Contributed to *DougSchrift: A Collection of Squibs and Puzzles presented to Doug Pulleyblank*, 2025. https://arts-pulleyblank-2024.sites.olt.ubc.ca/

Keeping it (Output) Strictly Local:
A Finite State Model of Phonological Patterns in Kera*

Alexandra Lilia Ayala, Kaili Vesik and Zara Khalaji Pirbaluti
University of British Columbia

1 Preface

"It's simple, but it's not easy."

- Douglas Pulleyblank

The impetus for this paper was the authors' desire to express their deep gratitude towards Douglas Pulleyblank and his exemplary qualities as a teacher – one of which has been to encourage persistence, exploration and curiosity in his students about phonology outside their comfort zone. In that spirit, this paper aims to demonstrate that output strictly local transformations are capable of modelling a dataset containing vowel harmony and other phonological phenomena in Kera. This dataset (representative of some but not all the patterns of the language) comes from Odden (2005)'s textbook, and was chosen due to the authors' experiences TA-ing Doug's undergraduate phonology course. The analytical approach, meanwhile, was inspired by his ongoing reference to Tier-based Strictly Local functions ("But can we do it using TSL?!") during a graduate directed reading course on vowel harmony.

2 Introduction to Kera

Kera (ISO639-3:ker) is a language within the eastern subclass of the Chadic language branch in the Afro-Asiatic language family. It is spoken by 50,0000 people in Southwestern Chad, south of the town of Fianga in major towns and also parts of Cameroon (see Figure 1 from Pearce, 2007b). Other closely-related languages in the East Chadic classification include Kwang. Previous literature on Kera includes work on phonetic descriptions (Pearce, 2011), phonology (Pearce, 1999; 2006a; b; c; 2007a; b; 2008; 2009), grammar, lexicon, collection of texts and anthropological data by Ebert (1975, 1976, 1979, 2003), and syntax (Camburn, 1984). The following descriptions and facts about the language are taken from Pearce (2011; 2007b).

*We want to express our gratitude to Anne-Michelle Tessier, who edited this paper and without whom this very long squib would be a very long mess, and to Gunnar Ólafur Hansson, who sneakily passed along the lesson plans that kicked off this project. All mistakes are our own.

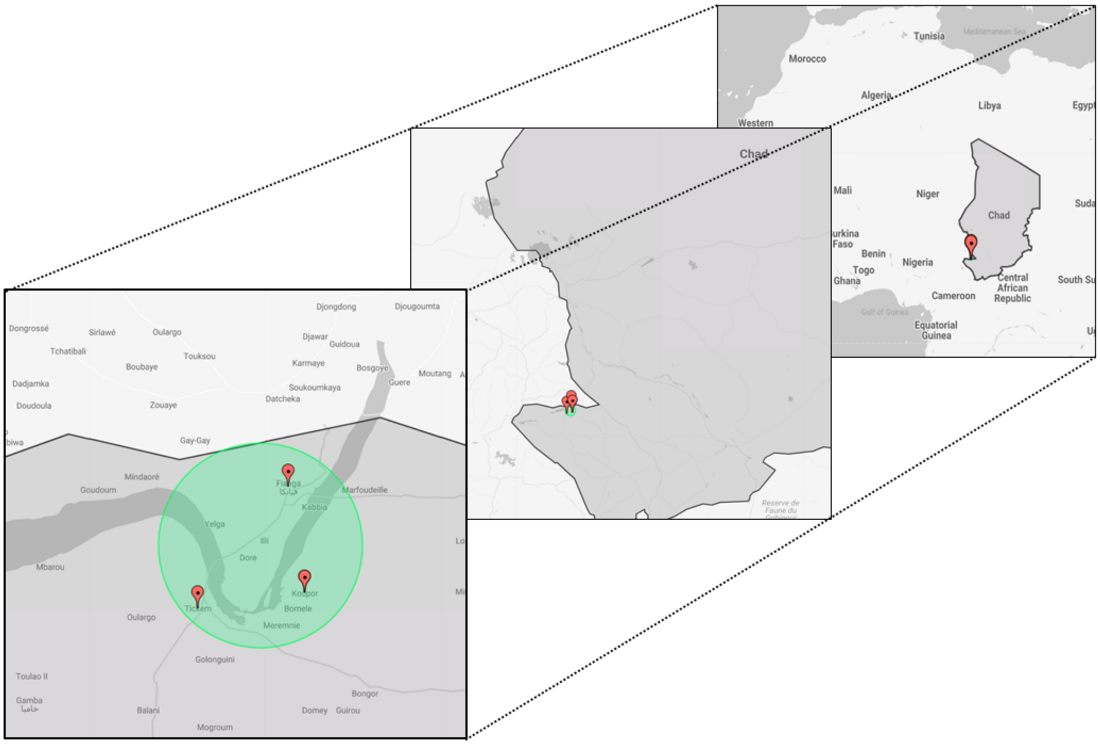

Figure 1. Areal map of the villages (Fianga, Tickem, Koupor) where Kera is spoken in Chad (Pearce, 2007b).

2.1 Consonants

Kera's consonant inventory includes 24 consonants, shown in the IPA chart below (Table 1). The obstruents are contrastive in voice. It is also worth mentioning that voicing interacts with tone, and shows socio-linguistic and dialectal variation. Moreover, the labiodental flap /ⱱ/ is one of the unique consonants only found in 80 languages of the world, which is said to have entered the Kera inventory via contact with surrounding languages.

Table 1. Consonant inventory of Kera

	Labial		Alveolar		Palatal		Velar		Glottal
Plosive/Affricate	p	b	t	d	tʃ	dʒ	k	g	ʔ
Fricative	f	v	s	z					h
Implosive		ɓ		ɗ					

Nasal		m		n				ŋ	
Flap		ⱱ		ɾ					
Lateral Approximant				l					
Approximant		w				j			

2.2 Vowels

The core vowel inventory contains 6 vowels in Kera /i ɨ u ɛ ɔ a/, which are specified in the table below with four standard binary features of [±high], [+/-low], [+/-front] and [±back]. Vowel length is contrastive. (The full Kera vowel inventory includes contrastively nasalized vowels, and ATR allophonic pairs for the non-high vowels, but these will not be relevant to the analyses here and so will not be discussed further.) We will use the term 'central' to refer to vowels which are specified [-front, -back] for ease of reference.

Table 2. Vowel inventory of Kera

	+front, -back	-front, -back	-front, +back
+high, -low	i iː	ɨ ɨː	u uː
-high, -low	ɛ ɛː		ɔ
-high, low		a	

2.3 Syllables and Tones

The attested syllable structures are (C)V, (C)Vː, (C)VC. When light syllables are combined, the resulting surface forms are either [CVCVː] (phrase-finally) or [CVC] (phrase-medially); surface strings of CVCV are uncommon (though found in shorter function words, and at the end of certain nouns). Word-initial onsets are preferred, as a non-contrastive glottal stop typically appears at the beginning of vowel initial words. Word medially, almost all consonants can be in coda position, while word-final codas must be sonorants.

Kera is a tonal language, like many other Chadic languages. The three level tones are L, H, M. Tone is mostly lexical, but grammatical tone also exists in the language.

3 Analysis-Neutral Account

3.1 Dataset

The dataset presented here is from Odden (2005: 213-214).

Inalienable nouns and possessive pronoun suffixes:

(1)	se:nen	‘my brother'
	se:nem	‘your (masc) brother'
	si:ni	‘your (fem) brother'
	si:nu	‘his (masc) brother'
	se:na	‘her (fem) brother'
	se:neŋ	‘your (pl) eat'
(2)	cɨ:rɨn	‘my head'
	cɨ:rɨm	‘your (masc) head'
	ci:rɨ	‘your (fem) head'
	cu:ru	‘his (masc) head'
	cɨ:rɨ	‘her (fem) head'
	cɨ:rɨŋ	‘your (pl) head'
(3)	gi:din	‘my belly'
	gi:dim	‘your (masc) belly'
	gi:di	‘your (fem) belly'
	gi:du	‘his (masc) belly'
	gi:dɨ	‘her (fem) belly'
	gi:diŋ	‘your (pl) belly'

Present verbs and direct object suffixes:

(4)	haman	‘eat me'
	hamam	‘eat you (masc)'
	hɨmi	‘eat you (fem)'
	hɨmu	‘eat him'
	hama	‘eat her'
	hamaŋ	‘eat you (pl)'
	kolon	‘change me'
(8)	gunun	‘wake me'
	gunum	‘wake you (masc)'
	guni	‘wake you (fem)'
	gunu	‘wake him'
	gunɨ	‘wake her'
	gunuŋ	‘wake you (pl)'
(9)	ʔasan	‘know me'

(5)

kolom	‘change you (masc)'
kuli	‘change you (fem)'
kulu	‘change him'
kola	‘change her'
koloŋ	‘change you (pl)'

(6)

haran	‘give me back'
haram	‘give you (masc) back'
hɨri	‘give you (fem) back'
hɨru	‘give him back'
hara	‘give her back'
haraŋ	‘give you (pl) back'

(7)

ɲɨfan	‘meet me'
ɲɨfam	‘meet you (masc)'
ɲɨfi	‘meet you (fem)'
ɲɨfu	‘meet him'
ɲɨfa	‘meet her'
ɲɨfaŋ	‘meet you (pl)'

ʔasam	‘know you (masc)'
ʔɨsi	‘know you (fem)'
ʔɨsu	‘know him'
ʔasa	‘know her'
ʔasaŋ	‘know you (pl)'

(10)

ʔapan	‘find me'
ʔapam	‘find you (masc)'
ʔɨpi	‘find you (fem)'
ʔɨpu	‘find him'
ʔapa	‘find her'
ʔapaŋ	‘find you (pl)'

(11)

bɨlan	‘want me'
bɨlam	‘want you (masc)'
bɨli	‘want you (fem)'
bɨlu	‘want him'
bɨla	‘want her'
bɨlaŋ	‘want you (pl)'

Past verbs and direct object suffixes:

(12)

ɲafnan	‘met me'
ɲafnam	‘met you (masc)'
ɲɨfni	‘met you (fem)'
ɲɨfnu	‘met him'
ɲafna	‘met her'
ɲafnaŋ	‘met you (pl)'
ɲafla	‘you must meet!

(13)

balnan	‘wanted me'
balnam	‘wanted you (masc)'
bɨlni	‘wanted you (fem)'
bɨlnu	‘wanted him'
balna	‘wanted her'
balnaŋ	'wanted you (pl)'
balla	‘you must want!’

Other words:

ba	'not'	bɨpa	'no more'
pa	'again'		

3.2 Generalization and Observations

3.2.1 Segmentation of morphemes

Starting with the data above, the table below lists the root morphemes that we have segmented from each word. All the roots are monosyllabic with the syllable shape CV(ː)C. Most roots can have one or two forms (14-15, 17-26), while one root has a third alternate form (16). Some roots with multiple surface alternants have high vowels that alternate with mid vowels, matching for front/backness {o~u, eː~iː} (17-18). Another set of roots' vowels alternate between a low and high central vowel {a, ɨ} (19-23). (This same pattern seems to be found with the root [ba] 'not' and its alternate form [bɨ] (cf. 24, 26), although this morphological relationship is not entirely clear, so we list both forms as potential roots.) Some roots containing central vowels begin with glottal consonants {ʔ, h} (21-23), which Odden (2005) tells us are specified as [+low].

(14)	gun	'wake'	(21)	ŋaf, ŋɨf	'meet'
(15)	giːd	'belly'	(21)	har, hɨr	'give'
(16)	cɨːr, ciːr, cuːr	'head'	(22)	ʔap, ʔɨp	'find'
(17)	kol, kul	'change'	(23)	ʔas, ʔɨs	'know'
(18)	seːn, siːn	'brother'	(24)	ba	'not'
(19)	ham, hɨm	'eat'	(25)	pa	'again'
(20)	bal, bɨl	'want'	(26)	bɨpa	'no more

There are 2 aspectual suffixes and 6 person/gender/number (henceforth pgn.) suffixes in the Odden (2005) dataset. The orders of exponents observed are root+pgn and root+asp+pgn.

Table 3. Surface representations of each suffix.

perfective	-n
imperative	-l

	Singular	Plural
1	-an, -en, -on, -ɨn, -un, -in	---
2.masc	-am, -em, -om, -ɨm, -um, -im	-aŋ, -eŋ, -oŋ, -ɨŋ, -uŋ, -iŋ
2.fem	-i	
3.masc	-u	---
3.fem	-a, -ɨ	---

The 1.sg, 2.masc.sg, and 2.pl all have a vowel followed by a nasal consonant. The vowel in these suffixes always matches the root vowel, except in the present tense forms of ‘want’ and ‘need’ (e.g., respectively, [bɨlan], [bɨlam], [bɨlaŋ]; [ŋɨfan], [ŋɨfam], [ŋɨfaŋ]). For now, we segment this vowel with the suffix because the vowel appears adjacent to the nasal even when one of the aspectual suffixes intervenes between the root and the object marker (e.g., compare [bɨlan] ‘want me’ and [balnan] ‘wanted me’). Moving on, the 2.fem.sg is invariably [-i] and 3.masc.sg is invariably [-u], while the 3.fem.sg surfaces as both [-a] and [-ɨ].

3.2.2 Generalizations about alternations in roots and suffixes

In the perfective, the vowels in the non-glottal-initial CV(:)C roots can surface as both [a] and [ɨ] (27d, 28d). In the present tense, these roots only appear with their high variant [ɨ] (27a-27c, 28a-28c), while the glottal-initial roots exhibit both [a] and [ɨ] vowels (29-30). In the 1sg and 3sg.masc forms of non-glottal-initial roots (27a, 27c, 28a, 28c), only the [ɨ] alternants surface (e.g., b**ɨ**lan, b**ɨ**lu), whereas the glottal-initial roots in (29a, 29c, 30a, 30b) surface with [a] in the 1sg and [ɨ] in the 3sg.masc (e.g., h**a**man, h**ɨ**mu). It is important to note that in the perfective forms (27d, 28d), there are two consonants separating the root and suffix vowels (aCCa) rather than one (ɨCa).

(27a) bɨlan ‘want me’

(27b) bɨli ‘want you (f.)’

(27c) bɨlu ‘want him’

(27d) balnan ‘wanted me’

(28a) ŋɨfan ‘meet me’

(28b) ŋɨfi ‘meet you (f.)’

(28c) ŋɨfu ‘meet him’

(28d) ŋafnan ‘met me’

(28e) ŋɨfnu ‘met him’

(29a) haman ‘eat me’

(29b) hɨmi ‘eat you (f.)’

(29c) hɨmu ‘eat him’

(30a) ʔasan ‘know me’

(30b) ʔɨsu ‘know him’

With respect to roots + suffixes, the [ɨ] variant of glottal-initial roots surfaces when the root is suffixed by either the 2.fem [-i] (31b, 32b, 33b) or 3.masc [-u] (31c, 32c, 33c, 34b) (e.g., bɨli, bɨlu). For roots that alternate mid and high vowels, the high variants surface before these same suffixes (33-34). Only root variants that have a high vowel (31-32) can precede the high-vowel suffixes but roots that are mid (33-34) and low (27b-c, 28b-c, 28e, 29b-c, 30b) do not co-occur with these suffixes.

(31a)	gunun	'wake me'	(33a)	kolon	'change me'
(31b)	gunu	'wake him'	(33b)	kulu	'change him'
(32a)	giːdin	'my belly	(34a)	seːnen	'my brother'
(32b)	giːdu	'his belly	(34b)	siːnu	'his brother'

The root for 'head', with its three [ɨː], [iː] and [uː] allomorphs, surfaces with [iː] before the 2.fem.sg suffix [-i] (35b), with its [uː] variant before the 3.masc.sg suffix [-u] (35c), and with [ɨ] elsewhere in the paradigm (35a).

(35a)	cɨːrɨn	'my head'
(35b)	ciːri	'your (fem) head'
(35c)	cuːru	'his head

Finally, roots are not the only locus of alternation – the 3.fem.sg suffix can also surface as either [-a] or [-ɨ]. The [-ɨ] variant is found after roots with high vowels (36-38), while the [-a] variant occurs after roots with both high and non-high vowels (39-41). This pattern includes the non-glottal-initial roots, whose vowels alternate between [ɨ] and [a] but always take the suffix [-a] in the 3.fem.sg (42a-b).

(36)	cɨːrɨ	'her head'	(40)	seːna	'her brother'
(37)	giːdɨ	'her belly'	(41)	hama	'eat her'
(38)	gunɨ	'wake her'	(42a)	bɨla	'want her'
(39)	kola	'change her'	(42b)	balna	'wanted her

3.2.3 Summary of phonotactic restrictions and exceptions

The following phonotactic restrictions hold of the entire data set:

- ***Complex onsets and codas:** No word-initial or word-final consonant sequences are found, and word-internal sequences have maximally two consonants
- ***CaCa:** No sequences of two low vowels are found separated by a single consonant, *unless* the first [a] is preceded by a glottal (ʔaCa, haCa). Note that two intervening consonants makes the string grammatical (aCCa).
- ***[-hi]..[+hi]:** Non-high vowels are not observed before high vowels
- ***H[-low]:** Non-low vowels are not observed after glottals (i.e., only [a] is found after glottals).

In addition, there are at least two phonotactic restrictions which hold of a subset of roots:

- **[+hi]..[-hi]/central root:** Non-high vowels are not observed after high vowels unless the root exhibits the central vowel [ɨ]~[a] alternation and does not start with a glottal.
- ***[ɨ]...{i, u}/central root:** The high central vowel [ɨ] is not observed before [i] or [u] unless the root exhibits the central vowel [ɨ]~[a] alternation.

3.2.4 Summary of root types

Table 4 summarizes the root types and their phonotactic behaviour. Along the top of the table are the observed phonotactic restrictions just established. If a root type has two alternants, one of which violates a restriction and the root surfaces with the *other* alternant, its cell in the column is checked with '✔' to indicate that the restriction has been respected. If a root with two alternants surfaces with the *violating* alternant, the cell is marked with an '✗', to indicate that the restriction has not been respected. If a root type does not have any alternant that violates a restriction, it is marked with a long dash '—'.

Table 4. Summary of the phonotactic restrictions and root allomorphy

Type	Root Type	Alternants	*CaCa	*-hi..+hi	*H[-lo]	*+hi..-hi	*ɨ..{i, u}
I	invariant roots (Ci:C & CuC)	(N/A)	—	—	—	—	—
II	[+hi] roots with three alternants	Cɨ:C ~Ci:C ~Cu:C	—	—	—	✔	✔
III	a) central roots, with [+/-hi] alternants	Ca(C) ~Cɨ(C)	✔	✔	—	✗	✗
	b) glottal-initial central roots	HaC~HɨC	—	✔	✔	✔	✗
IV	[-lo] roots with [+/-hi] alternants	CoC~CuC	—	✔	—	✔	—
		Ce:C~CiC	—	✔	—	✔	—

For example, the green-shaded cell's checkmark indicates that Type IV roots respect the ban on [+hi]..[-hi] sequences, meaning that the [-hi] root alternant surfaces before the low suffix [-a]:, e.g. [k**o**la], *[k**u**la]. On the other hand, the red-shaded cell's X indicates that (Type IIIa) roots do violate [+hi]..[-hi] ban, e.g. [b**ɨ**la]. (As will be discussed below: choosing the +lo root alternant would have violated a different ban, *CaCa, e.g. *[b**a**la].)

4 Output Strictly Local Analytical Approach

4.1 Phonological Analysis and Assumptions

4.1.1 Underlying representations

We begin by establishing URs for each type of root found in Table 4 above. Uncontroversially, we propose that the URs for invariable roots (Type I) match their surface forms. For Type III and IV roots, we propose that their URs have a [-hi] vowel, since their [+hi] alternants are fully predictable. For Type II roots, we propose that their UR vowel is underlyingly central, e.g. the root for 'head' is underlyingly /cɨ:r/. With these assumptions, we boil down the root types to the following revised set in Table 5, with examples of each:

Table 5. Revised set of root types, with sample URs

Root Types	UR Type	Sample UR	Surface alternants	Gloss
I: [+hi], non-central	/CiːC/	/giːd/	giːd	'belly'
	/CuC/	/gun/	gun	'wake'
II: [+hi], central	/CɨːC/	/cɨːr/	cɨːr, ciːr, cuːr	'head'
IIIa: [+lo], non-glottal initial	/Ca(C)/	/ŋaf/	ŋaf, ŋɨf	'meet'
IIIb: [+lo], glottal initial	/HaC/	/ham/	ham, hɨm	'eat'
IV: [-hi, -lo]	/CoC/	/kol/	kol, kul	'change'
	/CeːC/	/seːn/	seːn, siːn	'brother'

Now turning to the suffixes, we again propose that the URs for the invariable suffixes match their surface representations, i.e. /-i/ and /-u/. Next, we propose that the nasal-final suffixes are singleton consonants in the URs, and that the vowels are epenthetic; recall from section. 3.2.4 that these vowels either match the root vowel, or else surface as [-hi], and that epenthesis is motivated since the language does not allow complex codas. Finally, as with the roots, we propose that the UR for the 3.fem suffix is /-a/, since its high counterpart [ɨ] only appears following high vowels. This means that [-a] suffix alternant found after Type IIIa roots like [ŋɨfa] is triggered by this *root's* underlying low vowel, since underlying /ɨ/ would condition the high vowel suffix. Our hypotheses for the inflectional suffixes are summarized in Tables (6-b) below:

Table 6. URs for inflectional suffixes (for morphological information, see Pearce 2007b)

a) Tense and aspect suffixes

perfective	/-n/
imperative	/-l/

(b) Object and possessive markers

	singular	Plural
1	/-n/	---
2.masc	/-m/	/-ŋ/
2.fem	/-i/	
3.masc	/-u/	---
3.fem	/-a/	---

4.1.2 Derivational rules

As foreshadowed, our analysis will involve stepwise derivations carried out through Output Strictly Local (OSL) functions and mapped by Finite State Transducers (FSTs; Oncina et al., 1993). Any over- or under-application of the rules which realize the restrictions in Table 4 will therefore be attributable to ordering effects, just as with classic SPE-type rules in feeding or bleeding orders. Given the shared derivational nature of the two frameworks, we present in this section the following set of six re-write rules and their crucial orderings. These will serve as the basis for our OSL transducers, though ultimately our analysis will not translate each of these rules precisely or directly.

R1: Vowel Copy $\emptyset \rightarrow V_i$ / V_i C _ C $]_{syll}$

R2: High V Agree ɨ → [αfront, βback] / _ C [αfront, βback, +high]

R3: Post Raising V → [+high] / [V, +high] C_0 ___

R4: Low Dissimilation a → ɨ / ___ C a

R5: Guttural Lowering ɨ → [+lo] / [+lo] _

R6: Pre Raising V → [+high] / ___ C_0 [V,+high]

Vowel Copy feeds Low Dissimilation, since copying /a/ creates the illicit sequence *[Ca..a] (43).

(43a) /bal-n/ → VC: [balan] → LD: [bɨlan] VC >> LD

(43b) /bal-n/ → LD: N/A → VC: *balan *LD >> VC

Low Dissimilation precedes Guttural Lowering, or else [ʔa..a] would not surface (44).

(44a) [ʔasan] → LD: [ʔɨsan] → GL: [ʔasan] LD >> GL

(44b) [ʔasan] → GL: N/A → LD: *ʔɨsan *GL >> LD

Guttural Lowering precedes Pre-Raising in a counterfeeding order, creating [Hɨ] sequences (45).

(45a) /ham-u/ → GL: N/A → PreR: [hɨmu] GL >> PreR

(45b) /ham-u/ → PreR: [hɨmu] → GL: *hamu *PreR >> GL

Post-Raising precedes Low Dissimilation in a counterfeeding order, resulting in [ɨ..a] sequences (46).

(46a) /bal-a/ → PR: N/A → LD: [bɨla] PostR >> LD

(46b) /bal-a/ → LD: [bɨla] → PR: *bɨlɨ *LD >> PostR

High V Agree precedes Pre-Raising in a counterfeeding order, which results in disharmonic [ɨ..i] and [ɨ..u] sequences (47).

(47a) /bal-i/ → HVA: N/A → PreR: [bɨli] HVA >> PreR

(47b) /bal-i/ → PreR: [bɨli] → HVA: *bili *PreR >> HVA

High V Agree precedes Post-Raising in a counterfeeding order, which results in disharmonic [u..ɨ] and [i..ɨ] sequences (48).

(48a) /gun-a/ → HVA: N/A → PostR: [gunɨ] HVA >> PostR

(48b) /gun-a/ → PostR: [gunɨ] → HVA: *guni *PostR >> HVA

Final crucial order: {VC, HVA} >> PostR >> LD >> GL >> PreR

4.2 Finite State Representations of the OSL Grammar

A Finite State Transducer (FST) is defined as a six-tuple $\tau = (Q, \Sigma, \Gamma, q_0, F, E)$ (Oncina et al., 1993; Chandlee, 2014), where:

Q: A finite set of states.

Σ: An input alphabet; a finite set of input symbols.

Γ: An output alphabet; a finite set of output symbols.

q_0: An initial state, where $q_0 \in Q$.

F: A finite set of final states, where $F \subseteq Q$.

E: A finite set of *edges*, where $E \subseteq (Q \times \Sigma \times \Gamma \times Q)$, which is a relation between all states Q and the alphabet of input symbols Σ, and the alphabet of output symbols Γ and all states Q. The edges are associated with a transition function δ: $\forall(q,a,o,q') \in E$, where q is a current state, a is an input symbol, o is an output string, and q' is the next state.

Γ^* is the set of all possible output strings and Σ^* is the set of all possible input strings. Since this analysis consists of multiple transducers which feed forward into the next (see section 5), not all transducers receive the same input strings and they do not produce the same output strings, which is to say that every transducer has a different Γ^* and Σ^*, and so these will be individually defined by the transducer in section 5.

In terms of memory, the FSTs defined here can only process input windows of k=1. The input alphabet Σ is derived from all the symbols in the Odden (2005) dataset. The output alphabet Γ is identical to the input alphabet, except that it also includes *v*, which is a vowel placeholder symbol. The empty string λ belongs to both alphabets, and the long vowel diacritic ː is a symbol on its own.

Σ: {a, e, o, i, ɨ, u, h, ʔ, r, l, m, n, ŋ, ɲ, p, c, k, b, d, ɡ, f, s, ː, λ}

Γ: {a, e, o, i, ɨ, u, h, ʔ, r, l, m, n, ŋ, ɲ, p, c, k, b, d, ɡ, f, s, ː, λ, *v*}

We further use the following symbols to define subsets of the input and output alphabets; these subsets are identically defined in the input and output alphabets, which will be relevant to the identity map transducer we assume in section 4.3.1:

C $\subseteq \Sigma$: {h, ʔ, r, l, m, n, ŋ, ɲ, p, c, k, b, d, ɡ, f, s}

V $\subseteq \Sigma$: {a, e, o, i, ɨ, u}

$V_{+hi} \subseteq$ V: {i, ɨ, u}

$V_{-hi} \subseteq$ V: {a, e, o}

PerHiV ("Peripheral High Vowels") $\subseteq V_{+hi}$: {i, u}

¬PerHiV (= V - PerHiV): {ɨ, a, e, o}

[+lo] ⊆ *Σ*: {h, ʔ, a}

[-lo] ⊆ *Σ*: {r, l, m, n, ŋ, ɲ, p, c, k, b, d, g, f, s, e, o, i, ɨ, u}

4.3 Hidden Functions

4.3.1 The Identity Map Transducer

We assume there is a dedicated transducer that functions as an identity map (Chandlee & Jardine, 2024). In this transducer, every state recognizes every symbol in the alphabet *Σ*, which will then trigger a transition to that symbol's state and will output that same symbol. For example, if the transducer receives <a> in its input, regardless of whatever its current state is, it will transition to the A state, and it will output <a>.

Table 7. State-transition table of the identity function transducer.

Next State / Current State	A	E	O	…
A	a:a	e:e	o:o	…
E	a:a	e:e	o:o	…
O	a:a	e:e	o:o	…
…	a:a	e:e	o:o	…

For representational simplicity, we use $\Sigma_{x\in X}:\Gamma_{x\in X}$ mappings, where *x* is the same member of identical subsets *X* of the input and output alphabets, to indicate where the Identity map has been invoked. The transitions we use to invoke the identity map are as follows:

{C:C, V:V, V_{+hi}:V_{+hi}, V_{+lo}:V_{+lo}, [+lo]:[+lo], ¬PerHiV:¬PerHiV, ¬a:¬a, ¬ɨ:¬ɨ}

Although input symbols are not individually represented, the identity map function ensures that whatever element of the input set is read, its identical symbol from the output set will be written. In other words: ¬a:¬a means that if a transducer reads a <p>, it will not output <ʔ>, it will output <p>.

4.3.2 The Raising Function Transducer

In addition to the Identity Map transducer, we also assume a transducer that recognizes the proper subset V of the alphabet, and outputs its [+high] counterpart. Again, for representational simplicity, we use indices (e.g., $V{:}V_{[+hi]}$) to indicate where the raising function has been applied.

Table 8. State-transition table of the raising function transducer (vowel length irrelevant)

Next State / Current State	Ɨ	I	U	…
A	a:ɨ	e:i	o:u	…
E	a:ɨ	e:i	o:u	…
O	a:ɨ	e:i	o:u	…
…	a:ɨ	e:i	o:u	…

5 Visualised Rules

In this section, the rules proposed in section 4.1 are translated into FSTs. These transducers will be applied in the same crucial order as the rules with two important differences: (i) strings are reversed before they are input into a transducer (Heinz & Lai 2013; Chandlee 2014; Burness 2022), and (ii) transducers can only read and write from left to right (i.e., they belong to the class of OSL Left Subsequential functions). These processes are illustrated below in Figure 2.

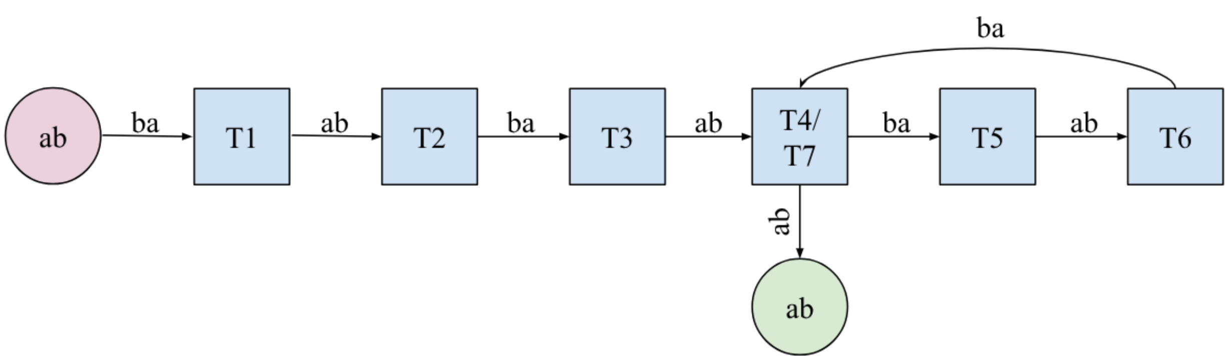

Figure 2. Example of a string <ab> running through each transducer in order. The red circle indicates the precedence relations between the segments of the initial input string and the green circle indicates the precedence relations of the final output. Each arrow indicates that the string has been reversed. String reversal occurs before entering a transducer (blue squares), as well as after exiting T7 as the final output.

For instance, given a string <ab>, that string will be reversed as <ba> before it enters the first transducer, T1. This reversed string is then processed from left to right (i.e., in this order of precedence: 0=b, 1=a) by the transducer. The output of T1 is then fed into the next transducer, but first, it is once again reversed, this time as <ab>, which swaps the order of precedence (i.e., 0=a, 1=b). This reversal continues before every transducer, always being read left-to-right, until it is output by the final transducer and reversed one last time to restore the original order of precedence of the input.

Additionally, T4/T7 (the Raising transducer), is run twice, but it does not see the same precedence relations each time. T4 receives the reversal <ab> after T3, and it also receives the reversal <ba> after T6. By running the same string in flipped orders of precedence, our analysis preserves the crucial ordering from section 4.1 without requiring two different machines to carry out the same process in two directions. Our use of a single machine also aligns with Pearce (2007b)'s analysis of raising harmony as bidirectional (p. 94).

5.1 Transducer 1: Vowel Epenthesis

5.1.1 Definition of transducer

Q: {λ, Coda, Nucleus, Epenthesis}

q_0: {λ}

F: {λ}

Σ^*: {nmah, mmah, …, ŋnfaŋ, alfaŋ}

Γ^*: {n*v*mah, m*v*mah, …, ŋ*v*nfaŋ, alfaŋ}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.1.2 for a visual representation of this transducer's transitions.

5.1.2 Description

The underlying form of a word is first reversed before it is fed into the first transducer (e.g., <baln> becomes <nlab>), and the FST proceeds to read the newly reversed string from left to right. In consonant-final words, it will read and write the first consonant faithfully, but then if it sees another consonant, it will output that consonant and a placeholder vowel *v* before it. In vowel-final words and in word-medial contexts, the transducer will output the placeholder vowel after the third consonant it sees in a row.

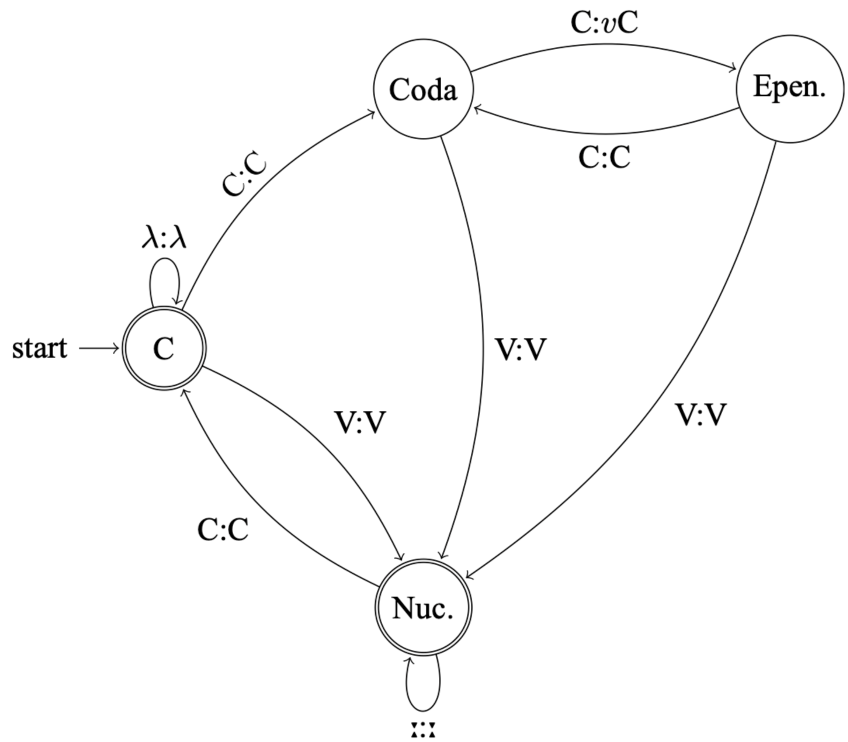

Figure 3. Vowel Epenthesis

5.1.3 Predictions

The main prediction is that the epenthetic vowel in word-final clusters should surface before the last consonant, which correctly captures all the data in the Odden (2005) dataset. However, this transducer also predicts that the epenthetic vowel will surface between the first and second consonants in a word-medial three-consonant sequence, whereas Pearce (2007b) reports vowel insertion after the *second* consonant VC*v*CCVC, shown in (49):

(49a) /mirk-t-u/	[mirkutuː]	‘greeted him (habitual)’	rev: <ːutukrim>[1]
(49b) /gɔld-t-ɛ/	[gɔldetɛː]	‘search (habitual)’	rev: <ːɛtedlɔg>
(49c) /lugd-t-i/	[lugditi]	‘curse you (f) (habitual)’	rev: <itidgul>

[1] For ease of comparison, the empirical data is given in its original order of precedence. For transducers that have reversed the string to the flipped order of precedence from the original input, the reversed versions of surface forms are given.

If there is a four-consonant sequence, this transducer predicts two epenthetic vowels placed before the second and fourth consonants in left-edge (word-final) contexts like C*v*CC*v*CVC, and only one epenthetic vowel in word-medial contexts before the third consonant in the sequence VCC*v*CCVC. In the former context, epenthesis in Kera actually occurs before the second and third consonants C*v*C*v*CCVC (50a), which is not what we predict. However, in the latter context, epenthesis does occur before the third consonant in the sequence, as it does here (50b) (data from Pearce 2007b: 94).

(50a) /mirk-t-n/	[mirkitin]	'greets me repeatedly'	rev: <ni̲ti̲krim>
(50b) /mirk-t-n-nu/	[mirkutnuː]	'greeted him repeatedly'	rev: <ːuntu̲krim>

Finally, this transducer allows illegal consonant clusters at the right edge (word-initially). Although there are no contexts for word-initial epenthesis in the Odden (2005) dataset, such words are found in Pearce's (2007b) dissertation as in (51). In this context, epenthetic vowels surface in between the first and second consonants, corresponding to the rightmost two consonants in the reversed string:

(51a) /k-mirwi/	[kimirwi]	'new m.'	rev: <iwrimi̲k>
(51b) /k-pirkɨ-w/	[kɨpɨrkɨw]	'mountains'	rev: <wɨkrɨpɨ̲k>

5.2 Transducer 2: Vowel Copy and Overwrite

5.2.1 Definition of transducer

Q: {Overwrite *v*, e, a, i, ɨ, u o}

q_0: {Overwrite *v*}

F: {Overwrite *v*}

*Σ**: {ham*vn*, ham*v*m, …, ŋafn*v*ŋ, ŋafla}

*Γ**: {haman, hamam, …, ŋafnaŋ, ŋafla}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.2.2 for a visual representation of this transducer's transitions.

5.2.2 Description

The output string of the first transducer is once again reversed before it enters the second transducer, which reads it from left to right (e.g., <n*v*lab> becomes <bal*v*n>). It will read and write any segment faithfully, but if it encounters the placeholder vowel *v*, then it will overwrite that vowel with the last vowel it saw.

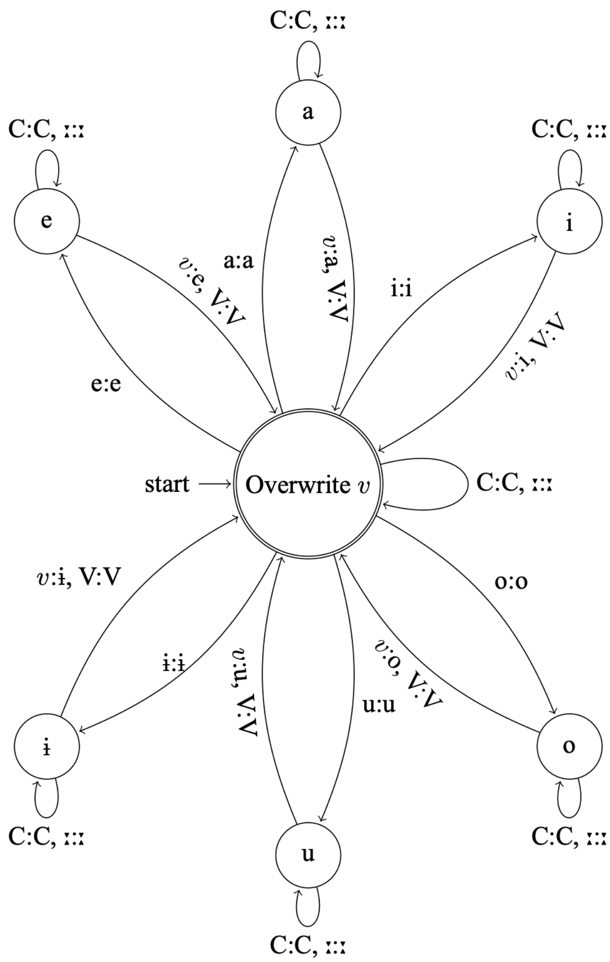

Figure 4. Vowel Copy and Overwrite

5.2.3 Predictions

This transducer overwrites the vowel placeholder with the most recent vowel it has seen. It also predicts that the overwritten vowel will always be short, and that its source will always be to the left of itself. However, while Pearce (2007b) confirms that the epenthetic vowel never lengthens (p. 97), she also shows that when the root and suffix have different vowels, the preference for what she calls "feature filling" comes from the suffix (52), and not the vowel (p. 95).

(52) /mirk-t-u/ [mirkutu] 'greets him repeatedly'

5.3 Transducer 3: High Vowel Agreement

5.3.1 Definition of transducer

Q: {λ, i, u, ɨ}

q_0: {λ}

F: {λ, i, u, ɨ}

*Σ**: {namah, mamah, …, ir:ɨc, …, ŋanfaŋ, alfaŋ}

*Γ**: {namah, mamah, …, ir:ic, …, ŋanfaŋ, alfaŋ}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.3.2 for a visual representation of this transducer's transitions.

5.3.2 Description

The output of the second transducer is reversed before it enters the third transducer, which then reads it from left to right (e.g., <cɨ:ri> becomes <ir:ɨc>). If the machine reads one of the peripheral high vowels {i, u}, then it will enter that vowel's state, and it will output any consonants it encounters faithfully. If it encounters a central high vowel, then it will output that vowel as either {i, u}, depending on which one was seen first. If it encounters a vowel other than {i, u}, then it will move to that vowel's state and output any consonant or vowel it encounters faithfully, unless it reads another peripheral high vowel, in which case it will revert back to that vowel's state.

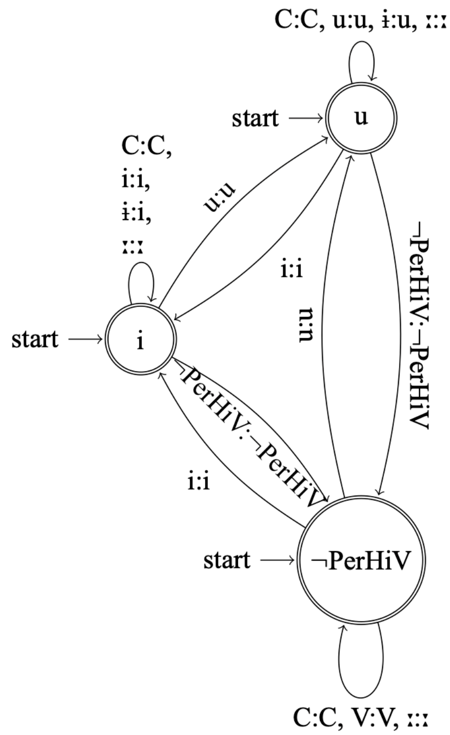

Figure 5. High Vowel Agreement[2]

5.3.3 Predictions

This machine only changes <ɨ> to the right of a peripheral high vowel, and so it can output sequences like [i..ɨ] (rev: <ɨ..i>) and [u..ɨ] (rev: <ɨ..u>), which are not attested in Kera. Pearce (2007b) says that there are no suffixes which end in underlying /ɨ/, so these sequences cannot be ruled out in polymorphemic words. In monomorphemic words, all root vowels are identical unless they are extrametrical. Pearce analyses the rightmost light syllables as unfooted, and therefore untouched by harmony. The only such example we could find is in (53 below) (Pearce 2007b:110).

(53) /giidirɨ/ [(gii).(di).rɨ] 'tribe, species' rev: <ɨridiig>

[2] Transducers can only have one start state q_0, but this FST shows three start states. This is just for graphical simplicity. The real transducer (i.e., its mathematical definition, see s. 5.3.1) only has one start state, and from that start state, it can transition to any of the three vowel states. Adding a single start state graphically, however, results in too many arcs mapping to and from every state to read clearly.

If there are multiple sets of high vowels in a word, this machine predicts several agreement domains. For instance, an input containing a sequence <u..ɨ..i..ɨ> will be output as <u..u..i..i>. To our knowledge, this sequence is unattested.

Finally, this transducer cannot account for the fact that the sequence [ɨ..i] (rev: <i..ɨ>) can sometimes surface. As mentioned above, unfooted syllables do not undergo harmony (54a). Additionally, the definite article /-ŋ/ preserves the vowel quality of these extrametrical vowels (54c), even though they become phonologically footed by the weight of the new coda (cf. 54b).

(54a)	/pɨrki/	[(pɨr).ki]	'mountain'	rev: <i̲krɨ̲p>
(54b)	/k-pɨrki-w/	[(kɨ.pɨr).(kɨw)]	'mountains'	rev: <wɨkrɨpɨk>
(54c)	/pɨrki-ŋ/	[(pɨr).(kiŋ)]	'the mountain'	rev: <ŋi̲krɨ̲p>

5.4 Transducer 4: Raising

5.4.1 Definition of transducer

Q: $\{\neg V_{+hi}, V_{+hi}, V_{+hi}C\}$

q_0: $\{\neg V_{+hi}\}$

F: $\{\neg V_{+hi}, V_{+hi}, V_{+hi}C\}$

*Σ**: {haman, hamam, …, guna, …, ŋafnaŋ, ŋafla}

*Γ**: {haman, hamam, …, gunɨ, …, ŋafnaŋ, ŋafla}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.4.2 for a visual representation of this transducer's transitions.

5.4.2 Description

The output string of the third transducer is reversed before it enters the fourth transducer, which reads it from left to right (e.g., <urːuc> becomes <cuːru>). The fourth transducer reads inputs until it encounters a [+hi] vowel followed by any number of consonants. Then, the next vowel it reads will be output as [+hi], after which the machine returns to its initial (waiting) state. All other inputs produce faithful (identical) outputs.

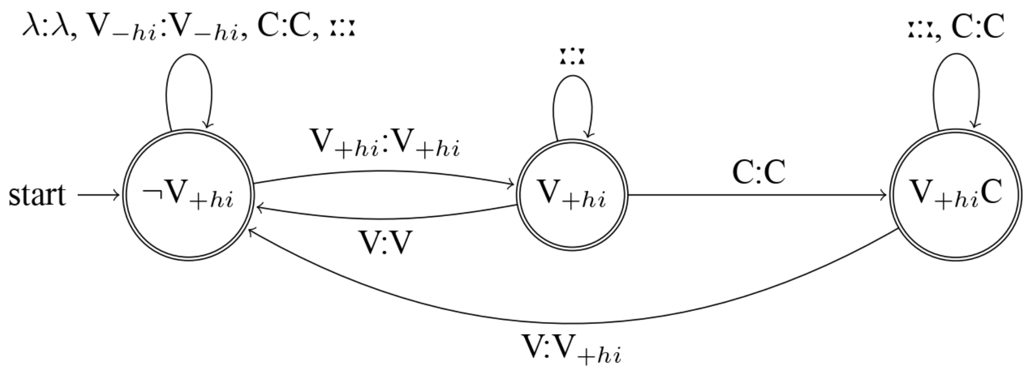

Figure 6. Raising

5.4.3 Predictions

This transducer raises the second of any non-overlapping pair of vowels that it encounters (separated by any number of consonants) in which the first is [+hi]. The examples below show input-output mappings for words longer than two syllables:

- /CiCaCaC/ → [CiCɨCaC]
- /CɨCoCuCeC/ → [CɨCuCuCiC]

5.5 Transducer 5: Low Dissimilation

5.5.1 Definition of transducer

Q: {¬a, a, aC}

q_0: {¬a}

F: {¬a, a, aC}

*Σ**: {namah, mamah, …, ŋanfaŋ, alfaŋ}

*Γ**: {namɨh, mamɨh, …, ŋanfaŋ, alfaŋ}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.5.2 for a visual representation of this transducer's transitions.

5.5.2 Description

The output string of the fourth transducer is reversed before it enters the fifth transducer, which reads it from left to right (e.g., <gi:dɨ> becomes <ɨd:ig>). The fifth transducer reads inputs until it reads an /a/ followed by any single consonant. At that point, if it reads another /a/ it will output an [ɨ] and return to its initial (waiting) state. All other inputs produce faithful (identical) outputs.

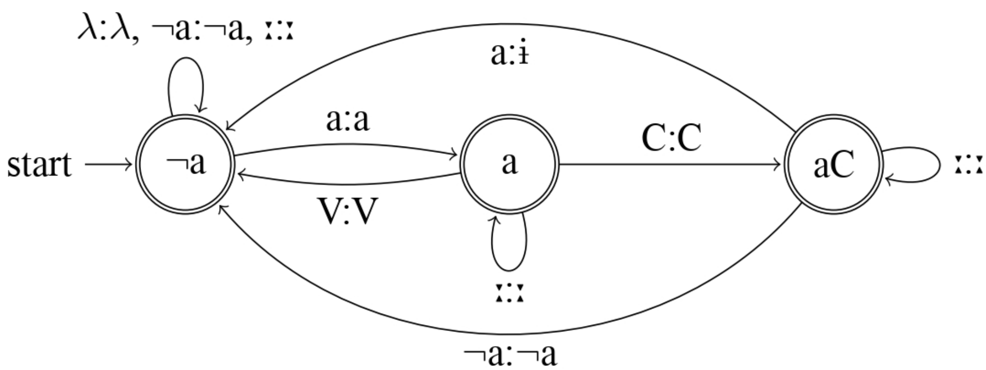

Figure 7. Low Dissimilation

5.5.3 Predictions

This transducer raises the second of any non-overlapping pair of /a/s (separated by precisely one consonant). If it were to encounter a word longer than two syllables it would map inputs to outputs in the following ways:

- /CaCaCaC/ → [CaCɨCaC]
- /CaCaCaCaC/ → [CaCɨCaCɨC]

5.6 Transducer 6: Guttural Lowering

5.6.1 Definition of transducer

Q: {[-lo], [+lo]}

q_0: {[-lo]}

F: {[-lo], [+lo]}

Σ^*: {hɨman, hɨmam, …, ŋafnaŋ, ŋafla}

Γ^*: {haman, hamam, …, ŋafnaŋ, ŋafla}

See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.6.2 for a visual representation of this transducer's transitions.

5.6.2 Description

The output string of the fifth transducer is reversed before it enters the sixth transducer, which reads it from left to right (e.g., <namɨh> becomes <hɨman>). The sixth transducer reads inputs until it encounters a [+lo] segment. At that point, if it reads an /ɨ/ immediately, it will output an [a] and return to its initial (waiting) state. All other inputs produce faithful (identical) outputs.

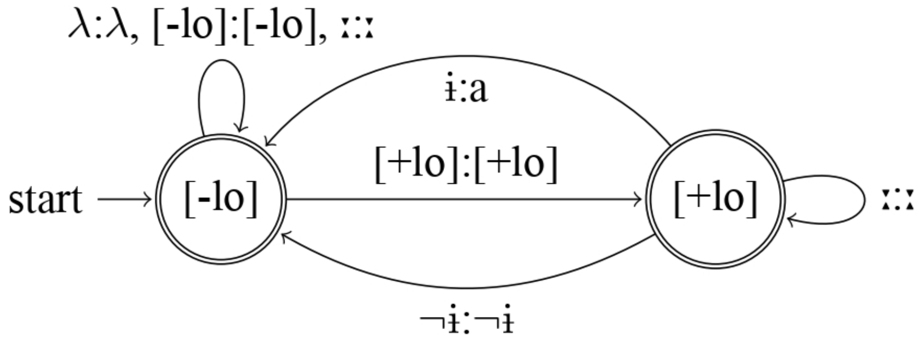

Figure 8. Guttural Lowering

5.6.3 Predictions

This transducer lowers /ɨ/ immediately following any [+lo] segment, which here means either [h] or [ʔ]. (In principle this machine could also lower an /ɨ/ immediately following a [+lo] vowel, but such vowel sequences are not permitted by the language's phonotactics.)

5.7 Transducer 7: Raising

5.7.1 Definition of transducer

*Σ**: {namah, mamah, …, unːes, …, ŋanfaŋ, alfaŋ}

*Γ**: {namɨh, mamɨh, …, unːis, …, ŋanfaŋ, alfaŋ}

See section 5.4.1 for the full definition of this transducer. See section 4.2 for a definition of the input and output alphabets, and see the FST diagram in 5.4.2 for a visual representation of this transducer's transitions.

5.7.2 Description

The output string of the sixth transducer is reversed before it enters the seventh transducer, which reads it from left to right (e.g., <seːnu> becomes <unːes>). The seventh stage applies the raising transducer again (see Section 5.4); however, this time it is moving through the string in what is effectively the opposite direction.

Finally, the output string of the seventh transducer is reversed one more time in order to produce the final surface form.

6 Applying the Transducers

In Table 9 we follow the underlying form /ʔap-m/ (find-2.MASC, "find you (masc)") through all of the transducers in detail, to demonstrate the application of each machine and produce a final output.

Table 9. Detailed derivation of /ʔap-m/ → [ʔapam].

Underlying Form **Pre-processing**	**Transducer 1** **Vowel epenthesis**	**Transducer 2** **Vowel copy**	**Transducer 3** **High vowel agree**
Input: ʔap-m Reverse: mpaʔ	Input: mpaʔ m p a ʔ → m m p a ʔ → *v*p m p a ʔ → a m p a ʔ → ʔ Output: m*v*paʔ Reverse: ʔap*v*m	Input: ʔap*v*m ʔ a p *v* m → ʔ ʔ a p *v* m → a ʔ a p *v* m → p ʔ a p *v* m → a ʔ a p *v* m → m Output: ʔapam Reverse: mapaʔ	Input: mapaʔ m a p a ʔ → m m a p a ʔ → a m a p a ʔ → p m a p a ʔ → a m a p a ʔ → ʔ Output: mapaʔ Reverse: ʔapam
Transducer 4 **Raising**	**Transducer 5** **Low dissimilation**	**Transducer 6** **Guttural lowering**	**Transducer 7 (4)** **Raising**
Input: ʔapam ʔ a p a m → ʔ ʔ a p a m → a ʔ a p a m → p ʔ a p a m → a ʔ a p a m → m Output: ʔapam Reverse: mapaʔ	Input: mapaʔ m a p a ʔ → m m a p a ʔ → a m a p a ʔ → p m a p a ʔ → ɨ m a p a ʔ → ʔ Output: mapɨʔ Reverse: ʔɨpam	Input: ʔɨpam ʔ ɨ p a m → ʔ ʔ ɨ p a m → a ʔ ɨ p a m → p ʔ ɨ p a m → a ʔ ɨ p a m → m Output: ʔapam Reverse: mapaʔ	Input: mapaʔ m a p a ʔ → m m a p a ʔ → a m a p a ʔ → p m a p a ʔ → ɨ m a p a ʔ → ʔ Output: mapaʔ Reverse: ʔapam

Back in section 4's Table 5, we summarized eight different root types present in the Odden (2005) data. Table 10 shows the input-output derivations, including intermediate forms, for one specific example from each root type. All but one (imperative) suffixes are also represented in the table.

Table 10. Sample derivations by root type. Highlighted cells are those where the transducer produces an output different from its input (other than string reversals).

Root type:	Ci:C	CuC	Cɨ:C	CaC	Ca	HaC	CoC	Ce:C
UR:	gi:d-a	gun-ŋ	cɨ:r-u	bal-n-i	ba-pa	ham-n	kol-m	se:n-u
T1:	ad:ig	ŋ*v*nug	ur:ɨc	inlab	apab	n*v*mah	m*v*lok	un:es
T2:	gi:da	gunuŋ	cɨ:ru	balni	bapa	haman	kolom	se:nu
T3:	ad:ig	ŋunug	ur:uc	inlab	apab	namah	molok	un:es
T4:	gi:dɨ	gunuŋ	cu:ru	balni	bapa	haman	kolom	se:nu
T5:	ɨd:ig	ŋunug	ur:uc	inlab	apɨb	namɨh	molok	un:es
T6:	gi:dɨ	gunuŋ	cu:ru	balni	bɨpa	haman	kolom	se:nu
T7 (T4):	ɨd:ig	ŋunug	ur:uc	inlɨb	apɨb	namah	molok	un:is
SR:	gi:dɨ	gunuŋ	cu:ru	bɨlni	bɨpa	haman	kolom	si:nu

7 Discussion

In this paper, we have demonstrated that using a different analytical approach than previously found, namely OT (Pearce 2007b) and rule derivations (Pulleyblank, p.c., 2006; Hansson, p.c., 2008), can capture the phonological processes observed in Kera. In this approach, by using chains of left-subsequential OSL functions and reversing input strings prior to each OSL function the transducers were able to predict phonological alternations in the Odden (2005) dataset. The combination of string reversal and chained transducers used in our approach can account for the exceptionality in height harmony patterns. In particular, progressive and regressive harmony (i.e., raising) share properties such as their triggers (high vowels), targets (non-high vowels), and domain (the morphological word). This suggests the need for a single mechanism that can look both forwards and backwards from a given target (e.g., a..ɨ..a → ɨ..ɨ..ɨ).

We do note that other alternative strategies for capturing bidirectionality exist. From our present perspective, they include: (i) having two different machines, one leftward and one rightward; or (ii) using ISL functions, where any mismatched sequences within a bounded window of non-high and high vowels in the input are output together as all high (e.g., with a window size of 3: <e..u..e> → <i..u..i>). The former approach would fail to treat raising as a unified process. The latter cannot account for disharmonic exceptions, which in our account are predicted by ordering the Low

Dissimilation process *before* Raising's progressive application (e.g. bɨpa), but *after* its regressive one (e.g., unattested: *a..ɨ). By reversing input strings, we are both able to eliminate a mechanistic redundancy and also use ordering to explain this aspect of data's exceptionality. We also note that this exceptionality is not straightforwardedly captured in Pearce (2007b)'s parallel OT analysis (see her discussion, p.128).

It is also important to acknowledge that many other points outside of the small dataset of the focus of this paper (Odden, 2005: 213-214) remain to be accounted for. Our analysis did not consider domain differences for certain processes; adding prosodic boundaries symbols to the transducers could account for phenomena like the non-targeting of extrametrical vowels. Additionally, this dataset lacked any data points bigger than two syllables in length or prefixes. As a result our analysis is only limited to disyllabic root-suffix words where the roots affected the alternations in the suffix(es) and vice-versa. Some of our transducers' predictions might not be borne out in longer words with different foot structures and/or increased morphological complexity.

References

Burness, P. (2022). Non-local phonological processes as Multi-Tiered Strictly Local maps [PhD Dissertation]. University of Ottawa.

Camburn, J. (1984). A Relational Grammar Approach to Kera Syntax. *Work Papers of the SIL*, University of North Dakota 28. 1-49.

Chandlee, J. (2014). *Strictly Local Phonological Processes* [PhD Dissertation]. University of Delaware.

Chandlee, J. and Jardine, A. (2024). Using locality and natural classes to infer underlying representations and a phonological grammar [Under Revision].

Ebert, K. (1975). Sprache und Tradition der Kera (Tschad),TeilI:Texte. Berlin: Reimer.

Ebert, K. (1976). Sprache und Tradition der Kera (Tschad), Teil II: Lexikon. Berlin: Reimer.

Ebert, K. (1979). Sprache und Tradition der Kera (Tschad), Teil III: Grammatik. Berlin: Reimer.

Ebert, K. (2003). 'Come' and 'Go' as Discourse Connectors in Kera and Other Chadic Languages. In Erin Shay and Uwe Seibert (eds.), *Motion, direction and location in languages: in honor of Zygmunt Frajzyngier*, 111-122. Amsterdam & Philadelphia: John Benjamins Publication.

Heinz, J., & Lai, R. (2013). Vowel Harmony and Subsequentiality. *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*, 52–63.

Odden, D. (2005). Doing an analysis. In *Introducing Phonology* (pp. 169–224). Cambridge University Press. https://doi.org/10.1017/CBO9780511808869.009

Oncina, J., Garcia, P., & Vidal, E. (1993). Learning subsequential transducers for pattern recognition interpretation tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *15*(5), 448–458. https://doi.org/10.1109/34.211465

Pearce, M. (2006a). The Interaction between metrical structure and tone in Kera. In Bert Remijsen & Vincent J. van Heuven (eds.), *Between stress and tone*: Special issue of *Phonology* 23(2), 259–286.

Pearce, M. (2006b). Evidence against voicing spread in Kera. *UCL Working Papers in Linguistics* 18, 345–355. Department of Phonetics and Linguistics, University College London.

Pearce, M. (2006c). Enhancement between voice onset time (VOT) and F0: Evidence from Kera voicing perception. Presented at a special session on Enhancement and Multiplicity of Cues in Speech at the Acoustical Society of America, 151st Meeting, Rhode Island. [Abstract in *Journal of the Acoustical Society of America* 119(5), 3269.]

Pearce, M. (2007a). Iambicity in Kera. 3*7th Annual Conference on African Linguistics*, 66–76. Somerville, MA: Cascadilla Press.

Pearce, M. (2007b). *The interaction of tone with voicing and foot structure:* [PhD Dissertation]. University of London.

Pearce, M. (2008). Vowel harmony domains and vowel undershoot. *UCL Working Papers in Linguistics* 20, 115–140. Department of Phonetics and Linguistics, University College London.

Pearce, M. (2009). Kera voicing and tone interaction. *Lingua* 119(6), 846–864.

Pearce, M. (2011). Kera. *Journal of the International Phonetic Association*, 41(2), 249–258. doi:10.1017/S0025100311000168