Optimization and Quantization in Gradient Symbol Systems: A Framework for Integrating the Continuous and the Discrete in Cognition

Paul Smolensky[1], Matthew Goldrick[2], and Donald Mathis[1]
[1]*Department of Cognitive Science, Johns Hopkins University*
[2]*Department of Linguistics, Northwestern University*

Abstract

Mental representations have continuous as well as discrete, combinatorial aspects. For example, while predominantly discrete, phonological representations also vary continuously, as evidenced by instrumental studies of both grammatically-induced sound alternations and speech errors. Can an integrated theoretical framework address both aspects of structure? The framework we introduce here, Gradient Symbol Processing, characterizes the emergence of grammatical macrostructure from the Parallel Distributed Processing microstructure (McClelland & Rumelhart, 1986) of language processing. The mental representations that emerge, Distributed Symbol Systems, have both combinatorial and gradient structure. They are processed through Subsymbolic Optimization-Quantization, in which an optimization process favoring representations that satisfy well-formedness constraints operates in parallel with a distributed quantization process favoring discrete symbolic structures. We apply a particular instantiation of this framework, λ-Diffusion Theory, to phonological production. Simulations of the resulting model suggest that Gradient Symbol Processing offers a way to unify accounts of discrete grammatical competence with both discrete and continuous patterns in language performance.

The work discussed here was developed as one path for carrying out a research program that was already sketched by 1986[1]:

(1) A PDP approach to cognitive macrostructure

"another notion of levels which illustrates our view … is the notion of levels implicit in the distinction between Newtonian mechanics on the one hand and quantum theory on the other. …

   The basic perspective of this book is that many of the constructs of macrolevel descriptions … can be viewed as emerging out of interactions of the microstructure of

---

[1] Important precedents include Hofstadter (1979, 1985). Other approaches to combining continuous activation spreading and symbolic structure, but without distributed representations (in the sense used here), include the ACT systems (Anderson & Lebiere, 1998), the LISA model (Hummel & Holyoak, 2003) and a range of hybrid architectures (Wermter & Sun, 2000).

distributed models. … although we imagine that rule-based models of language acquisition … may all be more or less valid approximate macrostructural descriptions, we believe that the actual algorithms involved cannot be represented precisely in any of those macrotheories.

    … as we develop clearer understandings of the microlevel models, we may wish to formulate rather different macrolevel models … PDP mechanisms provide a powerful alternative set of macrolevel primitives … [e.g.,] "Relax into a state that represents an optimal global interpretation of the current input." (Rumelhart & McClelland 1986b:125-126)

The present work aims to make mathematically precise the emergence of cognitive macrostructure from its microstructure. In this research, the macrolevel descriptions of grammatical theory, in particular, are taken to be extremely good approximations, but ones in need of microstructural algorithms and the improvements that derive from them (Smolensky & Legendre, 2006). The sense of 'emergence' relevant here is that the new properties of the macrostructure are formally entailed by the basic properties of the microstructure; we do not refer to emergence through learning, and indeed learning plays no role in this article. Emergence of macrostructure, in a range of senses, has been a main theme in the work of Jay McClelland; work particularly relevant to the present paper includes McClelland (1993) as well as numerous articles cited below.

1. Introduction to Gradient Symbol Processing

Our exploration of the emergence of macro- from microstructure is in service of this question: *How do the continuous and the discrete, combinatorial aspects of mental representation interact?* This question looms large in many domains of higher cognition. A few illustrative issues in language are given in (2).

(2) Discrete/continuous interaction: Examples in language

    a. Phonology ([d] vs. [t]) and phonetics (Voice Onset Time = VOT = 20ms vs. 60ms) use discrete and continuous characterizations of knowledge, respectively, but it is widely recognized that there is a great deal of overlap in the substance of this knowledge (Boersma, 1998; Flemming, 2001; Hayes, Kirchner, & Steriade, 2004; Pierrehumbert, 2006). Can we build a formal, unified theory?

    b. In phonological encoding (mapping lexical /roz+s/ 'ROSE+PL' to phonological "roses"[2]), continuous activation-spreading computes outputs that are, to a good approximation, structured combinations of discrete speech sounds (or *segments*)— but these outputs are also gradient in subtle ways (Section 4.2). Can these two aspects be accounted for within a single integrated architecture?

    c. In many arenas of linguistic performance, continuous variables such as frequency and similarity interact strongly with discrete grammatical structure (frequency of [ps] as a syllable onset (3) vs. as a syllable coda; structural similarity in speech

---

[2] We sometimes use "xyz" (with double quotes) in lieu of the International Phonetic Alphabet to denote the mental representation of the pronunciation of the word (or pseudo-word) spelled xyz.

errors (Section 4.3)). Can we derive such interaction from the cognitive microstructure of grammar?

The facets of mental representations under discussion here are those concerning the information passed from one mental process to another—the structure of and relation between states of interacting components of a cognitive system. To make this discussion concrete, most of our discussion will focus on two such components proposed in the architectures of spoken language processing assumed by many researchers: *lexical processing* and *phonological encoding* (Dell, 1986; Garrett, 1975; Goldrick & Rapp, 2007; Levelt, Roelofs, & Meyer, 1999; Stemberger, 1985).

The state of the lexical component is a combinatorial representation composed of the stored sound structures of a set of morphemes chosen by a speaker to communicate a meaning—e.g., /roz/+/s/ for ROSE + PLURAL (slash-delimiters mark lexical representations).

The state of the phonological component is a combinatorial representation composed of a multi-set of phonetic segments related in a particular order and grouped into constituents such as syllables and stress feet—e.g., [PrWd (Ft [σró][σzəz]) "roses"] (square brackets denote phonological representations; Smolensky, Legendre, & Tesar, (2006:473–480) gives a mini-tutorial).

Both the lexical and phonological representations are discrete—to an excellent approximation. We shall see, however, that subtle gradient (i.e., non-discrete) effects are at work in phonological representations, and these are evidenced as small but systematic differences in the continuous representations of phonetics which arise, in our account, as a result of gradient differences in phonological representation (e.g., slightly different durations for vowels preceding [t] in one type of lexical item than in another type; Section 4.2).

In considering the relation between components of the cognitive system, we focus on relatively small time scales. For example, in the context of lexical and phonological processing, we consider a buffer of sufficiently modest size that it is a reasonable approximation to assume that the morphemes it contains are processed in parallel when computing the phonological representation. One parallel step of input-to-output mapping constitutes a single *relaxation* (or *settling*) of a component.

Although specifying the serial aspects of processing is critical for understanding many aspects of cognition (and an important area for future development of this work), it does not provide a general solution to how the discrete and continuous aspects of mental representation interact. Beginning shortly after the PDP books (Rumelhart, McClelland, & the PDP Research Group, 1986; McClelland, Rumelhart, & the PDP Research Group, 1986), much PDP research has implicitly pursued a serial strategy; the processing of structured mental representations (e.g., syntactic structures in sentences) has been modeled by focusing on the temporal relationships between components of the representations (e.g., the order in which lexical items appear in a sentence), and encoding this as serial temporal order of states of a network. These widely deployed models include recurrent network architectures (e.g., Jordan, 1986; Elman, 1990) and, more generally, systems that use iterated function systems to produce fractal encodings of structured mental representations (Tabor, 2000). Although this may accurately characterize some aspects of human cognition, in other

domains processing does not involve a series of strictly temporally-ordered selection points, the overall output being the temporal concatenation of all individually selected elements. In such serial systems, each constituent of an overall combinatorial output is computed in a separate relaxation (e.g., predicting the upcoming word in a sentence). This eliminates the possibility of multiple constituents being computed in mutually-interdependent selection processes. For example, in spoken word perception, listeners persist in representing ambiguous speech sounds over many segments; they do not commit to a single parse of the input until sufficient information is received (McMurray, Tanenhaus, & Aslin, 2009). We focus here on domains such as these.

Pursing the overall approach sketched in (1), we treat the discrete, symbolic, combinatorial characterizations of the inputs and outputs of a cognitive process such as phonological encoding as higher-level approximate descriptions of patterns of activity in a connectionist network: the macrostructure of the system is symbolic, the microstructure is PDP (see Figure 1). In the *Gradient Symbol Processing framework* that we present here, processing consists in continuous movement in a continuous state space of distributed activation patterns, a discrete subset of which constitutes the realizations of symbol structures. To produce an appropriately discrete output by the end of a relaxation, this continuous dynamics must end up at one of these special points—to a good approximation.
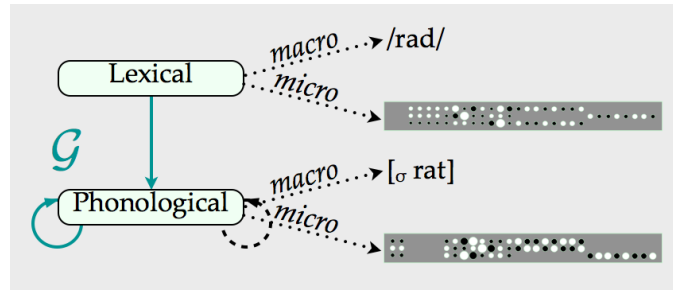

Figure 1. One parallel step of processing—one relaxation—in phonological encoding (German *Rad* 'wheel'). Input and output representations are Distributed Symbol Structures characterized at both macro- and microlevels. Evaluation (solid arrows) and quantization (dashed arrows) dynamics perform Gradient Symbol Processing.

Ignoring for a moment the connections drawn with a dashed arrow, Figure 1 indicates that there are feed-forward connections from the group of connectionist units hosting the lexical representation to that hosting the phonological representation. These, together with a set of recurrent connections among the phonological units, constitute the phonological grammar $\mathcal{G}$, in the following precise sense. If the pattern of activation over the lexical units is the discrete point in state space that is described symbolically as, say, /rad/—the German lexical ('underlying') form for *Rad* 'wheel'—then the solid connections will drive the phonological units towards the pattern of activity which is the discrete state described as (simplifying) [$_\sigma$ rat], the ('surface') phonological form that the grammar $\mathcal{G}$ specifies as the grammatical pronunciation of *Rad* (which, in isolation, is pronounced with a final [t]; this is German *syllable-final devoicing* (or *voicing neutralization*)).

The dashed arrow in Figure 1 indicates another set of recurrent connections among the phonological units; this is the technical core of the new contributions of the work reported here (the remaining techniques were presented as the general Integrated Connectionist/Symbolic cognitive architecture in Smolensky & Legendre, 2006). This second set of recurrent connections drives the phonological units to the discrete set of states that have a combinatorial symbolic description. The proposed theory of the dynamics these connections create is presented in Section 3. The need for such a dynamics is argued in Section 2, which formulates a general computational framework—Gradient Symbol Processing—that employs two functionally distinct but highly interdependent processes: evaluation of a continuum of alternative outputs, and *quantization* of this continuum so as to produce a single discrete combinatorial structure as output (ideally, the best-evaluated—i.e., *optimal*—one). Empirical tests of the theory via specific simple models are discussed in Section 4.

## 2. Discreteness and continuity of mental representations

Our first task is to computationally integrate two facets of mental representations in higher cognitive domains such as phonological production: discrete combinatorial structure and continuous similarity structure.

2.1. Combinatorial structure

In our view, an extremely fruitful hypothesis concerning higher cognition, especially clear in language, is that representations have a crucial property: they are systematic, structured combinations of constituent representations (Fodor & Pylyshyn, 1988; Pylyshyn, 1984). According to many phonological theories, for example, the mental representation of the syllable (σ) that is the pronunciation of *Rad* 'wheel' in German can be described as in (3a) or, equivalently, (3b). Each constituent can be analyzed as a *structural role* instantiated by a *filler* (3c) (Minsky, 1975; Rumelhart, 1975). The constituents of a given representation are connected via the fundamental combinatory operation of symbolic computation, *concatenation* (Partee, ter Meulen, & Wall 1990:432). Crucially for us, by adopting a *filler/role decomposition*, the representation can be viewed as an unordered set of *filler/role bindings* (3d) (Newell, 1980:142; Smolensky, 1990).

(3) Combinatorial structure(simplified) of a syllable σ in four equivalent notations

    a. [σ rat]

    b.
<pre>
                σ
              ╱ | ╲
     onset nucleus coda
        | | |
        r   a   t
</pre>
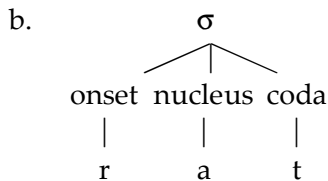
c.   Constituents: roles and fillers

role	*filler*
σ-onset	r
σ-nucleus	a
σ-coda	t

d.   Filler/role bindings: {a/σ-nucleus, t/σ-coda, r/σ-onset}

2.2. *Similarity structure*

Similarity of representations is a central psychological concept, used to explain many cognitive phenomena; a few examples are given in (4).

(4) Similarity-based psychological explanation: examples
   a.   Errors: the more similar an error response E is to the correct form, the more likely E (Goldrick, 2008).
   b.   Categorization: the more similar an item X is to the members/prototype of a category C, the more likely X is to be categorized as C (Kruschke, 2008).
   c.   Priming: the more similar a target T is to a prime P, the greater the facilitation of processing T when it is preceded by P (Gomez, Ratcliff, & Perea, 2008).

For the purposes of psychological explanation, it has proved fruitful to treat representational similarity as a continuous variable—this permits direct prediction of a number of continuous measures important for psychology; such is the case for each of the three citations in (4), as summarized in (5).

(5) Continuous similarity scale →
   a.   probability of error E
   b.   probability of classification as C
   c.   reaction time differences (primed vs. unprimed)

2.3. *Similarity of combinatorial representations*

To apply a continuous similarity notion to combinatorially structured representations S and S', we combine (i) the similarity of the fillers in S to those in S' with (ii) the similarity of the roles they fill. In the theory we adopt below, (6) will hold (see (11)).

(6) If $S = \{f_j/r_j\}_j$ and $S' = \{f'_k/r'_k\}_k$ are filler/role decompositions of structures S and S', then
$$\text{sim}(S, S') = \Sigma_j \, \Sigma_k \, \text{sim}(f_j, f'_k) \, \text{sim}(r_j, r'_k)$$

The contribution of *filler* similarity to psychological explanation of the type (4a) is illustrated in (7) (Shattuck-Hufnagel & Klatt, 1979:52).

(7) From
$$\text{sim}([k], [g]) > \text{sim}([k], [s]),$$

predict that the relative error probabilities of misproducing /kol/ 'coal' as [gol] 'goal' or as [sol] 'soul' obey[3]

$$p(/\underline{k}ol/ \rightarrow [gol]) > p(/\underline{k}ol/ \rightarrow [\underline{s}ol]).$$

The contribution of *role* similarity to psychological explanation of type (4a) is more subtle: see (8) (Vousden, Brown, & Harley, 2000).

(8) From

$$\text{sim}(\sigma_2\text{-onset}, \sigma_1\text{-}\underline{onset}) > p(\sigma_2\text{-onset}, \sigma_1\text{-}\underline{coda}),$$

predict that the relative error probabilities of producing target /kol rid/ 'coal reed' as "role keyed" or as "core lead" obey

$$p(/\text{kol }\underline{r}\text{id}/\rightarrow[\underline{r}\text{ol kid}]) > p(/\text{kol }\underline{r}\text{id}/\rightarrow[\text{ko}\underline{r}\text{ lid}]).$$

Here, the tendency of such speech errors to preserve syllable position is derived from the general principle that if two roles correspond to the same structural position (e.g., onset) within two tokens of a given type (e.g., σ_1 and σ_2), then these roles are more similar than when they correspond to different positions, all else equal. Thus an erroneous output in which [r] appears in the onset of the incorrect syllable ("$\underline{r}$ole") is more similar to the target ("coal $\underline{r}$eed") than is the erroneous output in which [r] appears in the coda of the incorrect syllable ("co$\underline{r}$e"). (See Section 4.3 below.)

2.4. Continuity + combinatorial structure

We propose here a framework, Gradient Symbol Processing, that unifies continuity of representations (and hence continuity of similarity) with combinatorial structure by pursuing a fundamental hypothesis of PDP: that at the microstructural level, mental representations are distributed patterns of activation over n simple numerical processing units—that is, vectors in $\mathbb{R}^n$ (Jordan, 1986a; Rumelhart, Hinton, & McClelland, 1986; Smolensky, 2006a:150–159).

In a vector space such as $\mathbb{R}^n$, the combinatory operation is *linear combination*, i.e., weighted summation or *superposition*. In such a *superpositional combinatorial representation*[4] (van Gelder, 1991), a constituent is a vector—e.g., (1, 2, 3)—and a composite structure is a vector—e.g., (11, 22, 33)—that is the sum of multiple constituent vectors—e.g., (11, 22, 33) = (1, 2, 3) + (10, 20, 30). It is in this precise sense that the output activation pattern in Figure 1 has constituent macrostructure than can be formally characterized as the structure [$_\sigma$rat].

In fact, our representational space is a *Hilbert space*, a vector space with a *dot product* (or inner product) that can be used to define similarity in the standard way (9).

(9) $\text{sim}(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{y} = \Sigma_k x_k y_k = \|\mathbf{x}\| \|\mathbf{y}\| \cos \measuredangle(\mathbf{x}, \mathbf{y})$

(Here $\|\mathbf{x}\|$ is the Euclidean length of $\mathbf{x}$ (i.e., $\sqrt{\Sigma_k x_k^2}$) and $\measuredangle(\mathbf{x}, \mathbf{y})$ is the angle formed in $\mathbb{R}^n$ by $\mathbf{x}$ and $\mathbf{y}$.) That distributed representations inherently encode similarity has long been

---

[3] Here and throughout we use underlining to draw attention to the elements critical in comparisons.
[4] Superpositional *representation* (over constituents) is formally related to, but conceptually distinct from, superpositional *memory* (over exemplars) (Rumelhart & Norman, 1983/1988).

emphasized as a central explanatory feature of PDP (Hinton, McClelland, & Rumelhart, 1986).

2.5. *Filler/role binding with the tensor product*

In the theory we pursue here, the activation pattern realizing a single constituent—a single filler/role binding—is defined as in (10) (Smolensky, 1990).

(10)   [vector realizing filler/role binding] = [vector realizing filler] ⊗ [vector realizing role]

The tensor product ⊗ is a generalization of the matrix outer product; the elements of the vector $\mathbf{x} \otimes \mathbf{y}$ consist of all numbers arising by taking an element of $\mathbf{x}$ and multiplying it by an element of $\mathbf{y}$; e.g., $(1, 2, 3) \otimes (10, 20, 30) = (10, 20, 30; 20, 40, 60; 30, 60, 90)$. Given a distributed representation of fillers and a distributed representation of roles, this yields a distributed representation of constituents in which there are systematic relations between, for example, a given filler in one role and the same filler in a different role (Smolensky, 2006a:175 ff.).

Crucially, the calculus of these *tensor product representations* makes it possible to work at a macrostructural level with *Distributed Symbol Systems* (Smolensky (2006a) gives a tutorial). This allows us to exploit general explanatory principles of continuous similarity (e.g., (4)) in the context of combinatorial representations.

To illustrate this point, consider the use of similarity to explain priming effects in visual word recognition (4c). Relative to dissimilar controls, orthographically similar primes (e.g., *honse* as a prime for HORSE) induce faster lexical decision times (Forster & Davis, 1984). Recent studies have demonstrated *transposition priming*; similar facilitation is observed when a nonword prime equals the target with two letters transposed (e.g. *hosre* for HORSE; Perea & Lupker, 2003). This has been explained by assuming that mental representations of orthographic form are structured such that strings containing the same letter in distinct serial positions (e.g., *sr* vs. *rs* in *hosre* vs. *horse*) have non-zero similarity (Gomez et al., 2008).

Tensor product representations allow us to utilize these explanations within a continuous representational space. We can compute, for example, that if r_1, r_2 are the first and second positions in a letter string, then (11) holds.

(11)   sim(AB, XY) = sim($\mathbf{A} \otimes \mathbf{r}_1 + \mathbf{B} \otimes \mathbf{r}_2$, $\mathbf{X} \otimes \mathbf{r}_1 + \mathbf{Y} \otimes \mathbf{r}_2$)
                    = sim($\mathbf{A},\mathbf{X}$) $\cdot$ sim($\mathbf{r}_1, \mathbf{r}_1$) + sim($\mathbf{B},\mathbf{Y}$) $\cdot$ sim($\mathbf{r}_2, \mathbf{r}_2$)
                    + sim($\mathbf{A},\mathbf{Y}$) $\cdot$ sim($\mathbf{r}_1, \mathbf{r}_2$) + sim($\mathbf{B},\mathbf{X}$) $\cdot$ sim($\mathbf{r}_2, \mathbf{r}_1$)

So if, say, sim(A, B) = 0, sim(A, A) = 1 = sim(B, B), then sim(AB, BA) = 2 sim($\mathbf{r}_1, \mathbf{r}_2$). Thus the similarity of the string AB and its transposition BA will be non-zero if and only if the encoding of position 1 and position 2 "overlap"—have non-zero similarity (i.e., are not orthogonal). This then is the crucial requirement for an encoding scheme for letter strings to predict transposition priming via (4c) (Fischer-Baum & Smolensky, forthcoming; see also Hannagan, Dupoux, & Christophe, in press).

The calculus of Distributed Symbol Systems allows us to abstract away from the particular numbers in activation patterns, numbers which constitute the microstructural representations in a neural network. This calculus, it turns out, enables representations with recursive structure, like that of binary trees, and enables the computation, in a single

massively parallel step of a simple linear associator network, of any mapping in an important class of recursive functions (Smolensky, 2006b:324).

The force of the PDP principle asserting that mental representations are distributed is that *no single unit is devoted to encoding a single symbolic constituent*: we do *not* have "1 constituent = 1 neuron" (nor "1 constituent = 10 dedicated neurons"; Feldman & Ballard, 1982, 1981:209). As we will shortly see, this turns out to be crucial because it means that to produce a discrete output, the job of ensuring that a role is filled by exactly one symbol, with activation level 1, cannot be carried out locally, by a single unit. Producing discrete outputs becomes a major technical challenge (Section 3.1) that turns out to have important conceptual consequences.

Tensor product representations formalize ideas of 'conjunctive coding' already deployed in early PDP models (e.g., McClelland & Kawamoto, 1986), themselves preceded by 'distributed memory models' (Murdock, 1982; Pike, 1984). Tensor products also serve as the basis for a number of connectionist computational architectures making use of 'vector symbolic' representations (Levy & Gayler, 2008). These architectures generally compress full tensor product representations into a smaller vector space, trading space resources for precision (and analyzability)—although (contrary to widespread but misinformed opinion) the size of tensor product representations is *not* in fact problematic.[5] In addition to reducing

---

[5] These compression schemes rely on random patterns over large numbers of units which, on average, are roughly orthogonal; the law of large numbers allows various types of cross-talk to be managed to some degree, when augmented with essential 'clean-up' processes to remove noise. Such schemes are interesting and important for a number of reasons, but not, we think, for the reason normally given: that standard tensor product representations (TPRs) are too large and must be compressed (thereby sacrificing the precise representation, similarity encoding, and depth of analysis that the simple structure of TPRs make possible). The size of TPRs is often greatly exaggerated; for example, the case claimed by Marcus (2001:106) to require $24,300,000 = (10 \cdot 3)^5$ units actually requires $7,280 = 10[3^{5+1} - 1]$ (assuming here and henceforth that the filler vectors are binary, as in most compression schemes).

TPRs are not recommended for use by Google®. But for buffer sizes for which *human* parallel processing is plausible (Section 1), the size of TPRs is generally not excessive. With an alphabet of $32,768 = 2^{15}$ symbols (e.g., words), strings of length 10 require a TPR with 150 (= 15·10) units. With an alphabet of $65,536 = 2^{16}$ symbols, depth-6 binary trees parsing strings of length up to $64 = 2^6$ symbols require $2,032 (= 16[2^{6+1} - 1])$ TPR units. With concept symbols represented as distributed patterns in a 1000-dimensional semantic space (allowing 10^{302} concepts), conceptual structures encoded as binary trees of depth up to 7 (with 128 terminal nodes) require 255,000 TPR units.

Actual cognitive models using "compressed" representations tend, in fact, to be significantly *larger* than their corresponding TPR networks. Plate (2000) uses 2048 units, 1363% larger than the corresponding TPR (180 units). Gayler & Levy (2009) use 10,000 units, 144% larger than the corresponding TPR (4,096 units). The three models discussed in Hannagan, Dupoux, & Christophe (2010), (each with 1000 units), are either 290% or 1463% larger than the corresponding TPRs (256 or 64 units): these models *approximately* encode strings of length up to 8 with an alphabet of 8 symbols; with TPRs, the same 1000 units can *precisely* encode strings of length 50 with an alphabet of $2^{20} > 1$ million symbols. (Even if we require that all fillers be linearly independent, 900 units can encode strings of length 30 with an alphabet of 30 symbols.)

There may well be computational or empirical reasons that noisy, compressed representations (with their concomitant clean-up processes) enable better cognitive models than do TPRs (with no clean-up processes), but to our knowledge such arguments have yet to be provided; size (let alone efficiency) seems unlikely to provide those arguments.

the number of units, nonlinearities have been used to compress the range of activations. An early such architecture deployed the Holographic Reduced Representations of Plate (1991, 2000, 2003). Subsequent developments use a variety of different compression schemes (for reviews: Gayler, 2003; Kanerva, 2009; Smolensky & Tesar, 2006).

2.6. Aside: Why distributed representations?

Because of the import of distributed representations for the subsequent analysis, we momentarily interrupt the main line of argument to list in (12) some of the types of motivations that have led us, like many others, to assume that at the microstructural level mental representations are distributed activation patterns—as opposed to local representations, with activation restricted to a single connectionist unit. We recognize of course that this assumption, while widely accepted, is controversial in some quarters (Barlow, 1972; Bowers, 2002, 2009; Feldman, 1989; Page, 2000).

(12)   Motivations for studying distributed representations
      a.   Neuroscience: population coding is pervasive (Pouget, Dayan, & Zemel, 2000; Sanger, 2003)
         i.   Many stimuli excite a neuron to some degree
        ii.   Many neurons are excited by one stimulus
      b.   Internal (hidden unit) representations arising from connectionist learning are widely distributed (Churchland & Sejnowski, 1992)
      c.   Computationally more powerful in many respects (Hinton, McClelland, & Rumelhart, 1986; Hinton & Anderson, 1981); examples:
         i.   Similarity is directly encoded
           ✦ Similar spelling ⇒ similar pronunciations
        ii.   Number of possible representations is exponentially increased
           ✦ Color: 3 units ⇒ infinitely many hues
        iii.   Acuity is improved
           ✦ Coarse coding: broadly tuned units give higher accuracy

2.7. Generating representations: Continuous activation and blends

In addition to continuous similarity, another continuous facet of mental representations has played an important explanatory role in many cognitive domains, including psycho-linguistics, even in frameworks other than PDP. During computation, a mental representation contains 'partial activation' of alternative structures, activation levels forming a continuum. So, for example, all else equal, perception of spoken word X is slower if many words sound similar to X (Luce & Pisoni, 1998); this is explained by assuming that, because of their similarity to X, these other words become partially active (McClelland & Elman, 1986); they *compete* with the correct word, so it takes longer for the correct word to become fully active, that is, perceived.

    The degree of activation of structure X at time t, $a_X(t)$, can be interpreted broadly as the amount of evidence accrued by time t that X is relevant to the current mental task. That is, $a_X(t)$ is the estimate at time t of the 'goodness' of X in the current context: computing $a_X(t)$ is

a process of evaluation, implemented in networks by continuous spreading-activation algorithms that amount to evidence gathering. During the intermediate stages of processing, mental representations typically contain multiple partially activated structures—a *blend*. Producing a discrete output requires eliminating blends in favor of a single, fully-activated structure: a *pure state*, interpretable macroscopically as a single symbol structure.

As a concrete example, consider the McClelland & Rumelhart (1981; Rumelhart & McClelland, 1982) model of visual letter perception and word recognition. Initially, activation flows from the units denoting features (line segments) in the stimulus to the units denoting letters; in a given position, the unit for the correct letter receives the most activation, but all letters sharing some of the features of the stimulus also receive some activation. Initially, there is a blend in which multiple letters are partially active; the more similar a letter is to the stimulus, the stronger its representation in the blend. The same goes for the representation at the word level.

In a vector space, describing blends is straightforward. If $\mathbf{v}_W$ is the vector encoding a word W, then, say, $0.8\,\mathbf{v}_{ROT} + 0.6\,\mathbf{v}_{ROD}$ is simply a blend of the words ROT and ROD in which the strengths of the words ROT, ROD in the blend are 0.8, 0.6. A *pure* representation, as opposed to a blend, is exemplified by $1.0\,\mathbf{v}_{ROT} + 0.0\,\mathbf{v}_{ROD} = \mathbf{v}_{ROT}$.

Early in the processing of an input, then, mental representations are typically blends. The key question now is, *when a component relaxes into a final output state, are representations blends or pure?* It turns out that the combinatorial structure of representations plays an important role in determining the answer.

2.8. Ambiguity of blends of superpositional combinatorial representations

Consider a mental state **a**, a balanced blend of two syllables, [slɪt] 'slit' and [šrɛd] 'shred'. Assume for simplicity a representation in which the fillers are phonological segments and the roles are *first-segment*, *second-segment*, etc.[6] (as opposed to the more psycholinguistically accurate (3)). Then we have the result in (13).

$$
\begin{aligned}
(13)\quad 0.5\,\mathbf{v}_{[slɪt]} + 0.5\,\mathbf{v}_{[šrɛd]} &= 0.5\,(\mathbf{s}{\otimes}\mathbf{r}_1 + \mathbf{l}{\otimes}\mathbf{r}_2 + \mathbf{\textsc{i}}{\otimes}\mathbf{r}_3 + \mathbf{t}{\otimes}\mathbf{r}_4) + 0.5\,(\mathbf{š}{\otimes}\mathbf{r}_1 + \mathbf{r}{\otimes}\mathbf{r}_2 + \mathbf{\varepsilon}{\otimes}\mathbf{r}_3 + \mathbf{d}{\otimes}\mathbf{r}_4) \\
&= 0.5\,[(\mathbf{s}{+}\mathbf{š}){\otimes}\mathbf{r}_1 + (\mathbf{r}{+}\mathbf{l}){\otimes}\mathbf{r}_2 + (\mathbf{\varepsilon}{+}\mathbf{\textsc{i}}){\otimes}\mathbf{r}_3 + (\mathbf{d}{+}\mathbf{t}){\otimes}\mathbf{r}_4] \\
&= 0.5\,(\mathbf{š}{\otimes}\mathbf{r}_1 + \mathbf{l}{\otimes}\mathbf{r}_2 + \mathbf{\textsc{i}}{\otimes}\mathbf{r}_3 + \mathbf{t}{\otimes}\mathbf{r}_4) + 0.5\,(\mathbf{s}{\otimes}\mathbf{r}_1 + \mathbf{r}{\otimes}\mathbf{r}_2 + \mathbf{\varepsilon}{\otimes}\mathbf{r}_3 + \mathbf{d}{\otimes}\mathbf{r}_4) \\
&= 0.5\,\mathbf{v}_{[šlɪt]} + 0.5\,\mathbf{v}_{[srɛd]}
\end{aligned}
$$

This blend of [slɪt] and [šrɛd] is identical to a balanced blend of [šlɪt] ("shlit") and [srɛd] ("sred"): this state is ambiguous.[7] This is *not* true of a symbolic state representing an equal degree of belief that the word is "slit" or "shred": the concatenatory combination operation of symbolic representation does not lead to the ambiguity we have seen arising from

---

[6] Using *contextual* roles (Smolensky, 1990; essentially, *n*-grams) rather than *positional* roles alters but does not eliminate blend ambiguity. If strings, e.g., ABC, are represented through bigrams, e.g., {BC, AB}, then $\mathbf{v}_{AB} + \mathbf{v}_{XY}$ is an unambiguous mixture, but an even blend of ABC and XBY equals an even blend of XBC and ABY (see also Prince & Pinker, 1988).

[7] Crucially, (under the standard requirement that role vectors be linearly independent) the superpositions involved in a *pure* state do *not* yield ambiguity; e.g., [slɪt] is not ambiguous with [stɪl], because $\mathbf{v}_{[slɪt]} = \mathbf{s}{\otimes}\mathbf{r}_1 + \mathbf{l}{\otimes}\mathbf{r}_2 + \mathbf{\textsc{i}}{\otimes}\mathbf{r}_3 + \mathbf{t}{\otimes}\mathbf{r}_4 \neq \mathbf{s}{\otimes}\mathbf{r}_1 + \mathbf{l}{\otimes}\mathbf{r}_4 + \mathbf{\textsc{i}}{\otimes}\mathbf{r}_3 + \mathbf{t}{\otimes}\mathbf{r}_2 = \mathbf{v}_{[stɪl]}$ (Smolensky, 1990).

superpositional combination. This ambiguity also does *not* arise with completely local connectionist representations, in which the entire string [slɪt] is represented by a single unit, completely dissimilar from the representation of [šlɪt]. Nor does ambiguity arise with (linearly independent) distributed representations of atomic (non-combinatorial) content.

Suppose that the representation in (13) is an intermediate state in the phonological component of speech perception; in this blended state, the phonological component has not yet committed to a single interpretation of the input. In a symbolic system, this component could produce as output a list of possible interpretations, each with an associated degree of belief or strength of evidence, and let downstream processes use their knowledge to choose among them. But in our PDP system, this is not an option. For it is exactly the phonological component that has the knowledge that "shlit" and "sred" are not possible English words; [šl] and [sr] are not possible English syllable onsets. So for the phonological system to output the blend (13) is for that system to fail to apply its knowledge; downstream components may not (and presumably do not) have the knowledge needed to reject the possible interpretations "shlit" and "sred", so phonology cannot pass this decision on to them. In order for the phonological component to express its knowledge, it cannot output a blend like (13): it must choose among the alternative interpretations that it knows to be possible English words, committing to either the pure output "slit" or the pure output "shred": (14).

(14) With superpositional combinatorial representations, to apply its knowledge a process must resolve blends and relax into a pure state.

It remains possible (and often necessary) for a process to choose its pure output based on continuous input from other processes that are running in parallel.

In Gradient Symbol Processing, a state of a component that is very close to a pure state will have nearly identical effects on other components as would that pure state itself. So in (14) we intend 'a pure state' to mean 'a state very close to a pure state', putting aside for now the question of whether approximately-but-not-exactly-pure states are cognitively relevant.

The process of settling on a single, (approximately) pure, symbolically-interpretable state from a continuum of alternatives will be called quantization. Quantization is the key new ingredient in Gradient Symbol Processing.

2.9. The Optimization-Quantization Principle

Combining the conclusions of Sections 2.7 and 2.8 gives (15).

(15) In combinatorial domains, a mental process consists of
    a. evaluating a continuum of alternative possible output representations, and
    b. *quantizing* to produce a pure symbolic one—ideally, the best-evaluated or *optimal* one.

As noted in Section 2.5, because of the principle "1 symbol ≠ 1 neuron", outputting a pure state is not as straightforward in a PDP system as in local connectionist networks such as the

McClelland and Rumelhart (1981) model considered above. In local models, mutual inhibition between individual units that encode mutually inconsistent interpretations suffices to perform the quantization operation. Early in computation, the state of a component is a rich blend, but mutual inhibition eventually effects a choice among alternatives, with the alternative receiving the most activation from the input (the best-evaluated or optimal choice) being the favored outcome. The localized piece of hardware—abstract neuron—devoted to encoding each symbol is responsible for ensuring that at the end of computation, the activation of that neuron = symbol is either 1 or 0. With *distributed* combinatorial representations, the "winner-take-all" dynamics that assures that each role has at most one filler (with activation 1.0) requires more than simple mutual inhibition. In Section 3 we take up this challenge.

2.10. Representations in Gradient Symbol Processing: Summary

We summarize these remarks concerning mental representations in (16).

(16)   In higher cognition, mental representations form a Distributed Symbol System:
  a. They carry information between mental processes.
  b. They have combinatorial structure.
  c. They form a continuous space
      ✦ of continuous blends
      ✦ with continuous similarity relations.
  d. Those *ultimately* output by a mental process component are pure (unambiguous).
  e. They are produced by
      ✦ evaluation/optimization, and
      ✦ quantization.

3. Processing: Subsymbolic Optimization-Quantization

In this section we develop a theory of the technical apparatus instantiating Gradient Symbol Processing; this system must perform the optimization and quantization processes needed to output a pure, ideally correct, combinatorial representation. The goal is a theory of processing that allows grammatical knowledge to be effectively exploited, within an activation-based computational architecture of the sort that has become the workhorse of psycholinguistic research. We begin with quantization.

3.1. Quantization: Projecting to the grid

The quantization process can be viewed as projecting the representational state to the *grid* formed by pure representations. Figure 2 shows a 2-dimensional slice through a high-dimensional continuous space of syllable representations. The square of four dots is the grid slice: each dot corresponds to a pure syllable such as [dɛt]. Between and around the dots are states that are blends; one such blend is shown in the figure, but there is a continuum of blends filling out an entire 2-d plane. Since the representations are distributed, *each point of the grid corresponds to a distributed pattern*, a vector comprising *n* activation values.
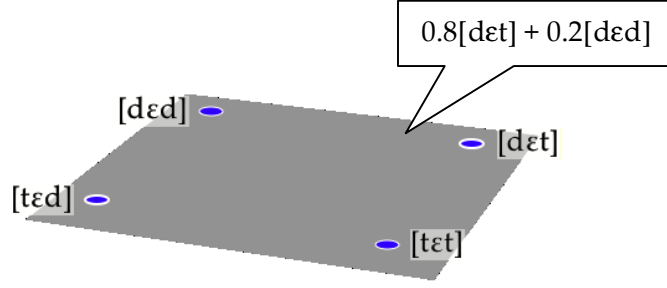
Figure 2. The four dots constitute a slice of the grid of pure states for CVC syllables.

We employ a spreading activation algorithm—a continuous quantization dynamics $\mathcal{D}_Q$—that creates an *attractor* at all and only the points of the grid, using the competitive Lotka-Volterra equation (Baird & Eeckmann, 1993:Sec. 2.6)[8]. This dynamics is isotropic, so that all attractors are equivalent; it is the optimization dynamics discussed below, not the quantization dynamics, that pushes the system toward the preferred (optimal) attractor basin. $\mathcal{D}_Q$ is a distributed non-linear winner-take-all dynamics, achieving a competitive effect like that of lateral inhibition but having attractors that are *distributed* activation patterns as opposed to states with activation localized to a single unit. This dynamics is implemented by recurrent connections among the units hosting the phonological representation; these are the connections indicated by the dashed arrow in Figure 1.

3.2. Optimization I: Grammars as numerical evaluation functions

Putting aside quantization for the moment, we pass to evaluation/optimization. In phonological production, the evaluator of alternative outputs is the phonological grammar $\mathcal{G}$. The key to incorporating grammar into a continuous PDP network is to realize $\mathcal{G}$ as a numerical *Harmony function* $H_\mathcal{G}$; this is called a *Harmonic Grammar* (Legendre, Miyata, & Smolensky, 1990, 2006; Pater, 2009). The arguments to the function $H_\mathcal{G}$ are (i) a lexical form, such as /rad/ (German 'wheel'), and (ii) a candidate pronunciation, e.g., [rat]. The numerical value $H_\mathcal{G}$(/rad/, [rat]) is the grammar's evaluation of how good [rat] is as a pronunciation of /rad/. This is computed by grammatical well-formedness constraints such as those shown in (17).[9]

---

[8] The dynamical equation is

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

and **M** = **F**⊗**R**, with **F** = matrix of filler (symbol) patterns; **R** = matrix of role (position) patterns. δ is the Kronecker delta: $\delta_{jk} \equiv [1 \text{ IF } j=k \text{ ELSE } 0]$

[9] Our discussion adopts the standard assumption that German stops like /d,t/ differ in the feature [voice]; use of the feature [spread glottis] instead (Jessen & Ringen, 2002) would change nothing here.

(17)   Harmonic Grammar tableau for German 'wheel'

weights:	3	2	$H_{\mathcal{G}}$
/rad/→	MARK$_{voi}$	FAITH$_{voi}$	
a. [rad]	*		−3
b. ☞ [rat]		*	−2

In (17) we consider two alternative pronunciations—*candidates*—*a* and *b*; candidate *b* is correct for the German grammar. The constraint MARK$_{voi}$ is violated by final voiced stop consonants like [d].[10] The star beneath MARK$_{voi}$ in row *a* indicates that the candidate [rad] violates that constraint. The final voiceless [t] of [rat] does not violate MARK$_{voi}$ so there is no star in the MARK$_{voi}$ column of row *b*. The constraint FAITH$_{voi}$ requires that the pronounced form be faithful to the segments' voicing features in the lexical form; this is violated by [rat] because it is not faithful to the voicing in the lexical form's final /d/, hence the star in row *b*. The candidate [rad], in contrast, satisfies FAITH$_{voi}$.

For this lexical form /rad/, the two constraints here *conflict* in the technical sense that no candidate pronunciation satisfies them both; the competition goes to the candidate violating the *weakest* constraint. For a Harmonic Grammar has a *weight* for each constraint; in (17), FAITH$_{voi}$ is weakest because its weight, 2, is lower than the weight, 3, of MARK$_{voi}$. So the optimal candidate is *b*, indicated by the pointing finger. The Harmony of the pair (/rad/, [rat]) is −2: starting from 0, each violation lowers the Harmony by an amount equal to the weight of the constraint violated. Thus the Harmony of *a*, $H_{\mathcal{G}}$(/rad/, [rad]), is −3; the highest-Harmony option, the optimal output, is *b*, [rat], with Harmony −2.

It is a characteristic of the German grammar that final lexical /d/ is pronounced [t]: this is because in this grammar, MARK$_{voi}$ is stronger than FAITH$_{voi}$. In the English grammar, however, the reverse is true, and final lexical /d/ is pronounced faithfully, as [d]. This bit of cross-linguistic variation between English and German consists in two different strategies (encoded in weights) for resolving the conflict between two constraints.

This framework, Harmonic Grammar, quickly gave rise to *Optimality Theory* (Prince & Smolensky, 1991, 1993/2004), in which constraint strength is grammatically encoded as a rank within a hierarchy, as opposed to a numerical weight (see Legendre, Sorace, & Smolensky, 2006 for comparisons). Optimality Theory hypothesizes that the grammatical constraints are the same in all languages, that only the relative strengths of these constraints—only the grammars' means of resolving constraint conflict—differ. This means it is possible to formally compute the cross-linguistic typology of possible grammars from a hypothesized set of constraints. Viewing grammars (phonological, syntactic, semantic, …) as Harmony optimizers proves quite useful for linguistic theory (see the electronic archive http://roa.rutgers.edu/). This perspective is also crucial for relating grammar to PDP.

---

[10] In traditional linguistic terminology, a dispreferred element like [d] is called *marked* (Jakobson, 1962; Trubetzkoy, 1939/1969); here, this means it violates the well-formedness constraint MARK$_{voi}$.

3.3. Optimization II: Networks as optimizers

The upshot of the previous subsection is that the output of the phonological encoding process (a pronunciation) should be the representation that maximizes Harmony, given its input (a lexical representation). How can such optimal states be computed?

Among the earliest major results about the global properties of PDP networks is that summarized in (18) (Cohen & Grossberg, 1983; Golden, 1986, 1988; Hinton & Sejnowski, 1983, 1986; Hopfield, 1982, 1984; Smolensky, 1983, 1986; for a tutorial, see Smolensky 2006b).

(18)   For many types of neural network $\mathcal{N}$, local rules for spreading activation have an emergent property:
   a.   the Harmony $H_{\mathcal{N}}$ of the network as a whole increases over time, where
   b. $H_{\mathcal{N}}(\mathbf{a})$ is the *well-formedness* of the activation pattern $\mathbf{a}$ spanning the entire network—the extent to which $\mathbf{a}$ satisfies the micro-constraints encoded in the connections and units—computed as:

   $H_{\mathcal{N}}(\mathbf{a}) \equiv H^0{}_{\mathcal{N}}(\mathbf{a}) + H^1{}_{\mathcal{N}}(\mathbf{a})$ where

   $H^0{}_{\mathcal{N}}(\mathbf{a}) \equiv \Sigma_{\beta\gamma}\, a_\beta W_{\beta\gamma} a_\gamma$ is the *core Harmony*, which depends only the connection weight matrix $\mathbb{W} \equiv \{W_{\beta\gamma}\}$ of $\mathcal{N}$[11], and

   $H^1{}_{\mathcal{N}}(\mathbf{a}) \equiv -\Sigma_\beta \int^{a_\beta} f^{-1}(a)\, da$ is the *unit Harmony*, which depends only on the activation function f of the units in $\mathcal{N}$.

   c.   An example of a micro-constraint encoded by a weight is "$W_{\beta\gamma} = -5$", which encodes the constraint "units β and γ should not be active simultaneously (strength = 5)"

Such networks, then, compute optimal representations: Harmony maxima. Whereas *deterministic* spreading activation algorithms lead to *local* Harmony minima—states with higher Harmony than any neighboring state—computing *global* Harmony maxima requires *stochastic* spreading activation algorithms, which exploit randomness. And it is the global Harmony maxima we need for grammatical outputs. For our stochastic Harmony-maximizing network, we choose a simple *diffusion process* (Movellan, 1998; Movellan & McClelland, 1993): a probabilistic search algorithm that increases Harmony by gradient ascent on average, but with random deviations superimposed; the variance of these deviations is proportional to T (the 'temperature'), a parameter which decreases to 0 during computation. This process, called $\mathcal{D}_{\mathcal{G}}$, is defined in (19), which also states the relevant emergent property of this process.

(19)   The random process defined by the stochastic differential equation[12]

---

[11] We assume the presence of a 'bias unit' with constant activation value $a_0 = 1$; then each weight $W_{\beta 0}$ functions as a bias on unit β. This just simplifies notation.

[12] The difference equation used in the computer simulations is

$$\Delta a_\beta(t + \Delta t) = \sum_\gamma W_{\beta\gamma} a_\gamma(t)\Delta t + \sqrt{2T\Delta t}\,\mathcal{N}_t(0,1)$$

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

converges to a probability distribution in which the probability of an activation pattern **a** is

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

so that as $T \to 0$, the probability that the network is in the globally-maximum-Harmony state approaches 1.

Note that the stochastic aspect of this dynamics, the 'thermal noise', is responsible for producing *correct* responses—for finding *global* Harmony optima. Because, when given limited processing time, these methods are not guaranteed to succeed, this dynamics will sometimes produce errors: but not because noise or damage—unmotivated for the correct functioning of the system—has been injected for the sole purpose of generating errors.

3.4. Optimization III: Networks as grammars

Section 3.2 showed how to formalize a grammar $\mathcal{G}$ as a numerical function, $H_\mathcal{G}$—a measure of *grammatical* Harmony (well-formedness), the discrete global optima of which are the grammatical representations. Section 3.3 showed how stochastic neural networks can compute globally optimal representations, with respect to the *network* Harmony function $H_\mathcal{N}$. These results concerning maximization of macrostructural $H_\mathcal{G}^{[macro]}$ and microstructural $H_\mathcal{N}^{[micro]}$ well-formedness can be combined because of yet another result:

(20) Given a second-order Harmonic Grammar $H_\mathcal{G}$, we can design a neural network $\mathcal{N}$ such that for any representation *s* on the grid of pure states:

$$H_\mathcal{N}^{[micro]}(\mathbf{s}) = H_\mathcal{G}^{[macro]}(\mathbf{a_s}),$$

where **s** is the symbolic macrolevel description of *s* and $\mathbf{a_s}$ is the activation vector realizing **s**, the numerical values of which constitute the connectionist microlevel description of *s* (Smolensky, 2006c:330 ff.)

A Harmonic Grammar is 'second order' if each individual constraint considers no more than two constituents at a time (as is the case for $FAITH_{voi}$ and $MARK_{voi}$ in (17)). In the theory we propose here, the second-order constraint $\mathbb{C}_{AB}[h]$ that assesses a Harmony reward of *h* (negative if a penalty) for each co-occurrence of constituents A and B is encoded as the weight matrix $\frac{1}{2}h[\mathbf{v}_A\mathbf{v}_B^T + \mathbf{v}_B\mathbf{v}_A^T]$; a first-order constraint $\mathbb{C}_A[m]$ assessing Harmony *m* for each occurrence of A is encoded as the bias vector $m\mathbf{v}_A$. Formal languages (defined by rewrite rules, e.g. (21))—at all complexity levels of the Chomsky Hierarchy—can be specified by second-order Harmonic Grammars (Hale & Smolensky, 2006). A re-write rule such as S → N V (S a start symbol) is implemented as the constraints $\{\mathbb{C}_{S_x}[-2], \mathbb{C}_{N_x}[-1], \mathbb{C}_{V_x}[-1], \mathbb{C}_{S_xN_{0x}}[+2], \mathbb{C}_{S_xV_{1x}}[+2]\}_x$, where A_x is the constituent with filler *A* bound to the role of tree

---

where each $\mathcal{N}(0, 1)$ is a pseudo-random draw from a standard normal distribution; the variance of random disturbances is thus 2*T*.

position *x*; 0*x* and 1*x* denote the left- and right-child of node *x* (Smolensky, 2006a:184). The weight matrix $\mathbb{W}_{\mathcal{G}}$ implementing the second-order Harmonic Grammar $\mathcal{G}$ is simply the sum (superposition) of all connection weights and biases contributed by all the rules of $\mathcal{G}$. Following (18b), the full Harmony function $H_{\mathcal{G}}$ consists in this core contribution $\mathbf{a}^{\mathrm{T}}\mathbb{W}_{\mathcal{G}}\mathbf{a} \equiv H_{\mathcal{G}}^0(\mathbf{a})$ from the rules of $\mathcal{G}$ plus a term H^1 that depends not on the grammar but on the activation function of the units. Adopting the simplest choice, linear units, gives $H^1(\mathbf{a}) = -\tfrac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{a}$

In general, the state in $\mathbb{R}^n$ with highest evaluation—with maximal Harmony—proves to be not a pure structure but a blend of well-formed constituents.[13] So in addition to the Harmony-maximizing optimization dynamics $\mathcal{D}_{\mathcal{G}}$ pushing the representation towards grammatical well-formedness, the discretizing, quantization dynamics $\mathcal{D}_Q$ discussed in Section 3.1 is truly needed in order to push the representation towards the grid—to produce a pure response.

To complete the micro-/macro- integration, we now elaborate Figure 1, giving Figure 3.
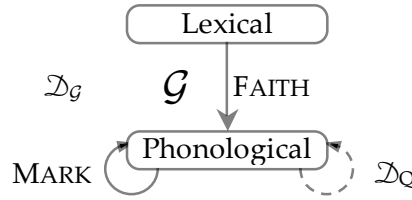

Figure 3. The functional interpretation of the dynamics.

The solid arrows encode the grammar $\mathcal{G}$: the connections between the lexical and phonological components encode the FAITHFULNESS constraints (requiring a match, like FAITH$_{\mathrm{voi}}$ in (17)), while the connections within the phonological component encode the MARKEDNESS constraints (requiring good sound structure, like MARK$_{\mathrm{voi}}$ in (17)). Together these solid-arrow connections generate the optimization *dynamics* $\mathcal{D}_{\mathcal{G}}$, which favors

---

[13] As pointed out to us by Colin Wilson, this can be understood through the following concrete example. Consider a dimension of activation space *a* encoding the [±voice] feature of the final consonant in (17) ([d] vs. [t]). FAITH$_{\mathrm{voi}}$ (strength φ) favors higher values of *a* (i.e., [+voice], matching the lexical form /rad/) while MARK$_{\mathrm{voi}}$ (strength μ) favors lower values of *a* (i.e., [–voice]); and μ > φ. It is not surprising that the optimal compromise turns out to be a value that is primarily low, but pulled up somewhat relative to the situation where the force from FAITH$_{\mathrm{voi}}$ is downward (/rat/). This is because the two constraints contribute to $H_{\mathcal{G}}^0$ bias terms φ*a* and –μ*a*, so $H_{\mathcal{G}}(a) = H_{\mathcal{G}}^0(a)^0 + H_{\mathcal{G}}^1(a) = \varphi a - \mu a - \tfrac{1}{2}a^2$. The scale of {φ, μ} is arbitrary, so we can choose them to satisfy φ + μ = 1, in which case we can rewrite the Harmony as $H_{\mathcal{G}}(a) = -\tfrac{1}{2}\varphi[a-1]^2 - \tfrac{1}{2}\mu[a-(-1)]^2 + \tfrac{1}{2}$, which can be interpreted as follows. A penalty of strength φ is paid for the deviation of *a* from a target +1, and a penalty of strength μ for deviation of *a* from –1: FAITH$_{\mathrm{voi}}$ pushes towards a target +1, MARK$_{\mathrm{voi}}$ towards –1. (These targets are the values of *a* that maximize Harmony when each constraint is present in isolation.) The value of *a* maximizing $H_{\mathcal{G}}(a)$ is easily seen to be $a^* = \varphi - \mu = \varphi \cdot (1) + \mu \cdot (-1)$, a weighted average of the targets. So, e.g., for (φ, μ) = (0.1, 0.9), we have $a^* = 0.1 - 0.9 = -0.8$. On the discrete 'grid' {1, –1}, the optimal choice is simply *a* = –1, and the effect of the weaker force is null; in the continuous state space, the optimum reflects all forces. In general, the optimum is a blend of constituents favored by various constraints; in Section 3.5, for example, the Harmony optimum is an equal blend of both grammatical trees.

representations that are well formed under $\mathcal{G}$. The dashed-arrow connections generate the *quantization dynamics* $\mathcal{D}_Q$ of Section 3.1, which favors grid states—pure discrete structures.

3.5. *The Problem of Mutually-Dependent Choices*

How must the optimization dynamics $\mathcal{D}_G$ and quantization dynamics $\mathcal{D}_Q$ be combined? To address this important issue, it proves easier to shift our working example to one in syntax—the simplest, stripped-down case adequate to illustrate the key problem.

The grammar $\mathcal{G}$ in (21a) generates a language $\mathcal{L}$ containing only two sentences, the trees in (21b). From the perspective of Harmonic Grammar, the grammatical sentences of $\mathcal{L}$ are those trees that have maximal Harmony, given no input: both trees in (21b) have the same, maximal Harmony value, while all other trees, e.g., [s Is Al] or [s Al Al], have lower Harmony. This grammar involves only MARKEDNESS constraints and the lower component of Figure 3; there is no input and hence no need for FAITHFULNESS or even an upper component. (The lower component is now computing a syntactic rather a phonological structure, but formally the model is the same.) When we run our network, it should (with high probability) end up in a grid state corresponding to one of the two trees of $\mathcal{L}$.

(21) a. A nanogrammar $\mathcal{G}$ b. Its nanolanguage $\mathcal{L}$

> Start symbols: {S, S2}   S   S2
> S   → Al Is   ⟋‾‾‾⟍ = [s Al Is]   ⟋‾‾‾⟍ = [s2 Is Al]
> S2 → Is Al   Al   Is   "Al is."   is   Al   "Is Al?"

The maximum-Harmony continuous state for this grammar turns out to be of the form α([s Al Is] + [s2 Is Al]): this is an equal blend of the two grammatical trees but is not a discrete state itself: each role has two fillers, one corresponding to each valid tree. This blend has higher Harmony than either of the two pure states in $\mathcal{L}$. This is typical: blends of well-formed structures have higher Harmony than pure grammatical structures (see footnote 13). So while the optimization dynamics is pushing the network towards a particular blend state, the quantization dynamics is pushing (isotropically) towards all pure grid states. *Among those pure states*, the highest-Harmony trees are those of $\mathcal{L}$. We need the optimization and quantization dynamics to coordinate in such a way as to drive the network to one of those two optimal grid states.

To achieve this, as the quantization dynamics is forcing a choice of a single filler for each role, the optimization dynamics must ensure that the choices made in different roles are mutually compatible according to the grammar. If the network starts to favor, say, Is for the left-child role, then it must also be driven to favor S2 for the root node role as well as Al for the right-child role. The choices among fillers for each of the three roles, effected by the quantization dynamics, are *mutually dependent*; the dependencies are determined by the grammar, that is, are encoded in the optimization dynamics. Thus the optimization dynamics $\mathcal{D}_G$ and the quantization dynamics $\mathcal{D}_Q$ must operate *simultaneously*.

But in order for the final state to be a grid state, the quantization dynamics must be dominant by the end of the relaxation process: the optimization dynamics is opposing the

quantization dynamics' push to the grid. To meet these requirements, we have adopted the simplest solution we could devise: the λ-*method*.

(22)   The λ-method for combining optimization and quantization

The total dynamics $\mathcal{D}$ is a weighted superposition of the optimization and quantization dynamics, with the weight shifting gradually from optimization to quantization. As computation time t proceeds, the weighting parameter λ_t goes from 1 to 0, and the total dynamics shifts gradually from pure optimization to pure quantization. At time t,

$$\mathcal{D}_t = \lambda_t \, \mathcal{D}_\mathcal{G} + (1 - \lambda_t) \mathcal{D}_Q$$

(That is to say, the rate/direction of change of the activation vector over time is a λ_t-weighted sum of the rates/directions of change specified by the two dynamics.)

We can visualize the λ-method as in Figure 4. As $\lambda \to 0$, the Harmony surface in effect grows steeper and steeper peaks at the grid points, as blend states are penalized more and more. ("In effect" because $\mathcal{D}_Q$ is not actually the gradient of any Harmony function; these figures are schematic, as are the λ values.) The network state is like an ant climbing uphill as the surface beneath constantly shifts; the goal is to end up at the highest peak.

3.6. Computation in Gradient Symbol Processing: Summary

We summarize these conclusions concerning mental processes in (23).

(23)   Mental processing principles of Subsymbolic Optimization-Quantization
   a.   At the macro-level, evaluation of potential outputs is via a Harmony function encapsulating a grammar $\mathcal{G}$: $H_\mathcal{G}$.
   b.   At the micro-level, optimization is performed by spreading activation while lowering randomness to zero, thus maximizing $H_\mathcal{N}$. This dynamics is $\mathcal{D}_\mathcal{G}$; its attractor is a blend of well-formed constituents.
   c.   On the grid of pure outputs, $H_\mathcal{G} = H_\mathcal{N}$.
   d.   Quantization is performed by a dynamics $\mathcal{D}_Q$ that creates an attractor at every grid point.
   e.   Optimization and quantization run in parallel: the full dynamics is a superposition of them, weighted increasingly toward $\mathcal{D}_Q$ as computation proceeds.
   f.   *The only discrete representation ever evaluated—the only one ever <u>constructed</u>—is the output itself.*

The particular instantiation of Subsymbolic Optimization-Quantization we have proposed here is λ-*Diffusion Theory*, summarized in (24).

(24)   λ-Diffusion Theory (an instance of Subsymbolic Optimization-Quantization)
   a.   *Optimization:* by diffusion dynamics (19) with dynamic randomness
   b.   *Quantization:* by competitive Lotka-Volterra dynamics (note 8)
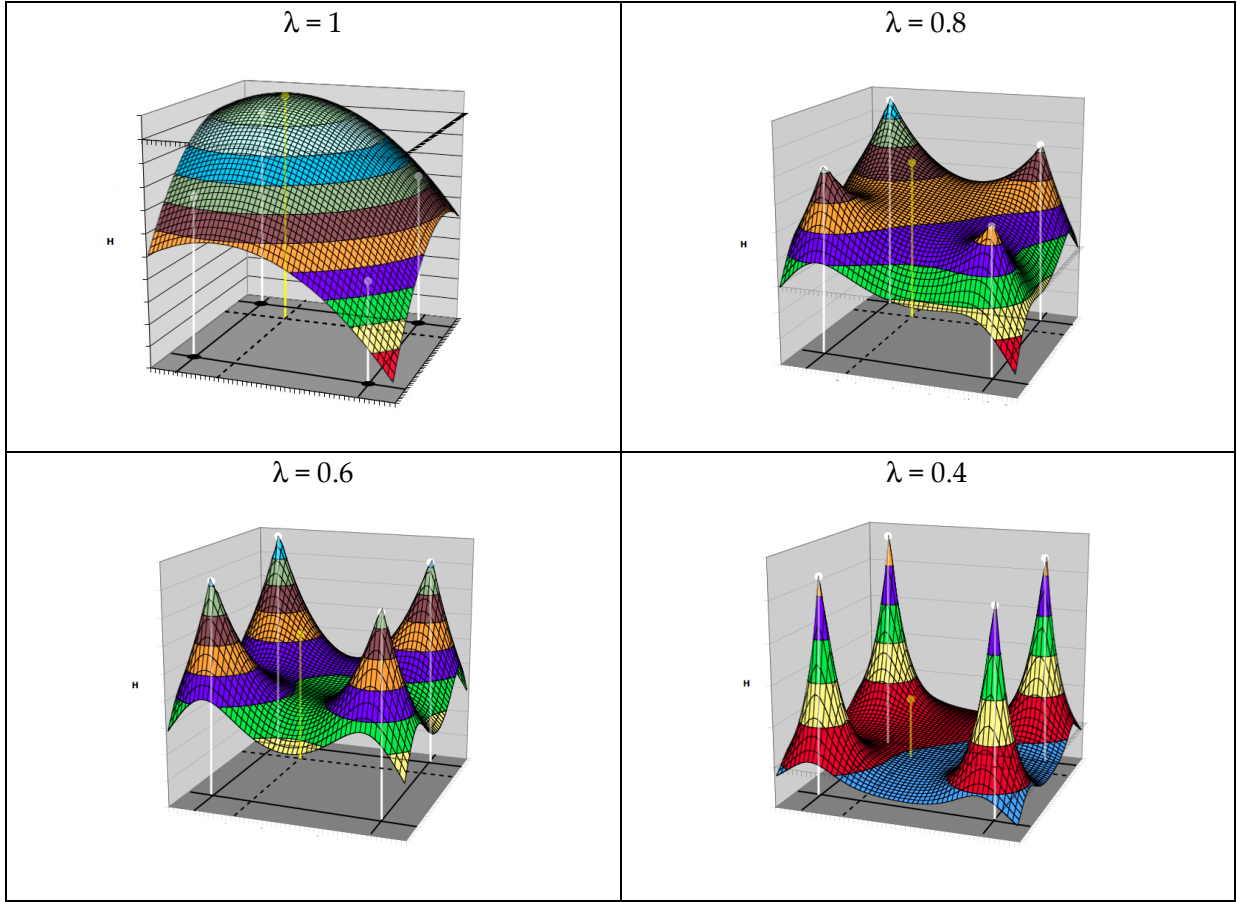   c.   *Combination:* by dynamically-weighted superposition, the λ-method (22)

Figure 4. The effective Harmony surface as $\lambda \to 0$ during computation (schematic). The correct output is the grid point corresponding to the highest peak. The solid lines on the floor intersect at the grid states; the dashed lines, at the blend that optimizes Harmony.

In many connectionist models (including PDP models), when a single response is required, there is (explicitly or implicitly) a layer of localist units, one per response, with each unit inhibiting all the others, generating a winner-take-all dynamics in which one unit typically ends up with all the activation: this is the response selection dynamics of these models, the counterpart to our quantization. To apply such an approach to the general problem under consideration here, where selection is not among a fixed set of atomic responses, but rather among an open-ended set of combinatorial structures, a single unit would need to be dedicated to each possible combinatorial output (as in what Pinker & Prince (1988) dub the 'whole-string binding network' of Rumelhart & McClelland (1986a)). The approach we are proposing avoids this, using combinatorially-structured distributed representations as the attractors of the selection dynamics.

The general issue of quantization has received considerable attention in architectures using compressed tensor product representations (Section 2.5). To eliminate the noise introduced by compression, researchers have utilized 'clean-up' processes that use the noisy

retrieved vectors to select the best-matching source representation. More recently, Levy & Gayler (2009) and Gayler & Levy (2009) have focused on the specific issue of quantization more directly. As in our framework, Levy and Gayler utilize two interleaved dynamical processes: parallel evaluation of possible distributed output representations in a hill-climbing procedure, and a distributed version of winner-take-all. In Levy and Gayler's theory, the relative contribution of these two processes is constant; in our λ-method, the relative weighting of quantization increases as computation proceeds. A second important difference is that we utilize stochastic optimization—a necessary feature for finding global Harmony maxima (Section 3.3) and a critical component of our explanation of empirical phenomena in language processing (Section 4.3).

Outside of compressed tensor product representations, response selection has also been addressed in many connectionist models. These have typically focused on cognitive domains that lack mutually-dependent choices, however. For example, in the domain of word reading, Plaut, McClelland, Seidenberg, & Patterson (1996) argue that successful generalization to novel words requires developing 'componential attractors' over the sublexical correspondences between orthographic and phonological representations (e.g., mapping the letter D to the sound [d]). Critically, for these componential attractors the choices are mutually independent. The choice of which pronunciation to generate for one part of the string is independent of the decision to generate a pronunciation for another part of the string (i.e., Plaut et al.'s networks acquire attractors with "orthogonal sub-basins" (p. 88) for each part of the string). For example, in generating output [fæd] for input FAD, the decision to pronounce F as [f] is independent of the decision to pronounced D as [d]. When correct processing cannot be accomplished by mutually independent decisions (e.g., for the word YACHT), Plaut et al.'s networks acquire far less componential attractors. Critically, in the linguistic domains we have discussed above, we require both combinatorial output representations and mutually-dependent choices; this forces us to posit distinct computational mechanisms.

4. Empirical tests

Having motivated and laid out our framework, Gradient Symbol Processing, and a specific instantiation, λ-Diffusion Theory, we now ask whether the theory can address empirical issues in linguistic competence and performance, via specific models constructed within the theory. With respect to competence, we investigate whether the theory does indeed allow us to solve the problem of mutually-dependent choices (Section 4.1) and whether both discrete and continuous aspects of grammatical knowledge can be modeled (Section 4.2). Then (Section 4.3) we summarize modeling results concerning phonological production performance which are reported in detail elsewhere. Our ultimate goal is to develop analytic results proving that the theory (or one of its models) has certain key properties, but at this point we can only report model-simulation results concerning these properties. Simulation files can be downloaded from the online supplemental materials at http://faculty.wcas.northwestern.edu/matt-goldrick/gsp.

4.1. *Is the Problem of Mutually-Dependent Choices solved?*

To test whether λ-Diffusion Theory can allow us to handle the critical problem identified in Section 3.5, we modeled the nanogrammar of (21) using the implementation of the corresponding Harmonic Grammar described in Section 3.4. In this *Two-Trees Model*, distributed, orthogonal role vectors were used to implement the three positions of the simple trees (root, left child, right child) and distributed, orthogonal filler vectors were used to implement the possible fillers for each of these positions (S, S2, Al, Is). One set of filler and role vectors were pseudo-randomly generated for this model. Grid states consisted of all possible role/filler bindings (e.g., not just [s Al Is] and [s2 Is Al] but also [s Is Is], [Al Al S], etc.) There were 12 input and 12 output units.

   As noted in Section 3.5, for the Problem of Mutually-Dependent Choices, we do not consider an input: both grammatical outcomes are equally well formed; the input to the network was therefore set to 0. Temperature and λ were initially set to relatively high values and slowly decayed. We assumed that the network had settled on a solution when the rate of change for output unit activations fell below a certain threshold.

   The results of 100 runs of a simulation of the Two-Trees Model suggest that λ-Diffusion Theory solves, with a high degree of accuracy, the particular Problem of Mutually-Dependent Choices posed in Section 3.5 (two runs are shown in Figure 5). In every run, the network converged to one of the equally well-formed grammatical trees (54% [s Al Is] and 46% [s2 Is Al]). By superimposing optimization and selection, our framework enables grammatical computation over combinatorial representations in a continuous space.
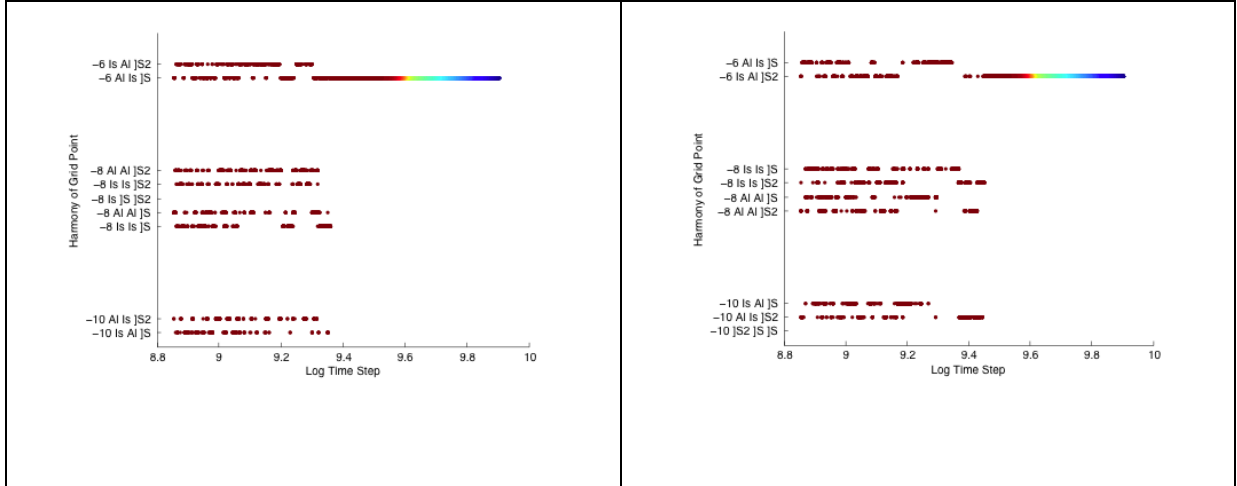

Figure 5. Two runs of a simulation of the Two-Tree Model generating two different trees grammatical in the language (21b). At each time step (horizontal axis), the graph shows (on the vertical axis) the grid state (pure tree) nearest to the current state (i.e., the currently visited $\mathcal{D}_Q$-attractor basin). Red (early) indicates larger and blue (late) smaller distance to the grid. Grid points are arranged vertically by their Harmony; points with the same Harmony are separated arbitrarily for visibility.

4.2. Can discrete and continuous aspects of phonological competence be successfully modeled?

In this section, we instantiate λ-Diffusion Theory with the *Neutralization Model*, which embodies the simple two-constraint phonological grammar discussed in (17). As discussed in Section 3.2, two different discrete outcomes arise from different weightings of the conflicting constraints. In German, MARK$_{voi}$ dominates FAITH$_{voi}$; final lexical /d/ is therefore pronounced [t]. In the English grammar, however, the reverse is true, and final lexical /d/ is pronounced faithfully, as [d]. Our first goal is to confirm that this basic discrete contrast between two languages can be captured by the Neutralization Model.

Our second goal is to examine the ability of the theory to model continuous aspects of phonological competence. Instrumental studies in a number of languages have documented that in many cases 'neutralized' forms—e.g., where final lexical /d/ is pronounced, grammatically, as [t]—have small but significant phonetic differences from their non-neutralized counterparts.[14] For example, in German, when final lexical /d/ is pronounced grammatically as [t], the preceding vowel is significantly longer compared to the vowel preceding a lexical /t/ that is pronounced as [t] (Port & O'Dell, 1985). However, in other cases, neutralization appears to be relatively complete; for example, Kim & Jongman (1996) find no significant phonetic distinctions when manner distinctions are neutralized in Korean.

In the Gradient Symbol Processing framework, these continuous phenomena are explained by the same factors that account for discrete patterns—namely, the interaction of conflicting constraints. Within the high-dimensional continuous space of phonological representations, FAITHFULNESS constraints implemented in the optimization dynamics will prefer points that lie closer to the target representation. Given that speakers do not have infinite time to compute the target phonological representation, λ will not have time to decay completely to 0. Since quantization $\mathcal{D}_Q$ will therefore never completely dominate optimization $\mathcal{D}_\mathcal{G}$, the influence of these FAITHFULNESS constraints can cause the output of the network to deviate from grid points that violate FAITHFULNESS. For example, if the lexical representation is /d/, but (as in German) relatively stronger MARKEDNESS causes the network to converge to the region of the grid point for /t/, FAITHFULNESS constraints—acting over the continuous space of phonological representations—will pull the network's output in the direction of the grid point corresponding to /d/.

Now a primary feature of similarity encoded through distributed representations is that similar inputs are mapped to similar outputs (Hinton, McClelland, & Rumelhart, 1986:81 ff.); we therefore assume that, through the phonetic interpretation process (not modeled), such a deviation in the phonological representation will manifest itself phonetically as a deviation towards the phonetic properties of the faithful output (including, in German, longer length of a preceding vowel).

---

[14] Syllable-final devoicing, as in German, entails that a contrast that can occur in the onset of pronounced forms, e.g., that between [d] and [t], is 'neutralized' in coda: there is no such contrast syllable-finally, where only [t] is grammatical. A lexical coda /d/ is 'neutralized' to [t], while a lexical coda /t/ is pronounced as a 'non-neutralized' [t].

Furthermore, the *quantitative* strength of MARKEDNESS relative to FAITHFULNESS will determine the *degree* of deviation. When MARKEDNESS is very strong (as in Korean), FAITHFULNESS will have less of an effect, resulting in smaller deviations from the grid point. Quantitative variation in relative constraint strength thus potentially accounts for the cross-linguistic contrast between languages exhibiting significant incomplete neutralization and those with relatively complete neutralization.

To examine these discrete and continuous phenomena, we modeled grammar fragments that focused on the processing of consonants, ignoring vowels (following (17), using the grammar-encoding methods of Section 3.4). In the Neutralization Model, a phonological representation is a sequence (simultaneously represented) of two syllables, each consisting of an onset and a coda, each position containing a single consonant that was specified for place of articulation and voicing (8 output and 8 input units). Consonants could either have coronal (e.g., /t, d/) or dorsal (/k, g/) place, and be voiced (/d, g/) or voiceless (/t, k/). FAITHFULNESS constraints FAITH$_{voi}$ and FAITH$_{place}$ penalized output representations that did not have, in each syllable position, the same feature values as the input. MARK$_{voi}$ penalized the [+voiced] feature in coda position (see Section 3.2). The weighting of FAITHFULNESS was held constant at 1.0 and the strength of MARKEDNESS was varied among 0.05 (less than 1.0, corresponding to a language with no neutralization, e.g., English), 1.25 (slightly greater than 1.0, corresponding to a language with incomplete neutralization, e.g., German) and 12.25 (much greater than 1.0, corresponding to a language with relatively complete neutralization, e.g., Korean). In all simulations, the threshold for network settling was such that λ did not decay to 0 (at settling time, $\lambda \approx .01$).

We simulated the production of two two-syllable phonological representations; one had a voiced velar coda in the first syllable and the other a voiceless velar coda (/tag.tak/ vs. /tak.tak/). (The second syllable plays no role in the discussion here.) We simulated 10 productions of each input. To index the degree of coda neutralization, we compared the output activation of the fillers in the first syllable coda[15] across inputs (i.e., the output for coda /g/ vs. coda /k/). For each input, the Euclidean distance between the filler activations was calculated for all pairings of the 10 phonological output representations. When MARKEDNESS was weaker than FAITHFULNESS (0.05 vs. 1.0 weighting), the voiced coda /g/ was fully pronounced; lexical /g/ mapped to output [g] (and, as always, lexical /k/ mapped to output [k]). This yielded a strong contrast between the outputs for the two lexical inputs in the continuous representational space (mean Euclidean distance: 1.34; standard error: 0.0006). When MARKEDNESS was stronger than FAITHFULNESS (1.25 or 12.25 vs. 1.0 weighting), neutralization occurred; for lexical /g/ (as well as lexical /k/) the closest grid point in the output was [k]. However, the degree of neutralization varied with the strength of MARKEDNESS. When MARKEDNESS was relatively weak (1.25 weighting), the Euclidean distance between outputs was significantly larger (mean: 0.032; s.e.: 0.001) than the case

---

[15] The 'activation' of, say the [k] filler here is the dot product of (i) the distributed representation for [k] in the coda of the first syllable, and (ii) the representation of the corresponding constituent of the output.

where MARKEDNESS was relatively strong (a 12.25 weighting yielded a mean distance of 0.008; s.e.: 0.0004).

This example illustrates how the Gradient Symbol Processing framework can provide a unified account of both variation in discrete outcomes (whether a grammar allows or neutralizes a contrast between voiced stops in coda) as well as continuous variation (the degree to which voicing neutralization is complete). Of course, the grammar fragment we have utilized here is extremely simple (but still non-trivial); phonological grammars typically involve many constraints operating over highly complex multidimensional symbolic representations. Since the mechanisms proposed here are fully general, we aim to explore the computational properties of more complex grammars in future work.

4.3. Can discrete and continuous performance phenomena be explained?

In the Gradient Symbol Processing framework, the competence and performance of the cognitive system are deeply connected. Both are accounted for by the same set of principles—those that define the knowledge of the cognitive system (i.e., the Harmony function specifying grammar $\mathcal{G}$: $H_{\mathcal{G}}$) and its computation (Subsymbolic Optimization-Quantization). Critically, these principles also allow a unified account of discrete and continuous patterns in experimental data. In this section, we focus on one specific aspect of grammatical knowledge, FAITHFULNESS constraints. In conjunction with our computational principles, FAITHFULNESS constraints allow us to formalize similarity-based psychological explanations (Section 2.2) of both discrete and continuous performance phenomena.

Similarity has played a critical role in accounting for a number of discrete empirical patterns relating to speech errors. Similar sounds are more likely to interact in spontaneous speech errors than dissimilar sounds (see Vousden, Brown, & Harley, 2000, for a review); in tongue twister tasks, higher error rates are observed for sequences with similar segments (see Wilshire, 1999, for a review). For example, a substantial number of errors involving /r/ are observed when it is in the context of the highly similar segment /l/ (e.g., "<u>r</u>eef <u>l</u>eap" → "<u>l</u>eaf <u>r</u>eap"); fewer /r/ errors are observed in the context of the less similar segment /b/ (e.g., "<u>r</u>eek <u>b</u>ead" → "<u>b</u>eak <u>r</u>eed"). *Structural* similarity also influences errors; sounds are more likely to interact when they occur in similar syllable positions (Vousden et al., 2000; Wilshire, 1999). For example, more /r/-/l/ errors are observed in sequences like "<u>r</u>eef <u>l</u>eap" (where both segments are in onset) than in sequences like "<u>r</u>eef pee<u>l</u>" (where one segment is in onset and another in coda).

In Gradient Symbol Processing, the macrostructural property of sensitivity to representational similarity emerges from the microstructure of computation. The preceding sections (4.1, 4.2) illustrate models under the most favorable processing conditions—a slow decay of λ and T. In those cases, λ-Diffusion allowed the model to settle on the most Harmonic grid point. But when the model is forced to produce outputs quickly (as participants must do in a tongue-twister task), we expect errors to result. As summarized in (25) below, we hypothesize that the distribution of these errors will reflect the stochastic structure of Harmony optimization (19).

(25) Error Hypothesis: The probability of a correct or incorrect response x, $p(x)$, is an exponentially increasing function of $H_{\mathcal{G}}(x)$:

$$p(x) \propto \exp(H_{\mathcal{G}}(x)/T), \text{ for some } T$$

Equivalently:　　$\log p(x) \propto H_{\mathcal{G}}(x) - k, \text{ for some } k$

Similarity-based explanations of speech error patterns are a specific instantiation of this general hypothesis. FAITHFULNESS constraints form a critical part of the structure of a grammar $\mathcal{G}$. These constraints are violated by phonological representations that fail to preserve the structure of the input along some particular dimension. Their presence within the grammar entails that (all else being equal) output structures that better match the structure of the input will have higher Harmony than those that do not. The probability of an error will therefore be a function of its similarity to the target (defined precisely by the structure of FAITHFULNESS constraints).

To test the Error Hypothesis, we instantiated λ-Diffusion Theory in the *Tongue-Twister Model* of a tongue-twister task. Like the model described in the previous section, this model produced sequences of two CVC syllables (e.g., "sag can"). Syllable number (first/second) and syllable position (onset/coda) were combined into recursive distributed role vectors (e.g., $\mathbf{r}_{\text{Onset}/\sigma_1} = \mathbf{r}_{\text{Onset}} \otimes \mathbf{r}_{\sigma_1}$; Smolensky, 2006a:182 ff.; pseudo-random vectors in $\mathbb{R}^2$ were constrained to satisfy $\text{sim}(\mathbf{r}_{\sigma_1}, \mathbf{r}_{\sigma_2}) = 0.25$, $\text{sim}(\mathbf{r}_{\text{Onset}}, \mathbf{r}_{\text{Coda}}) = 0.1$). Distributed filler vectors represented four consonants. These consisted of a pair of highly similar consonants (e.g., /k/ and /g/; dot product of filler vectors: 0.5) and a pair of less similar consonants (e.g., /s/ and /n/; dot product of vectors: 0.25); across pairs, similarity was low (dot product: 0.1). A set of filler vectors in $\mathbb{R}^4$ meeting these conditions were generated pseudo-randomly, once for this model (there were 16 input and 16 output units). FAITHFULNESS constraints (e.g., 'onset of input syllable 1 = onset of output syllable 1') penalized output representations that were not identical to the input. No MARKEDNESS constraints were present in the modeled grammar.

Production of two different tongue twisters was modeled. The first target syllable in each sequence was the same (e.g., "sag"). The second target syllable was constructed such that similar consonants occurred in the same syllable positions (e.g., "sag knack") or opposite positions (e.g., "sag can"). When λ was allowed to slowly decay from a high starting value (1.0), the system produced both target sequences correctly in each of 100 runs. To simulate the increased speed of the tongue twister task, the initial value of λ was decreased (to 0.015). This causes the network's response time to substantially decrease; at this faster rate, it produced many errors. As shown in Figure 6, the results were consistent with the qualitative patterns observed in experimental speech-error data. Errors on the first syllable (identical across sequences) are more likely to involve more similar segments, and are more likely to involve segments in the same syllable position.

The Error Hypothesis (25) goes beyond qualitative patterns to make *quantitative* predictions about the relative probability of errors. The results in Figure 7 suggest that these predictions are fairly accurate; the Harmony of an output form is a good predictor of its output probability. This suggests that in λ-Diffusion Theory, the properties of performance errors are closely connected to the computational principle of stochastic Harmony

optimization—the key to achieving *competence* within Gradient Symbol Processing. In future work, we plan to explore the degree to which these quantitative predictions account for the empirical distributions of speech errors arising in phonological encoding.
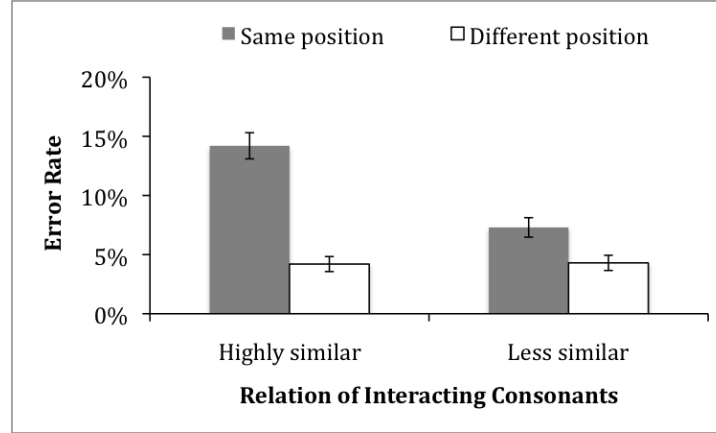

Figure 6. First-syllable error rates in 1,000 runs of a simulation of the Tongue-Twister Model productions of two tongue-twister sequences. Error bars indicate standard error.
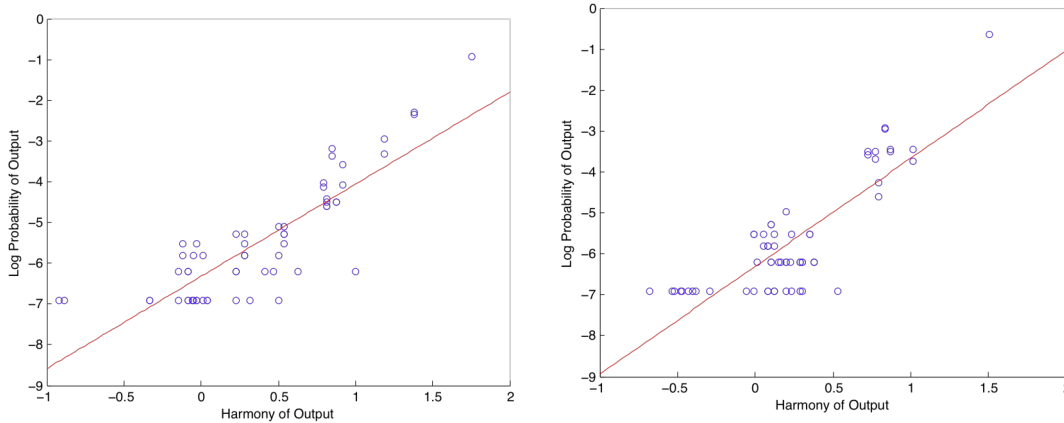

Figure 7. Harmony of grid point (horizontal axis) vs. log probability that grid point was selected as the network output (vertical axis) in 1,000 simulated productions of two tongue-twister sequences (left panel: "sag knack"; right panel: "sag can"). Solid line indicates linear regression fit; compare (25).

In addition to accounting for discrete phenomena such as likelihood of error outcomes, the concept of similarity has played a role in understanding the continuous properties of speech errors. Recent work has shown that the phonetic properties of speech errors reflect properties of the intended target. For example, in an error like 'big' → "pig", the [p] tends to have a shorter voice onset time (VOT) compared to correctly produced instances of 'pig' (Goldrick & Blumstein, 2006). Speech error outcomes thus tend to be slightly similar to the intended target *within continuous phonetic space*.

Parallel to the account of incomplete neutralization in the previous section, our framework allows us to use the same principles that govern discrete error outcomes to account for these continuous error phenomena. For example, if the target grid point is [b], but too-rapid processing causes the network to converge to the region of the grid point for [p], FAITHFULNESS constraints will pull the network's output towards the grid point corresponding to the target [b]—producing a phonetic deviation towards the properties of the target (including a shorter VOT).

To test this hypothesis, we focused on the most frequent errors in the simulation above (involving similar consonants in the same syllable position; e.g., "sag knack" → "sa<u>ck</u> knack"). Following experimental studies of speech errors, we compared these [k] error outcomes to correctly produced [k]s in the same sequence (e.g., correctly produced coda /k/ in "knack"). As in the simulations reported in 4.2, the threshold for network settling was such that λ did not decay to 0 (at settling time, $\lambda \approx .01$). As shown in Figure 8, the [k] filler is significantly less active in errors, reflecting the influence of FAITHFULNESS constraints on the continuous aspects of phonological encoding.
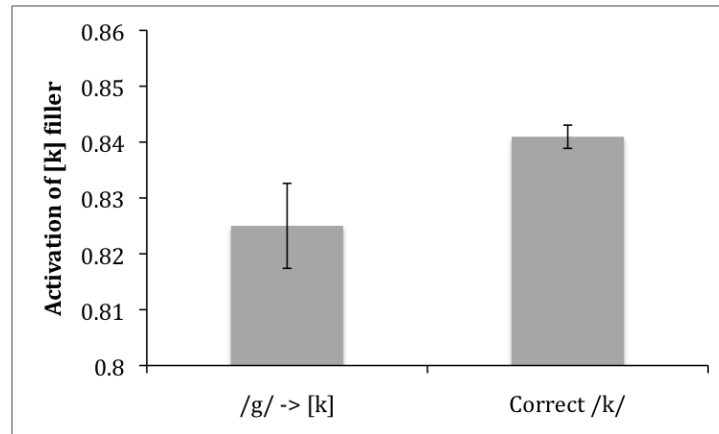

Figure 8. Mean activation of the [k] filler in errors and correct productions. Error bars indicate standard error.

These examples show how λ-Diffusion Theory provides a single, uniform framework that: one, yields formal similarity-based explanations of both discrete and continuous empirical patterns in speech production; and two, makes quantitative predictions about these patterns. Note that although this discussion has focused on the relationships between similarity and errors induced by FAITHFULNESS, our error hypothesis (25) also makes quantitative predictions about the relationship between error probability and other aspects of the grammar (i.e., MARKEDNESS; see Goldrick & Daland, 2009, for a recent review of relevant speech error data). We plan to examine these predictions more closely in future work.

5. Summary and conclusion

The Gradient Symbol Processing framework developed here aims to account for the emergence (i.e., the formal entailment) of the macrostructural descriptions of grammatical theory from the microstructural algorithms that underlie language processing. Pursuing this PDP research program has, we believe, led to new insights into a central issue in cognition: the relationship between the continuous and the discrete aspects of mental representation and processing.

An extensive body of research across many cognitive domains, including language, motivates the use of combinatorial mental representations. At the same time, a PDP research program requires respecting the microstructural constraints imposed on cognitive theories: combinatorial representations must be processed by continuous, distributed algorithms. Tensor product representations allow us to define Distributed Symbol Systems that respect both macro- and microstructural principles. However, such superpositional representations lead to a clear processing issue—the need to resolve ambiguous blend states.

To address this, we have proposed that in the Gradient Symbol Processing framework, cognitive processing of Distributed Symbol Systems consists of Subsymbolic Optimization-Quantization. This is the superposition of two dynamical processes, optimization and quantization, operating upon the same representation, i.e., within the same connectionist units. A grammar $\mathcal{G}$ provides a formal specification, $H_{\mathcal{G}}$, of the linguistic knowledge with respect to which representations are optimized during processing. $H_{\mathcal{G}}$ is a numerical function sensitive to combinatorial symbolic structure yet operating over a continuous representational space. To resolve ambiguous blend states, superimposed upon optimization of grammatical knowledge is a distributed quantization process that favors pure discrete symbol structures. A particular instantiation of the framework, λ-Diffusion Theory, proposes specific types of dynamical systems to realize optimization and quantization, and a particular mode of superimposing them. This theory can in turn be implemented in models, simulations of which offer successful accounts of specific empirical phenomena.

By specifying how discrete structural knowledge emerges from a continuous representational and processing substrate, Gradient Symbol Processing supports a theoretical unification of discrete and continuous empirical phenomena. The same grammatical principles that specify discrete phonological competence also account not only for discrete patterns in speech errors but also continuous phonetic variation in both errorful and non-errorful speech.

Acknowledgements

at Johns Hopkins, particularly Colin Wilson, and audiences at Cambridge University, Ecole Normale Supérieure, the 2009 GLOW Conference, the 5th International Workshop on Language Production, Oxford University, Max Planck Institute for Psycholinguistics, Northwestern University, Royal Netherlands Academy of Arts and Sciences, University of Arizona, University of Chicago, Workshop on Dynamical Systems in Language, and Yale University.

References

Anderson, J. R., & Lebiere, C. J. (1998). *The Atomic Components of Thought*. Mahwah, NJ: Erlbaum.

Baird, B., & Eeckman, F. (1993). A normal form projection algorithm for associative memory. In M. H. Hassoun (Ed.), *Associative neural memories* (pp. 135-166). New York, NY: Oxford University Press.

Barlow, H. B. (1972). Single units and sensations: A neuron doctrine for perceptual psychology? *Perception, 1*, 371-392.

Bird, H. (1998). Slips of the ear as evidence for the postperceptual priority of grammaticality. *Linguistics, 36,* 469–516.

Boersma, P. (1998). *Functional Phonology: Formalizing the interactions between articulatory and perceptual drives*. The Hague: Holland Academic Graphics.

Bowers, J. S. (2002). Challenging the widespread assumption that connectionism and distributed representations go hand-in-hand. *Cognitive Psychology, 45*, 413-445.

Bowers, J. S. (2009). On the biological plausibility of grandmother cells: Implications for neural network theories in psychology and neuroscience. *Psychological Review, 116 (1)*, 220-251.

Churchland, P. S. & Sejnowski, T. J. (1992). *The computational brain*. Cambridge, MA: MIT Press.

Cohen, M. A., & Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, 13*, 815–825.

Dell, G. S. (1986). A spreading activation theory of retrieval in sentence production. *Psychological Review, 93*, 283-321.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science, 14*, 179 –211.

Feldman, J. (1989). Neural representation of conceptual knowledge. In L. Nadel, L. A. Cooper, P. Culicover, & R. M. Harnish (Eds.), *Neural Connections, Mental Computation* (pp. 68–103). Cambridge, MA: MIT Press.

Feldman, J. A., & Ballard, D. H. (1982). Connectionist models and their properties. *Cognitive Science, 6*, 205–254.

Fischer-Baum, S., & Smolensky, P. (forthcoming). An axiomatic approach to cognitive science: The case of transposition priming.

Flemming, E. (2001). Scalar and categorical phenomena in a unified model of phonetics and phonology. *Phonology, 18*, 7-44.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition, 28*, 3–71.

Forster, K. I. & Davis, C. (1984). Repetition priming and frequency attenuation in lexical access. *Journal of Experimental Psychology: Learning, Memory and Cognition, 10*, 680-698.

Garrett, M. F. (1975). The analysis of sentence production. In G. H Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory* (Vol. 9, pp. 133-177). New York: Academic Press.

Gayler, R. W. (2003). Vector Symbolic Architectures answer Jackendoff's challenges for cognitive neuroscience. In Peter Slezak (Ed.), *ICCS/ASCS International Conference on Cognitive Science* (pp. 133-138). Sydney, Australia: University of New South Wales.

Gayler, R. W., & Levy, S.D. (2009). A distributed basis for analogical mapping. In B. Kokinov, K. Holyoak, & D. Gentner (Eds.), *New frontiers in analogy research; Proceedings of the Second International Analogy Conference—Analogy 09* (pp. 165-174. Sofia, Bulgaria: New Bulgarian University Press).

Golden, R. M. (1986). The "Brain-State-in-a-Box" neural model is a gradient descent algorithm. *Mathematical Psychology, 30–31*, 73–80.

Golden, R. M. (1988). A unified framework for connectionist systems. *Biological Cybernetics, 59*, 109–120.

Goldrick, M. (2008). Does like attract like? Exploring the relationship between errors and representational structure in connectionist networks. *Cognitive Neuropsychology, 25*, 287-313.

Goldrick, M., & Blumstein, S. E. (2006). Cascading activation from phonological planning to articulatory processes: Evidence from tongue twisters. *Language and Cognitive Processes, 21*, 649-683.

Goldrick, M., & Daland, R. (2009). Linking speech errors and phonological grammars: Insights from Harmonic Grammar networks. *Phonology, 26*, 147-185.

Goldrick, M., & Rapp, B. (2007). Lexical and post-lexical phonological representations in spoken production. *Cognition, 102*, 219-260.

Gomez, P., Ratcliff, R., & Perea, M. (2008). The overlap model: A model of letter position coding. *Psychological Review, 115 (3)*, 577-600.

Hale, J., & Smolensky, P. (2006). Harmonic grammars and harmonic parsers for formal languages. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 393–415). Cambridge, MA: MIT Press.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental and cognitive psychology. *Behavioral and Brain Sciences, 21*, 803-865.

Hannagan, T., Dupoux, E., & Christophe, A. (in press). Holographic string encoding. *Cognitive Science*.

Hayes, B., Kirchner, R., & Steriade, D. (Eds.). (2004). *Phonetically-Based Phonology*. Cambridge University Press.

Hinton, G. E., & Anderson, J. A. (Eds.). (1981). *Parallel models of associative memory.* Mahwah, New Jersey: Erlbaum.

Hinton, G. E., & Sejnowski, T. J. (1983). Optimal perceptual inference. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Hinton, G. E., & Sejnowski, T. J. (1986). Learning and relearning in Boltzmann Machines. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 282–317). Cambridge, MA: MIT Press.

Hinton, G. E., McClelland, J. L., & Rumelhart, D. E. (1986). Distributed representation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 77–109). Cambridge, MA: MIT Press.

Hofstadter, D. R. (1979). *Gödel, Escher, Bach: An eternal golden braid.* Basic Books.

Hofstadter, D. R. (1985). Waking up from the Boolean dream, or, subcognition as computation. In D. R. Hofstadter, *Metamagical themas: Questing for the essence of mind and pattern* (pp. 631–665). Basic Books.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA, 79*, 2554–2558.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences USA, 81*, 3088–3092.

Hummel, J. E., & Holyoak, K. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological Review, 110(2)*, 220-264.

Jakobson, R. (1962). *Selected Writings I: Phonological Studies*. The Hague: Mouton.

Jordan, M. I. (1986a). An introduction to linear algebra in parallel distributed processing. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 365–422). Cambridge, MA: MIT Press.

Jordan, M. I. (1986b). Serial order: A parallel distributed processing approach. Institute for Cognitive Science Report 8604. University of California, San Diego. Reprinted (1997) in J. W. Donahoe and V. P. Dorsel (Eds.), *Neural-network models of cognition: Biobehavioral foundations* (pp. 221-277). Amsterdam: Elsevier Science Press.

Kanerva, P. (2009). Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation, 1*, 139-159.

Kim, H., & Jongman, A. (1996). Acoustic and perceptual evidence for complete neutralization of manner of articulation in Korean. *Journal of Phonetics, 24*, 295–312.

Kruschke, J. K. (2008). Models of categorization. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 267-301). New York, Cambridge University Press.

Legendre, G., Miyata Y. & Smolensky, P. (1990a). Harmonic Grammar—A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society* (pp. 388–395). Hillsdale, NJ: Lawrence Erlbaum.

Legendre, G., Miyata, Y., & Smolensky, P. (2006). The interaction of syntax and semantics: A Harmonic Grammar account of split intransitivity. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 417–452). Cambridge, MA: MIT Press.

Legendre, G., Sorace, A., & Smolensky, P. (2006). The Optimality Theory-Harmonic Grammar connection. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 2: Linguistic and philosophical implications* (pp. 339–402). Cambridge, MA: MIT Press.

Levelt, W. J. M., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *Behavioral and Brain Sciences, 22*, 1-75.

Levy, S. D., & Gayler, R. W. (2008). Vector Symbolic Architectures: A new building material for Artificial General Intelligence. *Proceedings of the First Conference on Artificial General Intelligence (AGI-08)*. IOS Press.

Levy, S. D., & Gayler, R. W. (2009b). "Lateral inhibition" in a fully distributed connectionist architecture. In A. Howes, D. Peebles, & R. Cooper (Eds.), *Proceedings of the 9th International Conference on Cognitive Modeling — ICCM 2009* (pp. 318-323). Manchester, UK.

Luce, P. A., & Pisoni, D. B. (1998). Recognizing spoken words: The neighborhood activation model. *Ear & Hearing, 19*, 1-36.

Marcus, G. F. (2001). *The algebraic mind: Integrating connectionism and cognitive science*. Cambridge, MA: MIT Press.

McClelland, J. L. (1993). Toward a theory of information processing in graded, random, and interactive networks. In D. E. Meyer & S. Kornblum (Eds.), *Attention & Performance XIV: Synergies in experimental psychology, artificial intelligence and cognitive neuroscience* (pp. 655-688). Cambridge, MA: MIT Press.

McClelland, J. L. & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review, 88*, 375–407.

McClelland, J. L., & Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the microstructure of cognition. Vol. 2: Psychological and biological models* (pp. 272–325). Cambridge, MA: MIT Press.

McClelland, J. L., Rumelhart, D. E., & the PDP Research Group (1986). *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 2: Psychological and biological models.* Cambridge, MA: MIT Press.

McClelland, J.L., & Elman, J. L. (1986). The TRACE model of speech perception, *Cognitive Psychology, 18*, 1–86

McMurray, B. Tanenhaus, M. K., & Aslin R. N. (2009). Within-category VOT affects recovery from "lexical" garden-paths: Evidence against phoneme-level inhibition. *Journal of Memory and Language, 60*, 65-91.

Minsky, M. (1975). A framework for representing knowledge. In P. H. Winston (Ed.), *The psychology of computer vision* (pp. 211–277). McGraw-Hill.

Movellan, J. R. (1998). A learning theorem for networks at detailed stochastic equilibrium. *Neural Computation, 10*, 1157-1178.

Movellan, J. R., & McClelland, J. L. (1993). Learning continuous probability distributions with symmetric diffusion networks. *Cognitive Science, 17*, 463-496.

Murdock, B. B., Jr. (1982). A theory for storage and retrieval of item and associative information. *Psychological Review, 89*, 316–338.

Newell, A. (1980). Physical symbol systems. *Cognitive Science, 4*: 135–183.

Page, M. (2000). Connectionist modelling in psychology: A localist manifesto. *Behavioral and Brain Sciences, 23*, 443-512.

Partee, B. H., ter Meulen, A., & Wall, R. E. (1990). *Mathematical methods in linguistics.* Boston, MA, Kluwer Academic Publishers.

Pater, J. (2009). Weighted constraints in generative linguistics. *Cognitive Science, 33*, 999-1035.

Perea, M. & Lupker, S. J. (2003). Transposed-letter confusability effects in masked form priming. In S. Kinoshita & S. J. Lupker (Eds.), *Masked priming: State of the art* (pp. 97-120). Hove, UK: Psychology Press,.

Pierrehumbert, J. (2006) The next toolkit. *Journal of Phonetics, 34*, 516-530.

Pike, R. (1984). Comparison of convolution and matrix distributed memory systems for associative recall and recognition. *Psychological Review, 91*, 281–294.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition, 28*, 73–193.

Plate, T. A. (1991). Holographic Reduced Representations: Convolution algebra for compositional distributed representations. *Proceedings of the International Joint Conference on Artificial Intelligence.*

Plate, T. A. (2000). Analogy retrieval and processing with distributed vector representations. *Expert Systems: The International Journal of Knowledge Engineering and Neural Networks; Special Issue on Connectionist Symbol Processing, 17(1)*, 29–40.

Plate, T.A. (2003). *Holographic reduced representation: Distributed representation of cognitive structure.* Stanford: CSLI.

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., & Patterson, K. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review, 103(1)*, 56–115.

Port, R., & O'Dell, M. (1985). Neutralization of syllable-final voicing in German. *Journal of Phonetics, 13*, 455–471

Pouget, A., Dayan, P, & Zemel, R. (2000). Information processing with population codes. *Nature Reviews Neuroscience 1*, 125-132

Prince, A., & Pinker, S. (1988). Wickelphone ambiguity. *Cognition, 30*, 188–190.

Prince, A., & Smolensky, P. (1991). Notes on connectionism and Harmony Theory in linguistics (Technical report CU-CS-533-91). Boulder, CO: Computer Science Department, University of Colorado at Boulder.

Prince, A., & Smolensky, P. (1993/2004). *Optimality theory: Constraint interaction in generative grammar.* Technical report TR-2, Rutgers Center for Cognitive Science, Rutgers University, New Brunswick, NJ. Technical report CU-CS-696-93, Department of Computer Science, University of Colorado, Boulder. Revised version, 2002: ROA-537-0802, Rutgers Optimality Archive, http://roa.rutgers.edu. Published 2004, Oxford: Blackwell.

Pylyshyn, Z. W. (1984). *Computation and cognition: Toward a foundation for cognitive science.* Cambridge, MA, MIT Press.

Rumelhart, D. E. (1975). Notes on a schema for stories. In D. G. Bobrow and A. Collins, (Eds.), *Representation and understanding* (pp. 211–236). Academic Press.

Rumelhart, D. E., Hinton, G. E., & McClelland, J. L. (1986). A general framework for parallel distributed processing. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 110-146). Cambridge, MA: MIT Press.

Rumelhart, D. E., & McClelland, J. L. (1982). An interactive activation model of context effects in letter perception: Part 2. The contextual enhancement effect and some tests and extensions of the model. *Psychological Review, 89*, 60–94.

Rumelhart, D. E., & McClelland, J. L. (1986a). On learning the past tenses of English verbs. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the microstructure of cognition. Vol. 2: Psychological and biological models* (pp. 216–271). Cambridge, MA: MIT Press.

Rumelhart, D. E., & McClelland, J. L. (1986b). PDP models and general issues in cognitive science. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 110-146). Cambridge, MA: MIT Press.

Rumelhart, D. E. , McClelland, J. L., & the PDP Research Group (1986). *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations.* Cambridge, MA: MIT Press.

Rumelhart, D. E., & Norman, D. A. (1983/1988). Representation in memory. (1983). Technical Report No. 116, La Jolla, CA: UCSD Center for Human Information Processing. (1988). In R. C. Atkinson, R. J. Herrnstein, G. Lindzey, & R. D. Luce (Eds.), *Steven's handbook of experimental psychology.* New York, NY: Wiley.

Sanger, T. D. (2003). Neural population codes. *Current Opinion in Neurobiology, 13*, 238–249

Shattuck-Hufnagel, S., & Klatt, D. H. (1979). The limited use of distinctive features and markedness in speech production: Evidence from speech error data. *Journal of Verbal Learning and Verbal Behavior, 18*, 41-55.

Smolensky, P. & Legendre, G. (2006). *The harmonic mind: From neural computation to Optimality-Theoretic grammar (Vol. 1: Cognitive architecture; Vol. 2: Linguistic and philosophical implications).* Cambridge, MA: MIT Press.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence.*

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of Harmony Theory. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 194–281). Cambridge, MA: MIT Press.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist networks. *Artificial Intelligence, 46,* 159–216.

Smolensky, P. (2006a). Formalizing the principles I: Representation and processing in the mind/brain. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 147–205). Cambridge, MA: MIT Press.

Smolensky, P. (2006b). Optimization in neural networks: Harmony maximization. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 355–403). Cambridge, MA: MIT Press.

Smolensky, P. (2006c). Tensor product representations: Formal foundations. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 271–344). Cambridge, MA: MIT Press.

Smolensky, P., Legendre, G., & Tesar, B. B. (2006). Optimality Theory: The structure, use and acquisition of grammatical knowledge. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 453–544). Cambridge, MA: MIT Press.

Smolensky, P., & Tesar, B. B. (2006). Symbolic computation with activation patterns. In P. Smolensky & G. Legendre (Eds.), *The harmonic mind: From neural computation to Optimality-Theoretic grammar. Vol. 1: Cognitive architecture* (pp. 235–270). Cambridge, MA: MIT Press.

Stemberger, J. P. (1985). An interactive activation model of language production. In A. W. Ellis (Ed.) *Progress in the psychology of language* (Vol. 1, pp. 143-186). Hillsdale, NJ: Lawrence Erlbaum Associates.

Tabor, W. (2000). Fractal encoding of context-free grammars in connectionist networks. *Expert Systems: The International Journal of Knowledge Engineering and Neural Networks, 17(1)*, 41–56.

Trubetzkoy, N. (1939/1969). *Principles of phonology* (translation of *Grundzüge der Phonologie*). Berkeley: University of California Press.

van Gelder, T. (1991). What is the "D" in PDP? A survey of the concept of distribution. In W. M. Ramsey, S. P. Stich, & D. E. Rumelhart (Eds.), *Philosophy and connectionist theory* (pp. 33-59). Hillsdale, NJ: Lawrence Erlbaum Associates.

Vousden, J. I., Brown, G. D. A., & Harley, T. A.. (2000). Serial control of phonology in speech production: A hierarchical model. *Cognitive Psychology, 41*, 101-175.

Warner, N., Jongman, A., Sereno, J., & Kemps, R. (2004). Incomplete neutralization and other sub-phonemic durational differences in production and perception: Evidence from Dutch. *Journal of Phonetics, 32*, 251-276.

Wermter, S., & Sun, R. (Eds.). (2000). *Hybrid neural systems.* Heidelberg: Springer.

Wilshire, C. E. (1999). The "tongue twister" paradigm as a technique for studying phonological encoding. *Language and Speech, 42*, 57-82.