

Algorithms + Data Structures = Speech Production Programs

Mayfest 2026

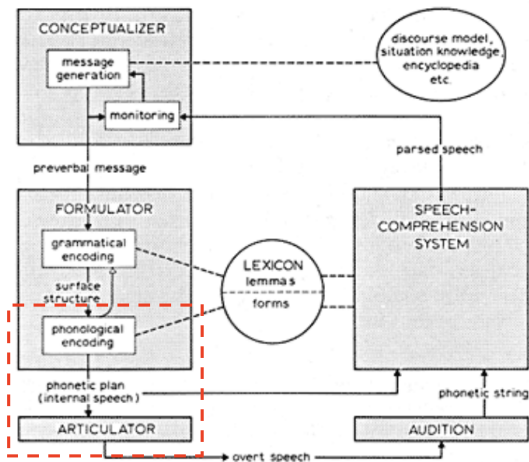
Scott Nelson

University of Illinois Urbana-Champaign

May 30, 2026

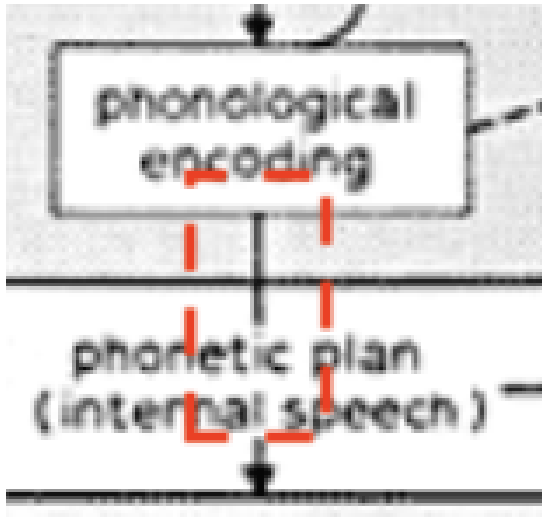


The Finish Line...



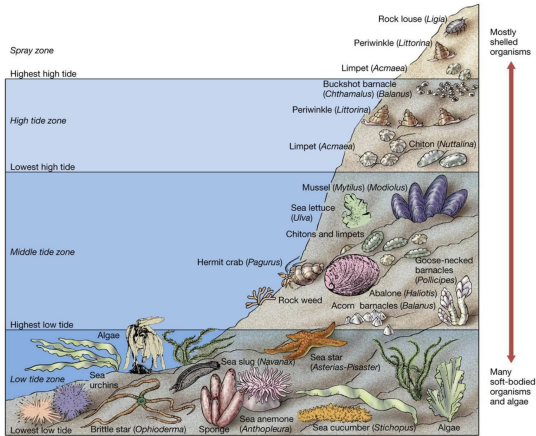
(Levelt, 1993)

The Finish Line...



(Levelt, 1993)

Phonology? Phonetics?



(Scobbie, 2005)

Rules & Representations

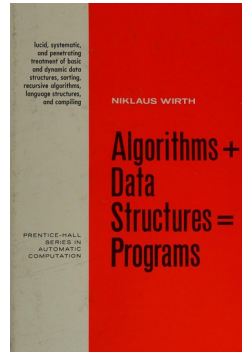
“If the representations are right, then the rules will follow.”



McCarthy (1988)

Program Construction

“Programs, after all, are concrete formulations of abstract algorithms based on particular representations and structures of data [...] Decisions about structuring data cannot be made without knowledge of the algorithms applied to the data and [...] the structure and choice of algorithms often strongly depend on the structure of the underlying data.”



Wirth (1978, pp. xii–xiii)

Overview

- ▶ In this talk I will abstract and share a high level perspective on the “speech production program”.
- ▶ Most of the focus will be at the level of a phonological production grammar and how it might interface with long-term memory representations and motor-planning.
- ▶ I will assume the phonological grammar operates over symbolic categories and explore the consequences for our understanding of phonological/phonetic production when making this assumption.

- ▶ The talk has two major parts:
 - ① How should we structure the speech production function?
 - ▶ i.e., Algorithms(ish).
 - ② Do we need to encode gestural information directly?
 - ▶ i.e., Data Structures.

Why Abstraction?

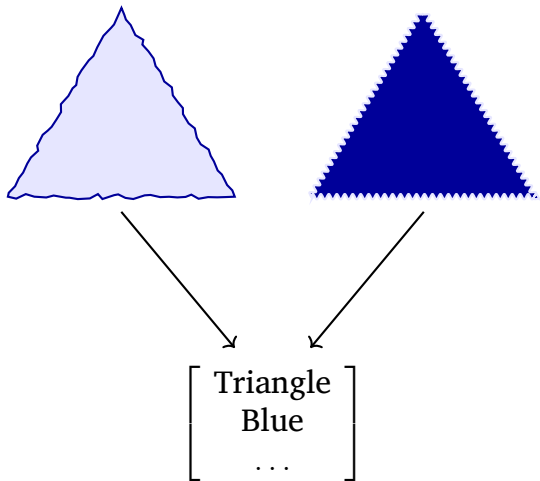
*One of the aims and advantages of abstraction is to make connections between different situations that might previously have seemed very different. It might seem that abstraction takes us further away from “real” situations. This is superficially true, but at the same time **abstraction enables us to make connections between situations that are further apart from one another.***



Cheng (2022) - *The joy of abstraction: An exploration of math, category theory, and life*

Abstraction helps find *necessary* conditions

- ▶ What structural properties *must* representations have in order to support speech?



Describing Grammatical Architecture with Typed Functions

Key Idea

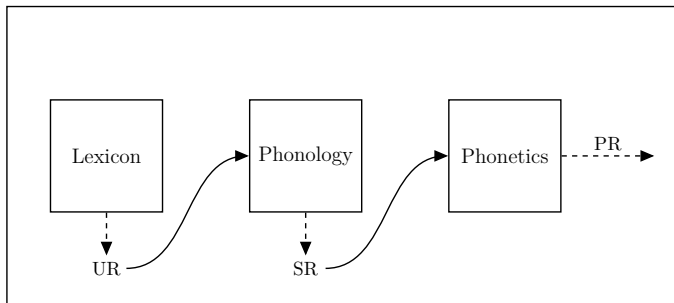
I show how to use function types to rethink about the way that certain types of information interact within a system. This view reveals that qualitative structure rather than quantitative implementations can explain a good chunk of phenomena. From this, it follows that the symbolic vs. dynamic distinction can be subtle and may rely at times on auxiliary assumptions about structure.

Computational *Structure*

- ▶ **Typed functions**, a generalization of the λ -calculus (a universal model of computation) can bring together two definitions of computation.
- ▶ $A :: B \rightarrow C$ describes a function A that maps B type things to C type things.
- ▶ An **information processing device** (A) takes **some information** (B) and processes it into **something new** (C).

Sipser (2013); Marr (1982); Pierce (2002); Church (1932, 1933)

Modular Feed-Forward Model



- The common view of the production process in generative linguistics is often referred to as the **modular feed-forward model**.

Pierrehumbert (2002); Bermúdez-Otero (2007); Kenstowicz (2010)

Thinking in Functions

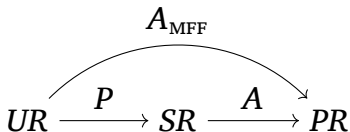
- ▶ Phonology is a function that maps *UR*'s to *SR*'s
($P :: UR \rightarrow SR$)
- ▶ Phonetics is a function that maps *SR*'s to *PR*'s
($A_{\text{MFF}} :: SR \rightarrow PR$)

Roark and Sproat (2007), Heinz (2018)

Modular Feed-Forward Model Redux

- ▶ Since phonology has the type $UR \rightarrow SR$ and phonetics has the type $SR \rightarrow PR$, the Modular Feed-Forward Model is a composed function $MFF :: UR \rightarrow PR$.

- ▶ $A_{MFF}(P(UR)) = PR$



Moving beyond the MFF Model

- ▶ Some problems for the MFF Model:
 - ▶ Incomplete Neutralization
 - ▶ Variation in Homophone Duration
 - ▶ Word Specific Effects
 - ▶ ...
- ▶ Solution from Dynamics (Gafos and Benus, 2006):
 - ▶ ...it is both necessary and promising to do away with the metaphor of precedence between the qualitative phonology and the quantitative phonetics, without losing sight of the essential distinction between the two...

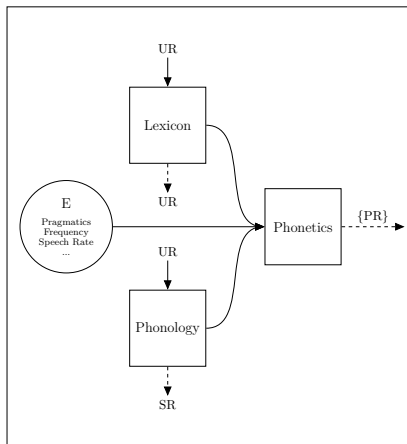
Port and Leary (2005); Gahl (2008); Pierrehumbert (2002)

An Abstract Solution

What we continue to stress in this paper is that language production involves the interaction of lexical, phonological, and extra-grammatical factors which the modular feedforward model fails to capture. Since this idea is able to be expressed using different types of mathematical formalisms, we believe that this idea is not a property of the specific mathematical implementation, but rather a property of the high-level architecture (a “model” in our terms).

Nelson and Heinz (2026) - *The Blueprint Model of Production*

The Blueprint Model of Production



Nelson and Heinz (2026)

The Blueprint Model of Production

- ▶ The inputs to the phonetics function are the lexicon and phonology functions, as well as extra-grammatical information that influences production.
 - ▶ The lexicon is a function with type $L :: UR \rightarrow UR$.
 - ▶ Phonology is *still* a function with type $P :: UR \rightarrow SR$.
 - ▶ E is a cover type for extra-grammatical information.
- ▶ When a function takes a function as an input it is called a **higher-order function**.
- ▶ The phonetics function in the Blueprint Model of Production has type $A_{BP} :: L \rightarrow P \rightarrow E \rightarrow \{PR\}$.

- ▶ Simulations show that the Blueprint Model of Production can produce systematic gradient phonetic effects with discrete phonological knowledge.
 - ▶ Final Devoicing in German
 - ▶ Tonal Near Merger in Cantonese
 - ▶ Epenthesis in Lebanese Arabic
 - ▶ Homophone duration variation in English

Nelson and Heinz (2026)

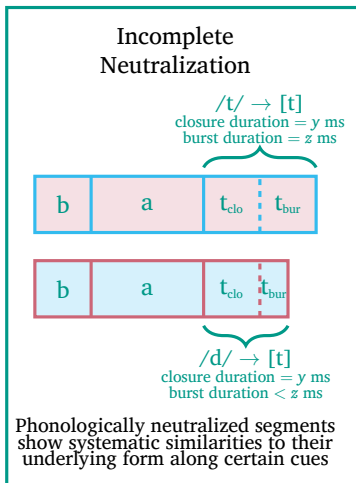
German Final Devoicing

- ▶ German final devoicing is a classic example of **neutralization** where a lexical contrast is eliminated due to a (morpho)phonological process.

/bad/	→	[bat]	‘bath’
/bad + en/	→	[baden]	‘to bathe’
/bat/	→	[bat]	‘ask’
/bat + en/	→	[baten]	‘asked’

(Dinnsen and Garcia-Zamor, 1971)

Incomplete Neutralization



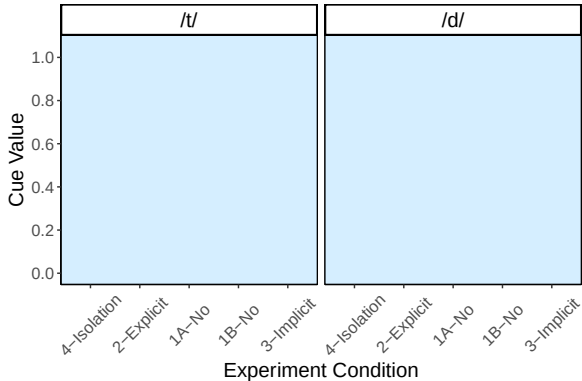
(Port et al., 1981; Port and O'Dell, 1985, among **many** others)

Simulating Empirical Phenomena

*“Computational-level theories of capacities are not quantitative but qualitative models: They reveal the internal formal structure of a system, or the **invariants** that allow it to exercise a particular capacity across situations, conditions.”*

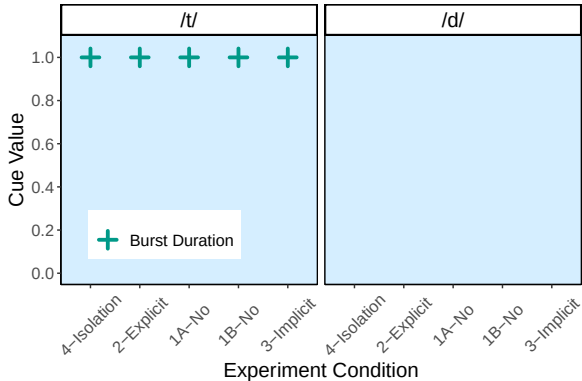
van Rooij and Baggio (2021); Simon (1990); Newell and Simon (1975)

Simulating Incomplete Neutralization with the Blueprint Model of Production



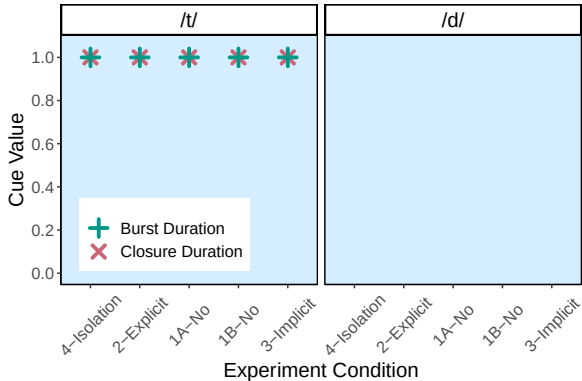
Port and Crawford (1989)

Simulating Incomplete Neutralization with the Blueprint Model of Production



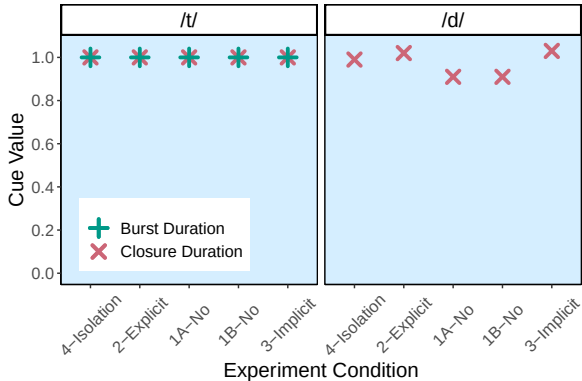
Port and Crawford (1989)

Simulating Incomplete Neutralization with the Blueprint Model of Production



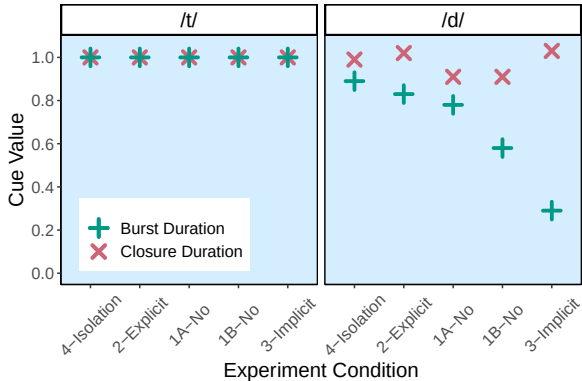
Port and Crawford (1989)

Simulating Incomplete Neutralization with the Blueprint Model of Production



Port and Crawford (1989)

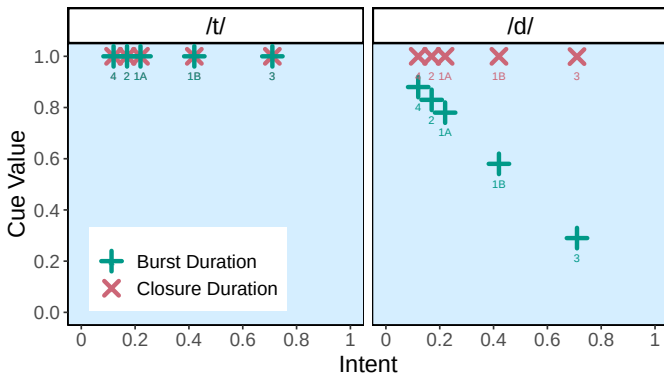
Simulating Incomplete Neutralization with the Blueprint Model of Production



Port and Crawford (1989)

Simulating Incomplete Neutralization with the Blueprint Model of Production

Blueprint Model Prediction for German Final Devoicing



► $c = c_{UR} \times i^{\alpha} + c_{SR} \times (1 - \alpha)$

Reconsidering Gestural Phenomena

▶ Gestural Hiding

- ▶ *perfect memory, must be, ground pressure*
- ▶ Magnitude of alveolar gesture **similar** between “deleted” and present forms.
- ▶ Weak evidence for output being a blend of lexical + phonological forms (with deletion rule)

▶ Gestural Reduction

- ▶ *late calls*
- ▶ Magnitude of alveolar gesture **reduced** between “deleted” and present forms.
- ▶ Stronger evidence for output being a blend of lexical + phonological forms (with deletion rule)

Browman and Goldstein (1990); Beckman (1996); Nolan (1992)

Reconsidering Gestural Phenomena

- ▶ Hiding and reduction have been used to argue against a theory which includes a phonological grammar.
- ▶ Every gesture specified in the lexical representation must be present in the output.
 - ▶ Cannot be deleted but can be reduced spatially or temporally

Takeaway

If we take the idea of the Blueprint Model seriously, we would expect to find traces of the lexical representation even in non-neutralizing contexts. This behavior is therefore not evidence against symbolic grammars, but rather symbolic grammars + the modular feedforward view of the world.

Browman and Goldstein (1990); Beckman (1996); Nolan (1992)

Local Summary

- ▶ The assumed structure of a system influences how data are interpreted.
 - ▶ New structure = possible new interpretation of old data
- ▶ The Blueprint Model of Production shows that we can tweak one parameter (structure of interface) and keep another unchanged (discrete phonological knowledge) to explain thought-to-be-problematic phenomena.
- ▶ Again, abstraction allows us to make connections in ways that may not seem obvious.
 - ▶ Philosophical vs. empirical justification for one type of theory over the other?
 - ▶ Multiple realizability

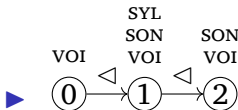
Formally Comparing Gestural Representations with Symbolic Strings

- ▶ Mathematical logic and model theory provide a tool for determining the expressivity of different types of phonological knowledge.
- ▶ More powerful logic $\rightarrow$ more expressive pattern/more computational power needed.

Rogers et al. (2010, 2013); Strother-Garcia (2019); Oakden (2020)

Model Theory - Structure Building

- ▶ A *model signature* determines what properties elements in a structure can have and how the elements are related.
 - ▶ Basic “feature string” signature: $\langle \triangleleft, \{R_f \mid f \in \mathcal{F}\} \rangle$
- ▶ A *structure* is the result of denoting which properties of a signature hold for a given domain $\mathcal{D}$.



- ▶ All structures that satisfy a set of constraints $T = \phi_1, \phi_2, \dots$ are the *model* of T .
 - ▶ $\phi_1 := \exists x[\text{SYLLABIC}(x)]$

Representational Example - *dish*

$\langle D = \{0, 1, 2\},$

$\triangleleft = \{(0, 1), (1, 2)\},$

VOICE = $\{0, 1\},$

SONORANT = $\{1\},$

CONSONANT = $\{0, 2\},$

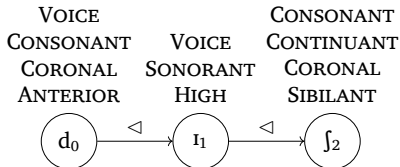
CONTINUANT = $\{2\},$

CORONAL = $\{0, 2\},$

ANTERIOR = $\{0\},$

SIBILANT = $\{2\},$

HIGH = $\{1\}, \rangle$



Constraint Example - *dish*

$$\phi_1 \stackrel{\text{def}}{=} \forall x [(\neg \text{SONORANT}(x) \wedge \neg \exists y [x \triangleleft y]) \rightarrow \neg \text{VOICE}(x)]$$

$$\phi_2 \stackrel{\text{def}}{=} \forall x [\text{CONSONANT}(x) \rightarrow \exists y [\neg \text{CONSONANT}(y) \wedge x \triangleleft y]]$$

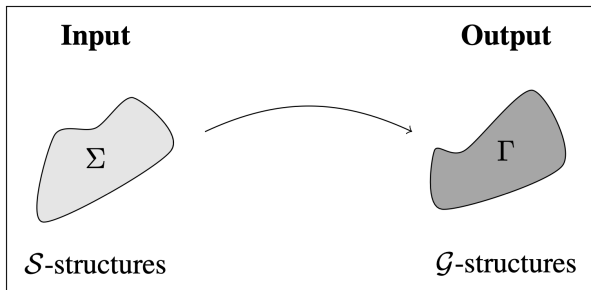
$$\phi_3 \stackrel{\text{def}}{=} \exists x [\neg \text{CONSONANT}(x)]$$

- ▶ The model for *dish*...
 - ▶ **satisfies** ϕ_1 (don't end in a voiced obstruent)
 - ▶ **does not satisfy** ϕ_2 (all consonants must be directly followed by a non-consonant)
 - ▶ **satisfies** ϕ_3 (contain at least one non-consonant)

Model Theory - Logical Transductions

- ▶ A logical transduction is a function $f : \Sigma \rightarrow \Gamma$ defined by a program $\rho = \{\phi_1, \dots, \phi_n\}$.
- ▶ Each function ϕ_i within the program defines how to represent a property within the model signature Γ defined using only properties from the model signature Σ .
- ▶ These can define string-to-string maps...
 - ▶ $\phi_b(x) := a(x) \wedge \exists y, z[y \triangleleft x \triangleleft z \wedge c(y) \wedge d(z)]$
- ▶ Or mappings between two different representational languages...
 - ▶ $\phi_{\text{voi}}(x) := d(x) \vee b(x) \vee g(x) \vee \dots$

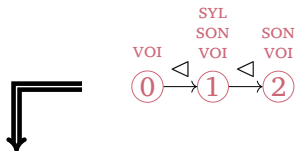
Model Theory - Relating Representations



- ▶ We can **interpret** a structure $\mathcal{G}$ in signature Γ as a structure $\mathcal{S}$ in signature Σ .
- ▶ This gives us a **function** from $\mathcal{S}$ -structures to $\mathcal{G}$ -structures.

Czarnecki (2025)

Model Theory - Relating Representations



$$\phi_{\text{voi}}(x) \stackrel{\text{def}}{=} d(x) \vee n(x) \vee a(x)$$

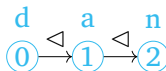
$$\phi_{\text{son}}(x) \stackrel{\text{def}}{=} n(x) \vee a(x)$$

$$\phi_{\text{syl}}(x) \stackrel{\text{def}}{=} a(x)$$

$$\phi_d(x) \stackrel{\text{def}}{=} \text{voi}(x) \wedge \neg \text{son}(x) \wedge \neg \text{syl}(x)$$

$$\phi_n(x) \stackrel{\text{def}}{=} \text{voi}(x) \wedge \text{son}(x) \wedge \neg \text{syl}(x)$$

$$\phi_a(x) \stackrel{\text{def}}{=} \text{voi}(x) \wedge \text{son}(x) \wedge \text{syl}(x)$$



Nelson (2022)

Definition:

An interpretation $K : U \rightarrow V$ gives us a construction of an internal model $\tilde{K}(\mathcal{M})$ of U from a model $\mathcal{M}$ of V . We find that U and V are bi-interpretable iff, there are interpretations $K : U \rightarrow V$ and $M : V \rightarrow U$ and formulas F and G such that, for all models $\mathcal{M}$ of V , the formula F defines an isomorphism between $\mathcal{M}$ and $\tilde{M}\tilde{K}(\mathcal{M})$, and, for all models $\mathcal{N}$ of U , the formula G defines an isomorphism between $\mathcal{N}$ and $\tilde{K}\tilde{M}(\mathcal{N})$.

Friedman and Visser (2014); Strother-Garcia (2019); Oakden (2020)

Notational Equivalence

- Conditions for notational equivalence:
 - 1 Two models do not differ in their empirical predictions.
 - 2 Two models represent the same set of abstract properties, differing only superficially.
 - 3 Two models are QF-bi-interpretable.

Definition:

A word model M_1 is $\mathcal{L}$ -interpretable in terms of another, M_2 , if one can write a transduction from M_1 to M_2 using logic $\mathcal{L}$. If M_1 is $\mathcal{L}$ -interpretable in terms of M_2 and vice versa, then we say the two are $\mathcal{L}$ -bi-interpretable.

Fromkin (2000); Strother-Garcia (2019); Oakden (2020)

Examples of bi-interpretability in phonology

- ▶ Hierarchical syllable trees, dot-encoded string representations, prosodic feature strings
- ▶ Classic autosegmental representations and Q-theoretic representations
- ▶ Segmental strings and segmental feature strings
- ▶ Tonal feature geometries with and without register nodes
- ▶ Unified and v-feature segmental feature geometries
- ▶ **Segmental strings and Articulatory Phonology coupling graphs???**

Strother-Garcia (2019); Oakden (2020); Jardine et al. (2021); Nelson (2022, 2024); Danis (2025)

Is this enough for notational equivalence?

- ▶ Weak vs. strong generative capacity
 - ▶ bi-interpretability = weak
 - ▶ bi-interpretability + natural class preserving = strong
- ▶ This only makes sense when comparing two theories with similar base assumptions (purely symbolic computation)

Miller (1999); Danis (2025)

An alternative perspective

- ▶ Two phonological representations are equivalent if the logic needed to translate between them is less expressive than the logic needed to compute the phonological grammar.
- ▶ I'm going to assume first-order logic is the most we need for phonology, but it's almost certainly a more specific fragment of FO.

Heinz (2018); Lambert (2022); Idsardi (2025)

Gestural Representations

- ▶ Articulatory Phonology is (primarily) a mono-stratal theory of phonology based around gestural representations.
- ▶ **Gestures** are tasks/goals that are defined via **tract variables**:
 - ▶ **Constriction Location**: y-dimension; labeled with landmarks like “alveolar”, “velar”, ...
 - ▶ **Constriction Degree**: x-dimension; labeled with degrees like “closed”, “critical”, ...

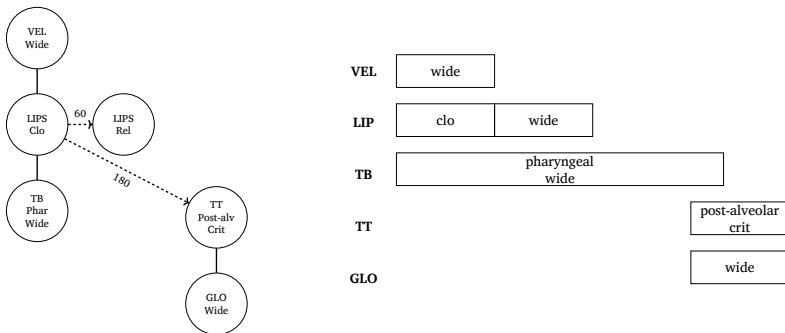
Browman and Goldstein (1986, 1995, *inter alia*)

Coupling Graphs as Lexical Representations

*“In this model, planning oscillators associated with the set of gestures in a given utterance are coupled in a pairwise, bidirectional manner specified in a coupling graph (or structure) that is part of **the lexical specification of a word**” (p. 38).*

Nam (2007)

Coupling Graphs as Lexical Representations



- ▶ Representations for *mash*
 - ▶ Coupling Graph (left)
 - ▶ Gestural score (right; automatically determined by left)

Nam (2007)

Coupling Graph Model Signature

Relation	Label
$\diamond$	In-phase
$\triangleleft_{180}$	Anti-phase
$\triangleleft_{60}$	Abutting
$\triangleleft_{30}$	Eccentric

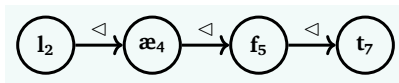
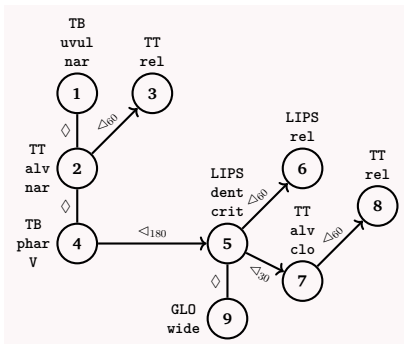
- 4 binary relations based on some phase relations in Articulatory Phonology

Coupling Graph Model Signature

Relation	Label	Relation	Label
LIPS	Labial Articulator	rel	Constriction Degree: release
TT	Tongue Tip Articulator	pro	Constriction Location: protruded
TB	Tongue Body Articulator	dent	Constriction Location: dental
VEL	Velum Articulator	alv	Constriction Location: alveolar
GLO	Glottis Articulator	palv	Constriction Location: postalveolar
clo	Constriction Degree: closed	pal	Constriction Location: palatal
crit	Constriction Degree: critical	vel	Constriction Location: velar
nar	Constriction Degree: narrow	uvul	Constriction Location: uvular
V	Constriction Degree: vowel	phar	Constriction Location: pharyngeal
wide	Constriction Degree: wide		

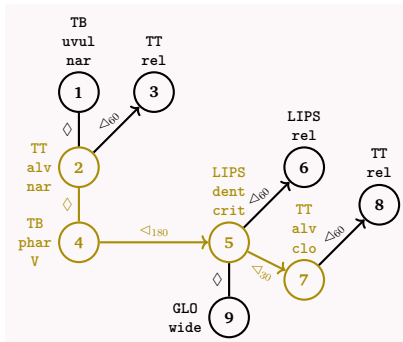
► Unary labeling relations.

Structures for *laughed*



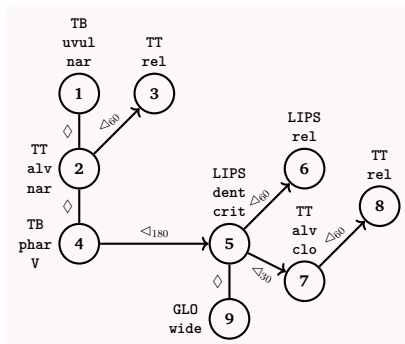
Translating between Structures

The key insight that allows for determining a string from a coupling graph is the notion of a “spine” which is made of what have been called “head gestures” (Gafos, 2002; Smith, 2018).

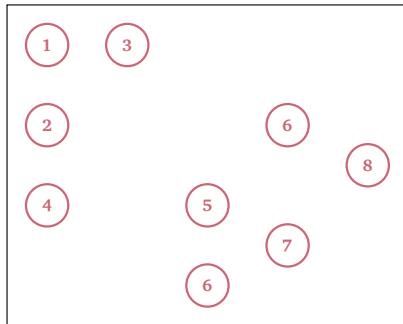


Translating from coupling graph to string

Input



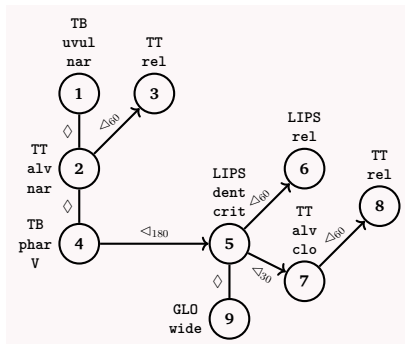
Workspace



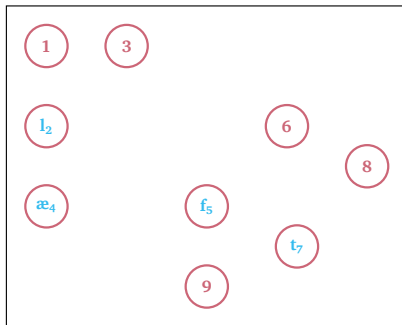
► $C := \{1\}$

Translating from coupling graph to string

Input



Workspace



$$\varphi_1(x) \stackrel{\text{def}}{=} \text{TT}(x) \wedge \text{alv}(x) \wedge \text{nar}(x) \wedge \exists y[x \oslash y \wedge \text{TB}(x) \wedge \text{uvul}(x) \wedge \text{nar}(x)]$$

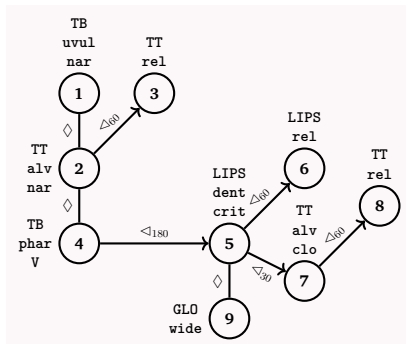
$$\varphi_{\text{æ}}(x) \stackrel{\text{def}}{=} \text{TB}(x) \wedge \text{phar}(x) \wedge V(x)$$

$$\varphi_f(x) \stackrel{\text{def}}{=} \text{LIPS}(x) \wedge \text{dent}(x) \wedge \exists y[x \oslash y \wedge \text{GLO}(y)]$$

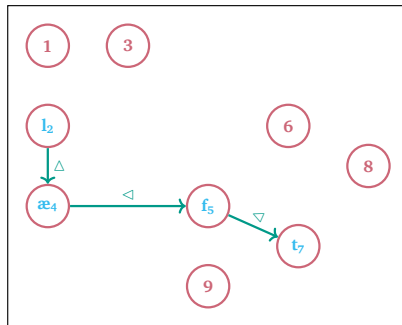
$$\varphi_t(x) \stackrel{\text{def}}{=} \text{TT}(x) \wedge \text{alv}(x) \wedge \text{clo}(x) \wedge \exists y[x \oslash y \wedge \text{GLO}(y)]$$

Translating from coupling graph to string

Input



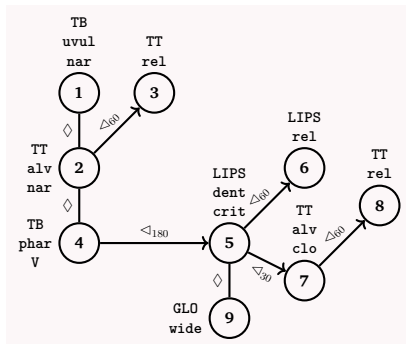
Workspace



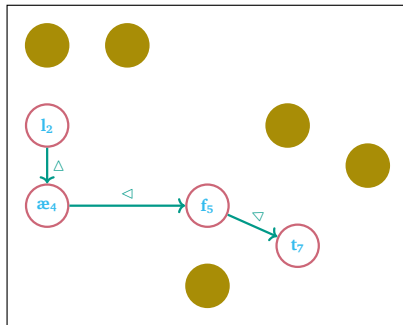
$$\varphi \triangleleft (x, y) \stackrel{\text{def}}{=} (x \triangleleft_{180} y) \vee (x \triangleleft_{30} y) \vee (x \triangleright y \wedge \forall(y) \wedge \neg \exists z [x \triangleleft_{180} z])$$

Translating from coupling graph to string

Input



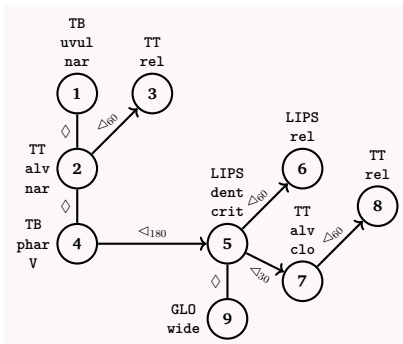
Workspace



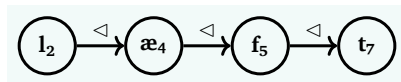
$$\varphi_{\text{license}}(x) \stackrel{\text{def}}{=} \neg \text{rel}(x) \wedge \neg \text{GLO}(x) \wedge \neg \text{VEL}(x) \neg \text{SecArc}(x)$$

Translating from coupling graph to string

Input



Output



Expansion

- ▶ Going from coupling graph to string removes information. What happens when we have to expand the representation and add more information by going from a string to a coupling graph?

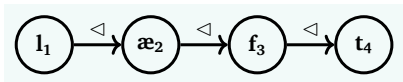
Expansion

- ▶ Here, there is no issue. In fact, the TADA software which implements the Articulatory Phonology system has a program called GEST which turns strings into coupling graphs.
- ▶ GEST is written in Perl which is an imperative programming language which lists the steps that must be taken to solve the problem.

The logical transductions are equivalent to logical programs which provide declarative statements on the specs of the program. They tell us what properties must hold of any program solving the problem.

Translating from string to coupling graph

Input



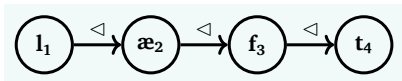
Workspace



► $C := \{1, 2, 3, 4\}$

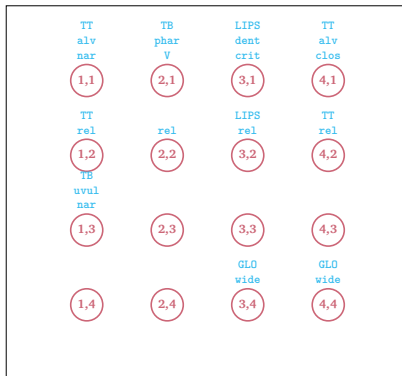
Translating from string to coupling graph

Input



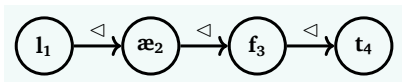
$$\begin{aligned}
 \varphi_{\text{LIPS}}^1(x) &:= f(x) & \varphi_{\text{phar}}^1 &:= \text{æ}(x) \\
 \varphi_{\text{LIPS}}^2(x) &:= \varphi_{\text{LIPS}}^1(x) & \varphi_{\text{uvul}}^3 &:= l(x) \\
 \varphi_{\text{TT}}^1(x) &:= t(x) \vee l(x) & \varphi_{\text{clos}}^1 &:= t(x) \\
 \varphi_{\text{TT}}^2(x) &:= \varphi_{\text{TT}}^1(x) & \varphi_{\text{crit}}^1 &:= f(x) \\
 \varphi_{\text{TB}}^1(x) &:= \text{æ}(x) & \varphi_{\text{v}}^1 &:= \text{æ}(x) \\
 \varphi_{\text{TB}}^3(x) &:= l(x) & \varphi_{\text{nar}}^1 &:= l(x) \\
 \varphi_{\text{GLO}}^4(x) &:= t(x) \vee f(x) & \varphi_{\text{nar}}^3 &:= l(x) \\
 \varphi_{\text{dent}}^1 &:= f(x) & \varphi_{\text{wide}}^4 &:= t(x) \vee f(x) \\
 \varphi_{\text{alv}}^1 &:= t(x) & &
 \end{aligned}$$

Workspace



Translating from string to coupling graph

Input



$$\varphi_{\Diamond}^{1,1}(x, y) := x \triangleleft y \wedge \text{æ}(y) \wedge \neg \text{æ}(x)$$

$$\varphi_{\Diamond}^{1,3}(x, y) := (x = y) \wedge \text{l}(x)$$

$$\varphi_{\Diamond}^{1,4}(x, y) := (x = y) \wedge \text{t}(x) \vee \text{f}(x)$$

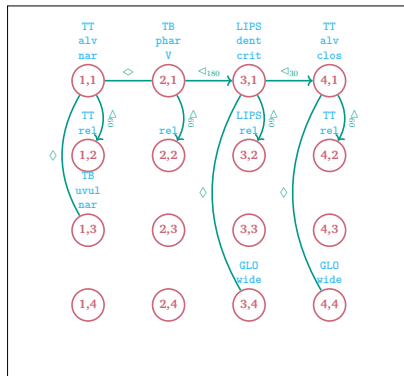
$$\varphi_{\triangleleft_{180}}^{1,1}(x, y) := x \triangleleft y \wedge \text{æ}(x) \wedge \neg \text{æ}(y)$$

$$\varphi_{\triangleleft_{60}}^{1,2}(x, y) := (x = y)$$

$$\varphi_{\triangleleft_{30}}^{1,1}(x, y) := \neg \text{æ}(x) \wedge \neg \text{æ}(y) \wedge$$

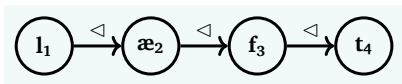
$$\exists z[z \triangleleft x \wedge \text{æ}(z)]$$

Workspace



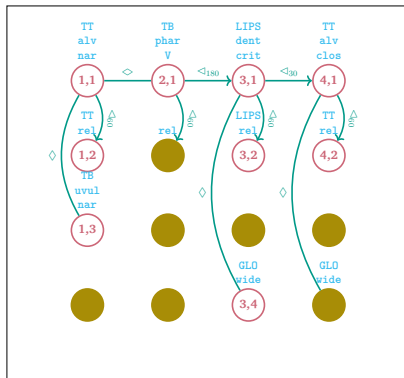
Translating from string to coupling graph

Input



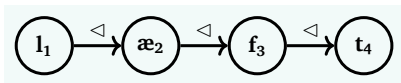
$\varphi_{\text{license}}^1(x) := \text{True}$
 $\varphi_{\text{license}}^2(x) := f(x) \vee t(x) \vee l(x)$
 $\varphi_{\text{license}}^3(x) := l(x)$
 $\varphi_{\text{license}}^4(x) := t(x) \vee f(x) \wedge \neg \exists y[y \triangleleft x \wedge f(y) \vee t(y)]$

Workspace

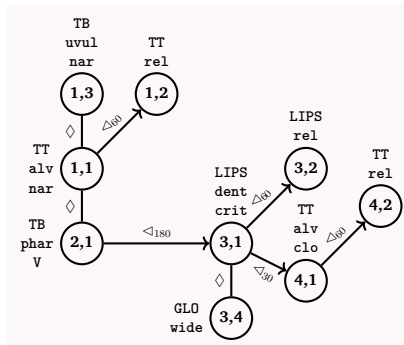


Translating from string to coupling graph

Input



Output



What now?

- ▶ “At present, there is no real consensus on how to deal with categorical alternations in AP” (Hall, 2010).

We can use the equations from this section to translate between constraints and transformations. These provide a way to embed these types of analyses directly into the language of Articulatory Phonology and vice versa.

- ▶ I'll use the example of the English past-tense alternation (specifically determining when epenthesis occurs).

Model Theory - Phonological Transformations

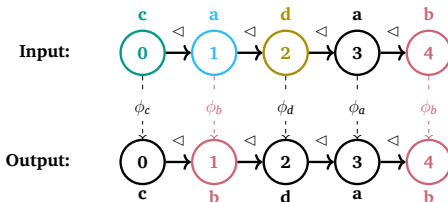
$$a \rightarrow b/c_d$$

$$\phi_a(x) \stackrel{\text{def}}{=} a(x) \wedge \neg \exists y, z [y \triangleleft x \triangleleft z \wedge c(y) \wedge d(z)]$$

$$\phi_b(x) \stackrel{\text{def}}{=} b(x) \vee (a(x) \wedge \exists y, z [y \triangleleft x \triangleleft z \wedge c(y) \wedge d(z)])$$

$$\phi_c(x) \stackrel{\text{def}}{=} c(x)$$

$$\phi_d(x) \stackrel{\text{def}}{=} d(x)$$



English Past Tense

vote	[vout]	voted	[vour ^{id}]
load	[loud]	loaded	[lour ^{id}]
tag	[tæg]	tagged	[tæg ^d]
sigh	[saɪ]	sighed	[saɪ ^d]
grease	[gɹis]	greased	[gɹis ^t]
back	[bæk]	backed	[bækt ^t]

Theoretical Accounts of the Past Tense Alternations

► Rules

$$\text{► } \emptyset \rightarrow \text{i} / \begin{bmatrix} -\text{son} \\ -\text{cont} \\ +\text{cor} \end{bmatrix} - \begin{bmatrix} -\text{son} \\ -\text{cont} \\ +\text{cor} \end{bmatrix}$$

$$\text{► } [-\text{son}] \rightarrow [-\text{voi}] / [-\text{voi}]_\#$$

► Optimality Theory

$$\text{► NOGEM, AGREE(VOICE) } \gg \text{ DEP(V) } \gg \text{ IDENT(VOICE)}$$

Baković (2005)

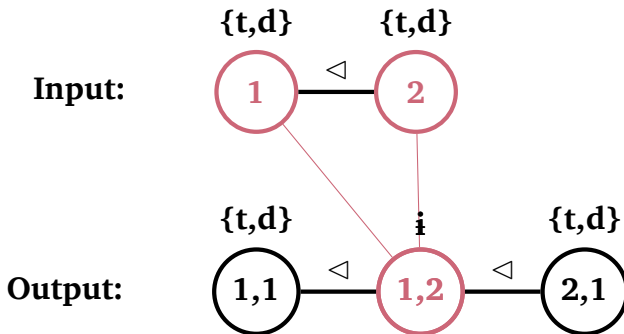
► Coupling Graphs

- Couple the TT closure gesture of the suffix coupling graph anti-phase to a stem-final vowel constriction gesture.
- Couple the TT closure gesture of the suffix eccentric phase to a stem-final non-coronal obstruent constriction gesture.
- Couple the TT closure gesture of the suffix eccentric phase to a stem-final coronal obstruent release gesture.

Goldstein (2011)

Rules and English Past Tense Epenthesis

$$\varphi_i^2(x) \stackrel{\text{def}}{=} (t(x) \vee d(x)) \wedge \exists y[x \triangleleft y \wedge t(y) \vee d(y)]$$



- ▶ “if there is a sequence of two coronals, then on the output insert a [i] between those coronals.”

OT and English Past Tense Epenthesis

- ▶ The constraint ranking essentially gets us an IF . . . THEN . . . ELSE statement.

If there is a sequence of two coronal plosives on the input (violating NOGEM), then changing the voicing valuation will not make a difference in harmony. The only way to salvage a candidate is by epenthesis due to the ranking.

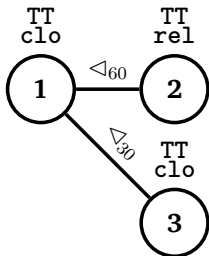
- ▶ $\varphi_i^2(\mathbf{x}) := \text{IF } \text{cor}(\mathbf{x}) \wedge \text{cor}(\mathbf{s}(\mathbf{x})) \text{ THEN true ELSE false}$
- ▶ In the FO language used previously, this is equivalent to the rule formula.

Bhaskar et al. (2020); Chandlee and Jardine (2021); Bhaskar et al. (2023)

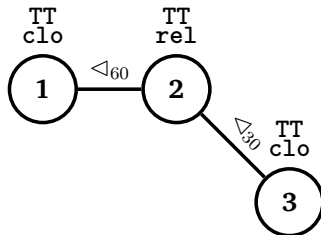
Dynamic Grammar and English Past Tense Epenthesis

$$\varphi_{\triangleleft_{30}}(x, y) \stackrel{\text{def}}{=} \text{TT}(y) \wedge \text{clo}(y) \wedge \exists z[z \triangleleft_{30} y \wedge \text{TT}(z) \wedge \text{clo}(z) \wedge z \triangleleft_{60} x]$$

Input:



Output:



- If we abstract just a little bit, the extension of the Dynamic Grammar includes pairs where, if a given substructure exists in an input graph, then the output graph changes the coupling structure such that epenthesis occurs.

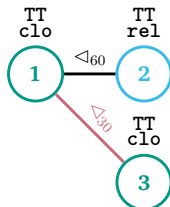
Dynamic Grammar and English Past Tense Epenthesis

$$\varphi_{\triangleleft}(x,y) \stackrel{\text{def}}{=} (x \triangleleft_{180} y) \vee (x \triangleleft_{30} y) \vee (x \diamond y \wedge \text{V}(y) \wedge \neg \exists z[x \triangleleft_{180} z])$$

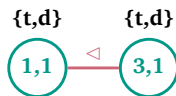
$$\varphi_{\text{license}}(x) \stackrel{\text{def}}{=} \neg \text{rel}(x) \wedge \neg \text{GLO}(x) \wedge \neg \text{VEL}(x) \neg \text{SecArc}(x)$$

$$\varphi_{\text{t,d}}(x) \stackrel{\text{def}}{=} \text{TT}(x) \wedge \text{clo}(x)$$

Input:



Output:



- The substructure that drives epenthesis in the Dynamic Grammar is equivalent to the substructure driving epenthesis in the Rule Grammar.

Local Summary

- ▶ Model theory provides an abstract way to directly translate, and therefore make connections, between symbolic strings and coupling graphs. FO logic is relatively restricted (and we can probably go lower in expressive power...)
- ▶ It appears that any theory of phonology needs a way to change basic structure when new morphemes are added. If we think about symbolic phonology as operating over equivalence classes of coupling graph structure then the gap between theories narrows.
- ▶ Still plenty of technical stuff to work out:
 - ▶ Variation, long-distance phenomena, coupling strength, stiffness, ...

Conclusion

Big Takeaway 1

- ▶ Making verbal theories formal allows us to see the exact consequences of the system we are constructing.
- ▶ The phonetics-phonology interface is itself a “parameter” in our theory space.
 - ▶ Is our phonological theory wrong or is our idea about the transduction of phonological structure into phonetic form wrong?
- ▶ The blueprint model of production allowed us to rethink the necessity of gestural representations in our *phonological representations*.

van Rooij and Blokpoel (2020); Quine (1951); Duhem (1954); Nelson (2024)

Big Takeaway 2

- ▶ Our understanding of the expressive needs of phonological knowledge can help put a restriction on the explicit types of information that is *necessary* to store in our long-term memory representations.
- ▶ Maybe symbols are enough?
 - ▶ Still leaves the questions why these patterns and not others.
 - ▶ Should we require this of our representational system?
- ▶ The ability to generate gestural representations + blueprint model weakens any claims beyond reductionism for preferring our theory *be* nonlinear dynamics.

Nelson (2024); Fodor (1974)

Thank You!

Bibliography I

- Baković, E. (2005). Antigemination, assimilation and the determination of identity. Phonology, 22(3):279–315.
- Beckman, M. E. (1996). When is a syllable not a syllable. Phonological structure and language processing: Cross-linguistic studies, pages 95–123.
- Bermúdez-Otero, R. (2007). Diachronic phonology. In de Lacy, P., editor, The Cambridge handbook of phonology, pages 497–517. Cambridge.
- Bhaskar, S., Chandlee, J., and Jardine, A. (2023). Rational functions via recursive schemes. arXiv preprint arXiv:2302.03074.
- Bhaskar, S., Chandlee, J., Jardine, A., and Oakden, C. (2020). Boolean monadic recursive schemes as a logical characterization of the subsequential functions. In Language and Automata Theory and Applications: 14th International Conference, pages 157–169. Springer.
- Browman, C. P. and Goldstein, L. (1990). Tiers in articulatory phonology, with some implications for casual speech. Papers in laboratory phonology I: Between the grammar and physics of speech, pages 341–376.
- Browman, C. P. and Goldstein, L. M. (1986). Towards an articulatory phonology. Phonology, 3:219–252.
- Browman, C. P. and Goldstein, L. M. (1995). Dynamics and articulatory phonology. In Mind as Motion, chapter 7, pages 175–193. MIT Press.
- Chandlee, J. and Jardine, A. (2021). Computational universals in linguistic theory: Using recursive programs for phonological analysis. Language, 97(3):485–519.
- Cheng, E. (2022). The joy of abstraction: An exploration of math, category theory, and life. Cambridge University Press.
- Church, A. (1932). A set of postulates for the foundation of logic. Annals of mathematics, pages 346–366.
- Church, A. (1933). A set of postulates for the foundation of logic. Annals of mathematics, pages 839–864.
- Czarnecki, V. (2025). The logic of linearization: Interpretations of trees via strings. Proceedings of the Society for Computation in Linguistics, 8(1).
- Danis, N. (2025). Logical transductions are not sufficient for notational equivalence. In Avelino, G., Balihaxi, M., Colvin, Q., Czarnecki, V., Joo, H., Wang, C., Zobarlar, U., Jardine, A., and McCollum, A., editors, Proceedings of the 2023 and 2024 Annual Meetings on Phonology. University of Massachusetts Amherst Libraries.

Bibliography II

- Dinnsen, D. and Garcia-Zamor, M. (1971). The three degrees of vowel length in German. Research on Language & Social Interaction, 4(1):111–126.
- Duhem, P. M. M. (1954). The Aim and Structure of Physical Theory. Princeton University Press.
- Fodor, J. A. (1974). Special sciences, (or: the disunity of science as a working hypothesis). Synthese, 28:97–115.
- Friedman, H. M. and Visser, A. (2014). When bi-interpretability implies synonymy. Logic Group Preprint Series, 320:1–19.
- Fromkin, V. A. (2000). Linguistics: an introduction to linguistic theory. Blackwell.
- Gafos, A. I. (2002). A grammar of gestural coordination. Natural language & linguistic theory, 20(2):269–337.
- Gafos, A. I. and Benus, S. (2006). Dynamics of phonological cognition. Cognitive science, 30(5):905–943.
- Gahl, S. (2008). Time and thyme are not homophones: The effect of lemma frequency on word durations in spontaneous speech. Language, 84(3):474–496.
- Goldstein, L. (2011). Back to the past tense in English. In Gutiérrez-Bravo, R., Mikkelsen, L., and Potsdam, E., editors, Representing language: Essays in honor of Judith Aissen, pages 69–88.
- Hall, N. (2010). Articulatory phonology. Language and Linguistics Compass, 4(9):818–830.
- Heinz, J. (2018). The computational nature of phonological generalizations. In Hyman, L. and Plank, F., editors, Phonological Typology, Phonetics and Phonology, chapter 5, pages 126–195. De Gruyter Mouton.
- Idsardi, W. J. (2025). Smplish concepts for phonology. Talk given at NPhCxi at Concordia University.
- Jardine, A., Danis, N., and Iacoponi, L. (2021). A formal investigation of q-theory in comparison to autosegmental representations. Linguistic Inquiry, 52(2):333–358.
- Kenstowicz, M. (2010). The cambridge handbook of phonology. Language, 86(1):216–219.
- Lambert, D. (2022). Unifying Classification Schemes for Languages and Processes With Attention to Locality and Relativizations Thereof. PhD thesis.
- Levelt, W. J. (1993). Speaking: From intention to articulation. MIT press.
- Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. CUMINCAD.

Bibliography III

- McCarthy, J. J. (1988). Feature geometry and dependency: A review. Phonetica, 45(2-4):84–108.
- Miller, P. H. (1999). Strong generative capacity: The semantics of linguistic formalism. CSLI.
- Nam, H. (2007). A gestural coupling model of syllable structure. PhD thesis, Yale University.
- Nelson, S. (2022). A model theoretic perspective on phonological feature systems. Proceedings of the Society for Computation in Linguistics, 5(1).
- Nelson, S. (2024). The Computational Structure of Phonological and Phonetic Knowledge. PhD thesis, Stony Brook University.
- Nelson, S. and Heinz, J. (2026). The blueprint model of production. Phonology.
- Newell, A. and Simon, H. A. (1975). Computer science as empirical inquiry: Symbols and search. In ACM Turing award lectures, pages 113–126.
- Nolan, F. (1992). The descriptive role of segments: Evidence from assimilation. In Docherty, G. J. and Ladd, D. R., editors, Papers in Laboratory Phonology 2: Gesture, Segment, Prosody. Cambridge University Press.
- Oakden, C. (2020). Notational equivalence in tonal geometry. Phonology, 37(2):257–296.
- Pierce, B. C. (2002). Types and programming languages. MIT press.
- Pierrehumbert, J. (2002). Word-specific phonetics. Laboratory phonology, 7.
- Port, R. and Crawford, P. (1989). Incomplete neutralization and pragmatics in German. Journal of phonetics, 17:257–282.
- Port, R., Mitleb, F., and O'Dell, M. (1981). Neutralization of obstruent voicing in German is incomplete. The journal of the Acoustical Society of America, 70(S1):S13–S13.
- Port, R. F. and Leary, A. P. (2005). Against formal phonology. Language, 81(4):927–964.
- Port, R. F. and O'Dell, M. L. (1985). Neutralization of syllable-final voicing in German. Journal of phonetics.
- Quine, W. V. O. (1951). Two dogmas of empiricism. Philosophical Review, 60(1):20–43.
- Roark, B. and Sproat, R. (2007). Computational approaches to morphology and syntax, volume 4. Oxford University Press.

Bibliography IV

- Rogers, J., Heinz, J., Bailey, G., Edlefsen, M., Visscher, M., Wellcome, D., and Wibel, S. (2010). On languages piecewise testable in the strict sense. In The Mathematics of Language: 10th and 11th Biennial Conference, MOL 10, Los Angeles, CA, USA, July 28-30, 2007, and MOL 11, Bielefeld, Germany, August 20-21, 2009, Revised Selected Papers, pages 255–265. Springer.
- Rogers, J., Heinz, J., Fero, M., Hurst, J., Lambert, D., and Wibel, S. (2013). Cognitive and sub-regular complexity. In Formal Grammar: 17th and 18th International Conferences, FG 2012, Opole, Poland, August 2012, Revised Selected Papers, FG 2013, Düsseldorf, Germany, August 2013. Proceedings, pages 90–108. Springer.
- Scobbie, J. M. (2005). The phonetics phonology overlap. In QMU Speech Science Research Centre Working Papers, WP-1.
- Simon, H. A. (1990). Invariants of human behavior. Annual review of psychology, 41(1):1–20.
- Sipser, M. (2013). Introduction to the Theory of Computation. Cengage Learning, third edition edition.
- Smith, C. M. (2018). Harmony in gestural phonology. PhD thesis, University of Southern California.
- Strother-Garcia, K. (2019). Using model theory in phonology: a novel characterization of syllable structure and syllabification. PhD thesis, University of Delaware.
- van Rooij, I. and Baggio, G. (2021). Theory before the test: How to build high-verisimilitude explanatory theories in psychological science. Perspectives on Psychological Science.
- van Rooij, I. and Blokpoel, M. (2020). Formalizing verbal theories. Social Psychology, 51(5):285–298.
- Wirth, N. (1978). Algorithms + data structures = programs. Prentice Hall PTR.