Binarity, branchingness, and size effects

Jennifer Bellik (jbellik@ucsc.edu) & Nick Kalivoda (nkalivod@ucsc.edu)

UC SANTA CRUZ

Introduction

- In prosody, longer strings tend to be parsed into more constituents.
    - Ex. two feet in (**a**bra)ka(**da**bra) but one foot in sha(**zam**)
- Such *size effects* are commonly captured using Binarity constraints (e.g., Inkelas & Zec 1990, Ito & Mester 1992, Prince & Smolensky 1993/2004; Sandalo & Truckenbrodt 2002, Prieto 2007, Selkirk 2011, Elfner 2012).
- Implementations of Binarity come in two major flavors, which often aren't distinguished:
    - **Branch-counting Binarity** (formalized in [1]) requires a node to branch into two children (of any category), and echoes syntactic notions of binarity.
    - **Leaf-counting Binarity** (formalized in [2]) reflects a rhythmic conception of binarity, derived from foot-building. It requires a node to contain two dominated nodes of some particular category (e.g. σ), and sometimes lower categories.

(1) Binarity-Branches(K) = Bin-Br(K): Assign a violation for every node of category K with more than two branches (immediate children, of any category).

(2) Binarity-Leaves(K,L) = Bin-Lv(K, L): Assign a violation for every node of category K that dominates more than two nodes of category L at any level, where L< K.

Big question: Are both versions of Binarity necessary and desirable for understanding the syntax-prosody interface?

Proposal:

- Binarity in prosody is best conceived of as counting branches, regardless of category.
- Category-sensitive leaf-counting binarity should be restricted to counting rhythmic categories.

Binarity under Weak vs. Strict Layering

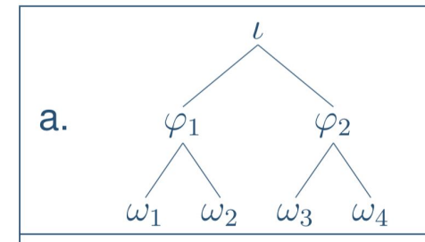
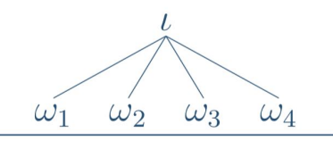
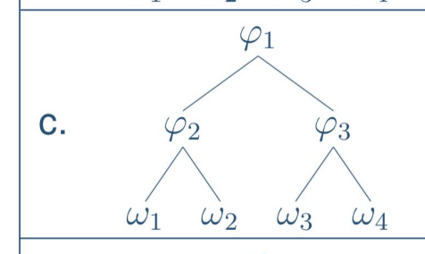
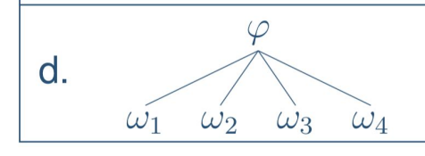
Under Strict Layering: No difference between branch- and leaf-counting

- In structures that conform to *Strict Layering* (Selkirk 1984), no level-skipping (non-exhaustive parsing) or level-doubling (recursion) is permitted.
- Therefore, every branch in a Strictly Layered tree corresponds to a child of the next lower prosodic category, and counting branches is equivalent to counting leaves of the next lower prosodic category.
    - Ex. Strictly Layered (3a,d) incur the same violations under branch- and leaf-counting binarity.
- So in effect, branch-counting binarity is already employed in analyses with Strict Layering, such as Shih (2017), Prieto (2007), and Sandalo & Truckenbrodt (2002), even when the analysis defines binarity in leaf-counting terms.

Under Weak Layering: Branch-counting and leaf-counting pull apart

- *Weak Layering* (Ito & Mester 1992/2003) permits recursion (3c) and level skipping (3b)
- As a result, not every branch corresponds to a child of the next lower prosodic category.
    - Recursion produces children of the same category as the parent
    - Non-exhaustive parsing produces children of an even lower prosodic category
    - Ex.: in (3c), φ_1 has only two branches (to φ_2 and φ_3) and satisfies Bin-Br, but φ_1 violates Bin-Lv since it dominates four leaves (ω_1, ω_2, ω_3, ω_4).

(3) *Constraint comparison*

$[_{XP}\ X_1\ [_{XP}\ X_2\ [_{XP}\ X_3\ X_4\ ]]]]$	Bin(φ,Br)	Bin(φ,ω)	*Layering*
a. 			strict
b. 			weak (*Exh)
c. 		φ_1	weak (*NonRec)
d. 	φ	φ	strict

Selected References. Bellik, J. & N. Kalivoda. 2016. Adjunction and Branchingness Effects in Syntax-Prosody Mapping. In Hansson, G.O., A Farris-Trimble, K. McMullin, & D. Pulleyblank (eds.), *Supplemental Proceedings of the 2015 Annual Meeting on Phonology.* **Elfner, E. 2012.** Syntax-Prosody Interactions in Irish. University of Massachusetts - Amherst dissertation. **Ito, J. & A. Mester. 1992/2003.** Weak Layering and Word Binarity. Linguistic Research Center, LRC-92-09, University of California, Santa Cruz. **— 2013.** Prosodic subcategories in Japanese. Lingua 124, pp. 20-40. **— 2015.** The perfect prosodic word in Danish. Nordic Journal of Linguistics, Vol. 38, No. 1, pp. 5-36. **Kalivoda, N. 2018.** Syntax-Prosody Mismatches in Optimality Theory. Ph.D. Thesis, UC Santa Cruz. **Prince, A. & P. Smolensky.** 1993/2004. Optimality Theory: Constraint Interaction in Generative Grammar. **Selkirk, E. 1984.** *Phonology and Syntax: The Relation between Sound and Structure*. MIT Press. **— 2011.** The syntax-phonology interface. In J. Goldsmith, J. Riggle & A. Yu (eds.) The Handbook of Phonological Theory, 2nd edition.

Branch-counting motivates size-driven recursion

Branch-counting binarity (1) assigns a violation to nodes that branch into more than two immediate children. Therefore, it prefers candidates with more prosodic structure.

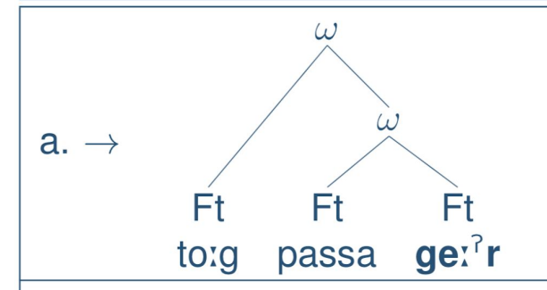
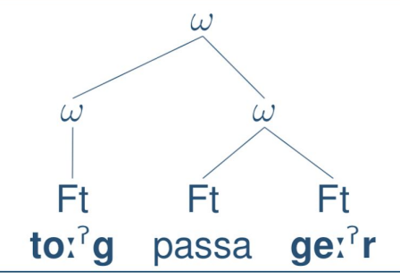
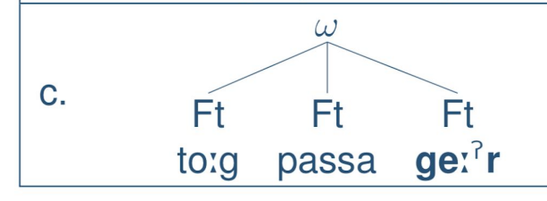
Case study 1: Size-effects in Danish compound words

- Danish glottal accent ('stød') diagnoses the right edge of a prosodic word in Danish
- Stød reveals length-driven differences in compound phrasing (Ito & Mester 2015):

(4) a. [$_\omega$ to:g [$_\omega$ passa**ge:ʔr**]] 'train passenger' b. [$_\omega$ [$_\omega$ passa**ge:ʔr**] [$_\omega$ **to:ʔg**]] 'pass. train'

- These size effects can be derived with the ranking Bin-Br >> NonRec >> Match (5).
    - **NonRecursivity** penalizes recursive ωs and outranks Match, ruling out the isomorphic (4b) and favoring flat structures like (4c).
    - **Bin-Br** compels the building of a recursive ω (4a) when a flat structure would create a ω with more than two branches (4c).
    - Leaf-counting **Bin(ω,Ft)** would not have the same effect: the maximal ω still dominates three feet even when recursive sub-structure intervenes.

(5) *Tableau for Danish 'train passenger'*

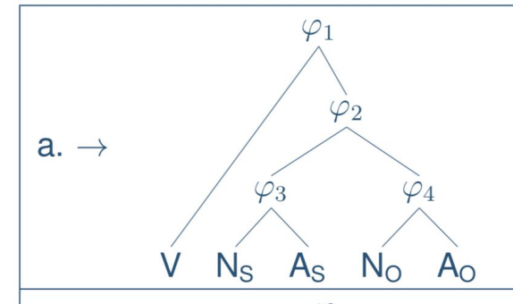
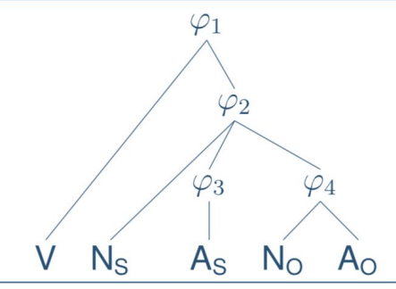
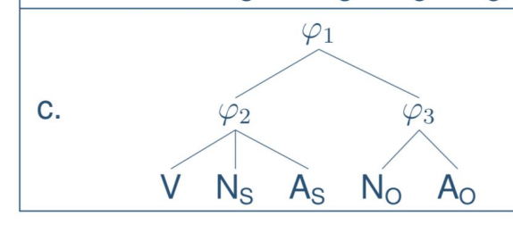
$[_{X^0}\ [_{X^0}$ to:g$]\ [_{X^0}$ passage:r$]]$	Bin-Br	NonRec	Match	Bin(ω,Ft)
a. → 		*	*	*
b. 		**!		*
c. 	*!		**	*

Case study 2: Size-effects in Irish phrasing

In Irish, LH shows the left boundary of φ_{NonMin}, and HL shows the right boundary of φ (from Elfner 2012). This phrasing is governed by the constraint ranking in (6) (Elfner 2012).

- $[_{\Sigma P}V[_{TP}[_{NP}N]][_{VP}[_{NP}NA]]]]\rightarrow(_{\varphi}(_{\varphi}VN)(_{\varphi}NA))$
- Elfner attributes this to **StrongStart** (Selkirk 2011), which penalizes ($_\varphi$ ω φ ...) structures, and outranks Match.
- But StrongStart violations are tolerated in order to avoid **Bin-Br** violations (6).
- To avoid a binarity violation in winner (6a), binarity must be assessed by counting branches for φ_1.
- (6a)'s φ_1, which dominates five ωs, still violates leaf-counting **Bin-ω**, but is binary branching.

(6) *Tableau for Irish* (Elfner 2012)

$[_{\Sigma P}\ V\ [_{TP}\ [_{DP}\ N_S\ A_S]\ [_{VP}\ [_{DP}\ N_O\ A_O]]]]$	Bin-Br	StSt	Match	Bin-ω
a. → 		φ_1		φ_1, φ_2
b. 		$\varphi_1!, \varphi_2$	TP, DP$_S$	φ_1, φ_2
c. 	$\varphi_2!$		TP, DP$_S$	φ_1, φ_2

Similar examples are found in Kimatuumbi (Kalivoda 2018) and Mandarin (Shih 2017).

Upshot: Only branch-counting motivates size-effects and a closer syntax-prosody match.

Leaf-counting motivates size-driven category change

Leaf-counting binarity counts dominated nodes of a lower prosodic category (cf. Dresher & van der Hulst 1998), and can motivate a size-driven change of category.

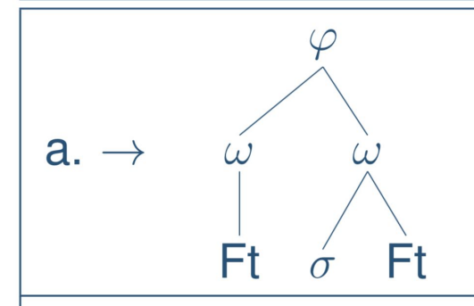
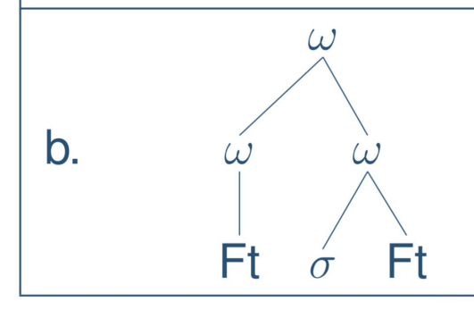
- In (3a,b), the violation of leaf-counting Bin-Lv(φ,ω) is avoided by punting the binarity violation up to the level of ι, so that (3a,b) outperform their counterparts (3c,d).
- Unlike Bin-Lv, Bin-Br does not favor the level-skipping (3b) over (3a,c)--it doesn't motivate a category change for the root prosodic node.
- The case study shown below, where Ft- and σ-counting Bin-Lv promotes ω to φ, is the only instance of such a category change known to us.

Case study: Japanese compound phrasing

Category-promotion can be seen in Japanese compounds (Ito & Mester 2013).

- Compounds consisting of no more than two feet are parsed into a prosodic structure rooted in ω (as diagnosed by ω-internal rendaku voicing, compound accent, etc.)
- But in compounds of the form [[Ft][σ Ft]], the category of the root node changes to φ instead (7a), so that no ω dominates more than two feet or syllables.
- This category-change is driven by leaf-counting binarity (Bin-Lv(ω, [Ft, σ])): each ω can dominate only up to two nodes of lower prosodic categories.

(7) *Tableau for Japanese compound*

[[Ft] [σ Ft]]	Bin (ω, [Ft, σ])	Bin (ω, Br)
a. → 		
b. 	*!	

- Branch-counting binarity can't drive this category-change—both (7a,b) satisfy Bin-Br.

If category-promotion when only occurs when rhythmic categories, not interface categories, are counted, then Con should only include versions of Bin-Lv that count feet and syllables, not ω or φ (i.e., Bin-Lv(ω, [Ft, σ]), but not Bin-Lv(φ,ω) or Bin-Lv(ι,φ))

Leaf-counting's undesirable typological predictions

Typologies are larger under leaf-counting binarity. In a case study of Kinyambo phrasing, leaf-counting systems predicted typologies that were, on average, 80% larger than branch-counting typologies, with an average of 10.67 more languages (Bellik & Kalivoda 2016).

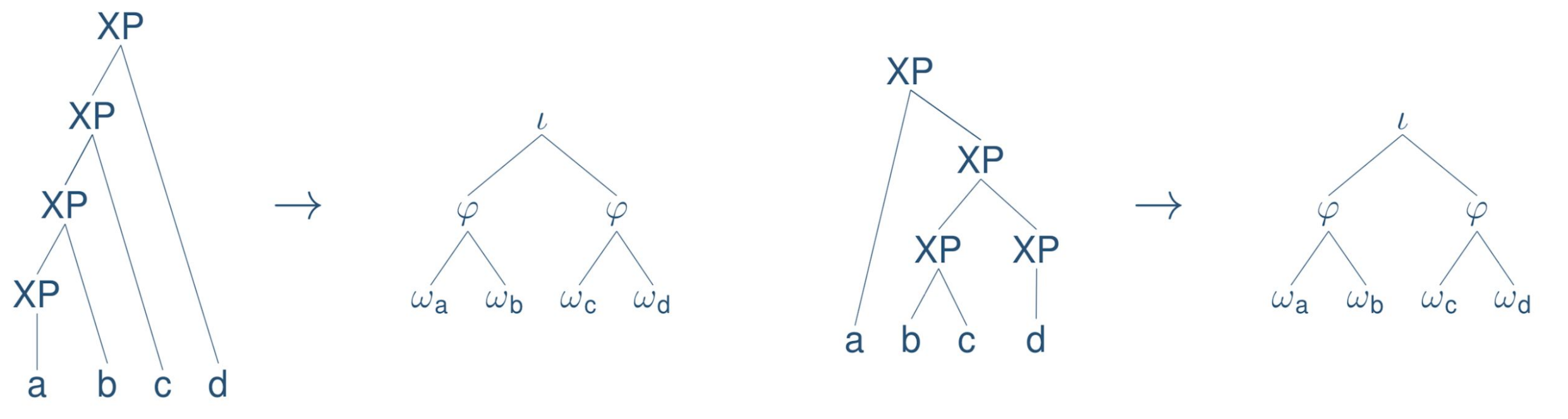
(8) *Number of languages in Bellik & Kalivoda (2016) typologies (4 inputs from Kinyambo)*

	Pair 1 (Match, low)	Pair 2 (Align, low)	Pair 3 (Match, high)	Pair 4 (Align, high)	Pair 5 (Match, high & low)	Pair 6 (Match, high & low)
Bin-Lv lgs	19	40	17	30	15	28
Bin-Br lgs	17	15	18	12	11	12
|Br|-|Lv| ((|Br|-|Lv|)/|Br|)	2 (12%)	25 (167%)	-1 (-5%)	18 (150%)	4 (36%)	16 (133%)

Leaf-counting binarity predicts a language that completely disregards syntax.

- If Bin-Lv were top-ranked, syntaxes like [a [b [c [d]]], [[ab][cd]] and [[[[a] b] c] d] would all be optimally parsed as ($_\iota$ ($_\varphi$ a b) ($_\varphi$ c d)). Note the mapping of the top XP to ι, not φ. To our knowledge, this is unattested.

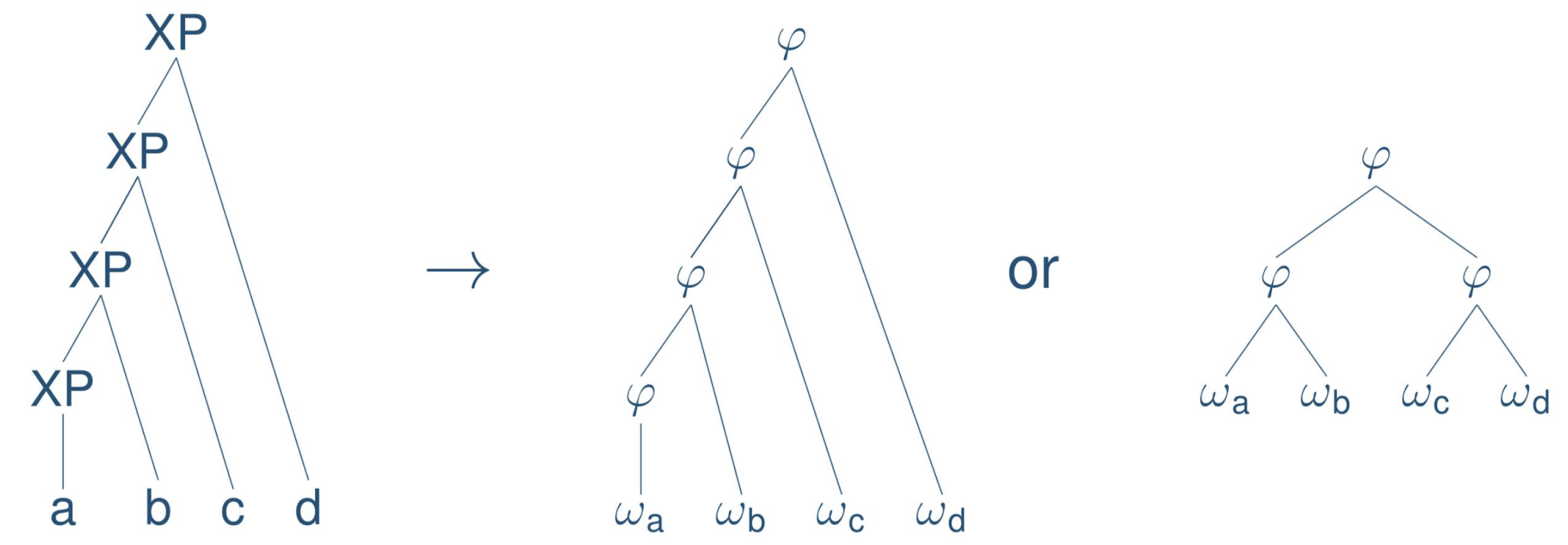
(9)



In Japanese, [[[[a] b] c] d] → ($_\varphi$ ($_\varphi$ a b) ($_\varphi$ c d)) (Kubozono 1989). However, this is not the same as ($_\iota$ ($_\varphi$ ab) ($_\varphi$ cd)) above, since there is a φ containing a, b, c, d. This 4-leaved φ violates Bin-Lv, so this is not an argument in favor of Bin-Lv *per se* (Kalivoda 2018).

- In contrast, Bin-Br never compels syntax-prosody mismatches since the syntactic input is already binary-branching.

(10)



Further disadvantages of leaf-counting binarity

Leaf-counting binarity is usually redundant, even under Weak Layering

- Leaf-counting is only essential if it rules out an intended loser that does *not* violate branch-counting. Call such phrasings that violate Bin-Lv but not Bin-Br **leaf-violators**.
- Most leaf-violators can be ruled out by other prosodic well-formedness constraints. For all phrasings of three words (generated using SPOT [Bellik, Bellik & Kalivoda 2018]), *all* leaf-violators incurred additional penalties from StrongStart, EqualSisters, or NonRecursivity, compared to all non-leaf-violating possible optima.

Leaf-counting is more computationally complex. Branch-counting only examines a node's immediate children—a local search. But leaf-counting requires a global search through theoretically unbounded levels of recursion for all nodes of some lower category.

Conclusion

Branch-counting and leaf-counting versions of binarity differ significantly in their predictions:

- **Size effects**: Branch-counting motivates size-driven recursion or splitting, deriving the desired size effects. Leaf-counting cannot predict such size effects.
- **Category change**: Leaf-counting does motivate size-driven category change, which is only attested for ω being promoted to φ.
- **Typology**: Branch-counting predicts smaller typologies than leaf-counting, and leaf-counting predicts a language we believe to be unattested.

Furthermore, leaf-counting is almost always redundant. Con and the size of predicted typologies could be constrained by restricting Bin-Lv to counting rhythmic categories (σ, Ft), while allowing branch-counting binarity to apply at all levels of the prosodic hierarchy.