Pyeong Whan Cho*, Matthew Goldrick and Paul Smolensky

Incremental parsing in a continuous dynamical system: Sentence processing in Gradient Symbolic Computation

Abstract: Any incremental parser must solve two computational problems: (1) maintaining all interpretations consistent with the words that have been processed so far and (2) excluding all globally-incoherent interpretations. While these problems are well understood, it is not clear how the dynamic, continuous mechanisms that underlie human language processing solve them. We introduce a Gradient Symbolic Computation (GSC) parser, a continuous-state, continuous-time stochastic dynamical-system model of symbolic processing, which builds up a discrete symbolic structure gradually by dynamically strengthening a discreteness constraint. Online, interactive tutorials with open-source software are presented in a companion website. Our results reveal that the GSC parser can solve the two computational problems by moving to a non-discrete blend state that evolves exclusively to discrete states representing contextually-appropriate globally-coherent interpretations. In a simulation study using a simple formal grammar, we show that successful parsing requires appropriate control of the discreteness constraint strength (a *quantization policy*). With inappropriate quantization policies, the GSC parser makes mistakes that mirror those made in natural language comprehension (garden-path or local-coherence errors). These findings suggest that the GSC model offers a neurally plausible solution to these two core problems.

Keywords: incremental parsing, Gradient Symbolic Computation, dynamical systems models

1 Introduction

Language is a discrete combinatorial system (Hockett 1960). Symbols (e.g., phonemes, morphemes, phrases) combine with one another to create new, different structures with different meanings. While this combinatorial structure is beneficial for expressing a near-infinite variety of meanings, it presents challenges for perception of speech and text. As each symbol is linked to multiple combinations of symbols and, therefore, distinct meanings, this combinatorial structure leads to *local* ambiguity. Human language comprehension is incremental (Altmann and Kamide 1999; Bever 1970), with interpretations built over partial input before a whole symbol string is presented. Then, the question naturally follows: How does the human language processing system handle local ambiguity in incremental processing?

Any incremental processing system must accomplish two computational goals: (1) keeping all interpretations consistent with context, the symbols that have been processed, without choosing one over the others (*temporary ambiguity*), and (2) excluding all interpretations inconsistent with context (*context dependency*). For example, consider a sentence "Dogs yawn." After hearing "dogs", we need to consider every possible interpretation (e.g., "Dogs yawn", "Dogs sleep", "Dogs bark", "Dogs hate cats"). An early commitment to one (e.g., "Dogs bark") over the others can create difficulty when processing the second word. At the same time, when processing the second word "yawn," we need to reject interpretations inconsistent with context so that we do not choose "Cats yawn" instead of "Dogs yawn".

Sometimes, the human language processing system fails to achieve these two computational goals. We may choose one interpretation over the other when both are consistent with the linguistic input (the *garden-path* effect; Bever 1970; Frazier 1987). For example, there are two interpretations consistent with a sentence beginning "The horse raced past the barn…" In the first, *raced* is the main verb; in the second, *raced* is a passive participle in a reduced relative clause (e.g., "The horse (that) raced past the barn fell."). Although both are possible, listeners almost always assume the first interpretation, and have great difficulty revising this expectation. We refer to this issue as the problem of *temporary ambiguity*.

At other times, we may fail to use context, accepting globally incoherent interpretations. For example, when reading "The coach smiled at <u>the player tossed a frisbee</u> …" the underlined string forms a *locally coherent* subject-predicate sequence, even though such a structure is impossible given the preceding context (in a grammatical sentence, *tossed a frisbee* must be a reduced relative clause modifying *player*). In spite of its impossibility, readers show sensi-

***Corresponding author**: **Pyeong Whan Cho**, Department of Cognitive Science, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218, E-mail: pcho4@jhu.edu
Matthew Goldrick, Department of Linguistics, Northwestern University, 2016 Sheridan Rd., Evanston, IL 60208
Paul Smolensky, Department of Cognitive Science, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218

tivity to the presence of this locally coherent structure (Gibson 2006; Konieczny 2005; Tabor, Galantucci and Richardson 2004). We refer to this as the challenge of *context dependence*.

Rational models of language comprehension (e.g., Hale 2001; Jurafsky 1996; Levy 2008) solve the two problems by updating a conditional probability distribution across different interpretations as input is incrementally processed. Suboptimal behaviors are explained by assuming limited resources available for incremental processing (e.g., Jurafsky 1996; Levy, Reali and Griffiths 2008) or uncertainty about the previously processed input symbols (e.g., Levy, Bicknell, Slattery and Rayner 2009).

Although they explain *why* and *what* computational goals must be achieved (Marr 1982), these rational models typically do not explain *how* to achieve them at a mechanistic level (but see Hale 2011). For a more comprehensive understanding of human language processing, we need a mechanistic model working on specific representational and processing assumptions. We pursue a novel incremental processing model that has a clear symbolic interpretation, is reasonably analyzable, and provides neurally plausible solutions to these computational problems (Smolensky and Legendre 2006). None of the previously proposed models (e.g., Elman 1990; Hale 2011; Lewis, Vasishth and Van Dyke 2006; Tabor and Hutchins 2004; Tabor, Juliano and Tanenhaus 1997; Vosse and Kempen 2000) satisfy all of these desiderata (but see Carmantini, beim Graben, Desroches and Rodrigues 2017 for one such proposal, focusing on the problem of temporary ambiguity).

The core of our account is the *stable blend state hypothesis*:

– *Conjunctive blends*: After every symbol input, the human language processing system builds a conjunctive blend of multiple possible interpretations which can evolve to globally consistent interpretations and does not evolve to globally incoherent interpretations.
– *Stability of blend states*: For the system to hold multiple interpretations until a biasing input comes in later, the blend state must be stable.

In the next section, we propose our incremental processing model which has the ability to form a stable blend of different discrete structures. Then, we report a simulation study, showing that the model can solve the core computational problems in incremental processing. We discuss when and why the model may fail to parse a sentence. In the General Discussion, we evaluate our model and discuss future directions. Throughout, we point out the parallel presentation of the ideas, and their computer implementation, available in a companion website.[1]

2 A Gradient Symbolic Computation parser

Smolensky, Goldrick and Mathis (2014) proposed the Gradient Symbolic Computation (GSC) framework, which is a continuous-time, continuous-state stochastic dynamical system that gradually builds a discrete symbolic structure in a continuous representation space. We apply the GSC model to incremental processing, focusing on the crucial transient dynamics. In this section, we briefly introduce the GSC model in the incremental processing context using a simple formal language. Specific simulation results will be reported in the next section.

2.1 Grammar

To make the core computational problems clear, we consider a formal language that poses – in the purest, simplest possible form – the two core incremental processing computational problems (temporary ambiguity and context dependency) found in languages of any complexity. This language L consists of 4 sentences:

– S1=‘A B’
– S2=‘A C’
– S3=‘D B’
– S4=‘D C’

After processing a first word ‘A’, an optimal incremental processing system must be able to keep both S1 and S2 as candidate interpretations and exclude S3 and S4 as inconsistent with context, in this case, the first word.
We characterize L by a phrase structure grammar G consisting of the following rules:

1 Software and online Supplementary Materials are available in a companion website: https://cloud.sagemath.com/projects/18c77389-a5c3-49de-946a-7593b53d3fb2/

- S → S[1] | S[2] | S[3] | S[4]
- S[1] → A B
- S[2] → A C
- S[3] → D B
- S[4] → D C

The target parse trees of the four sentences are

- T1=[$_{\text{S}}$ [$_{\text{S[1]}}$ [A B]]]
- T2=[$_{\text{S}}$ [$_{\text{S[2]}}$ [A C]]]
- T3=[$_{\text{S}}$ [$_{\text{S[3]}}$ [D B]]]
- T4=[$_{\text{S}}$ [$_{\text{S[4]}}$ [D C]]]

See Figure 2(b) for an illustration of the parse tree for T1 (and see Supplementary Materials 1 for the use of bracketed symbols such as S[3]).

2.2 Representation

To introduce our representational framework, consider T1 = [$_{\text{S}}$ [$_{\text{S[1]}}$ A B]] (Figure 1b). Assign a unique label to each of four positions (called *roles*) in the tree. Specifically, assign (0,1), (1,2), (0,1,2), and (0,2) to the four positions. These are *span roles*; role (i, j) holds the category label of the constituent (if any) that spans from position i to position j (where position j lies just after the j-th symbol in the string being parsed). A span role (i, j, k) spans between positions i and k and has two daughters, one spanning from i to j and the other spanning from j to k (see Figure 1c). In T1, each role is occupied by a *filler*. For example, a filler A occupies a role (0,1) to form a *filler/role binding* A/(0,1). Then, the tree can be viewed as an unordered set of filler/role bindings T1 = [$_{\text{S[1]}}$ A B] ≡ {B/(1,2), S/(0,2), A/(0,1), S[1]/(0,1,2)} (for more details, see Hale and Smolensky 2006).

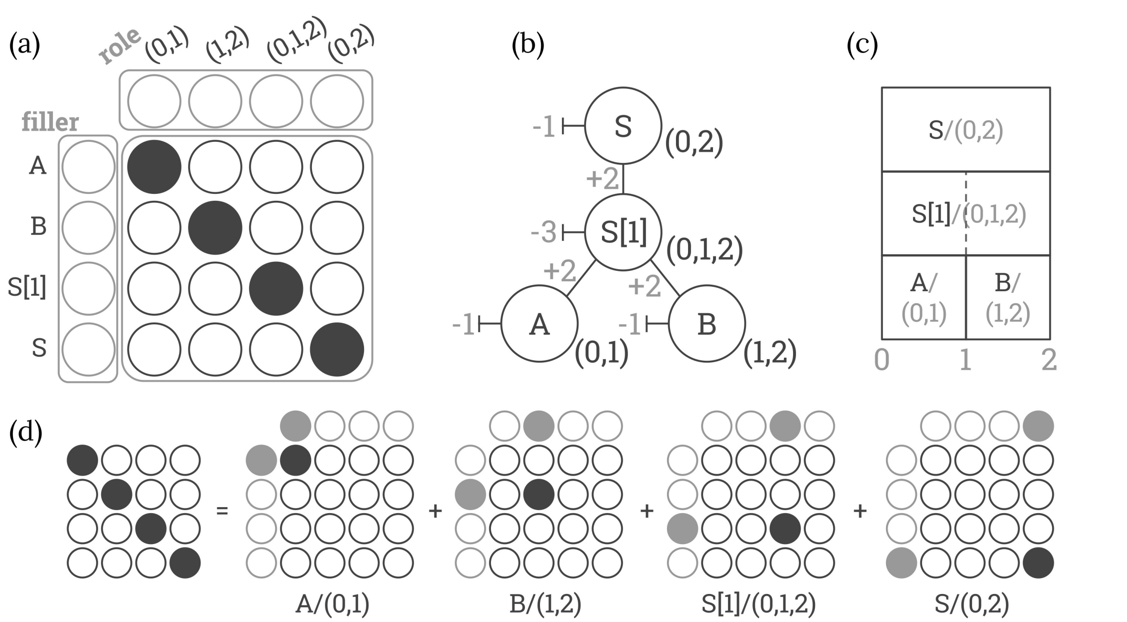

Figure 1: (a) Representation of T1. Only a subset of filler and role units is shown. (b) Grammatical constraints implemented as weight and bias values. (c) Span roles. (d) Superposition (vector sum) of filler/role bindings.

A binding of filler f and role r is represented in a continuous representation space (over connectionist processing units) as the tensor (outer) product of **f** (an activation vector[2] representing f) and **r** (an activation vector representing r). The representation of an entire symbolic structure – a set of f/r bindings – is simply the superposition (sum) of the representations of the constituent bindings (see Figure 1a and 1d).

The companion website provides software for constructing and visualizing tensor product representations, along with additional examples (see Supplementary Materials 2).

2.3 Constraints

Parsing in the network is accomplished via satisfaction of three kinds of constraints (see Supplementary Materials 1−3 for detailed explanation and additional interactive examples):

Grammatical constraints. A Harmonic Grammar (Hale and Smolensky, 2006) implements the phrase structure grammar. The central idea is to connect bindings that produce grammatical mother-daughter relationships by excitatory connections. For example, in T1 (Figure 1b), such connections mean that whenever a binding A/(0,1) is activated, it will activate its mother S[1]/(0,1,2) which in turn activates its grammatical daughter B/(1,2) and its grammatical mother S/(0,2). Each of these grammatical constraints is *local*; in T1, a constraint between A/(0,1) and S[1]/(0,1,2) is considered independently of another constraint between B/(1,2) and S[1]/(0,1,2). The degree to which each constraint is independently satisfied can be quantified by the product of the activations of the mother and daughter bindings; summing over all local constraints gives a global constraint-satisfaction measure H_G (the *harmony* of the representational state relative to the grammatical constraints).

Baseline constraint. Every binding must be at a baseline activation state: in the present example, 0.5 for every binding. A measure H_B quantifies the level of satisfaction of the constraint.

Discreteness (or quantization) constraint. Every role must be occupied by a single filler (so that the system commits to a particular structural analysis). A measure H_Q quantifies the degree of satisfaction of the discreteness constraint.

The total harmony H is defined as follows:

$$H(\boldsymbol{a};\boldsymbol{e},q) = H_G(\boldsymbol{a};\boldsymbol{e}) + H_B(\boldsymbol{a}) + qH_Q(\boldsymbol{a})$$

where $\boldsymbol{a}$ is the activation state vector, $\boldsymbol{e}$ is an external input vector (see below), and q (≥ 0) is the strength of the discreteness constraint (called *quantization strength* or *discreteness pressure*) relative to the grammar and baseline constraints.

2.4 Processing

The GSC parser seeks to maximize total harmony by stochastic gradient ascent. Formally, the state change is as follows:

$$d\boldsymbol{a} = \nabla_{\boldsymbol{a}} H(\boldsymbol{a};\boldsymbol{e},q(t))dt + \sqrt{2T}dW$$

where T is a computational temperature (determining the amount of noise) and W is the standard Wiener process. In the present study, T will be fixed to a small value.[3] See beim Graben and Gerth (2012) for a related approach to using tensor-product representations and harmony maximization to parse Minimalist grammars.

It is assumed that the GSC parser increases q gradually because its goal is to build a discrete symbolic structure. We refer to a particular update schedule of $q(t)$ as a *quantization policy*.

2 These local, "one-hot," encodings, can be easily converted to distributed representation by a simple linear transformation (Smolensky 1990).

3 Smolensky et al. (2014) used simulated annealing (S. Geman and Geman 1984), with time-varying T to reach globally optimal states. Here, the problem is not avoiding local optima, but finding the correct one of two global optima: the one consistent with previous context. It suffices to hold T at a fixed, small value so that information from previous parse states (critical to incremental processing) is not destroyed by large random noise; the small noise suffices to avoid getting stuck at critical points that are not maxima.

2.5 The GSC solution to incremental processing problems

Returning to our simple language example, we hypothesize that after processing a first word ‘A’, the human language processing system builds a blend state representing something like [$_S$ (**A**+B+C+D) (A+**B**+**C**+D)] where boldface indicates higher activation. With a second word input and gradually growing discreteness pressure, the blend state will evolve to the state representing either [$_S$ A B] or [$_S$ A C], but neither [$_S$ D B] nor [$_S$ D C].

To understand processing dynamics, we need to investigate the topological structure of the harmony surface $\{(\boldsymbol{a}, H(\boldsymbol{a}))\}$. Figure 2 shows the proposed dynamic solution to the two computational problems by visualizing the harmony surface at two different points q in processing. An interactive tutorial can be found in Supplementary Materials 3.

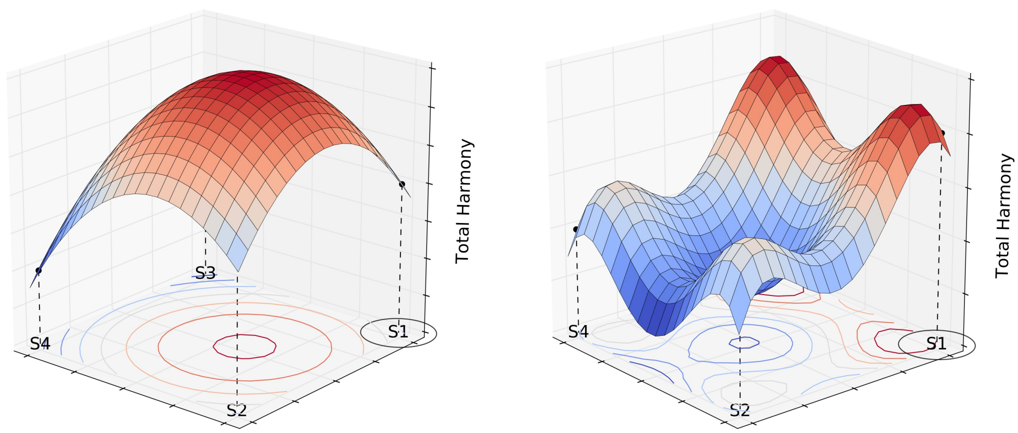

Figure 2: Schematic diagrams of the harmony surface (animated version available in Supplemental Materials). The xy-plane represents every possible mental state. Only a small subset of the states (points S1−S4) represents grammatical symbolic structures. The z-axis represents total harmony, the goodness of the mental state. On average, the system climbs a local hill to reach a local optimum. The model’s goal is to reach a target state S1 (circled) for incremental input of S1=‘A B’. (left) ‘A’ is presented when q is low: S1 and S2 have highest harmony (among discrete states) because each has ‘A’ as its first word. The baseline constraint contributes more than the discreteness constraint (q is low) to the total harmony so there is a single, global optimum which is biased toward S1 and S2. (right) ‘B’ is presented when q is high; now S1 and S3 have highest harmony as both have ‘B’ as their second word. Because of the high value of q, the surface has multiple humps each containing a discrete symbolic state. The S1 and S2 humps are broader and higher than the S2 and S4 humps due to the bottom-up input ‘B’. The location of the global optimum in the left panel is covered by the S1 hump in the right panel. Thus, with only a small level of noise, the system is likely to move to S1 (rather than S3), although both are consistent with ‘B’.

As q increases over the course of parsing, multiple harmony humps emerge because the blend states are penalized to increasing degrees. Because the model performs gradient ascent, a set of states covered by one local hump, a *basin of attraction*, is separated from the states covered by the others. The model commits to one set of states – corresponding to a meaningful parsing decision.

The GSC model parses sentences by (i) moving to a blend state which can evolve to any globally consistent interpretation (addressing the issue of temporary ambiguity) and (ii) separating that blend state from other blend states which evolve to structures inconsistent with the prior context (addressing context dependence). Both goals are achieved simultaneously by committing to particular sets of states – i.e., increasing the quantization strength q – at an appropriate pace (Cho and Smolensky 2016). If q increases too quickly, the model may make a commitment to one state over other possible continuations too early – a garden path error. If q increases too slowly, the model will fail to separate the target blend state from others that yield discrete structures inconsistent with context – a local coherence error.

In the next section, we examine this account via a simulation study.

3 Simulation

3.1 Method

Ten instances of the GSC model implementing grammar G were equipped with randomly generated distributed representations of f/r bindings. These parsed each of four sentences in L 10 times with one of three quantization policies (Figure 3).

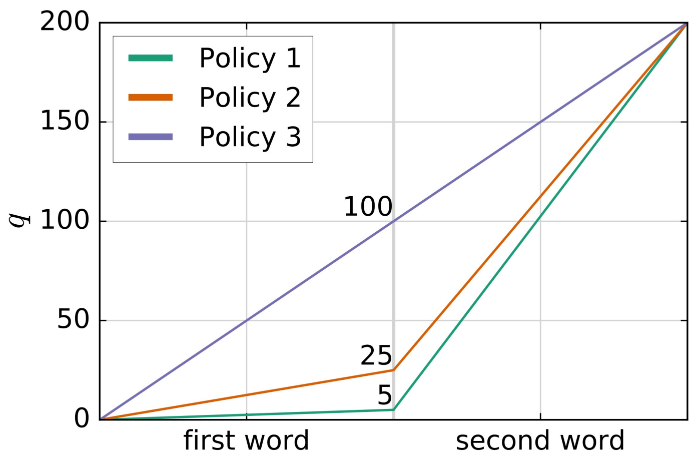

Figure 3: Three quantization policies.

The initial state of the model was determined by adding a small amount of Gaussian noise (SD = 0.02) to the globally optimal state when $q = 0$. Then, the model was given a sentence word by word. External input is provided for a fixed amount of time to the f/r binding corresponding to the current word (e.g., when ‘A’ is presented as a first word, the binding A/(0,1) is activated by external input). When the next word is presented, this binding no longer receives external input (e.g., when ‘B’ is the second word, the binding B/(1,2) is activated and the external input to A/(0,1) is removed). During processing, q was updated following one of three quantization policies (Figure 3).

Complete details of the simulations, including software enabling readers to construct their own simulations, can be found in Supplementary Materials 4.

3.2 Results

Figure 4 presents the discrete symbolic state that is closest to the model’s final activation state in each of three conditions. Parsing accuracy was near ceiling with Policy 2. Deviating from this policy resulted in errors. When a second word was presented when q was too low (Policy 1), the model behaved as if it forgot the past and chose any structure consistent with the present input (e.g., either T1 or T3 for input S1). When the wrong option was selected, this produced a local coherence error. When q increased too quickly before a second word was given (Policy 3), the model committed to either of the two choices that were consistent with context, the first word (e.g., either T1 or T2 for input S1). When the wrong option was selected, this produced a garden-path error.

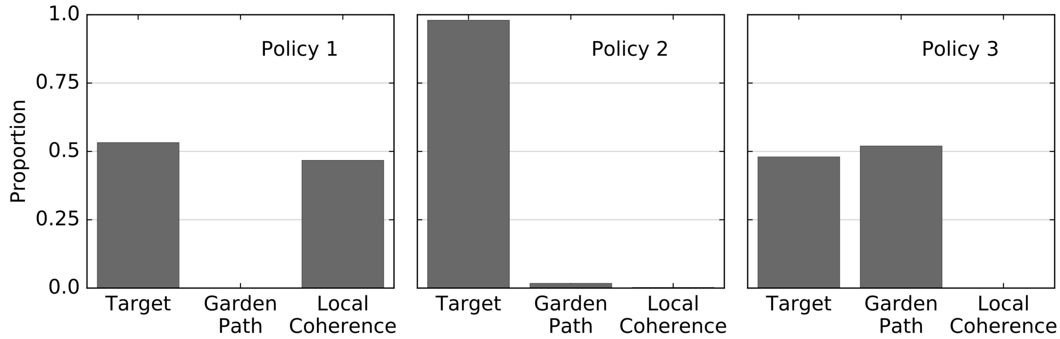

Fig. 4: Distributions of symbolic outputs for each quantization policy (N = 400; 10 model instances x 4 sentences x 10 trials each).

The source of each error type can be seen in the change in parser states over time (Figure 5). As shown in the left panel, local coherence errors result when the model processed the first word 'A' with insufficiently strong commitment to particular parses. The model therefore failed to distinguish between the structures consistent with the input (S[1] and S[2]) and those inconsistent with input (S[3] and S[4]). When the second word 'B' was presented, therefore, the model randomly chose S[1] (the target) or S[3] (a local coherence error), as both were consistent with the second word.

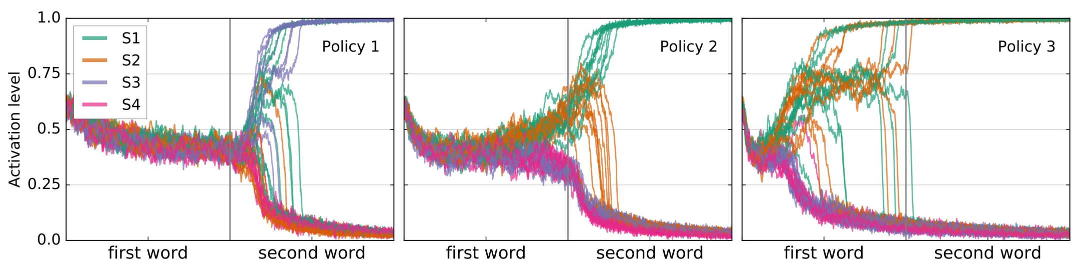

Figure 5: Ten (of 100) sample activation histories for input sentence S1. Only the full-string bindings (e.g., S[1]/(0,1,2)) for each output structure (e.g., [A B]) are shown.

As shown in the right panel, garden path errors result when the model processed the first word while over-committing to a particular parse. The model therefore chose one of the two contextually-appropriate structures (the target, S[1], or the garden path error, S[2]) over the other, blocking the influence of the second word.

Correct parsing (center panel) occurred when the first word input activated, in the form of a stable blend, both of the structures consistent with the input (S[1] and S[2]), separating these from inconsistent structures (S[3] and S[4]). This meant that when the second word was presented S[1] was chosen over S[3] (as the latter had already been separated from the blend). Thus, the model could reach the target state T1.

Figure 6 provides a global view of model performance, presenting the whole activation state of the parser at three time points with Policy 2. The left panel shows the expected activation state before the model is given a sentence. Note that not all fillers are equally activated in each role. For example, in role (0,1), fillers A and D are more active than fillers B and C. This is because these first-position bindings are in grammatical mother-daughter relationships with complete grammatical sentences (whereas B and C are not found in any sentence in first position). The middle panel presents the activation state after processing the first word 'A'. Due to the external input, filler A is strongly active in the (0,1) role. In role (0,1,2), S[1] and S[2] are more strongly active than the other fillers (as they are consistent with this input). Likewise, in role (1,2), the activations of two fillers B and C are equal and higher than the activations of other fillers; in this way, the model *predicts* possible continuations. The right panel shows that with the second word input, the model built the target structure T1.

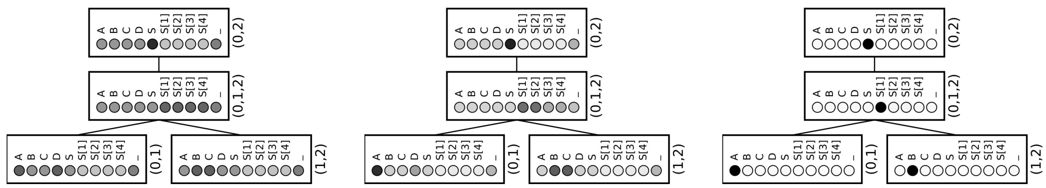

Figure 6: Expected activation states (with no noise) of fillers within each role (white = 0, black = 1) at three time points (left: before processing S1='A B' and q = 0; middle: after processing 'A' and q = 25; right: after processing 'B' and q = 200) with Policy 2. Note: the final filler symbol '_' denotes null, the absence of any filler in that particular role.

4 General discussion

Any incremental parser must address two core computational problems: maintain all interpretations consistent with context (temporary ambiguity) while simultaneously excluding all interpretations inconsistent with context (context dependency). The GSC parser solves this by exploring a "garden of forking paths" (Borges 1962; see Figure 7).

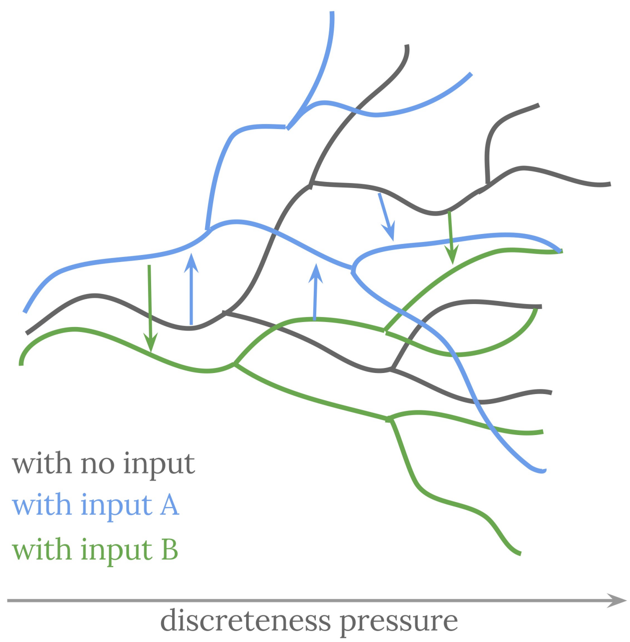

Figure 7: The GSC model explores a "garden of forking paths" (Borges 1962), the representation space augmented with the pressure to commit to a particular discrete parse. There are multiple versions of the garden, one for each external input (e.g., blue and green) and one for no word input (gray). Only one version can exist at a time. When the external input is updated, the garden transforms into another version; the parser's state moves to a nearby point on a newly emerged path (indicated by blue and green arrows). Each branching is a bifurcation (in the technical sense from dynamical systems theory: Cho and Smolensky 2016); the point at which a blend state loses its stability due to the increased tension among competing filler/role bindings. As the discreteness pressure increases, the model is forced to choose one branch over the others, which leads to different parses (terminals of the paths).

If the degree to which the GSC parser commits to particular parses is properly regulated, it can blend the future and encode the past into its present state. When the degree of commitment is not well regulated, the GSC model becomes irrational. When the commitment level is too low, the model fails to encode information in the present word input; this means it will ignore the past information when it processes a next word (leading to local coherence errors).

If the commitment level is too high, the model will choose one out of many parses consistent with the current context. The model therefore fails to recognize temporary ambiguity (leading to garden path errors)[4].

We emphasize that each blend state is not a representation of a probability distribution over parses discussed in structural probabilistic models (e.g., Jurafsky 1996). The (blend) state changes continuously but the set of possible parses (i.e., reachable terminals in Figure 7) and the probability distribution over the parses change discretely at bifurcation points. Each blend state represents a unique, inseparable mixture of (partially activated) filler/role bindings that needs to be unblended gradually by dynamics; computation is completely parallel. For a related but different approach, see beim Graben, Gerth and Vasishth (2008) where multiple interpretations are represented as the superposition of discrete partial structures (i.e., sets of fully activated filler/role bindings) each of which is computed independently.

The GSC approach is still in its infancy. Before concluding, we briefly discuss two potentially challenging problems for scaling up the model. (a) A large amount of local ambiguity (present in more complex grammars) can make it much more difficult for the GSC model to successfully parse sentences. In each role, locally ambiguous symbols have an advantage over locally unambiguous symbols. Ambiguous symbols have more connections to other symbols in other roles (reflecting their association to multiple structural options); ambiguous symbols therefore tend to win the competition regardless of context. (b) To parse variable length sentences with a fixed number of roles, the GSC model binds irrelevant roles with the null filler. This is challenging, especially in the face of favored locally ambiguous symbols; it is difficult for the model to activate the null symbol in the irrelevant roles before locally ambiguous fillers are chosen as winners. One promising approach to these issues is to adopt role-specific quantization policies, such that the discreteness pressure is stronger for elements spanning the positions already encountered in the string. For example, when processing the second word, elements spanning the first and second word will have strong discreteness pressure, but elements spanning positions beyond the second word will have weak discreteness pressure.

Although there are clearly many avenues for elaborating this approach, we believe that the GSC model represents a promising approach for satisfying the key desiderata of algorithmic accounts of sentence processing. It provides a neurally plausible, yet symbolically interpretable, means of solving the core problems in incremental processing.

Acknowledgement: We thank Geraldine Legendre, Akira Omaki, Kyle Rawlins, Ben Van Durme, and Colin Wilson for their contributions to this work, and gratefully acknowledge the support of NSF INSPIRE grant BCS-1344269. We thank Paul Tupper for suggesting the form of the H_Q function used in this work.

References

Altmann, Gerry T. M. & Yuki Kamide. 1999. Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition* 73(3). 247–264.

beim Graben, P. & Sabrina Gerth. 2012. Geometric representations for minimalist grammars. Journal of Logic, Language and Information 21, 393–432.

beim Graben, P., Sabrina Gerth & Shravan Vasishth. 2008. Towards dynamical system models of language-related brain potentials. *Cognitive Neurodynamics* 2(3), 229–255.

Bever, Thomas G. 1970. The cognitive basis for linguistic structures. In John R. Hayes (ed.), *Cognition and the development of language*, 279–362. New York: John Wiley.

Borges, Jorge Luis. 1962. The garden of forking paths (1941). In Anthony Kerrigan (ed.), trans. Helen Temple & Ruthven Todd, *Ficciones*. New York: Grove Press.

Carmantini, Giovanni S., Peter beim Graben, Mathieu Desroches & Serafim Rodrigues. 2017. A modular architecture for transparent computation in recurrent neural networks. *Neural Networks* 85, 85–105.

Cho, Pyeong Whan & Paul Smolensky. 2016. Bifurcation analysis of a Gradient Symbolic Computation model of incremental processing. In A. Papafragou, D. Grodner, D. Mirman & J. C. Trueswell (eds.), *Proceedings of the 38th Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society.

Elman, Jeffrey L. 1990. Finding structure in time. *Cognitive Science* 14(2). 179–211.

Frazier, Lyn. 1987. Sentence processing: A tutorial review. In M. Coltheart (ed.), *Attention and Performance XII: The Psychology of Reading*, 559–586. Lawrence Erlbaum Associates.

Geman, Stuart & Donald Geman. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6(6). 721–741.

Gibson, Edward. 2006. The interaction of top–down and bottom–up statistics in the resolution of syntactic category ambiguity. *Journal of Memory and Language* 54(3). 363–388.

4 For globally ambiguous input (e.g., a prepositional-phrase attachment ambiguity), the model will try to hold multiple interpretations (by keeping q in a low value) until it becomes clear no additional input is informative for parsing decisions; at this point, q is increased in order to force (stochastic) selection of one parse.

Hale, John. 2001. A probabilistic Earley parser as a psycholinguistic model. *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, 1–8. (NAACL '01). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1073336.1073357.

Hale, John. 2011. What a rational parser would do. *Cognitive Science* 35(3). 399–443.

Hale, John & Paul Smolensky. 2006. Harmonic Grammars and harmonic parsers for formal languages. In Paul Smolensky & Géraldine Legendre (eds.), *The harmonic mind: From neural computation to optimality-theoretic grammar. Volume I: Cognitive architecture*, 393–416. The MIT Press.

Hockett, Charles F. 1960. The origin of speech. *Scientific American* 203. 89–96.

Jurafsky, Daniel. 1996. A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science* 20(2). 137–194.

Konieczny, Lars. 2005. The psychological reality of local coherences in sentence processing. *Proceedings of the 27th annual conference of the cognitive science society*, 1178–1183.

Levy, Roger. 2008. Expectation-based syntactic comprehension. *Cognition* 106(3). 1126–1177.

Levy, Roger, Klinton Bicknell, Tim Slattery & Keith Rayner. 2009. Eye movement evidence that readers maintain and act on uncertainty about past linguistic input. *Proceedings of the National Academy of Sciences* 106(50). 21086–21090.

Levy, Roger, Florencia Reali & Thomas L. Griffiths. 2008. Modeling the effects of memory on human online sentence processing with particle filters. In D. Koller, D. Schuurmans, Y. Bengio & L. Bottou (eds.), *Proceedings of the 21st International Conference on Neural Information Processing Systems*, 937--944. USA: Curran Associates Inc.

Lewis, Richard L., Shravan Vasishth & Julie A. Van Dyke. 2006. Computational principles of working memory in sentence comprehension. *Trends in Cognitive Sciences* 10(10). 447–454.

Marr, David. 1982. *Vision: A computational investigation into the human representation and processing of visual information*. San Francisco: W.H. Freeman.

Smolensky, Paul. 1990. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence* 46(1). 159–216.

Smolensky, Paul, Matthew Goldrick & Donald Mathis. 2014. Optimization and quantization in gradient symbol systems: a framework for integrating the continuous and the discrete in cognition. *Cognitive Science* 38(6). 1102–1138.

Smolensky, Paul & Géraldine Legendre (eds.). 2006. *The harmonic mind: from neural computation to optimality-theoretic grammar. volume 1: cognitive architecture*. Cambridge, MA: The MIT Press.

Tabor, Whitney, Bruno Galantucci & Daniel Richardson. 2004. Effects of merely local syntactic coherence on sentence processing. *Journal of Memory and Language* 50(4). 355–370.

Tabor, Whitney & Sean Hutchins. 2004. Evidence for self-organized sentence processing: Digging-in effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 30(2). 431–450.

Tabor, Whitney, Cornell Juliano & Michael K. Tanenhaus. 1997. Parsing in a dynamical system: An attractor-based account of the interaction of lexical and structural constraints in sentence processing. *Language and Cognitive Processes* 12(2–3). 211–271.

Vosse, Theo & Gerard Kempen. 2000. Syntactic structure assembly in human parsing: A computational model based on competitive inhibition and a lexicalist grammar. *Cognition* 75(2). 105–143.