Strict Locality in Syntax

Kenneth Hanson

Stony Brook University

March 4, 2023

Introduction (1)

Linguistic patterns belong to very simple classes of formal languages, known as subregular languages (Heinz 2018; Graf 2022a).

The strictly local (SL) languages are used to model finitely bounded dependencies.

- In phonology: local phonotactics
- In syntax: lexical (category) selection

Introduction (2)

This presentation: syntax contains more SL patterns than is usually thought, covering essentially the full range of what is possible.

I use a formalism based on command strings (c-strings) in order to enable direct comparisons with phonology, and visualizations using finite state automata.

- Lexical selection – branching and looping paths
- Functional hierarchies – linear order, optionality
- Adjunct ordering – linear order, optionality, iteration

Proposal: SL computations are the basis for linguistic structure building across domains.

Overview

1. Introduction to SL
    - Examples from phonology
2. Generalizing SL to trees using c-strings
    - Dependency trees
    - C-string
3. SL in syntax:
    - Lexical selection
    - Functional hierarchies
    - Adjunct ordering
4. Beyond local dependencies

Strictly Local Languages (1)

Defining characteristic: a string is well-formed if all of its substrings (of some fixed length) are well-formed

SL-k: SL for substrings of length *k*

Example: CV Alternation (SL-2)

$\Sigma = \{C, V\}$ k = 2   G = {C, V, CV, VC, C, V}

	Word	Substrings (k=2)
✓	$CVCVC$	C, CV, VC, CV, VC, C
✓	VCV	V, VC, CV, V
✗	$CVCCV$	$C, CV, VC, **CC**, CV, V$
✗	$VCVV$	$V, VC, CV, **VV**, V$

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$	s	j	u	n	k	a	n	$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\boxed{\$ \quad s} \quad j \quad u \quad n \quad k \quad a \quad n \quad \$$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$ \boxed{s \quad j} $ u n k a n $

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$ \quad s \quad \boxed{j \quad u} \quad n \quad k \quad a \quad n \quad $

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\$ \quad s \quad j \quad \boxed{u \quad n} \quad k \quad a \quad n \quad \$$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\$ \quad s \quad j \quad u \quad \boxed{n \quad k} \quad a \quad n \quad \$$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\$ \quad s \quad j \quad u \quad n \quad \boxed{k \quad a} \quad n \quad \$$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\$ \quad s \quad j \quad u \quad n \quad k \quad \boxed{a \quad n} \quad \$$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

Strictly Local Languages (2)

Example: Japanese phonotactics (SL-2)

Syllable template: (C) (j) V (N)

Example words: aoi, kotowaza, sjunkan

$$\Sigma = \{C, j, V, N\} \qquad k = 2$$

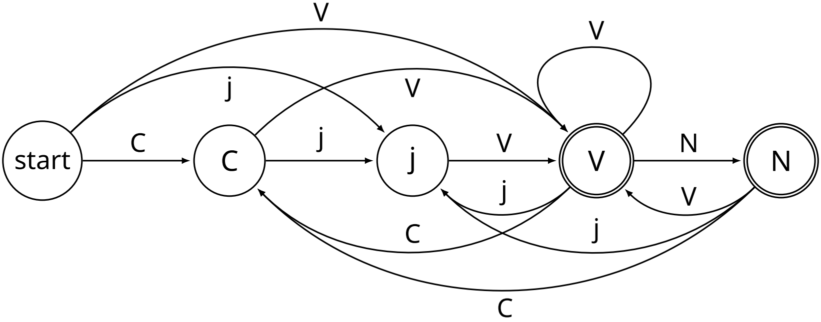
$$G = \left\{ \begin{array}{lllll} \$C & & & VC & NC \\ \$j & Cj & & Vj & Nj \\ \$V & CV & jV & VV & NV \\ & & & VN & \\ & & & V\$ & N\$ \end{array} \right\}$$

$$\$ \quad s \quad j \quad u \quad n \quad k \quad a \quad \boxed{n \quad \$}$$

Note: geminates are omitted. The grammar with geminates is also SL-2.

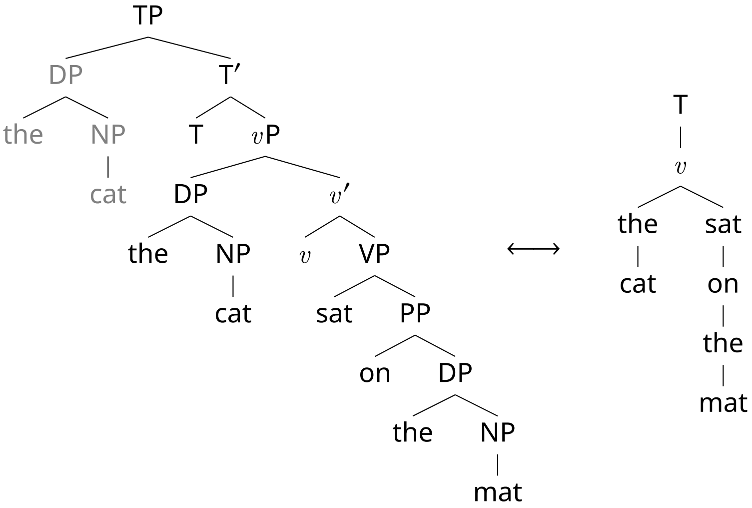
Strictly Local Languages (3)

We can represent an SL grammar visually using a finite-state automaton (FSA).

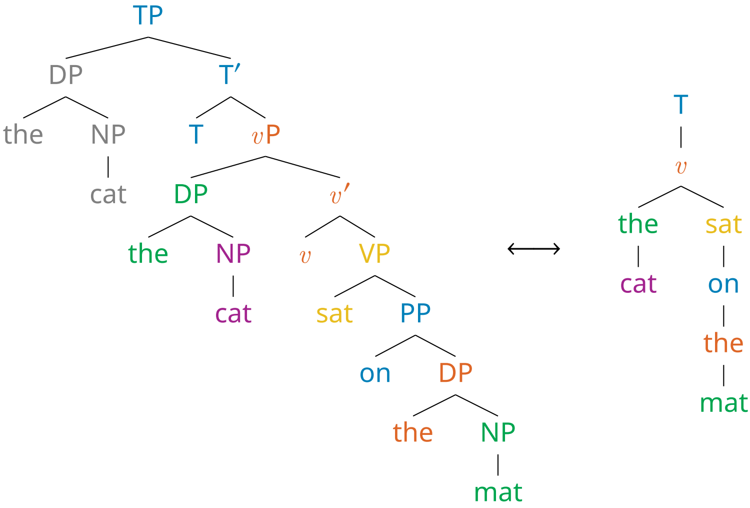


SL is a subclass of the languages expressible by FSAs.

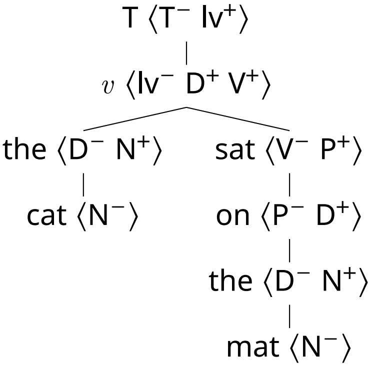
Dependency Trees



Dependency Trees

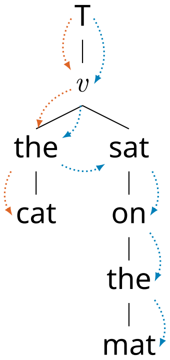


Dependency Trees



Note: the feature system is based on Minimalist Grammar (Stabler 1997).

C-Strings



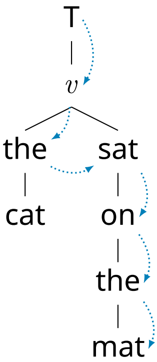
$\longrightarrow$

T · v · the · sat · on · the · mat

T · v · the · cat

See Graf and Shafiei (2019) for details.

Lexical Selection (1)



$\longrightarrow$

T · v · the · sat · on · the · mat

T	$\langle \mathrm{T}^- \ \mathrm{lv}^+ \rangle$
v	$\langle \mathrm{lv}^- \ \mathrm{D}^+ \ \mathrm{V}^+ \rangle$
the	$\langle \mathrm{D}^- \ \mathrm{N}^+ \rangle$
sat	$\langle \mathrm{V}^- \ \mathrm{P}^+ \rangle$
on	$\langle \mathrm{P}^- \ \mathrm{D}^+ \rangle$
the	$\langle \mathrm{D}^- \ \mathrm{N}^+ \rangle$
mat	$\langle \mathrm{N}^- \rangle$

Lexical Selection (2)

What the SL grammar looks like

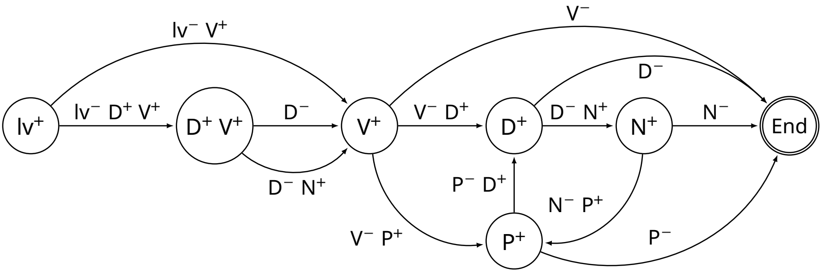
$\Sigma = \{\langle T^- \; lv^+\rangle, \langle lv^- \; V^+\rangle, \langle lv^- \; D^+ \; V^+\rangle, \langle V^-\rangle, \langle V^- \; D^+\rangle, \langle V^- \; P^+\rangle, \ldots\}$

$k = 3$

$$G = \left\{ \begin{array}{ccc} \cdots & \langle V^- \; D^+\rangle & \langle D^-\rangle \\ \cdots & \langle V^- \; D^+\rangle & \langle D^- \; N^+\rangle \\ \cdots & \langle D^- \; N^+\rangle & \langle N^-\rangle \\ \cdots & \langle D^- \; N^+\rangle & \langle N^- \; P^+\rangle \\ \cdots & \langle lv^- \; D^+ \; V^+\rangle & \langle D^- \; N^+\rangle \\ \langle lv^- \; D^+ \; V^+\rangle & \langle D^- \; N^+\rangle & \langle V^- \; D^+\rangle \\ \vdots & \vdots & \vdots \end{array} \right\}$$

Lexical Selection (3)

FSA Representation



Functional Hierarchies (1)

Example: English clausal hierarchy

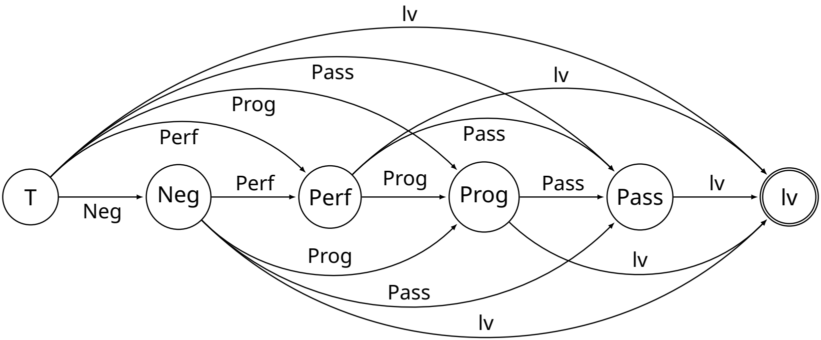
T < (Neg) < (Perf) < (Prog) < (Pass) < v < V

Ex. "He might$_{\text{T}}$ not$_{\text{Neg}}$ have$_{\text{Perf}}$ been$_{\text{Prog}}$ being$_{\text{Pass}}$ watched."

This is also SL-2!

$$G = \left\{ \begin{array}{lllll} \text{T Neg} & & & & \\ \text{T Perf} & \text{Neg Perf} & & & \\ \text{T Prog} & \text{Neg Prog} & \text{Perf Prog} & & \\ \text{T Pass} & \text{Neg Pass} & \text{Perf Pass} & \text{Prog Pass} & \\ \text{T lv} & \text{Neg lv} & \text{Perf lv} & \text{Prog lv} & \text{Pass lv} \end{array} \right\}$$

Functional Hierarchies (2)



Adjunct Ordering (1)

Adjectives and adverbs often have a preferred order.

1. opinion
2. size
3. shape
4. age
5. color
6. origin
7. material
8. purpose

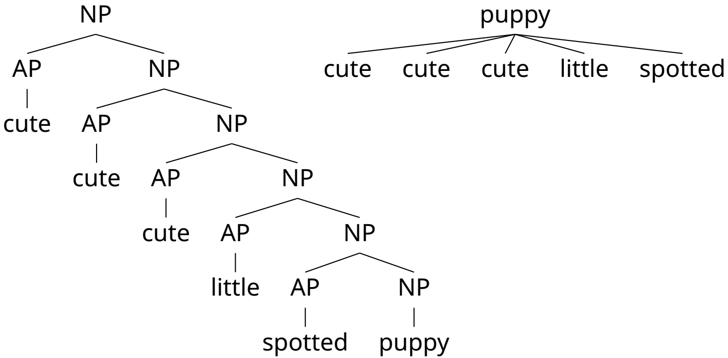
✓ *cute little spotted puppy*
? *little cute spotted puppy*
? *cute spotted little puppy*
?? *little spotted cute puppy*

Items in the same group can be iterated.

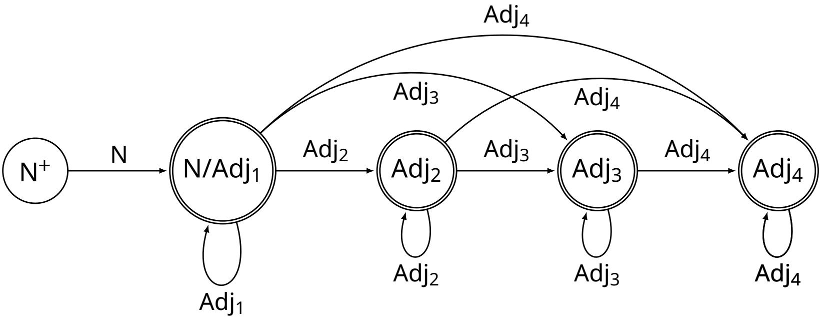
✓ *cute cute cute little spotted puppy*

Adjunct Ordering (2)

PS tree and dependency tree for "cute cute cute little spotted puppy"

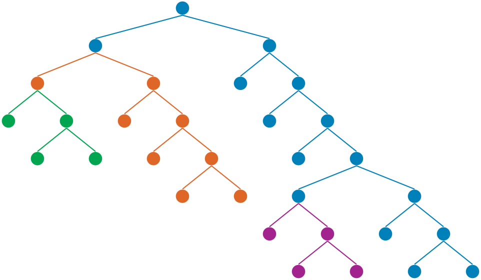


Adjunct Ordering (3)



Which c-strings do we use?

Answer: those that trace the complement spine of the tree, or of a subtree. See Graf and De Santo (2019) for details.



What about long-distance phenomena?

Most long-distance phonological dependencies are in the class tier-based strictly local (TSL), a generalization of SL in which non-salient items are ignored (Heinz 2018).

Most long-distance syntactic phenomena are TSL (or a close variant of TSL) over trees.

- Movement (Graf 2022b)
- Case (Vu et al. 2019)
- Anaphora and NPI licensing (Graf and Shafiei 2019)
- Agreement (work in progress)

Conclusion

Functional hierarchies and adjunct hierarchies are unsurprising from a computational perspective — they are just further examples of SL patterns.

Syntax and phonology are very similar in their computational properties, as highlighted by the c-string perspective.

SL computations are a good candidate for the basis of linguistic structure building.

Thank you!

Acknowledgments:

- This work is partly funding by Thomas Graf's NSF CAREER Award: *Abstract Universals in (Morpho)Syntax: Computational Characterizations and Empirical Implications*.
- Thanks to Thomas Graf and the Stony Brook Mathematical Linguistics Reading Group for comments and discussion.

References

Graf, Thomas (2022a). "Subregular linguistics: bridging theoretical linguistics and formal grammar". In: *Theoretical Linguistics* 48.3-4. DOI: `doi:10.1515/tl-2022-2037`.

— (2022b). "Typological implications of tier-based strictly local movement". In: *Proceedings of the Society for Computation in Linguistics 2022*.

Graf, Thomas and Aniello De Santo (2019). "Sensing Tree Automata as a Model of Syntactic Dependencies". In: *Proceedings of the 16th Meeting on the Mathematics of Language*. Toronto, Canada: Association for Computational Linguistics.

Graf, Thomas and Nazila Shafiei (2019). "C-command dependencies as TSL string constraints". In: *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*.

Heinz, Jeffrey (2018). "The computational nature of phonological generalizations". In: *Phonological Typology, Phonetics and Phonology*.

Stabler, Edward P. (1997). "Derivational minimalism". In: *Logical Aspects of Computational Linguistics*. Springer.

Vu, Mai Ha et al. (2019). "Case assignment in TSL syntax: A case study". In: *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*.