Prospects for a Radically Minimalist OT: Towards a learning algorithm[1]

Diego Krivochen, UNLP

e-mail: diegokrivochen@hotmail.com

Introduction:

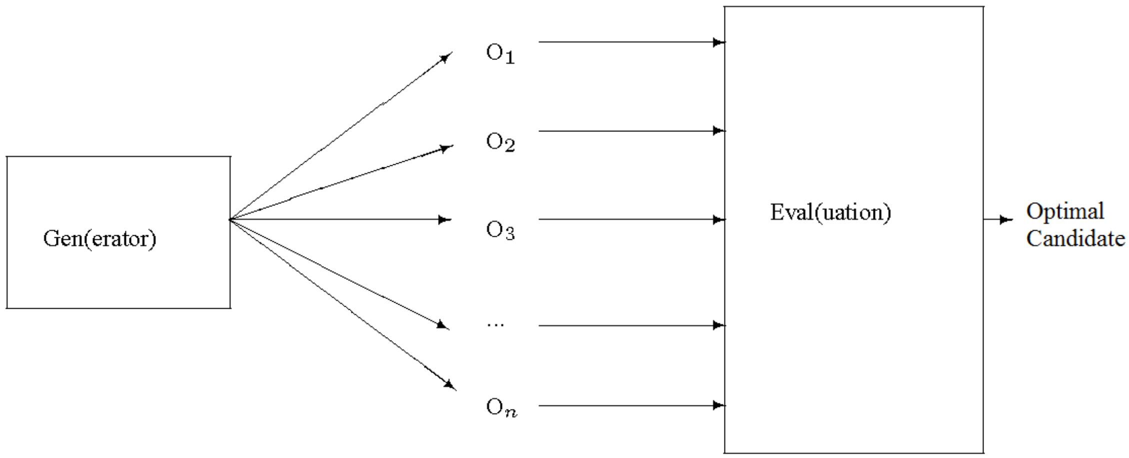
This paper has a simple and at the same time ambitious purpose: to open the door for mathematical learning algorithms in a Radically Minimalist crash-proof version of Optimality Theory. This objective will be pursued with Minimalism as a *program* -not as a theory- in its most radical version; and the theoretical substance will be provided by OT. A secondary but not less important aim of this paper is to show that mathematical formalizations of natural languages do not necessarily imply a "metaphor", but it is possible to work with the hypothesis that natural objects *are in themselves* mathematical structures (Tegmark, 2007). Such a formalization will be the first step in order to allow a more fluent interdisciplinary scientific exchange with formal sciences.

On the Nature and Form of Constraints:

Optimality Theory (Prince & Smolensky, 1993/2004; Kager, 1999; Barbosa, et. al., 1998) is a theory of the architecture of the mental grammar that is based on the idea that: (i) there are conditions upon output representations, (ii) those conditions (or "constraints", as they are commonly referred to) are violable to a certain extent, and (iii) those conditions are not universally articulated into "modules" (as in the GB theory) but organized in sets and following a ranking that depends on the relevant language (which accounts for language diversity in areas so different like phonology and syntax). Those constraints were first developed for phonology, but the theory has been expanded to all other domains of the grammar. The architecture that results from such a theory has the following form (adapted from Müller, 2011b):

[1] To the extent that a paper can be dedicated, I would like to dedicate this work to Michael T. Putnam, who helped me greatly in understanding the mechanisms behind variants of OT and also discussed with me many of the topics that I deal with here. Needless to say, any mistakes in interpretation are exclusively my responsibility.

(1)



The status of GEN is not quite controversial, should one assume we are dealing with a version of Minimalist *Merge* operation. Apart from anti-minimalist proposals like Pesetsky & Torrego's (2007), which require α to probe a feature on β to merge, free Merge proposals (Chomsky, 1995; Boeckx, 2010; Krivochen, 2012) are plausible and relatively accepted theories about the generative component. We will not focus our attention on *generation* but rather on *evaluation*. The form of *Merge* we will assume here is the simplest possible, one which does not entail labeling or any other theoretical complication (taken from Krivochen, 2012):

(2) "**Definition 3:** *concatenation* defines a *chain* of coordinates $\{(x, y, z\ldots n)W_X \ldots (x, y, z\ldots n)W_Y\ldots(x, y, z\ldots n)W_n\}$ 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."

Constraints, on the contrary, are quite problematic, from both a theoretical and an empirical point of view. There are three basic characteristics of OT-like constraints:

a) They are ***universal***
b) They are ***ranked***
c) They are ***violable***

(b) and (c) are claimed to be the main difference between *constraints* and *parameters*, since *universality* is a pretention of both OT and orthodox Minimalism. We will revisit this apparent opposition below.

Constraints can be classified according to the aspect of the relation input-output that it evaluates, which gives us the following classification (to be revised below):

d) *Faithfulness* constraints: include those constraints that prevent symbols to be deleted or added in the course of the derivation (DEP / MAX constraints), as well as ban operations that internally modify a syntactic object (IDENT constraints).

e) *Markedness* constraints: require that the output of GEN meets some structural well-formedness requirement (e.g., syllabic structure).

In this section we will directly present some problems with the current ontology of OT *constraints*, and then reflect upon possible ways to maintain the spirit of OT while solving those problems, in some cases, or directly preventing them from arise, in some others. General objections to the constraint-based OT-program as presented in Prince & Smolensky (2004) and Müller (2011b) include the following:

a) Where do constraints come from? Are they part of UG (as some claim)? How could OT manage a non-existence hypothesis, like RM and a strong version of Survive Minimalism, in which FL is nothing more than a dynamic (*n*-dimensional) workspace? How are these constraints really represented in the mind, from a biological point of view?

b) Let us assume there is a number X of constraints. Some of these, say, X_{-4} apply to a L1, and a different subset X_{-3} apply to L2. This is a serious problem, since there are constraints that could plausibly apply to a single language (which takes us back to EST rules). This critic could be avoided if one accepts the claim that constraints are *universal*, which in turn licenses a different critic: what is the difference between a *constraint* and a *parameter* (more on this below)? Moreover, how can L1 metalinguistically account for the set of constraints that do not apply to it? The development of a metalanguage is problematic (see Kempson's criticism of feature-composition semantics for a similar argument).

c) Regarding the ontology of the hierarchy, we find the following problem: let us assume that there is a hierarchy between constraints, so that they have the form $<<C_1\ldots> C_2\ldots>>$. Which is the principle ruling the hierarchy? In syntactic X-bar representations, we have the three axioms, mainly *endocentrism*. Here, there seems to be no ruling principle. If there is not, there is a fundamental problem with the very concept of hierarchy. But, what is worse, if there is, then there is an even bigger problem: X-bar phrase structure can be seen as a set-theoretical phrase structure (see specially Di Sciullo & Isac, 2008; Panagiotidis, 2010). If this is so, that is, if there is a set relation between constraints such that C_x cannot apply unless C_y, which is higher in the hierarchy, then the application of the most specific constraint can appear *instead* of long chains of constraints. This would work in the same way as simplification procedures express the complex feature composition [+ concrete], [+ animate], [+ human] simply as [+ human], since it presupposes (as it requires) the other two[2]. If there is a true hierarchy, things should work this way, and many constraints could be eliminated in favor of the most specific ones, as the satisfaction of some constraints must be obtained in order for some others to arise in EVAL. But

[2] In formal terms, we can say that if $[F_1] >> \{[F2]\ldots[Fn]\}$ (a set of features / constraints), then $\{[F2]\ldots[Fn]\} = [F1]$, or simply create a rewrite rule that allows us to rewrite a set $S = \{[F2]\ldots[Fn]\}$ as [F1].

> the drawback would be that the more specific the constraint, the less likely it be universal. Consequently, *elimination of redundancy*, which would be desirable in any system, is apparently fatal for an OT evaluator.

Our position will be that OT-like constraints, and representational filters in general, are *inductively* abstracted in a speech community, thus having no theoretical *explanatory* entity but fulfilling *descriptive* adequacy requirements. Another important point is that no constraint system can be fixed: an *n* number of constraints cannot be static, in the sense that it is not likely that the same number of constraints is necessary in every point of brain maturation.

<u>Different kinds of constraints?</u>

In this section we will analyze a taxonomy that underlies most current syntactic research, and was made explicit in Müller (2011a): filters are sorted out according to "*varying degrees of complexity*". We will present each category and the corresponding objections, before giving our own transmodular definition of *complexity* and its importance in syntactic theory.

- *Derivational Constraints:* apply to Merge / Move operations.

We have already defined Merge in a mathematical way, such that it is not constrained by any "featural" condition, as it seems to be the optimal scenario (Cf. Pesetsky & Torrego, 2007). In other words:

> "***Merge*** *is a* **free unbounded operation** *that applies to an n-number of objects sharing format, either ontological or structural.*"
>
> Taken from Krivochen (2012: 5)

A constraint cannot possibly apply to an *operation*, but to its *output*: there is simply no way to evaluate or optimize an operation which is free by nature. The very same example quoted by Müller supports our point of view:

(3) <u>Subjacency Condition (Chomsky (1977)):</u>

In a structure α ... $[_{\beta}$ ... $[_{\gamma}$ ... δ ...] ...] ..., *movement of* δ *to* α *cannot apply if* β *and* γ *are* bounding nodes.

The very formulation of the constraint is crystal clear: a determined representation is not suitable as input for the application of a transformational-structure building rule (or, in more general terms, the establishment of a dependency). This is not a constraint over Move, but over the *input* of the operation.

Moreover, even if the aforementioned operation was somehow constrained (in a physical system X), a constraint could assess whether those constraints are met, which are different from the operation itself. Therefore, our position will be that *there are no so-called derivational constraints*. Constraints evaluate *objects*, and *derivations* are ***not*** objects, *representations* are.

- *Representational Constraints*: apply to an output operation.

Our objection here has to do with the number of constraints and the learning algorithm that is required. We will posit only *one* constraint in the mature state of the grammatical system, which applies in an extremely local way, being *Transfer* determined by interface conditions, which are *input* constraints (determined by *Interpretative Syetems)* rather than *output* constraints.

- *Global Constraints:* apply to a whole derivation.

These kind of constraints (e.g., (G)MLC) are simply unformulable in a *phase-driven framework* and the reason is simple: if constraints are said to apply in the syntax, and syntactic objects are transferred by phase, there is no possible way in which a "whole derivation" is present *at once* in the generative workspace. In any case, *global constraints* should apply *at the interfaces*, which is not what OT has proposed, to our knowledge.

- *Translocal Constraints*: apply to a set of competing output representations to pick out an optimal representation.

This is the classic OT-like constraint, which filters out suboptimal representations. Translocal constraints are traceable to Chomsky's (1957) meta-theoretical desideratum over linguistic theory: it must provide an *evaluation procedure for competing grammars*. In those early days, the criterion was *simplicity*, defined in terms of "*the set of formal properties of grammars that we shall consider in choosing among them*" (Chomsky, 1957: 53). We will redefine *simplicity* for symbolic representations below, as a complement to this more general definition of simplicity as a meta-theoretical requirement. Now, in OT there is no such global criterion, which could very well work to rule the constraint hierarchy, and this undermines seriously the very concept of hierarchy. Constraints are ranked according to each paper/author's needs, and new constraints are created *ad hoc* to "demonstrate" an X proposal. This, we will try to get rid of while presenting our Radically Minimalist version of OT.

- *Transderivational Constraints*: apply to sets of derivations, picking out an optimal derivation.

Our objection to this kind of constraints is simple: there are no such things as "sets of derivations" in any T_X, as derivations unfold in real time, interface-driven. Transderivational constraints are necessarily, if formulable, translocal constraints. Again, a *derivation* cannot be filtered out, but the resulting representation at a T_X.

These last two constraints explicitly express a characteristic of OT-syntax that we would like to revisit here, specially in connection to the learning algorithm we will present below: OT *requires* GEN to produce many alternative outputs, thus leading to a crash-rife syntactic model. In a crash-proof model (Putnam & Stroik, 2012; Putnam, 2009), GEN should optimally generate *one* candidate, which is in turn an *optimal* candidate. However, is this possible in early stages of grammar development? We believe not. This is why we are strongly against a fixed constraint system, which is not able to show the ontogenetic evolution from a crash-allowing mental grammar to a crash-proof one.

<u>On Constraints and Parameters:</u>

We will make a short note on Constraint-based syntax and Principles & Parameters-based syntax. OT-defenders claim that there are strong differences between those two kinds of models, and argue in favor of the first type. We will, in turn, argue that there is not much difference, not enough to distinguish two programs of research, at least.

Let us review some apparent differences:

- "Constraints are *violable*, whereas Principles are *not*"

This is not so, specially in the early P&P framework. Barrier violations were allowed, and the notion of "*increasing degree of ungrammaticality*" was put forth in direct proportion to the number of crossed bounding nodes / barriers. Even now, a principle like the PIC is not clear to apply in cases of so-called "sidewards movement" and scrambling.

- Parameters have two values, whereas Constraints are simple one-sided statements.

Let us review a well-known pair of constraints, posited by Müller (2011b), among others:

(4) *WH-CRITERION (WH-CRIT):*
Wh-items are in SpecC[wh].

θ-ASSIGNMENT (θ-ASSIGN):
Internal arguments of V are c-commanded by V.

In our opinion, these so-called "constraints" are nothing more than the two possible settings of the "Wh-parameter", with the extra elements and relations "Spec-of", "Wh-feature", "c-commanded-by". In this sense, the proposal is even less economical (and thus biologically implausible) than standard P&P approaches. Moreover, there is significative overlapping between θ-ASSIGN and *MOVE, which favors elements *in situ*.

- Constraints are *ranked*, whereas Parameters are *not*

This is another misunderstanding of recent developments within the P&P framework. Uriagereka (2007) asks whether OT and the MP are compatible or not. Putting aside the question of whether the problem itself is real (if OT is a theory and MP a methodological program, then there should be no problem at all), Uriagereka distinguishes *macro-parameters* from *micro-parameters*. Macro-parameters are early set and determine the *possibility* of setting more specific, possibly *more local*, parameters. These micro-parameters, external to the "core" UG, are subjected to change over time, more easily than macro-parameters, which are closely associated to linguistic typology. There is, then, a rank in current parametrical theory, but the same criticisms apply: which is the principle ruling the hierarchy? In our presentation, we will dispense with both systems as they are conceived now and develop a whole new framework based on Krivochen's (2011a, et. seq.) Radical Minimalism and Putnam's (2010) *crash-proof* proposals.

<u>EVAL revisited:</u>

In OT, the EVAL component is in charge of filtering out illegitimate representations generated by GEN. That evaluation, in *most* current versions of OT, applies at a global level (see Embick, 2010 for discussion), even though *globalism does not necessarily follow from an OT-like architecture*. The main problem we see here is that global evaluation is both computationally inefficient and biologically implausible. If we consider OT-like constraints in a local domain, things get only a little better: the problems with the ontology of the hierarchy remain. In turn, extremely local optimization (Müller, 2011b; Heck & Müller, 2010) leads to a proposal in which *all SO are –stipulatively- phases*, as in the strongest interpretations of Epstein & Seely (2002): bear in mind that the word *optimization* is used, not *evaluation*. If a SO is an optimal candidate, there is no reason why it should not be transferred *ipso facto*, regardless interface legibility conditions: a blind application of Pesetsky's (1995) *Earliness Principle*. Extremely local optimization leads to a crash-rife model, in which, as Epstein & Seely (2006, 2007) and Putnam (2010) notice, there is no place for "momentarily crashing" objects, which they refer to as a "strict crash" model. Let us take the following sentence (from Putnam, 2010: 4):

(5) Rats like cheese

The derivation of this sentence involves the creation of the following object in a derivational point T_X,

(6) $[_V$ like $[_D$ cheese$]]$

Which is not interpretable, either in an orthodox Minimalist framework (violation of the Theta-criterion, as there is an external role not assigned) or in a Radically Minimalist framework (no explicature can be built if the C-I interface takes that SO from the workspace). Extremely local optimization as presented in Müller (2011b) and Heck & Müller (2010) inevitably leads to crash almost at every point (i.e., partial representation created) in a derivation. This can be better understood if we consider two key assumptions underlying the GB conception of "well-formedness" (taken from Epstein & Seely, 2006: 179):

(7)

a. All and only grammatical sentences have 'well-formed' derivations.
b. A derivation is well-formed only if at every point in the derivation no principle is violated.

The goal of a newly revised OT (which we will pursue) would be then to model a local *evaluation* algorithm which is compatible with a softer version of crash-proofness: at some points in a derivation, violations must be tolerated, at least until the next derivational step and the evaluation of the output. The definition of *crash* that best suits this logic is referred to as "Soft-crash" (Putnam, 2010: 8):

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

<u>Radical Minimalism and Harmonic Serialism:</u>

Radical Minimalism's version of *harmonic serialism* is expressed in the following principle:

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

The *or* clause allows "soft crashes" like (3), as the merger between [$_V$ like] and [$_D$ cheese] does not generate a fully interpretable object for *both* interfaces, but it increases the informational load, and so avoids entropy, which is the main aim of interface-driven generation. In fact, we could even argue that *optimality* is an epiphenomenon, or a particular case of this increase in the informational load in a symbolic structure in which a fully interpretable object is built. DFI and our definition of *phase* require *real-time interpretation*, that is, each generated structure is analyzed and if things are going well within tolerable boundaries (i.e., the *next* derivational step), the derivation continues.

This on-line interpretation may sound more costly than waiting until the whole derivation is finished and only then evaluate it, but a dynamic definition of *phase* requires C-I to determine if a syntactic object is fully interpretable in order to be transferred. Our *evaluation* system will take the name of *Analyze*, a derivational step (not an operation) in which the interfaces evaluate the output of GEN dynamically using as “constraints” their own legibility conditions.

Manipulation of elements is free, but, of course, as certain patterns emerge as *frequently relevant*, those structures are built as the first option to be considered by the semantic component. Therefore, by resorting to adjustment of neurological connections (roughly, statistical learning as described, among others, by Thornton & Tesan, 2007), we can account for the generation of convergent structures without stipulations constraining *syntax*, which is a simple, general-purpose concatenation function, the GEN algorithm as formalized in (2). This will acquire major importance when we present *language* as a *chaotic system* in following sections.

<u>How the system works:</u>

<u>Notes:</u>

- Merge is *Monotonic Merge by Ontological Format* unless explicitly indicated.
- Parallel derivations have been discussed elsewhere, see Uriagereka (2002). We will maintain that mechanic, as C-I_2 also appears to be able to work in parallel when deriving *higher-level explicatures* and *implicatures* (Wilson & Sperber, 2003).
- We will just build a *complete thematic domain in a ditransitive structure*, leaving the derivation of higher nodes (Mod, Asp, T) to the reader.
- We will assume that every instance of Merge generates an interface-legitimate object (recognized by *Analyze*, which is *not an operation*, but merely the interfaces “peering into the syntax”, as in Boeckx’s system and finding the minimal legible units, i.e., *phases* to be transferred to the relevant module), something that is not necessarily the case in a *restrictivist system* with free Merge (Cf. Frampton & Guttman, 2002). In any case, the generation of “momentarily” illegible structures can be tolerated because of *Soft Crash*, as we assume there is a “local derivational unit” to repair the violation. We will also label according to what we have said here and in previous works.
- “Categorial interpretations”: we will assume that D collapses the ψ-state to N (i.e., sortal entity) without excluding [cause], T collapses it to V (extending-into-time perspective) and P collapses it into A (see Mateu, 2000a, b). Common sense may dictate that the primitive *cause* appears only in verbal (i.e., eventive) structures, but there is an aspect of the C-I_1-syntax interface that we have mentioned elsewhere and is essential to this: *this interface (and possibly, all other interfaces) is not transparent* (i.e., there is no exact correlation between a Relational Semantic Structure and its syntactic realization, as well as there is no exact isomorphism between the representations manipulated by two modules, even respecting the *Conservation Principle*).

- The Case interpretations will follow Krivochen (2010c) ***Case Sphere Theory***, which we very briefly summarize here:

Nominative: *read off* from a {Time, {D}} local relation, and *interpreted thematically* (in the explicature building process, see Sperber & Wilson, 2003) as Agent / Force

Accusative: *read off* from a {Cause, {D}} local relation, and *interpreted thematically* as Theme, the object (Figure) located in / moving towards, etc. a Ground

Dative: *read off* from a {P, {D}} local relation, and *interpreted thematically* as Location, the Ground in Talmy's terms.

1) *C-I_1-NS* interface: instantiate RSS elements following Conservation Principle.
2) *A-List*: provide procedural elements according to the instructions received from C-I_1
3) *NS* Merge ($D_{[CaseX]}$, √) = {D, √}
4) *C-I_2* Label {$D_{[CaseX]}$, √} = {D, {$D_{[CaseX]}$, √}} *This* {D} *will be taken as a unit for the purpose of future operations*. Incidentally, {$D_{[CaseX]}$, √} "categorizes" √ as N, following our definition.
5) *C-I_2* Analyze: not fully interpretable unit: D has a quantum dimension in its ψ-state.
6) *NS* Merge (P, {$D_{[CaseX]}$}) = {P, {$D_{[CaseX]}$}} P*'s procedural instructions collapse* [$Case_X$] *on* {D} to DAT sphere.
7) *C-I_2* Label {P, {$D_{[DAT]}$}} = {P, {P, {$D_{[DAT]}$}}}
8) *C-I_2* Analyze: {D}'s referential properties depend on the cumulative influence of Time, Aspect and Modality, if it is a common name. Proper names are taken to be inherently manipulable by C-I (see Krivochen, 2010a). Not fully interpretable yet. Relational element P requires another element (a *figure*).
9) *NS* Merge ($D_{[CaseX]}$, √) in parallel to (1) = {$D_{[CaseX]}$, √} Labeling and Analyzing also take place. No procedural head can collapse {D}'s Case dimension, so the structure is not yet fully interpretable.
10) *NS* Merge by *Structural Format* ({D}, {P, {P, {D}}}) = {{D}, {P, {P, {D}}}}
11) *C-I_2* Label {{D}, {P, {P, {D}}}} = {P}.
12) *C-I_2* Analyze: {D} has a [$Case_X$] quantum dimension still uncollapsed. Not fully interpretable. Therefore, P is not interpretable either.
13) *NS* Merge ([event], {P}) = {[event], {P}}

14) *C-I₂* Label {[event], {P}} = {event, {[event], {P}}}

15) *C-I₂* Analyze: idem (12)

16) *NS* Merge ([cause], {event}) = {[cause], {event}} Procedural instructions on [cause] can collapse [$Case_X$] on the closest {D} structure to ACC sphere.

17) *C-I₂* Analyze: is {P} now fully interpretable? Let us assume P [WITH], which gives the P domain a clausal flavor (Krivochen, 2010d), since the analysis of Double Object Constructions show that P [WITH] is semantically equivalent to V [HAVE]. P is then a fully interpretable object, no quantum dimensions are left on their ψ-state.

18) *NS* Transfer {P}

19) *C-I₂* Label {[cause], {event}} = {cause, {[cause], {event}}}

20) *C-I₂* Analyze: two procedural instructions will cause collapse, since there is no √ to provide the "semantic substance" needed for an explicature to be built. [cause] licenses an external position, forcing the system to "wait one more turn".

21) *NS* Merge (D, √) in parallel = {D, √}. Idem (9).

22) *NS* Merge by *Structural Format* ({D}, {cause, {[cause], {event}}}) = {{D}, {cause, {[cause], {event}}}}.

23) *C-I₂* Label {{D}, {cause, {[cause], {event}}}} = {cause, {{D}, {cause, {[cause], {event}}}}}

Our local evaluation system (qualifiable as a version of "*harmonic serialism*"), since it is feature-free, is more flexible, allowing the working memory to host structures of variable complexity before *Transfer* applies and the interfaces take what they can minimally read. Given the fact that FL is nothing more than a workspace originated from the intersection of two systems (CI / SM) and the activation of the prefrontal cortex, it exists *within* those systems, a proposal that is very similar in spirit to Putnam & Stroik's (2012) *Survive Minimalism*. Therefore, it is only natural that the so-called "external systems" (which are not external at all, in our proposal or in Putnam & Stroik's) can have access to the derivational steps. Now, it is essential for linguistic theory to have some formal definition of *complexity*, since otherwise it is impossible to even start the search for "principles of efficient computation" which, apparently, should prefer *simplicity* over *complexity*. Another issue, with stronger focus on OT-models, is whether constraints are *complexity-sensitive*, that is, if they apply only to objects up to a certain degree of complexity. We will propose a definition that allows a formal characterization of the notion, without closing ourselves to linguistics:

(10) *An object α is more complex (or less simple) than β iff: (a) building of α involves more derivational steps than β (i.e., the application of a further algorithm) (b) α does not imply a further drastic effect on the output over β.*

This is, we define *complexity* only when there is an *interpretative system* involved and taking into account legibility conditions. Thus, for example, a *sentence* is not "more complex" than a *word* if the interface effects we want to achieve can be obtained with a sentence but *not* with a word. Just like Relevance, *complexity* is a notion that is to be defined *at the interfaces* and as a *cost-benefit* relation. And, in the case of sub-personal systems, it can also be defined in biological terms, which are nothing more than specific instantiations of more general mathematical/physical principles. The line of reasoning is the following: take, for example, Tegmark's (2007) *Mathematical Universe Hypothesis*:

(11) **Mathematical Universe Hypothesis (MUH)**

Our external physical reality is a mathematical structure.

A (mathematical) structure is defined as a set S of abstract entities and relations $\{R_1, R_2 \ldots R_n\}$ between them. Essentially, MUH is not restricted to *our* Universe, and in a multiverse theory, this is a very relevant claim. Each Universe, then, is defined as a set of entities and relations, in very much the same way we have defined the basic tenets of Radical Minimalism. In this context, formalization is essential (see Krivochen, 2012 for a mathematical formalization of RM and cf. Collins & Stabler, 2012, for a formalization of orthodox Minimalism). Now, accepting that MUH is valid for *every possible Universe*, we have to see how physics is affected. A physical claim is a *particular instantiation* in a given Universe of a pair {E, O}, in which E is in itself an entity or group of entities and O the relevant operation upon E. Basically, $R_X = \{E, O\}$ & $R_X \in W_X$, where W is an *n*-dimensional workspace. Physical statements depend on the existence of a Universe and an external physical reality, something a mathematical claim can dispense with. Necessity[3], then, is a matter of mathematics, but not of physics. In Radical Minimalism, *necessity* in this strong sense is embodied in the notion of the *concatenation function*, which gives rise to any kind of structure (we repeat the definition for the reader's comfort):

(12) *Concatenation* defines a *chain* of coordinates $\{(x, y, z \ldots n)W_X \ldots (x, y, z \ldots n)W_Y \ldots (x, y, z \ldots n)W_n\}$ 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.

Now, if a physical claim is a particular instantiation of a (portion of a) mathematical structure, what is the place of biology within this hierarchy? Our position is that *biology is a particular instantiation of a physical system*, if we take "physical system" in the sense of "specific portion of the Universe taken for analysis". Biological statements are not formal, and we could very well say that biology is the science of the contingent, in the sense

[3] We understand *necessity* in the strongest possible sense: a proposition is *necessary* if and only if it is true in *every possible Universe*. The claim that complexity derives from some form of a concatenation function seems *necessary* to us in this sense.

that *necessity* in our strong sense is simply unconceivable. Any biological claim is ultimately a mathematical claim but with a higher level of specificity and involving entities in a determined Universe. Thus, Biolinguistics, important though it is, must not be taken as the ultimate field of inquiry.

Following this line, we will *mathematically* formalize current OT-like constraints, and develop a theory of dynamic development that obeys a different learning algorithm. This will allow us not only to gain descriptive adequacy, but also reach *justificative adequacy* (Krivochen, 2011a) and start answering the question of *why the architecture is the way it is*, not only *what* it is like or *how* it works.

A learning algorithm for OT-syntax?

In its current versions, OT is a *static* constraint theory, in which a variable *n* number of constraints (depending on the author) operate upon GEN outputs regardless stage in the ontogenetical development (from an empirical point of view) or *Occam's Razor* (from a theoretical point of view). Let us explicit the current OT mathematical algorithm:

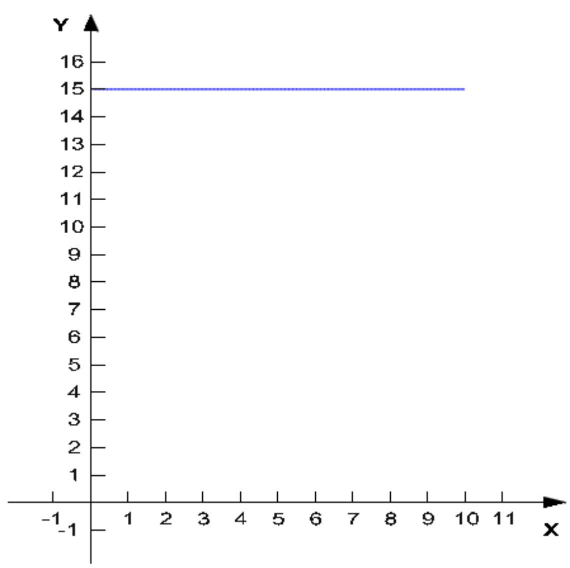
(13) For a system with *n* constraints, the function that describes an OT-grammar development is $f(x)=(x/y)+n$, in which x is T and y is *number of constraints*.

The aforementioned function gives us a static function, in which the values of y remain the same for all x, in n. For example, let us consider a system with 15 constraints, and apply the equation:

(14) $f(x)=(x/y)+15$

The result is a graphic like (15):

(15)



In this graphic, there is no account of development, as we can see. The number of constraints remains the same as times goes by, which is biologically implausible since neural networks have memory (Elman, 1985; Altmann, 2002). Networks encode information in the form of neural activation patterns that form a continuum, and learning (central notion in these models) is seen as an adjustment of patterns of activation, whether supervised or not: supervised learning, the system provides network input and the output expected, and the network must abstract the rule. In unsupervised learning, it is only provided input to the network, and the rule is tested a posteriori. The patterns of activation are electro-chemical stimulation must exceed a certain "threshold" to be meaningful. This threshold would be innate, the only “unlearnable” aspect in a connectionist network. Each relevant element in any mental workspace we analyze corresponds to a specific activation pattern, representing a unique electro-chemical perturbation. Networks generally consist of three layers of neurons: an input layer, a middle layer and an output layer. Suppose we have a network that interprets graphic stimuli (letters) and relates them to signifiers (acoustic images, borrowing the term from Saussure). Facing a grapheme that stimulates the activation threshold, neurons produce a given input pattern corresponding to the stimulus. This pattern is transmitted to an intermediate layer of neurons, which calculates the total activation force, and then the result is sent to an otuput layer of neurons (e.g., phonemes. See Altmann, 2002). However, the network has not learned anything yet, since activation thresholds are, in principle, arbitrary. In the mid-'80s, Jeffrey Elman proposed a connectionist model of neural networks that had memory which, exposed to linguistic stimuli, were able to "learn" to produce linear categorical expectations in the form of activation patterns (which encoded distributional information of each part of speech, clearly structural information) and to create "composite patterns" in the event that more than one possibility is feasible in a some point of the parsing process. Elman’s model incorporates to the above mechanism some "copy" neurons in the intermediate layer, each connected to an intermediate neuron. The information is transmitted from input neurons to the intermediate layer, and then copied to the copy neurons. When the intermediate neurons receive new information from the input neurons also receive a copy of the prior information from copy neurons, so that the pattern of activity of neurons intermediate incorporates both the new information and the network’s past reaction to similar activation stimuli, and that is sent to the output neurons. Thus, combining patterns, the network is able to anticipate what will come in the light of the experience of what it has already parsed. As an example, let us consider the case of a root like √CAT. Is it possible to form a verbal morpheme (that is, a syntactic terminal node) to which a phonological piece /kæt/ corresponds? Yes, if the underlying construal has [CAT] (the generic concept) in a legitimate position. We cannot form a verb $[_V$ cat] from a semantic construal where CAT is on Spec-R (the causative node), since we would be conflating the Spec into a Head, and such an operation would require many stipulations. This verb would be an *impossible word*[4]. If

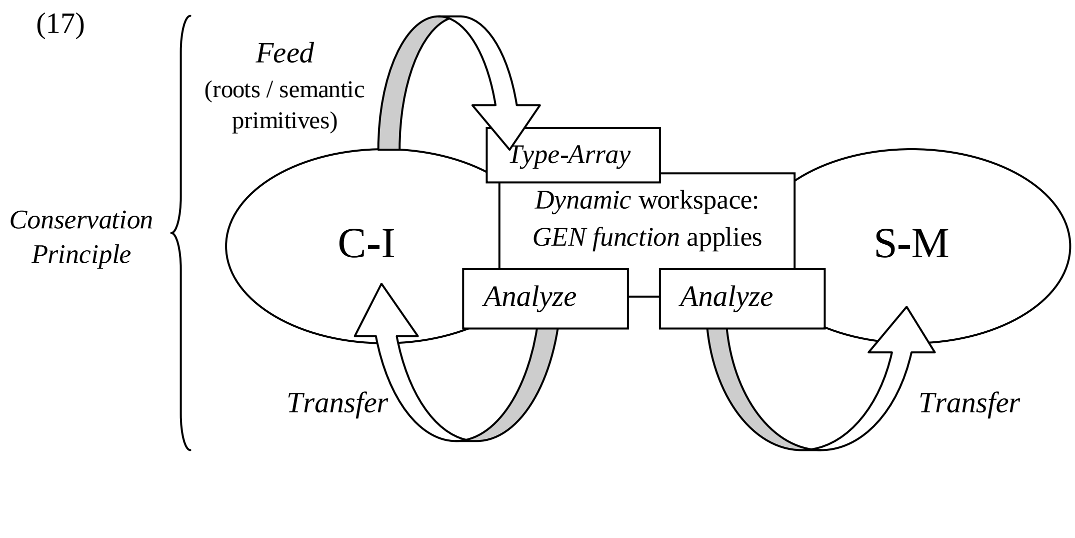
[4] The explanation for *impossible words* is very simple: Let us assume that we have $[_{XP}$ ZP $[_{X'}$ $[X_0]$ YP]] and X_0 is defective, either phonologically or semantically (see Hale & Keyser, 2002). If we consider the diachronic dimension of the derivation, as soon as we have [X’ $[X_0]$ YP], following the *Earliness Principle*, the *conflation* process must occur. There is no need

[CAT] is on Compl-*r*, for example, we could form a *locatum / location* verb [$_V$ cat] (for example, "to cat a mouse", meaning [CAUSE [GO [[mouse] [TO] [cat]]]] in Mateu's terms), and that would merely be a yet *uncoined word*, but perfectly *possible*, and “*parseable*”. The *Morpheme Formation Constraint* does not help when the morpheme has been formed according to “long-known principles of syntax” (think of GB’s principles, for example). However, *rutinized neurological connections* do. Bear in mind that only Merge comes “for free” (by conceptual necessity), the "lexicon" (by which we mean the *inventory of phonological pieces*, a purely socio-historical product), is *learned*. Learning is a process of adjustment of neurological connections, and when recurrent neurological flows (to use a metaphor) are rutinized, the connection is made quicker, almost automatic (in a Fodorian way). Our point here is that the program of static *n*-constraints is simply biologically implausible. What we need is a system that:

(16) A. Provides means for explaining development in a dynamic way
B. Is fully explicitable in mathematical terms, such that it can complete a wider theory of physical systems
C. Is compatible with an FL-non-existence hypothesis (Krivochen, 2011a et. seq.; see also Putnam & Stroik, 2012 for a weaker version)

This is the kind of model we have in mind. To begin with, we will explicit the architecture of the system as we have done with traditional OT (taken from Krivochen & Kosta, 2012):

(17)



Now, we will explicit the functioning of the system. We agree with Broekhius & Vogel (2010) in that Crash-proof does not entail filter elimination, but we will add that positing a fixed system of *n* constraints that remain

(and, what is more, it would be an anti-economical option) to wait until ZP is merged. *Dynamic Full Interpretation* also plays a role, as it *triggers* conflation to generate a fully interpretable object. See below.

the same during a *T* period of time is highly implausibly because of basic considerations of brain maturation: in a system with a free GEN (like Radical Minimalism), all kinds of candidates could be assembled, but generation is constrained by legibility conditions in an *extremely local harmonic serialism* (see the sample derivation above). In a system with "invasive interfaces" like the one we are building, in which $C_{(HL)}$ has ***no specificity*** and is the result of pre-frontal neocortex activation, FL is unteneable, and, moreover, superfluous. If there is no FL, then there is no UG, as UG is the theory of FL's initial state. Going back to the algorithm in (12), it is impossible that, for X = 0, Y = *n*. But it is also impossible that *n* maintains stable during the whole time *T*, since this would mean that the neurological networks have learned nothing from constant candidate filtering, and all constraints are equally necessary in every stage of development, against which we argue. For us, an optimal design would be one in which, for X = 0, Y = 0, constraints are abstracted via *non-supervised learning* in a speech community and in the end, from *n*, only *one* remains, that one which derives from necessity (and, interestingly enough, is a principle that has parallels in other physical systems): *Dynamic Full Interpretation* (Krivochen, 2011b, c, d, 2012):

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

Now, we will try to explicit what an RM algorithm would look like, by describing the state of the mental grammar at critical points of the development of the constraint system:

(19) Consider: $x = 1$ as the initial state of the mental system of constraints, $x = 2$ as an intermediate stage and $x = 3$ as the final mature state; furthermore, consider n as the maximal number of constraints.

(20) For $x \leq 1$, y asymptotically tends to 0

(21) For $x = 2$, $y = n$

(22) For $x \geq 3$, $y = 1$

(20) accounts for the potentiality of acquiring constraints from inductive abstraction in a speech community, as *y* never equals 0. As constraints act at the interface levels, this can be seen as the maturation of an innate, genotypic capacity in contact with some form[5] of the phenomenological world. There is a rising curve towards the maximum value in $x = 2$, an intermediate stage in the development in which all constraints are active and working (21). Between 2 and 3 is that stage in which so-called "redundancy rules" (or better, procedures) would apply: if a constraint C_1 only applies after C_2, and there is no C_3 that can also license the appearance of C_1, then C_1 is kept and C_2 is subsumed to C_1, and the system simplifies *itself*, looking for optimality. In this respect, a system that we could call "language" would be *chaotic* in the technical sense: it is sensitive to initial conditions, although the outcome is not unknown at all, as we will see, since we claim that OT can evolve to become the

[5] i.e., Jackendoff's "real world" or "projected world", depending on the philosophy the reader prefers, realistic or not.

simplest exponent of crash-proof via Radical Minimalism. An “everything goes” situation is not desirable, and we claimed that RM is a kind of crash-proof grammar, so some kind of (non-stipulative, third-factor) filter is required: this is where DFI enters the game.

Conclusion: Radically Minimalist OT as a model for crash-proof grammars

In this section we will go deeper in the point we have risen in the last paragraphs: is “language” a chaotic system? It *is* in the technical sense. Let us review some characteristics of chaotic systems and then go back to the discussion of whether language is one of them:

(23) Chaotic systems are:

- *Open* to external influence
- *Complex* (i.e., contain subsystems)
- *Dynamic* (i.e., change over time)

We have seen that language fulfills each of these intensional requirements: it is *open* because constraints are not abstracted by a mind unless in contact with data; it is *complex* because *syntax* interfaces with other two components (one of which –SM- is parasitic, but nonetheless an interface); and it is *dynamic* because the number of constraints changes over time, as they are subsumed to other, more fundamental constraints until reaching the optimal scenario: a crash-proof system with only one constraint. A caveat is in order here: the more constraints we have, the more stable the system will be: after a certain number of constraints, a “threshold”, the change is suddenly perfectly ordered and predictable. If we have only few constraints, the result will be a graphic that will tend to infinity instead of achieving internal balance after a certain period of time. In other words, the higher *n* is, the faster the system will get to the final state of $y = 1$[6]. In the end, we cannot fully dispense with constraints, but we cannot maintain *n* also for reasons we have already seen, so we will subsume all constraints (which, by this time X = 3, are already neurologically represented) to our *Dynamic Full Interpretation*. This gives balance to the system, what is normally called “negative feedback”: after an external perturbation in the system (in this case, say, contact with the “external physical reality”), it balances itself, thus getting to the stable state we are interested in. System self-organization is expressed by Bernárdez (2001) using Shannon’s equation for entropy augmentation (taken from Bernárdez, 2001: 9):

[6] This point is of great relevance for learnability theories: a Quantum Mind is not limited by number of elements to be learnt, moreover, if we accept that “language” is a chaotic system, once an external perturbation has affected the system, it will balance itself quicker in direct proportion to the value of N and *p*. In practical terms, this means that a plausible learnability theory does not necessarily imply a theory with little to learn, but a theory in which learnable elements can be found in relations with one another on regular basis after some time *T*.

(24)

$$H = -\sum_{i=1}^{N} p_i \log p_i$$

Where N is the number of individuals (i.e., constraints) and *p* the possibility of finding *i* of the N individuals in active state. This equation would be valid for X = 2, the so-called "intermediate state". Once all other constraints are subsumed to DFI, we have a system with the advantages of both free generation *and* crash-proof systems (Cf. Boeckx, 2010):

- GEN is mathematically defined as a universal, unbounded and free concatenation function.
- EVAL is dynamically conceived, which gives OT more biological plausibility: P&P's static character is replaced by a dynamic system of constraints, sensitive to external factors during a time T.
- DFI is a third-factor constraint, since it depends on *interface effects*: if the application of an operation generating SO_1 does not increase the informational load of SO_{-1}, it is an *illegitimate* step, and thus should not apply at all. This allows us to permit *soft crash* effects while still having a crash-proof syntax in the relevant sense, within *local* domains. This is an improvement over current versions of *harmonic serialism*, in which evaluation and optimization tends to be *too local* to be non-trivial or just crash-rife.

The advantages of adopting a RM version of OT are clear, we think. The result is a non-stipulative crash-proof syntax which is fully explicit in mathematical terms, with which we are a step closer to the highest level of scientific abstraction. Our only hope is that this is the first of many interdisciplinary works in the topic, and bridges between Linguistics, Physics and Mathematics can be built beyond specific theories and stipulations.

Bibliography:

Barbosa, P. et al. (eds.) (1998). *Is the best good enough? Optimality and competition in syntax*. Cambridge, Mass.: MIT Press.

Bernardez, E. (2001) *De monoide a especie biológica: aventuras y desventuras del concepto de lengua*. Ms. Universidad Complutense de Madrid.

Boeckx, C. (2007). *Eliminating Spell-Out. Linguistic Analysis* 33 (3‑4) (Dynamic Interfaces,

K.K. Grohmann (ed.)): 414‑425

(2010) *A tale of two minimalisms: Reflections on the plausibility of crash-proof syntax, and its free-merge alternative*. In Putnam, M. (ed.).

Broekhuis, H & R. Vogel (2010) *Crash-proof syntax and filters*. In Putnam, M. (ed.)

Chomsky, N. (1957) *Syntactic Structures*. Berlin. Mouton de Gutyer.

(1977): *On Wh-Movement*. In: P. Culicover, T.Wasow & A. Akmajian, eds., *Formal Syntax*. Academic Press, New York, pp. 71–132.

(1995) *The Minimalist Program*. MIT Press.

(2005) *Three Factors in Language Design*. Linguistic Inquiry 36, 1–22.

Di Sciullo, A-M. & D. Isac, (2008) *The Asymmetry of Merge.* In Biolinguistics 2.4: 260–290

Embick, D. (2010) *Localism versus Globalism in Morphology and Phonology*. Cambridge, Mass. MIT Press.

Epstein, S. & T.D. Seely. 2006. *Derivations in minimalism.* Cambridge: CUP.

Frampton, J. & S. Gutmann. 2002. “Crash-proof syntax.”, in Sam D. Epstein and T. Daniel Seely (eds.), *Derivation and Explanation in the Minimalist Program*. Oxford: Blackwell. 90-105.

Grimshaw, J (1997): *Projection, Heads, and Optimality*, Linguistic Inquiry 28, 373–422.

(1998): *Constraints on Constraints in Optimality Theoretic Syntax*. Ms., Rutgers University,

New Brunswick, New Jersey

Heck, F & G Müller (2007): *Extremely Local Optimization*. In: E. Brainbridge & B. Agbayani,

eds., *Proceedings of the 26th WECOL*. California State University, Fresno, pp. 170–183.

Heck, F & G Müller (2010): *Extremely Local Optimization*. Ms., Universität Leipzig. To appear

in Hans Broekhuis and Ralf Vogel (eds.), *Derivations and Filtering*. Equinox Publishing.

Kager, René (1999): *Optimality Theory*. Cambridge University Press, Cambridge.

Krivochen, D. (2011a) *An Introduction to Radical Minimalism I*. IBERIA, n°3. Vol 2.

(2011b) *Unified Syntax*. Ms. UNLP.

(2012) *Towards a Geometrical Syntax*. Ms. UNLP.

Krivochen, D. & P. Kosta (2012) *Empty Categories:Their Ontology and Justification*. Ms. UNLP / Universität Potsdam.

Leung, T. (2010) *On the mathematical foundations of crash-proof grammars*. In Putnam, M. (ed.)

Müller, G. (2011a) *Constraints on Displacement: a Phase-Based Approach*. Amsterdam, John Benjamins.

(2011b) *Optimality Theoretic Syntax*. Ms. Universität Leipzig.

Panagiotidis, P. (2010) *Functional Heads, Agree and Labels*. Ms. University of Cyprus.

Pesetsky, D. & E. Torrego (2007): *The syntax of valuation and the interpretability of features*. In: S. Karimi, V. Samiian & W. K. Wilkins, eds., "Phrasal and clausal architecture: Syntactic derivation and interpretation". Benjamins, Amsterdam, pp. 262–294.

Prince, Alan & Paul Smolensky (2004): *Optimality Theory. Constraint Interaction in Generative Grammar*.

Blackwell, Oxford.

Putnam, M (ed.) (2010) *Exploring Crash Proof Grammars*. Amsterdam, John Benjamins. [Language Faculty and Beyond, 3].

Putnam, M. (2010) *Exploring crash-proof grammars: An introduction*. In Putnam, M. (ed.)

Putnam, M. & T. Stroik (2010) *Syntactic relations in Survive-minimalism*. In Putnam, M. (ed.)

(2011) *Syntax at Ground Zero*. Ms. Penn State University & University of Missouri-Kansas.

(forthcoming) *The Structural Design of Language*.

Thornton, R., & Tesan, G. (2007). Categorical acquisition: Parameter setting in Universal Grammar. *Biolinguistics*, 1, 49-98.

Uriagereka, J. (2007) *Clarifying the Notion "Parameter"*. Biolinguistics, 1.1.