SIMULTANEOUS UNSUPERVISED LEARNING OF FLAMENCO METRICAL STRUCTURE, HYPERMETRICAL STRUCTURE, AND MULTIPART STRUCTURAL RELATIONS

Dekai Wu
HKUST, Human Language Technology Center, Department of CSE, Hong Kong
dekai@cs.ust.hk

ABSTRACT

We show how a new unsupervised approach to learning musical relationships can exploit Bayesian MAP induction of stochastic transduction grammars to overcome the challenges of learning complex relationships between multiple rhythmic parts that previously lay outside the scope of general computational approaches to music structure learning. A good illustrative genre is flamenco, which employs not only regular but also irregular hypermetrical structures that rapidly switch between 3/4 and 6/8 *mediocompas* blocks. Moreover, typical flamenco idioms employ heavy syncopation and sudden, misleading off-beat accents and patterns, while often elliding the downbeat accents that humans as well as existing meter-finding algorithms rely on, thus creating a high degree of listener "surprise" that makes not only the structural relations, but even the metrical structure itself, ellusive to learn. Flamenco musicians rely on both complex regular hypermetrical knowledge as well as irregular real-time clues to recognize when to switch meters and patterns. Our new approach envisions this as an integrated problem of learning a bilingual transduction, i.e., a structural relation between two languages—where there are different musical languages of, say, flamenco percussion versus *zapateado* footwork or *palmas* hand clapping. We apply minimum description length criteria to induce transduction grammars that simultaneously learn (1) the multiple metrical structures, (2) the hypermetrical structure that stochastically governs meter switching, and (3) the probabilistic transduction relationship between patterns of different rhythmic languages that enables musicians to predict when to switch meters and how to select patterns depending on what fellow musicians are generating.

1. INTRODUCTION

Little work has been done on automatic algorithms for learning across multiple parts in rhythmically complex music genres such as flamenco, despite a respectable history of work on automatic meter finding utilizing a wide range of underlying modeling paradigms. Repetition of patterns is a central feature in many automatic meter finding algorithms such as that of Steedman [23]. Early approaches were rule-based (e.g., Longuet-Higgins and Steedman [16]), while others employed neural nets (e.g., Desain and Honing [6]) or preference rules (e.g., Povel and Essens [19], Temperley and Sleator [28]). More recent approaches are based on probabilistic modeling, such as the generative models of Cemgil *et al.* [4]), Raphael [20], and Temperley [27].

Generally, however, these approaches are based on relatively simplistic assumptions about straightforward duple, triple, or 4/4 meters accompanied by the regular occurrence of strong accents on or near the downbeat. These assumptions are widely understood to apply primarily to Western music conventions rather than worldwide music genres that can be rhythmically much more complex. Flamenco conventions, for example, often defy for example what Lerdahl and Jackendoff [14] termed the "Strong Beat Early" rule—omitting the downbeat accent is a typical idiom, and in fact the strong beat is often understood in flamenco to be late. Moreover, flamenco rhythms employ continual meter switching in both regular and irregular ways, with a complex hypermetrical language governing the switching, and make frequent use of polyrhythm in addition.

Even less work has been done to date on computational approaches to analysis of flamenco. Models such as those of Diaz-Banez *et al.* [7], Gomez and Bonada [12], Guastavino *et al.* [11], Mora *et al.* [17], or Thul and Toussaint [29] represent various intriguing attacks on specific aspects of flamenco, but do not attempt to actually induce the musical structures.

To attack more complex rhythmic forms such as these, we propose an approach based on unsupervised induction of stochastic transduction grammars. On one hand, this follows the generative modeling paradigm of GTTM [14] and Steedman [24] or [25] in which various aspects of music can each be modeled as languages that can be generated by formal grammars. On the other hand, to facilitate automatic learning and scaling up of the models, we formulate the task in terms of stochastic grammars that describe probabilistic models of musical structure.

The majority of previous work on stochastic grammatical models for music employs flat Markov models and/or hidden Markov models (HMMs). For example, both the Continuator model of Pachet [18] and the Factor Oracle models of Assayag *et al.* [2] use Markov models to learn

music improvisation conventions-an approach further explored by François *et al.* [8] and [9]. A grammar induction approach for learning jazz grammars under Markovian assumptions is proposed by Gillick *et al.* [10]. Relatively little has been done on musical structure modeling using stochastic context-free grammars [13]. Related work on unsupervised learning of CCMs (a variant of SCFGs) for musical grammars includes that of Swanson *et al.* [26], or the Data Oriented Parsing approach of Bod [3].

Stochastic grammars are excellent for describing individual aspects of music. Much of music, however, is about the loosely coupled relationships between *multiple* strands of different kinds of sequences taking place in parallel.

Our new approach differs from previous stochastic grammatical models of music in that we (1) shift to *bilingual* stochastic transduction grammars instead of conventional monolingual stochastic grammars, allowing us to model learning structural *relations* between different musical languages of separate percussive flamenco parts, and (2) apply a new grammar induction strategy that searches for the Bayesian MAP (maximum *a posteriori*) model encompassing metrical relations, hypermetrical relations, and probabilistic transduction relations in a single integrated process.

2. STOCHASTIC TRANSDUCTION GRAMMARS

In classic formal language theory, a **transduction** is a relation between two languages, which is exactly what we wish to induce. A **transduction grammar** or **translation grammar** (TG) is a bilingual grammar of transductions, and describes structured relations between two languages [1], [15]. (An equivalent term "synchronous grammar" used only in computational linguistics is not as long established or widely understood throughout computer science.)

Thus **stochastic transduction grammars** are probabilistic bilingual grammars of transductions, and describe structured relations between two languages *probabilistically*—which means that stochastic TGs do not suffer from the overly rigid constraints of non-stochastic transduction models, and can be automatically learned [31]. In a stochastic transduction grammar, a probability distribution is imposed over the space of possible derivations. This is typically done by associating a conditional probability with each rule, representing the probability that any nonterminal symbol matching the left-hand-side of the rule generates children matching the right-hand-side of the rule. Techniques have been developed for numerous tasks utilizing stochastic transduction grammars including aligning bilingual corpora, unsupervised segmentation and annotation of bilingual corpora, automatic induction of bilingual correspondences, grammar induction for stochastic TGs, and so on [32].

In this paper we will make use of stochastic **monotonic transduction grammars**, which in terms of generative capacity sit in the hierarchy of transduction grammars between stochastic finite-state transducers and linear inversion transduction grammars, as detailed in [32].

A monotonic transduction grammar or MTG (equivalent to the "simple syntax-directed transduction grammar" or "simple SDTG" of Aho and Ullman) in normal form is a tuple $\langle N, \Sigma, \Delta, S, R \rangle$ where N is a finite nonempty set of nonterminal symbols, Σ is a finite nonempty set of input languge symbols, Δ is a finite nonempty set of output language symbols, $S \in N$ is the designated start symbol, and R is a finite nonempty set of syntax-directed transduction rules on the forms:

$$S \to A, \quad A \to \varphi, \quad A \to e/f$$

where $A \in N$, $\varphi \in NNN^*$, and e and f are **terminal** symbols representing musical event segments as follows.

Strings in both the languages represent sequences of symbolic musical event tokens. A "sentence" is a full musical passage for a single instrumental part, whereas a "bisentence" is a matched musical passage with both instrumental parts. For convenience of musical interpretation, instead of writing musical sequences using linguistic string notation, we shall use conventional music staff notation for musical event segments e and f, as in Figure 1. Just as we consider the monolingual terminal symbols e and f to represent musical event segments, we consider the bilingual e/f notation to denote a **biterminal** symbol representing a *parallel pair* of musical event segments from different musical instruments. Technically, $e/f \in (\Sigma^* \times \Delta^*) - (\epsilon/\epsilon)$, in which we exclude the degenerate case of pairing a zero-length empty segment ϵ with another zero-length empty string ϵ to avoid unnecessary complications arising from infinite recursion.

3. TRANSDUCTION GRAMMAR INDUCTION

In this section we describe our new model for unsupervised induction of stochastic transduction grammars. For concrete examples of the abstract model, see Section 4.

Minimizing description length We begin with the overall Bayesian model whose posterior we wish to maximize. We seek the maximum *a posteriori* (MAP) model given the data; that is, we attempt to optimize the posterior probability following Bayes' rule

$$P(\Phi \mid D) = \frac{P(\Phi)\, P(D \mid \Phi)}{P(D)}$$

where Φ is the model and D is the training data. The prior probability of the data is constant during search, which gives us the following search problem:

$$\underset{\Phi}{\operatorname{argmax}} P(\Phi)\, P(D \mid \Phi)$$

In our case, the probability of the data given the model can be determined through parsing since it is a grammar. The prior of the model is, however, somewhat more complicated because it must incorporate the effect of both the structure and the parameters of the model.

$$P(\Phi) = P(\Phi_G)\, P(\Phi_S \mid \Phi_G) P(\theta_\Phi \mid \Phi_S, \Phi_G)$$

Φ_G is a global prior over possible model formalisms, which we set to be the space of possible monotonic transduction grammars, $P(\Phi_S \mid \Phi_G)$ is a prior on the model

structure given the model formalism, and $P(\theta_\Phi \mid \Phi_S, \Phi_G)$ is a prior over the model parameters given the formalism and the structure. We approximate the prior over the model structure using the description length of the model:

$$-\log_2\left(P(\Phi_S \mid \Phi_G)\right) \propto \mathrm{DL}\left(\Phi_S\right)$$

The description length of a model is calculated by summing the length of all the rules, where each unique (monolingual) terminal segment is efficiently given a unique Huffman encoding. This avoids redundant double-counting of terminal segments that appear in more than one rule. The length of a symbol is proportional to $-\log_2\left(\frac{1}{M}\right)$ where $M = 2 + N + \Sigma + \Delta$ is the total number of symbols (N is the number of nonterminals, Σ is the size of the L_0 vocabulary and Δ is the size of the L_1 vocabulary). Thus, for example, reducing the number of distinct nonterminals in a grammar reduces its description length.

We set the prior over the model parameters to be a uniform Dirichlet distribution over right-hand sides given left-hand sides:

$$P(\theta_\Phi \mid \Phi_S, \Phi_G) = \prod_{i=0}^{N-1} \frac{1}{B(\alpha_0, \alpha_1, ..., \alpha_{R_{n_i}-1})} \prod_{j=0}^{R_{n_i}-1} \theta_\Phi^{n_i}(j)$$

where N is the number of nonterminals, R_{n_i} is the set of rules where n_i is the left-hand side, and $\theta_\Phi^{n_i}$ is a function that gives the rule probabilities for rule where the left-hand side is n_i. Fleshing out the search problem, we have:

$$\underset{\Phi_G, \Phi_S, \theta_\Phi}{\operatorname{argmax}} \; P\left(\Phi_G\right) P(\Phi_S \mid \Phi_G) P(\theta_\Phi \mid \Phi_S, \Phi_G) P(D \mid \Phi_G, \Phi_S, \theta_\Phi)$$

Recall that are restricting Φ_G to monotonic transduction grammars. We further divide the search into two phases: a top-down rule segmentation phase, which focuses on the structural induction to optimize $P(\Phi_S \mid \Phi_G)$ and $P(D \mid \Phi_G, \Phi_S, \theta_\Phi)$, and a parameter tuning phase, which focuses on $P(\theta_\Phi \mid \Phi_S, \Phi_G)$ and $P(D \mid \Phi_G, \Phi_S, \theta_\Phi)$.

Initializing model structure The induction procedure starts with a transduction grammar that memorizes the training data as well as possible, and generalizes from there. The transduction grammar that best fits the training data is the one where the start symbol rewrites to the full sentence pairs that it has to generate. It is also possible to add any number of nonterminal symbols in the layer between the start symbol and the bisentences without altering the probability of the training data. We take initial advantage of this by allowing for one intermediate symbol so that the start symbol conforms to the normal form and always rewrites to precisely one nonterminal symbol.

Our initial model thus consists of the rule $S \rightarrow A$ plus numerous rules of the form $A \rightarrow e_{0..T}/f_{0..V}$ where S is the start symbol, A is the nonterminal, T is the length of the output sentence, and V is the length of the input sentence.

Generalizing model structure In order to generalize the initial monotonic transduction grammar we need to identify parts of the existing biterminals that could be validly used in isolation, and allow them to combine with other segments. This is the very feature that allows a finite transduction grammar to generate an infinite set of sentence pairs; when we do this, we move some of the probability mass which was concentrated in the training data out to other data that are still unseen—i.e. we generalize from the training data. The general strategy is to propose a number of sets of biterminal rules and a place to segment them, estimate the posterior given the sets and commit to the best set. That is: we do a greedy search over the power set of possible segmentations of the rule set. This intractable problem can be reasonably efficiently approximated.

The key component in the approach is the ability to evaluate the change in *a posteriori* probability if a specific segmentation was made in the grammar. This can then be extended to a set of segmentations, which only leaves the problem of generating suitable sets of segmentations. Any segment that can be reused maximizes the model prior. The more rules we can find with shared biaffixes, the more likely we are to find a good set of segmentations.

Our algorithm takes advantage of the above observation by focusing on both the monolingual and bilingual affixes (i.e., prefixes or suffixes) found in the training data. Each **affix** or **biaffix** defines a set of lexical rules paired up with a possible segmentation. We evaluate the (bi)affixes by estimating the change in posterior probability associated with committing to all the segmentations defined by a (bi)affix. This allows us to find the best set of segmentations, and commit to as many of them as possible. Moreover, as we generate new nonterminal categories during this process, we also use affixes and biaffixes to suggest possible merges of the nonterminal categories. This minimizes the parsing efforts, which are more expensive. A priority queue based agenda keeps track of possible candidates for rule segmentation and nonterminal category actions, and always greedily commits at each step to the action that best improves overall posterior probability:

```
G = the transduction grammar
biaffixes_to_rules = index of G's transduction rules by their (bi)affixes
lhs_to_rules = index of G's transduction rules by their LHS nonterms
agenda = []        // Priority queue of actions by their DL impact on G
for each affix or biaffix x in G :
   delta = eval_seg_post(x, biaffixes_to_rules[x], G)
   if (delta < 0)
      agenda.add(SEGMENT, x, delta)
while agenda.pop(act, x) < 0 :
   if (act == SEGMENT)
      real_delta = eval_seg_post(x, biaffixes_to_rules[x], G)
      if (real_delta < 0)
        G, modified_rules = segment_rules(x, biaffixes_to_rules[x], G)
         for each pair y of nonterms serving as LHS of modified_rules
               that share a common (monolingual or bilingual) RHS :
            delta = eval_merge_post(x, biaffixes_to_rules[x], G)
            agenda.add(MERGE, y, delta)
   else if (act == MERGE)
      real_delta = eval_merge_post(y, lhs_to_rules[y], G)
      if (real_delta < 0)
         G = merge_nonterms(y, lhs_to_rules[y], G)
```

Note that *both* affixes and biaffixes are handled in `biaffixes_to_rules` (since an affix can be regarded as a special case of a biaffix where one of the two affixes is the empty string ϵ). We have written `eval_seg_post` and `segment_rules` as shorthand for the above-discussed evaluation of the impact of a rule segmentation action upon the

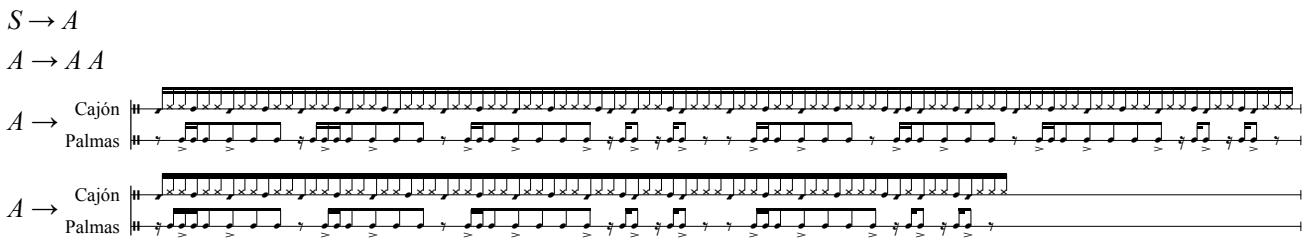


Figure 1. Initial transduction grammar with two training examples (see text). Lexical transduction rules are shown with their biterminals in *cajón* and *palmas* staves using standard music notation for sequences instead of character strings.

posterior. This consists of removing the rule being segmented, inserting the three new rules (one structural and two terminal) along with two new nonterminals, and uniformly distributing the segmented rule's probability mass over the three new rules; exact mathematical details are in [21]. Similarly, the shorthand `eval_merge_post` and `merge_nonterms` denote consolidating the probability mass of two nonterminal categories.

The change in the model prior is easy to estimate, as it is proportional to the change in grammar length when the old rule is removed and the new rules are inserted (keeping in mind that a rule can only be added once, so if it already exists inserting it will not change the description length).

The change in the probability of the data given the model is expensive to get through biparsing the data, so instead we accumulate enough statistics during biparsing to be able to make an educated guess. We follow the analogous procedure after merging two nonterminals.

Optimizing model parameters Although the iterative segmentation of the rules result in reasonable parameters, there is still room for improvement. In this phase we consider the model structure to be fixed, and optimize the model parameters to give the highest possible posterior probability, i.e., we fix Φ_S (to be what we arrived at using the algorithm described in the previous section), as well as Φ_G (which remains fixed as MTGs). The two remaining free factors in the MAP are thus: $P(\theta_\Phi \mid \Phi_S, \Phi_G)$ and $P(D \mid \Phi_G, \Phi_S, \theta_\Phi)$—the prior over the parameters and the conditional probability of the data given the complete model.

The prior is, as described earlier, a uniform Dirichlet distribution over all the rules, which can be described using a concentration parameter. To get the conditional, we have to biparse the training data, and to maximize it, we perform expectation maximization [5], as a special case of EM as specified for inversion transduction grammars by [30]. This requires biparsing, which we do with the cubic time biparsing algorithm described in [22].

4. RESULTS

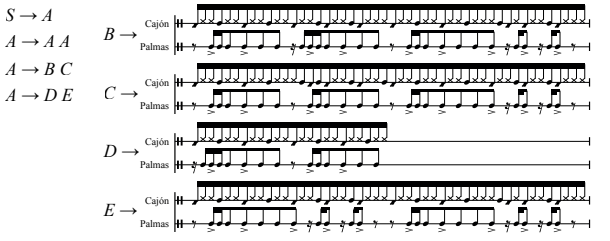
For our experiments we chose the *buleriás* form of flamenco because of its metrical and hypermetrical complexity. Various passages from multi-track recordings were collected, from which were taken approximately 5 minutes of aligned *cajón* (box drum) and *palmas* (clapping percussion) tracks. Symbolic "tick" notes were extracted from the *cajón* and *palmas* tracks via heuristics primarily relying on a combination of volume and frequency range filters. For the *cajón* the notes were separated into "bass", "tone", and "tip" categories. For the *palmas* the notes were separated into ordinary or accented notes. All notes were quantized into 1/16th note intervals.

As no meaningful gold standard for mechanically evaluating the quality of the learned model exists, only subjective evaluations are possible. Moreover, variance in flamenco expectations is extremely large, and our subjective evaluations were so close to 100% accuracy so as to be swamped by statistical variance in human judgments. Reasons for the high accuracy of the model can best be seen by tracing specifically how it learns, looking at a small concrete subset of the training set.

Two short training passages are shown in Figure 1. Note that in our induction method's rule segmenting strategy, the initial transduction grammar starts out containing one rule for each training example, each belonging to the same generic category A as shown by the left-hand-side nonterminal. In addition, the grammar contains the start rule and a low-probability "glue rule" that allows arbitrary concatenation of any valid sequences when all else fails.

Learning metrical structure In the early iterations, the MDL-driven induction algorithm primarily works toward learning transduction patterns that are a single full *compas* in length, i.e., a twelve beat cycle. This accurately mirrors the primary (mixed meter) structure that is most common and most fundamental in flamenco.

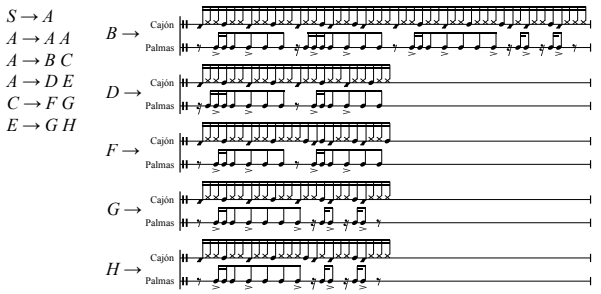
The two lexical transduction rules in the initial transduction grammar of Figure 1 share no biaffixes, but the first half of the first lexical rule's *cajón* part is repeated at the end of the second lexical rule's *cajón* part. Thus, the first induction decision is to segment both lexical transduction rules so as to gain the bits from efficiently encoding their common *cajón* sequence, instead of redundantly enumerating the same string twice. This yields a revised grammar



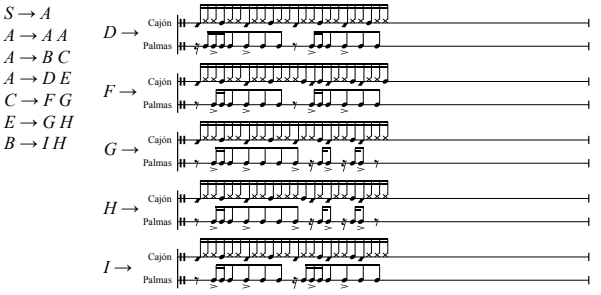
with shorter description length, that introduces new nonter-

minals so as to generate the same transduction as before.

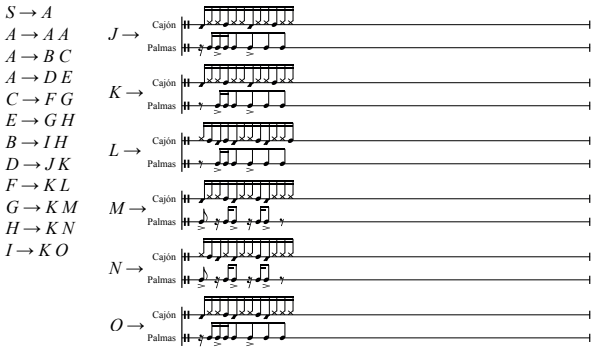
In the new grammar, a shared biaffix is found, between the second half of the C rule and the first half of the E rule. Segmenting both rules, again introducing new nonterminals as needed, yields:



This has created another shared biaffix—the new H rule is a bisuffix of the B rule. Segmenting the B rule yields:



Note that the algorithm has learned the full *compas* length patterns. Typically, in subsequent iterations, the induction process moves on to gradually learn *mediocompas* patterns that are only six beats in length. Here, the next segmentation, based on the shared bisuffix of the D and F rules and biprefix of the G, H, and I rules, has this effect:

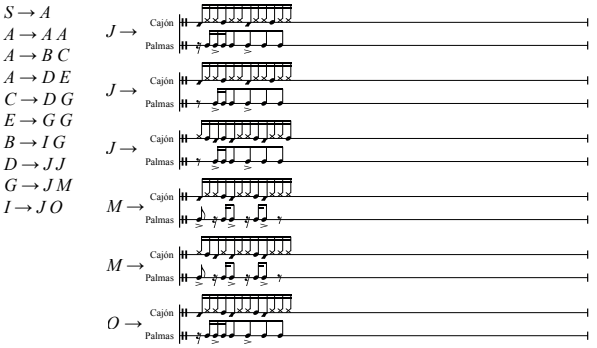


At this point, there are no more rules that can be segmented to lower the description length. However, we notice that it is still possible to improve the posterior probability of the model, because some of the newly created nonterminal categories might be merged in such a way as to improve the model structure prior $P(\Phi_S \mid \Phi_G)$by reducing the model's description length in terms of the number of nonterminal categories, without introducing so much error that it excessively decreases the likelihood of the data $P(D \mid \Phi_G, \Phi_S, \theta_\Phi)$.

The nonterminals that have just been created generate four candidates for merging, since the algorithm considers each pair of nonterms serving as the LHS of modified rules that share a common RHS: $J \approx K$, $J \approx O$, $K \approx L$, and $M \approx N$. Of these, $K \approx L$ gives by far the best improvement in posterior probability, so we replace all instances of L in the grammar with K instead.

The next best merge is $J \approx K$, so we replace in turn all instances of K (including those formerly L) with J. In so doing, the updated rules for D and F become identical, both now with $J\ J$ on the right-hand-side. Thus we propagate the merging to replace all instances of F with D, at no additional cost. Similarly, merging $M \approx N$ slightly improves the posterior, and in so doing, the updated rules for G and H become identical so we also merge H into G.

Merging $J \approx K$, $K \approx L$, and $M \approx N$ inherently introduces generalizations that are able to (bi)parse new examples outside the training set. The iteration terminates without merging $J \approx O$ which would introduce too much error to improve the posterior. The final transduction grammar induced is thus:



Accurately reflecting flamenco norms, induction has categorized the mediocompas patterns into three distinct nonterminal types: J comprises patterns in 6/8 meter, while M comprises patterns in 3/4 meter, and O is a polyrhythmic pattern that crosses both. The fact that a distinction between the two meters can be learned—even though our current approach does not incorporate any explicit a priori model of accented pulses at constant repeated intervals—arises from the transduction grammar induction's natural integration of metrical structure learning together with hypermetrical structure learning, as discussed below.

Between the 6/8 and 3/4 transduction patterns, certain common *palmas* sequences appear in both. This correctly reflects conventional flamenco usage of palmas (playing a similar function to clave patterns in Afro-Latin genres). However, the 3/4 transduction patterns also tend more frequently to relate certain *palmas* sequences that do not generally appear with 6/8 patterns. Such patterns tend to have notes that align more naturally with 6/8 accents. *Palmas* sequences are useful to learning because of their clave-like function, even though they are not as consistent as Afro-Latin clave, and are usually silent on what would be the strong downbeat in most mainstream dance music forms.

Learning hypermetrical structure Among the many flamenco forms, the *buleriás* style is particularly aggressive about using *mediocompas* patterns in irregular ways. The learned rule $G \rightarrow J\ M$ models a regular full alternating meter *compas*, while $I \rightarrow J\ O$ models the same polyrhythmically against a 6/8 *mediocompas* feel. The rule $D \rightarrow J\ J$ models a full compas in 6/8 *mediocompas* feel. The

rules for B, C, and E model longer *compas* pairs, with typical idioms such as staying in 6/8 meter until the final fourth *mediocompas*. This behavior is naturally emergent from the MDL-driven induction (even more so on larger training sets).

Learning probabilistic transduction relationships The induced transduction grammar is potentially useful in a wide variety of applications, which we plan to investigate in detail in our next steps. Currently, the learned model can already be used to predict suitable accompaniment for either instrumental part. Given a previously unseen *cajón* sequence, a Viterbi parse translates the sequence into the most probable *palmas* sequence with nearly perfect accuracy. Similarly, given a new *palmas* sequence, the model can translate it into the most probable *cajón* sequence, which is currently generally acceptable though not necessarily musically optimal. Our future work will focus on further refining the predictive accuracy for accompaniment from a stylistic standpoint.

The new MDL-driven transduction grammar induction method we have introduced is the first to (1) exploit opportunities to compress both monolingual affixes and bilingual affixes, (2) exploit regularities in either single language to help segment rules describing both languages, and (3) exploit both monolingual and bilingual regularities to induce categories for longer hypermetrical patterns. We anticipate numerous further applications beyond the flamenco genre.

5. ACKNOWLEDGMENTS

This material is based upon work supported in part by the Hong Kong Research Grants Council (RGC) research grants GRF620811, FSGRF13EG28, GRF621008, and GRF612806; the Defense Advanced Research Projects Agency (DARPA) under BOLT contract no. HR0011-12-C-0016, and GALE contract nos. HR0011-06-C-0022 and HR0011-06-C-0023; and by the European Union under the FP7 grant agreement no. 287658. Thanks to Markus Saers, Karteek Addanki, Chi-kiu Lo, and Meriem Beloucif for assistance with implementation. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the RGC, EU, or DARPA.

6. REFERENCES

[1] Alfred V. Aho and Jeffrey D. Ullman. *The Theory of Parsing, Translation, and Compiling (Volumes 1 and 2)*. Prentice-Halll, Englewood Cliffs, NJ, 1972.

[2] Gérard Assayag, Georges Bloch, Marc Chemillier, Arshia Cont, and Shlomo Dubnov. OMax Brothers: A dynamic topology of agents for improvization learning. In *First ACM Workshop on Audio and Music Computing Multimedia*, pages 125–132, 2006.

[3] Rens Bod. Stochastic models of melodic analysis: Challenging the gestalt principles. *Journal of New Music Research*, 30(3), 2001.

[4] Ali Taylan Cemgil, Bert Kappen, Peter Desain, and Henkjan Honing. On tempo tracking: Tempogram representation and Kalman filtering. *Journal of New Music Research*, 29(4):259–273, 2000.

[5] Arthur Pentland Dempster, Nan M. Laird, and Donald Bruce Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[6] Peter Desain and Henkjan Honing. *Music, Mind, and Machines: Studies in Computer Music, Music Cognition, and Artificial Intelligence*. Thesis Publications, Amsterdam, 1992.

[7] J. Miguel Díaz-Báñez, Giovanna Farigu, Francisco Gómez, David Rappaport, and Godfried T. Toussaint. El compás flamenco: A phylogenic analysis. In *BRIDGES: Mathematical Connections in Art, Music and Science*, pages 61–70, Southwestern College, Winfield, Kansas, Jul 2004.

[8] Alexandre R.J. François, Elaine Chew, and Dennis Thurmond. Mimi - a musical improvisation system that provides visual feedback to the performer. Technical Report 07-889, USC Computer Science Department, Apr 2007.

[9] Alexandre R.J. François, Isaac Schankler, and Elaine Chew. Mimi4x: An interactive audio-visual installation for high-level structural improvisation. In *IEEE International Conference on Multimedia and Expo (ICME 2010)*, pages 1618–1623, 2010.

[10] Jon Gillick, Kevin Tang, and Robert M. Keller. Machine learning of jazz grammars. *Computer Music Journal*, 34(3):56–66, Fall 2010.

[11] Catherine Guastavino, Francisco Gómez, Godfried Toussaint, Fabrice Marandola, and Emilia Gómez. Measuring similarity between flamenco rhythmic patterns. *Journal of New Music Research*, 38(2):129–138, 2009.

[12] Emilia Gómez and Jordi Bonada. Automatic melodic transcription of flamenco singing. In *Fourth Conference on Interdisciplinary Musicology (CIM08)*, Thessaloniki, Greece, Jul 2008.

[13] Karim Lari and Steve J. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56, 1990.

[14] Fred Lerdahl and Ray Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983.

[15] Philip M. Lewis and Richard E. Stearns. Syntax-directed transduction. *Journal of the Association for Computing Machinery*, 15(3):465–488, 1968.

[16] Hugh Christopher Longuet-Higgins and Mark J. Steedman. On interpreting Bach. *Machine Intelligence*, 6:221–241, 1971.

[17] Joaquín Mora, Francisco Gómez, Emilia Gómez, Francisco Escobar-Borrego, and José Miguel Díaz-Báñez. Characterization and melodic similarity of a cappella flamenco cantes. In *11th International Society for Music Information Retrieval Conference (ISMIR)*, pages 351–356, 2010.

[18] François Pachet. The continuator: Musical interaction with style. *Journal of New Music Research*, 32(3):33–341, 2003.

[19] Dirk-Jan Povel and Peter Essens. Perception of temporal patterns. *Music*, 2(4):411–440, Summer 1985.

[20] Christopher Raphael. A hybrid graphical model for rhythmic parsing. *Artificial Intelligence*, 137(1-2):217–238, May 2002.

[21] Markus Saers, Karteek Addanki, and Dekai Wu. Iterative rule segmentation under minimum description length for unsupervised transduction grammar induction. In Adrian-Horia Dediu, Carlos Martín-Vide, Ruslan Mitkov, and Bianca Truthe, editors, *First International Conference on Statistical Language and Speech Processing (SLSP 2013)*, volume 7978 of *LNAI*, pages 224–235, Tarragona, Spain, Jul 2013. Springer.

[22] Markus Saers, Joakim Nivre, and Dekai Wu. Learning stochastic bracketing inversion transduction grammars with a cubic time biparsing algorithms. In *11th International Conference on Parsing Technologies (IWPT'09)*, pages 29–32, Paris, Oct 2009.

[23] Mark J. Steedman. The perception of musical rhythm and metre. *Perception*, 6(5):555–569, 1977.

[24] Mark J. Steedman. The formal description of musical perception. *Music Perception*, 2:52–77, 1984.

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

[26] Reid Swanson, Elaine Chew, and Andrew S. Gordon. Supporting musical creativity with unsupervised syntactic parsing. In *AAAI Spring Symposium on Creative Intelligent Systems*, 2007.

[27] David Temperley. *Music and Probability*. MIT Press, 2007.

[28] David Temperley and Daniel Sleator. Modeling meter and harmony: A preference-rule approach. *Computer Music Journal*, 23(1):10–27, 1999.

[29] Eric Thul and Godfried T. Toussaint. On the relation between rhythm complexity measures and human rhythmic performance. In *Conference on Computer Science & Software Engineering (C3S2E '08)*, 2008.

[30] Dekai Wu. Trainable coarse bilingual grammars for parallel text bracketing. In *Third Annual Workshop on Very Large Corpora (WVLC-3)*, pages 69–81, Cambridge, MA, Jun 1995.

[31] Dekai Wu. Stochastic Inversion Transduction Grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404, Sep 1997.

[32] Dekai Wu. Alignment. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing*, pages 367–408. Chapman and Hall / CRC, second edition, 2010.