Algebraic Reanalysis of Phonological Spreading

Dakotah Lambert* and Jeffrey Heinz**

*Université Jean Monnet Saint-Étienne, CNRS
Institut d Optique Graduate School
Laboratoire Hubert Curien UMR 5516

**Department of Linguistics
Institute of Advanced Computational Science
Stony Brook University

Structures Formelles du Langage
CNRS
14.11.2023

Today

1. There are multiple extensionally equivalent ways to analyze iterative spreading processes.
2. Most phonologists view them as output-oriented, local processes partly because common-sense suggests such analyses are constrained.
3. An alternative analysis posits they are akin to long-distance harmony.
4. The algebraic analysis conducted here supports this latter view in a constrained way.
5. Furthermore, computational analysis of output-oriented processes indicates they are not as constrained as common-sense would have us believe.

Phonology:
A functional view

Iterative Spreading

Johore Malay ‘supervision’

/pəŋawasan/ → [pəŋãw̃ãsan]

1. [−cons] → [+nas]/[+nas]___ (left-to-right application)
2. Agree(nasal)>>IdentIO[nasal].

Phonological constraints and transformations are infinite objects

Extensions of grammars in phonology are infinite objects in the same way that circles represent infinitely many points.

Iterative Nasal Spreading: Intensional Descriptions

1. [−cons] → [+nas]/[+nas]___ (left-to-right application)
2. Agree(nasal)>>IdentIO[nasal].

Iterative Nasal Spreading: Extensional Description

(ane, anẽ), (ame, amẽ), (anewa, anẽw̃ã),
(apila, apila),...(mawisaŋawala, mãw̃ĩsaŋãw̃ãw̃ã), ...

Many ways to describe the same extensions

1. $[-\text{cons}] \rightarrow [+\text{nas}] / [+\text{nas}]\_\_\_$ (left-to-right application)

2. $[-\text{cons}] \rightarrow [+\text{nas}] / [+\text{nas}][-\text{cons}]^{*}\_\_\_$ (simultaneous application)

Many ways to describe the same extensions

1. $[-\text{cons}] \rightarrow [+\text{nas}] / [+\text{nas}]\_\_\_$ (left-to-right application)

2. $[-\text{cons}] \rightarrow [+\text{nas}] / [+\text{nas}][-\text{cons}]^{*}\_\_\_$ (simultaneous application)

1. $[-\text{cons}] \rightarrow [+\text{nas}]/[+\text{nas}]\_\_\_$ (left-to-right application)

2. $[-\text{cons}] \rightarrow [+\text{nas}]/[+\text{nas}][-\text{cons}]^{*}\_\_\_$ (simultaneous application)

What kind of evidence could distinguish them?

Grammars describe string functions

function	Notes	
$f : \Sigma^* \to \{0, 1\}$	Binary classification	(well-formedness)
$f : \Sigma^* \to \mathbb{N}$	Maps strings to numbers	(well-formedness)
$f : \Sigma^* \to [0, 1]$	Maps strings to real values	(well-formedness)
$f : \Sigma^* \to \Delta^*$	Maps strings to strings	(transformation)
$f : \Sigma^* \to \wp(\Delta^*)$	Maps strings to sets of strings	(transformation)

What kind of functions are they?

Background

Research Goals

The computational nature of natural language characterizes the computations involved in knowing and learning natural languages. Its study spans all subdisciplines of linguistics.

Hypotheses about the computational nature of language:

1. Make typological and psycholinguistic predictions
2. Leads to new learning algorithms
3. Inform Machine Learning, Natural Language Processing, Cognitive Science, and other aspects of science and engineering.

Previously

It has been argued that the computational nature of phonological generalizations is

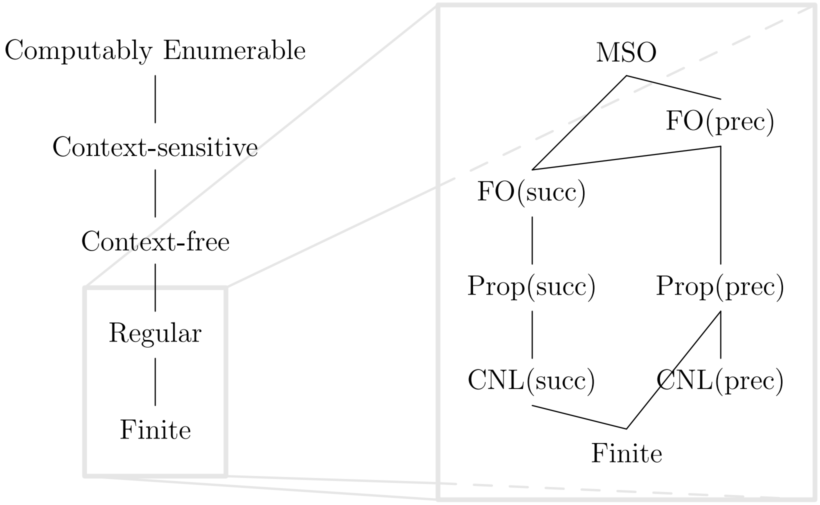
1. "regular", and in fact,
2. "less than" regular in a particularly "local" way

(Heinz, 2018; Graf, 2022, and others)

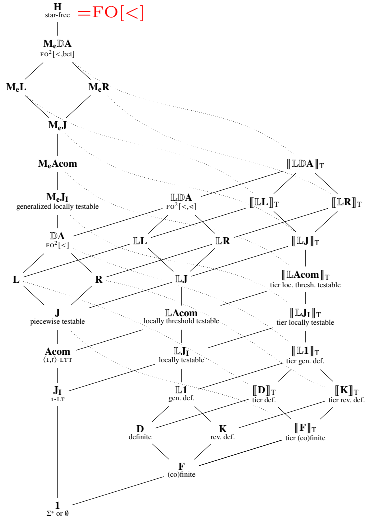
Requires two books:

- “encyclopedia of categories”
- “encyclopedia of types”

Wilhelm Von Humboldt

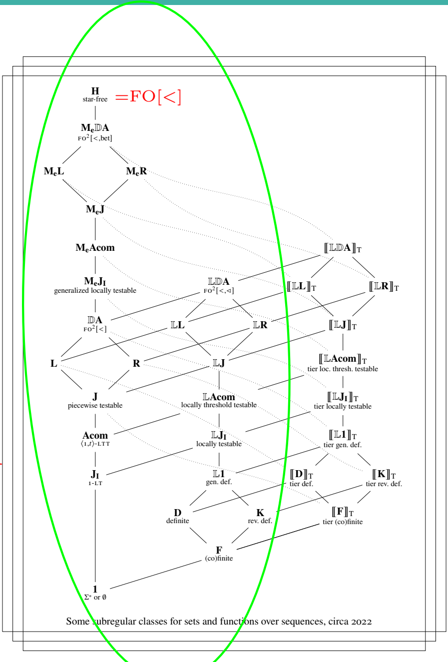

Computably Enumerable
Context-sensitive
Context-free
Regular
Finite
MSO
FO(prec)
FO(succ)
Prop(succ)
Prop(prec)
CNL(succ)
CNL(prec)
Finite

Algebraic Perspective (Lambert 2022, 2023)



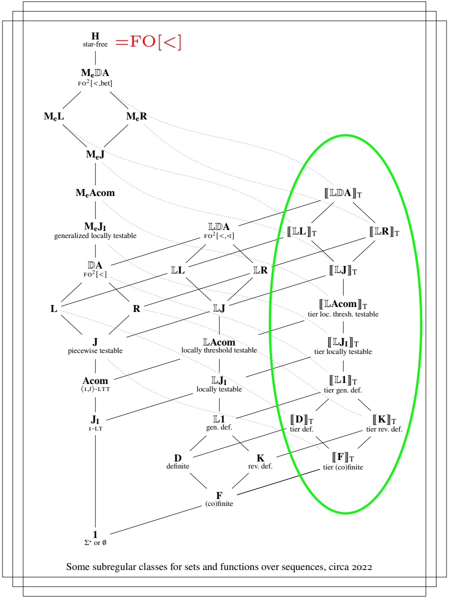
Some subregular classes for sets and functions over sequences, circa 2022

Algebraic Perspective (Lambert 2022, 2023)



Some subregular classes for sets and functions over sequences, circa 2022

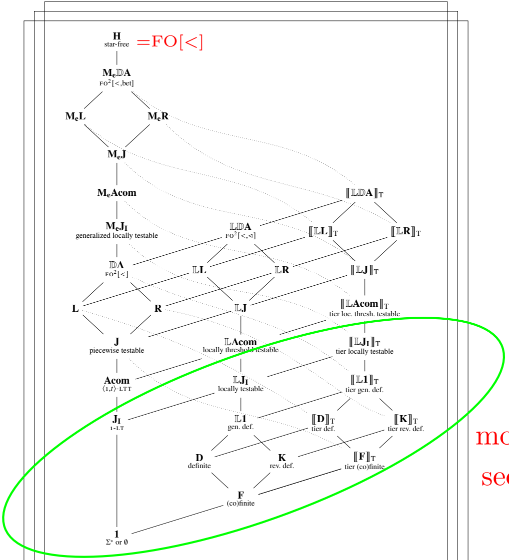
studied previously in mathematics and computer science



Some subregular classes for sets and functions over sequences, circa 2022

Lambert studied tier-projections

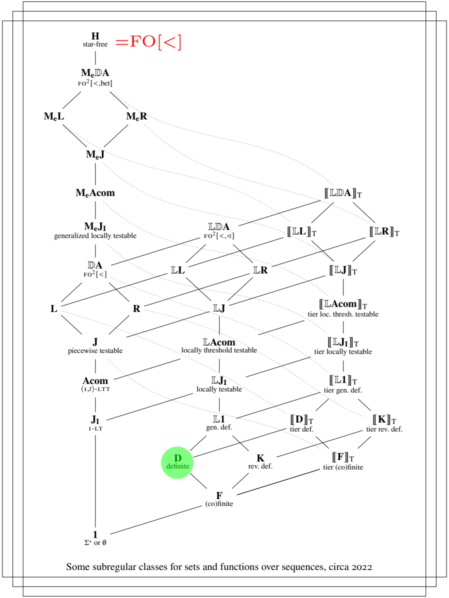
Algebraic Perspective (Lambert 2022, 2023)



Some subregular classes for sets and functions over sequences, circa 2022

morpho-phonology seems to be here?

Algebraic Perspective (Lambert 2022, 2023)

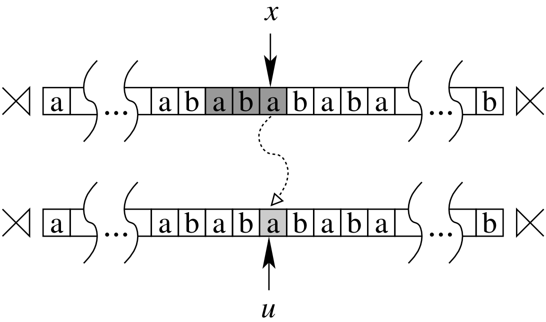


Some subregular classes for sets and functions over sequences, circa 2022

"Definite" means local

ISL, OSL, and Definite Functions

Input Strictly Local Functions (Chandlee 2014)



The output at position i only depends on the ith symbol and the previous $k-1$ symbols in the input.

Input Strictly Local Functions (con't)

- Have good empirical coverage of local phonological and morphological processes.
- Have efficient, interpretable and provably correct learning algorithms (given k).
- Have multiple, equivalent characterizations and are directly related to Strictly Local stringsets.

(Rogers and Pullum, 2011; Chandlee, 2014; Chandlee *et al.*, 2014; Jardine *et al.*, 2014; Lindell and Chandlee, 2016; Chandlee, 2017; Chandlee and Heinz, 2018)

Phonological Processes that are not ISL

1. Iterative Spreading (as in Johore Malay)
2. Long-distance vowel and consonant harmony

(Chandlee, 2014; Chandlee *et al.*, 2015; McMullin, 2016; Chandlee and Heinz, 2018; Burness and McMullin, 2019; Burness *et al.*, 2021)

Phonological Processes that are not ISL

1. Iterative Spreading (as in Johore Malay)
   Phonological view: local on the surface form

2. Long-distance vowel and consonant harmony
   Phonological view: local on a tier-projection of the surface form

(Chandlee, 2014; Chandlee *et al.*, 2015; McMullin, 2016; Chandlee and Heinz, 2018; Burness and McMullin, 2019; Burness *et al.*, 2021)

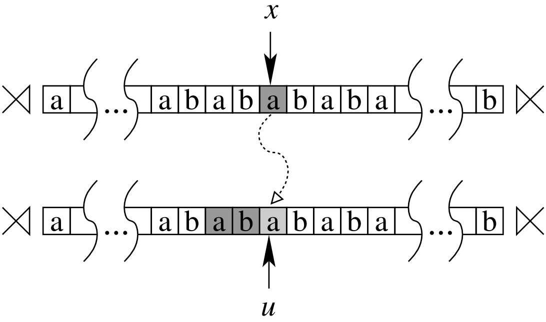
Phonological Processes that are not ISL

1. Iterative Spreading (as in Johore Malay)
   Phonological view: local on the surface form
   Functional class: Output Strictly Local Functions

2. Long-distance vowel and consonant harmony
   Phonological view: local on a tier-projection of the surface form
   Functional class: Tier-based Output Strictly Local Functions

(Chandlee, 2014; Chandlee *et al.*, 2015; McMullin, 2016; Chandlee and Heinz, 2018; Burness and McMullin, 2019; Burness *et al.*, 2021)

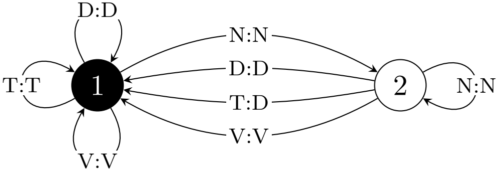
Output Strictly Local Functions



The output at position i only depends on the ith symbol and the previous $k-1$ symbols in the output (surface form).

/anta/ → [anda]

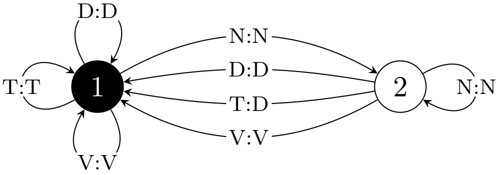
Minimal, Deterministic Finite-state Transducer



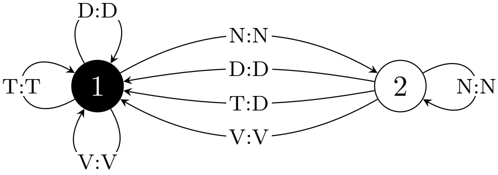
Symbol	Interpretation
V	vowel
N	nasal consonant
T	voiceless consonant
D	non-nasal, voiced consonant

/anta/ → [anda]

Minimal, Deterministic Finite-state Transducer



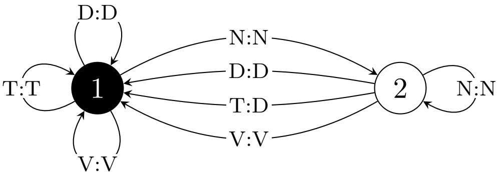
Post-nasal voicing is both 2-ISL and 2-OSL because one can determine the state by either the last input symbol or by the last output symbol.



There are two distinct actions that arise from the input letters.

$$\begin{aligned} \text{T, D, V}: &\quad \langle 1,2\rangle \mapsto \langle 1,1\rangle \\ \text{N}: &\quad \langle 1,2\rangle \mapsto \langle 2,2\rangle \end{aligned}$$

Algebraic Analysis (con't)



We follow the states to build a multiplication table.

$\cdot$	V	N
V	V	N
N	V	N

Definite Functions

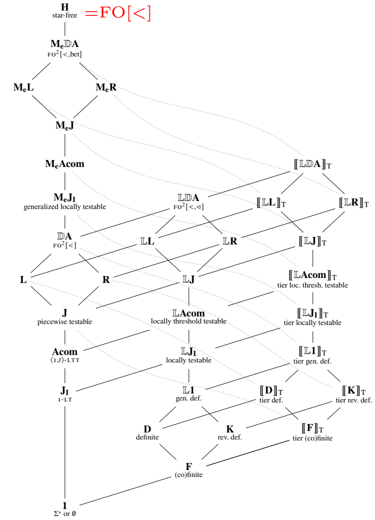
$\cdot$	V	N
V	V	N
N	V	N

Definitions

1. An element x is an **identity** if and only if for all y, it holds that $x \cdot y = y \cdot x = y$.
2. An element e is **idempotent** if and only if $e = e \cdot e$.
3. A function is **definite** if and only if for all elements x, and all idempotents e it holds that $x \cdot e = e$.
   (Visually, this means the idempotent elements “own” their columns.)

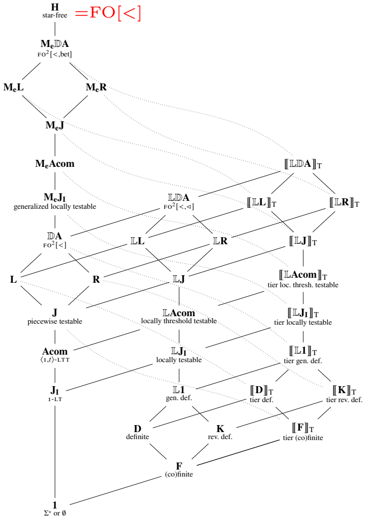
Theorem: Definite functions are exactly the input strictly local functions (Lambert and Heinz, 2023).

What are OSL functions algebraically?



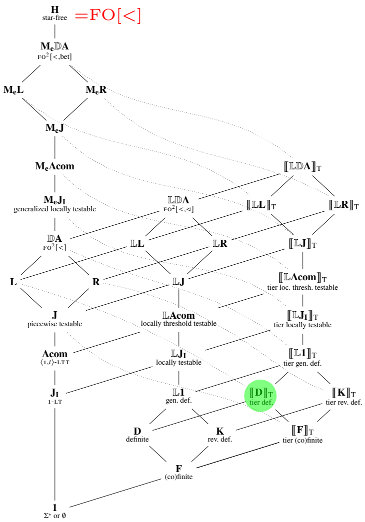
Some subregular classes for sets and functions over sequences, circa 2022

OSL functions can be found in every class! Lambert (2022)



Some subregular classes for sets and functions over sequences, circa 2022

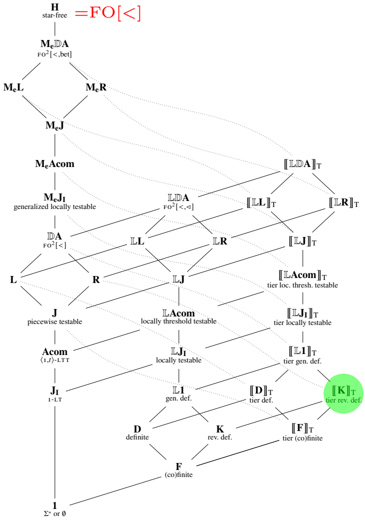
What are OSL functions algebraically?



Some subregular classes for sets and functions over sequences, circa 2022

But iterative spreading is here

What are OSL functions algebraically?



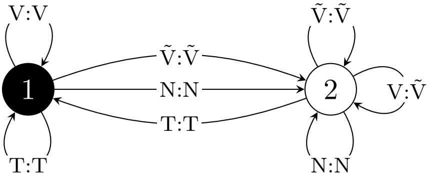
Some subregular classes for sets and functions over sequences, circa 2022

and long-distance harmony is here

Algebraic Analysis of Iterative Spreading

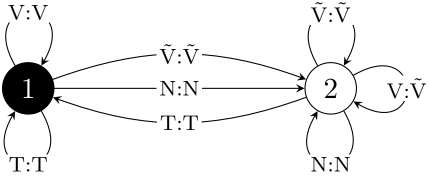
Minimal, Deterministic Finite-state Transducer

/pəŋawasan/ → [pəŋãw̃ãsan]



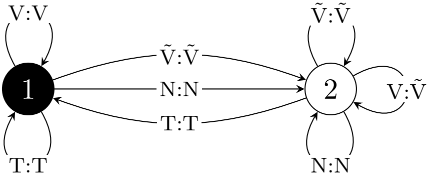
Symbol	Interpretation
V	oral vowel or semivowel
Ṽ	nasalized vowel or semivowel
N	nasalized consonant
T	any other symbol

Identifying Basic Behaviors



There are three distinct actions that arise from the letters.

$$\begin{aligned} \mathrm{V}: &\quad \langle 1,2\rangle \mapsto \langle 1,2\rangle \\ \mathrm{T}: &\quad \langle 1,2\rangle \mapsto \langle 1,1\rangle \\ \tilde{\mathrm{V}}, \mathrm{N}: &\quad \langle 1,2\rangle \mapsto \langle 2,2\rangle \end{aligned}$$



We follow the states to understand how they multiply.

·	V	T	N
V	V	T	N
T	T	T	N
N	N	T	N

Definite Functions

$\cdot$	V	T	N
V	V	T	N
T	T	T	N
N	N	T	N

Recall the definitions.

1. An element x is an **identity** if and only if for all y, it holds that $x \cdot y = y \cdot x = y$.
2. An element e is **idempotent** if and only if $e = e \cdot e$.
3. A function is **definite** if and only if for all elements x, and all idempotents e it holds that $x \cdot e = e$.
   (Visually, this means the idempotent elements "own" their columns.)

Conclusion 1: Iterative Spreading is not definite.

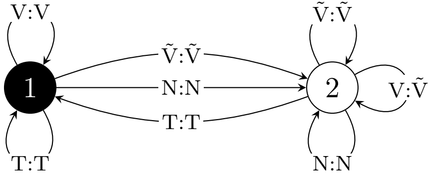
Definite Functions

·	V	T	N
V	V	T	N
T	T	T	N
N	N	T	N

Recall the definitions.

1. An element x is an **identity** if and only if for all y, it holds that $x \cdot y = y \cdot x = y$.
2. An element e is **idempotent** if and only if $e = e \cdot e$.
3. A function is **definite** if and only if for all elements x, and all idempotents e it holds that $x \cdot e = e$.
   (Visually, this means the idempotent elements “own” their columns.)

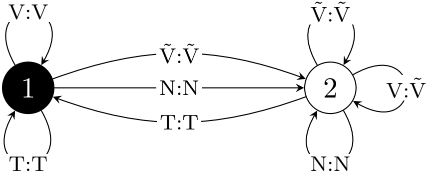
Conclusion 2: Iterative Spreading is tier-based definite! Why? Because removing the identity element reveals a definite structure!



·	V	T	N
V	V	T	N
T	T	T	N
N	N	T	N

- Elements projected to a tier matter – they can change state.
- Elements not projected to a tier never change state. These elements are **neutral**.
- Identity elements never change state!

Discussion



·	V	T	N
V	V	T	N
T	T	T	N
N	N	T	N

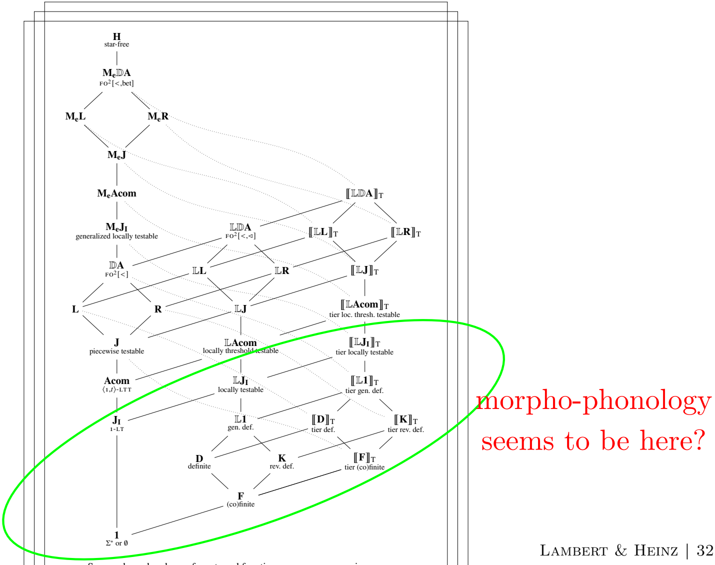
- The finite-state transducer is compatible with both perspectives.
- The algebraic analysis highlights a tier-based analysis: vowels and semivowels are neutral letters.

Conclusion

- Iterative spreading belongs to both OSL and Tier-based Definite classes.
- OSL processes are spread throughout the class of rational functions, and theories with global optimization (OT, HG) are not even bound by that.
- On the other hand, Tier-based Definite functions occupy a small well-behaved corner.
- To determine which makes for a better theory, studies of learnability and optionality can be brought to bear (in addition to restrictiveness, studied here).

Merci Beaucoup!

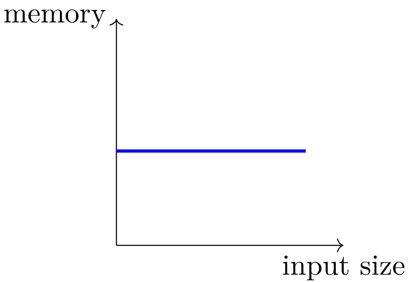
https://hackage.haskell.org/package/language-toolkit



Appendix

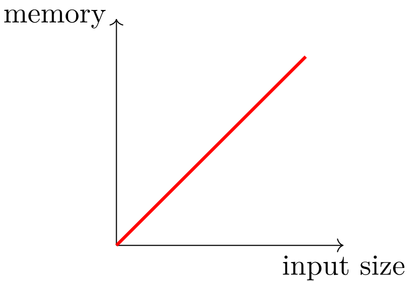
What "Regular" means

A set or function is regular provided **the memory required for the computation is bounded by a constant,** ***regardless of the size of the input.***



Blue line indicates a regular process

A set or function is regular provided **the memory required for the computation is bounded by a constant,** ***regardless of the size of the input.***



Red line indicates a non-regular process

Some computations important to grammar

- For given constraint C and any representation w:
  Does w violate C? How many times?

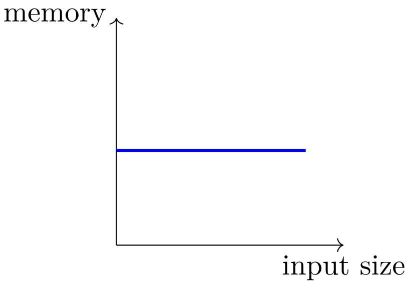
- For given grammar G and any underlying representation w:
  What surface representation(s) does G transform w to?
  With what probabilities?

Example: Vowel Harmony

Progressive
Vowels agree in backness with the first vowel in the underlying representation.
Majority Rules
Vowels agree in backness with the majority of vowels in the underlying representation.

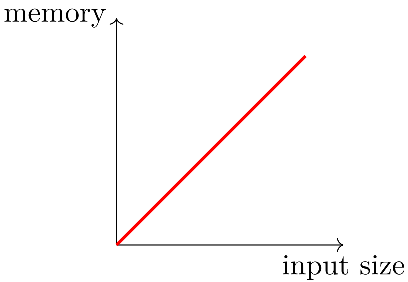
UR	Progressive	Majority Rules
/nokelu/	nokolu	nokolu
/nokeli/	nokolu	nikeli
/pidugo/	pidige	pudugo
/pidugomemi/	pidigememi	pidigememi

(Bakovic 2000, Finley 2008, 2011, Heinz and Lai 2013)



Progressive Harmony

Progressive and Majority Rules Harmony



Majority Rules Harmony

Some Perspective

Typological: Majority Rules is unattested. (Baković, 2000)

Psychological: Human subjects fail to learn Majority Rules in artificial grammar learning experiments, unlike progressive harmony. (Finley, 2008, 2011)

Computational: Majority Rules is not regular. (Riggle, 2004; Heinz and Lai, 2013)

Optimality Theory

1. There exists a CON and ranking over it which generates Majority Rules: Agree(back)>>IdentIO[back].

2. Changing CON may resolve this, but this solution misses the forest for the trees.

3. Global optimization over simple constraints requires non-constant memory (Gerdemann and Hulden, 2012; Lamont, 2021, 2022, and others)

Phonological generalizations are regular

Evidence supporting the hypothesis that phonological generalizations are finite-state originate with Johnson (1972) and Kaplan and Kay (1994), who showed how to translate any phonological grammar defined by an ordered sequence of SPE-style rewrite rules into a finite-state transducer.

Consequently:

1. Constraints on well-formed surface and underlying representations are regular since the image and pre-image of finite-state functions are regular. (Scott and Rabin, 1959)
2. Since virtually any phonological grammar can be expressed as an ordered sequence of SPE-style rewrite rules, this means "being regular" is a property of the functions that *any* phonological grammar defines.

Baković, Eric. 2000. Harmony, dominance and control. Doctoral dissertation, Rutgers University.

Burness, Phillip, and Kevin McMullin. 2019. Efficient learning of output tier-based strictly 2-local functions. In *Proceedings of the 16th Meeting on the Mathematics of Language*, 78–90. Toronto, Canada: Association for Computational Linguistics.

Burness, Phillip A., Kevin J. McMullin, and Jane Chandlee. 2021. Long-distance phonological processes as tier-based strictly local functions. *Glossa: a Journal of General Linguistics* 6.

Chandlee, Jane. 2014. Strictly local phonological processes. Doctoral dissertation, The University of Delaware.

Chandlee, Jane. 2017. Computational locality in morphological maps. *Morphology* 27:599–641.

Chandlee, Jane, Rémi Eyraud, and Jeffrey Heinz. 2014. Learning strictly local subsequential functions. *Transactions of the Association for Computational Linguistics* 2:491–503.

Chandlee, Jane, Rémi Eyraud, and Jeffrey Heinz. 2015. Output strictly local functions. In *Proceedings of the 14th Meeting on* *the Mathematics of Language (MoL 2015)*, edited by Marco Kuhlmann, Makoto Kanazawa, and Gregory M. Kobele, 112–125. Chicago, USA.

Chandlee, Jane, and Jeffrey Heinz. 2018. Strict locality and phonological maps. *Linguistic Inquiry* 49:23–60.

Finley, Sara. 2008. The formal and cognitive restrictions on vowel harmony. Doctoral dissertation, Johns Hopkins University, Baltimore, MD.

Finley, Sara. 2011. The privileged status of locality in consonant harmony. *Journal of Memory and Language* 65:74–83.

Gerdemann, Dale, and Mans Hulden. 2012. Practical finite state optimality theory. In *Proceedings of the 10th International Workshop on Finite State Methods and Natural Language Processing*, 10–19. Donostia–San SebastiÃ¡n: Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W12-6202

Graf, Thomas. 2022. Subregular linguistics: bridging theoretical linguistics and formal grammar. *Theoretical Linguistics* 48:145–184.

Heinz, Jeffrey. 2018. The computational nature of phonological generalizations. In *Phonological Typology*, edited by Larry Hyman and Frans Plank, Phonetics and Phonology, chap. 5, 126–195. De Gruyter Mouton.

Heinz, Jeffrey, and Regine Lai. 2013. Vowel harmony and subsequentiality. In *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*, edited by Andras Kornai and Marco Kuhlmann, 52–63. Sofia, Bulgaria.

Jardine, Adam, Jane Chandlee, Rémi Eyraud, and Jeffrey Heinz. 2014. Very efficient learning of structured classes of subsequential functions from positive data. In *Proceedings of the Twelfth International Conference on Grammatical Inference (ICGI 2014)*, edited by Alexander Clark, Makoto Kanazawa, and Ryo Yoshinaka, vol. 34, 94–108. JMLR: Workshop and Conference Proceedings.

Lambert, Dakotah, and Jeffrey Heinz. 2023. An algebraic characterization of total input strictly local functions. In *Proceedings of the Society for Computation in Linguistics*, vol. 6.

Lamont, Andrew. 2021. Optimizing over subsequences generates context-sensitive languages. *Transactions of the Association for Computational Linguistics* 9:528–537.

Lamont, Andrew. 2022. Optimality theory implements complex functions with simple constraints. *Phonology* 38:729–740.

Lindell, Steven, and Jane Chandlee. 2016. A logical characterization of input strictly local functions Presented at the Fourth Workshop on Natural Language and Computer Science, in affiliation with LICS 2016.

McMullin, Kevin. 2016. Tier-based locality in long-distance phonotactics: Learnability and typology. Doctoral dissertation, University of British Columbia.

Riggle, Jason. 2004. Generation, recognition, and learning in finite state Optimality Theory. Doctoral dissertation, University of California, Los Angeles.

Rogers, James, and Geoffrey Pullum. 2011. Aural pattern recognition experiments and the subregular hierarchy. *Journal of Logic, Language and Information* 20:329–342.

Scott, Dana, and Michael Rabin. 1959. Finite automata and their decision problems. *IBM Journal of Research and Development* 5:114–125.