Multi-Input Strictly Local Functions for Tonal Phonology

Hossep Dolatian and Jonathan Rawski
hossep.dolatian,jonathan.rawski@stonybrook.edu

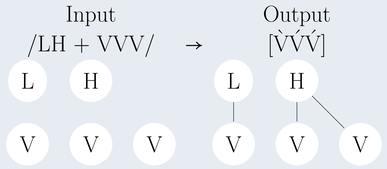
INSTITUTE FOR ADVANCED COMPUTATIONAL SCIENCE

Department of Linguistics, Stony Brook

TONE

Tone is autosegmental

- Vowels surface with H or L tone
- Underlyingly the Vs and T's are on separate tiers
- Mende has left-to-right spread in *fèlámá* 'juncture'



Question: How local are tonal processes?
Answer: Tradeoffs in computation and data structure

Computing Tone:

String

- Regular (Bird and Ellison, 1994) but not local (Jardine, 2016a)
- Encoding systems needed (Kornai, 1995)
- String-encoded tones are at most regular (Yli-Jyrä, 2013, 2015)

Autosegmental Representations (ASRs) = *tone as graph*

- Local language (Jardine, 2016b, 2017)
- Some functions are Autosegmental-Input Strictly Local (A-ISL) (Koser et al., 2019; Chandlee and Jardine, 2019)

Multi-tape finite-state transducer (MT-FST)

- = *compute a graph flattened to string tuples*
- Sycnhronous MT-FST (Kiraz, 2000, 2001; Hulden, 2009)
- Asynchronous Non-deterministic MT-FST (Kay, 1987; Wiebe, 1992)

Multi-Input Strict Locality (MISL)

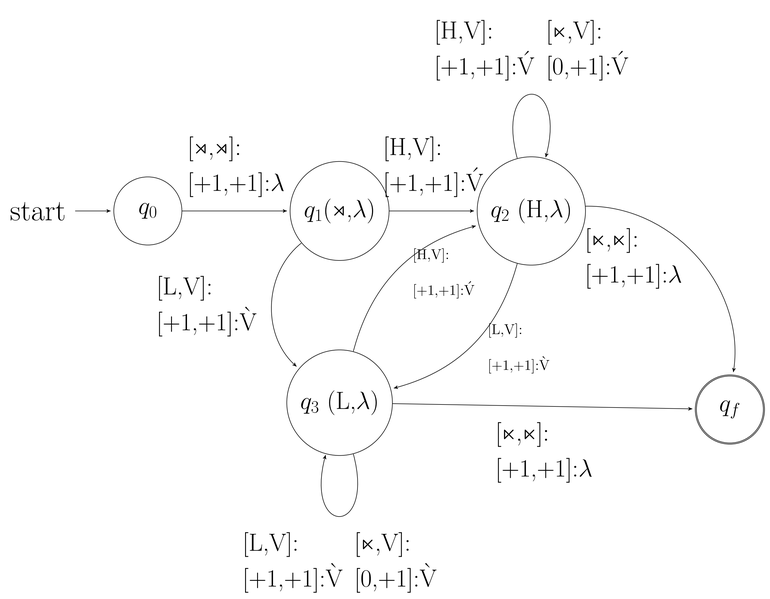
1-input function:	n-input function
$f(w)= v$	$f(\vec{w})= w$,
$w \in \Sigma^*$ is string	$\vec{w} \in (\Sigma^*)^n$ is tuple of strings w_i: [abc,de,fghi]
k is a natural number	$\vec{k}$ tuple of numbers k_i: [2,1,1]
suffix(w,k):	**suffix**$(\vec{w},\vec{k})$:
suffix(aba,2)=ba	**suffix**([abc,de,fghi], [2,1,1])= [bc,e,i]

Multi-Input Strictly Local
Input is 2+ strings on 2+ tapes
Deterministic + in one direction
Move (+1) or not (0) on a tape

M-ISL Multi-tape FSTs
f is MISL if its MT-FST M is
M is MISL if states are $\vec{k}$-suffix except initial and final state

Language-theoretic definition? In progress

Illustration: Mende Left-Right Spread



	Current state	Tone Tape	Vowel Tape	Output Symbol	Output string
1.	q_0	⋊HL⋉	⋊eaa⋉		
2.	q_1	⋊HL⋉ ⋊:+1	⋊eaa⋉ ⋊:+1	λ	
3.	q_2	⋊HL⋉ H:+1	⋊eaa⋉ e:+1	$\acute{e}$	$\acute{e}$
4.	q_3	⋊HL⋉ L:+1	⋊eaa⋉ a:+1	$\grave{a}$	$\acute{e}\grave{a}$
5.	q_3	⋊HL⋉ ⋉:0	⋊eaa⋉ a:+1	$\grave{a}$	$\acute{e}\grave{a}\grave{a}$
6.	q_f	⋊HL⋉ ⋉:+1	⋊eaa⋉ ⋉:+1	λ	$\acute{e}\grave{a}\grave{a}$

Table 1: Derivation of *HL + felama* over tone-vowel tiers *HL + eaa*

Mende is (2,1)-MISL =

- check window of size 2 on T: final T?
- check window of size 1 on V: any V?

Sample of Empirical Coverage

Language	Process	Pre-ass?	ISL	A-ISL	MISL	$\vec{k}$-value
Mende	Iterative left-right spread /LH + VVV/ → [ṲV́V́]	✗		✓*	✓	[2,1]
Kikuyu	Initial spread to two + final spread /LHLH + VVVVVVV/ → [ṲV̀V́V̀V́V́]	✗		✓*	✓	[2,3]
Hausa	Iterative right-left spread /LH + VVV/ → [ṲV̀V́]	✗		✓*	✓	[2,1]
Northern Shona	Edge-in + initial spread + medial spread /HLH + VVVVVV/ → [V́V́V̀V̀V́]	✗		✓*	✓	[4,6]
Kukuya	Quantity sensitive spreading /H + VVVV/ → [V́V́V́]	✗		✓*	✓	[4,2]
Rimi	Bounded tone shift /VV́VV/ → [VVV́V] /<H> + V<V>VV/	✓	✓	✓	✓	[1,2]
Zigula	Unbounded tone shift /VVV́VVV/ → [VVVV́V] /<H> + V<V>VV/	✓	✗	✓	✓	[1,3]
Bemba	Bounded tone spread /VV́VV/ → [VV́V́V] /<H> + V<V>VV/	✓	✓	✓	✓	[1,2]
Arusa	Unbounded deletion /V́ VV́VV/ → [V́VVVV] /<H> <H> + <V> V(VV)V/	✓	✗	✓	✓	[3,1]
Luganda	Bounded Meussen's rule /V́V́V́V/ → [V́V̀V̀V] /<H> <H> + <V>(VV)V/	✓	✓	✗	✓	[2,2]**
Shona	Alternating Meussen's rule /V́-V́-V́/ → [V́-V̀-V́] /<H>-<H>-<H> + <V>-<V>-<V>/	✓	✗	✗	✗	
Ndebele	Unbounded spreading to ante-penultimate /VVVVV/ → [V́V́VVV] /<H> + <V>VVVV/	✓	✗	✗	✓	[1,3]

Table 2: *Legend:* * Function was proved to be QFLFP by Koser et al. (2019), ** Function is MISL if the output is 2-tuple

References

Bird, S. and T. M. Ellison (1994). One-level phonology: Autosegmental representations and rules as finite automata. Computational Linguistics 20(1), 55–90.
Chandlee, J. and A. Jardine (2019). Autosegmental input strictly local functions. Transactions of the Association for Computational Linguistics 7, 157–168.
Hulden, M. (2009). Revisiting multi-tape automata for semitic morphological analysis and generation. In Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages, pp. 19–26. Association for Computational Linguistics.
Jardine, A. (2016a). Computationally, tone is different. Phonology 33(2), 247–283.
Jardine, A. (2016b). Locality and non-linear representations in tonal phonology. Ph. D. thesis, University of Delaware, Newark, DE.
Jardine, A. (2017). The local nature of tone-association patterns. Phonology 34(2), 363–384.
Kay, M. (1987). Nonconcatenative finite-state morphology. In Third Conference of the European Chapter of the Association for Computational Linguistics.
Kiraz, G. A. (2000). Multitiered nonlinear morphology using multitape finite automata: a case study on syriac and arabic. Computational Linguistics 26(1), 77–105.
Kiraz, G. A. (2001). Computational nonlinear morphology: with emphasis on Semitic languages. Cambridge University Press.
Kornai, A. (1995). Formal phonology. Garland Publishing Inc.
Koser, N., C. Oakden, and A. Jardine (2019). Tone association and output locality in non-linear structures. In Supplemental proceedings of AMP 2019.
Wiebe, B. (1992). Modelling autosegmental phonology with multi-tape finite state transducers. Master's thesis, Simon Fraser University.
Yli-Jyrä, A. (2013). On finite-state tonology with autosegmental representations. In Proceedings of the 11th international conference on finite state methods and natural language processing. Association for Computational Linguistics.
Yli-Jyrä, A. (2015). Three equivalent codes for autosegmental representations. In Proceedings of the 12th International Conference on Finite-State Methods and Natural Language Processing 2015 (FSMNLP 2015 Düsseldorf).