Acta Linguistica Hungarica, Vol. 57 (4), pp. 357–380 (2010)
DOI: 10.1556/ALing.57.2010.4.1

GP 2, AND PUTONGHUA TOO

SAŠO ŽIVANOVIĆ[a] – MARKUS ALEXANDER PÖCHTRAGER[b]

[a]Univerza v Ljubljani
Aškerčeva 2, 1000 Ljubljana, Slovenia
saso.zivanovic@guest.arnes.si

[b]Fen-Edebiyat Fakültesi, Boğaziçi Üniversitesi
Batı Dilleri ve Edebiyatları Bölümü
34342 Bebek/İstanbul, Turkey
markus.pochtrager@boun.edu.tr

Abstract: The article illustrates some of the salient features of Government Phonology (GP) 2.0 by axiomatising (a subclass of) the set of possible Putonghua forms.

We show that a phonological theory can profit by assuming that phonological representations are hierarchical, just like syntactic representations. A structural relation of c++command, a relative of the well-known c-command, is used heavily. The similarity with syntax is further underlined by the introduction of a phonological Binding Theory: illicit representations are prohibited by the LUxI Principles, the phonological counterpart of Principles A, B and C.

Keywords: phonology, Putonghua, hierarchical structures, c-command, binding

1. The name of the game

This article is designed to be a showcase for some of the salient features of Government Phonology (GP) 2.0. The choice of language is Putonghua, a somewhat artificial northern dialect of the Han language family and designated as the standard language of the People's Republic of China—roughly equivalent to the RP of the United Kingdom. Putonghua is an ideal "laboratory animal" in that it is rather fussy about what can occur with what in the internal structure of its constituent structure. The elements involved in Putonghua phonology are also subject to a variety of positional constraints.

Our goal, then, is to subject Putonghua data to analysis based on the nascent and very much immature theory we call GP 2.0. Such analyses are vital to the development of any would-be theory and will permit us to more accurately see the strengths and deficiencies of its current state as well as a fine tuning of the more successful ones of its postulates and, of course, the disposal of non-functioning components along with the formulation of entirely new ones.

Given these rather ambitious goals, as well as the limited space at our disposal, we are sadly unable to offer an analysis of the entire Putonghua onset–rime system. Most notably, we will not be dealing with the **L**-element in this study. As a consequence the role of the nasals *n* and *ŋ* in Putonghua rimes will not be addressed.

The present work takes earlier GP analyses as its starting point. Most notably, Goh (1996), Kaye (2001), Neubarth–Rennison (2002), Ferme (2002), Ferme–Živanović (2006) and Ferme (2009) provide the bulk of the data and the analysis which we now attempt to interpret according to the principles of GP 2.0. Readers familiar with classical analyses of Putonghua onset–rimes will notice certain differences between our analytic assumptions and those of the past. Most notably the division of labour between onset and rime is organised in a somewhat different way. To illustrate these differences let us consider the onset–rime sequence *kuai* (sometimes rendered as *kwai*). The traditional assumption is that the constituent division occurs between *k* and *uai*: *kuai*=*k*+*uai*. Kaye (2001) offers distributional arguments to suggest that this division is incorrect; in reality the division is *kw*+*ai*. This has the advantage of providing a much more uniform distribution of the rime *ai* and avoiding the postulation of a rime, *uai*, only occurring following velars (*k, g, h*) and so-called retroflex consonants (*ch, zh, sh*). This move involves the establishment of a labio-velar and labio-retroflex series of consonants: k^w, g^w, h^w and ch^w, zh^w, sh^w, respectively. This move also introduces a natural connection to the so-called alveo-palatal series, *q, j, x* which, for reasons of consistency, could just as well be transcribed with a superscript *y* (e.g. q^y). This move then eliminates the spurious rimes, *-iaŋ, -iau*, etc. which are more correctly rendered as *-aŋ* and *-au* which exist independently of this series. Thus, *q*+*iaŋ* is rendered as q^y+*aŋ* in our system.

We have also incorporated a somewhat different view of the Putonghua nuclear base which again owes much to the earlier works cited above. The central idea behind this approach is that Putonghua uses no melodic elements (**I**, **U**, **L** in GP 2.0) in its nuclear heads (xN positions). It is limited to the purely structural configurations underlying *a, ə* and *ɨ*. All other nuclei are deemed to be derived in part from melodic elements residing in preceding onsets (e.g. *bo* where **U** originates in the onset *b*) or from elements found in the specifier or complement positions of the nucleus phrase (NP) (as in *duo* ← *duə* and *bei* ← *bəi*, respectively).

From the above it also follows that Putonghua must possess two flavours of *l* and *n*: plain, as in *lu* and *nu*, and palatal (l^y and n^y), as in *lü* and *nü*. This follows from the claim that melodic elements are not present in Putonghua nuclear heads; *ü* consists of the element **U** residing in the specifier of NP and **I** originating in the onset. Our entire analysis of the set of Putonghua onsets and rimes is presented in Tables 1 and 2.

Table 1

Putonghua "syllables," part 1 (An = analysis, Pr = pronunciation, Py = pinyin)

An	_	_n	_ŋ	a	ai	an	aŋ	au	ə	əi	ən	əŋ	əu
Pr	ɨ/i/u	in/uɨn	iŋ/uŋ	a	ai	an	aŋ	au	ə/ɔ/ɛ	ei	ən/ɛn	əŋ	ou
Py	i	in/un	ing/ong	a	ai	an	ang	ao	e/o	ei	en/an	eng	ou
_				a	ai	an	aŋ	au	ə		ən		ou
b	bu			ba	bai	ban	baŋ	bau	bɔ	bei	bən	bəŋ	
c	cɨ			ca	cai	can	caŋ	cau	cə		cən	cəŋ	cou
ch	chɨ			cha	chai	chan	chaŋ	chau	chə		chən	chəŋ	chou
ch^w	chu	chuɨn	chuŋ	chua	chuai	chuan	chuaŋ		chuɔ	chuei			
d				da	da	dan	daŋ	dau	də	dei		dəŋ	dou
f	fu			fa		fan	faŋ		fɔ	fei	fən	fəŋ	fou
g				ga	gai	gan	gaŋ	gau	gə	gei	gen	gəŋ	gou
g^w	gu	guɨn	guŋ	gua	guai	guan	guaŋ		guɔ	guei			
h				ha	hai	han	haŋ	hau	hə	hei	hən	həŋ	hou
h^w	hu	huɨn	huŋ	hua	huai	huan	huaŋ		huɔ	huei			
j	ji	jin	jiŋ	jia			jiaŋ	jiau	jiɛ		jiɛn		jiou
k				ka	kai	kan	kaŋ	kau	kə		kən	kəŋ	kou
k^w	ku	kuɨn	kuŋ	kua	kuai	kuan	kuaŋ		kuɔ	kuei			
l				la	lai	lan	laŋ	lau	lə	lei		ləŋ	lou
l^y	li	lin	liŋ	lia			liaŋ	liau	liɛ		liɛn		liou
m	mu			ma	mai	man	maŋ	mau	mɔ	mei	mən	məŋ	mou
n				na	nai	nan	naŋ	nau	nə	nei	nən	nəŋ	nou
n^y	ni	nin	niŋ				niaŋ	niau	niɛ		niɛn		niou
p	pu			pa	pai	pan	paŋ	pau	pɔ	pei	pən	pəŋ	pou
q	qi	qin	qiŋ	qia			qiaŋ	qiau	qiɛ		qiɛn		qiou
r	rɨ					ran	raŋ	rau	rə		rən	rəŋ	rou
s	sɨ			sa	sai	san	saŋ	sau	sə		sən	səŋ	sou
sh	shɨ			sha	shai	shan	shaŋ	shau	shə	shei	shən	shəŋ	shou
sh^w	shu	shuɨn		shua	shuai	shuan	shuaŋ		shuɔ	shuei			
t				ta	tai	tan	taŋ	tau	tə			təŋ	tou
w	wu			wa	wai	wan	waŋ		wɔ	wei	wən	wəŋ	
x	xi	xin	xiŋ	xia			xiaŋ	xiau	xiɛ		xiɛn		xiou
y	yi	yin	yiŋ	ya			yaŋ	yau	yɛ		yɛn		you
z	zɨ			za	zai	zan	zaŋ	zau	zə	zei	zən	zəŋ	zou
zh	zhɨ			zha	zhai	zhan	zhaŋ	zhau	zhə	zhei	zhən	zhəŋ	zhou
zh^w	zhu	zhuɨn	zhuŋ	zhua	zhuai	zhuan	zhuaŋ		zhuɔ	zhuei			

Table 2

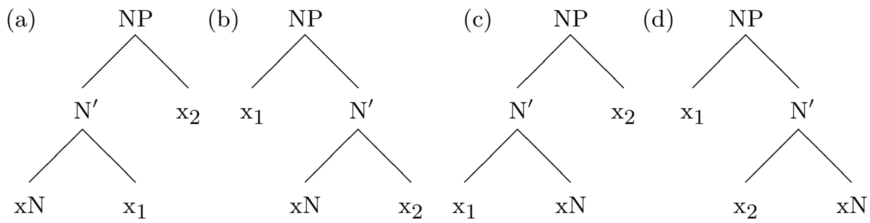
Putonghua “syllables,” part 2 (An = analysis, Pr = pronunciation, Py = pinyin)

An	i	iən	iau	iə	iəu	in	iŋ	u	uan	uə	uəi	uən	u_n	uŋ
Pr	i	iɛn	iau	iɛ	iou	in	iŋ	u/ü	uan	uɔ/üɛ	uei	ʐɛn	uɨn/ün	uŋ
Py	i	ian	iao	ie	iu	in	ing	u	uan	uo/ue/üe	ui	uan	un	ong
_														
b	bi	biɛn	biau	biɛ		bin	biŋ							
c														
ch														
ch^{w}														
d	di	diɛn	diau	diɛ	diou		diŋ	du	duan	duɔ	duei		duɨn	duŋ
f														
g														
g^{w}														
h														
h^{w}														
j								jü		jüɛ		jüɛn	jün	juŋ
k														
k^{w}														
l								lu	luan	luɔ			luɨn	luŋ
l^{y}								lü		lüɛ				
m	mi	miɛn	miau	miɛ	miou	min	miŋ							
n								nu	nuan	nuɔ				nuŋ
n^{y}								nü		nüɛ				
p	pi	piɛn	piau	piɛ		pin	piŋ							
q								qü		qüɛ		qüɛn	qün	quŋ
r								ru	ruan	ruɔ	ruei		ruɨn	ruŋ
s								su	suan	suɔ	suei		suɨn	suŋ
sh														
sh^{w}														
t	ti	tiɛn	tiau	tiɛ			tiŋ	tu	tuan	tuɔ	tuei		tuɨn	tuŋ
w														
x								xü		xüɛ		xüɛn	xün	xuŋ
y								yü		yüɛ		yüɛn	yün	yuŋ
z								zu	zuan	zuɔ	zuei		zuɨn	zuŋ
zh														
zh^{w}														

2. The structure of the rime

The main theoretical goal of the present paper is to argue that phonological domains exhibit hierarchical organisation. In particular, we believe that phonological structures bear a great resemblance to syntactic structures, as known in minimalist syntax: following Pöchtrager (2006), we assume that phonological domains are endocentric binary branching structures.[1]

In syntax, most researchers agree that every head projects twice, merging first with a complement and then with a specifier, and that the linear order of the head, the complement and the specifier is universally determined: the specifier precedes the head, which in turn precedes the complement (1b). We do not feel confident in claiming the same for phonological structures. First, Pöchtrager (2006), dealing with English and Estonian, proposed structures containing up to four projections. Second, concentrating on the first two projections (and it is only these that are relevant for our discussion of Putonghua rimes), he proposed different linear orderings for different categories. The linear orderings proposed for the NP and OP (nucleus and onset phrase) domain are the reverse of each other: head–complement–specifier (1a) and specifier–complement–head (like (1d), but imagine O instead of N), respectively. Third, his ordering, which works for English and Estonian, does not seem suitable for the analysis of Putonghua. Traditionally, a full-blown Putonghua rime consists of three positions: onglide, nucleus and offglide. Since we take the rime to be an NP and the nucleus an xN, the structure of the Putonghua rime must be either (1b) or (1c). We are thus forced to assume, at least temporarily, that the linear order of head, complement and specifier is language-specific.

(1)


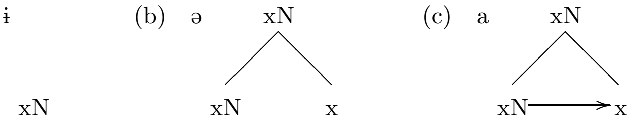
(xs are subscripted for ease of reference only; subscripts have no theoretical significance. The same goes for bar level marks (′, P) following the category label, as in syntax.)

Before we can discover whether (1b) or (1c) is the structure of the Putonghua rime, we have to explore the structure of its nuclear head. As discussed in section **1**, the only lexical nuclei of Putonghua are *ɨ, ə* and *a*.

[1] Due to space restrictions we cannot present the details of Pöchtrager's theory here. We ask the reader unfamiliar with the theory to refer to Pöchtrager (*op.cit.*, §2.4).

Their GP 2.0 representations, adopted from Kaye–Pöchtrager (2009), are given in (2). *ə* and *a* are **adjunction** structures: the result of the merger of terminal xN with its sister is again labeled xN. They differ in the presence (*a*) vs. absence (*ə*) of a **control** relation between xN and x.[2] We prohibit other lexical nuclei by assuming PTH1 and PTH2.[3]

(2) (a) ɨ (b) ə (c) a



PTH1 *In Putonghua, xN may not be annotated.*

PTH2 *In Putonghua, the sister of a terminal xN must be an unannotated terminal.*

Now consider the surface realisation of the nucleus. All vowels but *ɨ*, *ə* and *a* are the result of adjacent glides colouring the nucleus, *i*-glides yielding *e* or *ɛ*, and *u*-glides yielding *o* or *ɔ*, as illustrated in (3). Note that onglides and offglides yield open and closed mid vowels, respectively.

(3) diɛ, dei, duɔ, dou

What happens if both onglide and offglide are present in the rime? Which will colour the nucleus? Does the answer depend on the melody, and will it be either "**I**-glide" or "**U**-glide" (or "both")? Or does it depend on the structure, and will be one of "onglide" and "offglide" (and "both")? The data, presented in the table in (4), which shows the six possible scenarios about the realisation of the two lexical entries, is extremely clear on this point: it is the offglide that colours the nucleus.[4]

[2] Control is a special relation between a head and its sister. For the definition of its meaning, see Pöchtrager (2006) and Kaye–Pöchtrager (2009). For present purposes it is enough to say that it keeps *a* and *ə* apart.

[3] Since PTH2 is formulated without reference to adjunction, it does a bit more than simply prevent the annotation of *x* in the structures of (2). Why this is desired will become clear in section **3.3**.

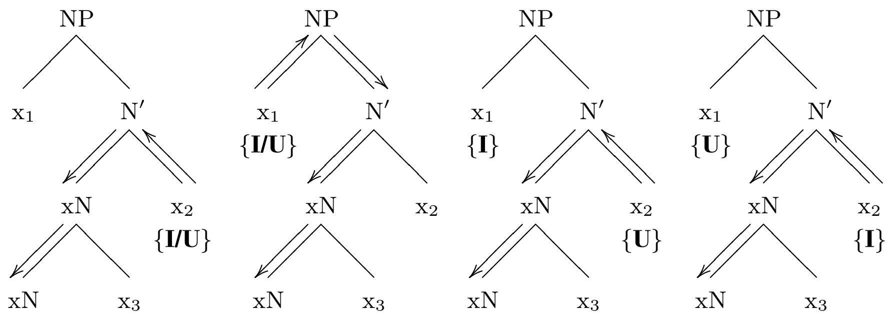
[4] The predictions adopt the observation that open *ɛ* and *ɔ* are the result of **I**- and **U**-colouring from the onglide, while closed *e* and *o* are the result of **I**- and **U**-colouring from the offglide.

(4)

	"onglide"	"offglide"	"both"	"**I**"	"**U**"	"both"
d+i+ə+u	*diɛu	diou	*diöu	*diɛu	diou	*diöu
d+u+ə+i	*duɔi	duei	*duɔ̈i	duei	*duɔi	*duɔ̈i

Since the offglide takes precedence in colouring the nucleus, we assume that it is hierarchically closer to the nucleus than the onglide: the structure of the Putonghua rime is (1b). Specifically, we propose the structures in (5) for the rimes with a lexical *ə* surrounded by *i* and/or *u*. Note that we assume that both x_1 and x_2 are always present, even if the rime contains no onglide or offglide at the surface.

(5) (a) (d)ei/(d)ou (b) (d)iɛ/duɔ (c) (d)iou (d) (d)uei

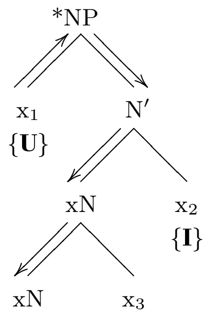


Conforming to (2), the trees in (5) are representations of rimes with lexical *ə* because xN branches into an adjunction structure $[_{xN}$ xN $x_3]$ where (the lower) xN does not control x_3. The surface quality of the nucleus (*e, ɛ, o, ɔ*) is the result of xN being **m-commanded** by an **I**- or **U**-annotated terminal.

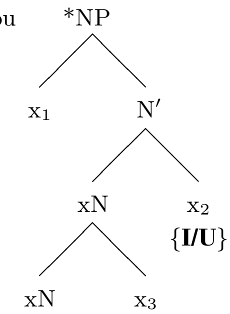
M-command is a novel concept introduced by Pöchtrager (2006). At this point, it suffices to note that it takes over the function of spreading used in autosegmental theories; this is the job it performs in (5). We will further discuss m-command in section **3**. (M-command relations are graphically shown using an **m-command path** from the m-commander to the m-commandee. Each tree in (5) contains only one m-command relation: in (5b), x_1 m-commands xN, and in the other trees x_2 m-commands xN.) Note furthermore that elements are annotations to terminals, instead of being associated to them by association lines.

Of course, we have not yet explained why the starred forms in (4) are ungrammatical. The data in (4) told us that the offglide takes precedence in colouring the nucleus. Our intuition was then to assume that the offglide is hierarchically closer to the nucleus than the onglide; i.e., that (1b) is the structure of the Putonghua rime. The formal account of the data in (4), however, still remains to be worked out. Currently, we are still facing the question why the m-command relations in (5) are just the way they are. Why, for example, is (6a), which equals (5d) but with xN being m-commanded by x_1 instead of x_2, ungrammatical? Similarly, why is there no (6b), whose structure would equal (5a) minus m-command? In the following section, we will work out a system which accounts for these facts.

(6) (a) *(d)uoi



(b) *(d)əi/(d)əu



3. Rime-internal restrictions

We know that both the onglide and the offglide can be either an *i*-glide or an *u*-glide, see (3). However, if a form contains both an onglide and an offglide, not each of the four logically possible combinations is attested, (7).[5] Furthermore, the attested possibilities vary with respect to the lexical nucleus. While a lexical *ə*-nucleus disallows only cooccurrences of the onglide and the offglide containing the same melody (7), a lexical *a*-nucleus is happy only with the combination of an *i*-onglide and an *u*-offglide (8).

(7) *diei, diou, *duou, duei

(8) *diai, diau, *duau, *duai

[5] In listing the unattested forms in (7)–(9), which contain both an on- and an offglide, we assume the following: (i) the nucleus *a* is coloured by neither onglide nor offglide, cf. Tables 1 and 2; (ii) the nucleus *ə* is coloured (only) by the offglide, and the result is a closed mid-vowel, cf. Tables 1 and 2 and the discussion in section **2**.

We analyze the observed restrictions as falling into two categories, with respect to whether the onglide and the offglide contain the same melody (homomelodic) or not (heteromelodic). Homomelodic restrictions are stricter. (i) Combinations of onglides and offglides containing the same melody are disallowed both by *ə*-nuclei and *a*-nuclei. (ii) They are disallowed even in the "surface" rimes (9), where we have argued that the "onglide" is not actually a part of the rime, but rather a part of the onset. (10) shows that this is in contrast to the prohibition against the combination of an *u*-onglide and an *i*-offglide illustrated in (8).

(9) *jiai, *jiei, *guau, *guou

(10) guai, guei

Homomelodic and heteromelodic restrictions will be accounted for by Universal 1 in subsection **3.2** and Universal 3 in subsection **3.3**, respectively. (We have decided to keep the accounts for homo- and heteromelodic restrictions separate, although we hope that future work will provide an elegant unified account.) Before we plunge into that, however, we provide some tools necessary for the job.

3.1. M-command and c++command

Pöchtrager (2006) replaces the association lines of GP 1.x and other autosegmental theories with annotation of terminals and m-command.[6] We have seen an example of this in (5) in section **2**. Pöchtrager's definition of m-command is given below.

(11) Melodic command (m-command) (from Pöchtrager 2006, 68)

(a) M-command is a binary relationship between two terminals, an m-commander and an m-comandee.

(b) Only heads (xN, xO) can be m-commanders.

(c) Only non-heads (unannotated x's) can be m-commandees.

(d) An m-commandee can be m-commanded only once, but an m-commander can m-command several times.

(e) An m-commanded point receives the same interpretation as its m-commander.

[6] However, the reader should be aware that there are significant formal differences between annotation and m-command on one hand, and association lines on the other. For a discussion of advantages of the former, see Pöchtrager (2006), Kaye–Pöchtrager (2009).

Our usage of m-command in section **2** was clearly at odds with the above definition. We had reversed the direction of the relation: non-heads were m-commanding the heads. Most interestingly, however, this turns out to be an advantage. What Pöchtrager's definition of m-command is missing is any kind of geometric constraint on m-command: the m-commander and m-commandee could be anywhere in the tree. Of course, Pöchtrager discusses the geometric constraints on a case-by-case basis, but never provides a generally valid constraint. However, such a constraint can be found! Its existence is mainly obscured by the habit of searching for a source–to–target oriented generalisation. Reversing the logic reveals a geometric constraint on m-command, employing the notion of c++command given below.

Definition 1 *Node α c-commands node β iff*

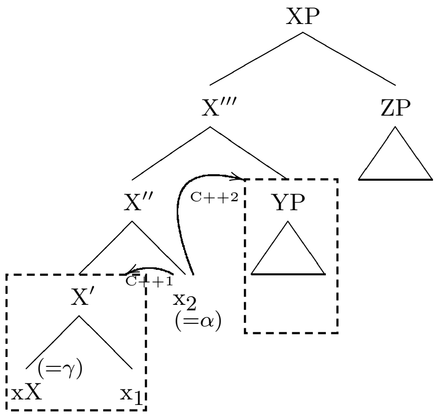
(C1) β is α's sister, or

(C2) α's sister dominates β.

Definition 2 *Node α c++commands node β iff*

(C++1) α c-commands β, or

(C++2) α is the highest terminal in the maximal projection immediately containing α (i.e. α is a terminal, α is a daughter of a projection of some γ, and α c-commands all other terminals which are daughters of some projection of γ), and α's mother c-commands β.



Going through Pöchtrager (2006) one can observe that in virtually all cases, the m-commandee c++commands (but not always c-commands!) the m-commander. We believe that this is a good reason to reverse the directionality of the m-command relation, to arrive at the usual situation where the "source" of the relation is hierarchically higher than its "target". We thus propose the following definition of m-command.

Definition 3 (M-command)

(M1) M-command is a binary relationship between two terminals, an m-commander and an m-comandee.

(M2) Only non-heads can be m-commanders.

(M3) Only heads can be m-commandees.

(M4) The m-commander must c++command the m-commandee.

(M5) An m-commander can m-command only once.

The above definition gives only the structural requirements on the m-command relation. Next, there is also an issue of interpretation. Pöchtrager (2006) deals only with issues of length, so his (11e) must be expanded. We do not provide a formal interpretation algorithm. Informally, we assume that an unannotated m-commander (under our reversed directionality) "transfers" to the m-commandee its length, while an annotated m-commander "transfers" its melody. For some complications, see section **4.2**.

Universal 1 in section **3.2** and the definitions of island and potential binder in section **3.3** will employ c++command. It should be noted, however, that the data in these two subsections, where we deal with rime-internal matters only, could just as well be explained by relying on the simpler notion of c-command. Rime-internally, c-command and c++command are extensionally equivalent, since the highest terminal node (x_1) of the only maximal projection (NP, the rime) is a daughter of the root node, which does not c-command anything (see (1b)). C++command will become crucial in section **4**.

Finally, note a difference between m-command and control on one side, and c-command, c++command, islands and (potential) binding on the other side. M-command and control are contingent relations. A lexical representation can either contain the m-command or control relation between two terminals or not (subject to certain constraints, of course). M-command relations can also be created in the course of derivation, resulting in so-called spreading or lengthening. C-command, c++command, islands and (potential) binding, on the other hand, cannot be lexical. They are merely tools we use to express the structural relationships in a tree. They can be read off the phonological structures. When we represent them graphically, we do this only for the reader's convenience.

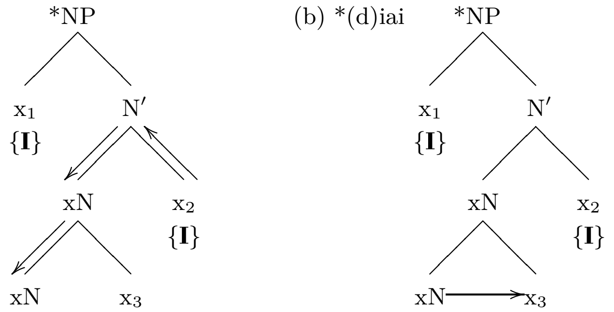
3.2. No self-c++command

Homomelodic restrictions depend solely on the geometry of the structure, and can be accounted for as follows.

Universal 1 (No self-c++command) *A structure containing a terminal annotated with some element c++commanding another terminal annotated with the same element is illicit.*

To see Universal 1 in action, consider the unacceptable forms **diei* and **diai*, represented in (12). The structures are ungrammatical, since the **I**-annotated x_1 c-commands (and thus c++commands) the **I**-annotated x_2.

(12) (a) *(d)iei (b) *(d)iai



Using the notion of c++command (or simple c-command) in Universal 1 allows us to express constraints holding among non-adjacent terminals—e.g. in (12), x_1 and x_2 interact although they are separated by an interpreted nucleus[7]—without claiming that any two terminals interact. For example, while we assume (as other GP practitioners do) that palatal consonants contain the element **I**, we would not want to prohibit cooccurrences of a palatal and an *i*-offglide: e.g. *shai* and *shei* are perfectly well-formed. We will show how this works in section **4.1**; that Universal 1 relies on c++command and not c-command will turn out to be crucial. Now we turn to heteromelodic restrictions.

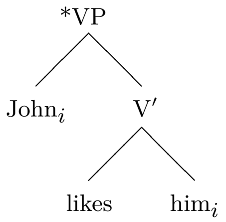
3.3. Binding Theory

We deal with heteromelodic restrictions by taking a leaf out of the syntacticians' book: we will define a **binding** relation, and declare structures (not) containing certain binding relations illicit. Remember how binding works in syntax. In (13a), *John* binds *him*, because (i) *John* c-commands *him*, and (ii) *John* and *him* are coindexed. Structure (13b) is then declared illicit by Principle B, which states that pronouns must not be bound in a local domain.

[7] Neubarth and Rennison (2002) formalise this particular interaction with the notion of **bridge**. If both the onglide and the offglide contain **I** (or **U**), they form a bridge which colours the intervening nucleus. The authors employ bridging to analyze forms such as *diɛ* and *duɔ*. An immediate problem is that they have to assume that **I** (or **U**) is not interpreted in the offglide.

(13) (a) *John$_i$ likes him$_i$.

(b)



The definition of phonological binding is given below. The work of declaring certain structures illicit, done by Principles A, B and C in syntax, is delegated to the **LUxI Principles**, given in Universal 3. We shall immediately provide the final version of the definitions and universals, and then discuss which data motivates which clause of the definitions. We start with the notion of **island**.

Definition 4 (Island) *A constituent is an **island** iff it is the smallest constituent containing an m-commander or a controller and its c++commanding domain.*

As an m-commander must c++command the m-commandee, an m-command island will always contain both m-commander and m-commandee. The same holds for control islands, since a head can only control its sister.

Let us see how the notion of islands can work. We propose the following universal, which makes m-command sensitive to islands.

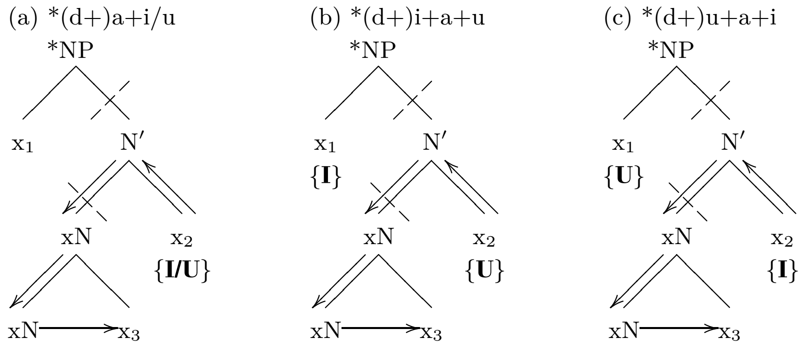
Universal 2 (No m-command into islands) *If a structure contains nodes α and β such that (i) α m-commands β, and (ii) there is an island containing β but not α, it is illicit.*

The above universal is motivated by the cross-linguistic observation about the stability of lexical *a*. For example, in Turkish vowel harmony, **I** spreads only into *ə*, but not into *a* (Pöchtrager 2010). In GP 2.0 terms: x{**I**} can m-command an xN that does not control and is thus not hidden in an island, whereas it cannot m-command an xN which controls and thus is hidden in an island, cf. (2).

In Putonghua, Universal 2 prohibits m-command from x_2 to xN in the structures in (14). Allowing this m-command relation would be incorrect, simply since there are no realisations that correspond to them, cf. Tables 1 and 2. (Islands are graphically represented by a dashed line above the root node of the island.)

After defining islands and showing how they influence m-command, we finally turn to the heart of the phonological binding theory, (potential)

(14)



binding and the LUxI Principles, which also depend on the notion of island.

Definition 5 (Potential binder) *Node α is a* ***potential binder*** *of node β iff*

(B1) α and β are non-head terminal nodes,

(B2) α asymmetrically c++commands β (i.e. α c++commands β, but β does not c++command α),

(B3) there is no island containing β but not α, and

(B4) α does not m-command.

Definition 6 (Binding) *Terminal α* ***binds*** *terminal β iff*

α is a potential binder of β, and

(B5) there is no closer potential binder of β (i.e. the smallest constituent containing both α and β contains no other potential binder of β).

Universal 3 (The LUxI Principles)

Principle L *(To be announced.)*

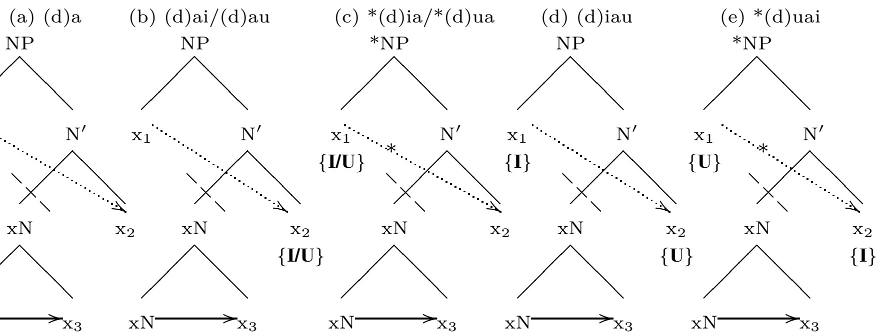
Principle U *(No requirements.)*

Principle x *A structure containing an annotated terminal binding an unannotated x is illicit.*

Principle I *A structure containing an annotated terminal binding x{**I**} is illicit.*

To see binding theory in action, consider *a*-nucleus rimes, represented in (15). The binding relations are the same for all forms in (15)—annotations do not play a role in determining the binding relations. By (B1), only non-head terminal nodes (x_1, x_2 and x_3) can participate in binding. The control relation between xN and x_3 makes the upper xN an island. Thus, by (B3), the only binding relation can be between x_1 and x_2. The requirement of (asymmetric) c++command (B2) makes it only possible for x_1 to bind x_2. This binding relation also satisfies (B4) and (B5). (A binding relation is graphically represented with a dotted arrow.)

(15)

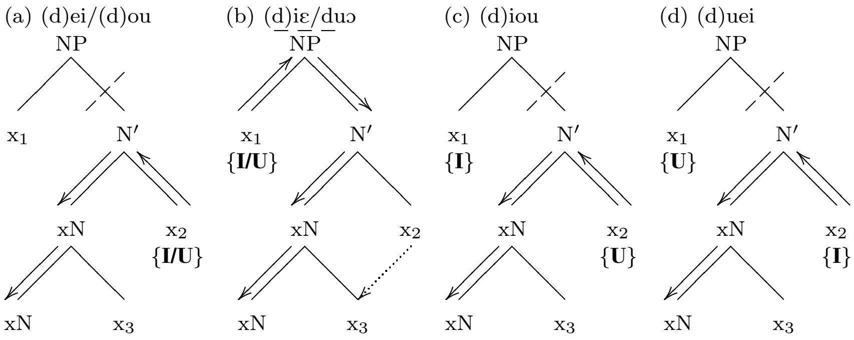


We now turn to the LUxI Principles. The forms **dia* and **dua* in (15c) are ruled out by Principle x: an unnanotated terminal (x_2) is bound by an annotated terminal (x_1). **duai* in (15e) is ruled out by Principle I: **I**-annotated x_2 is bound by an annotated x_1. The other forms are fine, though. *da* in (15a) is grammatical, since Principle x only disallows binding of unannotated terminals by annotated terminals; unannotated terminals binding unannotated terminals is allowed. Similarly for *dai* in (15b): it is grammatical, since Principle I only disallows binding of **I**-annotated terminals by annotated terminals; unannotated terminals binding **I**-annotated terminals are fine. Forms *dau* in (15b) and *diau* in (15d) are ok, since Principle U is silent: an **U**-annotated terminal does not care whether it is bound or not, or what binds it.

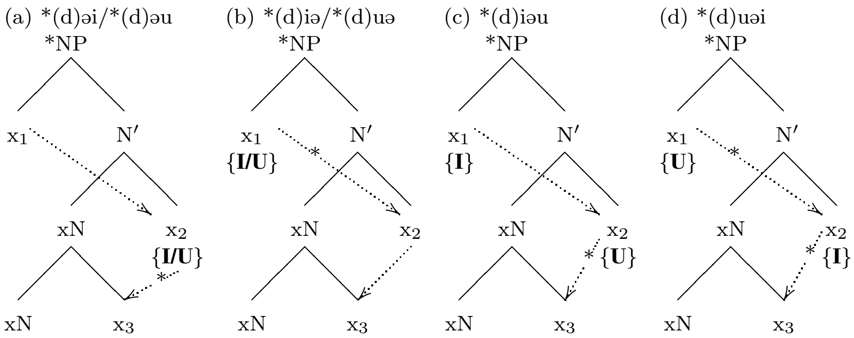
Next, we want to illustrate how m-command influences binding. Consider the forms with a lexical ə-nucleus. Their representations were already shown in (5). We repeat them below as (16), decorated by showing the islands and binding relations. The m-command relations from the annotated terminal (either x_1 or x_2) to xN prevent the creation of offending binding relations, either directly via the requirement that m-commanders cannot bind (B4), in (16b), or indirectly via m-command induced islands (B3), in the other structures. The structures of (16) with the m-command removed are shown in (17), decorated with the offending newly born binding relations.[8]

[8] Note that in (17), x_1 does not bind x_3. x_1 is a potential binder of x_3, but since x_2, which is c-commanded by x_1, is also a potential binder of x_3, x_1 does not bind x_3. However, the situation in (17) cannot be used as a motivation for the stipulation of (B5), since the structures are predicted to be illicit even in the absence of the x_1 to x_3 binding relation.

(16)



(17)

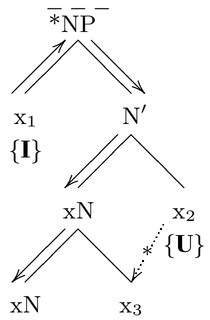


We are now in a position to answer the question posed at the end of section **2**. When both onglide and offglide are present in a rime (i.e., when both x_1 and x_2 are annotated), why can't the onglide colour the nucleus, i.e., why does adding an m-command from x_1 to xN in (17c) and (17d) not yield a grammatical structure? We can actually provide a genuine explanation of this observation. Consider what we would get: (18a) and (18b), where annotated x_2 binds unannotated x_3, in violation of Principle x. (The fact that NP is an island does not prevent the rise of this binding relation, since both x_2 and x_3 are contained in the island.)

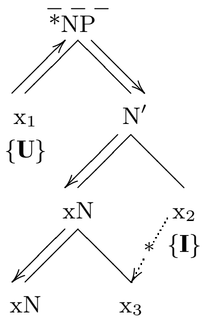
Let us now consider the final attested ə-nucleus form that remains to be discussed: *də*, shown in (19). We correctly predict it to be grammatical: it contains two binding relations, but none of them violate the LUxI Principles, specifically Principle x.

Finally, a loophole remains to be closed. Why couldn't a form like **diə* or **duə* arise from (20)? Nothing we have said so far makes (20) ungrammatical. We are forced to assume that Putonghua is parametrized so as to prohibit unannotated m-commanders.

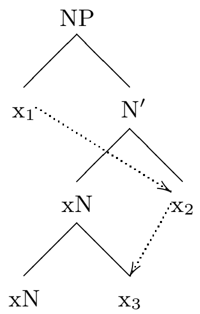
(18) (a) *(d)iɛu



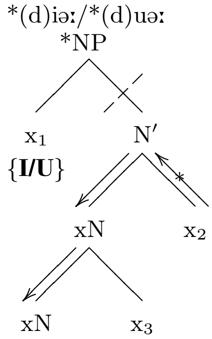
(b) *(d)uɔi



(19) (d)ə



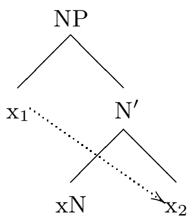
(20) *(d)iəː/*(d)uəː



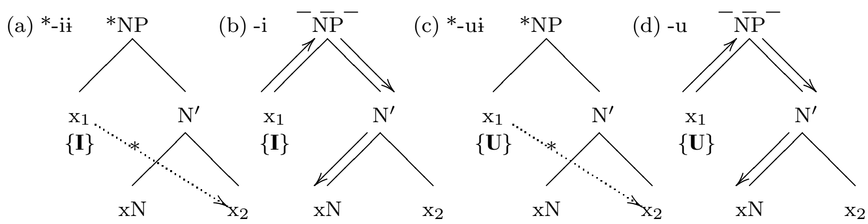
PTH3 *In Putonghua, m-commanders must be annotated.*

So far we have only dealt with *ə*- and *a*-rimes in this section. We now turn our attention to *ɨ*-rimes, i.e. rimes with a lexical nucleus *ɨ*. The simplest of such rimes, -*ɨ*, represented in (21), is clearly predicted to be grammatical by our system. (We leave open the question why it cooccurs with only a very limited set of onsets, see Tables 1 and 2.)

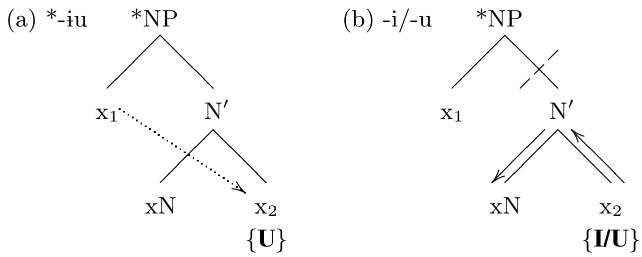
(21) (s)ɨ



PTH2, which states that the sister of a terminal xN must be an unannotated terminal, prohibits the annotation of x_2 in *ɨ*-rimes. Four structures thus remain to be considered, shown in (22). Binding theory prohibits the two without m-command, which would otherwise yield unattested forms. The forms with m-command are the attested rimes -*i* and -*u*.

(22)


Also note the following argument that the decision not to refer to adjunction in PTH2 was correct. Binding theory does not mark (23a) as ungrammatical—it is prohibited only by PTH2. Furthermore, (23b), prohibited by PTH2, is not declared illicit by the LUxI Principles. This is a correct result: although (23b) results in attested forms *-i* and *-u*, taking it as the actual representation of these rimes would predict wrong distributional properties of these rimes. Specifically, we would wrongly predict them to be combinable with palatal and labial onsets, respectively. (For discussion, see section **4.1**.)

(23)


This concludes our discussion of Putonghua rimes. We have set up a system which correctly axiomatizes the set of attested rimes. In the next section, we turn to the onset-rime interaction. We will see that our system is able to formalise this as well, if one adopts in addition (C++2), which has the effect of extending c-command to c++command, and (B5), which has the effect of making binding sensitive to structural distance. (Note that although we have included these clauses in the definitions, they have not yet played a decisive role in the discussion.)

4. Onset–rime interaction

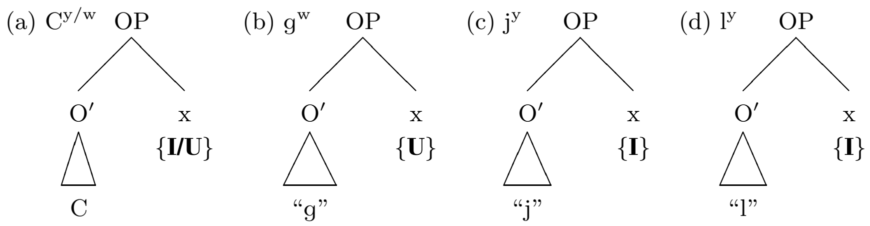
4.1. No self-c++command

We have argued in section **1** that not every surface onglide is a part of the rime. Some belong to the onset, as illustrated in (24). Such labialised and palatalised onsets cannot combine with a rime containing **U** and **I**, respectively, as shown in (25).

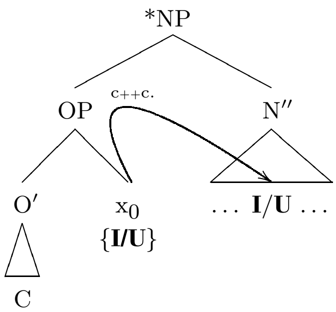
(24) gua = g^w + a, jia = j^y + a, lia = l^y + a

(25) *guau = g^w + au, *jiai = j^y + ai, *liai = l^y + ai

The restriction is of course the same as the restriction against homomelodic onglide and offglide in a true rime, shown in (7) and (8) and formalised using Universal 1. Can we use this universal to formalise the restriction illustrated in (25)? Yes, if we assume (26) to be the structure of palatalised/labialised onsets, where palatalisation/labialisation sits in the specifier of OP.

(26)


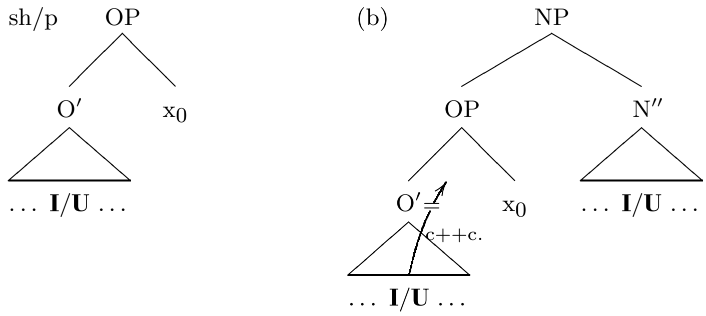
The forms in (25) are then prohibited by Universal 1, as shown in (27). Note that it is crucial that Universal 1 (No self-c++command) is not sensitive to simple c-command, but to c++command. x_0 c-commands only O′ and everything within it, but being the highest terminal in OP, it c++commands everything its mother c-commands: NP and everything within it.[9]

(27)


[9] Following Pöchtrager (2006), we assume that it is N that projects when the onset and the rime are merged.

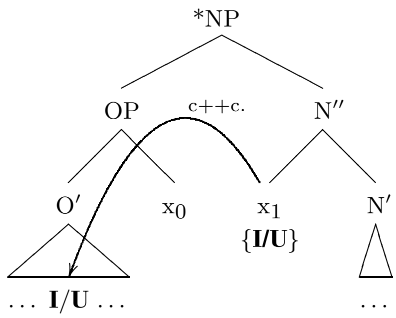
Importantly, Universal 1 does not prohibit a palatal (labial) onset coocurring with a rime containing **I** (**U**). Following Pöchtrager (2006), we assume that **I** (**U**) as a place definer is buried deep down within the OP, as shown in (28a). Thus, its c++commanding domain does not include the rime, and Universal 1 is not violated in forms such as *shai* or *pau*, schematized in (28b).

(28) (a)



If an **I** or **U** in the highest terminal of an OP prevents the OP from merging with a rime containing **I** or **U**, respectively, the reverse, schematized in (29), should also hold: if the highest terminal of an NP is annotated by **I** or **U**, it should be impossible to merge it with an onset containing **I** or **U**, respectively. As the data in (30) and (31) show, the prediction is borne out, giving another explanatory bite to the theory.

(29)



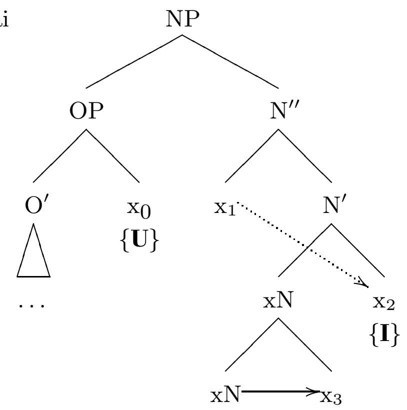
(30) (a) *puan, *puɔ, *puei, *puɨn, *puŋ
(b) *buan, *buɔ, *buei, *buɨn, *buŋ
(c) *muan, *muɔ, *muei, *muɨn, *muŋ
(d) *fuan, *fuɔ, *fuei, *fuɨn, *fuŋ

(31) (a) *shi, *shiɛn, *shiau, *shiɛ, *shiou, *shin, *shiŋ
(b) *zhi, *zhiɛn, *zhiau, *zhiɛ, *zhiou, *zhin, *zhiŋ
(c) *chi, *chiɛn, *chiau, *chiɛ, *chiou, *chin, *chiŋ

4.2. Binding Theory

The form *guai* in (32) demonstrates why we have required in (B2) that c++command be asymmetrical, and why (B5) is needed.

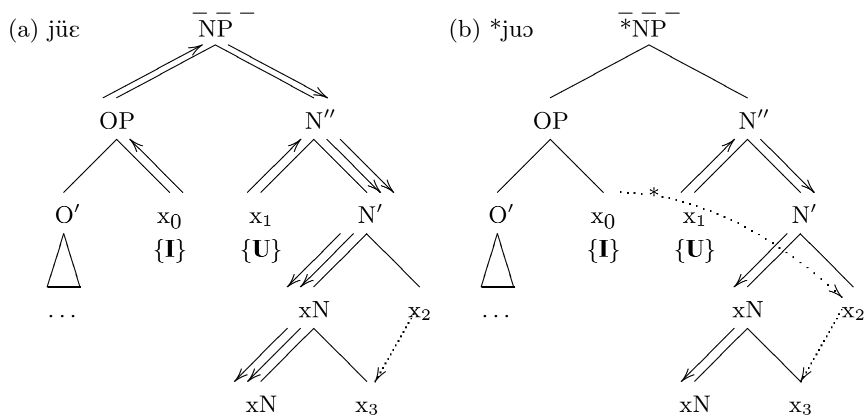
(32) guai



First, x_0 should not bind x_1, since this would result in a violation of Principle x. This is prevented by the requirement of asymmetrical c++command, which is not fulfilled in (32), because both x_0 and x_1 are the highest terminals in their maximal projections (OP and NP, respectively), so their c++command domains contain N″ and OP, respectively. Second, x_0 should not bind x_2, since this would lead to a Principle I violation. This is avoided by stipulating (B5), which states that only the closest potential binder will actually bind. Both x_0 and x_1 potentially bind x_2, but x_1 is contained in the smallest constituent containing both x_0 and x_2. The reverse is not true: the smallest constituent containing both x_1 and x_2 does not contain x_0. Therefore, x_1 but not x_0 binds x_2.

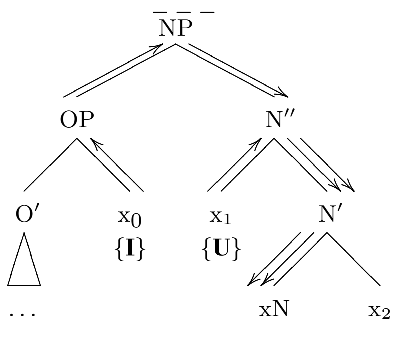
The system makes the correct predictions even in more complicated cases like *jüɛ*. The form is represented in (33a), where both x_0 and x_1 m-command xN. We have already seen why x_1 must m-command: otherwise it would bind x_2, violating Principle x; see (17b). x_0 must m-command for the same reason: otherwise it would bind x_2, as shown in (33b). (Note that there is no island intervening between x_0 and x_2. The island created by m-command from x_1 to xN is the whole tree, since x_1 c++commands into OP as well.)

(33)



While our theory predicts the contrast between (33a) and (33b), it does not entirely explain the situation. The problematic aspect is the interpretation of (33a); specifically, the meaning of m-command. If x_0 m-commands xN, how come that it is also $x_1\{\mathbf{U}\}$ (which cannot even be the target of m-command, according to its definition) that is coloured by $x_0\{\mathbf{I}\}$, i.e. why is (33a) pronounced *jüɛ*, not *juɛ*? A further, and probably not independent issue: if $x_1\{\mathbf{U}\}$ m-commands xN, why doesn't it colour the nucleus with its melody (as $x_0\{\mathbf{I}\}$ does), i.e., why is (33a) not pronounced *jüö*? To add to the puzzle, **U** in (34) does colour the xN, but crucially, the xN in (34) is not part of an adjunction structure.

(34) jü



We leave the issue—essentially the details of the meaning of m-command—to further research. Here we do no more than note that the puzzle is not limited to Putonghua. Turkish (cf. Pöchtrager 2010) exhibits essentially the same phenomenon. (35) summarises which elements spread where with respect to what melody the source (in rows) and target (in columns) position contain. We see that Putonghua and Turkish differ only in one point: in Putonghua, **U** will spread on ə on its own (**duə, duo*), while this will not happen in Turkish (*kuldən* 'from a servant', **kuldon*).[10]

(35)

Putonghua	ɨ	ə
I	**I**	**I**
U	**U**	**U**
I+U	**I** and **U**	**I**

Turkish	ɨ	ə
I	**I**	**I**
U	**U**	
I+U	**I** and **U**	**I**

5. Conclusion

We hope to have shown in this paper that a rich array of phonological properties of Putonghua can be made to follow from a small set of (hopefully) universal/parameterised principles, which are very similar in spirit to syntax. It has been interesting to discover the extent to which the tools employed by phonology and syntax overlap: possible structures are constrained by binding requirements in both syntax and phonology, but movement and deletion are absent from phonology, and the familiar c-command relation is not directly applicable to phonology, but must be amended into c++command.

As for the analysis of Putonghua, we are certainly still far from mining all the insights into the phonological properties of this language. We have deliberately simplified things by setting aside tonal phenomena, and rimes which contain the nasal consonants. Beyond Putonghua there is much to be done in testing the various principles we have found. Even within the wider group of Han languages there are superficial counterexamples to the 'no self c++command' constraint, among others. It is hoped that the highly constrained nature of the theory we have proposed will actually **force** us to an analysis of these forms consistent with the general principles.

What we have, then, is a tightly constrained theory which we claim to model the computational system of **all** phonologies. This severely restricts the numbers and kinds of hypotheses we can propose when trying to understand the phonological phenomena of any particular language. The

[10] However, there are Turkic languages which do behave like Putonghua (Charette–Göksel 1994; 1996).

closer we can get to there being exactly one hypothesis consistent with the principles of Phonology in any given case, the better. We submit that the approach taken in this paper represents a significant step in that direction.

References

Charette, Monik – Asli Göksel 1994. Vowel harmony and switching in Turkic languages. In: SOAS Working Papers in Linguistics & Phonetics 4 : 29–56.

Charette, Monik – Asli Göksel 1996. Licensing constraints and vowel harmony in Turkic languages. In: SOAS Working Papers in Linguistics & Phonetics 6 : 1–25.

Ferme, Alja 2002. Kitajski fonološki sistem v okviru tvorbene fonologije [The Chinese phonological system in generative phonology]. B.A. thesis, University of Ljubljana.

Ferme, Alja 2009. The palatal element hides in (some) non-palatals: The case of Mandarin Chinese. In: Sylvia Blaho – Camelia Constantinescu – Bert Le Bruyn (eds): Proceedings of ConSOLE XVI, 69–76. Student Organisation of Linguistics in Europe, Leiden.

Ferme, Alja – Sašo Živanović 2006. Palatalno sozvočje v kitajščini [Palatal harmony in Putonghua]. In: Azijske in afriške študije 10 : 1–13.

Goh, Yeng-Seng 1996. The segmental phonology of Beijing Mandarin. University of London, London.

Kaye, Jonathan D. 2001. A fresh look at Putonghua onset–rime pairs. Ms. (http://www.unice.fr/dsl/nis01/ORputon.pdf)

Kaye, Jonathan D. – Markus Alexander Pöchtrager 2009. GP 2.0. Paper presented at the GPRT6, Piliscsaba, Hungary, April 25, 2009.

Neubarth, Friedrich – John R. Rennison 2002. Mandarin Chinese phonemics and the X-bar theory of GP. Ms. (http://www.univie.ac.at/linguistics/gp/papers/neubarth_rennison_mandarin.pdf)

Pöchtrager, Markus Alexander 2006. The structure of length. Doctoral dissertation, Universität Wien.

Pöchtrager, Markus Alexander 2010. Does Turkish diss harmony? In: Acta Linguistica Hungarica 57 : 458–73.