Towards a Geometrical Syntax

Diego Gabriel Krivochen, University of Reading

e-mail: diegokrivochen@hotmail.com

Abstract:

In this paper we attempt a logic-mathematical formalization of the program for linguistic investigation that we have called Radical Minimalism, in order to make the model fully explicit. For those purposes, we have devised a logical model of syntax (understood as a general-purpose free and unbounded generative algorithm), with which we attempt to provide tools to capture the properties of mental workspaces and the cognitive organization of the phenomenological reality. We will combine our *Strong Radically Minimalist Thesis* with Tegmark's (2007) *Mathematical Universe Hypothesis*, and analyze how both strengthen in interaction gaining in empirical coverage and theoretical weight. The aim is to formalize a possibility for the study of language as a physical system from an interdisciplinary perspective: a joint work between Formal Linguistics, Logic, Physics and Mathematics.

Keywords: formal syntax, Radical Minimalism, mathematics, interfaces in language design

Comments 5/2015: This paper assumes a discrete approach to workspaces. More recently, I have developed a field-theoretical approach that is more in tone with dynamical systems theory, non-linear processes, and so on. A field-theoretical approach also works better when it comes to deal with dimensionality reduction for materialization purposes (Spell-Out). However, it is useful to see this as a first step towards that approach. Structures built here are both locally and globally Euclidean, whereas within the newer perspective we deal with manifolds, only locally Euclidean. We beg the reader to cherry-pick here, and see more recent works on phrase structure building and mapping for further reference. The section 'works in progress' might help at that respect.

1. INTRODUCTION

Our goal in this paper is to provide tools for the formalization of a model of syntax with bases on geometry and set theory, following the proposals made in Krivochen (2011a, b) and Krivochen & Miechowicz-Mathiasen (2012). Using formal tools devised within mathematical logic, we will first provide definitions of the substantive primitives we will work with and the operations that apply to such primitives, in order to generate structures of a higher order of complexity. The result, we expect, will constitute the prolegomenon to further studies of language within a mathematical framework, in which ambiguities in the formulation of principles and concepts disappear by adopting a universal notation traceable back to Frege's *Begriffsschrift* and Russell & Whitehead's *Principia Mathematica*, adapted to current demands of simplicity and economy on derivation and representation. Our vocabulary will be that used in set theory (element of, subset of, intersection, union), plus existential and universal quantifiers and the negation operator.

We have chosen geometry because it licenses the possibility of *n*-dimensional representations, unlike logics, whose representations, although hierarchical, are bi-dimensional, and therefore inaccurate to represent processes in the quantum model of the mind we work with (see Penrose & Hameroff, 2011; Penrose, 1997; Vitiello & Freeman, 2007; Krivochen, 2011a, b, among others). We will depart from a set of assumptions, partially shared with recent syntactic research (e.g., Culicover & Jackendoff, 2005; Uriagereka, 2012; Lasnik & Uriagereka, 2012):

- Language is a part of the natural world (a common claim in biolinguistics), therefore, it is a physical system.
- There is a computational component in charge of generating complexity departing from simpler objects, which we will refer to as the "syntax" or "generative component"
- The formal technology 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 proceed step-by-step in real time (*contra* Chomsky, 2007, who explicitly rejects the diachronic nature of the generative operation Merge)
- Constraints are only determined by interpretative systems. Derivations are unconstrained *per se*.
- Complexity arises in any system independently, unless there exists an interface system. Thus, Conceptual Structures CS manipulating generic concepts are syntactically derived (i.e., by means of a generative algorithm) without there being 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)

The main thesis of this framework is the following:

Strong Radically Minimalist thesis (SRMT):

All differences between physical systems are "superficial" and rely only on the characteristics of their basic units [i.e., the elements that are manipulated], *which require minimal adjustments in the **formulation** of operations and constraints* [that is, only notational issues]. *At a **principled level**, all physical systems are identical, make use of the same operations and respond to the same principles.*

2. SRMT and MUH:

In this section we spell out how our hypotheses are influenced by Tegmark's conceptions of the mathematical structure of the physical reality. We will develop our framework and then tackle the problems he finds in mathematical structures, to which we will come back in the conclusion. Our work is intimately related to Tegmark (2007) and his *Mathematical Universe Hypothesis*, stated as follows (Tegmark, 2007: 1):

1. **Mathematical Universe Hypothesis (MUH):**

Our external physical reality is a mathematical structure.

A (mathematical) structure is defined as a set $\mathbb{R}$ of abstract entities and relations $\{R_1, R_2 \ldots R_n\}$ between them.

The "physical reality" of the MUH is identical to the "physical systems" of SRMT, and the postulation of a subjacent mathematical structure is made explicit in SRMT with the mention of Universal *operations* and *principles*, which we have formulated in past works (Krivochen, 2011a, b, 2012; Kosta & Krivochen, 2011; Krivochen & Kosta, 2013). Our Radical Minimalism provides principles and operations that go in consonance with both MUH and its stronger version, the *Computable Universe Hypothesis* (Tegmark, 2007: 20):

2. **Computable Universe Hypothesis (CUH):**

The mathematical structure that is our external physical reality is defined by computable functions.

Tegmark explains this hypothesis as meaning that "(…) *the relations (functions) that define the mathematical structure* […] *can all be implemented as computations that are guaranteed to halt after a finite number of steps* (…)" (Tegmark, 2007: 20). This is clearly explained in the Radically Minimalist derivational dynamics, which we will formalize below: we only need one computational generative[1] operation, *concatenation*, which halts if and only if a fully interpretable object what we have called a *phase* (following Chomsky, 1998) is recognized by an interpretative system (in the case of language, either phonology or semantics), and *transferred* to (i.e., taken by) that system.

What linguistics can provide a mathematical approach like Tegmark's is not only empirical basis, but also strong predictions regarding a limited domain of the physical reality, in which those predictions, that follow from general hypothesis, can be put to test (Cf. Carnap, 1966). At the same time, a mathematical approach to the "natural world" results in a formalization that is most necessary for deepening our understanding of the very structure of reality, in our case, the structure and functioning of the mind-brain, resulting in a fully explicit model. This, in our opinion, can and should be expressed in unambiguous mathematical terms. In this work, we attempt to formalize Radical Minimalism in such a way that we can base syntax, understood as a generative mechanism operating there where complexity is necessary, in mathematics, following a geometrical model. This framework also reinforces Izard, Pica et. al. (2011)'s theory of geometry as a Universal mental constructions, upon Euclidean basis[2]. After having presented our formalized model in section 3, we will analyze to what extent RM can contribute to answer some questions that have arisen within the MUH framework and make the system more powerful.

3. Towards a Geometrical Syntax:

In this section we will provide some concepts and definitions that will configure our formal model. Afterwards, we will circumscribe ourselves to language, and analyze the consequences the adoption of this model has for the study of syntax[3].

[1] Contrarily to Chomsky (1995: 162, fn. 1), we do not understand "generative" as simply "fully explicit", but actually as "creating structural relations in real time". This is what a generative grammar should aim at explaining, in our conception. See Krivochen (2011a, 2012, 2013) for details and discussion.

2 In our framework, the mathematical workspace can be Euclidean or not (hyperbolic or elliptic) depending on interface requirements, see Krivochen & Miechowicz-Mathiasen (2012). What is innate, in the case of human mind, is the quantum processor that allows multiple derivations simultaneously and objects in multiple states while in the working area. Interface requirements derive from the interaction between interpretative components and the phenomenological world.

3 The reader may find it useful to confront our formalization with that of Collins & Stabler (2012), which formalizes orthodox minimalism following Chomsky's (1995, et. seq.) proposals.

Definition 1 (non-final): a point is an *n*-tuple (x, y, z…*n*), where *n* = number of dimensions[4] of a workspace W.

Definition 2: W is an *n*-dimensional *generative* workspace, that is, a workspace where structure is built via *concatenation*[5]. Taking two distinct workspaces W_X and W_Y, either:

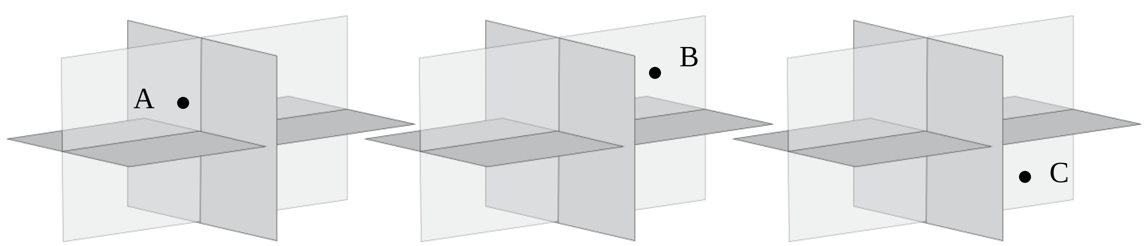
a) $W_X \equiv W_Y$ *iff* $\forall(x) \mid x \in W_X, x \in W_Y \wedge \nexists(x), x \in W_X, x \notin W_Y$

b) $W_X \neq W_Y$ *iff* $\nexists\ (x) \mid x \in W_X, x \in W_Y$

c) $W_X \cong W_Y$ *iff* $\exists(x) \mid x \in W_X, x \in W_Y$

This allows us to define the relations of *identity*, *difference* and *similarity* in set-theoretical terms. *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 (a)). 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 (b)). *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 (c)). Let us consider Figure (1) to exemplify the generative operation we are to formalize:

(1)



Workspace 1 (W1) Workspace 2 (W2) Workspace 3 (W3)

a) *Concatenate* (A ∈ W1, B ∈ W2, C ∈ W3)

[4] Throughout the paper, we will use the term "dimension" in the sense in which it is used in mathematics and physics, when saying, for instance, that a string has "one dimension" or a Calabi-Yau figure has six.

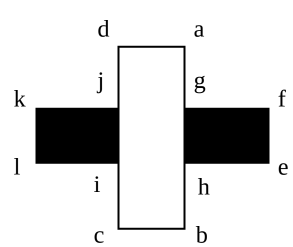
[5] See section 4.5 below for a brief analysis of interpretative vs. generative systems within the mind.

b) Triangle α = {A ∈ W1, B ∈ W2, C ∈ W3}

Of course, α is a *mental construct* in W_4, a workspace in which concatenation is interpreted. $\{W_1 \ldots W_4\}$ must be isodimensional for interface purposes, so that the coordinates can be joined in a single W (our W_4). [5/2015: within a field-theoretical framework, understanding elements entering relations as perturbations of a field, defined by *ket* vectors, this condition follows] If we are working with the mind-brain, W is at most 3-D, and, according to recent research (Gómez Urgellés, 2010; also Afalo & Graziano, 2008 for a failed attempt to elicit 4-D judgments), it should be *hyperbolic* since human perception tends to construct the outer physical reality as a hyperbolic space. In humans, Ws emerge from the interaction between the prefrontal cortex and relevant areas of the brain and is ***tri*-dimensional**. It is worth noticing that our mind-brain, which works in a tri-dimensional phenomenological world, can conceive the fourth dimension, but there is no account for dimensions > 4, while they are theoretically possible. A legitimate question would be whether the characteristics of the phenomenological world in which an *interpretative system* matures (say, the human conceptual system) determine its dimensional limitations. We believe they do, in the same way that the flow of Time, which can be cognitively defined as a *concatenation* of coordinates in 3-D workspaces (i.e., concatenation of "snapshots" creating the illusion of continuity), affects temporal schemes known as *Aktionsarten* (Vendler, 1957): in a lightspeed world there would be no *achievements* (i.e., telic-non durative events). To be concrete, a bomb could "be exploding" for as long as one can think.

Not only these kinds of abstract figures like (1b) require more than one W and a separate W to put the information together. Consider figure (2):

(2)



In (2) what we can *objectively* see is a white rectangle and two black rectangles at the sides. It is likely that each is perceived and processed in a separate W, such that rectangle {a, b, c, d} is processed, say, in W_1, {e, f, g, h}, in W_2 and {i, j, k, l} in W_3. However, the reader most likely constructed a single rectangle {e, f, k, l}, assuming that {g, h, i, j} are the points of intersection between a white rectangle and a single black rectangle. This operation of "reconstruction" must be performed in a further W, such that it is *isodimensional* to the others, which are in turn *isodimensional*: this means, not only having the same number of dimensions, but also *the same dimensions* (e.g., a Calabi-Yau figure has six specific dimensions, therefore, only a W with those very six dimensions will be taken as *isodimensional* to a Calabi-Yau figure). This procedure only makes sense if there is an *interpretative system* that has to read the information, and it is driven by economy: perceiving *two* figures is simpler than perceiving *three*: two figures allows the mind to activate the well-known *figure-ground* dynamics, and so we can say "there is a white rectangle above the black one", even if it only occurs in our conceptual space, and is ultimately an *inference*.

Workspaces, crucially, have *neurocognitive* entity: following Baddeley (1992, 2007); D'Espósito (2007); among others (see relevant articles in Arbib, 2003; Taylor et. al., 2011, for recent references), the working bench, or *working memory* is not a formal construct by nature (even if it can be formalized) but a biologically based result of the interaction between the pre-frontal cortex and different areas of the brain, depending on the kind of information to be computed (e.g., according to Pylyshyn, 2007, the parietal and temporal lobes are involved in the computation of Figure-Ground structures, in more familiar terms, Prepositional Phrases PPs).

Definition 3: *concatenation* defines a *chain* of coordinates in *n*-dimensional generative workspaces W of the form $\{(x, y, z\ldots n) \subset W_X, \ldots (x, y, z\ldots n) \subset W_Y, \ldots (x, y, z\ldots n) \subset W_n\}$ where $W_Y \equiv W_X \equiv W_n$ or $W_Y \neq W_X \neq W_n$. If $W_X \neq W_Y$, they must be isodimensional in the sense specified above.

The *concatenation* function is frequently called *Merge* (Chomsky, 1998, 2005, 2007 and much related work), which we defined informally as follows (Krivochen, 2011a, to which we refer the reader for discussion and review of recent proposals):

Merge *is a* ***free unbounded operation*** *that applies to an n number of objects sharing format, either ontological or structural.*

[5/2015: see 'On Phrase Structure Building: Towards a non uniform theory of syntactic structures', *The Linguistic Review* 32(3) for thorough discussion and a more detailed presentation of my proposal]

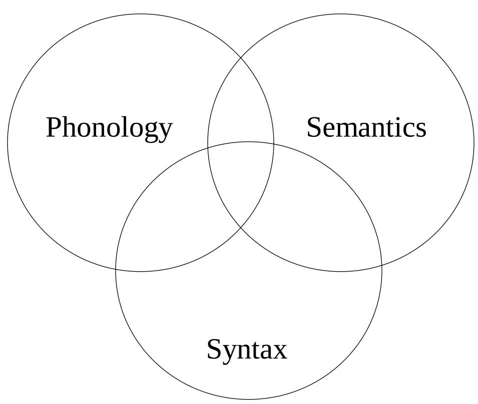
Definition 4: geometrical "figures", sentences, and other "observable" objects which constitute the *phenomenological world* are epiphenomenal results of *concatenation* in one or several W read off at an interpretative system.

This definition, which follows clearly from what we have been saying, can be formulated in a stronger way: *there is no physical reality beyond the interpretation of the concatenation function applied to an n-number of objects sharing format*. This is another way to express Tegmark's MUH; the so-called physical reality *is* a mathematical structure. We accept MUH without necessarily accepting ERH (the External Reality Hypothesis) as it is formulated: in any case, the notion of *external* and *reality* should be redefined (see the discussion in Jackendoff, 2002 for an interesting non-realist / non-solipsist perspective). This must not be interpreted as a plea for solipsism: the concatenation function applies to objects that are *external* to the human mind and perception plays no role in generation. Moreover, there is no need to resort to a *human* mind: an automaton with the algorithm incorporated and interpretative routines (based on Relevance Principles) could serve as well.

Definition 5: an *Interface Level* IL is an *interpretative system* that has access to a W in which *concatenate* applies and establishes legibility conditions regarding its output.

It is customary since Aristotle to distinguish two such levels involved in language: phonology (Phonetic Form PF in generative models) and semantics (Logical Form LF). We will maintain the generativist labels for the interpretative components associated to language as a whole (Chomsky, 1995), but make the additional note that they are not the only ones, nor does their involvement in a computational procedure guarantee the emergence of language. For example, assuming the following Venn diagram,

(3)



the following intersections seem to hold:

(4) Sem ∩ Phon = interjections (for example, Chomsky, 2005 claims that interjections lack a property to make them mergeable to other syntactic objects, therefore, their combinatorial potential is null, in his theory)

Sem ∩ Syn = conceptual structures (see Jackendoff, 2002; Culicover & Jackendoff, 2005; Mateu Fontanals, 2005; Taylor, et. al., 2011; also the sense in which D-Structure is interpreted in Uriagereka, 2008, 2012)

Syn ∩ Phon = musical capacity (Jackendoff & Lehrdal, 1983)

Syn ∩ Sem ∩ Phon = natural language

This definition allows us to have a characterization of so-called "natural language" as the intersection between a computational system and systems providing substance to be manipulated, without implying that these systems are not themselves computational (see also Stroik & Putnam, 2013 for a similar architecture)[6]. We will take it that, as the simplest case, all constraints over representations are interface conditions, and not intra-systemic stipulations. In other words, given the fact that *concatenation* is a free *n*-ary operation, it would be stipulative to restrict its power via computational filters. We thus leave the bargain to legibility conditions, which have to be formulated anyway if the syntactic component is to be regarded, as in orthodox Minimalism, as "perfect" in any non-trivial sense (Chomsky, 1995 and much subsequent work, also Brody, 2003, particularly Chapter 7). In our framework it does not make any sense to ask whether syntax is perfect or optimal, since it is an *n*-ary concatenation function, impoverished as it can be. What does make sense is to ask whether a particular output optimally solves interface requirements, but that is a completely different question (which might not arise in the former framework, see Culicover & Jackendoff, 2005: 14, ff. for discussion). It has also been argued that the "design" polemics is actually misleading, unscientific or trivial (Behme, 2013; Culicover & Jackendoff, 2005: 88-92; Borsley & Börjars, 2011: 3, to give some examples), but such a discussion goes beyond the scope of the present work. What we do claim is that no formalization should be based on a notion like "perfection" or undefined (rather, undefinable) concepts like UG or the very Faculty of Language (Cf. Collins & Stabler, to appear), since sets of features –essential for these approaches-, for example, are stated, not defined.

[6] The reader will notice we have not assumed the existence of UG of a language faculty, to this end. Cf. Collins & Stabler (to appear: 1) "*An I-Language is a pair <LEX, UG>…*". As it derives from our assumptions, we assume neither, but rather decompose the LEX and render UG unnecessary.

Theorem 1: all operations within Ws are interface-required in order to satisfy *Dynamic (Full) Interpretation*.

<u>Principle 1:</u> ***Dynamic (Full) Interpretation***: *any derivational step is justified only insofar as it increases the information and/or it generates an interpretable object*.

At this point, it is crucial to say that, if our view of syntax is that of a generative component that manipulates objects regardless their nature, then it can be applied to any so-called "complex object" insofar as complexity can be decomposed in layers of simple, atomic elements somehow concatenated for interpretation purposes. If this view is correct, all physical systems would have "derivations", in the sense of "successive applications of *concatenation* and subsequent interpretation" (insofar as the result of concatenation, complexity, is subjected to experimentation at higher levels of matter organization than, for example, the Planck scale). This formulation has another interesting consequence: *concatenation* (where there is an IL involved) turns out to be *anti-entropic* (a thesis put forth by Uriagereka, 2011). That is, *concatenation* is driven by the need to increase the informational load for IL *step by step* (not globally, as global information increase would result in extra information, also undesirable). Consider the following scenario:

(5) Concatenate (a, a) = {a, {a}}

In this derivational step, no information is added, thus, (5) is equivalent to (6):

(6) Concatenate (a) = {{a}}

Boban Arsenijevic (p.c) claims that "*{{a}} is non-trivial in at least one faculty: the arithmetic capacity. Hence, output conditions* [i.e., the conditions established by interpretative systems over the result of generative operations] *can't be that bare to favor a binary merge*". However, our position is that if *concatenation* is considered to be a module-neutral operation and we also assume DFI, that is, the assumption that the application of any operation must either lead to a legible object for an interface level IL or increase the informational load, applying *concatenation* to a single object (or to an object and an empty set, as in De Belder & van Craenenbroeck, 2011) is trivial in any faculty. If {a} is already legible in the relevant interface level, then why apply *concatenation* in the first place? What triggers generation if the interpretative levels do not have a tool like DFI to monitor the derivation locally? It would be computationally redundant to have such examples of "self-Merge" (but see Adger, 2011 for a theory based on such assumptions), and therefore far from Minimalist. It is true that the arithmetic capacity can work with singletons, but the function that applies to only one object is not *concatenation* (or Merge), but a successor function, as in Peano's axioms (see Leung, 2010). We will come back to this issue below.

Definition 6: ($W1_0 \ldots W1_n$) indicate successive derivational steps *within* a W.

This definition makes a point of the strongly derivational character of the computational system, at the same time it introduces the notion of derivational steps as essential in the theory: the *time* variable is to be taken into account when considering derivational steps (cf. Chomsky, 2007).

<u>On Information and Objects:</u>

<u>**Principle 2:**</u> ***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.*

This version of ConsP derives quite directly from that used in Lasnik & Uriagereka (2005: 27, 53), but with a plus: we do not refer only to lexical or linguistic derivations only, but, more generally, to any sequence of computational operations (Turing, Markovian, non-linear alike). Thus, the steps to solve an equation, the successive applications of *concatenation* required to generate a sentence, the step-by-step processing in a Turing machine, among other examples, cannot eliminate information. Relevantly, the reader must take into account that, since the output of a derivation (an expression) is not directly proportional to its input (lexical items), it is possible that the dynamics of a syntactic derivation are actually non-linear and therefore, not functions, instead of Turing-computable functions (cf. Watumull, 2012 for an opposing view).

Theorem 2: *Entropy* in a derivation occurs only when there is a *generative-interpretative* interaction. If there is no *interpretative system* to read a structure, there is no point in introducing *entropy*.

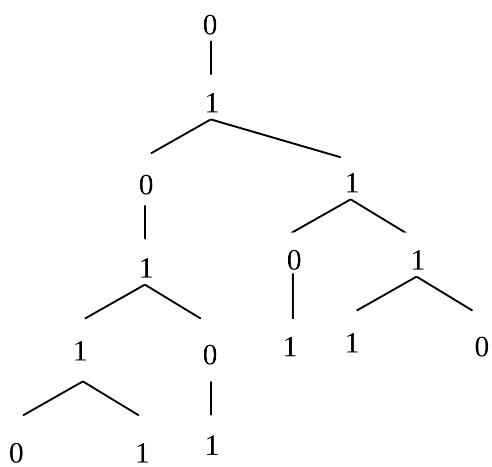
As we said in the section above, the anti-entropic character of computations is naturally derived if DFI holds. If correct, this would lead us to a highly desirable (partial) unification of language with other physical systems, both in the generative and interpretative aspects. Contrarily to the use in Chomskyan linguistics, where the generative operation Merge apparently comes "for free" (e.g., Chomsky, 2005, 2007) but is not justified outside the dubious notion of "Virtual Conceptual Necessity", in our framework the existence of a generative algorithm is required by the necessity to reduce entropy when a system interfaces with another. However, in the absence of interpretation (as it is the case with the arithmetical capacity) there is no point in introducing entropy. It would be incorrect, from this point of view, to say that the successor function in Peano's system is counter-entropic, because there is no system to handle the information to. Moreover, it is difficult to establish exactly what we mean by "information" in a purely numerical sequence dealing with natural numbers. The issue may not arise, however, if we are dealing with a sequence like Fibonacci, which can be generated as an emergent property per generation with a standard Lindenmayer grammar:

(7) $0 \rightarrow 1$

$1 \rightarrow 0, 1$

Developing the representation in a tree manner gives us the following result:

(8)



If the values of all nodes (either terminal or non-terminal) are added at any point W_n, the result will be, in top-down order, (0, 1, 1, 2, 3, 5…), the Fibonacci (Fib) sequence (see Uriagereka, 1998: Chapter 3, figure 3.10). However, it would be wrong to assume an L-grammar is the only way to generate the Fib sequence (it is possible to derive it via context-sensitive rules), whereas, if we assume the simplest scenario (eliminating features and stipulative triggers), we are left with no choice but *concatenation* for linguistic derivations.

Definition 7: LEX_S is the full set of *type*-symbols that can be manipulated by a computational system S, which applies in a generative W.

In our case, S = *concatenation*, there is nothing more to it. In Chomskyan linguistics, there are considerations of feature valuation (e.g., Pesetsky & Torrego, 2004, 2007; Wurmbrand, 2013, among many others) and *Agree* that restrict and enrich S beyond computational necessity: to generate the same output, we can use a simpler operation.

Definition 8: an *array* is a set of *types* drawn from LEX_S.

Definition 9: a *type* is an abstract element in a physical system Φ.

In our proposal, derived from Relevance Theory (Sperber & Wilson, 1986; Wilson & Sperber, 2003), there are two kinds of *types* in a *linguistic* derivation: those that convey *conceptual* meaning (i.e., *roots*) and those that convey instructions as to how to interpret the *relation* between conceptual types (*procedural* types). Determiner, Time and Preposition are *procedural types* (just like operators like [+], [-], etc. are procedural in Mathematics). The procedural or conceptual character of a node is of no importance to the syntax (as the syntactic component is purely generative, being blind to the characteristics of manipulated objects), it is read at the semantic interface, and only there is it of any relevance. We will come back to this below.

Definition 10: a *token* is an occurrence of a *type* within 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.

[5/2015: see 'Tokens vs. Copies: Displacement Revisited', to appear in *Studia Linguistica*, for details of implementation]

Definition 11: a *token* is never fully interpretable at the relevant Interface Level IL unless within a larger structure.

Within Linguistics, this definition is the RM way to formalize Distributed Morphology's "*categorization assumption*", adding empirical and theoretical coverage: we eliminate extra elements like "categorizers" (Panagiotidis, 2010; Fábregas, 2005; Marantz, 1997). This has been expressed as the *Conceptual-procedural interface symmetry* (Krivochen, 2012)*:*

There cannot be bare roots without having been merged with a procedural node or procedural nodes without having been merged with a root in the syntax-semantics interface.

Corollary: an element in W_X is always a *token*. Its interpretation depends on its own content and the structural position of its occurrence (i.e., local relations with other tokens, *conceptual* or *procedural*).

Token-collapse

Be S a set {α, β, …, *n*} of arbitrarily complex tokens in positions *P* within a derivational workspace W. An Interface Level *IL* will establish a dependency between the members of S *iff*:

a. The members of S are mutually disconnected

b. No two members of S belong to the same domain D, and there is no syntactic object SO, such that SO ∈ D and SO is logically equivalent to a member of S for interface purposes

c. The members of S are structurally identical as far as format is concerned

Notice that condition (b) implies both anti-locality *and* the mixed chains proposal, since a uniform chain would include intermediate occurrences of a SO within a domain (e.g., movement of Compl-V to Outer-Spec-*v* within the *v*P phase), which is here rejected on interface grounds. Condition (c) refers to whether the tokens in question are terminals or non-terminals (what we have referred to as 'structural format' in Krivochen, 2011), and rules out token-collapse involving a terminal and a full XP (which has interesting empirical consequences, particularly for sub-extraction, into which we will not go here).

Definition 1' (final): A point $p = (\{x, y, z\ldots n\} \in W_X)$ is a *token* of a single-*type* LEX_S.

Definition 1'': A lexical item LI is a structure $\{X\ldots\alpha\ldots\surd\} \in W_X$, where X is a procedural category (Determiner, Tense, Preposition), α is an *n* number of non-intervenient nodes for category recognition purposes at the semantic interface, and √ is a *root*.

Roots are pre-categorial linguistic instantiations of a-categorial generic concepts from the Conceptual-Intentional system. Generic concepts are "severely underspecified", since they are used by many faculties, and therefore cannot have any property readable by only some of them; otherwise, the derivation would crash in whatever faculty we are considering. Roots convey conceptual instructions, whereas functional nodes convey procedural instructions to the post-syntactic semantic parser. In formal terms,

$\surd = LEX_S$, where $S = \{\alpha_1 \ldots \alpha_n\}$ (a set of intensional properties)

<u>Procedural elements:</u> according to Escandell & Leonetti (2004), traditional *functional nodes* in generative syntax convey *procedural* instructions to the post-syntactic semantic parser as to how to manipulate a given semantic substance. The concept of "procedural instruction" can be better refined as follows:

<u>Procedural instructions:</u>

- Restrict reference in terms of a proper subset of the root. Each element X, Y, restricts the set in different ways, say:
    - $\surd = \{\alpha, \beta, \gamma, \lambda, \delta\}$
    - $[X, \surd] = \{\alpha, \beta, \gamma\}$
    - $[Y, \surd] = \{\gamma, \lambda, \delta\}$
- Provide instructions as to:
    - Where to retrieve information, assuming a massively modular architecture of the mind in which specialized modules interact in interpretation.
    - What kind of information to retrieve

A root √ is a semantic *genotype* (i.e., establishes potentials and limits for meaning variation, including argumental alternances –caused/uncaused- and category variation, see Panagiotidis, 2010; Hale & Keyser, 2002; Mateu Fontanals, 2002 for details). For all natural languages NL, it is the case that $\forall(x)$, $x = \surd$, NL $\ni$ x. LIs in particular NLs (English, Spanish, etc.) are *phenotypic* instantiations of the possibilities offered by a root, and it can be the case that $\exists$(NL) $\wedge$ NL $\ni$ x $\wedge$ NL $\not\ni$ LI. This view contrasts greatly with the highly relativistic view of Sapir-Whorf, which focuses on the particular influence context has in linguistic availability. While there is sure a socio-historical influence on coinage (for example), the underlying mechanism via which lexical structures are "assembled" must be universal (which does not amount to saying that it is provided by Universal Grammar, cf. Collins & Stabler, 2013), insofar as lexical production is not actually constrained by characteristics of particular roots: that is, any root is "nounizable" or "verbalizable":

(9) a. John dedicated a sonnet to Mary
    b. John sonneted Mary

While (9b) is not yet coined, it is perfectly parseable in roughly the sense of "John recitated Mary a sonnet"[7]. √ are always *types*. LIs are always *tokens* as they are interface readings of a specific syntactic configuration as depicted in **Definition 1''**.

[7] In Hale & Keyser (2002) terms, it would be a *locatum* structure, with incorporation of the root √sonnet. See Krivochen (2012: 119, ff. for discussion and details).

Theorem 3: (universality of procedural instructions) For all NL, and for the set of *all procedural elements* $P = \{P_1, \ldots, P_n\}$, it is the case that $NL \ni P$.

Corollary: Given a *specific* structured set $p = \{P_1 \{P_2 \{\ldots P_n\}\}\}$, it may be the case that $NL_X \not\ni p$.

This amounts to saying that, even if the procedural instructions Path and Place are available for all languages, it might be the case that a specific NL does not lexicalize (i.e., coin and materialize) both nodes if they belong to a single terminal. This assumption is fundamental, we believe, for comparative approaches, since it avoids the fallacy of Sapir-Whorf hypothesis while allowing the researcher to account for language specificity at PF without the need for Parameters (Cf. Chomsky, 1986 and much related work). For example, English [into] comprises both instructions, whereas Spanish [dentro] conveys mainly Place. A similar view (with respect to cartographical approaches) is hold by Sigurdsson (2013), even though he sets the focus on the universality of the functional skeleton (not on particular bundles of instructions, like event-cause; path-place; tense-aspect-modality; etc.). The interesting thesis we share with Sigurdsson is that universality does not entail belonging to UG, which we take to be a desirable conclusion.

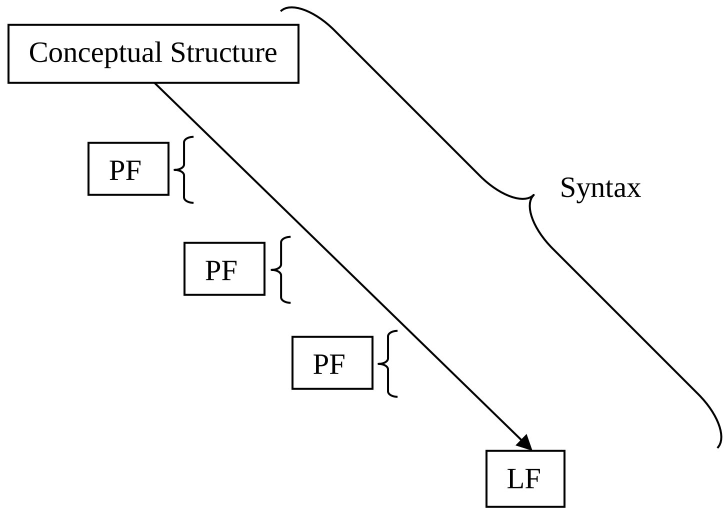
Definition 12: A *sentence* is an epiphenomenal result of *concatenation* of LIs between isodimensional Ws following interface requirements.

Definition 13: *format* is either *structural* or *ontological*. Ontological format is the nature of the elements involved in an operation, whereas structural format is the way in which these elements are organized.

On Operations:

The derivational mechanics we propose here follow from the architecture in Figure (6), where the beginning of a linguistic derivation is a pre-linguistic and independently existent conceptual structure CS (Jackendoff, 1987, 2002; Culicover & Jackendoff, 2005; also Uriagereka, 2008, 2012; Taylor, et. al., 2011) which, in our terms, embodies a speaker's *intention* (the *intentional* part of Chomsky's Conceptual-Intentional system). The aim of a linguistic derivation is to maintain all the information in the CS in a global level, while increasing the informational load in a local level (i.e., application after application of *concatenation*). Our architecture has the form of (10):

(10)



As it can be seen, the access to the ILs is cyclic (following the model of Uriagereka, 1999; see also Uriagereka, 2012 for discussion), and determined by interface conditions. We will come back to this below.

[5/2015: the architecture in (10) has been slightly revised in the light of Uriagereka's CLASH model and the concept of Dynamical Frustration. See 'A Frustrated Mind' for details]

Definition 14: *Select* instantiates a *type* in a W_X following **Principle 2**.

Definition 15: *Merge* concatenates LI-tokens in a W_X driven by the interfaces' constraint expressed in **Definition 11**.

Definition 16: *Analyze* evaluates the objects built via *Merge* in W_X in order to verify full interpretability in IL_X.

Definition 17: *Transfer* is the operation via which an Interface Level IL_X takes a fully interpretable object from W to proceed with further computations.

<u>Corollary:</u> if W_X interfaces with more than one IL, *Transfer* applies for each IL *separately*, all other things being equal

Defintion 18, as well as its corollary, constitutes the only halting algorithm in our framework, this is what makes the derivation stop just as DFI is what makes it go on. Moreover, we have no need to stipulate derivational domains *a priori*, contrarily to Chomskyan *phases* (Chomsky, 1999, 2005 and much related work).

Definition 18: *Merge*, *Analyze* and *Transfer* are both interface-driven and interface required.

Definition 19: 15, 16, 17 and 18, occurring cyclically, determine the derivational dynamics in W_X.

In a nutshell, derivations would proceed as follows:

Concatenate (α, β) = {α, β}
$Analyze_{IL}$ {α, β} [is {α, β} fully interpretable by IL?]
(Transfer {α, β]} if $Analyze_{IL}$ results in convergence at IL)

Our claim is that the strict distinction between computation and representation should not hold, as it is not a *sine qua non* condition for the distinction *conceptual – procedural* to remain. If the conceptual or procedural character of a unit is determined at the interface, as a reading of the relations it has established in the generative workspace via *concatenate*, then the dynamics above –which correspond, we believe, to a strongly derivational system- hold all the same. Conceptual elements provide the substance, whereas procedural elements provide instructions for interpretation in the form of relations that hold between concepts.

Definition 20: *category recognition* requires X and √ to be in the same workspace at the point in the derivation in which category is read off. X and √ may, however, have entered the derivation in different but isodimensional Ws.

Categories, phases and other units are not primitives of the syntactic theory, but arise as a result of the interaction of a free Merge system with interface conditions: the dynamics of the derivation and the legibility conditions of certain interpretative mental faculties or any other computational module. (see Krivochen, 2012; De Belder, 2011, Boeckx, 2010; also work in Distributed Morphology like Marantz, 1997 and Fábregas, 2005 and Exo Skeletal Models, see Borer, 2009 among others). Moreover, there is no distinction between "lexical derivations" and "syntactic derivations", and this goes beyond positing a single generative mechanism: there are just derivations, regardless the nature of the elements that are manipulated, since the generative operation is blind.

<u>On Linguistic Derivations:</u>

Definition 21: a *Derivation* is the finite set of computational steps that define a Syntactic Object SO, projecting a *type-Array* onto a legible object for IL via *token-Merge*.

A note is in order here: the finite character of the linguistic derivation is determined by memory issues and computational capacity of interface systems, which is limited by the biological substrate. Computations *in abstracto* are not finite by principle, and do not need to be. As we will see below, the presence of a "halting algorithm" in the form of IC only applies when there is *generation-interpretation* interaction. If a function is not interpreted (say, for example, Peano's successor function), there is no need to stop the derivation at any point.

Definition 22: a syntactic object *Converges* if and only if all of its components are interpretable by the relevant IL/s. (see Chomsky's 1995 Full Interpretation Principle, of which this constitutes simply a formulation)

On Dependencies:

Definition 23: if α and β are interface-associated via their coordinates in W, there exists a *Dependency* between α and β.

Definition 24: if $\alpha \in W_X$ and $\beta \in W_Y$ and either $W_X \equiv W_Y$ or $W_X \not\equiv W_Y$ and α and β are defined by the same *n*-tuple of coordinates in their respective *isodimensional* Ws, α and β are *bound* and the *dependency* is called *co-referentiality*.

Definition 25: *Reference* is location of a symbolic object in the conceptual multi-dimensional space via LI's coordinates, which are interpretable by the conceptual system C-I.

Definition 26: *Dependencies* are read off in IL, not in W.

We beg the reader to bear these definitions in mind, since they will be essential for a theory of deictic reference in *n*-dimensional workspaces (as opposed to both realist and descriptivist theories). Definition 24 in particular should have interesting consequences when applied to Binding Theory, since "governing categories" or other pre-defined domains give the interfaces no information about the conceptual location of, say, an anaphor and its binder. In this sense, LDA in Latin and Germanic languages would fall naturally within this framework (see Krivochen, 2013b for discussion of LDA in Latin under this framework).

On Locality:

Definition 27: A *dependency* is *Local* if and only if there is no intervenient object γ (of arbitrary complexity) such that: (i) the relation between α and γ is equivalent to that between α and β for interface purposes (ii) α, β and γ belong to the same W and (iii) γ is structurally closer to α than β

This definition, combined with definition (24), has the consequence that *no dependencies can be established in a generative W,* but at Interface levels, which we have expressed in definition (26). This also means that if there is no interpretative system involved, there are *no dependencies*. For example, in the structures generated by a mathematical algorithm that are never interpreted (see example (47) below) no dependencies can be posited. As we will see, this provides a possible solution to one of Tegmark's objections to fully explicit mathematical structures regarding halting algorithms.

Definition 28: P_W is a *phase* in a W if and only if it is the minimal object fully interpretable in IL.

Definition 29: *Transfer* applies only to *phases*.

Once we have defined the units we will work with, the operations that apply, and the conditions under which those operations apply to certain symbolic objects[8], we will devote the rest of the paper to the consequences the adoption of our model would have for the theory of natural language structure. Our hope, however, is that the model hitherto devised is not limited to natural languages, but can also serve mathematical or physical purposes, since we do not limit the power of our generative algorithm to just one domain in the natural world. Let us now consider some central questions in formal linguistics under this light.

4. Some Applications in Formal Linguistics

4.1 Merge and Phrase Structure

This framework allows us to manage symbolic representations in a way in which we can best capture their essential property, *hierarchy*, while accounting for epiphenomena in terms of interface conditions. Let us first review the classic X-bar theory axioms, which characterized

[8] The mental entity of these objects is not at stake here. Either if the symbolic structures our system generates exist only in the mind, or as external objects (assume, for example, that a mathematical structure has entity independently of a perceiving mind), our system holds because it makes no prediction at this respect. Thus, the Postal-Chomsky controversy regarding realism or conceptualism is completely tangential to this proposal: it could work in a conceptualist theory, in a realist one or in a third party without falling in contradictions.

phrase structure in the Government and Binding model (Chomsky, 1981 et. seq.) and underlie most current models of phrase structure to different extents:

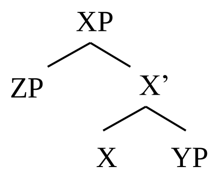
1. Endocentricity: every projection has a head

2. Projection: every head projects a phrase

3. Binary branching: every non-terminal node is binarily-branched

The kind of structures generated by this system is as follows:

(11)



These are bi-dimensional representations within the syntactic workspace, afterwards spelled-out. The 2-D character is determined by the necessity to generate "unambiguous paths" (Kayne, 1984) and c-command requirements for linearization, if linearization in fact follows the LCA (Kayne, 1994). It is commonly assumed that Spell-Out implies a "flattening" of the hierarchical structure onto a linear (Markovian) representation (Idsardi & Raimy, in press): our claim is that there is no *essential* difference between a tree-like representation and its linearized form[9]. Syntactic derivations have been claimed to be either *bottom-up* (Chomsky, 1995 and other orthodox works) or *top-down* (Zwart, 2009, in a different framework, Uriagereka, 1999), but always maintaining the underlying assumption that the *up-down* opposition is defining for syntactic processes: c-command is *top-down*, whereas m-command is *bottom-up*; Agree is also a *directional* operation (although there is no general agreement with regard to whether it is top-down or bottom-up, see Zeijlstra, 2011 and Putnam, Van Koopen & Dickers, 2011) and, given the fact that syntactic operations in orthodox Minimalist syntax are driven by the need to check features via Agree, directionality is centrally embedded within the theory. In orthodox accounts (Chomskyan minimalism, certain versions of HPSG, LFG) syntactic structures, then, are *bi-dimensional,* only differing from phonological representations in epiphenomenal characteristics, like *headedness* or

[9] Juan Uriagereka (2012) claims that monotonic Merge always derives Markovian objects, which, if true, would be a point in favor of our position.

binarity which are better expressed as semantic requirements over linguistic structures. If a true qualitative difference is to be found between syntax and phonological externalized structures, then the nature of syntax must be revisited, as we have done. In our system, there is a difference between *syntax* and *phonology* regarding the dimensions of each domain: syntax as a cognitive domain is, in principle, *n-dimensional*, and its conversion to phonology implies flattening the *n*-D structure into a Markovian object (Uriagereka, 2012), Turing-computable. It has been proposed (Uriagereka, 2008, 2012) that some structures, like adjuncts, already have a Markovian structure, and therefore they do not undergo this *dimensional flattening*, but it is not clear why, having more powerful tools at its disposition, the computational component would choose Markovian configurations for some structures but not for others (see Lasnik, 2011 and Lasnik & Uriagereka, 2012 for more discussion). For the time being, and as a very provisional perspective (but see Krivochen, 2013b for discussion), we will assume that linguistic syntactic structure is not predominantly Markovian, and Spell-Out (i.e., Transfer to PF) is precisely the "Markovization" necessary to make a structure manipulable by the interface systems of sound and articulation (Sensory-Motor, in Chomsky's terminology).

The optimal scenario when it comes to transfer would be that in which interface relations are based upon *pre-existing relations*, namely, those created in the syntactic configuration. *Ceteris paribus*[10]*, the nodes are spelled out mirroring the interface relation they maintain with the root, from the closest to the most detached*. From this claim, we derive that procedural nodes always have a closer relation with the root than peripheral nodes like Agreement, as they generate categorial interpretations in the semantic interface (as we have seen in **Definition 1''**). Let us now assume that a language L allows for an X number of combinations of semantically interpretable dimensions (causativity, person, number, location, tense…) in terminal nodes, i.e., *morphemes*. That language has a Y number of Vocabulary Items to Spell Out those dimensions specified to some extent regarding their distribution. The feasible scenarios (but not the logically possible ones, which are more) are the following:

(12)

i) X = Y

ii) X > Y

[10] This ceteris paribus condition refers to the availability of VI.

Of (12 i) we would say it is a morphologically very rich language, and it is possible that there is a language in which this occurs. Of such a language we will say that it is *maximally specified*. The normal situation for any natural language is (ii), in which the number of Vocabulary Items is inferior to the number of terminal nodes. This situation is called *Underspecification* (Hale & Marantz, 1993). Note that the phonological interface SM cannot read terminal nodes, but phonological matrices, so Vocabulary Insertion is a step that we cannot miss. This operation puts a VI in a terminal node; in a 1-to-1 relation in the simplest cases (we will see that this is not always the case, which makes the syntax-morphophonology interface a very complex object). In order for a VI to be inserted into a terminal node, it must match *some* of the features of that terminal node, there being a competence among several possible VI as to which one is more specified. In this way, universal constraints are replaced, *prima facie*, by a simple *subset principle*:

> ***Subset Principle:*** *The phonological exponent of a Vocabulary Item is inserted into a position if the item matches all or a subset of the features specified in that position. Insertion does not take place if the Vocabulary Item contains features not present in the morpheme. Where several Vocabulary Items meet the conditions for insertion, the item matching the greatest number of features specified in the terminal morpheme must be chosen.* (Halle 1997: 128).

Our model can account for the syntax-phonology interface more elegantly than orthodox Minimalism, in terms of dimensional relations between one domain and the other, taking transfer to imply "Markovization" of syntactic structure. This way, we attempt to capture the true properties of the Spell-Out operation (Uriagereka, 2012) and establish a more general principle for Transfer operations:

(13) **Dimensional Impoverishment:**

Transfer ***may*** *imply the "flattening" of an X-dimensional structure to an X-n dimensional structure, while respecting Conservation Principle.*

An interesting note is that, given the architecture in (10), this should not be the case in the syntax-semantics interface: the CS must be computationally more complex than a regular language. Moreover, it is possible it is not even Turing-computable, for all we know. Problems of Spell-Out as dynamic Markovization of syntactic structure are dealt with in Uriagereka (2012); we have proposed a dynamical approach to the problem in terms of the Center Manifold theorem.

4.1.2 Some considerations about "Simplest Merge" and the Economy of Derivations and Representations

Our definition of *Merge* (definition 3 and corollary) in terms of *concatenation* entails the existence of a free and unbounded generative engine, which is available to any mental faculty that requires combination of atomic objects to form complex structures. Our hypothesis is that that very generative operation is available in the natural world, and is the principle of all complexity found in the Universe. If, as MUH entails, the Universe is not only mathematical but also simple (and describable by a finite and optimally reduced number of principles), the combinatory operation must follow not only from conceptual necessity but also economy principles. The notion of *simplest Merge* (Zwart, 2009) enters the scene. Zwart's formulation of Simplest Merge is as follows (2009: 2):

(14) <u>Simplicity required in the derivational procedure:</u>

a) Merge manipulates a single element of N [a Numeration] *at each step of the derivation*
b) Merge manipulates each element from N only once

However, the requirements do not follow from interface conditions, and are thus stipulations. It is also fundamental not to confuse economy in the theory with economy in the physical world. Even though Zwart's arguments in favor of his proposal have advantages over the version he presents of orthodox Minimalism (which is not quite accurate, to our understanding), the "requirement" for simplicity in the derivational procedure is stipulated in a syntacticocentric way. As a consequence, those principles have no interface relevance, and can be dismissed. Even if *Merge* is actually simpler applying to only *one* element, its interface value is null:

(15) Merge (α) = {α}
Merge ({α}) = {{α}}
Merge ({{α}}) = {{{α}}}

And so on. We see that this is a violation of our *Principle 2*, DFI. Economy in the theory must not be stated as a principle, but follow from third factor-like constraints, related to the architecture of the relevant object and the systems that read the output.

At this point, arguments for top-down (Zwart, 2009) and bottom-up (Chomsky, 1998, 1999, 2005) derivations are equally implausible, since they both assume what we will call "the 2-D fallacy", which implies that tree-like representations have mental reality (a fallacy best expressed in Kayne, 1994: for the Linear Correspondence Axiom, which maps c-command relations into precedence relations, to be a real linearization procedure, syntactic structure *must* be invariably 2-D and preferably monotonic)[11]. Our multi-dimensional geometry has made a strong case against 2-D syntax, and argued strongly in favor of *n*-dimensional Ws in which operations are constrained only by the interpretative interfaces. Our revised version of "simplest Merge" is formulated as follows:

(16) *n*-dimensional Merge:

a) Merge manipulates an x number of elements at each step, being x interface-determined. If {{α}} is taken to be trivial, as we do, then x > 1, being 2 the minimal non-trivial number of objects.

b) Merge manipulates each ***token*** *only once in a derivation* $\subset W_X$*, but each* ***type*** *as many times as needed*.

This definition of Merge allows us to dispense with Movement in terms of *displacement*, but reinterpret it as *token-Merge* and *dependency* establishment between *tokens* at the relevant IL, with basis on their coordinates in isodimensional W. As it allows multiple *n*-dimensional W to be working at once, our system provides a better account of both monotonic and non-monotonic merge than most orthodox theories, as we will explore in the next section.

4.1.3 Complex-unit Merge

Let us assume the following transitive sentence:

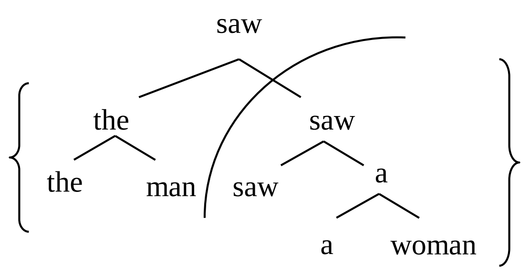
(17) The man saw a woman

[11] It seems that tree-like representations have problems not only in 2-D: Lasnik & Uriagereka (2005: 43) attempt to linearize a Calder mobile with LCA with no success (in our opinion), since a branch could c-command more than one terminal.

According to Uriagereka (1999, 2002), parallel derivational spaces are a conceptual necessity (Uriagereka, 1999: 121) when Merge applies to complex objects, already output of a Merge operation: for example, in the derivation of $[_{saw}$ the man saw a woman], $[_{the}$ the man] and $[_{saw}$ saw $[_{a}$ a woman]] are generated separately and then assembled externally. The derivation in his terms would proceed as follows[12]:

(18)

Derivation 1, *Merged* in W_1.



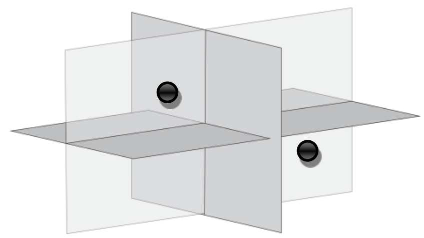
Derivation 2, *Merged* in W_2.

If the derivation occurred in just one space derivational space, then the units [the] and [man] would be merged to [saw a woman] sequentially (i.e., monotonically) and not as a constituent [the man], atomic for the purposes of further computations. That is, if we assume that [saw] would project a label indicating the properties of the complex object as a unit for the purposes of future computations, the final phrase marker would be the ill formed sequence: *$[_{saw}$ the $[_{saw}$ man $[_{saw}$ saw $[_{a}$ a woman]]]].

Coming back to our previous discussion, if an element is defined as a set of coordinates in a W, each set of coordinates depends on the number of dimensions in the relevant generative workspace, such that an element is to be defined by *all* of its coordinates in W. Given this scenario, let us see how an X phrase would be formed, say, a Determiner Phrase DP (assuming the simplest possible structure: {D, √}):

12 Nunes (2004) analyzes parasitic gaps using a mechanism very similar to the derivational Cascades model by Uriagereka (1999).

(19)



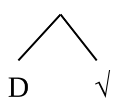
Workspace 1 (W1)

Both D and √ having the same ontological format (being both exclusively linguistic elements), Concatenate *can* (and thus *must*, otherwise we would have to come up with specific *ad hoc* conditions to restrict the operation at W) apply in the following form:

(20) Concatenate (D, √) = {(x, y, z) ⊂ W1, (x', y', z') ⊂ W1)}

The coordinates of the result of the operation (a DP, or {D}, construction) 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 (16):

(21)



Let us assume (21) is to be merged with an eventive structure (a Verb Phrase VP), say, to form (22):

(22) The man saw the book

Concatenation would then apply monotonically in a separate workspace but simultaneously:

(23) Concatenate (the, √book) = {(x, y, z) ⊂ W2, (x', y', z') ⊂ W2)

Concatenate ({the, book}, √saw) = {(x'', y'', z'') ⊂ W2, (x''', y''', z''') ⊂ W2}

The following derivational step takes both structurally identical units and merges them, generating the desired output.

Of course, more complex structures can be assembled, take (24) for instance:

(24) This is a problem John is tough to talk to about

The on-line generative processes in parallel workspaces generate the objects manipulated in (20) via successive applications of *concatenation*:

(25)

[[John is tough to talk to John] [about this problem]]

W_3

[John is tough to talk to John] W_1 → ← [about this problem] W_2

So, both objects are unified in a W_3 (which, for the purposes of the present discussion, could be identical to either W_1 or W_2). The complex object generated in (25) is in turn merged with the VP object in (26):

(26)

[This is a problem [[John is tough to talk to John] [about this problem]]]

W_3

[[John is tough to talk to John] [about this problem]] W_1 → ← [this is a problem] W_2

Now we have assembled the full object in (26) in the syntactic workspace, and C-I can interpret the object with no inconveniencies. For discussion about the Spell-Out of the tokens of [John] and [this problem] involved, see Nunes (2004) for a token-free, copy-oriented perspective and Krivochen (2013) for a type-token explanation.

4.1.4 How do we derive?

In this section we will analyze the functioning of our framework in linguistic derivations, so as to make differences with orthodox approaches clear for the reader. The generator function we have defined in **Definition 3** dispenses with featural requirements for Merge, such as Pesetsky & Torrego's (2007) *Vehicle Requirement on Merge* or Wurmbrand's (2013) *Merge Condition*. Free Merge has already been argued for in, for example, Chomsky (2004) and Boeckx (2010), but in a different venue (and within different intra-theoretical limits). Moreover, our derivational model is closer to that proposed by Putnam (2010) and Stroik & Putnam (2013) insofar as it is only interface-constrained: as we will see below, we propose an extremely local evaluation mechanism (OT's EVAL) *Analyze*$_{IL}$ (that is, *Analyze* from the interface level IL) that applies after each instance of *Concatenate* and determines whether the object is fully legible by the relevant interface. Echoing Boeckx (2007), *Analyze* can be seen as the interpretative systems (whichever they turn out to be) "peering into the syntax", and taking the minimal unit they can read, not as an artificial addition but as a natural consequence of having an impoverished generative operation applying in every faculty: the product of each application of the operation is accessible to the IL insofar as it is triggered by interface needs (otherwise, features and extra elements are difficult to elude). Merge (*concatenation*), as we take it, is a completely free operation that can apply as long as the objects to which it applies have the same format, motivated by interface conditions (this is, {α} is trivial at the interface levels, while {α, β} is not). In the linguistic W (bear in mind that *linguistic* does not assume any differences as to W, but merely as to the interfaces it interacts with), we have atomic elements (either roots or procedural elements), and we can say that they have the same format since they share a nature, they are linguistic instantiations of elements that, *per se*, are not manipulable by $C_{(HL)}$. The only attribute of Merge would be putting things together, without any restriction by principle as regards the nature or number of objects, since it would be a stipulation. Let us remind the reader of the derivational dynamics we propose:

(27) Concatenate (α, β) = {α, β}
Analyze$_{IL}$ {α, β} [is {α, β} fully interpretable by IL?]
(Transfer {α, β]} if Analyze$_{IL}$ results in convergence at IL)

If generation is restricted to a single operation *concatenate*, which is the optimal scenario (since there are no *a priori* Agreement restrictions, Cf. Pesetsky & Torrego 2004) and it occurs in an *n*-dimensional workspace (another null hypothesis: restricting dimensions to 2, as in traditional Kynean tree diagrams, is stipulative, derived from considerations of "unambiguous paths" and LCA, as we have mentioned earlier), it is only natural that the operation itself cannot *read* or evaluate what it has built. On the other hand, it is also only natural that the EVAL function is not separated from the interfaces but in itself be the set of Bare Output Conditions that each interface has. Then, if we depart from considerations of computational simplicity like "maintain as few structure at once in W", that is, "transfer as soon as you can" (bearing an obvious resemblance to Pesetsky's 1995 *Earliness Principle*), this "as soon as you can" is determined not by internal syntactic conditions (as they do not exist in our proposal), but by *Analyze*. In this way, the generative workspace is wiped clean several times along a derivation thus liberating working memory without the concomitant problems of defining, for example, endocentric transfer domains (i.e., chomskyan *phases*).

Having a set S of units S = {α, β, γ…}, *concatenation* allows unbounded *n*-ary combination, binarity being required for interface purposes, among them, *label recognition* (as described in Gallego, 2010): sticking to a strict version of the Inclusiveness Condition, we reject labels *in the syntax* because, if the syntactic component is purely generative (that is, strictly blind to the characteristics of the objects it manipulates), they fulfill no role. At the Conceptual Intentional interface, however, the situation is very different. "Labels", symbolic markers which indicate how a structural unit is to be interpreted for the purpose of further computation, are read off a structure, following Hale & Keyser's (1997: 40) claim that "if we knew the meaning, we would know the structure, perforce, because we know the meaning *from* the structure". The label of a syntactic object, that is, the instructions as to how that object is to be manipulated for the purposes of further computations after Transfer, is then *recognized* from a construal (crucially, not *created*, since it would violate the Inclusiveness Condition, a *faithfulness constraint* in OT terms) at the semantic interface, following considerations of legibility and convergence. However, if there is no *interpretation*, DFI does not apply (for example, when deriving mathematical structures, which have no meaning and therefore no halting algorithm, as Tegmark, 2007 points out, see also section 5 below), which is already a strong and interesting generalization about the functioning of the mind: free manipulation of elements in *n*-dimensional workspaces[13].

4.3 Features and derivations:

Orthodox Minimalism has posited that there are two "apparent imperfections" in the Faculty of Language: uninterpretable features and Movement, understood as literal displacement of constituents (Chomsky, 1995, 1998; cf. Krivochen & Kosta, 2013; Krivochen, 2013). Chomsky solves this apparent problem for the Strong Minimalist Thesis by saying that it is the need to

[13] See D'Espósito (2007) and particularly Baddeley (2007) for an overview of the importance of the notion of workspace (or 'working memory') for the study of the biology of language.

eliminate those features that are uninterpretable at the interfaces that triggers movement. Notice that, elegant though this theoretical move may seem, it presupposes (without providing evidence) the existence of:

a) A Faculty of Language
b) Features (which, in turn, imply both a *Dimension* and *n* number of *values*, or, in our terms, *outcomes*)
c) An operation to relate Features (i.e., Agree)
d) Displacement-as-(feature) Movement

All four represent a substantive complication for the theory, and imply the isolation of the study of language from the study of other mental faculties, since the "imperfections" are not architectural constraints of the mind-brain but intra-theoretical stipulations: Minimalism has *created* both a problem and its solution. Radical Minimalism is all about *eliminating* problems, and impoverishing the theoretical apparatus to the minimum, so that only conceptual necessity dictates what is required in the model, while maintaining empirical coverage. We have argued against the Case-as-a-feature system in previous works, now we will present a summary of our proposal. The reason we do this is because we will present a derivation driven by invasive interfaces, not by the need to value or check formal features, and such a position must be justified. The basic claim underlying all Agree machinery, as we understand it, is "things establish relations to one another" (Zeijlstra, p.c.). If that is the intuition, our system is completely compatible with it. Let us first summarize what is needed in an Agree-based syntax (Chomsky, 1999; Pesetsky & Torrego, 2004, 2007):

a) Dimensions (e.g., [Person], [Number], [Case])

b) Values (e.g., [1], [Singular], [Nominative])

c) An unvalued instance of a dimension [D] that acts as a *probe*, searching for a valued counterpart

d) An operation to relate a valued and an unvalued instance of the same dimension

Given DFI, the orthodox claim that some features enter the derivation unvalued and are therefore uninterpretable (Chomsky, 1999) and must thus be valued in a probe-goal relation (this process driving the derivation) is not an option: as already pointed out by Epstein & Seely (2002), an

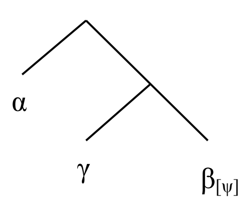
Agree-driven system requires the syntactic component to be sensitive to feature value. This implies that the computational component, which we have reduced to a single algorithm, is (at least) partly interpretative, since it must have access to inner substantive properties of the elements it manipulates, going beyond the identification of their format. In Krivochen (2011) we have argued in favor of a system in which [*val*-F] (that is, structures of the kind (value(Feature))) features are replaced by semantically interpretable dimensions comprising *in abstracto* all possible outcomes, what we have called "quantum dimensions" given this proposal's obvious parallel in Quantum Mechanics. The architecture we have outlined above points towards a theory in which there is no superfluous information in interface terms (i.e., no EPP, no formal Wh-, no undeletable Edge Features in so-called *phase heads*, see Chomsky, 2005), but only elements with a range of possible interpretations (as in our brief discussion of theta-roles above, expanded in Krivochen, 2012), the actual interpretation being determined only contextually *at the interface level*. Since Merge is blind, it cannot but put things together.

From an interface point of view, thus, *matching theory* (and its *valuation* alternative) *is unteneable* since formal features do not add anything informationally (that is, at the interfaces) as they must be *eliminated* before reaching them, and we therefore stick to the concept of *collapse* (analogous to its QM counterpart), which is nothing more than the act of *reading* a syntactic configuration:

23) *Collapse* is an interface reading of a *local dependency* in which the interpretation of an element is defined by another, structurally higher, whose distribution is specified enough.

In order for this to work, we need to assume that elements bear an inner potentiality for *category* and, in the case of *nominal constructions*, also Case in the sense defined above, as a purely semantically relevant dimension recognized at the interface level as a relation between symbols (occasionally Spelled-Out as a morpheme, epiphenomenally).
The tree-like 2-D structure would look like follows:

24)



There is a locality requirement in this licensing process, made explicit in **Definition 28**. Such a condition is a rephrasing of Relativized Minimality, but interface-based. It is not the "type" of element (A-position, A'-position, head) that determines its being intervenient in each case but the interface effects (i.e., "drastic effects on the output", in Grohmann's, 2003 terms) the possible configurations generate. Notice that there is an added condition on workspaces, which is absent in a polystratal theory that works with transformations upon phrase markers in a single workspace, as in transformational models of Chomskyan syntax. Thus, it would be accurate to classify our model as a "non-transformational" model of syntax: we posit no "hidden levels" of computation, nor representations that derive from one another via the application of a transformational rule (Chomsky, 1957).

Epistemologically, we have the advantage over theoretical physics that we have observable stretches of language where to test our hypothesis. So, let us take a ditransitive *Prepositional Indirect Object Construction* (PIOC):

(28) I$_{NOM}$ gave a book$_{ACC}$ to Mary$_{DAT}$
(Yo)$_{NOM}$ di un libro$_{ACC}$ a María$_{DAT}$
(Ego)$_{NOM}$ librum$_{ACC}$ Mariae$_{DAT}$ dedi

By looking at the same construction in other languages with rich casual morphology (like Latin, Sanskrit or Greek) it has been established that the {D} [I] has Nominative Case, [the book] has Accusative Case and {P} [to Mary] has Dative Case. But this is nothing more than a description, with no explanation as to why things are the way they are. We will not review the classic attempts of explanation (see Chomsky, 1981, 1995, among others), but go directly to our point.

The weak hypothesis would be that each nominal construction will bear a dimension [Case$_X$], whose value when entering the derivation will be a "ψ-function", or a complex vector[14] [$N\varphi + A\theta + D\lambda$]. That is, the quantum dimension will comprise all three possible values, as all three are possible states of the Case system for a particular DP, and since "syntax" is reduced to *concatenate* (a purely generative function), there is no problem in manipulating elements in their ψ-state as there is no "peeping into the box" as far as generation is concerned. However, [D$_X$] cannot be read by the interface levels, so the quantum dimension must "collapse" to one of the possible states. Here, it is not "measurement" but the reading off a configuration built by

[14] It has been pointed to us that this would imply that the mind, or at least some cognitive domains, is actually a vectorial field (but it could also be a tensorial field). We are currently doing research pursuing the consequences of this idea.

concatenation that does the work. Let us take the {P} structure as an example. We have the merger of {$to_{[TO]}$, $Mary_{[CaseX]}$} -and the subsequent "labeling" of the structure as {P} in Logical Form for semantic interpretability, since in a dynamic model interface-interpretation is in *real-time*-. At this point in the derivation, we are already in condition of collapsing the quantum dimension in [Mary], since we have a *procedural node* in a minimal configuration that can license that dimension in an XP (adapting the idea from Rizzi, 2004). This minimal configuration is defined in our terms *within phase boundaries*, following the definition given in **29**. Since the procedural information conveyed by P is essentially *locative*, the quantum case dimension on [Mary] will collapse to the *locative* sphere, i.e., *Dative*. A stronger hypothesis, perhaps more plausible, is that *the mere configuration is enough to license an interpretation*, without positing that there is a [D_X] beyond expositive purposes. The licensing takes place only if there is no closer functional / procedural head that can license the relevant interpretation, in order to respect Minimality. To account for this, we will draw another principle: ***Locality***, but as defined in **28** and **29**. Just like a particle cannot influence any other than its surrounding particles, within field theory, any object has a certain area of influence where it can license operations / interpretations if necessary. It is interesting that influence can be indirect, that is, a particle α may not be able to influence particle γ since particle β is "in the middle", but by influencing β, α will have an effect on γ, and thus we have a flexible definition of *Locality*, revisiting Rizzi's (2004, 2009) approach. That is what we call *compositionality*, in linguistics. We have to pay attention to the whole phase to understand, for example, the interpretation that a certain syntactic object receives at the interfaces.

4.4 Figure-Ground relations revisited:

Given the fact that figure-ground dynamics depend on the interaction between the prefrontal cortex and the temporal and parietal lobes respectively (Pylyshyn, 2007; Boeckx, 2010), our representation in terms of multiple workspaces geometrically defined working in parallel can provide further insight not only on language but on general perception and organization of phenomenological / perceptual information (see Krivochen & Miechowicz-Mathiasen, 2012 for argumentation and examples). Cognitive Linguistics has argued that entities are located within a conceptual space, which we have formally expressed as W. Our advantage is that the explicit case for multiple Ws implies optimization of computational efficiency, thus following from general economy principles. Regarding the specific case of Figure-Ground dynamics, which appears to be a fundamental opposition in the apprehension and organization of experience, the situation would be formally represented as follows:

(29) $W_X \cong W_Y \wedge \exists(x) \mid x \in W_X, x \in W_Y$

Representation in which there is a partial "overlapping" of spaces, an intersection, in set-theoretical terms. So, having a Figure *F*, and a Ground *G*, we need two workspaces to account for the biological basis of the complex structure:

(30)
$$F \in W_1$$
$$G \in W_2$$
$$\exists(x) \mid x = W_1 \cap W_2$$

In grammatical terms, *x* = *preposition* (P), as it is a procedural element that relates F and G in terms of *central* or *terminal* coincidence (Hale & Keyser, 1993, 1997, 2002; Mateu Fontanals, 2002). We now have a *logical* definition of P.

This definition allows us to establish some conditions on predication, assuming the localist hypothesis that all predication is ultimately *locative*, and relates a *figure* and a *ground*. The constraint we would like to posit is the following:

(31) *Given two conceptual entities α and β and an unspecified locative procedural element p, the entities can enter a locative relation iff p = α ∩ β. If there is no intersection between α and β, no locative relation can be established.*

This is a semantic restriction upon interpretation, but *by no means* combination: relations are read off configurations, not the other way around. Therefore, the freedom of *concatenation* is maintained.

4.4.1 Deixis and topological spaces:

In this section we will devote our attention to how deictic units fall within the figure-ground dynamics we have argued in favor of above. Deictic elements are linguistic units whose referential behavior is traditionally said to vary according to elements of the context of situation, more specifically, the participants and the spatio-temporal context[15]. This characterization, which underlies the whole French tradition (e.g., Benveniste, 1966) and is partly based on Jakobson and Jespersen's work, draws a line between two kinds of nominal constructions: in Russellian terms, those that can be replaced by a definite description *salva veritate* (common names, quantified expressions) and those that cannot[16]. Such a characterization introduces an asymmetry in the linguistic system that is not desirable, if a general theory is an alternative. However, the semantic particularities of deictic items were completely overlooked by Generative Grammar, after the interest that had arisen within structuralism due to Jakobson's works. Since to this day there is not a mainstream, widely accepted Generative theory of reference (but see Diesing, 1992; Heim, 2004 and Jackendoff, 2002 for non-orthodox approaches), deixis is "no man's land" for formal studies. Our perspective is completely different from those that have been mentioned: we will try to formalize deixis mathematically, in an attempt to show that there is no fundamental difference between structures with deictic elements and without them, when it comes to syntax and, even, semantics.

To this end, deictic units are taken here to be *vectors* (see note 14 above), whose elements are the same that are commonly assumed in physics and mathematics:

a) <u>Point of Origin</u>: array of coordinates {x, y, z,...n} in an *n*-dimensional space such that it is the starting point of the vector.
b) <u>Module</u>: the so-called "intensity" of the vector, the length of the segment that defines it.
c) <u>Direction</u> (or "line of action"): the straight line in which the vector is inscribed.
d) <u>Sense</u>: indicates to which side of the line of action the vector points. It is represented by an arrowhead.

We consider that the unit lexically encodes only *one* of the aforementioned elements: the *module*. Take, for example, the following opposition:

---

15 We are taking only this definition, not even considering those that claim that the semantics of deictic elements change because it is simply impossible: if the semantics change, there is no possible reference. However, if the semantics are underspecified enough (but stable), referential changes are perfectly possible.

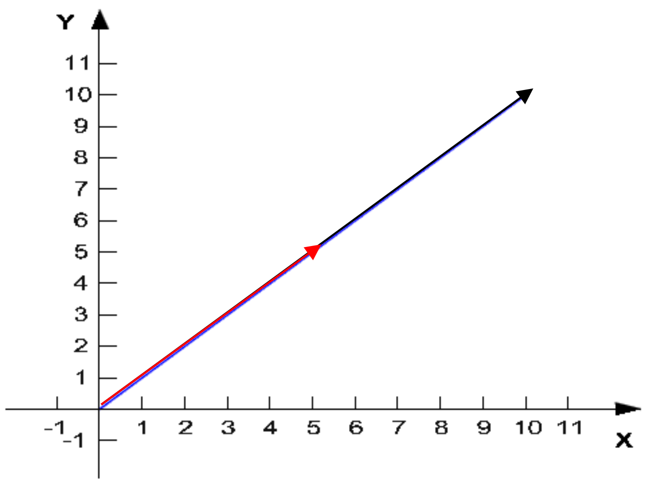
16 The first versions of Russell's theory of Definite Descriptions (1905) grouped demonstratives and personal deictic elements as "logically proper names", whose meaning is in itself the referred object. Nevertheless, in later writings Russell denied the personal deictic units the status of logically proper names and reserved the label for demonstratives. In any case, our theory makes no claim about the descriptionalism / direct reference debate, since we simply do not work with the same categories. Moreover, we crucially do not have any interest on the "truth conditions" question, since it falls out of the scope of linguistics altogether.

(32) *This-That*

Here-There

Now-Then

(33)



Our hypothesis will be the following: deictic units have an invariant *locative* nature, the only variants being the kind of information that has to be retrieved. In any case, the relation between a deictic item and the element it modifies is such that the deictic item *anchors* the reference of the modified element, which is semantically underspecified, in a (conceptual) spatial continuum. Accordingly, the simplest possible structure for any referential expression is the following:

(34) $(e_X, a_Y \ldots x_N)$

This is, an ordinate pair including an *entity* and an *anchor*, which we identify with either D or T[17]. In any case, the examples in (27) are *unaccusative* structures, where the {event} domain denotes a generic *stative atelic event*, categorial interpretation at the C-I interface being determined by the

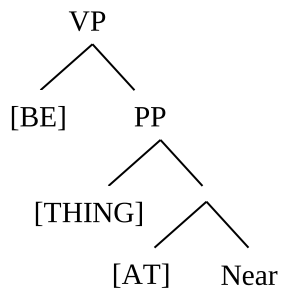
17 The motivation for the inclusion of an xN variable is straightforward: in an n-dimensional workspace, we cannot stipulate a priori how many coordinates will be needed for identification in particular cases. It is possible, and currently under research, that aY is in itself complex, thus we are left with a representation like the following:

i) (eX , (pY, qZ))

In (i), the second coordinate is complex. This may be required to formalize different localist systems, for example, left-to-right and bottom-up models for the numerical capacity.

local Merge of D or T. Roughly, the categorically underspecified structure (i.e., before the merger of D) would be as follows:

(35)



However, it would be inaccurate to pretend that the reference of a deictic item is strictly a point, since, crucially; their referential scope can either *widen* or *narrow*. See, for example, (36-38):

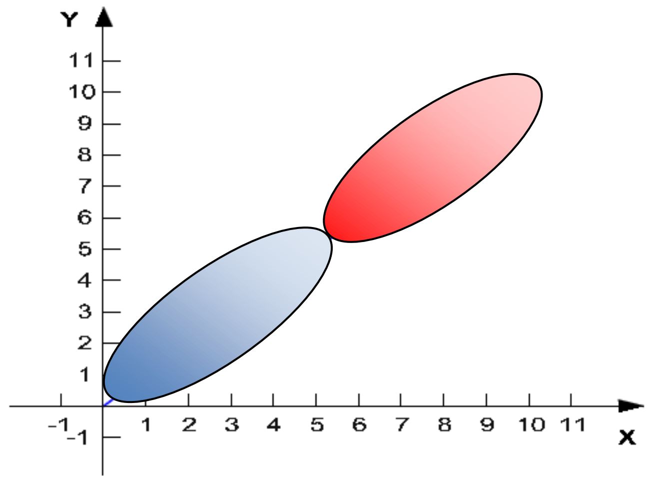
(36) It is very easy to have internet access *now* (Ref: past 15 years including utterance moment)

(37) There are no more dinosaurs *now* (Ref: last 60 million years)

(38) Do it *now* (Ref: a T immediate *after* utterance)

How can we explain these facts with our geometrical model of *deictics-as-vectors*? The solution comes from the mathematical concept of *topological space*. This means that the referent of a DP containing a deictic item (or, more generally, any DP) is not exactly a point, but a potentiality, just like it would be in Heisenberg's theory. A *root*, underspecified though it might be, determines a topological area in the conceptual space, possibly purely mathematical through resources like general memory capacities. A "functional" layer restricts that space, but the potentiality we mentioned above is built within the system. The direct consequence of the adoption of such a theory (i.e., the *uncertainty principle*) is that no amount of further specification, for example, via recursive use of relative clauses, can provide a mathematically exact coordinate. Therefore, we must reformulate (33) as (39):

(39)



Therefore, we say that a structure containing an element like “here” is to be conceptually located within the blue area, without the possibility of having absolute certainty about the coordinate matrix in the *n*-dimensional space (see Krivochen & Miechowicz-Mathiasen, 2012 for details). Rather than points *at which* we find the system, we have attractors *towards which* the system tends. Each step in a generative procedure restricts the relevant phase space in a different way, thus making it easier for the interpretative components to locate the relevant information.

A question at this point is whether our model applies to personal deixis, since *a priori* it seems to be more difficult to express it in locative terms. The fact that temporal deixis is subsumed to locative deixis is clear, but personal deixis somehow seems to resist the locative theory. Nevertheless, the so-called “projection” cases (when person features do not match the referent) can help us in our inquiry:

(40) A. (Doctor to patient) How are *we* feeling today?

B. Cómo nos senti*mos* hoy? (Spanish)

C. (Waiter to customer) Does *the sir* need anything else?

D. Necesita algo más *el señor*? (Spanish)

In these cases, we see that there is a displacement of the relevant coordinates. In (A-B), the topological space that is supposed to include the coordinates of both the speaker and the hearer is inferentially *narrowed* to include only the hearer, whereas in (C-D) the topological space that is supposed to include a third-person party, neither the speaker nor the hearer, is simply *shifted*[18] to the hearer's coordinates (there is no *widening* because the third person is not included as a referent, but *only* the second). Can our model account for these facts? Yes, but not without the addition of some extra assumptions about the syntactic structure of constructions including deictics.

There is a curious difference between English and Spanish regarding ellipsis in certain contexts:

(41)

a) I want those two *blue* toys. (pre-nominal Adj)
b) Quiero esos dos juguetes *azules*. (post-nominal Adj)

Let us ask the question "Which toys did you say you want?" (or something of the sort). The answer could be (d) in Spanish, but (c) is banned in English:

(42)

c) * I want those two blue Ø.
d) Quiero esos dos Ø azules.

Is there an inter-linguistic difference regarding the relative position of the root and other nodes (Num, Deg, Gen, etc.) within the {D} structure? Certainly, that would *not* be the optimal scenario, as it would require positing some sort of "*parameter*", by all means an undesirable scenario in an eliminative framework like our own. Besides, linearization via LCA (Kayne, 1994) does not work in an *n*-D model of syntax like the one we propose, as we have seen. We can explain that post-N Adj act in Spanish as *abridged restrictive relative clauses* (ARRC), whereas pre-N Adj are just qualifying Adj, that do not restrict reference[19]. Thus:

(43)

e) (? In my variety, but it depends on the register) *Azules* juguetes "blue toys" (the set of relevant blue things and the set of relevant toys are identical)

18 It is crucial to notice that our use of the word "shift" does not convey the same meaning as in Jakobson (1990) as a unit via which the code refers to the message.

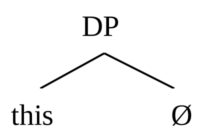
19 Interestingly, Czech exhibits the same alternance regarding the position of the Adj as Spanish, but it is an exception among Slavic languages (Peter Kosta, p.c.).

f) Juguetes *azules* “toys blue” (of the whole set of existent toys, just the blue ones: juguetes [que son] azules)

ARRC are enough to restrict the reference of the phonologically null root, so an explicature can be built. English ARRC are commonly PPs or Present Participle non-finite clauses, informationally heavy structures that go at the end of the nominal construction. Thus, pre-N (Num, Deg, Quality, etc.) elements cannot assure C-I convergence / legibility, as they *qualify* but do not restrict enough for C-I to identify a referent.

Coming back to demonstratives, their *pronominal* use, as we have analyzed it, show a structure in which all the information is conserved. Consider what would happen if we had posited something like:

(44)



We would have a null element merged with the bundle of vectorial dimensions: *initial point* (0:0), *sense*, *magnitude* and *direction*, which compositionally with Num_0 and Gen_0 give us *this, that, these, those*. In Spanish, Gender plays a role along with these dimensions, giving *esto/s, esta/s, eso/s, esa/s, aquello/s, aquélla/s* (compare with Latin *is/ea/id, iste/a/ud, ille/a/ud*). This configuration, however, reminds that of De Belder & van Craenenbroeck (2011), which we have criticized: if we accept that there is a RSS underlying these kind of elements, then, erasing all trace of the root would be a violation of the *Conservation Principle*. We return to our previous thesis: instead of [Ø] we have a generic root √THING, whose Spell-Out is *irrelevant* unless further specification is provided[20]. The presence of a root in these constructions is essential, since if there is a root, then there is malleable content, following Escandell Vidal & Leonetti (2011: 4):

> “(…) *In the cognitive pragmatic tradition, it is common to assume that conceptual representations are flexible and malleable, which means that they can be enriched, elaborated on and adjusted in different ways to meet the expectations of relevance. All the interpretive phenomena that are usually considered as instances of meaning modulation and ad hoc concept formation stem from this basic property (Wilson 2003,*

20 This proposal is reminiscent of that of Panagiotidis (2002), but the foundations of each approach differ greatly.

Wilson and Carston 2007). We claim that instructions, on the contrary, are rigid[21]*: they cannot enter into the mutual adjustment processes, nor can they be modulated to comply with the requirements of conceptual representations, either linguistically communicated or not.* (…)"

The variations in the "size" of the topological space and the processes of narrowing, widening or shifting of the reference can be only accounted for if we assume the presence of a malleable element within the structure. These considerations leave us with a desirable result: the syntax of deixis is *always locative*. Moreover, we have a proposal for the syntax of the minimal referential structure:

(45) *p* √THING

In this representation, *p* is a procedural node (say, D or T), conveying vectorial procedural instructions, that anchors the reference of the generic root (the *figure* above) and makes it interpretable by the post-syntactic semantic parser, licensing the "referent assignment" process in the construction of a full propositional form or *explicature* (Wilson & Sperber, 2003). ***The root determines a certain topographic space within our conceptual system, which the procedural node delimits and specifies***. Moreover, as Radical Minimalism works with a free-generator, our minimal proposal does not exclude cartographic approaches, but only include the relevant nodes when interface conditions require so. Thus, we do not have a fixed functional skeleton for syntactic structures, but nodes that are freely-merged following C-I requirements.

4.5 Two different kinds of systems:

We would like to make a distinction that we consider essential when building a theory about the mind and analogous systems: the distinction between *Generative systems* and *Interpretative systems*. This distinction is not only terminological, but has major consequences to the theory of Quantum Human Computer (Penrose & Hameroff, 2011; Penrose, 1997; Krivochen, 2011b, 2013b) since we will demonstrate that only *certain systems* allow elements in their *quantum state* (i.e., comprising all possible outcomes), which we will call the *ψ-state*.

a) *Generative Systems*: Generation equals *Merge*, a free, unbounded, blind operation that takes elements sharing either *ontological* or *structural* format and puts them together.

21 Notice the parallel with early DM (Noyer, 1998): f-morphemes' Spell-Out was said to be deterministic", whereas l-morphemes' Spell-Out was free.

For example, the syntactic component, the arithmetical capacity, the musical capacity and the pre-syntactic instance of the conceptual-intentional system.

b) *Interpretative Systems*: Interface systems, they have to read structural configuration build up by generative systems. For example, SM and CI.

An essential difference is that, as Generative systems are blind to anything but format, they can manipulate objects in their ψ-state and transfer them to the interface systems (to which we will come back later). *Transfer and interpretation is analogous to Heisenberg's observation: a structural relation between an element in its ψ-state and a procedural element / logical unit with specific characteristics collapses the quantum state onto one of the possible outcomes.* In the syntactic working area W, elements enter in their ψ-state and are blindly manipulated, merged together and transferred to the interface levels, where structural configurations determine unambiguous interpretations in terms of outcomes. The interfaces can "peer into" the syntactic W, to make sure a syntactic object (i.e., *any symbolic representation*) is transferrable: this is what we call "*Analyze*". That is, *collapse occurs as a result of the generative-interpretative tension reinterpreted as a tension between quantum systems and unambiguity requirements of legibility*.

4.6 On "Format":

Some further clarification is required regarding the notion of "format": we have said that it is the only requisite for *n*-elements to enter a *concatenation* relation, which makes it an essential concept in our framework, as it dispenses with the need of features as triggers for the generative operation. Let us thus define the notion:

a) *Ontological format* refers to the nature of the entities involved. For example, Merge can apply to conceptual addresses (i.e., roots) and procedural elements because they are all *linguistic* instantiations of conceptual information. It is especially useful if we want to explain in simple terms why Merge cannot apply cross-modularly: ontological format is part of the legibility conditions of individual modules.

b) *Structural format*, on the other hand, refers to the way in which elements are organized, e.g., *monotonically* / *non-monotonically* (or, in other terms, Markovian / non-Markovian). Merge optimally operates with the smallest non-trivial number of objects. Needless to say, given the fact that ontological format is a necessary condition for Merge to apply (principled because of interface conditions, whatever module we want to consider), the resultant structures will always consist on formally identical objects.

The concept of structural format is of particular interest because it has been problematic for orthodox theories of computation and linearization (at least until Uriagereka's 1999 MSO model). In the case of non-monotonic Merge, the elements that enter the computational relation share *structural format* (and, by transitivity, their sub-members share *ontological format*). The distinction to be done is not thus on Merge types (Cf. Citko, 2005, for example) but on the relevant format for each "type": successive manipulation of *ontologically identical* elements is called *monotonic merge*, whereas the introduction of a *structurally identical* unit into a derivation to be merged to the root (provided that the root is not an unary term) is called *non-monotonic merge*, with concomitant transfer consequences, detailed by Uriagereka (1999, 2002, 2012).

5. CONCLUSION

In this work we have formalized the framework we have called "Radical Minimalism", which is the result of applying the principles of substantive and methodological parsimony to the maximum departing from the assumption that Language is an epiphenomenon of the interaction between a generative system that is in principle the same for the whole "physical world" and an *n* number of interpretative systems, which impose constraints on the generator via *legibility conditions* (the characteristics that an object must have to be computable). In order to have a fully explicit model of the theory we have attempted to derive "syntax" in a very narrow sense (understood as the formal characteristics of natural languages) from conceptual necessity in a wider mathematical model, with concomitant effects on physics. We have tried to account for natural language as a physical system adopting SRMT and a strong version of MUH, which we understand as complementary hypotheses. Our ultimate goal is to simplify communication between different disciplines, by formalizing linguistic syntactic theory in such a way that it can be understood and enriched by mathematics and physics. To summarize, we will present some advantages of the Radically Minimalist model we have outlined so far:

a. Provides an account of multiple workspaces for parallel derivations (Chomsky, 1998; Uriagereka, 1999, 2002, 2012)
b. Deeper explanatory adequacy, as we take into account both the biological and computational implications of our claims with the perspective of the interface systems.

Includes biological and physical aspects in the formalization, being thus comparable to other models for natural objects (Cf. Collins & Stabler, to appear, who formalize only orthodox Chomskyan minimalism and remain within the scope of his proposals).

c. Generative systems are unified under a simple and fully explicit definition of *concatenation*: there is no *Faculty of Language* in the traditional sense, mental architecture becomes simpler and so does its formalization.

d. Only conceptually needed operations, optimally, just *Merge* and *Transfer*. Thus, relations R in a mathematical structure are only two, making a theory of the formal representation of linguistic knowledge in the mind-brain in relation to wider mathematical knowledge much more attainable[22].

6. REFERENCES

Amari, S. (2003) Learning and Statistical Inference. In Arbib (2003). 624-628.

Arbib, M. (2003) *The Handbook of Brain Theory and Neural Networks*. MIT Press.

Baddeley, A.D.: (1986) *Working Memory*. Oxford. OUP.

(1992) "Working Memory". In *Science* (255). 556-559.

(2007). *Working Memory, Thought, and Action*. Oxford: Oxford University Press.

Benveniste, E. (1966) *Problèmes de linguistique générale*, 1. Paris: Gallimard

Chomsky, N. (1965) *Aspectos de la Teoría de la Sintaxis*. Barcelona, Gedisa.

(1981) *Lectures on Government and Binding*. Dordrecht, Foris.

(1982) *Some Concepts and Consequences of the Theory of Government and Binding*. MA. MIT Press.

(1995) *The Minimalist Program* Cambridge, MA. MIT press.

[22] Tegmark (2007: 27) claims that "*A mathematical structure typically has a countably infinite number of relations*". We have provided principled explanations for halting algorithms and reduced the possible relations to two. Whether this model applies successfully to all cases, only time can tell.

(1998) Minimalist Inquiries: The Framework. *MITWPL 15*.

(1999) Derivation by Phase. *MITWPL 19*.

(2005) *On Phases*. Ms. MIT.

Collins, C. & E. Stabler (to appear) "A Formalization of Minimalist Syntax". To appear in Syntax. Retrieved from http://ling.auf.net/lingbuzz/001691

Culicover, P. & R. Jackendoff (2005) *Simpler Syntax*. Oxford: OUP.

D'Espósito, M. (2007) "From Cognitive to Neural Models of Working Memory". *Phil. Trans. R. Soc. B* 2007 362, 761-772

Diesing, M. (1992) *Indefinites*. Cambridge, Mass.: MIT Press.

Euclid *Elementos de Geometría*. Libros I-IV. Translation and notes by María Luisa Puertas Castaños. Madrid, Gredos (1991).

Fodor, J. (1983). *The Modularity of Mind*. Cambridge, MA: MIT Press.

García Mayoraz, J. (1989) *Entropía / Lenguajes*. Buenos Aires, Hachette.

Gómez Urgellés, J (2010) *Cuando las rectas se vuelven curvas: Las geometrías no euclídeas*. Madrid, RBA.

Greenberg, M. (1980) *Euclidean and Non-Euclidean Geometry*. New York. Freeman.

Hale, K. & S. J. Keyser (1993) "On Argument Structure and the Lexical Expression of Syntactic Relations". In Hale y Keyser (1993) *The View from Building 20*. Cambridge, Mass. MIT Press. 53-109.

(1997) "On the complex nature of simple predicators". In Alex Alsina, Joan Bresnan, and Peter Sells, eds., *Complex predicates*. CSLI Lecture Notes 64. Stanford, Calif.: CSLI Publications. 29-65.

Hausser, R. (1999) *Foundations of Computational Linguistics. Man-Machine Communication in Natural Language*. Springer.

Heim, I. (2004) Lecture notes on Indexicality. MIT. Retrieved from http://stuff.mit.edu/afs/athena/course/24/24.954/OldFiles/www/files/ind_notes.html

Heisenberg, W. (1999) *Physics and Philosophy*, New York: Prometheus Books

Izard, V; P. Pica et. al. (2011) *Geometry as a Universal Mental Construction*. In "Space, Time and Number in the Brain", Chapter 19.

Jackendoff, R. (1977) *X-bar Syntax: a Study of Phrase Structure*. Cambridge, Mass. MIT press.

(1987) *Consciousness and the Computational Mind*. Cambridge, Mass.: MIT Press.

(2002) *Foundations of Language*. Oxford: Oxford University Press.

Jakobson, R. (1990) "Shifters and Verbal Categories". In *On Language*. (ed. Linda R. Waugh and Monique Monville-Burston). 1990. 386-392

Kosta, P. & D. Krivochen (2011) "Some (Radically Minimalist) Thoughts on Merge". Presented at the conference "Minimalist Program, Quo vadis?" Universität Potsdam. September, 2011.

Krivochen, D. (2011a) "An Introduction to Radical Minimalism I: on Merge and Agree". In *IBERIA*, Vol. 3 n° 2. 20-65.

(2011b) "The Quantum Human Computer Hypothesis and Radical Minimalism: A Brief Introduction to Quantum Linguistics". *International Journal of Language Studies* Vol. 5 n° 4. 87-108.

(2012) *The Syntax and Semantics of the Nominal Construction*. Potsdam Linguistic Investigations 8, Frankfurt a. M., Peter Lang.

(2013a) "Tokens vs. Copies: Displacement Revisited". Under review. http://ling.auf.net/lingbuzz/001796

(2013b) On the Necessity of Mixed models: Dynamical Frustrations in the Mind. Ms. Under review. http://ling.auf.net/lingbuzz/001838

(in preparation) Unlimited Syntax. Ms. UNLP.

Krivochen, D. & P. Kosta (2013) *Eliminating Empty Categories*. Potsdam Linguistic Investigations 11, Frankfurt a.M., Peter Lang.

Krivochen, D. & K. Miechowicz-Mathiasen (2012) *Numerals, Numbers and Cognition: Towards a Localist Theory of the Mind*. Poster presented at *Cognitive Modules and Interfaces*, international workshop at SISSA, Trieste, Italy.

Lasnik, H. (2011) "What Kind of Computing Device is the Human Language Faculty?". In Di Sciullo, A-M. & C. Boeckx (Eds.) *The Biolinguistic Enterprise*. Oxford: OUP. 354-65.

Lasnik, H. & J. Uriagereka (2012) "Structure". In R. Kempson, T. Fernando, and N. Asher (eds.) *Handbook of Philosophy of Science Volume 14: Philosophy of Linguistics*. Elsevier, pp. 33-61.

Leonetti, M. y Escandell, M. V. (2004) "Semántica conceptual, semántica procedimental", In *Actas del V congreso de Lingüística General*. Madrid, Arco.

Leung, T. (2010) On the mathematical foundations of crash-proof grammars. In Putnam, M. (ed.) *Exploring Crash Proof Grammars*. Amsterdam, John Benjamins. 213-243.

Panagiotidis, P. (2002) *Pronouns, clitics and empty nouns. (Linguistik Aktuell 46).* Amsterdam & Philadelphia: John Benjamins.

Philosophy of Quantum Mechanics MIT course Spring 2005. Available at http://ocw.mit.edu/courses/linguistics-and-philosophy/ Downloaded on 16/7/2008

Poeppel, D., and Embick, D. (2006). 'The Relation Between Linguistics and Neuroscience', in A. Cutler (ed.), Twenty-first Century Psycholinguistics: Four Cornerstones. Mahwah: Lawrence Erlbaum, 173–89.

Putnam, M., M. Dieckers & M. Van Koopen (2011) The Directionality of Agree: Can a Universal Feature Valuation Operation be Established?. Presented in the conference The Minimalist Program: Quo Vadis? – Newborn, Reborn, or Stillborn? Potsdam, Germany - October 3rd -6th 2011. http://www.hum.uu.nl/medewerkers/j.m.vankoppen/handouts/potsdam%202011.pdf

Rasmussen, S., Karampurwala, H., Vaidshanath, R., Jensen, K., and Hameroff, S. (1990) "Computational Connectionism within Neurons". *Physica D: Nonlinear Phenomena* 42 (1–3): 428–449

Schrödinger, E. (1935) "The Present Situation in Quantum Mechanics". In *Proceedings of the American Philosophical Society*, 124.

Searle, J. (1980) “Minds, Brains and Programs”. In *Behavioral and Brain Sciences* 3 (3): 417–457

Smith, E. (1999) “Working Memory”. In *MIT Encyclopedia of Cognitive Sciences*. Cambridge, Mass. MIT Press.

Stabler, E. (1997) “Derivational minimalism”. in C. Retoré, ed. *Logical Aspects of Computational Linguistics.* Springer, 1997, pages 68-95.

(2011) “Computational perspectives on minimalism”. in C. Boeckx, ed, *Oxford Handbook of Linguistic Minimalism*, pp. 617-642.

Sova, R. (2005) The sound-meaning relation in the Standard Theory of Transformational Grammar. In *Prague Bulletin of Mathematical Linguistics*, 84. 31-53.

Taylor, K.; B. Devereux & L. Tyler (2011) “Conceptual structure: Towards an integrated neurocognitive account”, *Language and Cognitive Processes,* 26(9) 1368-1401.

Tegmark, M. (2003) “Parallel Universes”. In *Scientific American Magazine*. May, 2003. 41-51.

(2007) “The Mathematical Universe”. In *Foundations of Physics*. http://arxiv.org/pdf/0704.0646v2.pdf.

Uriagereka, J. (1999) “Multiple Spell-Out”. In *Working Minimalism*, S. Epstein, and N. Hornstein, (eds.) Cambridge, MA.: MIT Press

(2002a) “Multiple Spell-Out”. In Uriagereka, J. (2002) *Derivations: Exploring the Dynamics of Syntax*. London, Routledge.

(2002b) “Warps”. In Uriagereka, J. (2002).

(2012) *Spell Out and the Minimalist Program*. Oxford: OUP.

Vendler, Z. (1957) “Verbs and Times”, *Philosophical Review* 56 (1957). Linguistics in Philosophy (Ithaca, 1967).

Vitiello, G & W. Freeman (2007) “The dissipative quantum model of brain and laboratory observations”. *Electr. J. Theor. Phys.* **4**, 1-18 (2007)

Watumull, J. 2012. “A Turing program for linguistic theory”. *Biolinguistics,* 6(2): 222-245.

Wilson, D. & D. Sperber (2003) “Relevance Theory”. In Laurence Horn & Gregory Ward eds., *The Handbook of Pragmatics,* Blackwell, Oxford. 607-32

Zeijlstra, H. (2011) “There is only One Way to Agree”. Ms. University of Amsterdam.

Zwart, J-W (2009) “Prospects for Top-Down Derivations”. In *Catalan Journal of Linguistics* 8.