9 Match Theory and prosodic well-formedness constraints

Junko Ito and Armin Mester

9.1 Introduction[1]

Several strands of work in prosodic theory have recently converged around a number of common themes, from different directions. Selkirk (2009) (see also Elfner 2012) has developed a vastly simplified approach to the syntax-prosody mapping that distinguishes only three levels (word, phrase, and clause), and syntactic constituents are systematically made to correspond to phonological domains (Match Theory). In an independent line of research, a long string of papers reaching back into the 1980s has convincingly demonstrated that recursive structures are by no means an exclusive property of syntax, but also play a crucial role in phonology. Even though at variance with strict layering (Selkirk 1984; Nespor and Vogel 1986), the empirical existence of recursive prosody is undeniable, as first demonstrated by Ladd (1986, 1988), whose findings have been corroborated by Kubozono (1989, 1993,) Schreuder and Gilbers (2004), Gussenhoven (2005), Wagner (2005, 2010), Schreuder (2006), Kabak and Revithiadou (2009), Ito and Mester (2009b), Féry (2010), and van der Hulst (2010), to name a few, undermining a central tenet of orthodox prosodic hierarchy theory that supposedly sets phonology apart from syntax. Building on these empirical findings, Ito and Mester (2007, 2009a, 2013) have gone on to argue that, beyond its sheer existence, prosodic recursion allows for a vast, and much-needed, simplification in the inventory of prosodic categories themselves. The empirically necessary subcategories that the data of individual languages often seem to demand (such as the minor versus the major phonological phrase of Japanese, long established under these names since McCawley (1968), and rechristened into "accentual" versus "intermediate phrase" by Pierrehumbert and Beckman (1988)) are not separate categories, each existing on its own in some (or all?) language(s), but are rather instances of a single recursively deployed basic category. These results are very much in harmony with central ideas in Match Theory, and recent work (Selkirk 2011; Ishihara 2014) has successfully connected the two theories into a larger framework.

One of the hallmarks of Match Theory is the idea that the main force interfering with syntax-prosody isomorphism is not some kind of non-isomorphic

mapping algorithm flattening out the structure, as first contemplated in the *Sound Pattern of English* (Chomsky and Halle 1968: 372) and more fully worked out in later proposals, such as the edge-based theory built on one-sided alignment (Selkirk 1986). It is rather the effect of genuine phonological well formedness constraints on prosodic structure: Whenever such constraints dominate a MATCH constraint, prosodic structure is forced to diverge from its syntactic model, often in significant ways. A large part of the explanation of syntax-prosody mismatches, then, lies in the precise content of these well-formedness constraints that conflict with the entailments of isomorphism.

We will here survey four types of constraints that interact with MATCH constraints in this way: NOLAPSE (against tonal lapses, see Ito and Mester 2013), EQUALSISTERS (against sister nodes in prosodic trees of different levels in the prosodic hierarchy, see Myrberg 2013), and constraints enforcing binary branching in prosodic trees, both minimally (BINMIN) and maximally (BINMAX), which have been widely discussed in the literature.

The goal of this chapter is to cast a critical eye on the way these kinds of prosodic constraints interact with MATCH constraints in recent work in prosodic theory, our own included, and to isolate a number of issues that are in need of deeper investigation. We restrict ourselves to work within optimality-theoretic phonology (OT; see Prince and Smolensky 2004), and will make concrete proposals and conjectures at various junctures, identifying problem areas along the way.

The primary data we will use as a testing ground are taken from an area we are familiar with, the prosodic form of Tokyo Japanese utterances (henceforth "Japanese"), and leave for future exploration the more detailed investigation of other languages that might provide equally valuable or even better cues. Our overall goal, however, is not to provide a comprehensive analysis of Japanese but rather to subject a number of theories and ideas to an empirical test. That being said, at the end we hope to end up with some understanding of the kinds of constraints that are necessary to explain a prosodic system.

9.2 Tonal antilapse constraints

The first type of constraint we will take up militates against tonal lapses: stretches of low-toned material exceeding a certain limit, typically at the ends of words and phrases. In this section, we will first review relevant data from Japanese and their analysis, and then turn to a comparison with Basque.

9.2.1 *NoLapse in Japanese*

Ito and Mester (2013) develop an analysis of the way Japanese utterances are parsed into phonological phrases where NOLAPSE plays a central role in forcing the accentual fall to occur late in the word. A virtue of this approach is that the orientation of the accent toward the end of the word is explained by substantive tonal factors, not by stipulated formal right alignment.[2] The facts

here have been well known since Kubozono (1989, 1993), and we illustrate them with phrases consisting of two content words (after Vance 2008: 181). The parses assigned to these examples by the theory proposed in Ito and Mester 2013 appear in the second column in (1), where (1cd) crucially involve recursive phrasing, as first recognized by Kubozono.

(1)	S:	P:	Example:	Gloss:
a.	[$_{XP}$ [$_{XP}$ u] u]	($_\varphi$ u u)	[[Hiroshima-no] sakana-to]	'Hiroshima fish and ...'
b.	[$_{XP}$ [$_{XP}$ u] a]	($_\varphi$ u a)	[[Hiroshima-no] tamágo-to]	'Hiroshima eggs and ...'
c.	[$_{XP}$ [$_{XP}$ a] a]	($_\varphi$ ($_\varphi$ a) ($_\varphi$ a))	[[Okáyama-no] tamágo-to]	'Okayama eggs and ...'
d.	[$_{XP}$ [$_{XP}$ a] u]	($_\varphi$ ($_\varphi$ a) ($_\varphi$ u))	[[Okáyama-no] sakana-to]	'Okayama fish and ...'

Here and in what follows, φ stands for "phonological phrase", ω for a "phonological word", a for "accented ω", u for "unaccented ω", and syntactic and prosodic phrasing are labeled "S" and "P" and indicated by [...] and (...), respectively. The differences between these parses – flat phrasing in (1ab), recursive phrasing in (1cd)), but never exactly mirroring the syntax – are entirely due to the locations of accented and unaccented words. The beginning of a phonological phrase is in Japanese cued by a tonal rise. It is always possible, in careful pronunciation, to parse each word as a separate φ, with its own initial rise, but this is not the usual pattern. While two a's are each parsed as a separate phrase (because each accent has to be the head of a minimal phrase), u is typically phrased together with an adjacent a or u (because one-word phrases violate binarity). This is where Kubozono (1993: 150–154) discovered a directional asymmetry: While two u's are phrased together, u is only phrased together with a following a, not with a preceding a. So the results are (1a) (*uu*) and (1c) (*(a) (a)*), but (1d) (*(a)(u)*) with an initial rise at the beginning of the second word and (1b) (*ua*) without such a rise. The data are subject to considerable variation. We focus here exclusively on the majority patterns.

The analysis in Ito and Mester (2013) involves the four constraints in (2). Following Elfner (2012: 28), we sharpen the definition of matching in the following way: In order for a phonological constituent *p* to *match* a syntactic constituent *s*, *p* must exhaustively dominate all and only the phonological exponents of the terminal nodes that *s* exhaustively dominates. As a result, syntactic constituents introducing no overt terminal elements are invisible to the matching process, and do not need to be separately matched.

(2)   a.   AccentAsHead: Every accent is the head of a minimal phrase φ_{min}[3]
         Assign one violation for each accent that is not the head of a φ_{min}.
      b.   NoLapse-L: No tonal lapses. Assign one violation for each fully L-toned ω in φ.

c. MINIMALBINARITY-φ/ω: φ is minimally binary. Assign one violation for each φ that does not dominate at least two ω.

d. MATCH-XP-TO-φ: A phrase XP in syntactic constituent structure is matched by a corresponding phonological phrase φ in phonological representation.

Ranked as in tableau (3),[4] these constraints derive the different parses in (1). The syntactic pattern $[[u]x]$, where $x = a$ or u, is parsed as the non-isomorphic single φ (3ae) (ux), violating bottom-ranked MATCH-XP but satisfying higher-ranked BINMIN in the optimal way. However, $[[a]x]$ is parsed as (3im) $((a)(x))$, violating BINMIN: Isomorphic (3l) $((a)a)$ is out because the second a violates ACCASHEAD.

(3)	*Japanese with* NOLAPSE		NOLAPSE-L	ACCASHD	BINMIN	MATCH-XP
[[u]u]	a. ▶	(uu)				*
	b.	(u(u))			*W	*
	c.	((u)u)			*W	L
	d.	((u)(u))			**W	L
[[u]a]	e. ▶	(ua)				*
	f.	(u(a))			*W	*
	g.	((u)a)		*W	*W	L
	h.	((u)(a))			**W	L
[[a]a]	i. ▶	((a)(a))			**	
	j.	(aa)		*W	L	*W
	k.	(a(a))		*W	*L	*W
	l.	((a)a)		*W	*L	
[[a]u]	m. ▶	((a)(u))			**	
	n.	(au)	*W		L	*W
	o.	(a(u))		*W	*L	*W
	p.	((a)u)	*W		*L	

The most interesting case is $[[a]u]$ parsed as (3m) $((a)(u))$, with a rise on u, as depicted in (4d). The main tonal events in these examples are indicated with schematic pitch contours.

(4)
  a. =(3a) (*u* *u*) ⎫

  b. =(3e) (*u* *a*)   NoLapse-L fulfilled

  c. =(3i) ((*a*)(*a*)) in the winning candidates,

  d. =(3m) ((*a*)(*u*)) ⎭ where no ω is fully L-toned.

  e. =(3n) (*a* *u*) ⎫ NoLapse-L violated: the

  f. =(3p) ((*a*) *u*) ⎭ final *u* is fully L-toned after
                      the accentual fall.

A fully L-toned ω arises after an accentual fall unless it is in its own φ (thereby receiving the tonal rise on its own). The tonally low final *u* in the losing candidates (3np) violates NoLapse-L. By contrast, the leading *u* in (3ae) is tonally high, and does not violate NoLapse-L. In this analysis, the directional asymmetry thus has an explanation rooted in the very shape of the tonal melody of (Tokyo) Japanese ($_\%$LH- and%LH-H*L).

9.2.2 Comparison with Basque

Selkirk and Elordieta (2010) propose a slightly different analysis of Japanese along similar lines, the chief difference being that the phrasing (*au*) is ruled out not because the post-accentual *u* is a tonal lapse, but because it separates the accentual head *a* from the end of its phrase. Instead of NoLapse-L, their analysis appeals to the alignment constraint ALIGN-RIGHT(φ_{min}-HEAD, φ_{min}), which requires the accented word *a* to be at the right edge of its φ_{min}. Among the relevant candidates in (6), both ((*a*)*u*) (6p) and ((*a*)(*u*)) (6m), but not (*au*), fulfill ALIGN-R. In order to select (6m) over (6p), they introduce a further constraint in (5) into the analysis.

(5) EQUALSISTERS: Sister nodes in prosodic structure are instantiations of the same prosodic category.

Ranked above BINMIN, EQUALSISTERS[5] (Myrberg 2013: 75) selects the correct candidate ((*a*)(*u*)) in (6m).[6]

(6)	*Japanese with* ALIGN-R *and* EQUALSIS		ACCASHD	ALIGN-R	EQUALSIS	BINMIN	MATCH-XP
[[u]u]	a. ▶	(uu)					*
	b.	(u(u))			*W	*W	*
	c.	((u)u)			*W	*W	L
	d.	((u)(u))				**W	L

(6)		*Japanese with* ALIGN-R *and* EQUALSIS	ACCASHD	ALIGN-R	EQUALSIS	BINMIN	MATCH-XP
[[u]a]	e. ▶	(ua)					*
	f.	(u(a))			*W	*W	
	g.	((u)a)	*W	*W	*W	*W	L
	h.	((u)(a))				**W	L
[[a]a]	i. ▶	((a)(a))				**	
	j.	(aa)	*W	*W		*L	*W
	k.	(a(a))	*W	*W	*W	L	
	l.	((a)a)	*W	*W	*W	*L	
[[a]u]	m. ▶	((a)(u))				**	
	n.	(au)		*W		L	*W
	o.	(a(u))		*W	*W	*L	*W
	p.	((a)u)			*W	*L	

Selkirk and Elordieta (2010) insightfully contrast the Japanese phrasings in (1) with the minimally different phrasings of corresponding examples in Northern Bizkaian Basque (henceforth "Basque"), which has a pitch accent system tantalizingly close to that of Japanese, including a very similar word melody (for an overview, see Gussenhoven 2004: 170–184). The chief difference is that the *u* of [[*a*]*u*], in Japanese parsed as its own phrase ((*a*)(*u*)), is in Basque not parsed as a separate phrase. Selkirk and Elordieta (2010) express this by assigning the parse ((*a*)*u*) (where nothing in the data seems to signal the presence of the phrase (*a*), however).

(7) *Japanese:*   [[a]u]   →   ((a)(u))   [EQUALSISTERS >> BINMIN]
     Basque: [[a]u]   →   ((a)u)   [BINMIN >> EQUALSISTERS]

They derive the different outcomes in Japanese and Basque by the ranking scenario in (7), with the result shown in (8).

(8)		*Basque with* ALIGN-R *and* EQUALSIS	ACCASHD	ALIGN-R	BINMIN	EQUALSIS	MATCH-XP
[[u]u]	a. ▶	(uu)					*
	b.	(u(u))			*W	*L	*
	c.	((u)u)			*W	*L	W
	d.	((u)(u))			**W		W

(8)	*Basque with* ALIGN-R *and* EQUALSIS		AccAsHd	Align-R	BinMin	EqualSis	Match-XP
[[u]a]	e. ▶	(ua)					*
	f.	(u(a))			*W	*W	*
	g.	((u)a)	*W	*W	*W	*W	L
	h.	((u)(a))			**W		L
[[a]a]	i. ▶	((a)(a))			**		
	j.	(aa)	*W	*W	L		*W
	k.	(a(a))	*W	*W	*L	*W	*W
	l.	((a)a)	*W	*W	*L	*W	
[[a]u]	m. ▶	((a)u)			*	*	
	n.	(au)		*W	L	L	*W
	o.	(a(u))		*W	*	*	*W
	p.	((a)(u))			**W	L	

In our own analysis with NoLAPSE instead of ALIGN-R and EQUALSISTERS, the Basque system emerges when NoLAPSE ranks below BinMin, as shown in (9).[7]

(9)	*Basque with* NoLAPSE		AccAsHd	BinMin	NoLapse-L	Match-XP
[[u]u]	a. ▶	(uu)				*
	b.	(u(u))		*W		*
	c.	((u)u)		*W		L
	d.	((u)(u))		**W		L
[[u]a]	e. ▶	(ua)				*
	f.	(u(a))		*W		*
	g.	((u)a)	*W	*W		L
	h.	((u)(a))		**W		L
[[a]a]	i. ▶	((a)(a))		**		
	j.	(aa)	*W	L		*L
	k.	((a)a)	*W	*L		
	l.	(a(a))	*W	*L		*W
[[a]u]	m. ▶	(au)			*	*
	n.	(a(u))		*W	L	*
	o.	((a)u)		*W	*	L
	p.	((a)(u))		**W	L	L

In Section 9.3, we look more closely at the workings of EQUALSISTERS in larger structures, and encounter some advantages for an analysis building on a tonal NOLAPSE constraint.

9.3 Constraints on sister nodes

EQUALSISTERS (Myrberg 2013) is a new member of the set of separate and violable constraints that takes the place of traditional strict layering (Ito and Mester 1992; Selkirk 1996): Daughter nodes are not required to be exactly one level below their mother nodes in the hierarchy, but they need to be of the same level.

An interesting problem arises when we consider three-member syntactic constituents of the form $[[[x]y]z]$. In order to keep things simple, we restrict our attention to left-branching structures, which conform to the basic pattern of Japanese phrase structure (see Ito and Mester (2013) for right-branching structures). Complex noun phrases as in (10) are examples.

(10) a. [[[u]u]u] [$_{NP}$ [$_{NP}$ [$_{NP}$ amerika-no] tomodachi-no]   pasokon]
                  America -$_{GEN}$ friend -$_{GEN}$ PC
                  '(my) American friend's PC'
     b. [[[a]a]a] [$_{NP}$ [$_{NP}$ [$_{NP}$ isuráeru-no] kurasuméeto-no] rapputóppu]
                  Israel -$_{GEN}$ classmate-$_{GEN}$ laptop
                  '(my) Israeli classmate's laptop'

As we systematically build examples with all combinations of accented and unaccented words in the three N-positions, we end up with the picture in (11).

(11)   S: [[[1]2]3]   P: Schematically:

     a. [[[u]u]u]   ((uu)   u) } ((1 2) 3)
     b. [[[u]a]u]   ((ua)   (u)) ⎤
     c. [[[u]a]a]   ((ua)   (a)) ⎬ ((1 2) (3))
     d. [[[u]u]a]   ((uu)   (a)) ⎦
     e. [[[a]a]a]   (((a)(a)) (a)) ⎤
     f. [[[a]a]u]   (((a)(a)) (u)) ⎬ (((1)(2)) (3))
     g. [[[a]u]a]   (((a)(u)) (a)) ⎬
     h. [[[a]u]u]   (((a)(u)) (u)) ⎦

The eight syntactic structures are mapped to three different prosodic parses, depending on the accentedness of each constituent word: The relatively flat ((1 2) 3) is assigned only to [[[u]u]u] (11a); ((1 2) (3)), with a phrased

3, is assigned when 1 is *u* (except for [[[*u*]*u*]*u*]) (11b-d); (((1)(2)) (3)), where each word is its own phrase, is assigned when 1 is *a* (11e-h).

For an analysis based on the interplay of ALIGN-R, BINMIN, and EQUALSISTERS, these outcomes present a conundrum, brought out clearly by the comparative tableau (Prince 2000) in (12): In order for ((*uu*)*u*), without any unary phrase, to win over the competing sister-wise equal ((*uu*)(*u*)), we need BINMIN>>EQUALSISTERS (12a). But in order for the sister-wise equal ((*ua*)(*u*)) to win over the competing ((*ua*)*u*), which has one less unary phrase, we need EQUALSISTERS>>BINMIN (12b).

(12)

input	winner	loser	ACCASHD	ALIGN-R	BINMIN	EQUALSIS	MATCH-XP
a. [[[u]u]u]	((uu) u)	((uu) (u))			W	L	
b. [[[u]a]u]	((ua) (u))	((ua) u)			L	W	

The ranking contradiction disappears once NOLAPSE-L takes the place of ALIGN-R, since it is able to differentiate between the two cases: ((ˌua↓) u), but not ((ˌuu) u), ends on a low *u*, as indicated by the pitch arrows, violating NOLAPSE-L. This is shown in (13).
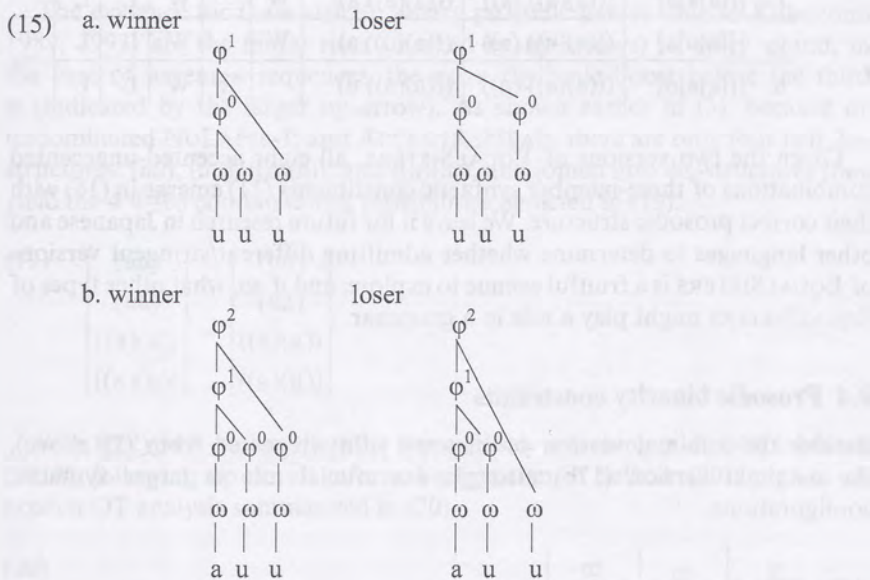
(13)

input	winner	loser	ACCASHD	NOLAPSE-L	BINMIN	EQUALSIS	MATCH-XP
a. [[[u]u]u]	((ˌuu) u)	((ˌuu) (u))				W	L
b. [[[u]a]u]	((ˌua↓)(ˌu))	((ˌua↓) u)		W	L	W	

But all is not well in EQUALSISTERS-land: A different confrontation of winner-loser pairs shows that the ranking contradiction between BINMIN and EQUALSISTERS persists even with NOLAPSE-L. All that is needed is a case where NOLAPSE-L does not differentiate the candidates. Let us takes the inputs [[[*u*]*u*]*u*] and [[[*a*]*u*]*u*], as in (14). In order for ((*uu*)*u*) to win over ((*uu*)(*u*)), we again need BINMIN>>EQUALSISTERS (14a). But in order for (((*a*)(*u*))(*u*)) to win over the competing (((*a*)(*u*))*u*), which has one less unary phrase, we need EQUALSISTERS >> BINMIN (14b). This time, NOLAPSE-L fails to resolve the ranking contradiction between EQUALSISTERS and BINMIN.

(14)

input	winner	loser	AccasHd	NoLapse	BinMin	EqualSis	Match-XP
a. [[[u]u]u]	((ₜu u) u)	((ₜ u u)(ₜu))			W	L	
b. [[[a]u]u]	(((ₜa↓)(ₜu))(ₜu))	(((ₜa↓)(ₜu)) u)			L	W	

As we consider the two pairs, a way out might suggest itself since the two cases differ in terms of the severity of sister inequality. This becomes clearer when we inspect more detailed representations, with explicit indications of projection levels, as in (15).

(15)   a. winner   loser

$$\varphi^1 \qquad\qquad \varphi^1$$

a. winner: φ¹ → φ⁰ , ω ; φ⁰ → ω ω ; ω ω ω → u u u
loser: φ¹ → φ⁰ , φ⁰ ; φ⁰ → ω ω ; ω ω ω → u u u

b. winner   loser

$$\varphi^2 \qquad\qquad \varphi^2$$

b. winner: φ² → φ¹ , φ⁰ ; φ¹ → φ⁰ , ω ; φ⁰ φ⁰ φ⁰ → ω ω ω → a u u
loser: φ² → φ¹ , ω ; φ¹ → φ⁰ , φ⁰ ; ω ω ω → a u u

Whereas the winner in (15a) has a pair of sister nodes (ω, φ°), the loser in (15b) has a pair (ω, φ¹). EQUALSISTERS theorists might seize on this difference and expand the constraint, in the familiar OT manner, into a family of constraints penalizing sister inequality of different degrees of severity. Let us assume, for concreteness, that besides the general EQUALSISTERS constraint (5) penalizing any difference in category between sister nodes, there is a more stringent constraint penalizing a situation where a category inequality is aggravated by a concomitant projection level inequality. We might call the more stringent constraint EQUALSISTERS-2, violated when λ^j is sister to κ^i, with $\lambda > \kappa$ and $j > i$. Ranked above BINMIN, EQUALSISTERS-2 removes the problem, as (16b) shows.

(16)

	input	winner	loser	AccAsHd	NoLapse	EqualSis-2	BinMin	EqualSis	Match-XP
a.	[[[u]u]u]	((uu) u)	((uu) (u))				W	L	
b.	[[[a]u]u]	(((a)(u)) (u))	(((a)(u)) u)			W	L	W	
c.	[[[u]a]u]	((ua) (u))	((ua) u)	W			L	W	
d.	[[[u]a]a]	((ua) (a))	((ua) a)	W			L	W	
e.	[[[u]u]a]	((uu) (a))	((uu) a)	W			L	W	
f.	[[[a]a]a]	(((a)(a)) (a))	(((a)(a)) a)	W		W	L	L	
g.	[[[a]u]a]	(((a)(u)) (a))	(((a)(u)) a)	W		W	L	L	
h.	[[[a]a]u]	(((a)(a)) (u))	(((a)(a)) u)		W	W	L	L	

Given the two versions of EQUALSISTERS, all eight accented-unaccented combinations of three-member syntactic constituents (11) emerge in (16) with their correct prosodic structure. We leave it for future research in Japanese and other languages to determine whether admitting different/stringent versions of EQUALSISTERS is a fruitful avenue to explore, and if so, what other types of EQUALSISTERS might play a role in a grammar.

9.4 Prosodic binarity constraints

Besides the minimal version of binarity (17a) (repeated from (2) above), the maximal version (17b) also plays a crucial role in larger syntactic configurations.

(17) Binarity constraints
MINIMALBINARITY-φ/ω: φ is minimally binary. Assign one violation for each φ that does not dominate at least two ω.
MAXIMALBINARITY-φ/ω: φ is maximally binary. Assign one violation for each φ that dominates more than two ω.

For four-member left-branching structures of the form [[[[1]2]3]4], Kubozono (1989, 1993) (see also Shinya, Selkirk, and Kawahara 2004) has convincingly shown that the prosodic structure is always such that the immediate daughters of the maximal φ contain two ω's, as exemplified with all accented words in (18a) (from Ishihara 2014) and all unaccented words in (18b).

(18)   Left-branching structure [[[[1]2]3]4]

   a.   S: [[[a]a]a]a]   [[[[Máriko-ga] nónda] wáin-no] niói]
                                Mariko-NOM drank wine-GEN smell
                                'the smell of wine that Mariko drank'

         P: (((a)(a))((a)(a)))   (((ˌMáˌriko-ga)(ˌnóˌnda)) ((↑wáˌin-no)(ˌnióˌi)))

   b.   S: [[[[u]u]u]u]   [[[Mamoru-ga] yonda]gakuchoo-no] uwasa]
                                Mamoru-NOM invited president-GEN rumor
                                'the rumor of the college president that Mamoru
                                invited'

         P: (((uu)(uu)))   ((ˌMamoru-ga yonda) (ˌgakuchoo-no uwasa))

The evidence for these strictly binary prosodic parses (due to Kubozono 1989, 1993) are the initial rises (marked by up-arrows) in every φ and, in the case of accented sequences, the extra rhythmic boost before the third ω (indicated by the larger up-arrow). As shown earlier in (3), because of undominated NoLapse-L and AccentAsHead, there are only four licit 2ω-structures: (*uu*), (*ua*), ((*a*)(*a*)), and ((*a*)(*u*)), and joined into 4ω-structures they yield the 4´4=16 combinatorial possibilities depicted in (19).

(19) $\begin{bmatrix} (uu) \\ (ua) \\ ((a)(a)) \\ ((a)(u)) \end{bmatrix} \begin{bmatrix} (uu) \\ (ua) \\ ((a)(a)) \\ ((a)(u)) \end{bmatrix}$

Why ((12)(34)), rather than the more closely matching ((12)3)4)? Ishihara (2014), following up on an informal suggestion in Selkirk (2011), gives an explicit OT analysis summarized in (20).

(20)

S:		P:	BinMax-φ	BinMin-φ	Match-XP
[[[[u] u] u] u]	a. ▶	((u u) (u u))	*		**
	b.	((u u) u) u)	**W		*L
	c.	((u)u) u) u)	**W	*W	L
	d.	(u u u u)	*		***W
[[[[a] a] a] a]	e. ▶	(((a)(a)) ((a)(a)))	*	****	*
	f.	(((a)(a)) (a)(a))	**W	****	L
	g.	((a) (a) (a) (a))	*	****	**W

The strictly binary candidate $((uu)(uu))$ wins because it does perfectly on BINMIN-φ and has only one violation on BINMAX-φ, at the cost of two unmatched XPs. The same holds for winning $(((a)(a))((a)(a)))$, with the difference that each a also constitutes its own φ because its accent needs to be the head of a phrase. The candidates considered here are all parsed into a single φ at the top level, violating BINMAX-φ at least once. This is the more important point made by Ishihara: There is a need for a separate, and higher-ranking, match constraint requiring XP_{max} to correspond to a φ, ruling out the prosodically non-cohering $(uu)(uu)$. We will return to the details in the next section, and here focus on the workings of the binarity constraints.

Low-ranking MATCH-XP decides in favor of $((uu)(uu))$ against the completely flat $(uuuu)$ (16d). The same holds for 3ω-structure (21), where $((uu)u)$ beats (uuu)–even though here, the empirical consequences are the same, with no medial rise predicted (there are no known cues for the end of a φ in Japanese), and Ishihara (2014) considers (uuu) to be the winner.

(21)

S:		P:	BINMAX-φ	BINMIN-φ	MATCH-XP
[[[[u] u] u]	a. ▶	((u u) u)	*		*
	b.	(u u u)	*		**W
	c.	((u u)(u))	*	*W	*

Further inspection reveals, however, that we cannot rely on low-ranking MATCH-XP to determine the winner once more constraints are in place. It turns out that we need an additional binarity constraint not counting ω's, but insisting on maximal binary branching. BINMAXBRANCH-φ (22) rules out, for example, the flat (uuu) and $(uuuu)$. which do not just contain more than 2 ω's but also have ternary and quaternary branching.

(22)   BINMAXIMALBRANCH-φ: φ obeys maximal binary branching: $*(_\varphi \alpha\beta^3...)$. Assign one violation for each φ that branches into more than two daughter nodes.

The following tableau anticipates this point and shows that BINMAX-φ/ω does not distinguish between (23a) and (23d), but BINMAXBRANCH-φ does. On the other hand, BINMAXBRANCH-φ does not distinguish between (23a-c).

(23)

S:		P:	BinMax Branch-φ	BinMax-φ	BinMin-φ	Match-XP
[[[[u] u] u] u]	a. ▶	((u u) (u u))		*		**
	b.	((u u) u) u)		**W		*L
	c.	((u)u) u) u)		**W	*W	L
	d.	(u u u u)	*W	*		***W

We will see below that while BinMax-φ is crucially dominated by other constraints, BinMaxBranch-φ is undominated and never violated in winning candidates. Pending further investigation into other systems, it is perhaps worth keeping in mind whether the undominated nature of this constraint may be an indication that Gen only produces unary and binary prosodic constituents.

Questions about further types of binarity constraints remain to be explored, such as binarity requirements on $\varphi_{[+max]}$, $\varphi_{[-max]}$, or $\varphi_{[-min]}$, and can only be properly addressed with additional empirical evidence. Thus Selkirk and Elordieta (2010), based on earlier work by Elordieta (2007), have proposed a constraint requiring exact binarity (at the φ-level) for IP-initial φ in Basque (BinMinMax-φ/φ, in our terms), a type of StrongStart constraint (Selkirk 2011), and we expect additional findings along these lines.

9.5 Syntax-prosody matching constraints

Just as there are several versions of binarity constraints, previous research has uncovered that syntax-prosody mapping in Japanese calls for higher-ranking specific versions of Match constraints requiring maximal XP's (24b) and non-minimal XP's (24c) to correspond to phonological phrases.

(24)   Match-XP constraints

    a.   Match-XP-to-φ (repeated from (2) above): A phrase XP in syntactic constituent structure is matched by a corresponding phonological phrase φ in phonological representation.

    b.   Match-XP$_{[+max]}$-to-φ: A maximal phrase XP in syntactic constituent structure is matched by a corresponding phonological phrase φ in phonological representation (Ishihara 2014).[8]

c.   MATCH-XP$_{[-min]}$-TO-φ: A non-minimal phrase XP in syntactic constituent structure is matched by a corresponding phonological phrase φ in phonological representation (Ito and Mester 2013).

As Ishihara (2014) correctly points out, without a higher-ranking MATCH-XP$_{[+max]}$, candidates (25be), without a corresponding φ for their maximal XPs, wrongly emerge as winners, because they do not violate BINMAX-φ, and the general MATCH-XP is ranked too low to rule them out.

(25)

S:		P:	MATCH-XP$_{[+max]}$	BINMAX-φ	MATCH-XP$_{[-min]}$	BINMIN-φ	MATCH-XP
[[[[u] u] u] u]	a. ▶	((uu)(uu))		*	*		**
	b.	(uu) (uu)	*W	L	*		***W
	c.	((uu)u)u)		**W	L		*L
[[[[a] a] a] a]	d. ▶	(((a)(a))((a)(a)))		*	*	****	**
	e.	((a)(a)) ((a)(a))	*W	L	*	****	***W
	f.	((((a)(a))(a))(a))		**W	L	****	*L

[MATCH-XP$_{[+max]}$ >> BINMAX-φ] is determined by the WL pairing in candidate rows (25be). On the other hand, the non-minimal version of MATCH-XP$_{[-min]}$ (24c) proposed in Ito and Mester (2013) is violated by the winning candidates (25a and d) (where the non-minimal XP [[[ω]ω]ω]. does not have a corresponding φ), and must be ranked below BINMAX-φ.

Instead of reviewing here the empirical rationale for this XP$_{[-min]}$ constraint given in Ito and Mester (2013) involving 3ω-sequences, we take a step back and explore an alternative way of thinking about this type of mismatch violation. In parallel OT, there is no need for syntax-prosody mapping to only go in one direction, from syntax to prosody, and it is also possible to check whether the surface prosody matches the syntax, as in fact suggested in Selkirk (2011). In ((uu)$_1$ (uu)$_2$), (uu)$_1$ is matched by [[u]u] in the syntax, but (uu)$_2$ has no syntactic correspondent. Rather than proliferating MATCH-XP constraints, we propose here that the constraint doing the crucial work is the prosody-syntax matching constraint MATCH-φ defined in (26).

(26) MATCH-φ: A phonological phrase φ in phonological representation is matched by a corresponding syntactic constituent in syntactic representation.

As formulated, (26) only looks for a corresponding syntactic constituent, not necessarily an XP. MATCH-φ can simply take the place of MATCH-XP$_{[-min]}$ in (25) above without any violation difference. We leave it as a question for future research whether MATCH-XP$_{[-min]}$ is still necessary, and similar questions can be asked about MATCH-φ$_{[+min]}$, MATCH-φ$_{[+max]}$, MATCH-φ$_{[-min]}$, and MATCH-φ$_{[-max]}$.

Why is MATCH-φ (replacing MATCH-XP$_{[-min]}$) a necessary ingredient of the analysis? For all-accented and all-unaccented 3ω-cases, MATCH-φ is not a crucial factor (illustrated here by the all W-markings in the rows of (27ab). On the other hand, for cases of mixed accented/unaccented 3ω-cases, the rebracketed candidates need to be ruled out by MATCH-φ>>BINMIN-φ in (28cd). For full tableaux with relevant candidates and detailed discussion, see Ito and Mester 2013, substituting MATCH-φ for MATCH-XP$_{[-min]}$.

(27)

	3ω-input	winner	rebracketed loser	MATCH-XP$_{[+max]}$	BINMAX-φ	MATCH-φ	BINMIN-φ	MATCH-XP
a.	[[[[u]u]u]	((uu) u)	(u (uu))			W		W
b.	[[[a]a]a]	(((a)(a)) (a))	((a) ((a)(a)))			W		W
c.	[[[a]u]a]	(((a)(u)) (a))	((a) (ua))			W	L	W
d.	[[a]u]u]	(((a)(u)) (u))	((a) (uu))			W	L	W

Moving on to the various 4ω-combinations in (28), rebracketed candidates are actually winners, and are selected by [BINMAX-φ >> MATCH-φ].

(28)

	4ω-input	rebracketed winner	loser	MATCH-XP$_{[+max]}$	BINMAX-φ	MATCH-φ	BINMIN-φ	MATCH-XP
a.	[[[u]u]u]u]	((uu) (uu))	(((uu) u) u)		W	L		L
b.	[[[a]a]a]a]	(((a)(a)) ((a)(a)))	(((a)(a)) (a))(a)))		W	L		L
c.	[[[[a]u]u]u]	(((a)(u)) (uu))	(((a)(u)) u) u)		W	L		L
d.	[[[[a]u]u]u]	(((a)(u)) ((a)(u)))	(((a)(u)) (a))(u))		W	L		L

The overall interplay of constraints from the MATCH and BINARITY families has the interleaved structure in (29).

(29)   MATCH   BIN   MATCH   BIN   MATCH
       MATCH-XP$_{[+max]}$ >>BINMAX-φ   >>MATCH-φ   >>MINBIN-φ   >>MATCH-XP

9.6 Summary and conclusion

Throughout our research reported in this chapter, we have relied on OTW(orkplace)[9] to guide us in precisely formulating the constraints, and verifying proper (non-contradictory) rankings. With all ten constraints discussed in this chapter (ranging from syntax-prosody match constraints, to structural constraints requiring binarity/equal sisters, to accentual/tonal constraints), OTW allows us to check our proposed constraints and rankings, and in the process, to also pinpoint problem areas and contradictory rankings, as well as find unintended consequences, unwanted winners, and so on. In (30) and (31), we present the OTW tableaux for two examples with crucial candidates showing the established rankings.

(30)

S:		P:	MATCH-MAXXP	NOLAPSE-L	ACCASHD	BINMAXBRANCH-φ	BINMAX-φ	MATCH-φ	EQUALSIS-2	BINMIN-φ	MATCH-XP	EQUALSIS
[[[a]u]u]	a. ▶	(((a)(u))(u))					1			3		
	b.	(((a)(u))u)					1		1W	2L		1W
	c.	((a)(uu))					1	1W		1L	1W	
	d.	((a(u))(u))			1W		1			2L	1W	1W
	e.	((au)u)		1W			1			L	1W	1W

In (30), the leftmost sequence of WL in row (b) shows [EQUALSIS>>BINMIN-φ], in row (c) [MATCH-φ >> BINMIN-φ], in row (d) [ACCASHEAD>>BINMIN-φ], and in row (e) [NOLAPSE-L>> BINMIN-φ].

(31)

		MATCH-MAXXP	NOLAPSE-L	ACCASHD	BINMAXBRANCH-φ	BINMAX-φ	MATCH-φ	EQUALSIS2	BINMIN-φ	MATCH-XP	EQUALSIS
S:	P:										
[[[[u]u]u]u]	► ((uu)(uu))					1	1			2	
	(((u)u)(uu))					1	1		1W	1L	1W
	(((uu)u)u)					2W	L	1W		1L	2W
	(uuuu)				1W	1	L			3W	
	(uu)(uu)	1W				L	1			3W	

In (31), the leftmost WL sequence in row (b) shows [BINMIN-φ>> MATCH-XP], in row (c) [BINMAX-φ >>MATCH-φ,] in row (d) [BINMAXBRANCH-φ>>MATCH-φ], and in row (e) [MATCH-MAXXP >>BINMAX-φ].

The overall constraint ranking, as diagrammed by OTW, is shown in (32), where we see five undominated constraints, alternating types of MATCH and BINARITY constraints as the main axis, and remaining interleaving constraints.
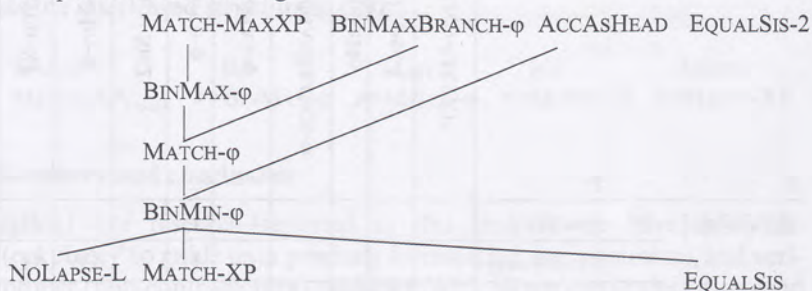
(32) Japanese constraint ranking:

NOLAPSE-L   MATCH-MAXXP   BINMAXBRANCH-φ   ACCASHEAD   EQUALSIS-2

BINMAX-φ

MATCH-φ

BINMIN-φ

MATCH-XP   EQUALSIS
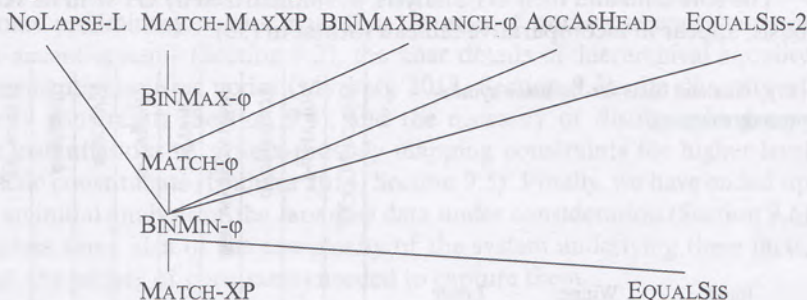
An additional bonus of OTW is that it not only produces the OT grammar under investigation but also the full factorial typology of the underlying constraint set. Within this typology, we indeed find the grammar generating the Basque system (Section 9.2.2), as shown in (33). It is instructive to compare it to the grammar of Japanese in (32) above.

(33)   Basque constraint ranking:

MATCH-MAXXP   BINMAXBRANCH-φ   ACCASHEAD   EQUALSIS-2

BINMAX-φ

MATCH-φ

BINMIN-φ

NOLAPSE-L   MATCH-XP   EQUALSIS

The only crucial difference between the two systems is the ranking of NOLAPSE-L and BINMIN-φ:

(34)   Japanese: [NOLAPSE-L >> BINMIN-φ]
       Basque: [BINMIN-φ >> NOLAPSE-L]

Less important is a difference in the ranking OF EQUALSISTERS-2. In the Basque system, it is not ranked w.r.t. BINMIN-φ because the input [[[*a*]*u*]*u*] comes out as (((*au*)*u*), that is, with the last *u* unphrased. This candidate fulfills both EQUALSISTERS-2 and BINMIN-φ, so no ranking of the two is required. In the Japanese system, the outcome is (((*a*)(*u*))(*u*)), fulfilling EQUALSISTERS-2 but violating BINMIN-φ. This means [EQUALSISTERS-2 >> BINMIN-φ].

The core data and their OT analysis, as summarized by OTW in its skeletal basis, appear in a comparative tableau format in (35).

(35) Skeletal basis for Japanese syntax-prosody mapping			MATCH-MAXXP	NOLAPSE-L	ACCASHEAD	BINMAXBRANCH-φ	EQUALSIS-2	BINMAX-φ	MATCH-φ	BINMIN-φ	MATCH-XP	EQUALSIS
	Input	Winner	Loser									
a.	[[[[u]u]u]u]	((uu)(uu))	(uu)(uu)	W					L	L		L	
b.	[[a]u]	((a)(u))	(au)		W						L	W	
c.	[[a]u]	((a)(u))	(a(u))			W					L	W	W
d.	[[[[u]u]u]u]	((uu)(uu))	(uuuu)				W			L		W	
e.	[[[a]u]u]	(((a)(u))(u))	(((a)(u))u)					W			L		W
f.	[[[[u]u]u]u]	((uu)(uu))	(((uu)u)(u))						W	L	W	L	W
g.	[[[u]u]a]	((uu)(a))	(u(ua))							W	L	W	W
h.	[[u]u]	(uu)	((u)u)								W	L	W
i.	[[[u]u]u]	((uu)u)	((uu)(u))								W		L

(36) Skeletal basis for Basque syntax-prosody mapping

	Input	Winner	Loser	Match-MaxXP	AccAsHd	BinMaxBranch-φ	EqualSis-2	BinMax-φ	Match-φ	BinMin-φ	NoLapse-L	Match-XP	EqualSis
a.	[[[u]u]u]u]	((uu)(uu))	(uu)(uu)	W				L	L			L	
b.	[[a]a]	((a)(a))	(aa)		W					L		W	
c.	[[[[u]u]u]u]	((uu)(uu))	(uuuu)			W			L			W	
d.	[[[[u]u]u]u]	((uu)(uu))	(((uu)u)(u))					W	L	W		L	W
e.	[[[u]u]a]	((uu)(a))	(u(ua))						W	L		W	W
f.	[[a]u]	(au)	((a)(u))							W	L	L	
g.	[[[u]u]u]	((uu)u)	((uu)(u))							W			L

In conclusion, it is remarkable that Match Theory (Selkirk 2011) has not only resulted in a more principled and more streamlined understanding of syntax-prosody mapping relations, but has also opened up new perspectives on many prosodic well-formedness constraints, as they impinge on the otherwise expected isomorphism between syntactic structure and prosodic form. Using well-known generalizations about the way phrasal structures in Japanese are parsed into prosodic units as a testing ground, this chapter has reported on some preliminary results regarding the role of anti-lapse constraints in pitch accent systems (Section 9.2), the finer details of hierarchical equality requirements on sister nodes (Myrberg 2013, Section 9.3), the diversity of binarity constraints (Section 9.4), and the necessity of distinguishing separate instantiations of syntax-prosody mapping constraints for higher-level syntactic constituents (Ishihara 2014; Section 9.5). Finally, we have ended up with an initial analysis of the Japanese data under consideration (Section 9.6) that gives some idea of the complexity of the system underlying these facts, and of the variety of constraints needed to capture them.

Notes

Foley, Nick Kalivoda, and Jason Ostrove as well as Jim McCloskey and Maziar Toorsanvandi. We are responsible for all remaining errors and shortcomings.

2   NoLapse fits into a larger family of tonal anti-lapse constraints that are at work in pitch accent languages, such as Ancient Greek (see Ito and Mester 2013: 31).

3   I.e., a φ not dominating another φ, as defined in Ito and Mester 2007, 2013.

4   The tableaux below are violation tableaux with added comparative markings (Prince 2000), with W's and L's appearing in the rows of losing candidates. "W" in a constraint column indicates the winner is favored by the constraint, "L" indicates the loser is favored, and no entry indicates a tie (i.e., the violation marks for the winner equal those of the loser). In order for a ranking tableau to be consistent, each L has to be preceded by a W in its row (in order to win, the winner needs to do better than each loser on the highest-ranked constraint that distinguishes the two, in Jane Grimshaw's succinct phrasing).

5   Formally called *Adjunction, under which name it appears in Selkirk and Elordieta 2010.

6   This tableau and the following are our interpretation (Selkirk and Elordieta 2010 provide no tableaux).

7   The winning candidate here is (*au*), which we take to be the correct outcome since we do not know of any evidence showing that the leading *a* is a separate phrase, as in Selkirk and Elordieta 2010. If there is indeed reason to assign the parse ((*a*)*u*), it will be the predicted winner once Align-R is added to the system and ranked above MinBin.

8   The definition here is slightly revised from that in Ishihara 2014, which requires a matching $\varphi_{[+max]}$. We require only a matching φ in order to allow for prosodic cliticization to $XP_{[+max]}$, for example.

9   OTWorkplace_X_83, version of 27 June 2015. The program developed by Alan Prince, Bruce Tesar, and Naz Merchant is open-source and distributed without charge, downloadable from https://sites.google.com/ site/otworkplace/. OTWorkplace is a software suite that, in the words of its authors, "uses Excel as a platform for interactive research with the analytical tools of modern rigorous OT".

Bibliography

Chomsky, N., and Halle, M. (1968) *The sound pattern of English*. New York: Harper & Row.

Elfner, E. (2012) Syntax-prosody interactions in Irish. Ph.D. Diss. University of Massachusetts, Amherst.

Elordieta, G. (2007) "Minimum size constraints on intermediate phrases". Retrieved 15 August 2015, www.icphs2007.de/conference/Papers/1682/1682.pdf.

Féry, C. (2010) "Recursion in prosodic structure", *Phonological Studies*, 13, pp. 51–60.

Gussenhoven, C. (2004) *The phonology of tone and intonation*. Cambridge: Cambridge University Press.

Gussenhoven, C. (2005) "Procliticized phonological phrases in English: Evidence from rhythm", *Studia Linguistica*, 59(2/3), pp. 174–193.

Hulst, H. van der. (2010) "A note on recursion in phonology" in Hulst, H. van der (ed.) *Recursion and human language*. Berlin: Mouton de Gruyter, pp. 301–342.

Ishihara, S. (2014) "Match Theory and the recursivity problem" in Kawahara, S., and Igarashi, M. (eds.) *Proceedings of FAJL 7: Formal Approaches to Japanese Linguistics*. MIT Working Papers in Linguistics 73. Cambridge, MA: MIT Press, pp. 69–88.

Ito, J., and Mester, A. (1992) "Weak layering and word binarity". Linguistic Research Center Working Paper LRC-92-09. UC Santa Cruz. Published in Honma, T., Okazaki, M., Tabata, T., and Tanaka, S. (eds.) (2003) *A new century of phonology and phonological theory: A festschrift for Professor Shosuke Haraguchi on the occasion of his sixtieth birthday*. Tokyo: Kaitakusha, pp. 26–65.

Ito, J., and Mester, A. (2007) "Prosodic adjunction in Japanese compounds" in Miyamoto, Y., and Ochi, M. (eds.) *Formal approaches to Japanese linguistics: Proceedings of FAJL 4*. Cambridge, MA: MIT Department of Linguistics and Philosophy, pp. 97–111.

Ito, J., and Mester, A. (2009a) "The extended prosodic word" in Grijzenhout, J., and Kabak, B. (eds.) *Phonological domains: Universals and deviations*. Berlin: Mouton de Gruyter, pp. 135–194.

Ito, J., and Mester, A. (2009b) "The onset of the prosodic word" in Parker, S. (ed.) *Phonological argumentation: Essays on evidence and motivation*. London: Equinox, pp. 227–260.

Ito, J., and Mester, A. (2013) "Prosodic subcategories in Japanese", *Lingua*, 124, pp. 20–40.

Kabak, B., and Revithiadou, A. (2009) "An interface approach to prosodic word recursion" in Grijzenhout, J., and Kabak, B. (eds.) *Phonological domains: Universals and deviations*. Berlin: Mouton de Gruyter, pp. 105–134.

Kubozono, H. (1993) The organization of Japanese prosody. Ph.D. Diss., University of Edinburgh. Tokyo: Kurosio Publishers.

Kubozono, H. (1989) "Syntactic and rhythmic effects on downstep in Japanese", *Phonology*, 6(1), pp. 39–67.

Ladd, D. R. (1986) "Intonational phrasing: The case for recursive prosodic structure", *Phonology Yearbook*, 3, pp. 311–340.

Ladd, D. R. (1988) "Declination 'reset' and the hierarchical organization of utterances", *Journal of the Acoustical Society of America*, 84, pp. 530–544.

McCawley, J. D. (1968) *The phonological component of a grammar of Japanese*. The Hague, The Netherlands: Mouton.

Myrberg, S. (2013) "Sisterhood in prosodic branching", *Phonology*, 30(1), pp. 73–124.

Nespor, M., and Vogel, I. (1986) *Prosodic phonology*. Dordrecht: Foris.

Pierrehumbert, J., and Beckman, M. (1988) *Japanese tone structure*. LI Monograph Series No. 15. Cambridge, MA: MIT Press.

Prince, A. S. (2000) Comparative tableaux. Rutgers Optimality Archive. ROA-376. Retrieved 15 August 2015 from http://roa.rutgers.edu/.

Prince, A. S., and Smolensky, P. (2004) *Optimality theory: Constraint interaction in generative grammar. RuCCS-TR-2*. Rutgers University, Brunswick, NJ, and University of Colorado, Boulder. Malden, MA: Blackwell.

Schreuder, M. (2006) Prosodic processes in language and music. Ph.D. Diss., Rijksuniversiteit Groningen, The Netherlands.

Schreuder, M., and Gilbers, D. (2004) "Recursive patterns in phonological phrases" in Bel, B., and Marlien, I. (eds.) *Proceedings of Speech Prosody 2004*. Nara, Japan: ISCA (SproSig), pp. 341–344.

Selkirk, E. (1984) *Phonology and syntax: The relation between sound and structure*. Cambridge, MA: MIT Press.

Selkirk, E. (1986) "On derived domains in sentence phonology", *Phonology*, 3, 371–405.

Selkirk, E. (1996) "The prosodic structure of function words" in Morgan, J. L., and Demuth, K. (eds.) *Signal to syntax*. Mahwah, NJ: Lawrence Erlbaum, pp. 187–213.

Selkirk, E. (2009) "On clause and intonational phrase in Japanese: The syntactic grounding of prosodic constituent structure", *Gengo Kenkyuu*, 136, 35–73.

Selkirk, E. (2011) "The syntax-phonology interface" in Goldsmith, J. A., Riggle, J., and Yu, A. (eds.) *The handbook of phonological theory*. 2nd edition. Oxford: Blackwell, pp. 435–485.

Selkirk, E., and Elordieta, G. (2010) "The role for prosodic markedness constraints in phonological phrase formation in two pitch accent languages". Handout of paper presented at Tone and Intonation in Europe (TIE) 4, Stockholm University, Department of Scandinavian Languages. Retrieved 15 August 2015, www.hum.su.se/pub/jsp/polopoly.jsp?d=13236&a=73666.

Shinya, T., Selkirk, E., and Kawahara, S. (2004) "Rhythmic boost and recursive minor phrase in Japanese" in Bel, B., and Marlien, I. (eds.) *Proceedings of Speech Prosody 2004*. Nara, Japan: ISCA (SproSig) pp. 345–348.

Vance, T. J. (2008) *The sounds of Japanese*. Cambridge: Cambridge University Press.

Wagner, M. (2005) Prosody and recursion. Ph.D. Diss., Massachusetts Institute of Technology.

Wagner, M. (2010) "Prosody and recursion in coordinate structures and beyond", *Natural Language and Linguistic Theory*, 28, pp. 183–237.