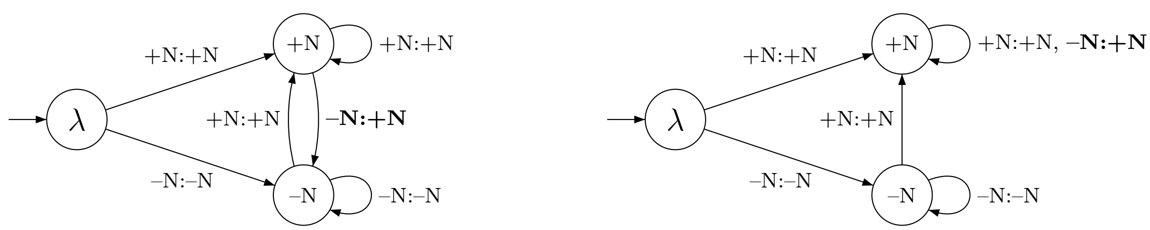
The computational nature of iterativity in long-distance phonology

Recent work investigating the formal complexity of phonological patterns has led to a deeper understanding of the computational properties that underlie attested processes. The current research investigates the distinction between processes that apply to a sequence of multiple targets ('iterative') and those that affect only the first target in such a sequence ('non-iterative'). From the perspective that phonological processes can be seen as *maps* from strings of input segments to strings of output segments, previous research suggests that non-iterative patterns are the result of input-based computation, whereas output-based computation generally results in iterative application. We will argue that for long-distance processes being computed over phonological tiers, this is not necessarily the case.

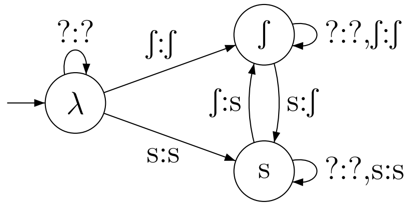
In modeling phonological interactions between adjacent segments, Chandlee (2014) defines two classes of string-to-string maps called the Input Strictly Local (ISL) and Output Strictly Local (OSL) functions. In terms of finite state transducers (FSTs), the difference between these is whether the landing state for each transition is dictated by the input segment being read (the left side of an ***a***:*x* transition), or the output segments being written at that step (the right side of an *a*:***x*** transition). Consider, for example, a process of progressive nasal spreading where segments become nasalized when immediately following another nasal segment. When modeled as an ISL function, as shown with the FST on the left in (1), the pattern will be non-iterative, since the relevant –N:+N transition lands in the '–N' state (the last input), which is not itself a trigger. This FST could be used to model nasal spreading in Auca (e.g., /wɑĩ-gɑ/ ↦ [wɑĩ-ŋɑ] 'good tooth'; Steriade, 1993). By contrast, in the OSL transducer shown on the right in (1), the corresponding transition is directed to the '+N' state. That is, the segment written to the output string is +N, which is precisely what triggers the alternation, and so the OSL process will apply iteratively to multiple targets (e.g., /mewah/ ↦ [mẽw̃ãh̃] 'to be luxurious' in Johore Malay; Onn, 1980).

(1) FSTs for nasal assimilation (ISL, left; OSL right).

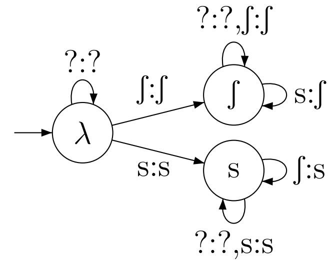


When generalizing this approach to account for long-distance phonology, these functions can be augmented with a tier comprised of relevant segments, resulting in the *Input Tier-based Strictly Local* (ITSL) and *Output Tier-based Strictly Local* (OTSL) functions (McMullin and Chandlee, 2018; Burness and McMullin, 2019). In terms of FSTs, transitions labeled with non-tier segments are required to simply loop back to the state where they originated. In this way, long-distance processes are able to skip over any irrelevant material that intervenes between trigger and target. This is shown below for an example process of sibilant harmony where sibilants must agree for anteriority across any distance, as in Aari (e.g., /ʃed-er-s-it/ ↦ [ʃederʃit] 'I was seen'; Hayward, 1990).

(2) ITSL FST for progressive sibilant harmony

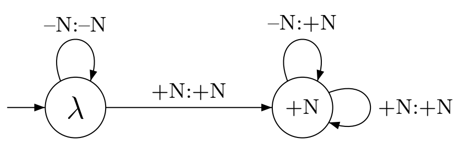


(3) OTSL FST for progressive sibilant harmony



Based in part on the observation that virtually all attested patterns of consonant harmony apply iteratively (Hansson, 2010), McMullin and Chandlee (2018) argue that such processes must result from output-oriented computation. We note that the FSTs above do indeed retain the (non-)iterative characteristics of their SL counterparts. That is, for an input string such as /ʃasasa/, the ITSL transducer in (2) outputs the string [ʃaʃasa], whereas the OTSL transducer in (3) outputs [ʃaʃaʃa]. However, we will show that when computing over phonological tiers, the generalization that 'ITSL implies non-iterative, OTSL implies iterative' no longer holds. Specifically, incorporating tiers into these functions results in a descriptive overlap between processes that are truly iterative (i.e., propagating to multiple targets one step at a time) and processes in which a single trigger affects multiple targets (even if they are not inherently iterative). Because of this, certain patterns that appear to be iterative can be characterized as ITSL functions, provided that they are asymmetric (e.g., sibilant harmony triggered by [–ant], but not [+ant]). Even local processes such as 'iterative' nasal assimilation can be construed in this way. For the example from Johore Malay above, [+nas] segments in the input string would be on the tier, but [–nas] segments would not, despite being targets of the process. This is shown with the ITSL FST below in (4).

(4) ITSL FST for progressive nasal assimilation with multiple targets



The overall goal of this research is to provide us with a better understanding of foundational differences between computing iterative and non-iterative processes, which is important for evaluating the predictions and viability of any theory of phonology.

Select references: • Burness, P. and McMullin, K. (2019). Efficient learning of Tier-based Strictly 2-Local functions. In *Proceedings of the 16th Meeting on the Mathematics of Language (MOL 16)*, pages 78–90. Association for Computational Linguistics. • Chandlee, J. (2014). *Strictly Local phonological processes*. PhD thesis, UD. • Hansson, G. Ó. (2010). *Consonant harmony: long-distance interaction in phonology*. University of California Press, Berkeley, CA. • McMullin, K. and Chandlee (2018). Output-based computation and unbounded phonology. Paper presented at AMP 2018, San Diego, CA.