only a few are commonly seen and the HMM only includes those states that it observes in the training data. The third value is the harmonic function F of the chord—T, D or S.

The HMM's transition distribution is decomposed as follows. The harmonic function is chosen first, conditioned on the previous harmonic function. Then a value is chosen for the vector from the previous tonal space coordinate, conditioned on the choice of harmonic function. It is possible to compute the transition probability between any two states on the basis of this vector since between them the substitution coordinate of the first state and the substitution and block coordinates of the second are sufficient to compute the vector travelled between the two points. This way of constructing the transition distribution makes it insensitive to transposition.

$$\begin{aligned} P_{tr}(Sub_i, Blk_i, F_i | Sub_{i-1}, Blk_{i-1}, F_{i-1}) = \\ P(F_i|F_{i-1}) \times \\ P(vector(Sub_i, Sub_{i-1}, Blk_i)|F_i) \quad (3) \end{aligned}$$

The emission distribution is decomposed as follows. The distribution is once again made insensitive to absolute pitch by modelling the difference between the pitch

[5] Mean CPU time was 9:22 min (s.d. 33:32) with the supertagger, and 34:17 (s.d. 75:23) without, running on a 2.6GHz AMD Opteron 6212 CPU.

[6] Strictly speaking, this would only be a tritone substitution if it also had a dominant function.

class after accounting for substitution and the observed chord root, once again using the $\delta(x,y)$ function. The substitution is chosen conditioned on the function of the state. Then the chord type is chosen, conditioned on both the substitution and the function.

$$P_{em}(CT_i, CR_i|Sub_i, Blk_i, F_i) = P(\delta(CR_i, Sub_i)|F_i) \times P(CT_i|\delta(CR_i, Sub_i), F_i) \quad (4)$$

The baseline model is trained in the same way as the supertagger, only this time the training data is chord sequences paired with their annotated tonal space paths. We refer to the baseline model as HMMPATH. PCCG and ST+PCCG will completely fail to assign a path in cases where a full parse cannot be found. This may be because the beam removes all derivations that permit a grammatical interpretation of the full sequence, or, in the case of ST+PCCG, because the supertagger fails to suggest a set of lexical categories from which a full interpretation can be derived. HMMPATH will assign some path to any sequence, since it is not limited to returning grammatical interpretations.

8 Experiments

8.1 Evaluation

We evaluate all models on the basis of the "one-best" tonal space path to which they assign highest probability. Paths are first transformed from a list of tonal space coordinates to a list of vectors between adjacent points. This means that a path which makes an incorrect jump (for example, to an enharmonic equivalent of the correct point) is only penalized for that mistake and not for all subsequent points. Each point also has an associated harmonic function, which is included in the evaluation.

We align this path optimally with the gold-standard tonal space path from the annotated corpus (preprocessed in the same way) using the Levenshtein algorithm (Wagner and Fischer, 1974) for efficiently finding the optimal alignment between the elements of two sequences. We report precision, recall and f-score of the aligned paths. Precision is defined as the proportion of points returned by the model that correctly align with the gold standard. Recall is the proportion of points in the gold standard that are correctly retrieved by the model. F-score is the harmonic mean of these two measures.

$$P = Aligned/(Aligned + Inserted)$$

$$R = Aligned/(Aligned + Deleted)$$

$$F = 2 \times PR/(P+R)$$

Since the points of the path carry two pieces of information, the coordinate (now the step vector) and a chord function, we allow a score of 0.5 to be assigned to a correct alignment of only one of these and use a cost function in the Levenshtein algorithm that reflects this. Without this modification, a model that was, for example, very good at recognizing substitutions, but poor at identifying which chords were tonics would score very badly on precision and recall, since no alignment would be counted where only the coordinate was correct.

We shall refer to this metric as *tonal space edit distance* (TSED).

8.2 Model Comparison

All models were trained on the jazz corpus described above, containing 76 fully annotated sequences. It is common to divide a corpus into a *training set*, used to train models, and a *test set*, for experimental evaluation. Often a further division of a *development set* is used to obtain intermediate experimental results or determine model hyperparameters. Since the small size of the corpus prohibits holding out a test set, we use *10-fold cross-validation* here. Each experiment is run 10 times, with $\frac{9}{10}$ of the data used to train the model and the remaining $\frac{1}{10}$ used to evaluate the trained model. This means that all data is used for evaluation, but no model is tested on data that it was trained on. We report the results combined from all partitions. Since the same dataset has been used, for example, in preliminary trials of higher-order HMMs (see section 7.3), this experiment should be thought of as equivalent to an evaluation on a development set, rather than on completely unseen data. There is therefore some danger of overfitting.

9 Results

The results of the three experiments are reported in table 1. Differences between systems are tested for statistical significance using Bikel's stratified shuffling test[7], with 100,000 random shuffles of the results from individual chord sequences. Results are reported as statistically significant for $p < 0.05$.

Table 1 shows that ST+PCCG and PCCG produce high-precision results. This is because, unlike the base-

[7] `http://www.cis.upenn.edu/~dbikel/software.html`, accessed Oct 2012.

Model	*P (%)*	*R (%)*	*F (%)*	*Cov. (%)*
HMMPATH	77.44	84.87	80.98	100
PCCG	**92.29***	88.78	90.50*	97.37
ST+PCCG	90.18*	**92.79***	**91.46***	100

Table 1: Evaluation of each model's prediction using 10-fold cross-validation on the jazz corpus. Each model is scored using TSED (see section 8.1), reporting precision (P), recall (R), f-score (F) and coverage (Cov.), all percentages. The best results are bold and * marks significant improvements over the baseline ($p < 0.05$), measured by 100,000 iterations of stratified shuffling.

line HMMPATH, they can only produce results that are permitted by the grammar and fail when they can find no such result. The corpus used here for training and evaluation includes only sequences to which it was possible to assign a harmonic interpretation using the grammar. The results reported for the models that use the grammar are, therefore, higher than would be expected on real-life chord sequences. HMMPATH suffers from relatively low precision, but nevertheless succeeds in recovering a high proportion of the annotated harmonic relations, as measured by the overall f-score. If the parser were applied to a larger test set covering a less constrained musical domain, it would suffer from lack of coverage. In such a case, HMMPATH could be used in a simple form of *backoff*, providing an analysis were the parser fails. We could expect this to reduce the model's precision slightly, but improve its recall. The resulting ST+PCCG+HMMPATH model is robust in that it is guaranteed always to produce some tonal space interpretation, but in many cases benefits from the high precision of the parser.

The three key conclusions to draw from these results are as follows. First, they show that HMMPATH is a reasonable model to use as a baseline for the task and to back off to when no grammatical result can be found. Experiments on a larger data set would without doubt suffer more severely from a lack of coverage and HMMPATH could be used as suggested here or in another, less aggressive form of backoff. Second, the results show that the use of a grammar to constrain the interpretations predicted by an HMM improves substantially over the purely short-distance information captured by the baseline HMMPATH model. Third, they show that the use of the AST algorithm with a simple Markovian supertagging model succeeds in speeding up the parser by a large factor (roughly a factor of 4), with no reduction in accuracy. This can be thought of as using a Markovian model to suggest some interpretations of the chords, but building the final analysis by enforcing the structural constraints encoded in the grammar. Although this strategy is not essential for the present task, it is likely to be important for tasks requiring larger models or relying on accurate parsing under time constraints, when the gain in speed offered by supertagging will be critical.

10 Conclusion

The parser described above uses a formal grammar of a kind that is widely used for NLP, and a statistical parsing model of a kind typically used in wide-coverage natural language parsers, to map chord sequences onto their underlying harmonic analysis in the tonal space of Longuet-Higgins (1962a). The jazz harmony corpus we used is small, but experience with wide-cvoverage CCG parsing for NLP suggests that these techniques will scale to larger datasets and other musical domains (Clark and Curran, 2007; Auli and Lopez, 2011).

The parsing model is built using supervised learning over a small corpus of jazz chord sequences, hand-annotated with harmonic analyses. The fact that a grammar-based musical parser using a simple statistical parsing model trained on a small amount of labeled data is more accurate than a baseline Markovian model may be taken as further evidence suggesting that music and language have a common origin in distinctively human cognition.

The baseline model we have described is based only on statistics over a short window of context (bigrams). In most cases where the parser fails to find a full interpretation of a chord sequence, it does successfully identify large cadences, but cannot find an interpretation of certain difficult chords. One possible way of constructing a coherent analysis in difficult cases would be to identify high-confidence partial analyses produced by the parser and back off to a less constrained model, such as HMMPATH, only for those passages that proved difficult for the grammar-based model. This appears to be a reasonable emulation of what a human listener does on encountering a confusing passage of music, picking up the thread as soon as an easily identifiable tonal centre or cadence is heard.

We have described models to analyse sequences of chords expressed in the form of chord symbols. Such sequences assume a certain amount of preprocessing, including division into segments of constant harmony, selection of prominent notes and some analysis of chord root. A natural extension would be to construct a model incorporating these tasks into the analysis process, accepting note-level input (in MIDI encoding, for example) and suggesting possible interpretations in the way the supertagger component of our parsing model does. Recent work, not reported here, suggests that the approach presented here will generalize to this more difficult task.

Acknowledgements

We thank the referees for JNMR. The work was supported by ERC Advanced Fellowship 249520 GRAMPLUS.

References

Auli, M. and Lopez, A. (2011). Training a log-linear parser with loss functions via softmax-margin. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 333–343, Edinburgh. Association for Computational Linguistics.

Baroni, M., Brunetti, R., Callegari, L., and Jacoboni, C. (1984). A grammar for melody: Relationships between melody and harmony. In *Musical Grammars and Computer Analysis*, pages 201–2188, Modena. Olschki.

Baroni, M., Maguire, S., and Drabkin, W. (1983). The concept of musical grammar. *Music Analysis*, 2(2):175–208.

Bernstein, L. (1976). *The Unanswered Question: Six Talks at Harvard*. Harvard University Press, Cambridge, MA.

Bod, R. (2002). Memory-based models of melodic analysis: Challenging the gestalt principles. *Journal of New Music Research*, 31:27–37.

Chemillier, M. (2004). Grammaires, automates, et musique. In Briot, J.-P. and Pachet, F., editors, *Informatique musicale*, pages 195–230. Hermès.

Chew, E. (2000). *Towards a Mathematical Model of Tonality*. PhD thesis, MIT.

Choi, A. (2011). Jazz harmonic analysis as optimal tonality segmentation. *Computer Music Journal*, 35(2):49–66.

Clark, S. and Curran, J. R. (2007). Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33:493–552.

Cooke, D. (1959). *The Language of Music*. Oxford University Press, Oxford.

Cooper, G. and Meyer, L. B. (1963). *The Rhythmic Structure of Music*. University of Chicago Press, Chicago, IL, USA.

Daniélou, A. (1968). *The Rāga-s of Northern Indian Music*. Barrie & Rockliff, The Cresset Press, London.

de Haas, W. B., Rohrmeier, M., Veltkamp, R. C., and Wiering, F. (2009). Modeling harmonic similarity using a generative grammar of tonal harmony. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, pages 1–6, Kobe. International Society for Music Information Retrieval.

Euler, L. (1739). *Tentamen novae theoriae musicae ex certissismis harmoniae principiis dilucide expositae*. Saint Petersberg Academy.

Fabb, N. and Halle, M. (2012). Grouping in the stressing of words, in metrical verse, and in music. In Rebuschat, P., Rohrmeier, M., Hawkins, J. A., and Cross, I., editors, *Language and Music as Cognitive Systems*, pages 4–21. Oxford University Press.

Hamanaka, M., Hirata, K., and Tojo, S. (2006). Implementing A Generative Theory of Tonal Music. *Journal of New Music Research*, 35(4):249–277.

Helmholtz, H. (1862). *Die Lehre von den Tonempfindungen*. Vieweg, Braunschweig. trans. Alexander Ellis (1875, with added notes and appendices) as *On the Sensations of Tone*.

Hockenmaier, J. and Steedman, M. (2002). Generative models for statistical parsing with Combinatory Categorial Grammar. In *Proceedings of the 40th Meeting of the Association for Computational Linguistics*, pages 335–342, Philadelphia, PA. Association for Computational Linguistics.

Honingh, A. and Bod, R. (2005). Convexity and well-formedness of musical objects. *Journal of New Music Research*, 34:293–303.

Huron, D. (2006). *Sweet Anticipation: Music and the Psychology of Music*. MIT Press, Cambridge, MA, USA.

Jeans, J. (1937). *Science and Music*. Cambridge University Press, Cambridge.

Johnson-Laird, P. N. (1991). Jazz improvisation: a theory at the computational level. In Howell, P., West, R., and Cross, I. J., editors, *Representing Musical Structure*, pages 291–326. Academic Press Ltd., San Diego, CA.

Johnson-Laird, P. N., Kang, O. E., and Leong, Y. C. (2012). On musical dissonance. *Music Perception*, 30(1):19–35.

Katz, J. and Pesetsky, D. (2011). The identity thesis for language and music. `http://ling.auf.net/lingBuzz/000959`.

Keiler, A. (1978). Bernstein's "The Unanswered Question" and the problem of musical competence. *The Musical Quarterly*, 64(2):195–222.

Keiler, A. (1981). Two views of musical semiotics. In Steiner, W., editor, *The Sign in Music and Literature*, pages 138–168. University of Texas Press, Austin TX.

Krumhansl, C. (1990). *Cognitive Foundations of Musical Pitch*. Oxford University Press, Oxford.

Lehne, M., Rohrmeier, M., Gollmann, D., and Koelsch, S. (2013). The influence of different structural features on felt musical tension in two piano pieces by mozart and mendelssohn. *Music Perception*, 31:171–185.

Lerdahl, F. (2001). *Tonal pitch space*. Oxford University Press, New York, NY.

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

Lindblom, B. and Sundberg, J. (1969). Towards a generative theory of melody. *Swedish Journal of Musicology*, 10(4):53–86.

Longuet-Higgins, H. C. (1962a). Letter to a musical friend. *The Music Review*, 23:244–248.

Longuet-Higgins, H. C. (1962b). Second letter to a musical friend. *The Music Review*, 23:271–280.

Longuet-Higgins, H. C. (1976). The perception of melodies. *Nature*, 263:646–653.

Longuet-Higgins, H. C. and Steedman, M. (1971). On interpreting Bach. *Machine Intelligence*, 6:221–241.

Marcus, M., Santorini, B., and Marcinkiewicz, M. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19:313–330.

Margulis, E. H. (2005). A model of melodic expectation. *Music Perception*, 22:663–714.

Marsden, A. (2010). Schenkerian analysis by computer: A proof of concept. *Journal of New Music Research*, 39(3):269–289.

Mathews, M. and Pierce, J. (1989). The bohlen-pierce scale. In Mathews, M. and Pierce, J., editors, *Current Directions in Computer Music Research*, pages 165–173. MIT Press, Cambridge, MA, USA.

Meyer, L. B. (1956). *Emotion and Meaning in Music*. University of Chicago Press, Chicago, IL.

Narmour, E. (1977). *Beyond Schenkerism*. University of Chicago Press, Chicago, IL.

Pachet, F. (2000). Computer analysis of jazz chord sequences: Is Solar a blues? In Miranda, E., editor, *Readings in Music and Artificial Intelligence*, pages 85–113. Harwood Academic Publishers.

Pearce, M. and Wiggins, G. (2006). Expectation in melody: The influence of context and learning. *Music Perception*, 23:377–405.

Piston, W. (1949). *Harmony*. Victor Gollancz, London.

Rameau, J. P. (1722). *Traité de l'harmonie*. Jean-Baptiste-Christophe Ballard.

Rebuschat, P., Rohrmeier, M., Hawkins, J., and Cross, I., editors (2011). *Language and music as cognitive systems*. Oxford University Press.

Riemann, H. (1893). *Vereinfachte Harmonielehre oder die Lehre von den tonalen Funktionen der Akkorde*. Augener & Co. trans. H. Bemerunge, as *Harmony Simplified, or the Theory of the Tonal Functions of Chords*.

Riemann, H. (1914). Ideen zu einer Lehre von den Tonvorstellungen. *Jahrbuch der Bibliothek Peters*, 21:1–26.

Rohrmeier, M. (2011). Towards a generative syntax of tonal harmony. *Journal of Mathematics and Music*, 5:35–53.

Rohrmeier, M. and Cross, I. (2009). Tacit tonality: Implicit learning of context-free harmonic structure. In *Proceedings of the 7th Triennial Conference of the European Society for the Cognitive Sciences of Music*, Jyväskylä. European Society for the Cognitive Sciences of Music.

Schenker, H. (1906). *Harmony*. University of Chicago Press, Chicago, IL. trans. E. Borgese.

Simon, H. and Sumner, R. (1968). Pattern in music. In Kleinmuntz, B., editor, *Formal Representations of Human Judgment*, pages 219–250. Wiley, New York.

Smoliar, S. W. (1976). Music programs: An approach to music theory through computational linguistics. *Journal of Music Theory*, 20(1):105–131.

Srinivas, B. and Joshi, A. (1994). Disambiguation of super parts of speech (or supertags): Almost parsing. In *Proceedings of the International Conference on Computational Linguistics*, Kyoto. Association for Computational Linguistics.

Steedman, M. (1984). A generative grammar for jazz chord sequences. *Music Perception*, 2:52–77.

Steedman, M. (1996). The blues and the abstract truth: Music and mental models. In Garnham, A. and Oakhill, J., editors, *Mental Models in Cognitive Science*, pages 305–318. Erlbaum.

Steedman, M. (2000). *The Syntactic Process*. MIT Press, Cambridge, MA.

Sundberg, J., Nord, L., and Carlson, R. (1991). *Music, language, speech, and brain (Wenner-Gren Symposium)*. Macmillan Press, Stockholm.

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

Temperley, D. (2007). *Music and Probability*. MIT Press, Cambridge, MA.

Tillmann, B. and Bigand, E. (2004). The relative importance of local and global structures in music perception. *The Journal of Aesthetics and Art Criticism*, 62(2):pp. 211–222.

Tymoczko, D. (2006). The geometry of musical chords. *Science*, 313:72–74.

Tymoczko, D. (2011). *A Geometry of Music: Harmony and Counterpoint in the Extended Common Practice*. Oxford University Press, Oxford.

Wagner, R. A. and Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the ACM*, 21(1):168–173.

Winograd, T. (1968). Linguistics and the computer analysis of tonal harmony. *Journal of Music Theory*, 12:2–49.

Appendix: Tonal Space Semantics

We introduce in the present paper an adaptation of CCG for grammars of tonal harmony. The formalism acts as a mechanism to map a surface – chords, in our case – onto a semantic interpretation – a tonal harmonic analysis. Each syntactic category is coupled with a logical form and, as syntactic categories are combined during parsing, a logical form representing the full harmonic analysis is built up.

We mentioned above that a logical form is constructed to represent a harmonic analysis in terms of movements about Longuet-Higgins' tonal space, but omitted the details of the representation we use. Here we set out the details of a representation suitable for our tonal semantics.

Tonic Semantics

The semantics of a tonic is a point in the tonal space. It is underspecified – it only specifies a point within an *enharmonic block* (see figure 18). It is therefore a coordinate between $\langle 0,0\rangle$ and $\langle 3,2\rangle$ and each coordinate denotes a different infinite set of positions in the space. Crucially, however, in the context of a full harmonic analysis, the coordinate represents a single point in the space, as we will see later in this appendix.

A single tonic chord receives as its logical form a single-element list containing such a coordinate. A logical form of this sort is associated with atomic lexical categories, such as both the occurences of C^T in figure 11.

Cadence Semantics

The semantics of a cadence step is a predicate representing a movement in the tonic space. An extended cadence is interpreted as the recursive application of each movement to its resolution.

Authentic cadences – left steps – use the *leftonto* predicate and plagal cadences – right steps – the *rightonto* predicate. For example, a single dominant chord resolving to a tonic $\langle 0,0\rangle$ would receive the logical form $\mathit{leftonto}(\langle 0,0\rangle)$, whilst a secondary dominant, resolving to a dominant, resolving to the tonic would receive $\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))$.

We define *leftonto* (and likewise *rightonto*) as being subject to a reduction when applied to a list, as in the case of a tonic resolution, as follows:

$$\mathit{leftonto}([X_0, X_1, ...]) \Rightarrow [\mathit{leftonto}(X_0), X_1, ...]$$

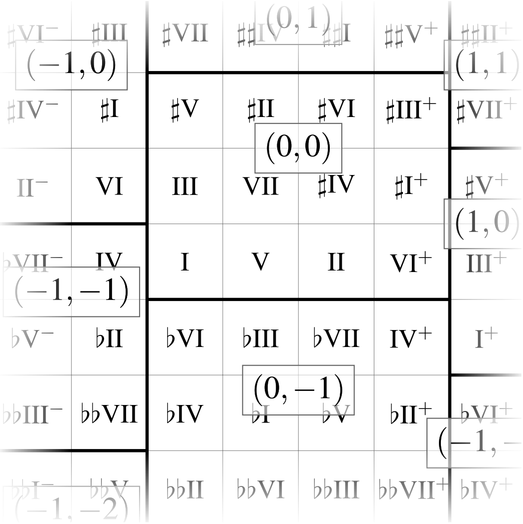

♯VI⁻ ♯III ♯VII ♯♯IV ♯♯I ♯♯V⁺ ♯♯II⁺
(0,1)
(−1,0)
(1,1)
♯IV⁻ ♯I ♯V ♯II ♯VI ♯III⁺ ♯VII⁺
(0,0)
II⁻ VI III VII ♯IV ♯I⁺ ♯V⁺
(1,0)
♭VII⁻ IV I V II VI⁺ III⁺
(−1,−1)
♭V⁻ ♭II ♭VI ♭III ♭VII IV⁺ I⁺
(0,−1)
♭♭III⁻ ♭♭VII ♭IV ♭I ♭V ♭II⁺ ♭VI⁺
(−1,−
♭♭I⁻ ♭♭V ♭♭II ♭♭VI ♭♭III ♭♭VII⁺ ♭IV⁺
(−1,−2)

Figure 18: Enharmonic blocks at the centre of the space. Each position within these 4x3 blocks is equated by equal temperament with the same position within every other block.

$$\begin{array}{ccc}
\mathrm{IIm}^7 & \mathrm{V}^7 & \mathrm{I} \\
\hline
\lambda x.\mathit{leftonto}(x) & \lambda x.\mathit{leftonto}(x) & [\langle 0,0\rangle] \\
 & \multicolumn{2}{c}{\overline{[\mathit{lefonto}(\langle 0,0\rangle)]}^{>}} \\
\multicolumn{3}{c}{\overline{[\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))]}^{>}}
\end{array}$$

Figure 19: Recursive interpretation of an extended authentic cadence, showing logical forms, but omitting syntactic types.

The example in figure 19 shows a two-step cadence – the familiar *IIm*^{7} *V*^{7} *I*. The derivation shows the combination of the semantics of each chord into the semantics for the sequence.

Throughout this chapter, derivations like this are written with the syntactic part of each syntactic type/logical form pair omitted. Naturally, these are all derivations that would be permitted by the syntactic types associated with these logical forms under the combinatory rules described in section 6.2 in the main article.

The recursive application of multiple cadence steps can be combined ahead of time, before their application to their resolution, using the composition operator,

$$\begin{array}{ccc}
\mathrm{IIm}^7 & \mathrm{V}^7 & \mathrm{I} \\
\hline
\lambda x.leftonto(x) & \lambda x.leftonto(x) & [\langle 0,0\rangle] \\
\hline
\multicolumn{2}{c}{\lambda x.leftonto(leftonto(x)) \quad >\mathbf{B}} & \\
\hline
\multicolumn{3}{c}{[leftonto(leftonto(\langle 0,0\rangle))] \quad >}
\end{array}$$

Figure 20: Recursive interpretation of an extended cadence, derived using the function composition combinator to combine the unresolved cadence before its resolution is encountered.

$$\begin{array}{ccc}
\mathrm{I} & \mathrm{IV} & \mathrm{I} \\
\hline
\lambda x.x & \lambda x.x & [\langle 0,0\rangle] \\
\hline
 & \multicolumn{2}{c}{[\langle 0,0\rangle] \quad >} \\
\hline
\multicolumn{3}{c}{[\langle 0,0\rangle] \quad >}
\end{array}$$

Figure 21: *I IV I* colouration of a tonic chord, interpreted as contributing nothing to the logical form.

associated with the composition combinator.

$$f \circ g \equiv \lambda x.f(g(x))$$

Figure 20 shows again the interpretation seen in figure 19, now produced by a derivation that uses the composition combinator.

Colouration Semantics

The lexicon includes some categories for interpreting colouration chords, which contribute nothing much to the functional structure of the harmony, but spice up the realisation. Accordingly, these are given an empty semantics (that is, the identity function), which simply ignores them.

A typical example of this is the sequence *I IV I*, often played during long passages of a *I* chord. This is really a form of plagal cadence and a fine grained analysis might treat it as such. However, for most analysis purposes we wish to ignore this very brief excursion from the tonic. Figure 21 shows an example derivation using this empty semantics.

In many cases, we do not even return to the tonic after our excursion, continuing with a cadence straight after the *IV*. This is the purpose of the backward-facing colouration lexical category (Colour-IVb in figure 14) and the semantics ignores the *IV* in the same way.

$$\begin{array}{cccc}
\mathrm{I} & \mathrm{IIm}^7 & \mathrm{V}^7 & \mathrm{I} \\
\hline
[\langle 0,0\rangle] & \lambda x.leftonto(x) & \lambda x.leftonto(x) & [\langle 0,0\rangle] \\
\hline
 & & \multicolumn{2}{c}{[lefonto(\langle 0,0\rangle)] \quad >} \\
\hline
 & \multicolumn{3}{c}{[leftonto(leftonto(\langle 0,0\rangle))] \quad >} \\
\hline
\multicolumn{4}{c}{[\langle 0,0\rangle, leftonto(leftonto(\langle 0,0\rangle))] \quad \mathbf{dev}}
\end{array}$$

Figure 23: A single tonic chord combined with a subsequent resolved recursive cadence using the development rule.

Development Semantics

The development combinatory rule combines sequences of tonic passages and resolved cadences into larger units, ultimately into a whole piece of music. Every logical form introduced so far has been a single-item list. The behaviour of the development rule's semantics is rather trivial. It simply concatenates its two arguments: the syntax ensures these are lists. The example in figure 22 shows a pair of resolved cadences being combined in this way. Figure 23 shows a derivation in which a single tonic combines with a subsequent resolved cadence.

Coordination Semantics

Logical forms representing unresolved cadences can be *coordinated* to share their eventual resolution. This is carried out by the special musical *coordination* combinator. The semantics of this combinator simply conjoins the cadence logical forms using the $\wedge$ operator. Note that, unlike in the logical semantics of natural language, this conjunction operator must preserve the order of its arguments.

$$A \wedge B \not\equiv B \wedge A$$

We can also reduce brackets to reflect the associativity of the conjunction operator.

$$\begin{aligned}
A \wedge B \wedge C &\equiv (A \wedge B) \wedge C \\
&\equiv A \wedge (B \wedge C)
\end{aligned}$$

$$A \wedge (B \wedge C) \Rightarrow A \wedge B \wedge C$$

$$(A \wedge B) \wedge C \Rightarrow A \wedge B \wedge C$$

The functions that denote cadences are simply conjoined by $\wedge$, as shown in figure 24.

The result is treated as a function that can be applied to its resolution. It reduces under application to a list in

$$\dfrac{\dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x))} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x))}}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x)) \wedge \lambda x.\mathit{leftonto}(\mathit{leftonto}(x))}\,\&$$

Figure 24: Two unresolved cadences combined using the coordination combinator.

$$\dfrac{\dfrac{\dfrac{\dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))}}{\lambda x.L(L(x)) \wedge \lambda x.L(L(x))}\,\& \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))}}{\lambda x.L(L(x)) \wedge \lambda x.L(L(x)) \wedge \lambda x.L(L(x))}\,\& \quad \dfrac{\text{I}}{[\langle 0,0\rangle]}}{[(\lambda x.L(L(x)) \wedge \lambda x.L(L(x)) \wedge \lambda x.L(L(x)))(\langle 0,0\rangle)]}\,>$$

Figure 26: More than two unresolved cadences can be combined using the coordination combinator.

the same way as *leftonto* and *rightonto*. An example is shown in figure 25. Note that the individual cadences are not actually applied to the resolution. More than two cadences can be coordinated to share the same resolution, as shown in figure 26. (The predicate *leftonto* is henceforth abbreviated to L to save space.)

The result of a coordination (once applied to its resolution) can become the recursive resolution of a prior cadence step (as in figure 27). However, this logical form will result in the same tonal space path as that which would have been produced by composing the VI^7 with the following IIm^7 V^7 before coordinating, shown in figure 28.

We therefore define the following equivalence in the logical forms and by convention reduce the left-hand side form to the right-hand side wherever possible.

$$A((B \wedge ...)(C)) \Rightarrow (A \circ B \wedge ...)(C)$$

Extracting the Tonal Space Path

The logical forms that come out of the above semantics represent certain constraints on paths through the tonal space. Although the tonic points are ambiguous in the representation, every point of a path can be inferred from a full logical form.

Let us first examine the constraints encoded in the various types of predicate. The most obvious constraint is on the point created by a left (or right) movement, denoted in the semantics by *leftonto* (or *rightonto*) predicates. In *leftonto*(p), the point at which the movement

$$\dfrac{\dfrac{\text{VI}^7}{\lambda x.L(x)} \quad \dfrac{\dfrac{\dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))}}{\lambda x.L(L(x)) \wedge \lambda x.L(L(x))}\,\& \quad \dfrac{\text{I}}{[\langle 0,0\rangle]}}{[(\lambda x.L(L(x)) \wedge \lambda x.L(L(x)))(\langle 0,0\rangle)]}\,>}{[L((\lambda x.L(L(x)) \wedge \lambda x.L(L(x)))(\langle 0,0\rangle))]}\,>$$

Figure 27: A recursive dominant chord may be applied to the result of using the coordination combinator. (C.f. figure 28.)

$$\dfrac{\dfrac{\dfrac{\dfrac{\text{VI}^7}{\lambda x.L(x)} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))}}{\lambda x.L(L(L(x)))}\,>\mathbf{B} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.L(L(x))}}{\lambda x.L(L(L(x))) \wedge \lambda x.L(L(x))}\,\& \quad \dfrac{\text{I}}{[\langle 0,0\rangle]}}{[(\lambda x.L(L(L(x))) \wedge \lambda x.L(L(x)))(\langle 0,0\rangle)]}\,>$$

Figure 28: An alternative derivation of the cadence in figure 27, resulting in an interpretation identical in the tonal space, leading to the definition equivalence of the two logical forms.

begins must be one step in the grid to the right of the first point of the path p. If the point (x,y) is fully specified, the whole path *leftonto*(*leftonto*$((x,y))$) is therefore also unambiguous.

Two cadences that share a resolution through coordination are constrained to end at the same point, since their points are constrained relative to their shared resolution.

There is no obvious constraint between items in the top-level list of tonics and resolved cadences. We assume that a step is made to the nearest (most closely tonally related) point that satisfies all other constraints. For example, take the following two logical forms:

1. $[\langle 0,0\rangle, \mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))]$
2. $[\langle 0,0\rangle, \mathit{leftonto}(\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle)))]$

The tonal space paths for these logical forms are shown in figure 29. The start of the second item in path 1 is dependent, ultimately, on the cadence resolution $\langle 0,0\rangle$. But this point is underspecified: we can choose for it any of the infinite points that lie at $\langle 0,0\rangle$ within their enharmonic block. Given an arbitrary choice of the first item's point at the central $(0,0)$, we will choose the same point for the end of the second item, since it puts the start of the second item (now $(2,0)$) as close as possible to $(0,0)$. A choice of $(-4,1)$ for the end point would also have been permitted by other constraints, but would have resulted in a larger jump between the two path fragments.

In path 2, however, the second item begins at a point further from its ending. In this case we will choose $(-1,1)$ as the start point for the second item by setting the $\langle 0,0 \rangle$ at its end to be at $(-4,1)$.

Note that the choice of the first point on the path is unimportant: two paths identical in form, but occurring at different positions in the space can be considered equivalent, since the only difference between them is their absolute pitch and we (uncontraversially) consider precise absolute pitch not to be pertinent to musical semantics. In both the above examples, we could have chosen $(4,-1)$, for instance, as the coordinate $\langle 0,0 \rangle$ at the start and the resulting paths would be considered identical to those we derived.

A simple algorithm can be constructed by means of a recursive transformation of the logical predicates to produce the flat tonal space path represented by a logical form produced by parsing. As well as demonstrating that any logical form is interpretable as an analysis in the tonal space, there are circumstances in which this transformation is of use. In section 8.1 in the main article we use path similarity between an output interpretation and the gold standard as an evaluation metric. The paths we compare are those produced by this algorithm.

The Extended Example: Full Analysis

As a full, real life example, the derivation in figure 30 shows the long extended cadence from *Alice in Wonderland* of which a purely syntactic derivation was shown in figure 16, now including logical forms.

$$\dfrac{\dfrac{\dfrac{\text{IIm}^7}{\lambda x.\mathit{leftonto}(x)} \quad \dfrac{\dfrac{\text{V}^7}{\lambda x.\mathit{leftonto}(x)} \quad \dfrac{\text{I}}{[\langle 0,0\rangle]}}{[\mathit{lefonto}(\langle 0,0\rangle)]}{\scriptstyle >}}{[\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))]}{\scriptstyle >} \quad \dfrac{\dfrac{\text{V}^7}{\lambda x.\mathit{leftonto}(x)} \quad \dfrac{\text{I}}{[\langle 0,0\rangle))]}}{[\mathit{lefonto}(\langle 0,0\rangle)]}{\scriptstyle >}}{[\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle)),\mathit{lefonto}(\langle 0,0\rangle)]}\,\textsf{dev}$$

Figure 22: A pair of fully resolved cadences combined using the trivial development rule.

$$\dfrac{\dfrac{\dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x))} \quad \dfrac{\text{IIm}^7\ \text{V}^7}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x))}}{\lambda x.\mathit{leftonto}(\mathit{leftonto}(x)) \wedge \lambda x.\mathit{leftonto}(\mathit{leftonto}(x))}\,\& \quad \dfrac{\text{I}}{[\langle 0,0\rangle]}}{[(\lambda x.\mathit{leftonto}(\mathit{leftonto}(x)) \wedge \lambda x.\mathit{leftonto}(\mathit{leftonto}(x)))(\langle 0,0\rangle)]}{\scriptstyle >}$$

Figure 25: The two unresolved cadences that were combined in figure 24 are combined with the resolution expected by both using the function application combinator.

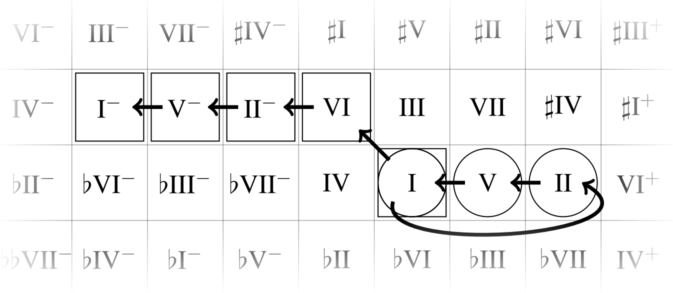

VI⁻ III⁻ VII⁻ ♯IV⁻ ♯I ♯V ♯II ♯VI ♯III⁺
IV⁻ I⁻ V⁻ II⁻ VI III VII ♯IV ♯I⁺
♭II⁻ ♭VI⁻ ♭III⁻ ♭VII⁻ IV I V II VI⁺
♭♭VII⁻ ♭IV⁻ ♭I⁻ ♭V⁻ ♭II ♭VI ♭III ♭VII IV⁺

Figure 29: The tonal space paths corresponding to two logical forms. $[\langle 0,0\rangle, \mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))]$ (circles) begins at *I*, $(0,0)$, jumps to *II*, $(2,0)$, and left-steps back to *I*. $[\langle 0,0\rangle, \mathit{leftonto}(\mathit{leftonto}(\mathit{leftonto}(\mathit{leftonto}(\langle 0,0\rangle))))]$ (squares) also begins at *I*, but jumps to VI, $(-1,1)$, and left-steps to I^-, $(-4,1)$.

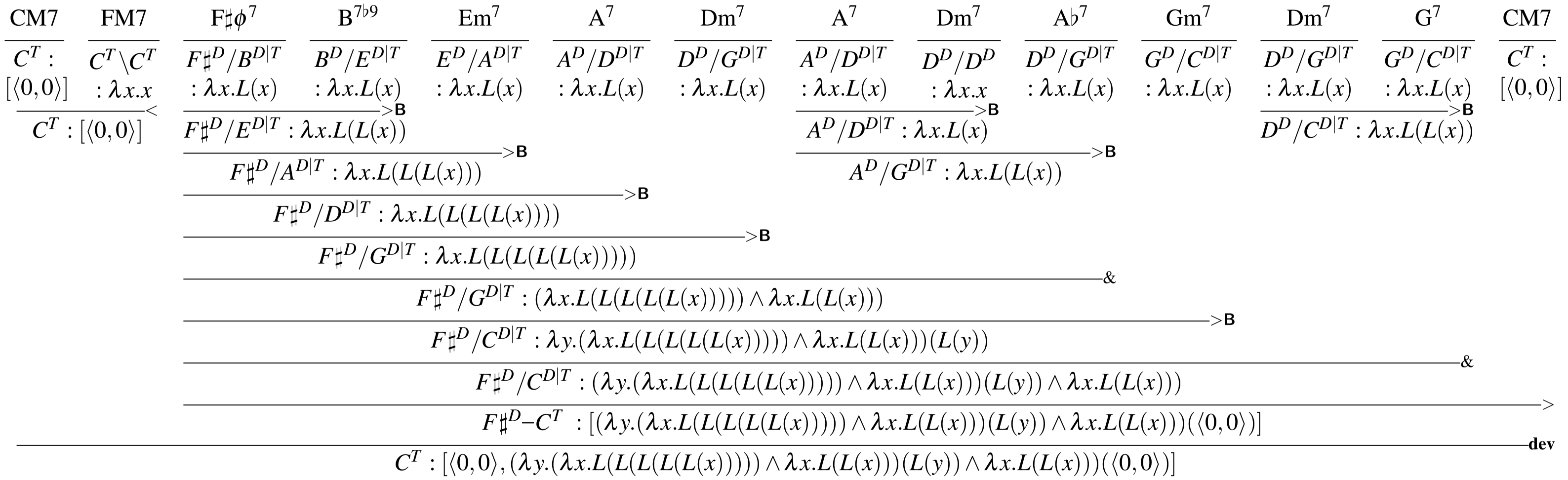

CM7 FM7 F♯ϕ7 B7♭9 Em7 A7 Dm7 A7 Dm7 A♭7 Gm7 Dm7 G7 CM7
C^T : [⟨0,0⟩]
C^T\C^T : λx.x
F♯^D/B^{D|T} : λx.L(x)
B^D/E^{D|T} : λx.L(x)
E^D/A^{D|T} : λx.L(x)
A^D/D^{D|T} : λx.L(x)
D^D/G^{D|T} : λx.L(x)
A^D/D^{D|T} : λx.L(x)
D^D/D^D : λx.x
D^D/G^{D|T} : λx.L(x)
G^D/C^{D|T} : λx.L(x)
D^D/G^{D|T} : λx.L(x)
G^D/C^{D|T} : λx.L(x)
C^T : [⟨0,0⟩]
<
C^T : [⟨0,0⟩]
>B
F♯^D/E^{D|T} : λx.L(L(x))
>B
A^D/D^{D|T} : λx.L(x)
>B
D^D/C^{D|T} : λx.L(L(x))
>B
F♯^D/A^{D|T} : λx.L(L(L(x)))
>B
A^D/G^{D|T} : λx.L(L(x))
>B
F♯^D/D^{D|T} : λx.L(L(L(L(x))))
>B
F♯^D/G^{D|T} : λx.L(L(L(L(L(x)))))
&
F♯^D/G^{D|T} : (λx.L(L(L(L(L(x))))) ∧ λx.L(L(x)))
>B
F♯^D/C^{D|T} : λy.(λx.L(L(L(L(L(x))))) ∧ λx.L(L(x)))(L(y))
&
F♯^D/C^{D|T} : (λy.(λx.L(L(L(L(L(x))))) ∧ λx.L(L(x)))(L(y)) ∧ λx.L(L(x)))
>
F♯^D–C^T : [(λy.(λx.L(L(L(L(L(x))))) ∧ λx.L(L(x)))(L(y)) ∧ λx.L(L(x)))(⟨0,0⟩)]
dev
C^T : [⟨0,0⟩, (λy.(λx.L(L(L(L(L(x))))) ∧ λx.L(L(x)))(L(y)) ∧ λx.L(L(x)))(⟨0,0⟩)]

Figure 30: Semantic derivation of the extended cadence from *Alice in Wonderland*.