ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Towards a Unified Model for Harmony and Voice-Leading

Robert Lieck
robert.lieck@epfl.ch

Daniel Harasim
daniel.harasim@epfl.ch

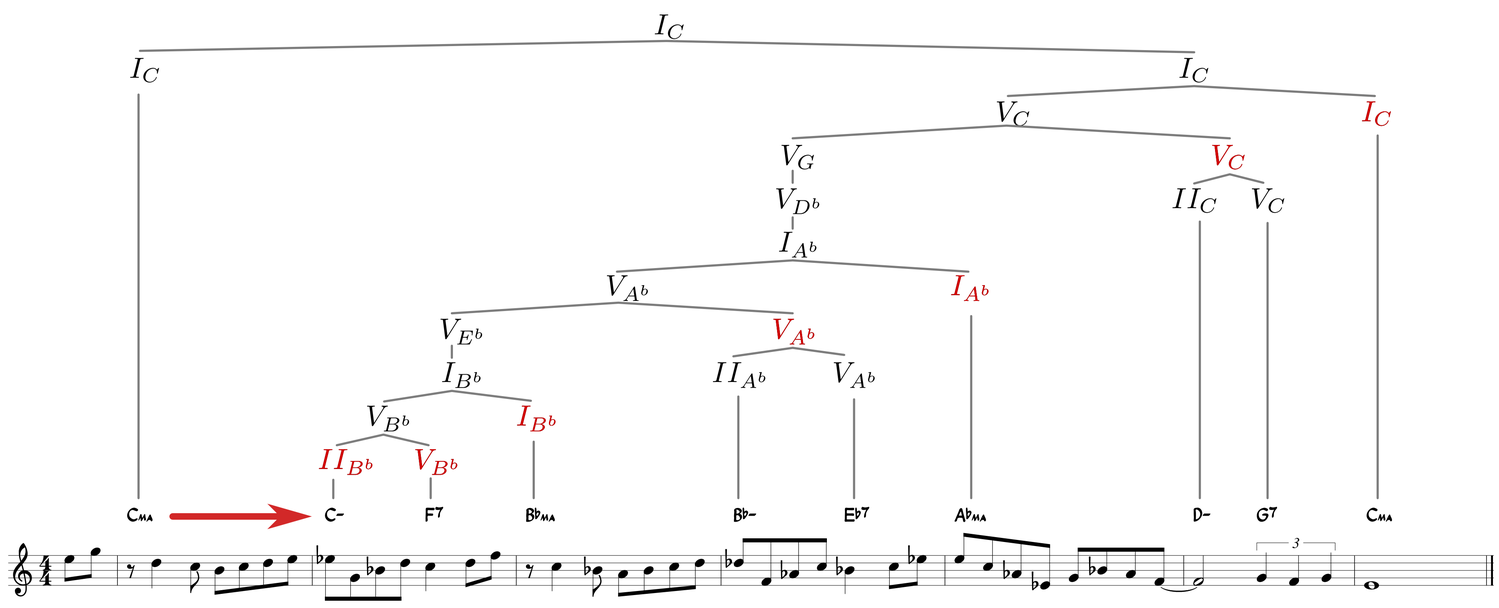
Martin Rohrmeier
martin.rohrmeier@epfl.ch

Digital and Cognitive Musicology Lab

Abstract

Harmony and voice-leading follow considerably different principles. Harmony describes potentially long-ranging dependencies and has mainly been modelled using *context-free grammars* (CFGs). Voice-leading, on the other hand, is a predominantly local phenomenon and has been addressed using sequential models such as *regular grammars* (RGs) or n-gram models. While combining CFGs and RGs is theoretically possible it is computationally intractable.

We show how parsing and generating sequences can still be solved efficiently. More generally, our approach allows to combine a CFG with an arbitrary sequence model. Additionally, we describe CFGs and a large class of sequence models in a unified framework that allows to train models based on musical corpora.



Intersection Grammar

The probability of a sequence s under the intersection grammar $H = G_{\mathrm{CFG}} \cap G_{\mathrm{RG}}$ is proportional to the product of probabilities under the *context-free grammar* (CFG) G_{CFG} and the *regular grammar* (RG) G_{RG}

$$p_{\mathrm{H}}(s) = \frac{1}{Z} p_{\mathrm{CFG}}(s)\, p_{\mathrm{RG}}(s) \tag{1}$$

with normalisation constant Z. Likewise, the probability of a syntax tree t is

$$p_{\mathrm{H}}(t) = \frac{1}{Z} p_{\mathrm{CFG}}(t)\, p_{\mathrm{RG}}(s \Leftarrow t)\ , \tag{2}$$

where $s \Leftarrow t$ is the sequence of terminal symbols corresponding to the syntax tree t. This means that the probability of sequences and syntax trees under the intersection grammar H can be efficiently computed up to the normalisation constant Z.

Parsing Existing Sequences

Parsing a given sequence s refers to computing the set of all matching syntax trees $\mathcal{T}$ along with their probabilities under the intersection grammar H. The probability $p_{\mathrm{H}}(t)$ of a tree $t \in \mathcal{T}$ under H factorises into the context-free component $p_{\mathrm{CFG}}(t)$ and the sequence component $p_{\mathrm{RG}}(s \Leftarrow t)$, where $p_{\mathrm{RG}}(s \Leftarrow t)$ is invariant for all trees $t \in \mathcal{T}$. Parsing can thus be accomplished based on the context-free component G_{CFG} only, using standard parsing algorithms. The unknown normalisation constant Z cancels out in this case.

Generating Sequences and Trees

Generating sequences from either component G_{CFG} or G_{RG} can be accomplished by standard means. However, generating sequences from the intersection grammar H is challenging due to the unknown normalisation constant Z. There are two solutions to this problem. One is to use a rejection sampling approach where proposal sequences are generated from one component and accepted with a probability corresponding to the other component. A second approach is to generate the sequence successively (symbol by symbol), which only requires renormalising the distribution for the current symbol in each step. With minimal modifications, both approaches can also be used to generate syntax trees.

A Unified Framework

Let Λ be a latent event space, Σ be an observable event space, L be a latent structure space such that a sequence of latent events $\lambda_1, \lambda_2, \ldots \in \Lambda^*$ creates a latent structure $l \in L$, and O be an observable structure space such that a sequence $\sigma_1, \sigma_2, \ldots \in \Sigma^*$ of observable events creates an observable structure $o \in O$. (For a *context-free grammar* (CFG), for instance, non-terminal symbols are in Λ, terminal symbols are in Σ, syntax trees are in L, and generated strings are in O.) A transition function

$$p(\lambda, \sigma \,|\, l, o) : L \times O \rightsquigarrow \Lambda \times \Sigma \tag{3}$$

then defines a joint stochastic process in latent and observable space.

A large class of models, including CFGs and *regular grammars* (RGs), can be described as feature-based models in the exponential family

$$p(\lambda, \sigma \,|\, l, o) = \frac{1}{Z(l, o)} \exp \sum_{f \in \mathcal{F}} \theta_f f(\lambda, \sigma, l, o) \tag{4}$$

$$Z(l, o) = \sum_{\lambda', \sigma'} \exp \sum_{f \in \mathcal{F}} \theta_f f(\lambda', \sigma', l, o)\ , \tag{5}$$

where $\mathcal{F}$ is a set of features and each feature f has a weight θ_f. In this framework, a CFG can be rewritten as

$$p(\lambda, \sigma \,|\, l, o) \propto \exp \sum_{r \in R} \log(w_r) [\![r(\lambda, \sigma, l)]\!]\ , \tag{6}$$

where w_r is the weight for rule r and $[\![r(\lambda, \sigma, l)]\!] \in \{0, 1\}$ indicates whether the rule matches the respective transition (that is, the rule's left-hand side has to match the latent structure l and the rule's right-hand side has to match the non-terminal and/or terminal events λ and σ). Likewise, a RG can be written as

$$p(\lambda, \sigma \,|\, l, o) \propto \exp \sum_{r \in R} \log(w_r) [\![r(\sigma, o)]\!]\ , \tag{7}$$

with the only difference being that the rules depend only on the observable structure o and the observable event σ. In previous work, we have developed a flexible feature-discovery method based on this kind of model (Lieck 2018) and demonstrated that the learned models achieve state-of-the-art performance for sequential prediction of monophonic music (Langhabel et al. 2017).

References

Langhabel, Jonas, Robert Lieck, Marc Toussaint, and Martin Rohrmeier (2017). "Feature Discovery for Sequential Prediction of Monophonic Music". In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017*. International Society for Music Information Retrieval Conference. Suzhou, China.

Lieck, Robert (2018). "Learning Structured Models for Active Planning: Beyond the Markov Paradigm Towards Adaptable Abstractions". PhD thesis. Stuttgart: Universität Stuttgart.

European Research Council
Established by the European Commission

Part of the ERC project no 760081: Principles of Musical Structure Building: Theory, Computation, and Cognition