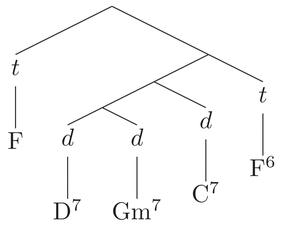
A Statistical Parser for Jazz Chord Sequences

Mark Granroth-Wilding
School of Informatics, University of Edinburgh
mark.granroth-wilding@ed.ac.uk

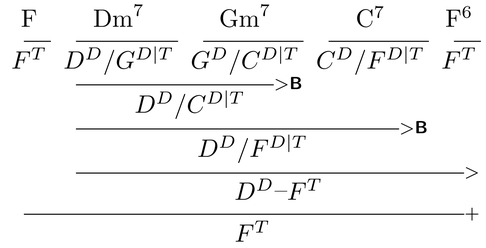
Mark Steedman
School of Informatics, University of Edinburgh
steedman@inf.ed.ac.uk



Hierarchical structure similar to the that of syntax in language can be identified in musical meter and harmony. Analysing these aspects of music involves deriving structure from an unstructured sequence of notes. Doing so is important for many music processing tasks, such as key-identification and score transcription.

The 3D tonal space described by Longuet-Higgins formalises the theory underlying tonal harmony. We apply techniques from natural language processing to the problem of processing musical harmony. We use a formal grammar of jazz chord sequences and wide-coverage parsing methods commonly used for analysing sentences to analyse harmonic structure in terms of the tonal space.

Using a small corpus of jazz chord sequences annotated with harmonic analyses, we show that the grammar can be used with simple statistical parsing to improve the quality of the analysis produced by a purely stochastic model.

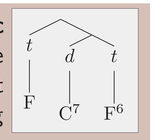


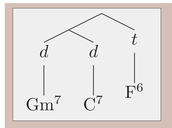
Harmonic analysis as syntax

Harmonic analysis involves inferring the **structure underlying the harmony** given the notes of a piece of music. It is analogous to parsing the **syntax** of a sentence. Given a surface form of musical notes, the harmonic structure is highly ambiguous.

Analysing **chord sequences** is easier: the music has been divided into chunks and the important notes have been selected. However, much ambiguity still remains. For now we handle chord sequences as input.

Chords are analysed as having a **function**: **dominant**, **tonic** or **subdominant**. A dominant chord creates a tension, leaving the listener expecting its **resolution** – a tonic chord rooted a perfect fifth below. This tension-resolution pattern is the basic building block of harmonic syntactic structure.



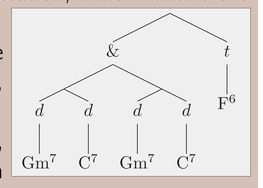


The dominant chord can be **tail recursive**: the resolution itself can function as a dominant and resolve again.

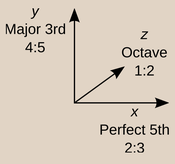
It can be **coordinated**: two consecutive dominant chords (or sequences) can share the same resolution, which follows the second one.

It can be **substituted**: in certain contexts, one dominant can be replaced by a chord on a different root, which plays the same role.

A **subdominant** chord behaves in a similar way, but resolves to a tonic chord rooted a perfect fourth below.

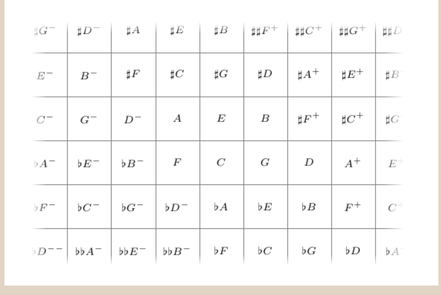


Musical semantics



The system of pitches used in Western tonal music is based on relations between the low components of the **harmonic series**. All intervals between notes are defined by ratios of small integers. In particular, the relations between notes can be expressed in terms of the first three distinct intervals in the harmonic series.

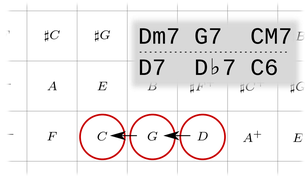
Longuet-Higgins (1979) formalised this system in a **three-dimensional infinite discrete space** of pitches. Ignoring octaves, we project this onto a **two-dimensional space**. A portion is shown right.



Equal temperament distorts the intervals to produce 12 semitones spaced equally over an octave and maps the plane to a torus, cycling in both dimensions.

We treat the justly intoned relations between notes or chords as seen in the tonal space as the **'semantics' of the music**, analogous to the denotational semantics of a sentence.

Two chord sequences that share the same semantics.
The semantics is represented as the path through the tonal space traced by the underlying chord roots.



Musical grammar

We use a linguistic grammar formalism, **combinatory categorial grammar (CCG)**, to model the syntactic structure of harmony. A syntactic tree is built from the chords up, constrained by categories assigned to the chords.

Our notation is based on standard CCG, modified for harmonic syntax.

$\frac{\mathrm{D}^7\ \mathrm{Gm}^7\ \mathrm{C}^7\ \mathrm{F}}{D^D–F^T}$ A category reflects the tonality of a span of music. It contains the (equal temperament) chord root on which the music starts and ends (*left*), and the chord's function.

The simplest category is the **tonic chord** (*right*). It starts and ends on the same root (abbreviated to one symbol).

$\frac{\mathrm{FM7}}{F^T}$

$\frac{\mathrm{C}^7}{C^D/F^T}$ A **dominant chord** (*right*) receives a forward-facing slash category, representing the **expectation of its resolution** to the following chord. This category will be able to combine with one that follows if it starts with the right chord root (here F).

In general, both tonic and dominant resolution will be permitted, allowing extended cadences to be interpreted (see the example above).

We add to each category a **compositional semantics** representing **points and movements** in the tonal space. Building the syntactic tree also builds a full interpretation of the chords as a path through the space.

Modelling

As with natural language grammars, **lexical ambiguity** makes full parsing infeasible. We apply **supervised statistical parsing** techniques from NLP.

ST+PARSE: We use the **supertagging** approach of Clark (2002) to restrict the set of possible interpretations of each chord on the basis of a short-distance model of its context (an **n-gram** model). These categories are then parsed to find a full interpretation of the sequence, if possible.

NGRAM: For comparison, we train a similar n-gram model which, instead of assigning a category to each chord, directly predicts a point in the tonal space.

ST+PARSE+NGRAM: A third model attempts to analyse the sequence using the supertagger and parser and uses the result given by the pure n-gram model if no full parse can be found (i.e. as a backoff model).

ST+PCFG+NGRAM: Finally, we use a generative statistical parsing model after Hockenmaier (2001) to guide the derivation produced given the supertagger's chosen categories. As above, this backs off to the n-gram model in the few cases where no parse can be found.

We use a small annotated corpus of jazz standards chord sequences to train and test the models.

Results

We report the precision and recall of the predicted path points, compared against the hand-labelled gold standard.

(Results removed from poster for publication in proceedings)

Unsurprisingly, ST+PARSE's results have high precision, since they are restricted to grammatical interpretations. The recall is low because the parser finds a result for only 75% of sequences. NGRAM finds a result for every sequence, but the results are of poorer quality.

ST+PARSE+NGRAM has a lower precision than ST+PARSE, because it uses the poorer quality result wherever it cannot find a grammatical one. It has a **higher f-score** than the first two. This shows (a) that NGRAM provides a **reasonable backoff model** for when ST+PARSE cannot find a grammatical result; and (b) that using the grammar to restrict interpretations improves the quality of the results.

ST+PCFG+NGRAM improves further on recall and f-score, since the parsing model guides the derivation so that it can find better interpretations and a better ranking of the results. The parser benefits from the fact that the statistical model trained on counts of mostly **local** dependencies is correctly projected onto **non-local** dependencies by the grammar.

Funded by ERC Advanced Fellowship 249520 GRAMPLUS and ESRC Postgraduate Studentship ES/H012648/1.

Clark, S. (2002). Supertagging for combinatory categorial grammar. *Proceeding of the Sixth International Workshop on Tree Adjoining Grammar and Related Frameworks*, (pp. 101–106).

Hockenmaier, J. (2001). Statistical parsing for CCG with simple generative models. In *Association for Computational Linguistics 39th annual meeting and 10th conference of the European Chapter*, vol. 39, (pp. 7–12).

Longuet-Higgins, C. (1979). The perception of music. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, *205*(1160), 307–322.