



Language, chaos and entropy: A physical take on biolinguistics

Diego Gabriel Krivochen

Abstract

In this paper we will try to provide arguments for the thesis that language is a physical system aiming at *justificative adequacy*: what architectural properties license the occurrence of certain emergent phenomena. We will claim that the derivational dynamics that can be found in language (and other systems of the mind) should be analyzed from the perspective of complex non-linear systems, as an open dynamic system. We will propose an oscillatory engine for linguistic computations, which yields cycles as a natural emergent property given mutually incompatible tendencies between output conditions: global semantic effects and local linearization requirements. This architecture, in which structure building is conditioned by irreconcilable conditions, configures a kind of dynamical system well known in physics: a *dynamical frustration*. We will attempt to show that interesting effects arise when we consider that there is a dynamical frustration at the core of cognitive dynamics.

Keywords: dynamical frustration, entropy, Physics of Language, derivations

1 Some introductory considerations¹

The framework we adopt here, which has been dubbed ‘Physics of Language’ PoL (a term coined by Douglas Saddy; it was also the title of a recent conference organized by Roger Martin in Sophia University, Tokyo) is a program which analyses

¹We would like to thank Doug Saddy, Massimo Piattelli-Palmarini, Giuseppe Vitiello, Juan Uriagereka, and David Medeiros for enriching and stimulating discussion about different aspects of the theory presented here. Doug Saddy has been particularly patient with our never ending questions, and for that he has our eternal gratitude. Speakers and audience at the First International Symposium on the Physics of Language (Sophia University, Tokyo) have enriched this work with pertinent questions and suggestions, and we owe them as well. The present paper has also benefited from the comments of an anonymous *Iberia* reviewer, which we have done our best to incorporate in the present version. We acknowledge partial funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 613465. The usual disclaimers apply.



natural language in cognition as a physical system. Since language is part of the natural world in a non-trivial way (that is, it not only occurs in the Universe, but is a system that can be assimilated to others in that Universe), as Chomsky and mainstream Generative Grammar have claimed over the years (Chomsky 1995 et. seq.; Boeckx 2010a, 2010b; Di Sciullo & Boeckx 2011, among many others), it is assumed to be ruled by the same laws *at a principled level*: abstracting material properties of the elements involved, the generation of complex structures is considered to be analogous in both mental and extra-mental domains. The PoL program strives to answer questions concerning the integration of what we refer to as *language* in a system of interacting cognitive capacities and furthermore advocates that language as a *physical system* should therefore not be studied in isolation, but rather in interaction with other systems. As this interaction occurs in the so-called ‘natural world’, it is constrained by physical laws which are in turn particular instantiations of mathematical possibilities, and restrict the amount of possible biological configurations (Uriagereka 1998). Considering this scenario, PoL proposes the following tenets:

- (1) a. Language is part of the ‘natural world; therefore, it is fundamentally a physical system.
- b. As a consequence of (a), it shares the basic properties of physical systems and the same principles can be applied, the only difference being the *properties of the elements* that are manipulated in the relevant system and the *scale* in which operations apply.²
- c. The relevant operations are taken to be very basic, simple and universal, as well as the constraints upon them, which are determined only by the interaction with other systems.

PoL licenses the possibility of looking for biological, physical and mathematical properties of computations (i.e., syntax) in three levels: description, explanation and justification. This claim is essential for our argument: we are not saying that language ‘is like’ a physical system and, therefore, a mathematical structure. We are saying that language *is* a physical system, as it exists in nature, is itself the result of the interaction of other systems and, as we will see below, shares deep properties with systems traditionally considered within the domain of physics, well beyond the level of metaphor. Interesting empirical predictions derive

²A reviewer pointed out that the space-time continuum and particle physics refute our thesis, since the former is a continuum whereas the latter is discrete: notice that the characteristics of the elements are irrelevant, since our focus is put on the operation that generates complexity in the physical world. In any case, a quantum field theoretical perspective would in turn refute the reviewer’s objection: particles are not discrete, but dynamical ‘local condensations’ of an underlying field. There is nothing ‘discrete’ in particles, if seen from QFT.

from the PoL perspective (regarding, for instance, displacement phenomena in natural language, the ontology of the structure-building algorithm, processing issues, brain dynamics ... see Krivochen 2015a, 2015c; Grindrod et al. 2014; Piattelli-Palmarini & Vitiello 2015, 2015; Medeiros 2015; Saddy 2016), and those can be observationally contrasted (we are consciously adopting a form of Carnap's 1966 model here) in order to determine the descriptive, explanatory and justificative power of the model. The description is 'the what', the explanation is 'the how', and finally, the justification is 'the why'. The latter has been either taken for granted or done in a truly non-minimalist way both substantively and methodologically (consider, for instance, a feature-valuation driven syntax, and the amount of stipulations it requires). Our effort, then, will focus on trying to set a radically minimalist alternative of justification, taking into account that a theory of the physical universe must address all three: in this attempt of methodological and substantive integration of the study of language within the more general study of physical (which does not amount to 'material', *contra* Postal 2012: 5) systems at different scales lies the main difference between PoL and other approaches (which by no means implies that those approaches are to be rejected). Attempting justification is what we understand as the ultimate goal of going 'beyond explanatory adequacy' (borrowing the term from Chomsky 2004). This has a direct consequence for the research program we are proposing: it is simply impossible to attempt to explain a problem from just one perspective, but we need to understand the multiple facets of a phenomenon: its computational dimension, the biological properties of the system that allows such a computation (attending to Marr's 1982 argumentation about the need of an *implementational level*) and the physical principles that license this biological configuration, expressed as a mathematical structure. From a methodological point of view, then, our work has consequences for the foundations of the 'biolinguistic enterprise' (Chomsky 2005, 2007; Boeckx 2010b; Di Sciullo & Boeckx 2011, among many others) and its place within linguistic inquiry.

In this particular paper, we will make explicit a set of formal assumptions regarding the architecture of a model for language in cognition, partially shared with recent syntactic research (Culicover & Jackendoff 2005, Uriagereka 2012, Lasnik & Uriagereka 2012):

- There is a computational system in charge of building symbolic structure, which we will refer to as the 'syntax'. This system is *not* domain-specific or encapsulated. Thus, we make no difference between FLN and FLB (*contra* Hauser et al. 2002), insofar as the so-called interfaces (sound-meaning) are computational (i.e., syntactic) in nature.
- The formal mechanics of the system is derivational but constraint-restricted: derivations are free in nature, but if the generative component interfaces

with interpretative components (e.g., semantics, phonology), input conditions of those systems restrict the possibilities of generation.

- Derivations are continuous, oscillatory processes (*contra* Chomsky 2007, where the diachronic nature of the generative operation *Merge* is explicitly rejected, and the derivational engine is primarily proof-theoretical).
- Constraints are only determined by interpretative systems. Derivations are syntactically unconstrained.
- Complexity arises in cognitive systems independently of each other, unless there exists a system that interfaces with them (e.g., the Conceptual-Intentional C-I component, in charge of semantic interpretation at both natural and non-natural meaning levels, and conceptual structure). Thus, for instance, Conceptual Structures CS manipulating generic concepts (see Jackendoff 2002; Moss et al. 2007; Uriagereka 2012, 2014, among others) are syntactically derived without there being natural language involved in any way.
- Semantic requirements shape and drive the linguistic derivation.
- Cognitive interfaces are opaque (i.e., there is no one-to-one mapping or isomorphism holding between systems).

The next sections will be devoted to make our system explicit: how do we derive? How do we evaluate the outputs of computations? What do we derive with? Those are all questions we will deal with in the rest of the paper. Moreover, we will make explicit some assumptions about the mathematical properties of the generative component which go far beyond orthodox assumptions.

1.1 How do we derive?

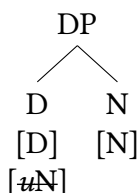
In this paper we will focus on prolegomena for a mathematical formalization of the biological and computational dimensions of language, provided that we assume that the physical principles that license a biological configuration are, in turn, particular instantiations of more general and less restrictive mathematical structures: we will pursue the more restrictive claim that the accurate mathematical model to formalize linguistic derivations is some form of non-linear dynamics (even though, as a reviewer correctly pointed out, this does not mean that all physical systems are non-linear, for instance, the *superposition principle* in quantum mechanics is a linear superposition of eigenstates). This is already a departure from orthodox generative models of language: the generative algorithms used from Chomsky (1959) onwards (summarized in the famous ‘Chomsky hierarchy’

for formal grammars and languages alike) assume linguistic derivations to be *uniformly* Turing-computable (Watumull 2012b, 2012a; Watumull et al. 2014), including as a subset the proposals of Joshi (1985) and Uriagereka (2012), who argue for a mildly-context sensitive computational system. Here and in past works we have argued that all-the-way-down binary Merge falls short in some instances, while assigning way too much structure to strings in some others (see Lasnik 2011, Krivochen 2015a, Bravo et al. 2015). We will argue that the computations that underlie human language as a cognitive capacity are continuous, parallel, and non-linear rather than strictly sequential, linear, and function-theoretic based (following Milner 2006, Goldin & Wegner 2005, Wegner 1997, among others), and thus cannot be captured by classical computational algorithms, even Turing machines. On the other extreme of the Chomsky hierarchy, some portions of natural language grammars (those that involve no hierarchy between constituents, like iteration and some kinds of coordinations, as shown in Krivochen 2015a and Krivochen & Schmerling 2015) can even be formalized via a finite state grammar (Uriagereka 2008; Lasnik 2011) with better results in terms of adjustment between structural description complexity and descriptive adequacy than higher-level phrase structure grammars or transformational grammars. In this respect, a uniform conception of the computational substratum of language is thus procrustean, as argued in Krivochen (2015a) and Lasnik & Uriagereka 2012 (but the claim can be traced as far back as Chomsky & Miller (1963)). The structure-building operation strictly considered, and concomitant operations (e.g., Label) are to be reformulated taking structural dynamicity into consideration, which in turn has crucial consequences for the model of the mind assumed by a given linguistic formalism. It must be noticed that the use of mathematical tools to formalize properties of language is not a new idea (see Chomsky 1955; Harris 1968; the papers collected in Jakobson 1961, among others). In recent times, for example Di Sciullo & Isac (2008), work within a *set-theoretical* framework in which the generative operation Merge is driven by *proper inclusion* between feature bundles of lexical items, that is, the set of intensional properties of item A must properly contain that of item B for them to Merge and thus build a more complex structure; assuming, as orthodox Minimalism does, that a lexical item is nothing but a bundle of such features – like Case, Person, Number, Tense, Category, etc. – some of which enter the derivational workspace valued (and are, therefore, legible by the relevant systems that interface with the computational system; in the case of language, *sound* and *meaning*) and other must be valued throughout the derivation (Chomsky 1995; Pesetsky & Torrego 2007; Panagiotidis 2009b, Wurmbrand 2014, among many others). For example, using Di Sciullo & Isac’s (2008) notation:

- (2) Lexical Item $\alpha = \{N\}$ (an interpretable categorial ‘noun’ feature)
 Lexical Item $\beta = \{D, uN\}$ (an interpretable categorial ‘determiner’ feature)

and an uninterpretable and unvalued ‘noun’ feature, to be valued via Merge with an element whose features are a proper subset of those of β)

Merge (α, β) =



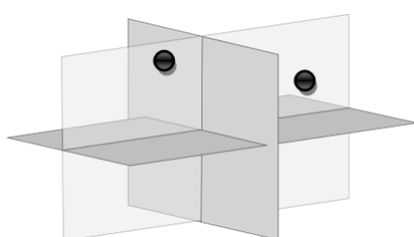
In a local relation created by Merge, the interface uninterpretable feature $[u\text{N}]$ is deleted, and the resultant syntactic object contains only elements which the interfaces can read.

We will pursue a different possibility here, since the set-theoretical approach to derivations require a rich feature system and an operation to relate those features (Agree) in order to ensure the generation of fully legible units for the systems of sound and meaning which interface with the so-called ‘Narrow Syntax’ (Chomsky 1995 et seq.). For example, Di Sciullo & Isac’s (2008) approach require a subset relation between lexical items to enter into a Merge relation, which requires a highly specified feature matrix for each lexical item (including categorial features, person, number, and other formal features like $[\text{Wh-}]$ for interrogative elements and so on) and also principles determining interpretability and valuation conditions: if a feature is to be defined as a valued dimension (e.g., + velar, in phonology, means ‘positive value for the dimension ‘velar’’). The theory grows in complexity without improving descriptive or explanatory adequacy, since free-Merge systems (Boeckx 2010b; Chomsky 2004) work just as fine, as we will attempt to demonstrate, if constraints over the derived structures are appropriately formulated. We will work with an essentially unrestricted structure-building formal operation, and no features (this is, no $[\text{value-F}]$ primitives and no Agree as the driving force of the syntactic component); all constraints in this system are determined at the point of interpretation for each derivational cycle. The general architecture resembles an Optimality Theory-like model, a GEN(erator) – EVAL(uator) dynamics: there is a structure-building operation and a series of interpretative routines, varying from system to system, which evaluate the output of each operation in real time (an architecture which is very close to the *interactional model of computation* of Milner 2006; Goldin & Wegner 2005, 2007). That said, our structure-building proposal involves relating points in cognitive spaces with a specific topology thus forming n -manifolds (Saddy 2016, Krivochen in progress). In strictly mathematical terms, the operation can be formulated as follows:

- (3) *Concatenation* defines a *chain* of coordinates in n -dimensional metric workspaces W of the form $(x, y, z \dots n) \subset W_x, \dots (x, y, z \dots n) \subset W_y, \dots (x, y, z \dots n) \subset W_n$.

Each set of coordinates, which for all intents and purposes can be identified with a vector if we model the result of *concatenation* as an orientable manifold, depends on the number of dimensions in the relevant generative workspace (a mental working bench, see Stroik 2009; Stroik & Putnam 2013 for related claims about the notion of working bench in syntax; and Uriagereka 1999, 2002b: for the first development of parallel computations in a Multiple Spell-Out model), such that an element *x* is to be defined by *all* of its coordinates in *W*. Given this scenario, let us see how a binary phrase marker *qua* manifold would be formed:

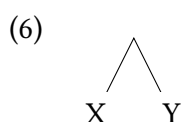
(4) Workspace 1 (W_1):



Concatenate applies in the following form to elements active at the same time in the same workspace:

(5) $Concatenate(X, Y) = \{(x, y, z) \in W_1, (x', y', z') \in W_1\}$

The coordinates of the result of the operation are defined as the Cartesian product of the (in this case) two sets of coordinates of the elements involved in the merger. In the more familiar tree form, the result would be represented as (6), which is *just a model*:



Let us assume (6) is a sortal structure, to be merged with an eventive structure, to form (6):

(7) He saw the book

Concatenation would apply monotonically all the way (Uriagereka 2002b), since the computational dependency between all objects involved in the derivation is uniform: we have monotonic predication relations all throughout the phrase marker:

(8) $Concatenate(\text{the, book}) = \{(x, y, z) \in W_1, (x', y', z') \in W_1\}$
 $Concatenate(\{\text{the, book}\}, \text{saw}) = \{(x'', y'', z'') \in W_1, (x''', y''', z''') \in W_1\}$

The following derivational step takes both structurally identical units and merges them, generating the desired output in (8). Needless to say, the model derivation in (8) is nothing but a model, and does not entail that the procedure in physical terms is sequential or proof-theoretical. A mechanistic implementation of structure building, however, is bound to step-by-step formalization, even if using parallel stacks (Medeiros 2015). In physical terms, if the ground state of cognitive dynamics is a topological space corresponding to an ultrametric field (see Roberts 2015 for an application of ultrametric considerations to the phrase structure template), relating elements basically amounts to making them interfere, which disrupts the ultrametricity of the field and introduces a metric between said elements. That is the idea of Uriagereka (2011) and Saddy (2016), and for a field approach we think it is essentially correct. While in an ultrametric space the distance function d between a point $x \in X$ and a point $y \in Y$ (for X and Y manifolds) is fixed and > 0 , the intersection of X and Y at x, y means that $d(x, y) = 0$. If such intersection happens, we actually have a well-established mathematical/topological property that determines a class of possible consequences. This property is the *center manifold theorem* (CMT):

When an attractor of a high-dimensional dynamical system [...] becomes unstable there is typically one direction in the high-dimensional space along which the restoring forces begin to fade; while in other directions, the stabilization mechanism remains strong. The one direction along which forces fade spans the center manifold. (Schöner 2009: 9)

This process reduces the dimensionality of a problem near equilibria, and is a well-known tool used to reduce the complexity of an infinite-dimensional problem to a finite set of solutions with low dimensions. Cyclic structure building, we argue, depends essentially on the CMT, which defines cycles at equilibria. If phrase markers are conceived of as manifolds whose dimensional complexity is affected by structure building operations (Krivochen 2015a; also Piattelli-Palmarini & Vitello 2015, 2016, for a use of quantum field theory's *ladder operators* σ^+ and σ^- in step-by-step derivations), there is an upper limit, a critical dimensional value determined by the properties of the neurocognitive substratum of the system after which no further structure can be accumulated in a single workspace without triggering the CMT.³ This will be essential for our definition of a *cycle*.

Returning to the structure building problem from a linguistic perspective, free generation has already been argued for in, for example, Chomsky (1995, 2004),

³In Krivochen (in progress) we proposed that 3 is that critical dimensional value, since we as humans do not seem to be able to operate in metric spaces of more than 3 spatial dimensions (mainly, because we do not need to do so in our phenomenological experience). In any case, the specific critical value that triggers the CMT is an empirical matter, and must be treated as such.

but from a different perspective: in his model, all operations are driven by features in the Narrow Syntax. Moreover, our derivational model is closer to that proposed by Putnam (2010) insofar as it is only interface-constrained. For the purposes of the formal modelling of derivations, we propose an extremely local evaluation mechanism, call it *Analyze_{IL}* (SO) (that is, analyze a syntactic object SO from the interface level IL) that applies after each instance of *Concatenate* and determines whether the object is fully legible by the relevant interface; in the case of language, Conceptual-Intentional C-I for meaning and Sensory-Motor S-M for sound. Assuming the discussion in Boeckx (2007, 2010a), *Analyze* can be seen as the interpretative systems (whichever they turn out to be) accessing the syntactic workspace in which Concatenation applies, and taking the minimal unit they can read. Merge, as we take it, is a completely free, *n*-ary operation that can apply as long as the objects to which it applies have the same format, motivated by interface conditions (this is, $\{\alpha\}$ is trivial at the interface levels, while $\{\alpha, \beta\}$ is not⁴). In the case of language we have lexical items⁵, and we can say that they have the same format (be them ‘lexical categories’ or ‘functional categories’) since they share a nature, they are linguistic instantiations of elements that, *per se*, are not manipulable or interpretable, like generic concepts (used by the visual capacity, for example, to help organizing entities in the phenomenological world; see Jackendoff 2002 for a linguistic take on conceptual structures, Moss et al. 2007 for a neurocognitive account). Once the ultrametricity of the ground state of the system dynamics has been disrupted by an external perturbation, the only formal attribute of concatenation would be putting things together (bringing manifolds closer and making them intersect, in the topological interpretation), without any restriction by principle pertaining to the nature or number of objects, since it would be a stipulation. An example of the formal model for derivational dynamics we are proposing is the following:

$$(9) \text{ Concatenate } (\alpha, \beta) = \{\alpha, \beta\}$$

⁴Boban Arsenijevic (p.c) claims that ‘ $\{\{a\}\}$ is non-trivial in at least one faculty: the arithmetic capacity. Hence, output conditions can’t be that bare to favor a binary merge’. However, our position is that if we assume a *strong* and *dynamic* version of Full Interpretation that states that *any derivational step must be interface-justified*, that is, the application of any operation must lead to a legible object the application of Merge to a single object is trivial in any faculty, as it does not contribute in any way to legibility. If $\{\{a\}\}$ is legible for the interface system the arithmetic faculty has to interact with, then why apply Merge in the first place, to apply Merge to a single object is trivial in any faculty. If $\{a\}$ is already legible in the relevant interface level, then why apply Merge in the first place? It would be computationally redundant, and therefore far from Minimalist. We maintain that binary Merge is a non-trivial option, just that it is not the only option or the simplest for all cases, *contra* Chomsky. We therefore reject any proposal of *unary Merge* on interface grounds.

⁵As a matter of fact, we have *roots* semantically defective and procedural features that make them interpretable at C-I, but we will use the term ‘lexical items’ for the time being. See *infra*.



Analyze_{IL} { α , β } [is { α , β } fully interpretable by the Interface Level IL? Does it contain superfluous elements?]

(if Analyze_{IL} results in legibility at IL, IL coopts the relevant unit and assigns it an interpretation)

The idea of ‘invasive interfaces’ is a natural result of making interpretative and generative systems interact within a formal architecture. If generation is restricted to a single operation *concatenate*, which is the optimal scenario and it occurs in an *n*-dimensional workspace, it is only natural that the operation cannot *read* or evaluate what it has built. On the other hand, it is also only natural that the evaluator function *Analyze* is not separated from the interfaces but in itself be the set of so-called Bare Output Conditions (or legibility conditions) each cognitive interface has. Then, if we assume considerations of computational simplicity like ‘maintain as few structure at once in *W* as possible’ for a system with finite resources, that is, ‘transfer as soon as you can’ (and, more generally, ‘apply any operation as soon as you can’). ‘As soon as you can’ is determined not by internal syntactic conditions like feature valuation timing (such conditions do not exist in our proposal), but by the system(s) that use and interpret syntactically derived cycles. In this way, the generative workspace is nearly emptied several times along a derivation thus liberating working memory without the concomitant problems of defining, for example, endocentric transfer domains (i.e., Chomskyan *barriers* / *phases*). The existence of cycles in PoL derives from the inner dynamics of the derivation, if structure building is understood as manifold interaction in a metric space. A cumulative derivation proceeds as syntactic cycles are completed, intersect with other manifolds, and reach a critical dimensionality value that spans the CMT, squeezing the manifold through a faded dimensional attractor, into a metric space of lower dimensionality but richer in distance functions. The implementation of a mechanism that can reduce the dimensionality of a manifold once a critical value is reached within an overall cumulative architecture in which these lower-dimensional manifolds can be inputs to further operations is one of the main features of our oscillatory computational dynamics. Carr (2006) defines the CMT as a way of simplifying otherwise very difficult (or downright unsolvable) problems involving multiple dimensions in dynamical systems, again, approaching critical points:

Centre manifold theory [sic] is a rigorous mathematical technique that makes this [dimensionality] reduction possible, at least near equilibria.

In sum, while some properties of complex, *n*-dimensional complex systems might be difficult to solve with ordinary differential equations, solutions in a cen-



ter manifold can always be described by a finite-dimensional system of such equations, which is indeed an advantage from both theoretical and empirical perspectives. If we are dealing with cumulative structure building, that can get high-dimensional very quickly as we add more and more elements to our phrase markers. The existence of equilibria points, which arise in a spin glass-like frustrated system like the one that we argue configures the ground state of cognitive dynamics (Saddy 2016; Saddy & Krivochen 2016) guarantees the CMT will apply regularly. Thus, we don't really let the cycles grow too big, but we don't need to stipulate that 'every phrase is a phase' (Epstein & Seely 2002) either.

1.2 What do we derive with? Some notes on architectural issues

We will now explicit the assumptions we make regarding the objects which the formal operation Concatenation manipulates in natural language derivations.

In our proposal, a derivation does not start with a Numeration (a set of tokens with numerical subindexes indicating how many times they will be used in a derivation, see Chomsky 1995; also Uriagereka 2008: 16⁶), but with a pre-linguistic purely conceptual structure, in the line of Fodor (1975) and, more recently, Jackendoff (2002), Culicover & Jackendoff (2005), Uriagereka (2008), and the sense in which D-Structure is understood in the CLASH model (Uriagereka 2012, 2014). That structure is syntactic in a wide sense, as concepts are structured (taking 'syntactic' not in the narrow sense of 'linguistically structured' but in a strict sense of 'structured').⁷ This conceptual structure, shaped by the speaker's intention to convey a certain propositional meaning through linguistic means, is what, in our proposal, drives *Select*, the selection of a subset of LEX, in turn a set of lexical *types*. The assumption we make at this respect is the following:

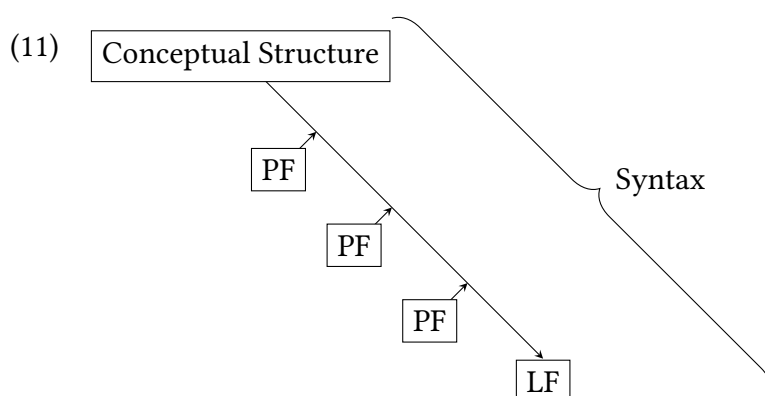
(10) **Minimal Selection:** *Select the minimal amount of types that can instantiate*

⁶[...] he [Chomsky] wants LA to be not a set of lexical types, but rather a set of tokens. [...] Chomsky wants to identify chains at LF as equivalence relations over the objects in the numeration, but for that he needs lexical tokens, not types. ' (Uriagereka 2008: 16)

⁷Cf. Culicover & Jackendoff (2005: 20 fn. 8): 'Algebraic combinatorial systems are commonly said to "have a syntax". In this sense, music has a syntax, computer languages have a syntax, phonology has a syntax, and so does Conceptual Structure. However, within linguistics, "syntax" is also used to denote the organization of sentences in terms of categories such as NP, VP, and the like. These categories are not present in any of the above combinatorial systems, so they are not "syntax" in this narrower sense. Throughout this book, we use "syntax" exclusively in the narrow sense.' In this paper, and in general within our theory, 'syntax' is used in the wider sense, for two main reasons: to begin with, there is no compelling evidence that the 'syntactic mechanisms' vary from one system to another (except insofar as the units affect the algorithm, in case that actually happens); and also, an adequately wide formalization of syntactic mechanisms could reveal deep facts about the structure of more than a single system. While Balari & Lorenzo (2013) claim that they belong to a rather extreme position, namely, that no portion of FL lacks a corresponding part in other cognitive systems, it is not clear at all why they maintain FL and not directly eliminate FLN/FLB as trivial.

a conceptual structure CS into a linguistic structure LS losing as few information as possible.

The intuition behind this assumption is clear: we want to instantiate a CS in the most economical way possible⁸, *ceteris paribus*. Given the fact that the CS includes not only rough propositional content but also added information (what most linguists would put under the ‘pragmatics’ label: inferences, and other extra-propositional which is, nonetheless, built upon the clues syntactic structure provides the semantic component with), the reference set for each potential derivation is unary: there is one and only one candidate which can express CS in an optimal way (a crash-proof syntax, in the sense of Putnam 2010) with a minimal amount of entropy per cycle. Assuming the existence of (some form of) a Lexicon for human language where units to be manipulated are stored, the formal operation *Select* builds an array of lexical *types* from that Lexicon (see Krivochen 2015c for details and implementation on a concrete derivation). The overall formal architecture we assume for language (heavily inspired in Uriagereka’s 2012 CLASH architecture), then, is the following:



The linearization of phrase markers is cyclic (i.e., units are taken by the SM interface as soon as possible in sub-representations), following the seminal proposal by Uriagereka (1999, 2002b). The long arrow represents the fact that the

⁸In slightly more technical terms, to be refined below, Selection must *reduce entropy*, at the cost of energy necessary to search the phase space for the relevant lexical attractors. If the theory of structure building we have developed in past works is correct, the generative algorithm, driven by interface requirements, should also be ‘counter-entropic’ at each derivational step. David Medeiros (p.c.) has commented at this respect that ‘My uninformed intuition strongly suggests that, *qua* natural system, it should rather maximize entropy. I had read something recently about extending this to the idea that natural systems tend to maximize *future* entropy, (which turns out to make different and better predictions in some domains).’ We have had no access to the predictions Medeiros mentions, but will address the problem by proposing an alternative system, based on *locally counter-entropic structure building*.

derivation is driven by semantic requirements, that is, the need to map the Conceptual Structure CS into a Logical Form LF reducing derivational entropy via Concatenation. The orthogonality between the CS-LF arrow and the PF arrows represent the dynamical frustration (coexistence of mutually incompatible requirements over the same system) between global semantic requirements and local linearizable cycles, along the lines of Uriagereka (2012) and Saddy (2016).

As far as the ‘derivational bricks’ are concerned, we assume only two types of elements in linguistic derivations, all stored in the Lexicon (following Sperber & Wilson 1995; Wilson & Sperber 2003; and Escandell Vidal & Leonetti 2000, 2004, 2011):

- (i) **Conceptual elements:** i.e., roots, semantically underspecified, malleable and generic meaning. *Roots* are pre-categorial linguistic instantiations of a-categorial generic concepts. Generic concepts are ‘severely underspecified’, since they are used by many faculties (organizing sensorial information, creating symbolic representations of the phenomenological world), and therefore cannot have any property readable by only some of them. Roots convey generic conceptual instructions, and their potential extension is maximal (expressible by the superset that properly contains all referential sets), given their semantic underspecification: bare roots have no (spatio-temporal) anchor.
- (ii) **Procedural elements:** provide instructions as to how to interpret a root (anchor it to a specific point in Space or Time) or a relation between roots -including complex structures-. Procedural elements, for expository purposes, can be identified with Functional Categories, but there is a fundamental difference: the procedural character of a node is of no relevance to syntax, as *Concatenate- α* , we have said, is *blind* to the inner characteristics of the elements it manipulates (that is why it can apply cross-modularly). ‘Procedurality’ is thus recognized and relevant *at the semantic interface*, not before. Following and expanding on Escandell Vidal & Leonetti (2000), we will identify Determiner, Tense, Cause, Complementizer, and Preposition as procedural elements. The instructions conveyed by procedural elements play two main roles:

- Restrict reference in terms of a (finite) proper subset of the root. Each element restricts the set of the root’s intensional characteristics in different ways. For example, for X and Y procedural elements:

$$\begin{aligned} \sqrt{} &= \{\alpha, \beta, \gamma, \delta \dots n\}^9 \\ (X, \sqrt{}) &= \{\alpha, \delta\} \end{aligned}$$

⁹Bare roots have maximal extension, in the following sense: the bare root $\sqrt{\text{APPLE}}$ denotes all



$$(Y, \surd) = \{\beta, \gamma\}$$

- Provide instructions as to:
 - where to retrieve information? (i.e., considering the mind to be massively modular, in which module should the parser look for the relevant information?)
 - what kind of information to retrieve? (i.e., eventive, sortal, causative, locative...?)

Therefore, *procedural elements convey locative meaning* in the sense that they relate a *figure* (i.e., the root) to a *ground* (a set of intensional properties), and they are thus predicators (i.e, functors).¹⁰

Formally, we have the following situation in the case of a lexical item LI:

- (12) A lexical item LI is a structure $\{X... \alpha... \surd\} \in W$, where X is a procedural category (Determiner, Time, Preposition) specified enough as regards distribution, α is an n number of non-intervient nodes for category recognition purposes at the semantic interface, \surd is a root and W an n -dimensional workspace.

Now, we will not assume a monolithic Lexicon, as in orthodox versions of the Minimalist Program, but work with the concept of *type-Array*. In such an Array,

the apples in the Universe that have existed, that exist, that will exist and also those that exist in non-factual possible worlds. In this sense, procedural elements restricting such maximal extension become indispensable for manipulation when Bare roots have maximal extension, in the following sense: the bare root \surd APPLE denotes all the apples in the Universe that have existed, that exist, that will exist and also those that exist in non-factual possible worlds. In this sense, procedural elements restricting such maximal extension become indispensable for manipulation when deriving semantic representations (i.e., Logical Forms). It is because of this maximal extension, also, that bare roots cannot be manipulated by C-I, which affects the label recognition algorithm.

¹⁰Consider, for instance, the formalization of the meaning of the prepositions ‘inside’ and ‘outside’ in terms of polar coordinates (Zwarts & Gärdenfors 2016: 11), for x = distance between a *trajector* (a.k.a. *figure*, see Zwarts 2003) and the *space of a landmark* (a.k.a., *ground*) $S(L)$; θ = angle between the trajector and the x-axis; ϕ = angle between the trajector and the positive z-axis; and r_L is the radius of the landmark L :

- (i) $inside(L) = \{ \langle x, \theta, \phi \rangle \in S(L) : x \leq r_L \}$
- (ii) $outside(L) = \{ \langle x, \theta, \phi \rangle \in S(L) : x > r_L \}$

Since landmarks are assumed to be circles in this work, if the linguistic space (as opposed to the ground state of cognitive dynamics) was ultrametric, then every point would be at a distance $x = r$, neither inside nor outside $S(L)$, or, rather, if $x = 0$, every point would be the *origo*. Importantly for a topologically based theory, the semantics of prepositions (and, more generally, localist theories of cognition) thus *requires* metricity. See, however, Roberts (2015) for an ultrametric approach to X-bar structure.

there are no numeric subindexes indicating how many times a unit is to be used in a derivation (as in Chomsky’s proposal), nor phonological features or formal features: there are just *types* to be instantiated as *tokens* in a derivation according to interface requirements. There is no *a priori* limit to the number of times a type can be instantiated as a token, but those established by C-I interface conditions, particularly Conservation of conceptual information (Lasnik et al. 2005; Krivochen 2011, 2015c).

Let us assume that we have an *Array* consisting on a root and a procedural node D(eterminer). Assuming, along orthodox lines, that the identifying label of the structure $\{\alpha, \beta\}$ must be *either* α *or* β (which seems to be the simplest option, as it does not include a new element in the derivation) for the purposes of further computations and interface recognition, the derivation could go either of the following ways (14) or (15):

$$(13) \text{ Concatenate } (\sqrt{}, D) = \{\sqrt{}, D\}$$

$$(14) \text{ Label } \{\sqrt{}, D\} = \{\sqrt{}, \{\sqrt{}, D\}\}$$

Or

$$(15) \text{ Label } \{\sqrt{}, D\} = \{D, \{\sqrt{}, D\}\}$$

(15) seems *prima facie* to collapse at the semantic interface. However, is it possible to derive this from interface conditions without appealing to additional assumptions, for example, Distributed Morphology’s *Categorization Assumption* (see Embick & Noyer 2007 for developments of the claim that bare roots are not interpretable at the interfaces)? In our model, in the line of Panagiotidis 2009a, roots are way too semantically underspecified to undergo *referent assignment*, and thus an explicature (i.e., a full propositional form with satisfied referential variables, see Sperber & Wilson 1995; Wilson & Sperber 2004) cannot be built. On the other hand, if we let D be the ‘label’ at the semantic interpretative component, the whole structure is interpreted as a *specified entity*, because of the rigidity of D’s procedural instructions: conceptual content can be narrowed or widened¹¹, but procedural content cannot (Escandell Vidal & Leonetti 2011). ‘Ill-formations’, therefore, are *interface-determined*; the structure building operation has nothing to do with them (thus, labelling proposals which follow from a narrowly syntactic operation like Minimal Search – e.g., Chomsky 2015 – turn out to be problematic). Let us consider a more extreme case:

$$(16) \text{ Concatenate } (D, T) = \{D, T\} \text{ (T = Tense)}$$

¹¹For example, the extension of $\sqrt{\text{BANK}}$ can be widened to include ATMs as well as financial institutions on the one hand, and narrowed to specify a small-scale bank instead of the World Bank.

(17) Label $\{D, T\} = \{D, \{D, T\}\} / \{T, \{D, T\}\}$

Neither alternative yields a usable object, as it is to be expected. There is no way of building an explicature out of that structure, no matter how C-I tries to interpret it, assuming that every input conveys the assumption of its own Optimal Relevance: there is no substance to delimit, no maximal extension to restrict. The phase space for this system is not unrestricted (as would be the case with a root without a procedural category): there is no phase space at all (conceivably, the vector components of each category cancel each other out, since one conveys sortality, and the other anchors an event). *Optimal Relevance cannot possibly be achieved*, in other words. It is obvious as well that there is nothing wrong with $\{D, T\}$ in strictly formal terms, given *Concatenate- α* . Any restrictions are interface-imposed.

Let us make some further clarifications on our notion of *Type-Array* and *Token-Merge*:

(18) A *type* is a (possibly unary) set of points in vector space¹² (thus, a phrase marker is an orientable manifold in vector space, see Saddy 2016 for discussion).

(19) A *token* is an occurrence of a *type* within an active workspace W_x . There are no *a priori* limits to the times a *type* can be instantiated as a *token* but those required by Interface Conditions IC.

The number of tokens required is determined by interface conditions, so that the minimal number of tokens leading to a convergent (i.e., fully interpretable) object is used, provided that the notion of ‘convergent object’ does not arise from look ahead (the syntactic component looking at the legibility conditions of the interface systems, which would lead us to a ‘bad’ crash-proof system), but rather from the interfaces ‘peering into’ (i.e., ‘invading’) the syntactic workspace (something similar to the proposal in Boeckx 2007, although the consequences drawn there are

¹²We briefly address an interesting issue raised by Thomas Stroik (p.c.) here: how come, if language is indeed a physical system, as we claim, that there are types and tokens, or blind Merge, something that is not found in other physical systems or processes (say, for example, the formation of a complex molecule)? Two aspects of the question are to be considered: on the one hand, our main thesis is that all physical systems are identical ‘at a principled level of abstraction’, not at a substantive level. By ‘principled level’, as we have argued in previous works, we mean architectural issues, particularly as it comes to complexity. Therefore, we argue that any kind of ‘complexity’ can be studied formally as an interpretative epiphenomenon of the interaction between simpler units, until getting to the unit that integrates another but is itself not composed by any other. On the other hand, and relatedly, the formation of a complex molecule is indeed limited by the characteristics of the units that are manipulated, that is, chemistry is not crash-rife, and there are compounds that are either unstable or directly do not appear under natural conditions. That is not due to a property of the union (e.g., covalent, ionic, metallic), but of the elements involved. Contrarily to endoskeletal models (GB, Minimalism, HPSG, LFG), it is not the properties of lexical entries that determine the shape of the syntactic configuration, but semantic interface requirements.

completely different from ours) and *Analyzing* whether a syntactic object is ready to be transferred. Boeckx (2010b) points out, and we agree, that feature bundles cannot be driving syntax, since these bundles are structures and that structure is (must be) syntactic in very much the road taken by Distributed Morphology (Halle & Marantz 1993; Embick & Noyer 2007, and much related work), where it is considered that roots and ‘f-morphemes’ (functional elements like inflectional morphemes and categorizers, to use the old DM term) are combined syntactically, with the same constraints that apply to a phrasal syntactic representation. The elements the syntax manipulates should optimally be *atomic*, and Merge should be taken, in his view, as a free-triggered unbounded operation, *Merge- α* . The whole argumentation of Boeckx’s (2010b) aims at ‘defeating lexicocentrism’, that is, the presence of a pre-syntactic instance where fully-fledged lexical items are taken from. Contrarily to Chomsky (2005) and Hauser et al. (2002), the ‘great leap forward’ (that is, the qualitative evolutionary difference between humans and non-humans which would have made language available) in the present proposal would be *the interaction between an n-ary concatenative formal operation, freely available throughout the mind, and sound-meaning systems, but not the concatenative operation itself*. This interaction yields a dynamically frustrated complex system, for the requirements of sound and meaning systems are mutually orthogonal (Uriagereka 2012, 2014). This perspective might set the agenda for an alternative biolinguistic kind of inquiry: interesting dynamics arise when we *limit* the power of an algorithm or the probability amplitudes in a phase space, rather than enrich those.

We have so far sketched the framework that will lead our inquiry, whose main claim is that language is an object with no particular idiosyncrasy within the natural world: it is, as any other portion of the Universe, a physical system in the technical sense. We consider language to be a physical system, and we characterize it using non-linear dynamics: this means that, as we said above, interface representations are not a direct function of the input to the derivation. Some basic properties of these systems are listed below (based on Hasselblat & Katok 2003; Ladyman et al. 2012; Baranger 2004; Boccarda 2002):

- (20) a. Open to external influence
- b. Complex (i.e., contain subsystems)
- c. Dynamic (i.e., change over time)
- d. Emergence (i.e., the collective behaviour of the system is not a linear function of the behaviour of its individual components)
- e. Nesting / hierarchical organization
- f. Existence of feedback loops

We assume that the cognitive systems exhibit the same phenomena as physical properties in (20).

Natural languages, as also pointed out by Saddy (2016) seem to fulfill all of the intensional requirements in (20): they are *open* because interface constraints to apply to an output¹³ are not abstracted by a mind unless in contact with data in the phenomenological world (so-called Primary Linguistic Data), and it is also a methodological mistake to study language in the mind-brain completely isolated from other systems insofar as those systems impose conditions upon language design: we are referring to physical limitations on possible neural networks (see Uriagereka 1998; Cherniak 2009), in turn deeply related to optimization algorithms; they are *complex* because *syntax* (the *generative* component widely understood) interfaces with other two components of interpretative nature and natural language comprises all three (Faculty of Language in the Broad sense, in Hauser et al. 2002), in fact, we argue that all that there is to the specificity of language is the interface of these systems, in turn deeply related to the property of *emergence*; they are *dynamic* in two relevant ways: firstly (and from an ontogenetic point of view), because the number of constraints that apply to the output of the generative component changes over time during acquisition, as they are subsumed to other, more fundamental constraints until reaching the optimal scenario: a crash-proof system with only one constraint. A caveat is in order here: the more constraints we have, the more stable the system will be: after a certain number of constraints, a ‘threshold’, the change is suddenly perfectly ordered and predictable. If we have only few constraints, the result will be a system tending to infinite complexity instead of achieving internal balance after a certain period of time. In the end, we cannot fully dispense with constraints upon the output of the generative component: from our perspective, it is precisely the interplay between these constraints which are mutually incompatible (global for semantics, local for morphophonology) which define a dynamical frustration that shapes the formal and substantive properties of language as a system (including the presence of Zipfian distributions, since they arise from the tension between Forces of Diversification and Unification, see Zipf 1949), including its neurobiological substratum (Hansel et al. 1995; Papo et al. 2014). In Krivochen (in progress), we argue that the processing of language, because of its structural characteristics, is *not* the computation of a function (*contra* Gallistel & King 2010), which transform a finite input into a finite output in polynomial time by means of a *closed* process involving serial applications of rules or operations. We claim that natural language processing makes use of *open* processes, and *multiple* (though finite) inputs, impossible to calculate or foresee *a priori*, interacting with *outputs* to generate representations of the structure and meaning of a linguistic stimulus.

The scenario we have sketched so far, then, consists on a free generator, in

¹³Whenever we talk about *constraints*, we have in mind an OT-like architecture, and the definition of *constraint* that follows from that architecture.

the sense that it is not constrained by intra-syntactic filters, and a set of constraints over usable outputs. The application of an extremely local evaluation procedure gives meta-stable *balance* to the system, what is normally called ‘negative feedback’: after an external perturbation in the cognitive system, it balances itself, thus getting to one of several possible meta-stable states (a property of spin glass configurations), which we are interested in analyzing.

As an *independent* hypothesis from *nonlinearity*, we will now address the question whether language is a *chaotic* system. It is essential to bear in mind that, while both chaos theory and complexity theory study non-linear dynamics, they are not to be identified, nor do they use the same tools for analysis: in this particular case, we argue that the emergent behavior is the result of the non-linear – syntactic – interaction between a finite number of elements, optimally, only underspecified roots (possibly very few *per* language) and procedural elements (P, T, C, v, D), combined syntactically; taking into account the legibility conditions of the interpretative systems the computational system interfaces with (C-I and S-M; meaning and sound), the result of the syntactic computation is only *partly* deterministic (not random, but not entirely deterministic either, as we argue in Krivochen in progress, where we make a case against seeing linguistic derivations as functions in the traditional Church/Turing-relevant sense). Given this scenario, we will ask ‘is language a chaotic system?’

2 Language as a chaotic system?

We will propose in this paper is that natural language *is* indeed a non-linear chaotic system in the technical sense of the expression (which, in turn, has far-reaching consequences for a theory of the mind in which this system is implemented, as we will see below). Our discussion, even though related to that in Uriagereka (1998, 2000, 2002a) and Saddy & Uriagereka (2004), as has been pointed out to us, has somewhat different foundations and consequences: we are not so much concerned with aspects of the narrow syntax, but set our focus on more general properties of physical systems, which happen to surface in language.

Since Uriagereka’s aforementioned articles have been pointed to us to be the first application of aspects of chaos theory and entropy to language within a generative framework, they deserve some comments as our proposal differs from his.

Uriagereka assumes the following tenets (2000: 865):

- (21) *Within local derivational horizons, derivations take those steps which maximize further convergent derivational options.*
- (22) *All else being equal, derivations involving fewest steps outrank their alternatives.*

We find some problems with these assumptions and their theoretical consequences, acknowledging that the framework was devised with the tools in hand at that point. To begin with, the concept of *local derivational horizons* is left undefined. To our knowledge, two options arise in the current state of the art in linguistics:

- (i) Chomskyan endocentric phases, that is, local derivational units headed by a certain kind of linguistic element and defined *a priori* (Chomsky 1998, 2008, 2015);
- (ii) each application of a structure-building algorithm determines a local derivational unit (Epstein & Seely 2002).

In any case, this is problematic, since not only the identity of the local domains is unknown, but also where they are defined (at the ‘narrow syntax’? At the interfaces?). In our present proposal, derivational cycles can be defined from two perspectives:

- (i) *Computationally*, as maximal-minimal finite-state compatible units for purposes of structure linearization (Uriagereka 2002b, 2012).
- (ii) *Informationally*, as sub-domains of locally decreasing entropy in a discontinuous oscillatory function.

A feature that is common to Uriagereka’s (2002b, 2012) approach and our own is that cycles are *minimally interpretable, maximally generable units*. This, which might seem a mere aphorism, is a way of summarizing what a cycle is and how (and why) it arises, given a system with generative-interpretative dynamics. Cycles are minimally interpretable units because nothing smaller than a cycle can be assigned an interpretation in terms of sortality, eventivity, or relations between them. And, they are maximally generable, because once a critical value for the manifold being generated is reached, a dimensional attractor fades due to the inherent limitations of the physical system we’re dealing with, and no further structure can be built up in that derivational current (Saddy 2016; Krivochen in progress). Given oscillatory dynamics subjected to the CMT of the kind we have presented here, a dynamically frustrated physical system with finite resources cannot hold on to a transferrable object. Unlike Chomskyan cycles (barriers, phases ...), which arise from theory-internal stipulations over both structure building and labeling (such that only certain nonterminal nodes configure phases), cycles in the two senses above are much more fundamental – since they arise from computational and physical properties of the system, respectively – and dynamical – insofar as we cannot say, *a priori*, whether a syntactic object is a cycle or not, as it



depends on the legibility conditions of the systems that are supposed to read that object.

Uriagereka's tenet given in (22) is particularly conflictive from a contemporary theoretical perspective as it overtly builds on Chomsky's (1995) seminal proposal about reference sets to compare derivations (see also Müller 2011 for discussion about transderivational constraints). Uriagereka, like Chomsky, assumes the existence of a set $R = \{D_1, D_2 \dots D_n\}$, where D are derivations, each built by taking 'decisions' about structure-building that, in turn, allow different possibilities always departing from the same set of lexical types. Derivations can be compared, in this system, with respect to the number of possibilities each derivational decision (e.g., Merge / Move) allows. According to Uriagereka, if an operation Σ allows an n number of continuations and an operation Σ' allows a number m of continuations, if $m < n$, then Σ increases derivational entropy and Σ' is to be preferred. Problems with this approach arise from the very beginning of the argumentation. To begin with, the system is syntactically based and centered, with the interpretative sound / meaning interfaces playing no role at all in determining cost / 'derivational entropy', in consonance with a strongly constructivist system which needs syntax to be partly interpretative in order to read previous derivational steps and, what is more, allow massive *look ahead* so that all possibilities have to be *fully derived* to be compared. In our opinion, this does not imply any optimization of the mechanism but, on the contrary, overloads the computation as there is no *a priori* way of limiting, say, m : derivations could be completely random, which undermines both the computational and implementational value of the theory. In theory, m could be *infinite*, as there is no element in the Lexical Array that necessarily conveys an instruction to end the derivation (i.e., there is no 'halting rule'), and no halting algorithm has been proposed within generativism beyond stipulations about phase heads (v^* , C in Chomsky's approach, see also Gallego 2010) and transfer of phase head complements.

Returning to the ground problem of *reference sets*, a *crash-proof* model like that argued for in Putnam (2010); Stroik & Putnam (2013), and Krivochen & Kosta (2013) has a reference set that is for all intents and purposes *unary*: in the terms we have introduced here, the system generates (i.e., derives) symbolic representations defined in vector spaces, the relation between sub-phrase markers as orientable manifolds defines an optimal derivation defined by the core vector components of each manifold involved, with peripheral vector components largely cancelling each other out (Feynman et al. 2005). In our proposal, like in Survive Minimalism (Putnam 2007; Stroik 2009; Stroik & Putnam 2013), there is no point in talking about competing derivations or degrees of optimality in the early Minimalist sense (see Müller 2011): if a symbolic representation is coopted by the interfaces, it is *optimal as it can be* given certain external conditions, something that has a very



close parallel in biological and physical systems, and optimization problems (finding the *best* from all *feasible* solutions, see Boyd & Vandenberghe 2004): it is *not perfection*, but *optimality*, that counts. In this respect, it is interesting to explore the relation between the non-function approach to Array-(PF, LF) pairs and the sensitive to initial conditions that characterize chaotic systems (see, e.g., Smith 1998): consider that the *n*th step of a derivation is characterized by the values of *x* variables (call them lexical elements, including both roots and procedural elements), those values being, e.g., category, case, etc. (what is usually referred to as ‘features’ of an element, even outside the feature-valuation approach). What we say is that the derivational stage *n + 1* is *neither* a linear function of *n* (thus, a linguistic derivation cannot be characterized by means of recurrence relations) *nor* a function of the Numeration, since unless additional stipulations and restrictions enter the game, it is not possible to have a fully deterministic path from Num to (PF, LF): for instance, the same Num can give rise to more than one convergent derivation:

- (23) Array: {who, know, v, T, C, believe, Bob}
 - a. Who knows who believes Bob?
 - b. Who knows who Bob believes?
 - c. Who believes who Bob knows?
 - d. Who believes who knows Bob?

...plus a huge number of non-convergent derivations, gibberish, and marginal sentences (e.g., *Bob believes who knows who?*, only acceptable as an echo question). Having a single starting point, there are multiple possible endings, not all of which are (PF, LF) convergent, and, most importantly, mainstream Minimalism has, to the best of our knowledge, no way to make a derivation fully deterministic, nor can all properties of a sentence / utterance be present in the Numeration: there is place for emergence and chaos in linguistic derivations. A theory of real-time development of linguistic derivations should, we argue, include the analogue of a system of dynamical equations, which capture the development of the values of the variables in time (e.g., specify the mechanisms via which category and case are ‘assigned’ to a lexical item at the interfaces), while impoverishing the narrow syntax itself to a mere concatenative engine. This move would help extending the limits of syntactic theorizing beyond natural language into other areas of interest, like music and mathematics. The property of nesting, which is, as we have seen, one of the characteristics of complex systems, is also a signature of so-called ‘chaos in space’ (Baranger 2004: 4-5), i.e., *fractal structures*. Memory limitations aside, the possibilities of structure nesting in natural language are infinitely many, insofar as even assuming a limited alphabet of symbols to work with (at any given

point, the lexicon of a natural language is finitely enumerable, even if new elements can in principle be added to it): as has been noticed by Uriagereka (1998), among others, the structure of natural language is that of a fractal, with different levels of complexity (a property that has been recognized since pre-structuralism, as ‘double articulation’). The question here is whether we are dealing with a self-similar fractal or not: that is, do successive ‘enlargements’ of a syntactic structure reproduce always the same object? In this respect, it is possible that we are in presence of self-similar structure to different degrees (a ‘quasi-fractal’, see Mandelbrot 1983), given a mixed computational system of the kind explored in Krivochen (2015a), and Bravo et al. (2015). Needless to say, a deeper research on the nature of the external conditions imposed over syntactic structure by the interface systems is crucial here, insofar as different system requirements might result in different kind of fractal structures (thus, the kind of structural complexity to be found in natural language is not the same as that to be found in mathematical thought): this is, we think, both a theoretical and an empirical challenge, for the regularities in the structure of the system could be based (as we argue) on deeper physical principles of matter organization, in turn affecting the possibilities for neural network connections and their computational emergent properties. The tension (‘frustration’) between n -dimensional semantic structures, and Markovian phonological strings (Uriagereka 2012; Krivochen 2015a, 2015c; Idsardi & Raimy in press) configures an oscillatory ‘stretching and folding’ pattern characteristic of non-linear chaotic dynamics (see, e.g., Baranger 2004: 8; Saddy & Uriagereka 2004; Saddy 2016), in which chunks of structure are *dynamically flattened* to be externalized (Lasnik and Uriagereka 2012: 21), a dynamics that can be captured by an oscillatory function with a period determined by semantic conditions (see Uriagereka 2002b for a model of Multiple Spell-Out based on phonological requirements, and Krivochen 2015a, 2015c, in progress for a complementary model based on semantic domains rather than phonologically linearizable structural chunks).

A defining property of complex chaotic systems is *entropy*, given by their dynamic character: the development of a system (ontogenetically, the mental grammar; but we are concerned with a local, derivational point of view) leads to an inevitable increase of its entropy, in the system we are interested in, from the conceptual structure CS shaping a speaker’s intention to the ultimate materialization / externalization of that intention as a linguistic stimulus. Let us provide a first naïve approximation of the concept of *entropy*:

(24) *Entropy is a measure for disorder.*

Whereas (24) is uncontroversial in modern physics, it is also of little informative value: we still do not have a definition of ‘disorder’ (or even of ‘order’, for that matter), and we lack also a circumscribed field for the definition to be scientifically useful. Let us enrich the definition, then:

- (25) *Entropy is the measure of the free energy that is released and cannot be used in a changing physical system, provided that the system is free and in normal temperature.*

Within information theory – and on a background of Markov models – Shannon (1948: Appendix 2), based on Boltzmann’s H theorem, formally defined the entropy H of a discrete random variable x (with possible values $\{x_1, \dots, x_n\}$) and probability mass function p_i (i.e., the probability that the random variable is in a point i of its phase space) as follows:¹⁴

$$(26) \quad H(x) = \sum_{i=1}^n p_i \log p_i$$

This is a definition that is both general enough to be applied to different physical systems (as long as there is information involved) and specific enough so that it is scientifically interesting. We will work under the hypothesis that the aforementioned informal definition (25) is valid for *every physical system*, and, *since biological systems are particular instantiations of physical systems, they are also analyzable from an entropic perspective*.¹⁵ We claim that biological systems are particular instantiations of physical possibilities since, for example, neurological optimization (‘non-genomic nativism’, Cherniak 2009) is restricted to the possibilities licensed by the physical organization of matter and the configurations it can adopt with basis on the physical laws that rule the phenomenological world. There is already a considerable amount of work within Biological Physics regarding the concept of entropy in biological systems (e.g., Udgaonkar 2001), but we will try to give another turn to the screw by analyzing entropy in linguistic derivational procedures, taking ‘derivation’ in a wide sense to be the successive application of the generative operation *concatenate- α* , limiting ourselves to natural language for methodological (and space) reasons. Let us formulate the hypothesis we will work with in the course of the paper:

- (27) *Language is a chaotic complex nonlinear system.*

This first formulation will do for the time being. Let us see how the system would work.

¹⁴It is to be noticed that Shannon provides formulae for more than one variable, $H(x, y)$. In these cases, x is the information source and y is the destination. We will not go deeper into this here, but refer the reader to Shannon (1948) for details.

¹⁵That is, the characteristics a biological system can adopt are constrained by the physical principles ruling the environment in which that biological system is to develop. Udgaonkar (2001) is a brief and good introduction to some problems of physical biology. Uriagereka (1998: ch. 1) also expresses some of these ideas, but in a more introductory way for those not familiar with the assumptions of generative biolinguistics.

3 Entropy in linguistic derivations

A *derivation* will be taken here to be a successive application of an algorithm, with no necessity that the output of that computation is a function of the input (see Goldin & Wegner 2005, 2007; Krivochen in progress, for discussion of the consequences this corollary has for the applicability of the Church/Turing thesis). In more concrete terms, a *derivation* as seen from a mechanistic perspective is the finite set S containing the objects created by the successive application of the *concatenation* function we defined in (3), and which we repeat here:

- (28) *Concatenation* defines a *chain* of coordinates in n -dimensional metric workspaces W of the form $(x, y, z \dots n) \subset W_X, \dots (x, y, z \dots n) \subset W_Y, \dots (x, y, z \dots n) \subset W_n$.

To provide the derivation of an object means to provide an account of the history of how that object came to be, given an input and a computational procedure, plus the conditions the output is to meet (in our case, sound-meaning interface conditions). Nothing impedes that concatenation as a purely formal operation applies indefinitely, which gives us infinite use of finite media, in Humboldt's terms. However, when an object derived via concatenation has to be interpreted by an interface system (which is not the case in arithmetic, for example), the derivation is driven by a principle we have dubbed Dynamic Full Interpretation, a step-by-step application of Brody's (1995) *Radical Interpretation*:

- (29) **Dynamic (Full) Interpretation:** Any derivational step is justified only insofar as it increases the information and/or it generates an interpretable object.

Let us see what this thesis means for the entropy of derivations. If entropy reaches its maximum level when all elements are equally likely to enter the active workspace, any restriction regarding the set of elements that can be merged in the following derivational step (see the definition of 'Soft crash' in Putnam 2010: 6), which includes this notion of *local derivational unit*) will make entropy decrease, since the system is asymptotically tending towards an *ordered state*. Let us see an example. A linguistic derivation starts when some items are drawn from the Lexicon, the full set of conceptual and procedural elements represented in a speaker's mind. Those items are *types*, and each of their instantiations in a derivation is a *token*. The first step in a derivation, then, is to form a *type-array*, containing types of all the elements that will be used to generate a sentence as faithful to the CS as possible. From a PoL perspective, each of these elements is a perturbation of the ground state of cognitive dynamics, which is argued in Saddy (2016), Saddy & Krivochen (2016), and Krivochen (in progress) to be an unrestricted, high-dimensional *ultrametric* space. Let us work with the following array:

$$(30) \text{ Type-Array} = \{D, P, \sqrt{\alpha}, \sqrt{\beta}\}$$

These perturbations cannot be related on an ultrametric space because of the topological properties of this space; usable outputs require the ultrametricity of the space to be disrupted. In order for the information held by this space to be put to use we need to translate critical properties of the complex manifolds into a lower dimensional and metric instantiation of this space. Saddy (2016) argues that

This initial space, uncountable in nature (i.e., qua set of points, its cardinality is greater than \aleph), gives us a very rich representation in terms of the organization of the sensoria, but since this space is unrestricted we cannot do anything with it: there is nothing that is not in there. In slightly more technical terms, there is no point in this unrestricted space that does not have a probability amplitude associated with it. This space has no hard conditions imposed upon it, which allows for very rich and high-dimensional representations – multidimensional complex manifolds – but also leaves us with no way to limit the phase space: there is no grammar that can generate such a space

That is we need a dynamic process that will project or translate information from the ultrametric space into a metric space that is mediated by strict topological and computational conditions. Given an unrestricted high-dimensional ultrametric space of the kind we mentioned above, there will be an infinite number of *n-dimensional* manifolds which represent dimensional surfaces and, provided the topology of the space is disrupted – as from external input – they may intersect and reach a critical dimensionality value, thus triggering the CMT.

Given the type-array in (30) and no further stipulations, any of these elements could enter the workspace *W* first, so the *first step in a derivation has the maximum entropy*. In Krivochen (in progress) we present a way to model a derivation in terms of paths within a vector space which minimize the associated Lagrangian, here we will present the problem in slightly more classical (generative) terms. Let us assume $\sqrt{\alpha}$ enters the derivation first, as would be expected in a 2-D tree-like representation¹⁶, but completely irrelevant in an *n-dimensional* phase space studied from a field-theoretical perspective. The introduction of an element in the phase space makes the entropy associated with that space decrease, since not all elements in the initial Array may enter the derivational space and undergo concatenation with $\sqrt{\alpha}$ *while respecting DFI*. Depending on the conceptual structure that this syntactic derivation is to instantiate, the possibilities for *P* vary, but

¹⁶See Di Sciullo & Isac (2008), Stroik & Putnam (2013), De Belder & van Craenenbroeck (2011) for examples of different criteria to determine which element enters the syntactic workspace first. All of the references assume some kind of featural defectivity in the element, an assumption we do not share.



assume for the sake of the argument that we want to build a fully-fledged prepositional phrase, containing two Determiner Phrases = {D, $\sqrt{\}$ } constructions related in a central / terminal coincidence manner by means of P (preposition). If this is the case, the possibilities for P = 0, since the conceptual structure requires sortal entities; and the possibilities for $\sqrt{\beta}$ = 0 as well since roots are semantically underspecified and cannot be interpreted by C-I unless within a larger structure containing a procedural element that indicates the semantic interface how to manipulate the semantic content conveyed by the root. This means that we have an optimal situation: only one type-candidate satisfies DFI. The following derivational step would be:

(31) *Concatenate* ($\sqrt{\alpha}$, D) = { $\sqrt{\alpha}$, D}

Notice that should we have introduced D first, the results would have been the same since generation does not care about order or ‘side’ of the tree in which symbols are inserted, unless one is willing to concede that the computational system is *both* generative and interpretative. If such a comment applies to traditional Kaynean trees (of the kind generated by the development of an L-grammar, as shown in detail in Uriagereka 2012), it sure does to symbolic structures beyond Turing computability, as it might be the case of conceptual structures and, within language, probably multidominance theories of displacement. For the purposes of future computations, this structure will be interpreted as a D(eterminer), this is what we call the *label* of the construction (otherwise, no information would have been *added* strictly speaking since the whole structure would still be interpreted as an underspecified root). Having thus a {D, $\sqrt{\}$ } structure, the situation changes: even though merging another D token is not possible if DFI is to be satisfied, we still have, *in principle*, three logically possible candidates, all equally likely to be instantiated as tokens in the derivation. This increases the entropy again and, if we had no way to determine what must come next, the derivation could not continue. In previous sections we have argued in favor of a pre-linguistic conceptual structure very much on the line of Jackendoff (1987, 1990, 2002), Culicover & Jackendoff 2005, Uriagereka (2012, 2014), that is instantiated linguistically via the *Conservation Principle*¹⁷ (hereafter, ConsP):

¹⁷Lasnik et al. (2005) borrow Conservation Principle from physics, and state the following law:

- (i) **1st Conservation Law:** *All information in a syntactic derivation comes from the lexicon and interpretable lexical information cannot be destroyed.*

The problem we find with this law is that it makes use of lexical information taken from a pre-syntactic and monolithic lexicon, which is the norm in orthodox Minimalism, but with which we will not work. Moreover, this formulation is limited to linguistic structures, whereas ours is wider in scope.

- (32) **Conservation Principle:** *information cannot be eliminated in the course of a derivation, but it must be instantiated in the relevant system in such a way that it can be read and it is fully preserved.*

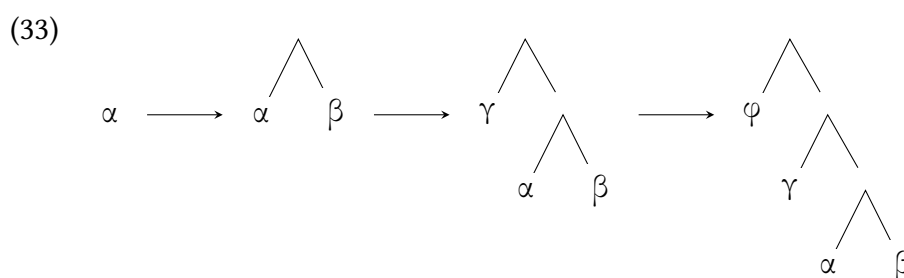
If this pre-linguistic conceptual structure was *locative*, then a $\{\{D\}, \{P, \{D\}\}\}$ construction is to be built in W . However, we can also derive the better candidate from purely post-syntactic interface conditions, which is another (perhaps less controversial) option. Assuming Brody's (1995) *Radical Interpretation Thesis*, every element must receive an interpretation (strengthening the hypothesis, not in *some* syntactic location, but in *every* syntactic location in which it appears at LF) and, we will add, if the element conveys procedural instructions as to how to relate conceptual elements, this information must be represented syntactically, so as not to lose information (and therefore incur in a violation of the ConsP). We have $\{P\}$ in the Type-Array, which conveys *locative* procedural instructions, in terms of *central-terminal coincidence relation* between a Figure and a Ground; see Talmy (1983, 2000); Hale & Keyser (1993, 2002); Mateu Fontanals (2002). If we introduce any other token into the derivation the procedural information conveyed by $\{P\}$ will be lost since there will be no available instruction to relate two structures, typically of the type $\{D, \sqrt{\}$, at the semantic interface: any other element, even though procedural, would not be relational, and the information conveyed by the CS would be modified. So, the ConsP favors the introduction of a $\{P\}$ token in W . This way, the derivation continues with variable levels of entropy, basically responding to the following dynamics: in a Cartesian system, for x = derivational steps (i.e., applications of *concatenation*), and entropy represented in the y axis such that 0 = complete order and 1 = complete disorder, the oscillatory dynamics of the system *tends* asymptotically towards 1 and 0 alternatively as structure is built and the relevant phase space for interpreting that structure is progressively restricted (see Saddy 2016; Uriagereka 2011 for related views). Once a syntactic object is recognized by the relevant interface as complete (i.e., fully interpretable), what we will call a 'cycle', global entropy goes up again.

We see that derivational entropy is reduced at the cost of energy required to disrupt the ultrametricity of the ground state of cognitive dynamics and make manifolds intersect, but this is so only *within a local domain*, where *locality* is defined according to interface conditions and by no means as an *a priori* concept. A local domain is a minimal interpretative unit which is also maximally generable insofar as its growth triggers the CMT at a critical dimensionality value (as proposed in Saddy 2016 and Krivochen in progress). This oscillatory system has the advantage of defining domains dynamically and in real time and without resorting to narrowly syntactic stipulations; because of this, the CMT-based definition of cycle is also system-neutral. Assume that new domains so defined above are created starting from separate (sequential or parallel, since there is no *a priori* limit to

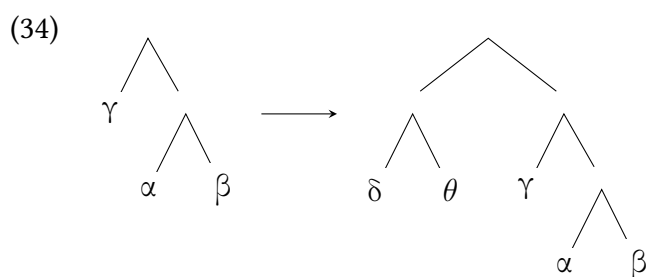
the number of W that can be activated simultaneously) Type-Arrays: *each time a new (interface-delimited) domain is built, the entropy value is maximum again since all elements are equally probable*. This means we have to consider entropy dynamics at two scales: global and local. *Global* entropy would be the specific value for a domain d , whereas *local* entropy is the value of entropy at every point in the derivation (a rough equivalent to *harmonic serialism* in OT-like models of syntax, see, e.g., McCarthy 2010). This interplay of scales defines a dynamical frustration of a specific kind, a *scale frustration* (Binder 2008).

3.1 Some local properties of derivations

In our derivational model we differentiate, following Uriagereka (1999, 2002a), *monotonic* and *non-monotonic* structures, since they have different computational properties and thus impact differently on the size of cycles. Monotonic concatenation refers to the application of the generative function in a successive way involving always a terminal node:



We see that the third step involves the inclusion of a terminal (i.e., non-branching node) γ which is merged with a non-terminal, $\{\alpha, \beta\}$, and the same happens in the fourth step, where φ is merged to a non-terminal $\{\gamma, \{\alpha, \beta\}\}$. The mechanism represented in (33) exemplifies this kind of application of the generative algorithm, which is referred to as *monotonic*. *Non-monotonic* merge involves *non-terminals*, as in (34):



In (34) we see that the second step involves the merger of two non-terminals, giving rise to a complex object. Each non-terminal, in turn, has been assembled monotonically in a separate workspace, and the unification takes place in a third



workspace (in our proposal) or at the interfaces (in Uriagereka’s). We will maintain the ‘multiple parallel workspaces’ mechanic here, since not only the syntactic component can be simplified with it, but also the semantic component C-I appears to be able to handle several representations in parallel when deriving pragmatic inferences (Wilson & Sperber 2003). The availability of parallel syntactic workspaces has also been justified within the PoL approach from a mathematical point of view. In our formal definition of *concatenation* we introduced the concept of *n*-dimensional workspace. As a reminder, *W* is an *n*-dimensional *generative* workspace. Considering two distinct workspaces W_X and W_Y , we find one of three formal scenarios:

- (35) a. $W_X \equiv W_Y \text{ iff } \forall(x) | x \in W_X, x \in W_Y \wedge \nexists(x), x \in W_X, x \notin W_Y$
- b. $W_X \neq W_Y \text{ iff } \nexists(x) | x \in W_X, x \in W_Y$
- c. $W_X \cong W_Y \text{ iff } \exists(x) | x \in W_X, x \in W_Y$

This allows us to define the relations of *identity*, *difference* and *similarity* between non-terminal syntactic objects in set-theoretical terms (see also Krivochen’s 2015c: 30 *Token Collapse* operation). *Identity* holds between W_X and W_Y if and only if every element of W_X is also an element of W_Y and vice versa (formalized in (35a)). If this condition obtains, there is also *logical equivalence* between W_X and W_Y in all relevant contexts. *Logical equivalence* is entailed by *identity*, as it is to be expected. *Difference* holds if and only if there is no common element between W_X and W_Y , which means that they are not set-theoretically related (formalized in (35b)). *Similarity* is a relation in which there are common elements *at least* -but not necessarily *only*- one, between W_X and W_Y (formalized in (35c)).

In a restrictivist derivational model, the generation of ‘momentarily’ illegible structures can be tolerated if we accept the definition of *Soft Crash* in Putnam (2010: 6), as we assume there is a ‘local derivational unit’ to repair any violation:

- (36) **Soft Crash:** *If a syntactic object α cannot be interpreted at an IL in any and all of its features, α is neither useable nor legible at IL, iff α cannot be combined with another local derivational unit that repairs the violation(s) of α .*

The concept of ‘local derivational unit’ is here restricted to the element introduced in the very next derivational step, provided that DFI holds.

Regarding how categories emerge in the derivational dynamics we have proposed, we will assume that the local relation between the procedural element D collapses the root’s categorial potential (think of it as the linear combination of all possible outcomes) to Noun (i.e., sortal entity) without excluding [cause] (e.g., in derived nominals from causative construals), T collapses it to Verb (extending-into-time perspective), and P collapses it into Adjective (see Mateu Fontanals 2002), assuming the classical categorial matrices minus P, which operates over roots rather

than being operated on (cf. Piattelli-Palmarini & Vitiello 2015, Piattelli-Palmarini & Vitiello 2016; for us, only roots are eigenvectors, D, T, and P are Hermitian operators¹⁸). The reference restriction process we mentioned above also applies to categories, as they are not primitives of syntactic theory but arise in the interaction between the syntactic component and the interpretative interfaces. Common sense may dictate that the primitive *cause* appears only in verbal (i.e., eventive) structures, but there is an aspect of the C-I/syntax interface that we have mentioned elsewhere and is essential to this: *this interface (and possibly, all other interfaces) is not transparent* (i.e., there is no exact correlation between a Relational Semantic Structure and its syntactic realization, as well as there is no exact isomorphism between the representations manipulated by two modules, even respecting ConsP). Consider the following example (lexico-semantic structures follow Jackendoff 2002; Culicover & Jackendoff 2005, and Mateu Fontanals 2002):

- (37) a. They destroyed the city = [T_{PastPerf} [EA [cause [event GO [location CITY [TO [√DESTRUCTION]]]]]]]]]
 b. The destroyer of the city = [D_{DefSg} [EA [cause [event GO [location CITY [TO [√DESTRUCTION]]]]]]]]]

The correlations we have established follow from strict distributional properties, and only distributionally specified enough elements are capable of generating a categorial interpretation (that is why we excluded *cause*). A lexical item, in our proposal, is nothing more than the interface-read local relation between a procedural node, distributionally specified enough and a root, allowing an *n* number of non-intervenient nodes in between, as in (12) above.

The linguistic instantiation of a particular CS can change the *format* in which information is presented, but it cannot delete information, nor can it tamper with the content. However, we cannot rule out *addition* of information, coming at least from two sources:

- (i) non-linguistic cognitive systems (in tune with interactive computation:

¹⁸Using Dirac notation, if we model roots as *ket* vectors, then the emergence of categories can be formalized as follows:

$$(i) H |\psi\rangle = \lambda |\psi\rangle$$

H is an *n*-by-*m* matrix, where all possible measure results are represented (what is referred to as the ‘observables’). $|\psi\rangle$ is called a ‘ket vector’ (or, in this configuration, an ‘eigenvector’), and is a vector of *n* components ordered in a row, with each component being a dimension along which measurement takes place. λ is an ‘eigenvalue’, which represents the result of measurement, typically 1 or 0. In the present proposal, the eigenvalue determines the categorial interpretation: for that, we need at least three possible eigenvalues, say 1 (nominal), -1 (verbal), and 0 (the linear superposition of those). See also Uriagereka et al. (2016).

sensory information coming from the immediate situational context, interferences from the Long Term Memory, etc.);

- (ii) redundancy countermeasures given the system's unavoidable entropy (including inflectional morphology).

That is, we acknowledge the fact that, if the *language-as-a-physical system* thesis is taken seriously and non-metaphorically, as we do, then derivations, unfolding in real time, are unavoidably entropic. The role of whichever processes we assume take place *between* CS and LF is to *minimize* entropy, in other words, operations must be justified in *faithfulness* terms (in the sense the word has in Optimality Theory). This is the global perspective, based on Conservation. At the level of the cycles, each time a cycle is completed the entropy of the derivation goes up again, but not to the initial value, for there is a trace of previous cycles in the workspace. Our model cannot pretend that the cognitive system just ignores previous cycles, because this would entail that there can be no global effects (i.e., cross-cycle dependencies). Entropy decreases at each derivational step at the cost of energy: the phase space for interpretation is restricted each time a phrase marker is extended. However, when a cycle is completed and coopted by the relevant IL, entropy levels go back up, to a level slightly below the maximal value for the immediately previous cycle. This yields a globally entropy minimizing system (Saddy 2016; Krivochen in progress; Uriagereka (2011)).

A derivation as a whole is thus a dynamical process, in which mutually orthogonal local and global (morpho-phonological and semantic) tendencies struggle, giving rise to frustrated cycles of decreasing entropy. This is characterized by an overall sinusoidal entropy pattern, with each period corresponding to a derivational cycle (an idea developed in Krivochen in progress). Of course, this struggle is not perceptible at certain scales (e.g., if we analyze individual constituents or if we adopt a strictly representational perspective), but is to be found when looking at the process as a whole.

The idea that there might be 'something left' from previous cycles (a property that arises naturally in L-grammars in the form of 'residues' and 'extensions', as pointed out in Saddy 2016 and Saddy & Krivochen 2016) can also benefit from Feigenson's (2011) considerations about the ways in which the working memory represents objects (a theory we have already introduced). If it is possible to 'atomize' a chunk of structure for the purposes of further computations (in formal terms, turning an object $|\alpha|$ of arbitrary complexity into a terminal $\#\alpha\#$, with a concomitant loss of dimensionality through squeezing via the center manifold), that would be akin to taking a number of individual objects and conceptualizing them as a *set* (examples of such an operation are memorizing telephone numbers), which

[...] *manages to evade the three-item limit of WM while still preserving access to individual representations of the sets' contents. This appears to rely on the hierarchical reorganization of items within memory* (Feigenson 2011: 16).

Even if the 'three item' limit could be challenged empirically, the overall proposal seems plausible to us, and provides an interesting platform for studies of derivational cycles in connection to memory issues. For the time being, we will simply say that, if the human mind is capable of performing operations of the kind Feigenson proposes with representations (which in turn requires representations to have a certain flexibility in their format, something we have argued for in Krivochen 2015a and Krivochen & Schmerling 2015), additional stipulations would be required to support a claim that these operations do *not* apply to linguistic representations because of some putative specific property. If the generative (structure-building) engine is not specific to a faculty (*contra* Hauser et al. 2002 and much subsequent work within 'biolinguistics') the computational tools required to perform the structural operations over objects Feigenson puts forth are thus available all throughout the mind, a perspective we consider highly desirable.

In the scenario we have sketched, it is very unlikely that entropy keeps going back to maximal entropy 'as combinations proceed' cycle after cycle (note that we are making no claim with respect to the properties of the combinatory engine in terms of the elements it manipulates or how it does that, see Krivochen 2015a, 2015c for details of our take on the issue): that, we would expect from a 'dumb system' with no operative memory, despite having a possibly infinite tape (an automaton, in the sense we have reviewed). A system with a finite, but nonzero, operative memory (a.k.a. working memory / episodic buffer / RAM ...) would not just wipe out each and every trace of previous cycles if derivations are indeed something close to 'incremental'. We will expand on this globally cumulative dynamics in the following section.

4 Derivational crumbs in a cycle-based system

Theoretical / conceptual issues are related to the desideratum that derivations be somehow globally 'cumulative' in a non-trivial sense. Uriagereka (2011) has proposed the following principle relating structure generation and entropy:

(38) **Reduction of Entropy at Merge (REM):** *Two fields Ψ_α and Ψ_β combine via Merge if and only if $S\Psi_\alpha, \Psi_\beta < [S\Psi_\alpha + S\Psi_\beta]$*

Notice that the condition (which relates the entropy S corresponding to two field perturbations that interfere via Merge) includes a biconditional, meaning there is no instance of Merge that does not reduce entropy. While this is desirable (and, in fact, can be seen as the interface-defined side of the Extension Con-

dition + Full Interpretation Principle coin that shapes derivations in Minimalism – at least in theory), it sure has to be relativized taking into account interface requirements at different derivational points, and also contemplating the possibility that $S\{\Psi_\alpha, \Psi_\beta\} = [S\{\Psi_\alpha\} + S\{\Psi_\beta\}]$ at least for a ‘derivational turn’ (see, e.g., Putnam 2010: 6 on the concepts of ‘hard’ and ‘soft’ crash¹⁹). For the time being, we assume that the REM condition applies by cycles, in consonance with the MSO model first developed by Uriagereka.

We would like to maintain the essential insight of Uriagereka’s (2011) REM, but at a global derivational level (i.e., not at the application of every structure building operation, as we have argued above), we want to restrict it to within cycles. Uriagereka (2011: 23) claims that ‘*the entire point of linguistic combination is to reduce uncertainty*’, with which we agree. However, the immediately following strengthening of the hypothesis is what we question:

As combinations proceed, we gain certainty; the ideal (final) combination between fields should reduce uncertainty to zero. We may see this procedure as another way of achieving representational stability, and we may take the REM to be the way the language faculty has to ensure that only objects whose certainty has either being achieved or stipulated [...] can be handled by the interpretive components. (Uriagereka (2011): 23)

Certainty always comes at a cost, and we follow the physical insight of Heisenberg’s *Uncertainty Principle*²⁰ in that absolute certainty is not attainable (as it would cancel out the whole term, if Δx or Δp were equal to 0). While we are fully aware that the Uncertainty Principle was formulated having in mind position and momentum of a particle, a wider interpretation (not unfaithful to the original formulation) acknowledges that, as we said above, certainty always comes at a cost. Representational stability, as Uriagereka points out, is desirable; however, we do not see it as something to be ‘achieved’ (in any teleological sense), but only as part of the dynamics of the system, which oscillates between stability and instability. Both are essential to the non-linear dynamics of the system, and derive problematic aspects of the interaction between systems with different computational and topological properties (a clear case, which we will analyze below, is linearization

¹⁹Michael T. Putnam (p. c.) comments in this respect that ‘*My feeling now is that the only “hard crash” that essentially must be avoided is the one at the end of the derivation as a well-formedness constraint as such.*’ Interestingly, this implies that measures of optimization *during* the derivational process are ‘soft’, thus allowing the system to hold a momentarily non-convergent piece of structure in the episodic buffer to see if a local operation can introduce an element to solve the ill-formedness of that piece of structure.

²⁰As a reminder, the Uncertainty Principle is enunciated formally as follows: $\Delta x \Delta p \approx \frac{\hbar}{2}$ (to be read: uncertainty in position times uncertainty in momentum is of the order of Planck’s reduced constant $\frac{\hbar}{2\pi}$ over 2).

of structure, which implies dimensionality reduction as well as drastic changes in the morphology – in the non-linguistic sense – of the relevant object).

A somewhat disconnected argument in favor of taking *maximal disorder* as an asymptote rather than as a true state of the system comes from a perspective related to brain dynamics. Assume we have a statistical model that assigns mental states an activation probability between 0 and 1 over discrete time spans (a simplified hypothetical scenario built by Spivey 2007: 43ff.). Even if we were assuming that mental processes are digital, and computation was function-based along the lines of Gallistel and King (2010) (none of which claims applies in the context of the present work), the problem still remains of why the system reacts and proceeds under sub-optimal conditions (i.e., with probabilities of less than 1). In this respect, Spivey (2007: 47) claims that

Under realistic circumstances, changing perceptual input and continuous motor output would generally prevent the system from ever achieving 1.0 activation [i.e., full certainty] for any mental state.

The reason is that response thresholds (cognitive, motor, etc.) are always lower than 1.0, that is, we make choices on the basis of incomplete information and with less than absolute certainty (a point that we made above, both in connection with interactive-based computation and the resolution of a dynamical frustration between velocity in response and accuracy in response). This allows the system to proceed swiftly ‘from near-attractor to near-attractor’ more quickly and efficiently, in response to new multimodal inputs and interactions (using Schönner’s terms). This is consistent with the idea that 0 and 1 are asymptotes rather than legitimate attractor points the system can ‘go into’ (Spencer et al. 2009: 109). Notice that this argument also prevents activation probabilities of 0.0, since

In real life, we are very rarely in situations where we have nothing at all on our minds, and then suddenly a stimulus impinges on our minds, instigating a new trajectory through mental space (Spivey 2007: 47)

In other words, we do not ever start processing from scratch; we never stop processing the world around us, and under interactive assumptions, outputs and inputs are entangled, which means that the output of a process is the input of another in a continuous ongoing process (see also Milner 2006: 6–7; Goldin & Wegner 2007: 2), something both behaviorists and cognitivists overlooked (Spivey 2007: 48). This is not incompatible with the idea presented above about rest states, for those states, as we specified, do not equal 0 (i.e., the eigenvector corresponding to the ground state $|n\rangle$ is most emphatically *not* $|0\rangle$), the resting state does



not correspond to a 0.0 activation (rather, $|0\rangle$ would be no activity at all, which corresponds to being dead).²¹

In order to support the argument that there are traces of previous derivational cycles in the workspace empirically, we should need evidence of phenomena involving cross-cycle dependencies. Moreover, those phenomena, if existent, should not be reducible to operations applying in a ‘punctuated’ fashion cycle-by-cycle (as would be the case of Wh-movement in orthodox Minimalism, see Abels & Bentzen 2009 for a generative perspective; Krivochen 2015c for a relativized cyclic proposal involving Wh- interrogatives, parasitic gaps, and multiple-gap constructions). Let us consider an example like (39):²²

(39) Every man and woman who went to a place wants to see it again

If the proposal about structure building made in Krivochen (2015a) is along the right lines (even in its general assumptions), then we are in the presence of a computationally mixed phrase marker, in which we can find dependencies belonging to different levels in the Chomsky Hierarchy co-existing within a phrase marker. The example is complex in its own right, so we will focus on two well-defined aspects of it: quantifier scope and agreement. Assuming the semantically-based structure for coordination proposed in Krivochen (2015a: 551ff.) and Krivochen & Schmerling (2015), the subject presents a universal quantifier [every] having scope over a finite-state unit [man and woman], which is derived via *n*-ary concatenation. This unit is actually the result of taking the objects [man] and [woman] and mentally representing them as a Markovian set, partially following Feigenson (2011). This unit is also modified by a restrictive relative clause RRCl (in traditional descriptive terms, which will do for the purposes of the present discussion) that select a subset of the potential *denotata* for [man and woman], namely, those [who went to a place]. Apart from the quantificational ambiguity in [a place] (which is also present in [every man and woman [RRCl]], and defines two possible interpretations, roughly a collective reading and an individual reading²³), the whole clause

²¹See Piattelli-Palmarini & Vitiello (2015, 2016) for more details about the use of Dirac notation for derivational stages.

²²This example, of course, owes much to historically prior examples like Peter Geach’s ‘donkey sentences’, but we have modified some aspects, like the coordination in the subject, to strengthen our point. However, a pure example of ‘donkey anaphora’ would work as well.

²³These readings could be expressed in terms of ‘strong’ (every / each) vs. ‘weak’ (all) quantifiers, following the lines of the distinction made in Krivochen (2012). There, we also hypothesized that weak Q are used only if the possibility of using the strong Q is for some reason not available, thus generating the (conventional? Generalized conversational?) implicature that the speaker cannot use the strong Q because it either does not apply or because the speaker lacks the relevant information to make a statement using the strong Q (not unlike Grice’s 1975: 49 examples of relative hierarchy among maxims).

enters an interesting relation with the universal Q: is it the case that [every] modifies [man and woman who went to a place] or does the RRCl modify [every man and woman]? So far, phenomena seem quite local. Here is when the [it] enters the game: clearly, there is great structural distance between the pronoun and its referential antecedent. Let us graph the relevant objects using familiar square bracket notation for constituency:

(40) [[every [man and woman] [who went to a place]] [wants to see it [again]]]

In (40), each bracketed object is a finite-state compatible element (a ‘command unit’, in Uriagereka’s 2002a terms), which means its structure can be exhaustively represented by means of finite state processes (so-called ‘monotonic’ structure building, Uriagereka 2002c: 151). Relations among these units are ‘non-monotonic’, and go beyond finite-state computability²⁴ (Uriagereka 2002b, 2012: 53; Krivochen 2015a). Now consider again the relation between [it] and its antecedent [a place]: the pronoun, which belongs to a command unit, establishes a relation with an element embedded within two levels of monotonicity. This relation is twofold: not only is there a semantic relation regarding referentiality, but also a morphological relation, which surfaces as number agreement (cf. [*every man and woman who went to a place_i wants to see them_{i/j} again], subindexes indicating the endophoric relations), but it is not deterministic: the singular morphology is also acceptable, in the collective reading. If the proposal about binding we made in Krivochen (2015b) is along the right lines, then the pronoun is a logical variable, whose phonological exponent depends on the relation it establishes with another element in the overall structure across the limits of a single cycle: such a relation goes beyond the limits of the monotonic unit containing [it]. How is this relation possible? Consider Feigenson’s concept of *set*, which allows access the relevant elements, even though these are conceptualized as a unit for WM purposes: it is plausible that an object of the kind of Feigenson’s *sets* remains active in the workspace as a ‘trace’ of previous cycles, optimally limiting the existence of these traces by the same means by which the limits of WM (or the ‘episodic buffer’) are determined. We will also see that the persistence of remnants of previous cycles as proposed above is essential to account for instances of binding that seem to defy non-stipulative attempts of explanation, and will also prove useful when considering islandhood and Wh- dependencies (all of which have been subsumed to *operator-variable* relations in MGG).

It seems that there are both theoretical and empirical arguments in favor of a certain amount of ‘globality’, which includes cross-cycle relations. How does this impact the entropy dynamics?

²⁴This does not amount to saying their *interpretation* can be carried out by means of a finite-state automaton, we are simply making a statement about the way in which elements are blindly put together.



If there are ‘traces’ or ‘remnants’ of previous cycles in the workspace, which can be projected back to previous derivational stages (again, following the L-grammatical formalism analyzed in Saddy 2016), then it is not possible that entropy goes back to the maximum level (asymptotically tending to 1) after the completion of a cycle, since the next cycle does not start ‘from scratch’. This means there is a difference between $Maxh$ (the maximum entropy value, that is, complete disorder) and the entropy level at the beginning of a cycle. We will call this difference Δh , for $h = \text{entropy}$. If derivations proceed in a cumulative manner, not only locally (within cycles, with each application of the structure building operation to active objects $\{a, b, \dots, n\}$) but also globally (i.e., across cycles, which is roughly Uriagereka’s 2011 idea of global entropy reduction), then it is reasonable to assume Δh is not constant, but increases as derivational cycles are completed and there are more ‘traces’ of what the system has already computed in the workspace. We shall say that Δh increases per cycle by a factor n , where $n\Delta h < Maxh$. We can now make the following hypothesis about the dynamics of the derivational system:

(41) **Hypothesis 1:** There is no derivation for which $Maxh - n(\Delta h) = 0$.

Corollary: $\frac{dh}{dt}$ cannot be 0 for $t > 0$.

Assuming the differentiation quotient $\frac{dh}{dt} = \frac{f(h+\Delta h) - f(h)}{\Delta h}$

Let $f(t) = h$, $f(t') = h'$ for all t , $t' = t + \Delta t$, such that $h' < h \leq Maxh$

In plain English now, derivations display cumulative effects, meaning we do not start from square one (or zero) at every derivational point: the phase space where attractors for our dynamical system can be found gets smaller and smaller (but the system never reaches complete certainty, as Spivey 2007 points out). The system zig-zags its way through a field of attractors, without ever falling into one (see Spencer et al. 2009 for a discussion about the status of cognitive representations as either stable attractors or near-equilibrium states) Hypothesis 1 captures this intuition, the idea that with every derivational step, entropy is reduced at the cost of introducing a new element in the working area, modifying its topological properties (and consuming energy). Since derivations display cyclic properties, the same cumulative informational effects can be extended to the cycle level: no cycle within a derivation starts at the same level of maximum entropy as the previous cycle; we have won something in the meantime, and that something is *certainty* with respect to the location of interpretative attractors. If the term $Maxh - n(\Delta h)$ was equal to 0, that would mean we would be in the presence of a non-cumulative process, in which each cycle takes us back to a state of near complete uncertainty. Moreover, global semantic effects would not be attainable, for the workspace would have to be wiped clean of variables, operators, etc. (otherwise we could not get back to $Maxh$). There are, as we suggested, theoretical



and empirical arguments in favor of accepting Hypothesis 1, or some equivalent formulation, as holding for syntactic derivations.

5 Conclusion

In this paper we have presented some issues pertaining to linguistic derivations from a PoL perspective, particularly, how a cyclic cumulative engine yields locally counter-entropic dynamics, at the cost of energy inputs (external perturbations which disrupt the ground state of cognitive dynamics, as argued in Saddy 2016; Krivochen in progress). A dynamical field-theoretic approach to cognition, at the levels of computation and physiology, not only reveals possible motivations for ubiquitous properties of cognition (like locality and chunking), but also integrates cognition with other physical systems. Such integration must be based on explicit claims about *scale* and the way in which different processes at different scales can emerge from the same underlying process; essentially an oscillatory engine. In connection with the last point, we have claimed that one of the most important properties of cognition is that it displays tensions between mutually incompatible tendencies at different levels: such a tension is called a *dynamical frustration*, and it is this frustration that derives cycles. Bearing in mind questions of scale, it is possible and even necessary to take advantage of developments in physics and mathematics (e.g., quantum field theory, electrodynamics, topology) in order to characterize aspects of cognition that resist characterization in classical terms (roughly, the classical view is the proposal that the mind is a digital computer, see Gallistel & King 2010 for a development of this perspective). On the approach presented in this paper, ‘syntax’ is a topological operation over metric and non-metric fields all throughout the mind.

References

Abels, Klaus & Kristine Bentzen. 2009. A note on the punctuated nature of movement paths. *Catalan Journal of Linguistics* 8, 19–40.

Balari, Sergio & Guillermo Lorenzo. 2013. *Computational phenotypes: towards an evolutionary developmental biolinguistics*. Oxford: Oxford University Press.

Baranger, Michel. 2004. *Chaos, complexity, and entropy*. Unpublished ms.

Binder, Philippe. 2008. Frustration in complexity. *Science* 320(5874), 322–323.

Boccara, Nino. 2002. *Modeling complex systems*. Berlin: Springer.

Boeckx, Cedric. 2007. *Understanding minimalist syntax: Lessons from locality in long-distance dependencies*. Oxford: Blackwell.

Boeckx, Cedric. 2010a. Defeating lexicocentrism. Unpublished ms.



- Boeckx, Cedric. 2010b. *Language in cognition*. Oxford: Blackwell.
- Boyd, Stephen & Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge.
- Bravo, Ana, Luis García Fernández & Diego Krivochen. 2015. On auxiliary chains: Auxiliaries at the syntax-semantics interface. *Borealis* 4(2), 71–101.
- Brody, Michael. 1995. *Lexico-logical form: A radically minimalist theory*. Cambridge, MA: MIT Press.
- Carnap, Rudolf. 1966. *Philosophical foundations of physics*. New York: Basic Books.
- Carr, Jack. 2006. Center manifold. *Scholarpedia* 1(12), 1826.
- Cherniak, Christopher. 2009. Brain wiring optimization and non-genomic nativism. In Massimo Piattelli-Palmarini, Juan Uriagereka & Pello Salabutu (eds.), *Of minds and language*, 108–119. Oxford: Oxford University Press.
- Chomsky, Noam. 1955. *The logical structure of linguistic theory*. Massachusetts Institute of Technology.
- Chomsky, Noam. 1959. On certain formal properties of grammars. *Information and Control* 2, 137–167.
- Chomsky, Noam. 1995. *The minimalist program*. Cambridge, MA: MIT Press.
- Chomsky, Noam. 1998. Minimalist inquiries: The framework. *MIT Occasional Papers in Linguistics* 15.
- Chomsky, Noam. 2004. Beyond explanatory adequacy. In Adriana Belletti (ed.), *Structures and beyond*, 104–131. Oxford: Oxford University Press.
- Chomsky, Noam. 2005. Three factors in language design. *Linguistic Inquiry* 36(1), 1–22.
- Chomsky, Noam. 2007. Approaching UG from below. In Uli Sauerland & Hans Martin Gärtner (eds.), *Interfaces + recursion = language?*, 1–29. New York: Mouton de Gruyter.
- Chomsky, Noam. 2008. On phases. In R. Freidin, C. P. Otero & M. L. Zubizarrieta (eds.), *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*, Cambridge, MA: MIT Press.



Chomsky, Noam. 2015. Problems of projection: Extensions. In Elisa Di Domenico, Cornelia Hamann & Simona Matteini (eds.), *Structures, strategies and beyond: Studies in honour of Adriana Belletti*, 3–16. Amsterdam/Philadelphia: John Benjamins.

Chomsky, Noam & George Miller. 1963. Introduction to the formal analysis of natural languages. In Duncan R. Luce, Robert R. Bush & Eugene Galanter (eds.), *Handbook of mathematical psychology 2*, 269–321. New York: John Wiley & Sons.

Culicover, Peter & Ray Jackendoff. 2005. *Simpler syntax*. Oxford: Oxford University Press.

De Belder, Marijke & Jeroen van Craenenbroeck. 2011. *How to merge a root*. Unpublished ms.

Di Sciullo, Anna Maria & Cedric Boeckx. 2011. *The biolinguistic enterprise: New perspectives on the evolution and nature of the human language faculty*, vol. 1. Oxford: Oxford University Press.

Di Sciullo, Anna Maria & Daniela Isac. 2008. The asymmetry of merge. *Biolinguistics* 2(4), 260–290.

Embick, D. & R. Noyer. 2007. Distributed Morphology and the syntax-morphology interface.

Epstein, Samuel David & T. Daniel Seely. 2002. *Derivation and explanation in the minimalist program*. Oxford: Blackwell.

Escandell Vidal, Victoria María & Manuel Leonetti. 2000. Categorías funcionales y semántica procedimental. In *Actas del Congreso Internacional de Gramática, Universidad de la Laguna*, 363–378. Madrid: Ediciones Clásicas.

Escandell Vidal, Victoria María & Manuel Leonetti. 2004. Semántica conceptual/semántica procedimental. In *Actas del V Congreso de Lingüística General*, Madrid: Arco.

Escandell Vidal, Victoria María & Manuel Leonetti. 2011. On the rigidity of procedural meaning. In Victoria María Escandell Vidal, Manuel Leonetti & Aoife Ahern (eds.), *Procedural meaning*, 81–102. Bingley: Emerald.

Feigenson, Lisa. 2011. Objects, sets, and ensembles. In Stanislas Dehaene & Elizabeth Brannon (eds.), *Space, time and number in the brain: Searching for the foundations of mathematical thought*, 13–22.



Feynman, Richard P., Albert Hibbs & Daniel Styer. 2005. *Quantum mechanics and path integrals*. New York: Dover.

Fodor, Jerry. 1975. *The language of thought*. Cambridge, MA: Harvard University Press.

Gallego, Angel. 2010. *Phase theory*. Amsterdam: John Benjamins.

Gallistel, Charles & Adam King. 2010. *Memory and the computational brain: Why cognitive science will transform neuroscience*. London: Blackwell.

Goldin, Dina & Peter Wegner. 2005. The Church-Turing thesis: Breaking the myth. In Barry S. S. Barry Cooper, Benedikt Löwe & Leen Torenvliet (eds.), *New computational paradigms*, 152–168. Berlin: Springer.

Goldin, Dina & Peter Wegner. 2007. The interactive nature of computing: Refuting the strong Church-Turing thesis. *Minds and Machines* 18(1), 17–38.

Grice, H. Paul. 1975. Logic and conversation. In Peter Cole & Jerry L. Morgan (eds.), *Syntax and semantics, vol. 3: Speech acts*, 41–58. London: Elsevier.

Grindrod, Peter, Zhivko Stoyanov, Garry Smith & James Douglas Saddy. 2014. Resonant behaviour of irreducible networks of neurons with delayed coupling. Submitted to *Physica D*.

Hale, Ken & Samuel J. Keyser. 1993. On argument structure and the lexical expression of syntactic relations. In Ken Hale & Samuel J. Keyser (eds.), *The view from Building 20: Essays in linguistics in honor of Sylvain Bromberger*, 53–109. Cambridge, MA: MIT Press.

Hale, Ken & Samuel J. Keyser. 2002. *Prolegomenon to a theory of argument structure*. Cambridge, MA: MIT Press.

Halle, Morris & Alec Marantz. 1993. Distributed Morphology and the pieces of inflection. In Kenneth Hale & Samuel Jay Keyser (eds.), *The view from Building 20: Essays in honor of Sylvain Bromberger*, 111–176. Cambridge, MA: MIT Press.

Hansel, David, Germán Mato & Claude Meunier. 1995. Synchrony in excitatory neural networks. *Neural computation* 7(2), 307–337.

Harris, Zellig. 1968. *Mathematical structures of language*. New York: Interscience Publishers.

Hasselblat, Boris & Anatole Katok. 2003. *A first course in dynamics: With a panorama of recent developments*. Cambridge: Cambridge University Press.



Hauser, Marc D., Noam Chomsky & W. Tecumseh Fitch. 2002. The faculty of language: What is it, who has it, and how it evolve? *Science* 298(5598), 1569–1579.

Idsardi, William & Eric Raimy. in press. Three types of linearization and the temporal aspects of speech. In Theresa Biberauer & Ian Roberts (eds.), *Principles of linearization*, 31–56. Berlin: Mouton de Gruyter.

Jackendoff, Ray. 1987. The status of thematic relations in linguistic theory. *Linguistic Inquiry* 18(3), 369–411.

Jackendoff, Ray. 1990. *Semantic structures*. Cambridge, MA: MIT Press.

Jackendoff, Ray. 2002. *Foundations of language*. Oxford: Oxford University Press.

Jakobson, Roman. 1961. *Structure of language and its mathematical aspects*. Providence, RI: American Mathematical Society.

Joshi, Aravind K. 1985. Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions? In David Dowty, Lauri Karttunen & Arnold Zwicky (eds.), *Natural language parsing*, Cambridge: Cambridge University Press.

Krivochen, Diego. 2011. An introduction to Radical Minimalism I: On Merge and Agree. *Iberia* 3(2), 20–62.

Krivochen, Diego. 2012. *The syntax and semantics of the nominal construction*. Frankfurt am Main: Peter Lang.

Krivochen, Diego. 2015a. On phrase structure building and labeling algorithms: towards a non-uniform theory of syntactic structures. *The Linguistic Review* 32(3), 515–572.

Krivochen, Diego. 2015b. *An interface approach to binding: Spelling Out referential expressions as a function of syntactic-semantic construal*. Unpublished ms.

Krivochen, Diego. 2015c. Tokens vs. copies: Displacement revisited. *Studia Linguistica*.

Krivochen, Diego. in progress. *Aspects of emergent cyclicity in language, physics, and computation*. PhD dissertation (in progress), University of Reading.

Krivochen, Diego & Peter Kosta. 2013. *Eliminating empty categories: A radically minimalist view on their ontology and justification*. Frankfurt am Main: Peter Lang.



Krivochen, Diego & Susan Schmerling. 2015. *Two kinds of coordination and their theoretical implications: An aspect-based approach*. Manuscript.

Ladyman, James, James Lambert & Karoline Wiesner. 2012. What is a complex system? *European Journal for Philosophy of Science* 3(1), 33–67.

Lasnik, Howard. 2011. What kind of computing device is the human language faculty. In Cedric Di Sciullo, Anna Maria & Boeckx (ed.), *The biolinguistic enterprise: New perspectives on the evolution and nature of the human language faculty*, 354–365. Oxford: Oxford University Press.

Lasnik, Howard & Juan Uriagereka. 2012. Structure at the bottom. In Ruth Kempson, Tim Fernando & Nicholas Asher (eds.), *Philosophy of linguistics*, 33–61. Amsterdam: Elsevier.

Lasnik, Howard, Juan Uriagereka & Cedric Boeckx. 2005. *A course in minimalist syntax: Foundations and prospects*. Oxford: Blackwell.

Mandelbrot, Benoît. 1983. *The fractal geometry of nature*. New York: W.H. Freeman.

Marr, David. 1982. *Vision: A computational investigation into the human representation and processing of visual information*. Cambridge, MA: MIT Press.

Mateu Fontanals, Jaume. 2002. *Argument Structure. Relational Construal at the Syntax-Semantics Interface*. PhD dissertation, Universitat Autònoma de Barcelona.

McCarthy, John. 2010. *An introduction to harmonic serialism*. University of Massachusetts, Amherst.

Medeiros, David. 2015. *Stack sorting syntax: Linking sound and meaning without trees or movement*. Unpublished ms.

Milner, Robin. 2006. Turing, computing and communication. In Dina Goldin, Scott Smolka & Peter Wegner (eds.), *Interactive computation: The new paradigm*, 1–8. Berlin: Springer.

Moss, Hellen, Lorraine Tayler & Kristen Taylor. 2007. *Conceptual structure*. Oxford: Oxford University Press.

Müller, Gereon. 2011. *Constraints on displacement: A phase-based approach*. Amsterdam: John Benjamins.

Panagiotidis, Phoevos. 2009a. *Four questions on categorization of roots*. Unpublished ms.



Panagiotidis, Phoevos. 2009b. *Functional heads, Agree and labels*. Unpublished ms.

Papo, David, Javier M. Buldú, Stefano Boccaletti & Edward T. Bullmore. 2014. Complex network theory and the brain. *Philosophical transactions of the Royal Society of London B* 369(1653).

Pesetsky, David & Esther Torrego. 2007. Probes, goals and syntactic categories. In *Proceedings of the 7th Tokyo Conference on Psycholinguistics*, Kyoto University, Japan.

Piattelli-Palmarini, Massimo & Giuseppe Vitiello. 2015. Linguistics and some aspects of its underlying dynamics. *Physics of Life Reviews* .

Piattelli-Palmarini, Massimo & Giuseppe Vitiello. 2016. Third factors in language design: some suggestions from quantum field theory. In James McGilvray (ed.), *Cambridge companion to Chomsky*, Cambridge: Cambridge University Press.

Postal, Paul M. 2012. *Chomsky's foundational admission*. Unpublished ms.

Putnam, Michael. 2007. *Towards a derivational syntax: Survive-minimalism*. Amsterdam: John Benjamins.

Putnam, Michael. 2010. Exploring crash-proof grammars. In Michael Putnam (ed.), *Exploring crash-proof grammars*, 1–17. Amsterdam: John Benjamins.

Roberts, Mark. 2015. Ultrametric distance in syntax. *Prague Bulletin of Mathematical Linguistics* 103(1), 111–130.

Saddy, Douglas. 2016. *Syntax and uncertainty*. Unpublished ms.

Saddy, Douglas & Diego Krivochen. 2016. Emergent oscillations and cyclicity: Physical aspects of frustrated cognitive systems. Presented at First International Symposium on the Physics of Language, Sophia University, Tokyo, 2016/03/04–05.

Saddy, Douglas & Juan Uriagereka. 2004. Measuring language. *International Journal of Bifurcation and Chaos* 14(2), 383–404.

Schöner, Gregor. 2009. Development as change of system dynamics: Stability, instability, and emergence. In John P. Spencer, Michael S. C. Thomas & James L. McClelland (eds.), *Toward a unified theory of development connectionism and dynamic system theory re-considered*, Oxford: Oxford University Press.

Shannon, Claude. 1948. *A mathematical theory of communication*. Bell System, Technical Journal.



Smith, Peter. 1998. *Explaining chaos*. Cambridge: Cambridge University Press.

Spencer, John, Sammy Perone & Jeffrey Johnson. 2009. *Dynamic field theory and embodied cognitive dynamics*. Oxford: Oxford University Press.

Sperber, Dan & Deirdre Wilson. 1995. *Relevance: Communication and cognition*. Oxford: Blackwell.

Spivey, Michael. 2007. *The continuity of mind*. Oxford: Oxford University Press.

Stroik, Thomas. 2009. *Locality in minimalist syntax*. Cambridge, MA: MIT Press.

Stroik, Thomas & Michael Putnam. 2013. *The structural design of language*. Cambridge: Cambridge University Press.

Talmy, Leonard. 1983. How language structures space. In Herbert L. Pick Jr. & Linda P. Acredolo (eds.), *Spatial orientation: Theory, research, and application*, 225–282. New York: Plenum Press.

Talmy, Leonard. 2000. *Toward a cognitive semantics*. Cambridge, MA: Massachusetts Institute of Technology.

Udgaonkar, Jayant. 2001. Entropy in biology. *Resonance* 6(9), 61–66.

Uriagereka, Juan. 1998. *Rhyme and reason*. Cambridge, MA: MIT Press.

Uriagereka, Juan. 1999. Multiple spell-out. In Norbert Hornstein & Samuel Epstein (eds.), *Working minimalism*, 251–282. Cambridge, MA: MIT Press.

Uriagereka, Juan. 2000. On the emptiness of the ‘design’ polemics. *Natural Language & Linguistic Theory* 18(4), 863–871.

Uriagereka, Juan. 2002a. Conceptual matters. In *Derivations: Exploring the dynamics of syntax*, London: Routledge.

Uriagereka, Juan. 2002b. Multiple spell-out. In *Derivations: Exploring the dynamics of syntax*, London: Routledge.

Uriagereka, Juan. 2002c. Warps: Some thoughts on categorization. In *Derivations: Exploring the dynamics of syntax*, London: Routledge.

Uriagereka, Juan. 2008. *Syntactic anchors: On semantic restructuring*. Cambridge: Cambridge University Press.

Uriagereka, Juan. 2011. *A sketch of the grammar in non-classical conditions*. Unpublished ms.



- Uriagereka, Juan. 2012. *Spell out and the minimalist program*. Oxford: Oxford University Press.
- Uriagereka, Juan. 2014. *Regarding the third factor: Arguments for a CLASH model*. Amsterdam: John Benjamins.
- Uriagereka, Juan, Roger Martin & Angel Gallego. 2016. Language combinatorics as matrix mechanics: Conceptual and empirical bases. Presented at First International Symposium on the Physics of Language, Sophia University, Tokyo, 2016/03/04–05.
- Watumull, Jeffrey. 2012a. The computability and computational complexity of generativity. *Cambridge Occasional Papers in Linguistics* 11, 311–329.
- Watumull, Jeffrey. 2012b. A Turing program for linguistic theory. *Biolinguistics* 6(2), 222–245.
- Watumull, Jeffrey, Marc Hauser, Ian Roberts & Norbert Hornstein. 2014. On recursion. *Frontiers in Psychology: Language Sciences* 4, article 1017.
- Wegner, Peter. 1997. Why interaction is more powerful than algorithms. *Communications of the ACM* 40(5), 80–91.
- Wilson, Deirdre & Dan Sperber. 2003. *Relevance theory*. Oxford: Blackwell.
- Wilson, Deirdre & Dan Sperber. 2004. Relevance theory. In Lawrence Horn & Gregory Ward (eds.), *The handbook of pragmatics*, 607–632. Oxford: Blackwell.
- Wurmbrand, Susi. 2014. *The Merge condition: A syntactic approach to selection*. Amsterdam: John Benjamins.
- Zipf, George. 1949. *Human behavior and the principle of least effort*. Cambridge, MA: Addison-Wesley.
- Zwarts, Joost. 2003. *Vectors across spatial domains: From place to size, orientation, shape, and parts*. Oxford: Oxford University Press.
- Zwarts, Joost & Peter Gärdenfors. 2016. Locative and directional prepositions in conceptual spaces: The role of polar convexity. *Journal of Logic, Language, and Information* 25(1), 109–138.



Language, chaos and entropy: A physical take on biolinguistics
Diego Gabriel Krivochen

Received: 2013/12/12

Revised: 2014/08/08

Accepted: 2014/09/02

Diego Gabriel Krivochen
University of Reading
d.g.krivochen@pgr.reading.ac.uk