Gradient Symbol Processing for Phonological Production

or

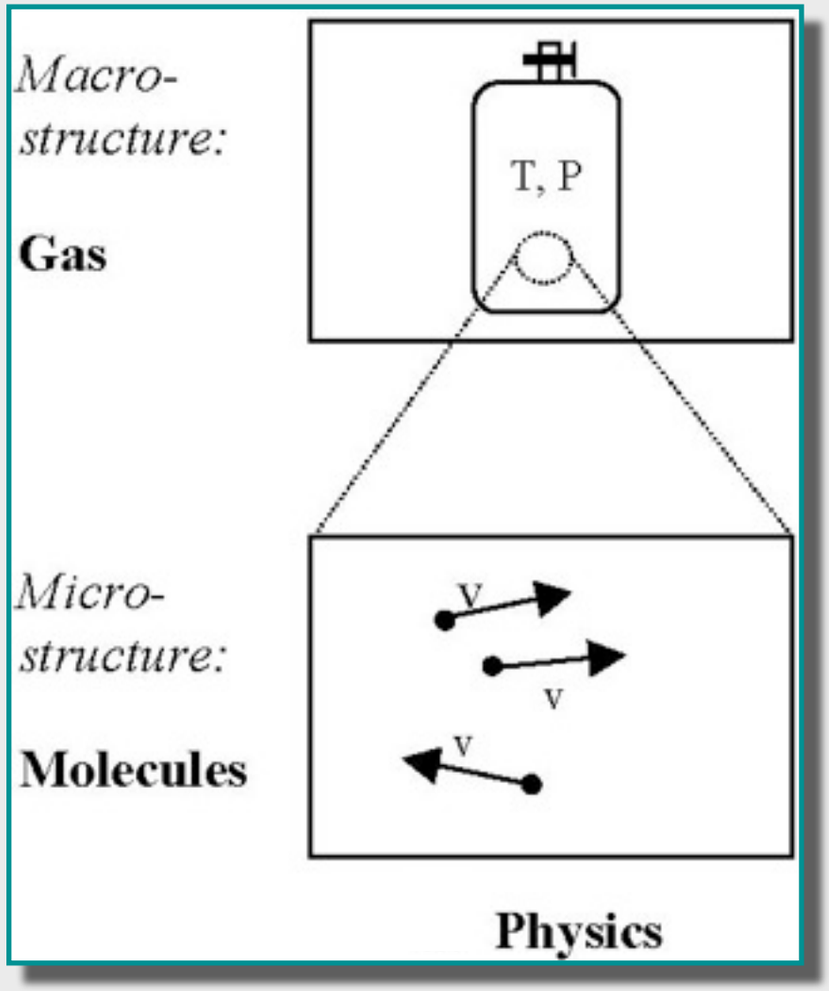
YACA:
Yet Another Cognitive Architecture

Joint work with:

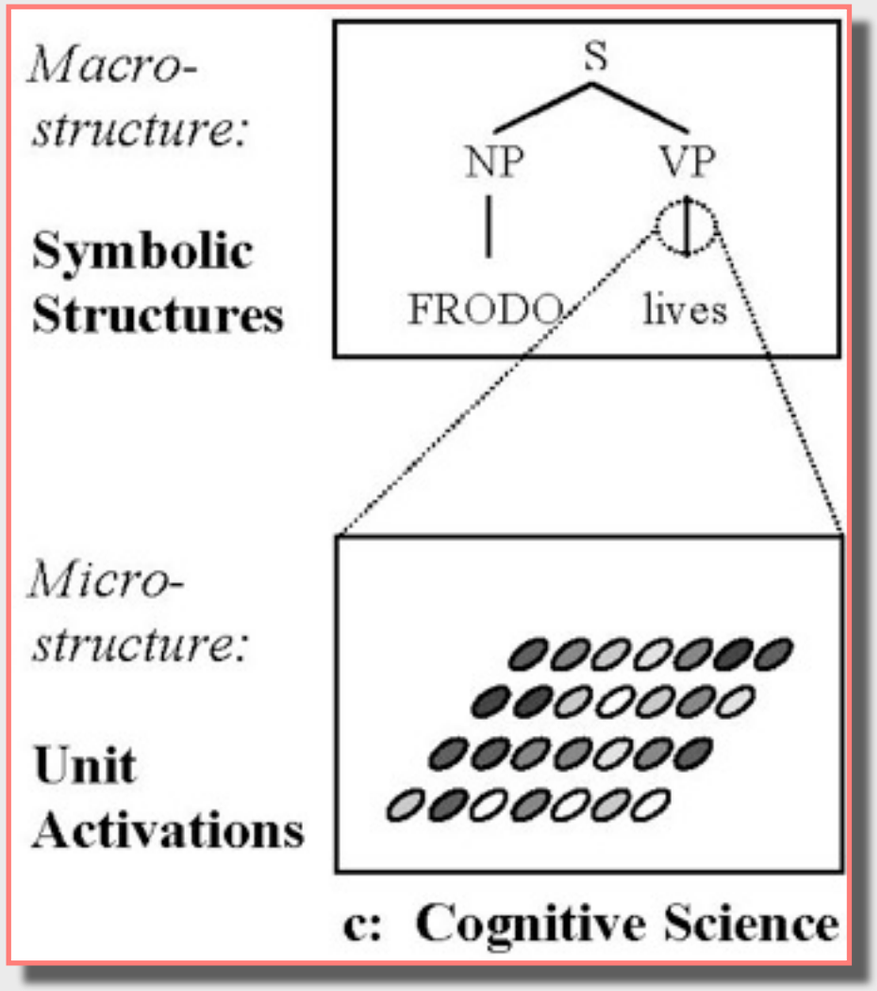
Matt Goldrick (Northwestern Linguistics)
Don Mathis (Johns Hopkins Cognitive Science)

Split-level architecture

The inspiration



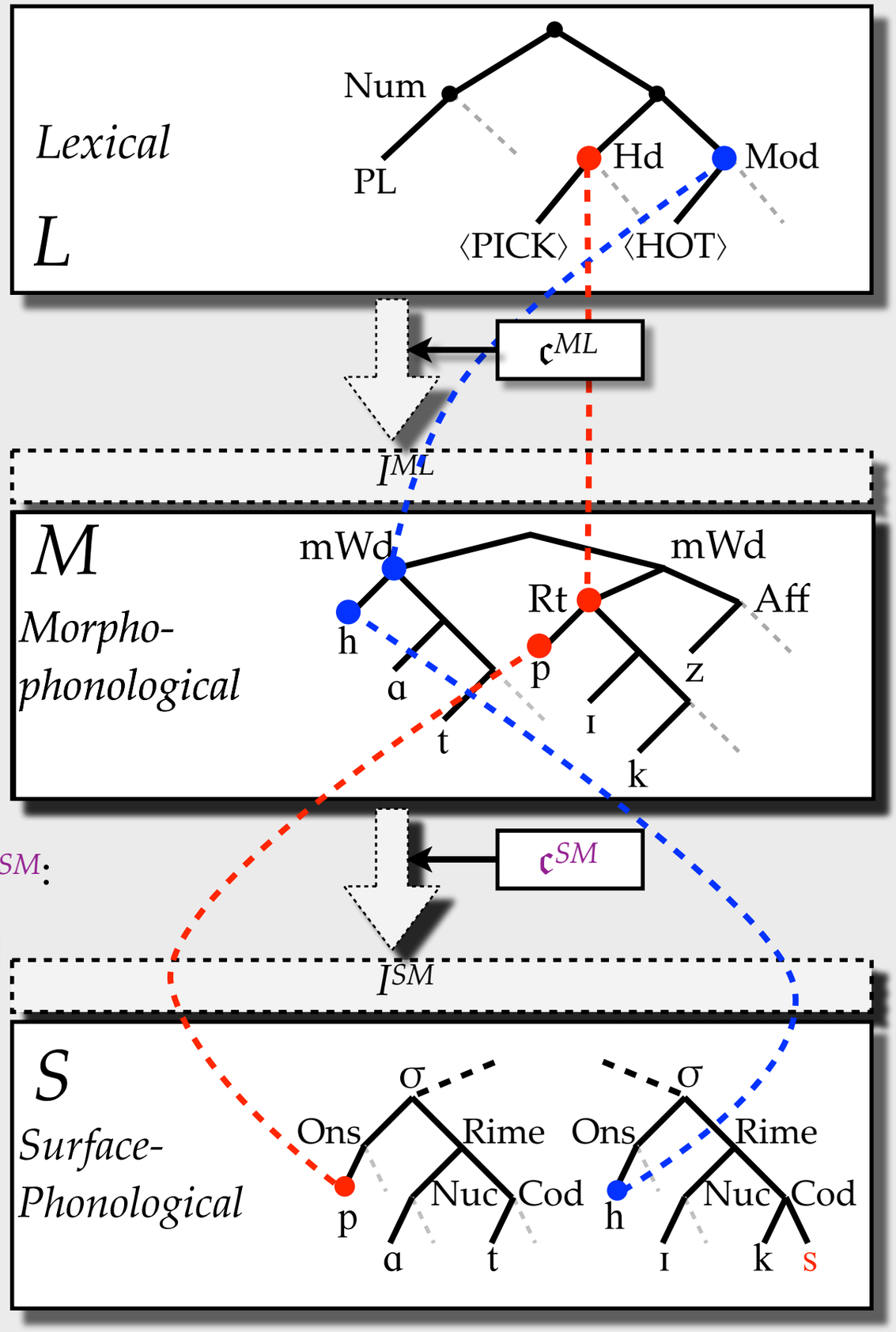
The proposal



General cognitive macro-architecture

Graph:
- Node:
  - Representation
    - information of a particular type
    - result of function computation
    - (Markedness constraints)
- Edge:
  - Input to function
  - Bears a correspondence relation
  - (Faithfulness constraints)

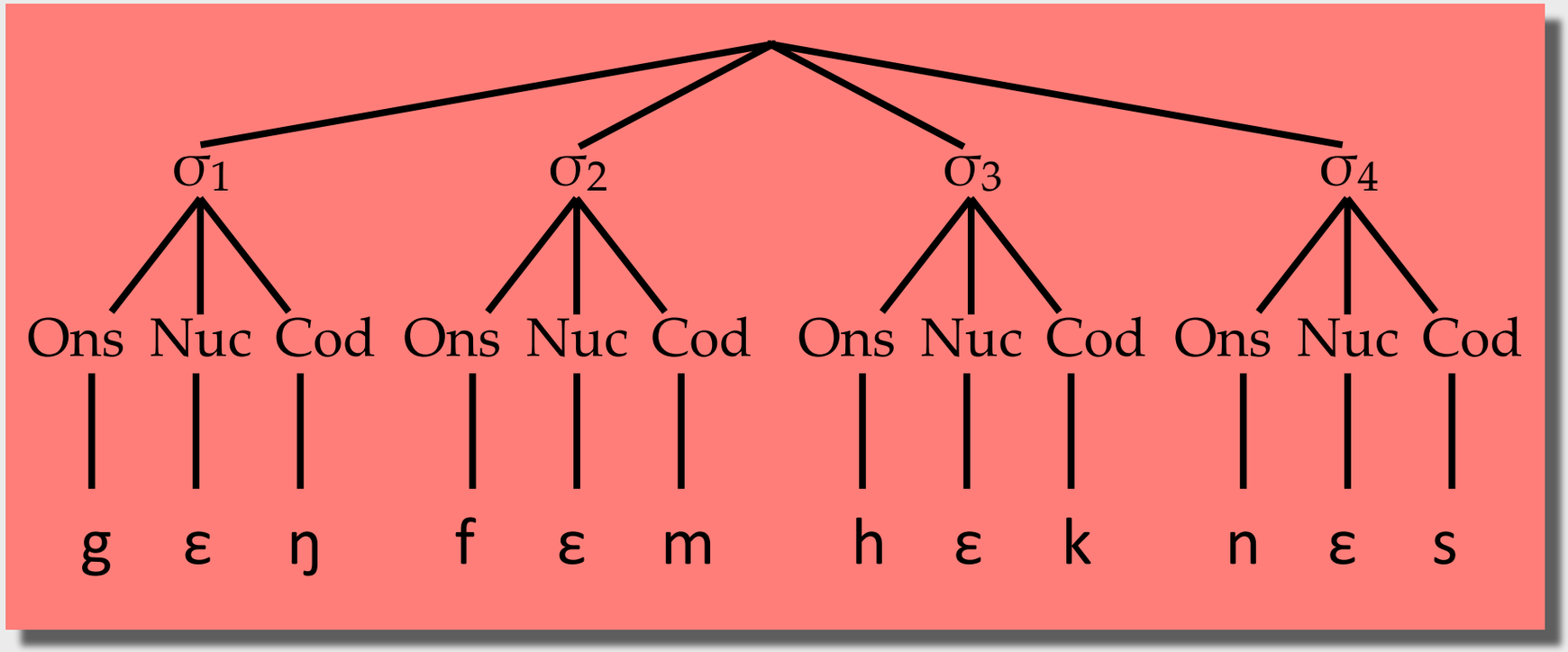
Processes at node:
- Optimization
- Quantization



Error in correspondence $\mathfrak{c}^{SM}$:

/hɑt pɪkz/ → [pɑt hɪks]

Representation



Symbol structures $\mathcal{S}$

Filler/role decomposition (possibly recursive):

$s = \{g/[\text{Ons}/\sigma_1], \eta/[\text{Cod}/\sigma_1], \varepsilon/[\text{Nuc}/\sigma_2], \ldots\} \subset \mathcal{F} \times \mathcal{R}$

(activation-)vector space embedding $\mathbf{v}_s \in \mathbf{F} \otimes \mathbf{R} = \mathbb{R}^n$ Random* vectors

[gɛŋ fɛm ...]: $\mathbf{v}_s = \mathbf{g} \otimes \mathbf{r}_{\text{Ons}/\sigma_1} + \boldsymbol{\eta} \otimes \mathbf{r}_{\text{Cod}/\sigma_1} + \boldsymbol{\varepsilon} \otimes \mathbf{r}_{\text{Nuc}/\sigma_1} + \mathbf{f} \otimes \mathbf{r}_{\text{Ons}/\sigma_2} + \cdots$

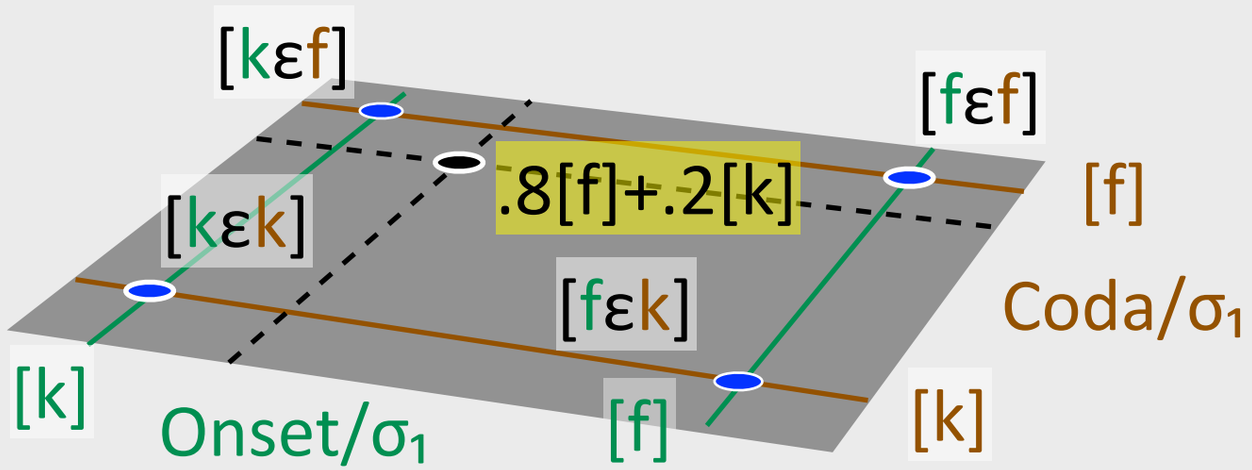
Image of embedding: 'the Grid' # of 'pure states'

$\mathbf{r}_{\text{Ons}/\sigma_2} = \mathbf{r}_{\text{Ons}} \otimes \mathbf{r}_{\sigma_2}$

* Capturing the similarity structure of roles (including recursive hierarchical structure) is a major feature of distributed tensor product representations

Representation

‘the Grid’



Grammars in networks

Gen: representations

Con: (OT grammar $\mathcal{G}$ →)

① HG grammar $H_{\mathcal{G}}$

② $\mathcal{G} \rightarrow \mathbf{W}_{\mathcal{G}}$ weight matrix of a network $\mathcal{N}$ s.t.

$H_{\mathcal{N}_0}(\mathbf{s}) = H_{\mathcal{G}}(s) \qquad \forall \mathbf{s} \in \#$ (= the Grid) — *iso-Harmonic embedding*

Harmony optimization in networks

Theorem. For any deterministic neural network in a certain class, during processing, Harmony continuously increases, reaching a *local* optimum.

- This is **network Harmony** $H_{\mathcal{N}}$ $= H_{\mathcal{N}_0} + H_{\mathcal{N}_1}$ $H_{\mathcal{N}_1}(\mathbf{a}) = \frac{1}{2}|\mathbf{a}|^2$

  $H_{\mathcal{N}_0}(\mathbf{a}) \equiv \sum_{\beta\gamma} a_\beta W_{\beta\gamma} a_\gamma$ — quadratic, dependent on **W**

Theorem. For any stochastic neural network in a certain class, during processing, the probability of visiting a state **a** approaches

$p(\mathbf{a}) \propto e^{H_{\mathcal{N}}(\mathbf{a})/T}$ (T = randomness parameter)

As $T \longrightarrow 0$, the probability the network is in a *globally* optimal state $\longrightarrow 1$.

- These networks use a **Diffusion Dynamics**

$$da_\beta = \sum_\gamma W_{\beta\gamma} a_\gamma \, dt + \sqrt{2T}\, dB_\beta = \frac{\partial H_{\mathcal{N}}}{\partial a_\beta} dt + \sqrt{2T}\, dB_\beta$$

Harmony optimization by diffusion

Processing: **Diffusion Dynamics**

- State moves in time so as to increase Harmony $H(s)$, on average
  - ✦ randomness in state changes with variance $\propto T$
    - ✦ during processing, $T \longrightarrow 0$
    - ✦ hence $p(s) \longrightarrow 0$ except for the state(s) with maximal Harmony
  - ✦ N.B.: randomness needed to find *global* Harmony maxima
    - ✦ not infallible: errors occur
    - ✦ from mechanism responsible for *correct* performance

☞ **Optimization** process

HG in networks

Theorem. Any rewrite-rule grammar can be expressed as a second-order Harmonic Grammar.

Theorem. For any second-order Harmonic Grammar $H_{\mathcal{G}}$, we can construct a recurrent network $\mathcal{N}$ with a harmony function $H_{\mathcal{N}}$ that provides an iso-Harmonic embedding

i.e., yields the same values as $H_{\mathcal{G}}$ on every pure (grid) state *s*:

$$H_{\mathcal{N}}(\mathbf{s}) = H_{\mathcal{G}}(s)$$

Corollary. For any Harmonic Grammar $H_{\mathcal{G}}$, we can construct a recurrent network $\mathcal{N}$ such that as $T \longrightarrow 0$, the probability the network is in a gradient state that is *globally* optimal w.r.t. $H_{\mathcal{G}} \longrightarrow 1$.

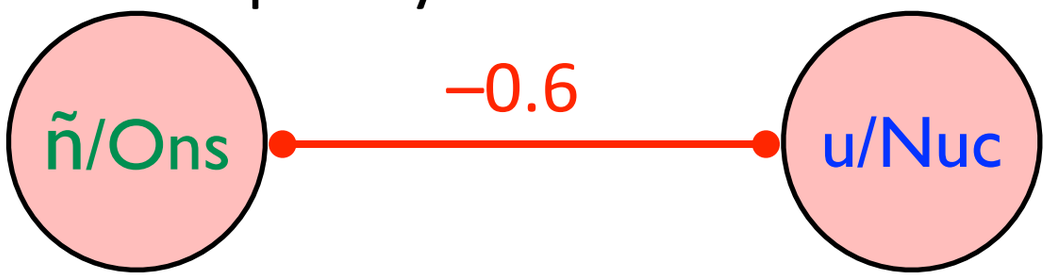
Constraints in connections

Spreading activation = finding optimal solution to weighted constraints

E.g., phonotactic constraint: *ñu (American *muse* vs. *news*)

Harmony maximization as constraint satisfaction

Consider this connection in a purely localist network:



Same constraint with distributed representations:

the weight matrix **W** such that

when activation patterns are re-described in a new coordinate system in which the representations become local,

W becomes equal to the connection above.

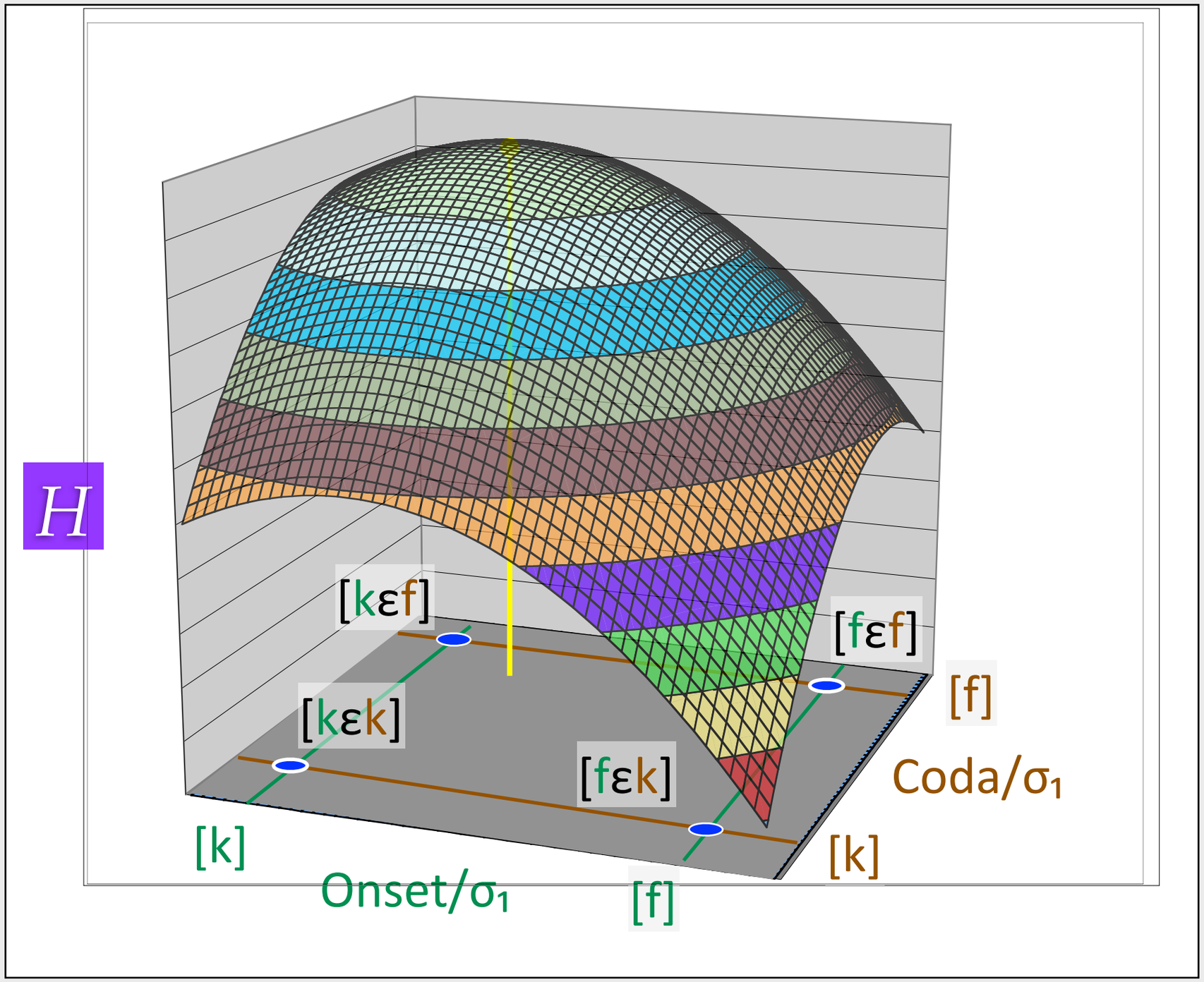
The troubles

Problems:

want a maximum of *H* at every grammatical structure

but *H* is quadratic: it can have only one global maximum

Unique global *H* maximum in $\mathbb{R}^n$



The troubles

Problems:

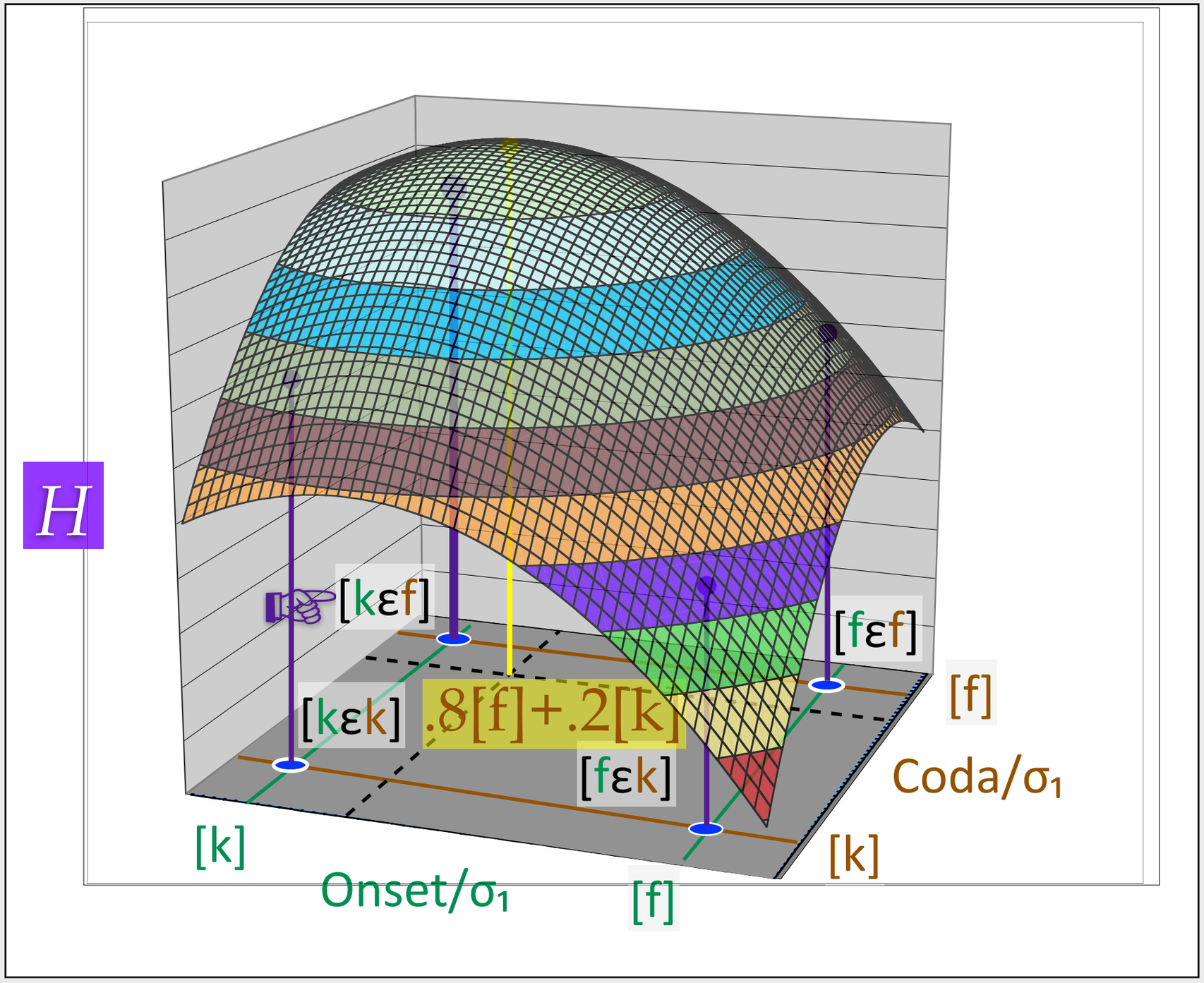
want a maximum of *H* at every grammatical structure

but *H* is quadratic: it can have only one global maximum

$\operatorname{argmax}_{\mathbf{a} \in \mathbb{R}^n} H_{\mathcal{N}}(\mathbf{a}) \notin$ Grid: it is a *blend* state

H restricted to the grid **can** have multiple maxima

Maximum is an impure state: a blend



Maximum is an impure state: a blend

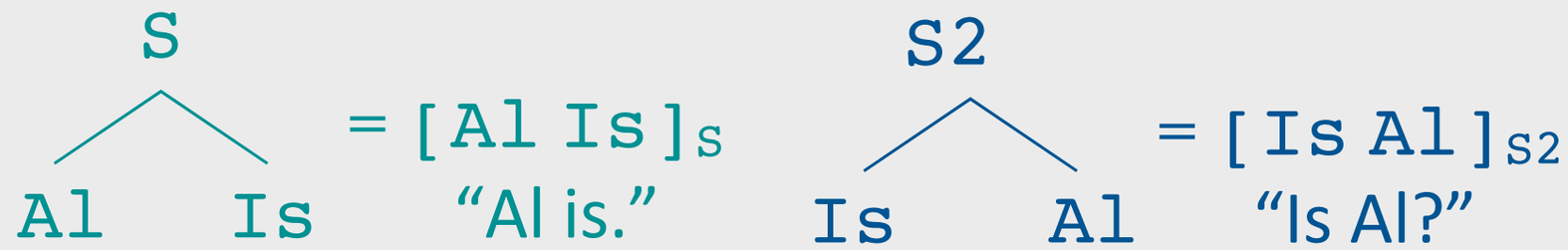
Corollary. For any Harmonic Grammar $H_{\mathcal{G}}$, we can construct a recurrent network $\mathcal{N}$ such that as $T \to 0$, the probability the network is in a (gradient) state that is *globally* optimal w.r.t. $H_{\mathcal{G}} \to 1$.

*This is a **blend** of well-formed constituents, not a globally coherent pure state.*
(A general problem, not limited to grammars.)

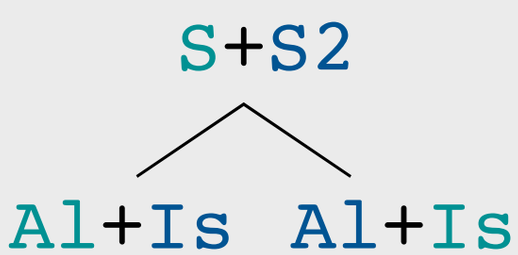
A nanogrammar $\mathcal{G}$

Start symbols: {S, S2}
S → Al Is
S2 → Is Al

Its nanolanguage $\mathcal{L}$



The global *H* optimum is proportional to



This is why we need ***quantization***.

Need for quantization

Problems:

want a maximum of *H* at every grammatical structure

but *H* is quadratic: it can have only one global maximum

$\text{argmax}_{\mathbf{a} \in \mathbb{R}^n} H_{\mathcal{N}}(\mathbf{a}) \notin$ Grid: it is a *blend* state

H restricted to the grid **can** have multiple maxima

H is meaningful only on the grid

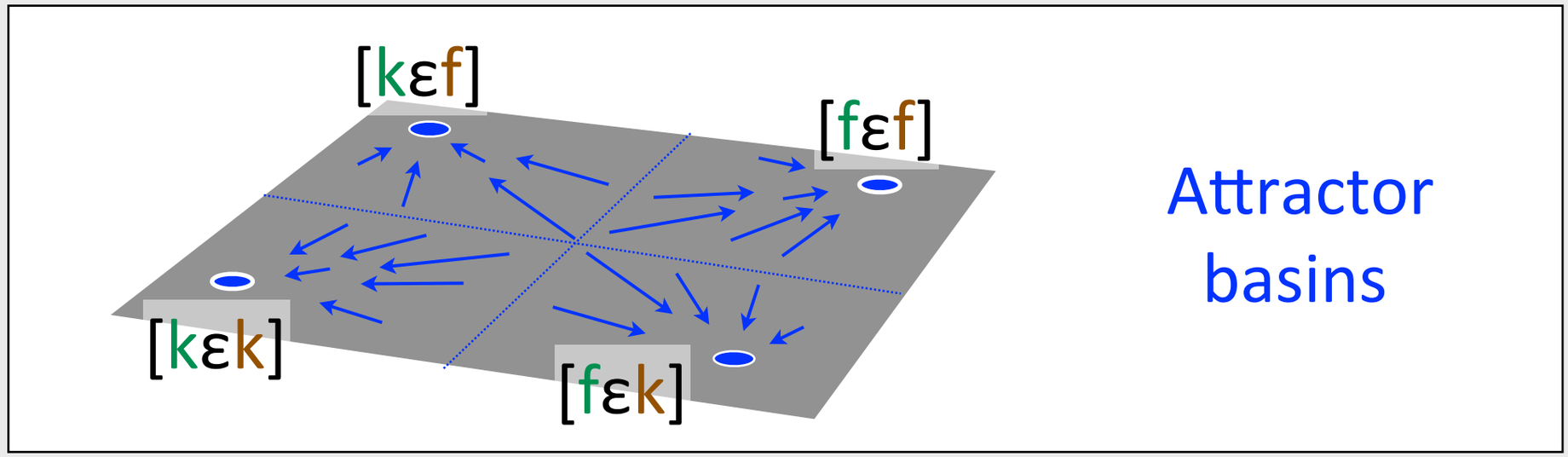
Proposal:

Add a *quantization* dynamics with an attractor at every $s \in$ Grid

Quantization: Projecting to the grid

Discretization Dynamics

- A spreading activation algorithm that creates an **attractor** at all and only the points of the grid
- Isotropic/symmetric/all attractors equivalent:
  - ✦ *optimization dynamics* pushes towards correct basin



- Distributed winner-take-all network (non-linear mutual inhibition)
  - ✦ Lotka-Volterra equations (Baird & Eeckmann 1993)

$$\frac{dx_\beta}{dt} = x_\beta - \sum_{\mu\nu} W_{\beta\mu\nu} x_\mu x_\nu \qquad W_{\beta\mu\nu} = \sum_{jk} M_{\beta k} M^{-1}_{k\mu} M^{-1}_{j\nu} (2 - \delta_{jk})$$

M = **F**⊗**R**, **F** = matrix of symbol (filler) patterns, **R** = of position (role)

Combined dynamics

Harmony Optimization Dynamics

- Diffusion; as processing proceeds, $T \longrightarrow 0$
- Pushes towards best gradient (blend) state
  - ✦ ignores discreteness

Quantization Dynamics

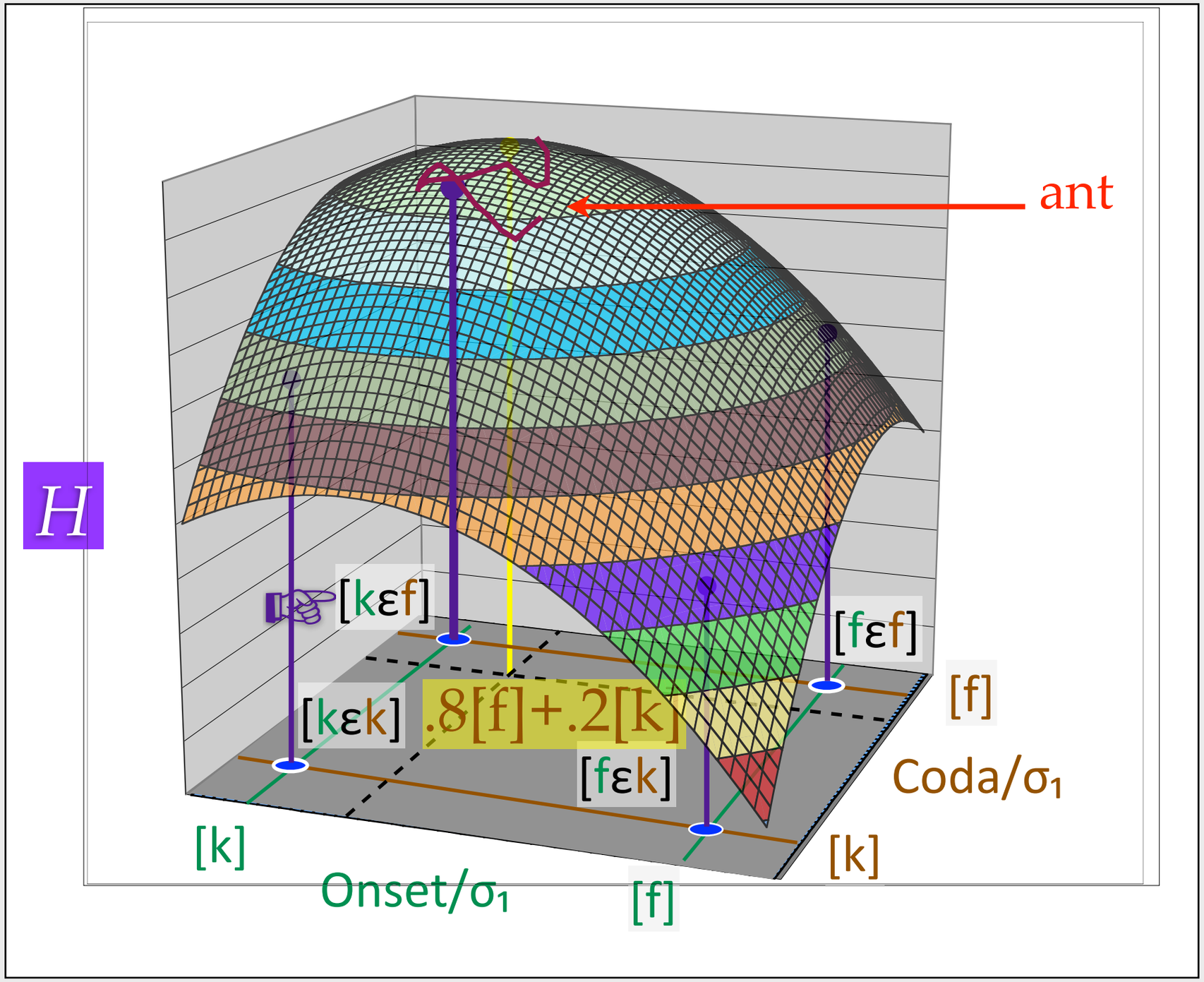
- Pushes towards the grid of discrete (pure) states
  - ✦ ignores well-formedness

Combination

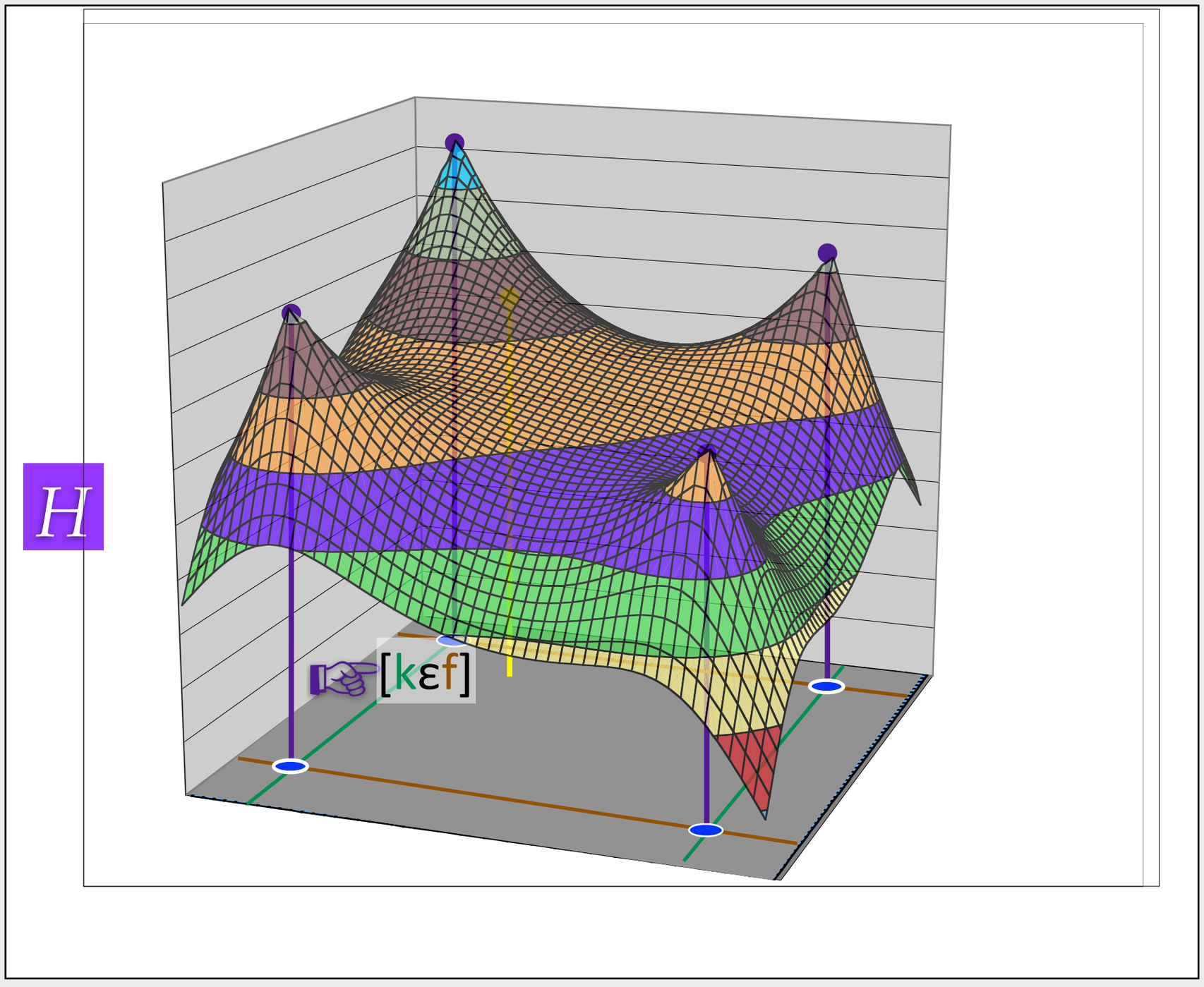
- The weighted sum of these two dynamics
- As processing proceeds, the relative weight of optimization
  - ✦ $\lambda \longrightarrow 0$
  - ✦ discretization pressure grows, dominates final computation

$$\mathcal{D} = \lambda \mathcal{D}_{opt} + [1-\lambda] \mathcal{D}_{quant}$$

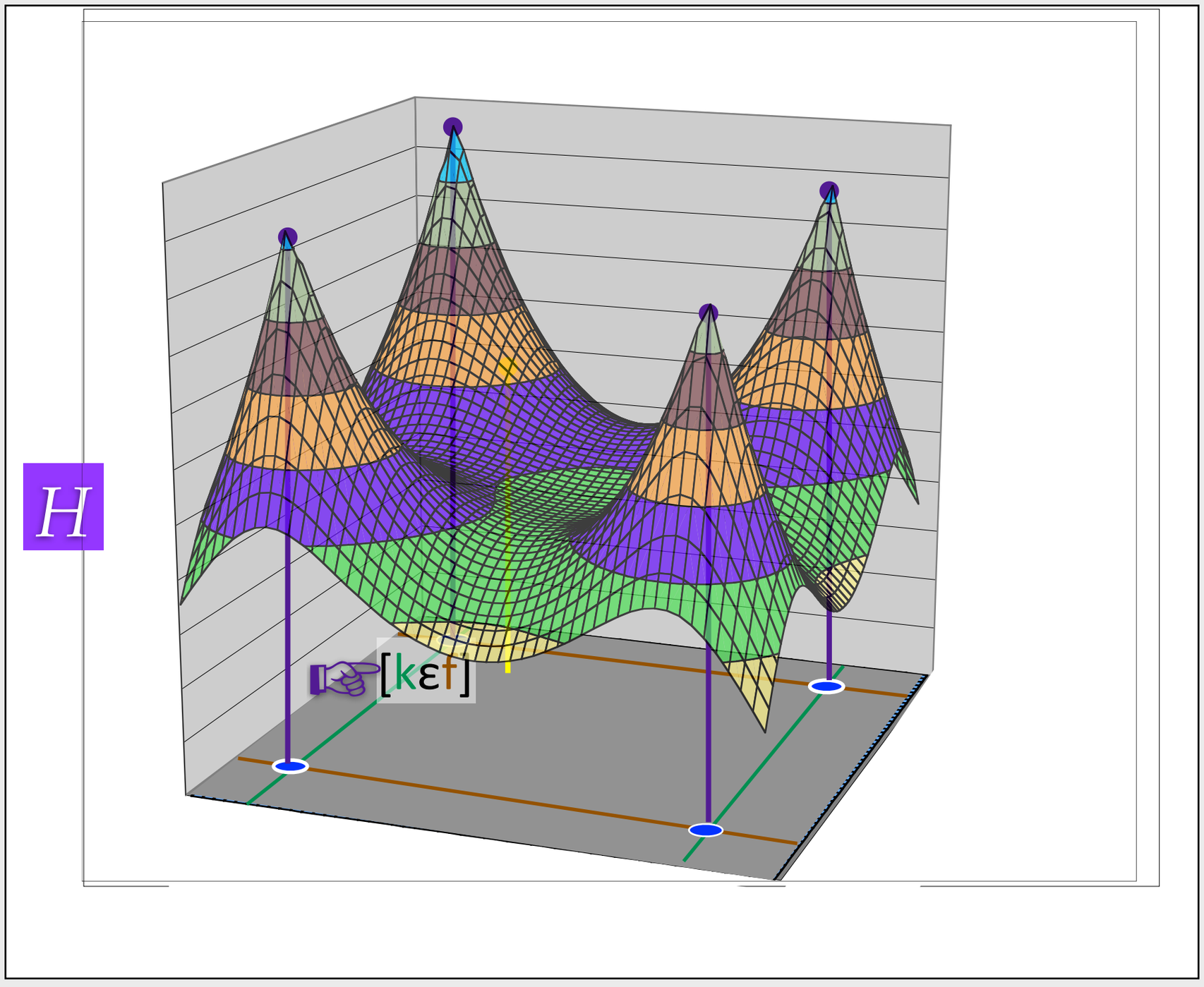
Combined dynamics as λ→0



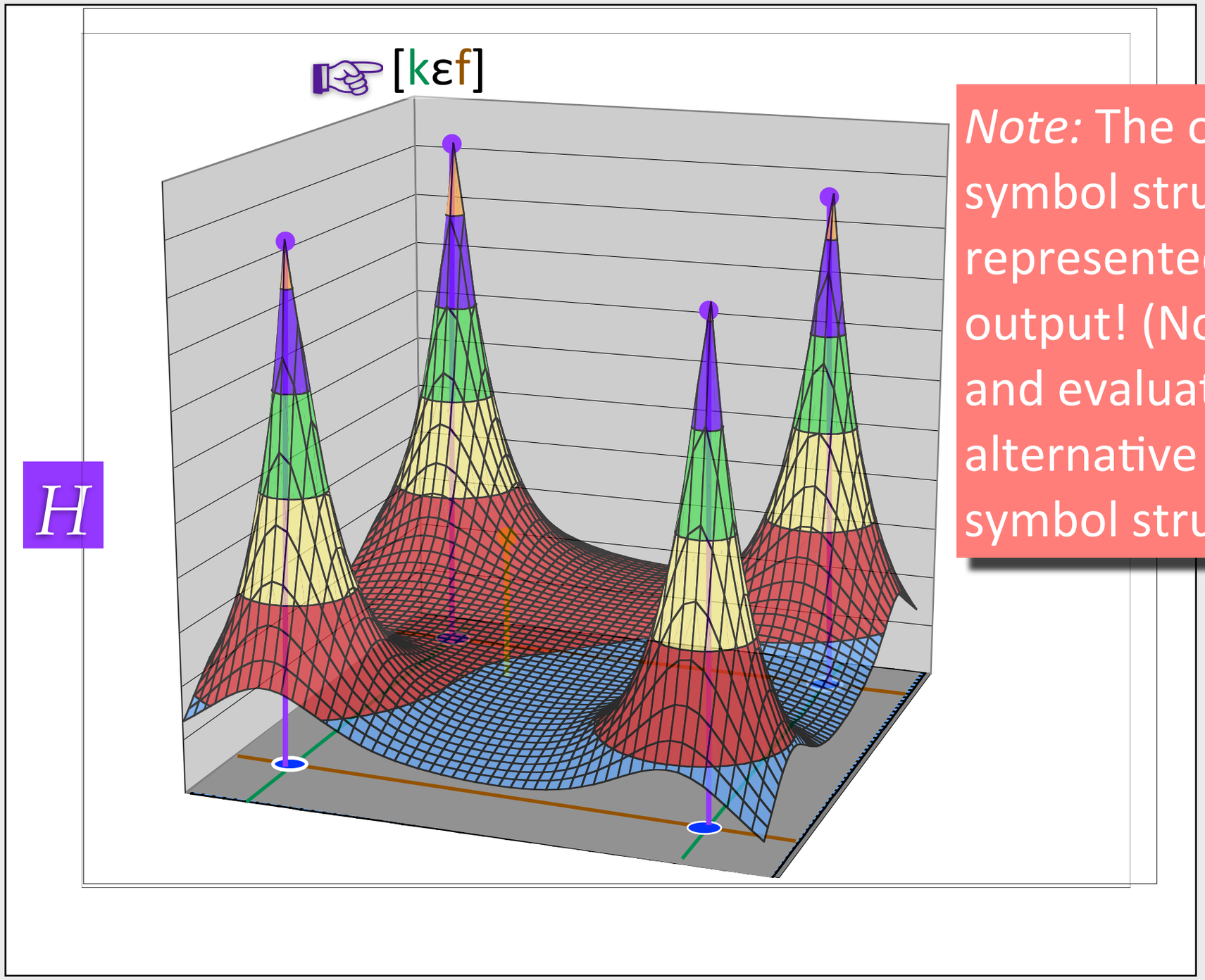
Combined dynamics as λ⟶0



Combined dynamics as λ⟶0



Combined dynamics as λ→0



Note: The only discrete symbol structure ever represented is the final output! (Not: generate and evaluate many alternative discrete symbol structures.)

The ant should end up at the highest peak
— or at an erroneous peak with *prob* ~ *H*

22

Coupled symmetry breaking

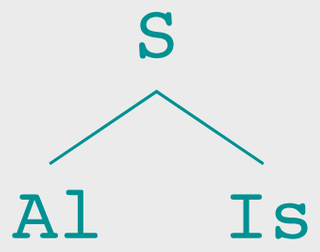
A nanogrammar $\mathcal{G}$

Start symbols: {S, S2}

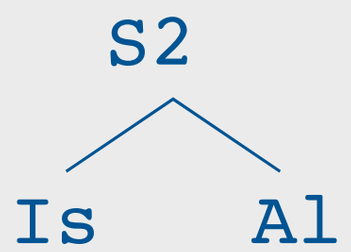
S → Al Is

S2 → Is Al

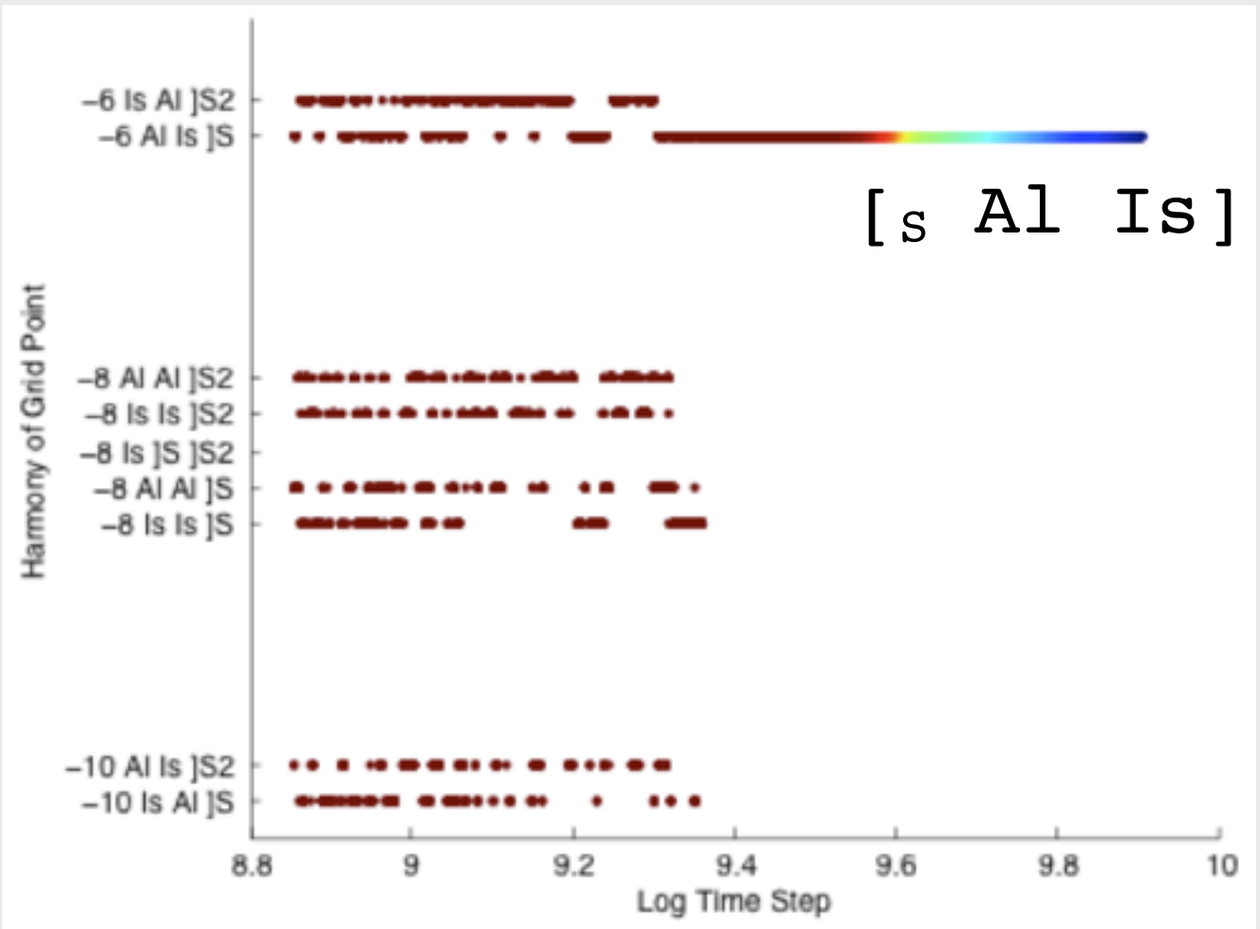
Its nanolanguage $\mathcal{L}$

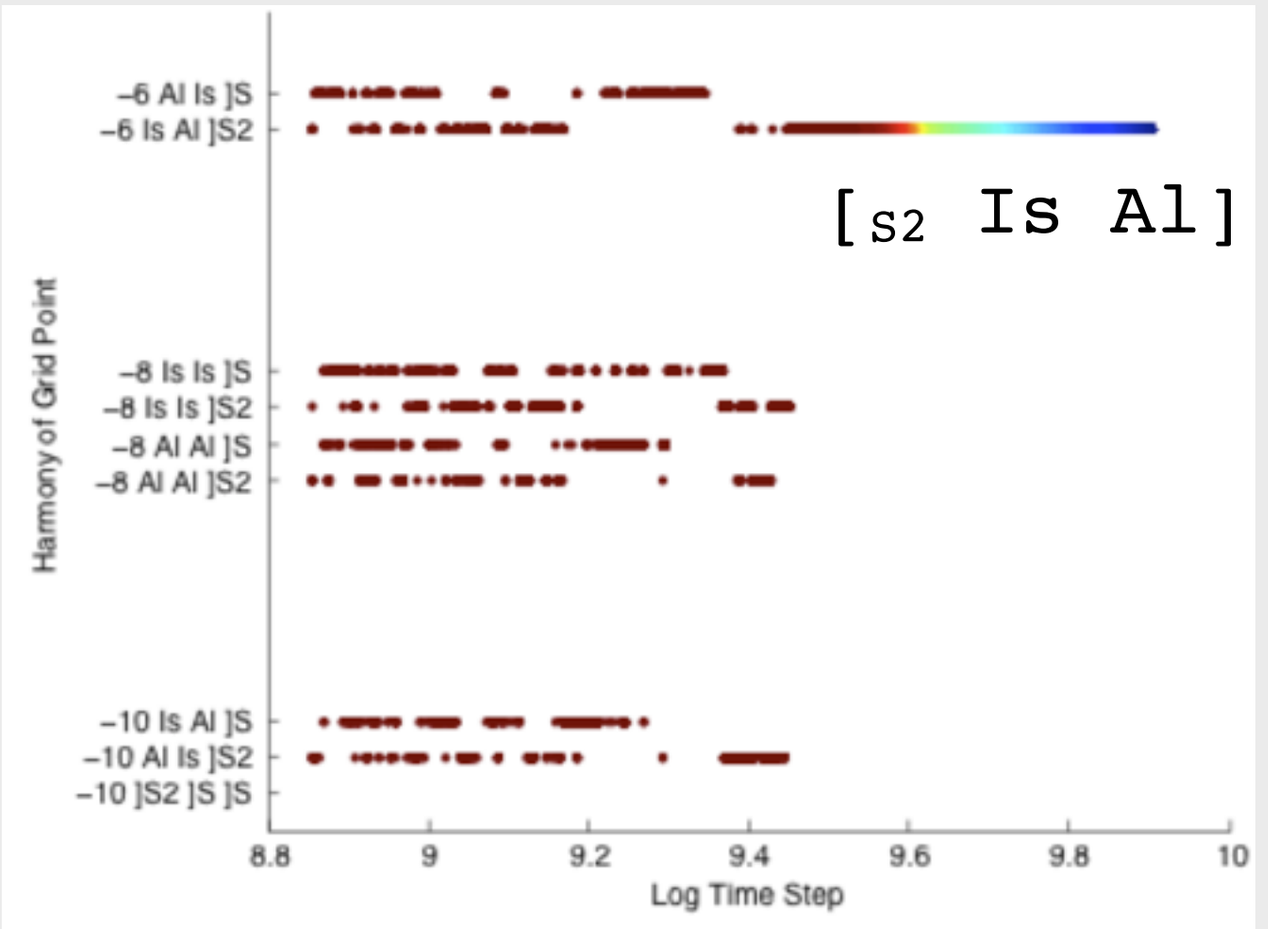


= [Al Is]S
"Al is."



= [Is Al]S2
"Is Al?"



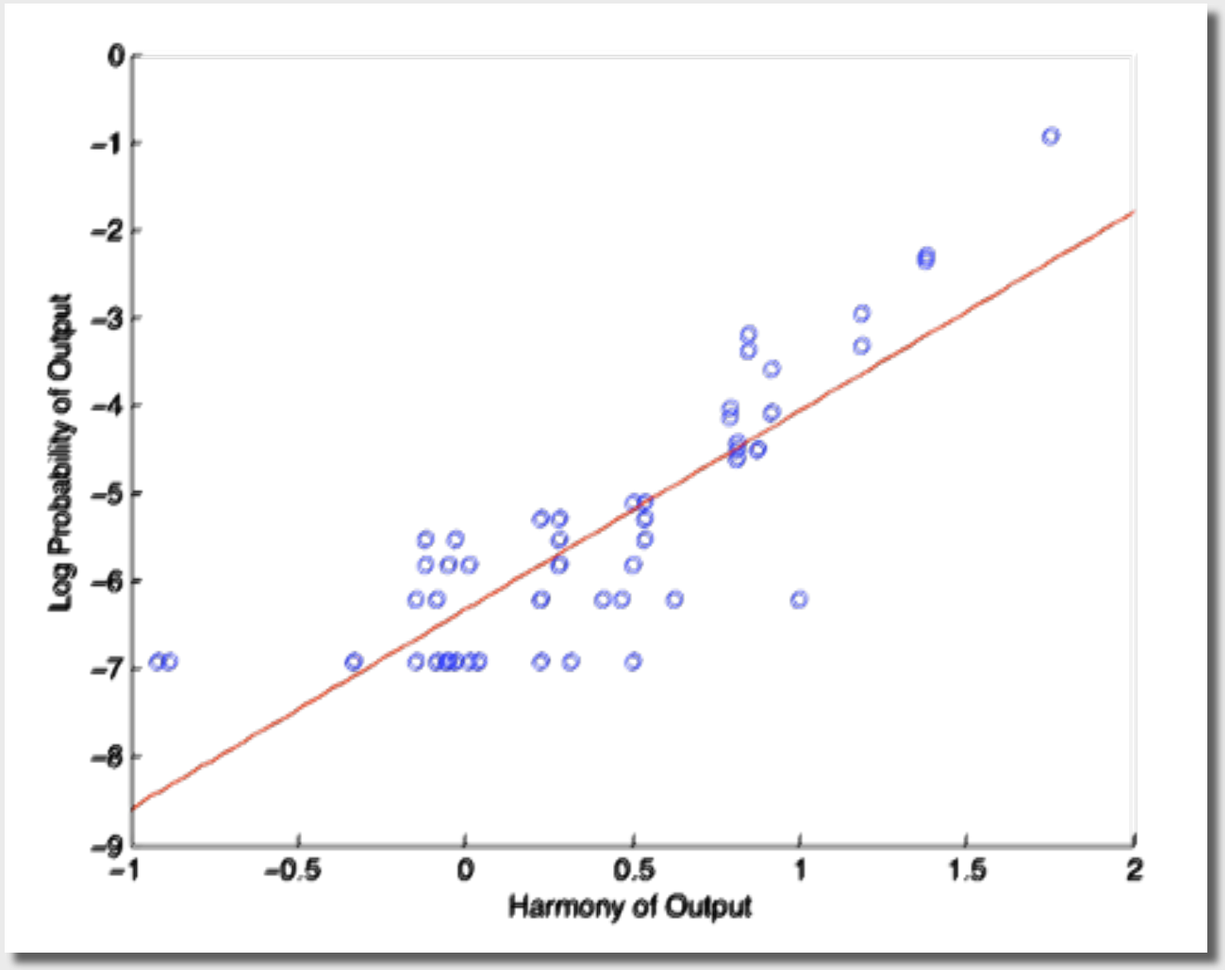


Explanation and Harmony

Explaining error patterns with Harmony

- The Harmony function is designed: we can understand it
  - ✦ *H* encodes the constraints of the problem domain, such that
  - ✦ the correct answer best-satisfies these constraints
- ¿ Probability of *s*: $p(s) \propto e^{H(s)/T}$ (*T* = randomness parameter)

  or equivalently: $\log p(s) \propto H(s) - k$
- */sag nak/* ⟶ [?]



Applications

Tongue-twister task

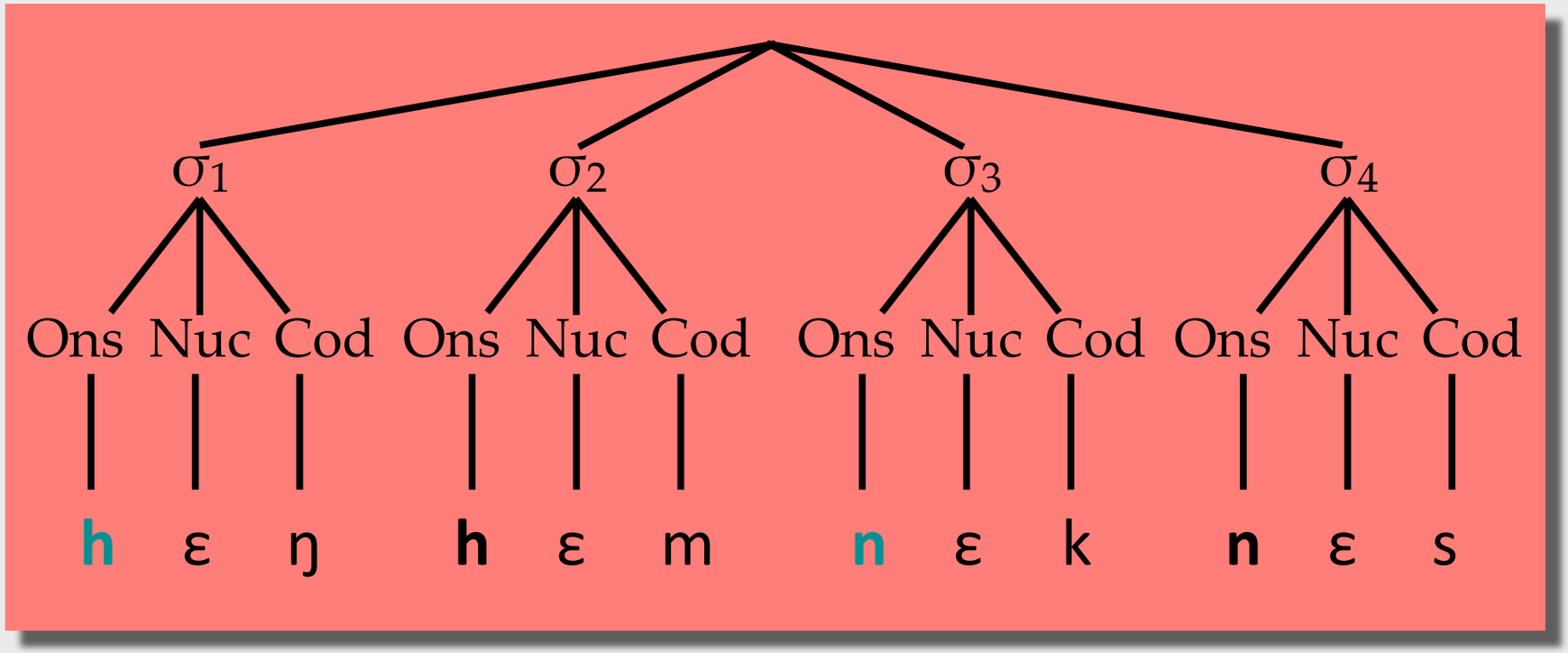
Incomplete neutralization

ITBerber syllabification

Tongue-twister task

Phonological production (**g**ɛŋ **h**ɛm **f**ɛk **n**ɛs → **h**ɛŋ hɛm **n**ɛk nɛs)*

- Final state of surface form component:



- An erroneous consonant is more likely
  - ✦ to be in a similar position (with respect to syllable structure)
  - ✦ to replace a similar consonant
  - ✦ to be in a position where it is more frequent (phonotactic probability)
    - ☞ never in a position forbidden by the English grammar (*kɛh):
      ‘errors are well-formed’

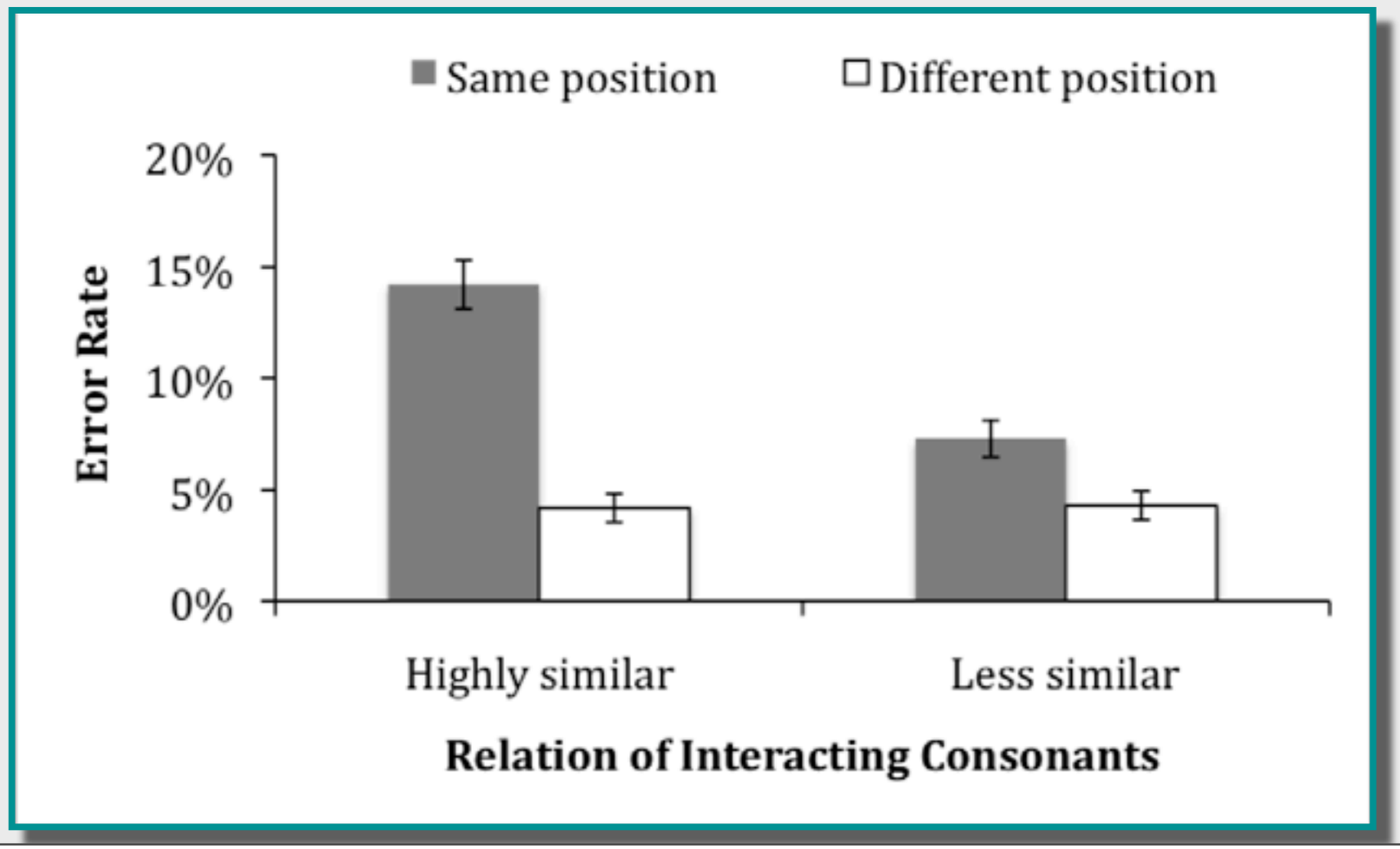
* Dell, Reed, Adams & Meyer 2000

26

Tongue-twister task

Phonological production (fɛŋ kɛg hɛm nɛs → fɛŋ hɛg nɛm nɛs)

- An erroneous consonant is more likely
  - ✦ to be in a similar position (with respect to syllable structure)
  - ✦ to replace a similar consonant



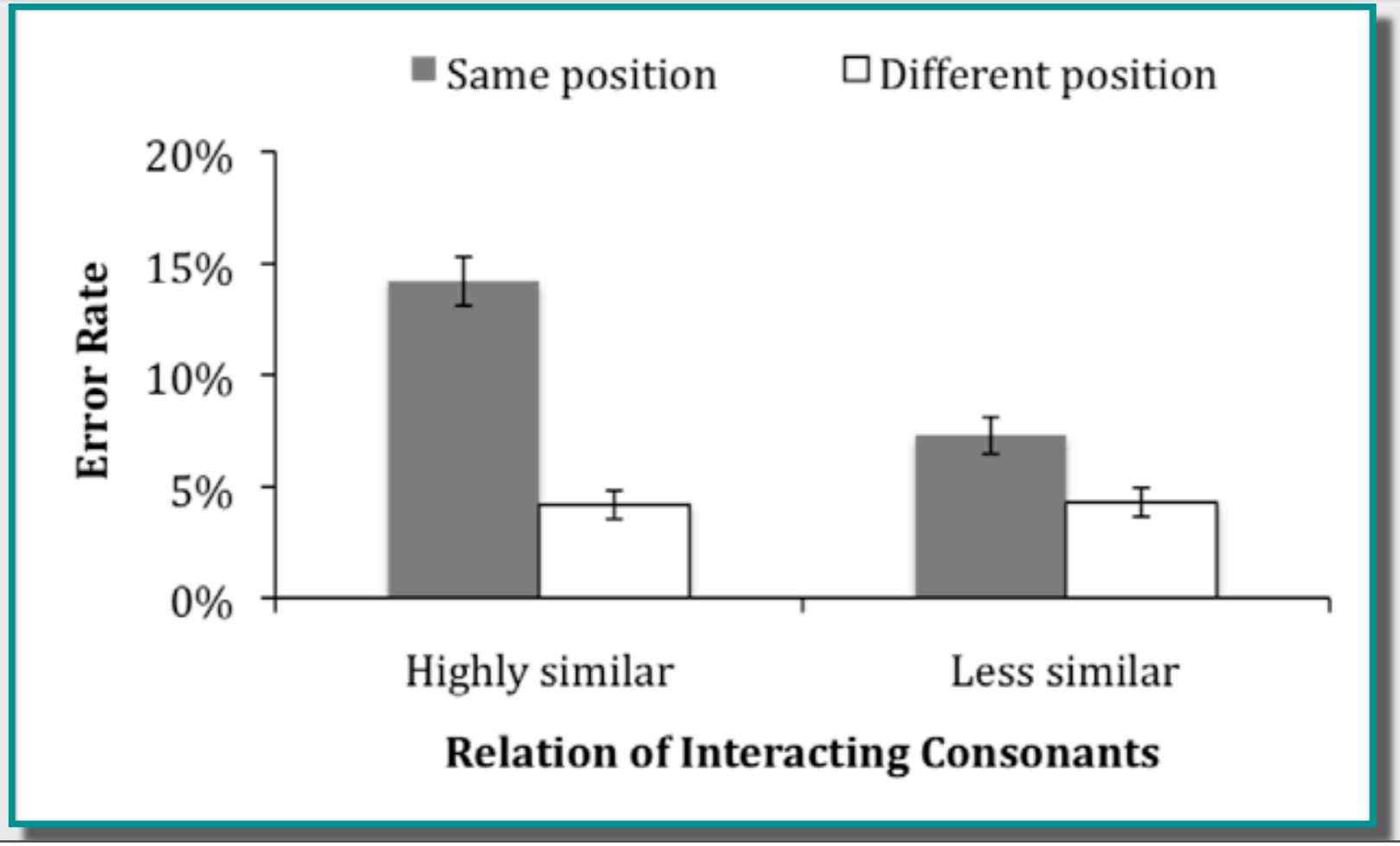
Tongue-twister task

Capturing the similarity structure of roles (including recursive hierarchical structure) is a major feature of distributed tensor product representations

$\mathbf{r}_{Ons/\sigma_2} = \mathbf{r}_{Ons} \otimes \mathbf{r}_{\sigma_2}$

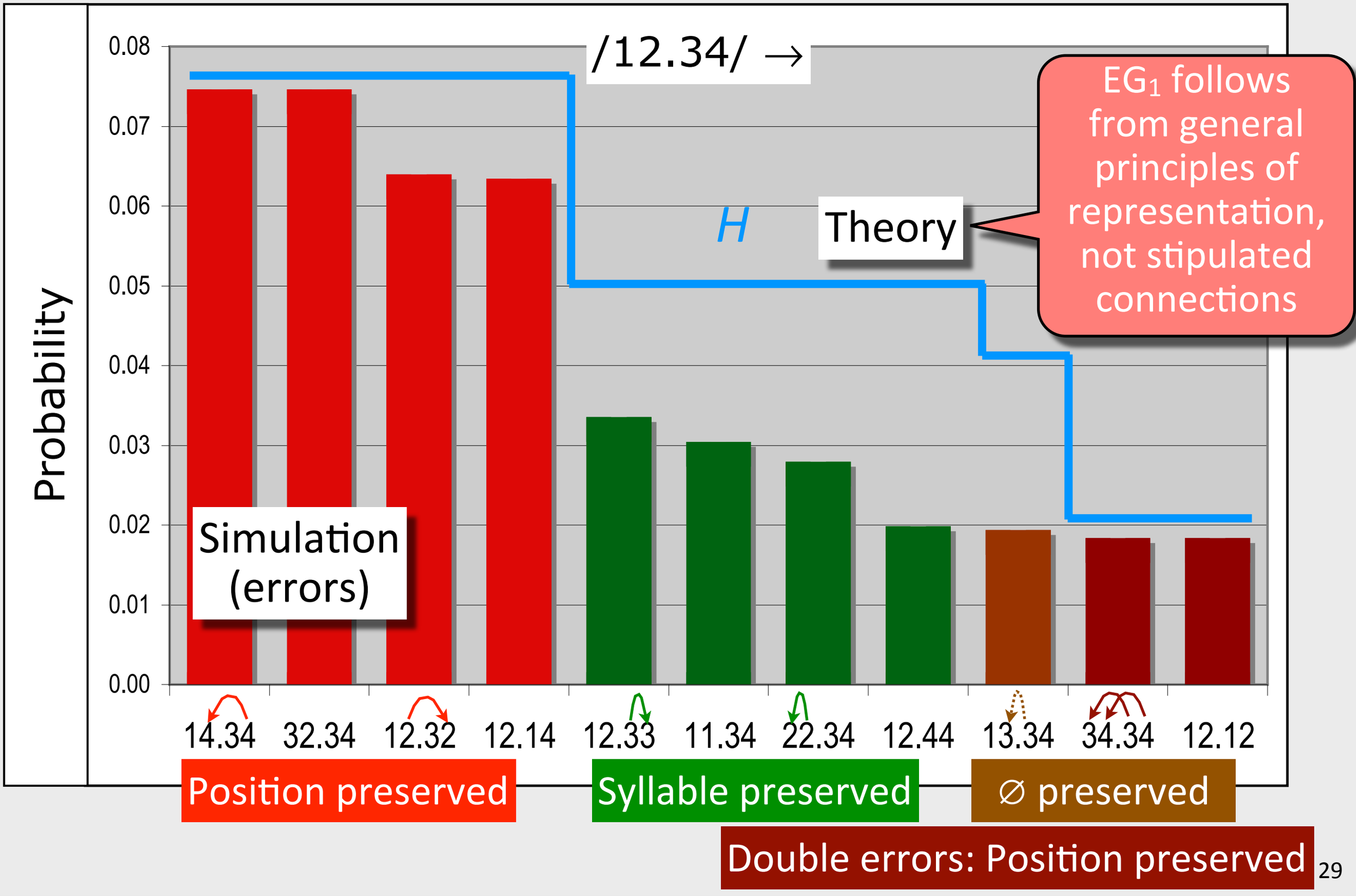
Preserve	Similarity	Simulations
Position	$\mathbf{r}_{\sigma_2} \cdot \mathbf{r}_{\sigma_1}$	0.4
Syllable	$\mathbf{r}_{Ons} \cdot \mathbf{r}_{Cod}$	0.1

$sim(\mathbf{r}_{Ons}, \mathbf{r}_{Cod}) < sim(\mathbf{r}_{\sigma_1}, \mathbf{r}_{\sigma_2})$ ⇐? similarity of consonant behavior across different positions < across different syllables]



Explanation and Harmony

EG_1. Errors tend to preserve position



Gradience in alternations

/rad/ → [rat] ‘wheel’ (German)

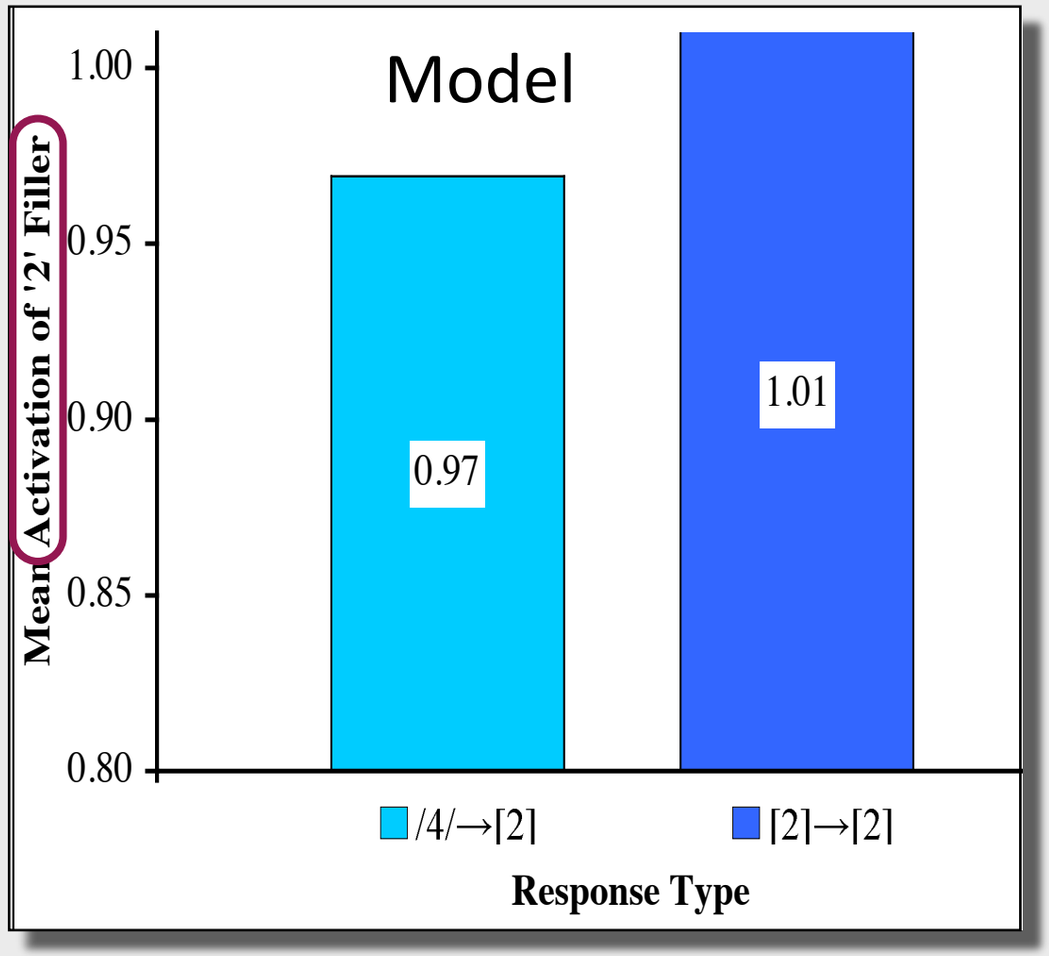
EG$_7$. In alternations, surface forms can show subphonemic traces of the underlying form.*

Alternations: MARK ≫ FAITH

E.g., *d_{+voi}/Coda ≫ FAITH(voi)

Model: ‘4_+’ vs. ‘2_-’ ∼ *d* vs. *t*

*4_+/Coda (1.5) ≫ FAITH (1.0)



Surface forms can show subphonemic ‘traces’ of the underlying form.

Review: Warner, Jongman, Sereno & Kemps, 2004 (*Journal of Phonetics*)

* Factors other than underlying form can also induce similar effects.

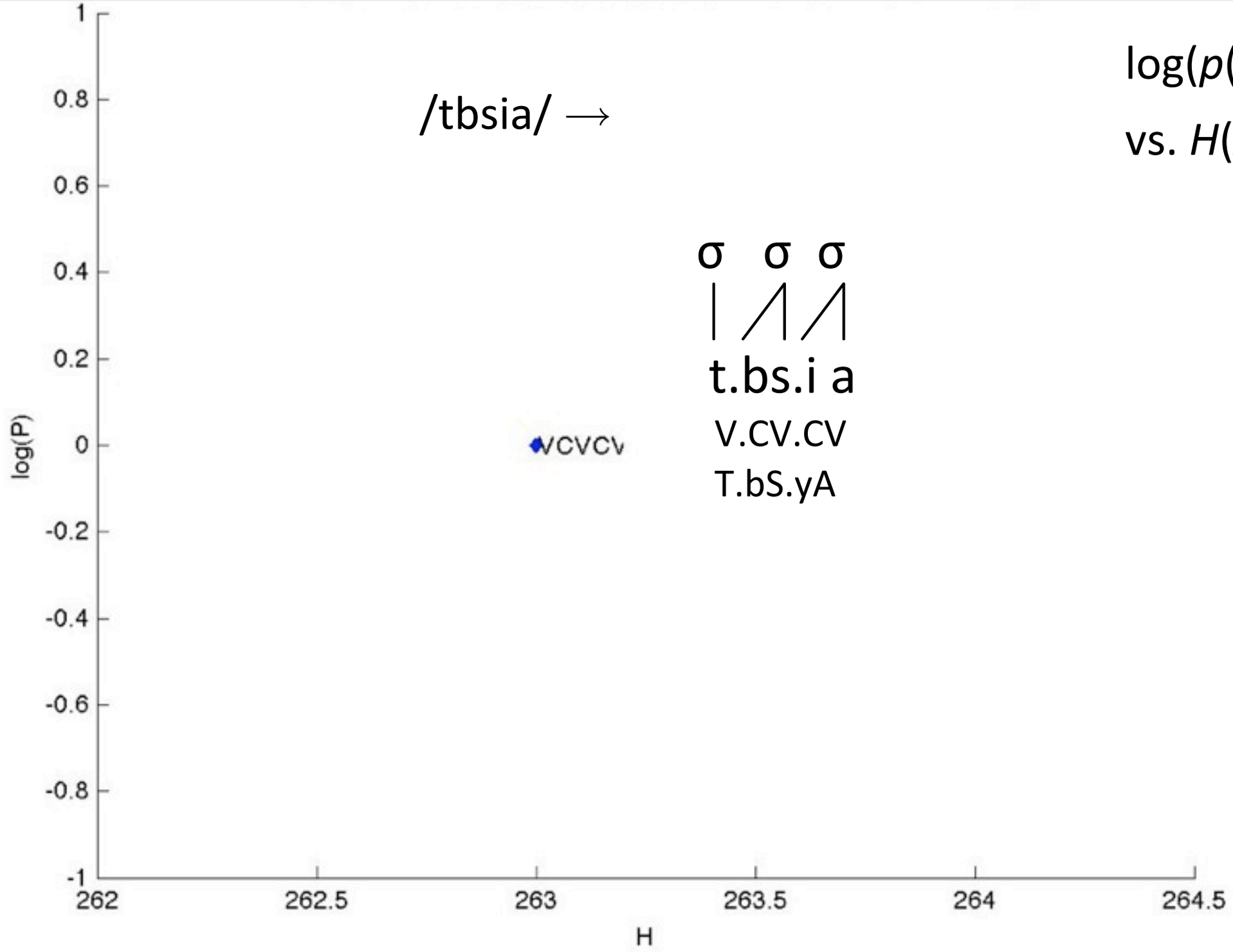
Berber syllabification

Model: derived *directly* from the HG version of the OT analysis of P&S93

/tbsia/ →

log(*p*(*x*))
vs. *H*(*x*)

σ σ σ
| ⟋ ⟋
t.bs.i a
V.CV.CV
T.bS.yA



Berber syllabification

/rmuai/ →

log(*p*(*x*))

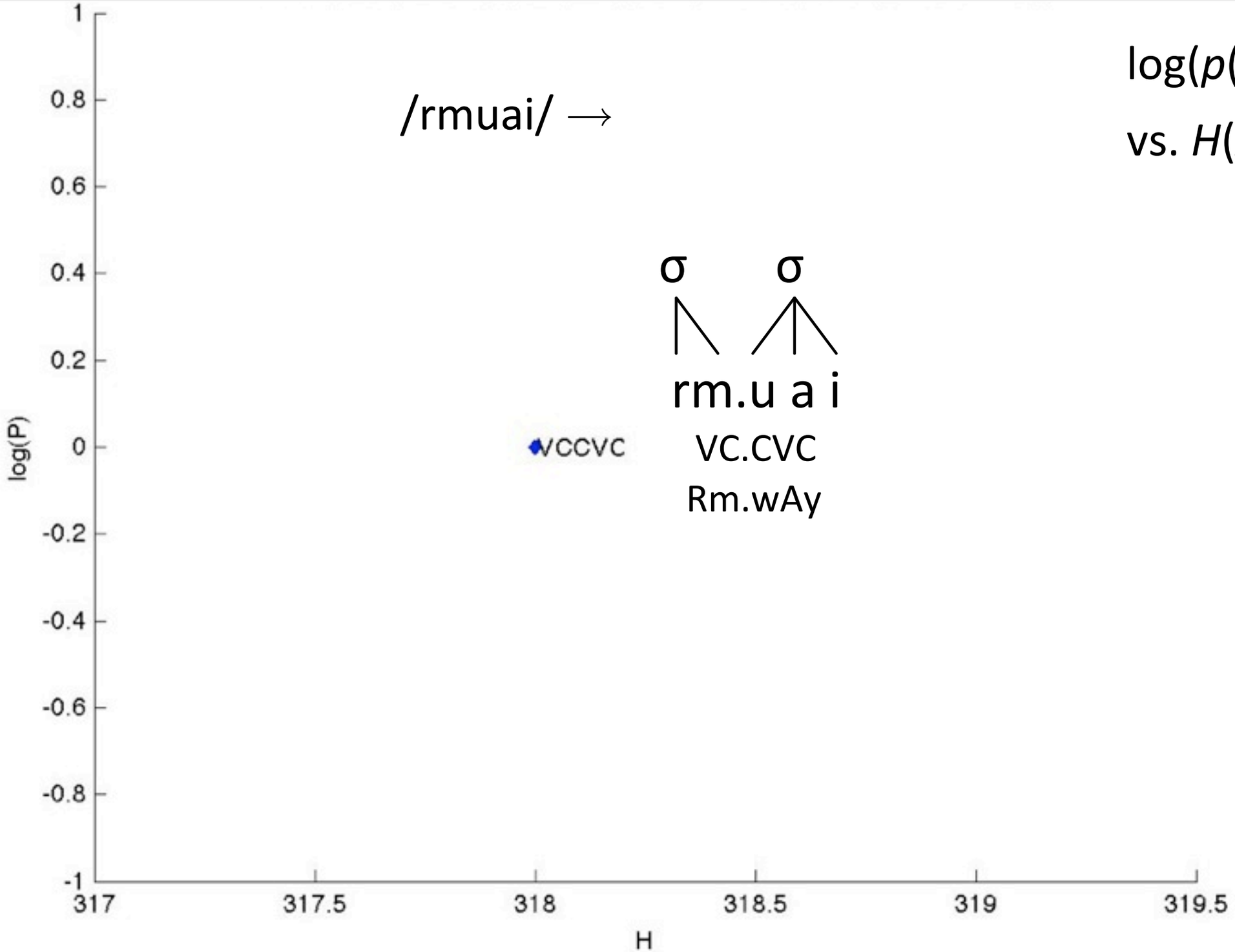
vs. *H*(*x*)

σ σ

rm.u a i

VC.CVC

Rm.wAy



Berber syllabification

/mruai/ →

log(*p*(*x*))

vs. *H*(*x*)

σ σ

m r.u a i

CV.CVC

mR.wAy