Towards a nontrivial notion of granularity in generative syntax

Chenchen Song, cs791@alumni.cam.ac.uk
University of Cambridge

Talk at Zhejiang University
Hangzhou, November 11, 2019

Overview

Progress

Recall: Hallmarks of generative syntax

> *Generative syntax [insists] on rigorous formal modeling of linguistic patterns.*
>
> —Taraldsen (Oxford Bibliographies)

Recall: Hallmarks of generative syntax

Generative syntax [insists] on rigorous formal modeling of linguistic patterns.

—Taraldsen (Oxford Bibliographies)

The underlying thesis of generative grammar is that sentences are generated by a subconscious set of procedures [, which] are part of our minds . . . . The goal of syntactic theory is to model these procedures.

In generative grammar, the means for modeling these procedures is through a set of formal grammatical rules. . . . These rules are thought to generate the sentences of a language, hence the name generative grammar.

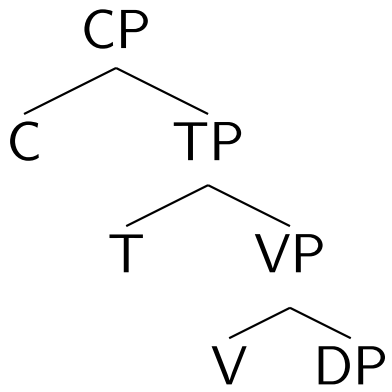
—Carnie (2013:6)

Formal tools used in generative syntax

Formal tools used in generative syntax

Among others:

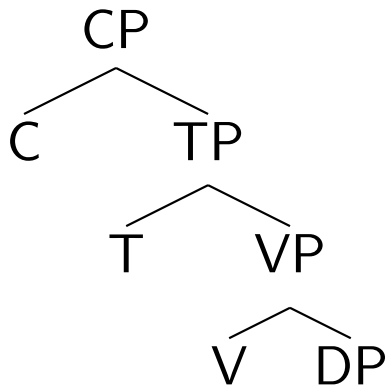
- Rooted trees (a type of graph)
  - e.g.,



Formal tools used in generative syntax

Among others:

- Rooted trees (a type of graph)
  - e.g.,

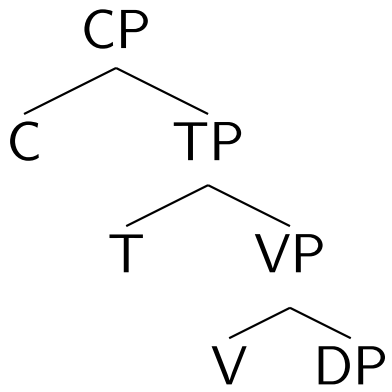


- Feature structures (a type of data structure)
  - e.g., [[PER: 1ST], [NUM: SG], [GEN: MASC], [CASE: ACC]]

Formal tools used in generative syntax

Among others:

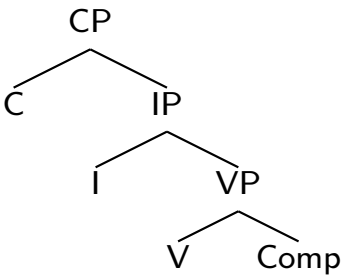
- Rooted trees (a type of graph)
  - e.g.,



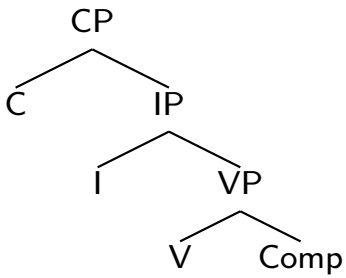
- Feature structures (a type of data structure)
  - e.g., [[PER: 1ST], [NUM: SG], [GEN: MASC], [CASE: ACC]]

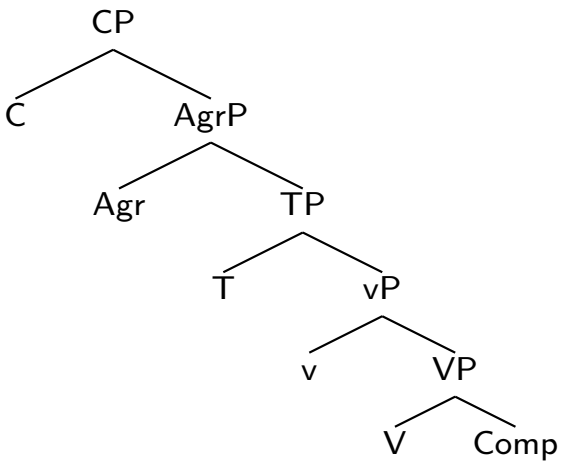
There is a trend in the usage of these tools, witnessed by two observations.

Observation 1: Trees grow

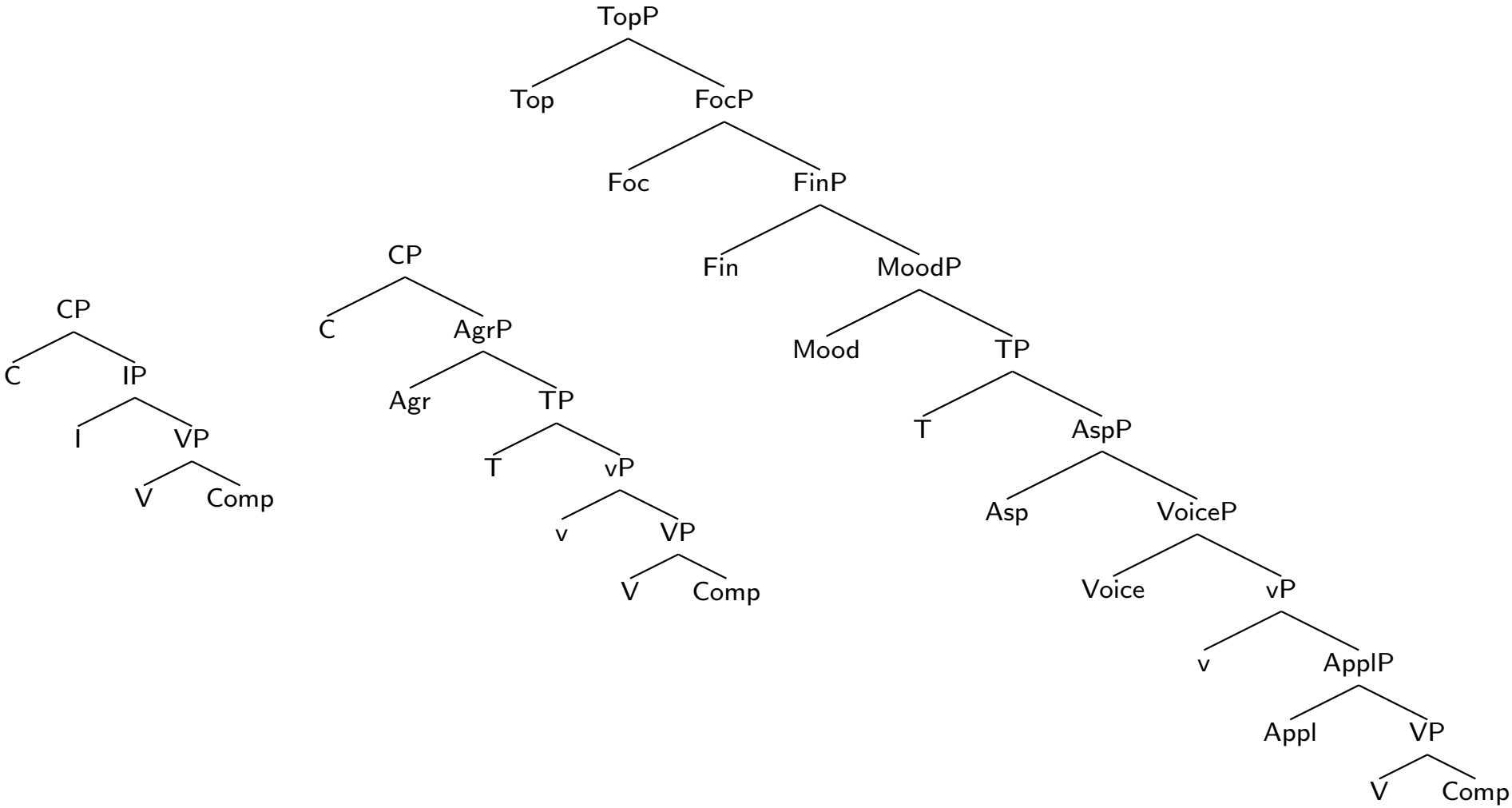


Observation 1: Trees grow

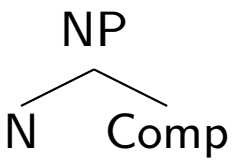




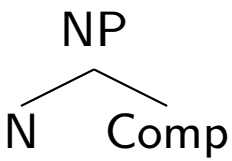
Observation 1: Trees grow

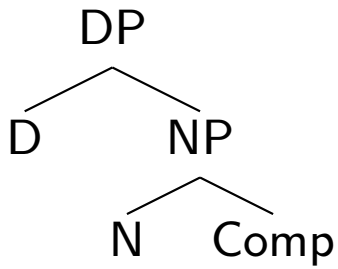


Observation 1: Trees grow

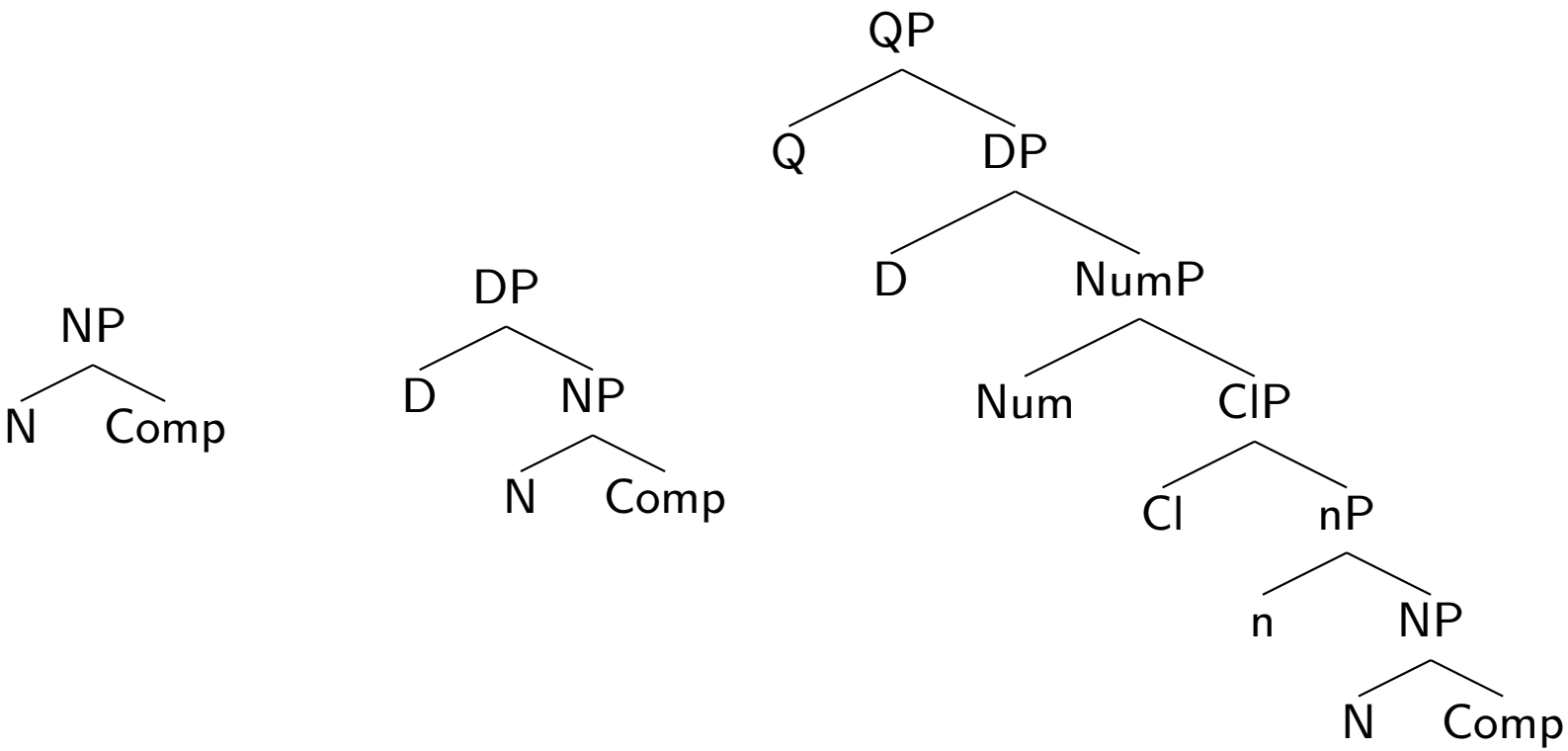


Observation 1: Trees grow





Observation 1: Trees grow



Observation 1: Trees grow

And the same has happened to adjectives and adpositions. . . [1]

- AdjP → [AgrP [DegP [QP [AP]]]]
- PP → [P_{Dir}P [P_{Stat}P [DegP . . . [DeicticP [AxPartP [PP]]]]]]]

Trees grow larger and larger in generative syntax, with more and more functional categories being proposed.

[1] See Corver (1997) and Cinque & Rizzi (2010).

Observation 2: Features expand

For example, the conceptions of NUMBER and PERSON have changed:

- [NUM:SG] → [NUM:[+ATOMIC, −AUGMENTED]] (Adger 2010)
- [PER:1ST] → [PER:[+AUTHOR, +PARTICIPANT]] (Harbour 2016)

Observation 2: Features expand

For example, the conceptions of NUMBER and PERSON have changed:

- [NUM:SG] → [NUM:[+ATOMIC, −AUGMENTED]] (Adger 2010)
- [PER:1ST] → [PER:[+AUTHOR, +PARTICIPANT]] (Harbour 2016)

So has the conception of lexical categories:

- Erstwhile: [CATEGORY:N/V]
- GB: [CATEGORY: ±N, ±V]
- Now: What do the symbols N and V really mean?
  - Panagiotidis (2015): [PERSPECTIVE: SORTAL/TEMPORAL]
  - Biberauer & Roberts (2015): no universally fixed definitions

What has changed?

What has changed?

1. Trees grow—but they still describe the same kind of syntactic objects (sentences, noun phrases, etc.)
   e.g., *I saw a dog* can be analyzed either as a simple CP-IP-VP structure or as a cartographic structure.

What has changed?

1. Trees grow—but they still describe the same kind of syntactic objects (sentences, noun phrases, etc.)
   e.g., *I saw a dog* can be analyzed either as a simple CP-IP-VP structure or as a cartographic structure.
2. Features expand—but they still describe the same kind of lexical items
   e.g., whether we describe *dog* by [NUM:SG] or [NUM:[+ATOMIC,−AUGMENTED]], we are still describing *dog*.

What has changed?

1. Trees grow—but they still describe the same kind of syntactic objects (sentences, noun phrases, etc.)
   e.g., *I saw a dog* can be analyzed either as a simple CP-IP-VP structure or as a cartographic structure.
2. Features expand—but they still describe the same kind of lexical items
   e.g., whether we describe *dog* by [NUM:SG] or [NUM:[+ATOMIC,−AUGMENTED]], we are still describing *dog*.

What have changed are not the objects of study but the analyses.

What has changed?

1. Trees grow—but they still describe the same kind of syntactic objects (sentences, noun phrases, etc.)
   e.g., *I saw a dog* can be analyzed either as a simple CP-IP-VP structure or as a cartographic structure.
2. Features expand—but they still describe the same kind of lexical items
   e.g., whether we describe *dog* by [NUM:SG] or [NUM:[+ATOMIC,−AUGMENTED]], we are still describing *dog*.

What have changed are not the objects of study but the analyses.
More specifically, what has changed is the granularity of analysis.

Granularity in generative syntax

An overall trend in generative syntax in the past half century is the increasing fine-grainedness of analyses. This is witnessed by some popular research paradigms such as cartography and distributed morphology.

- **Cartography** (Rizzi 1997, Cinque 1999, et seq.): accurately map the syntacticosemantic subtleties of natural language utterances
- **Distributed morphology** (Halle & Marantz 1993, 1994, et seq.): painstakingly decompose lexical items to their derivational atoms

Granularity in generative syntax

An overall trend in generative syntax in the past half century is the increasing fine-grainedness of analyses. This is witnessed by some popular research paradigms such as cartography and distributed morphology.

- **Cartography** (Rizzi 1997, Cinque 1999, et seq.): accurately map the syntacticosemantic subtleties of natural language utterances
- **Distributed morphology** (Halle & Marantz 1993, 1994, et seq.): painstakingly decompose lexical items to their derivational atoms

But as of 2019 there is no dedicated study on granularity. . . I will tentatively lay out some "foundations" for the notion. The aim is to inspire further discussion (as we are stepping into a not-yet-explored zone).

Progress

What is granularity? (first attempt)

Granularity is the level of detailedness in description or analysis.
It is usually but not necessarily theoretically driven.

Descriptive granularity

We can see more details with a microscope without considering any theoretical analysis.

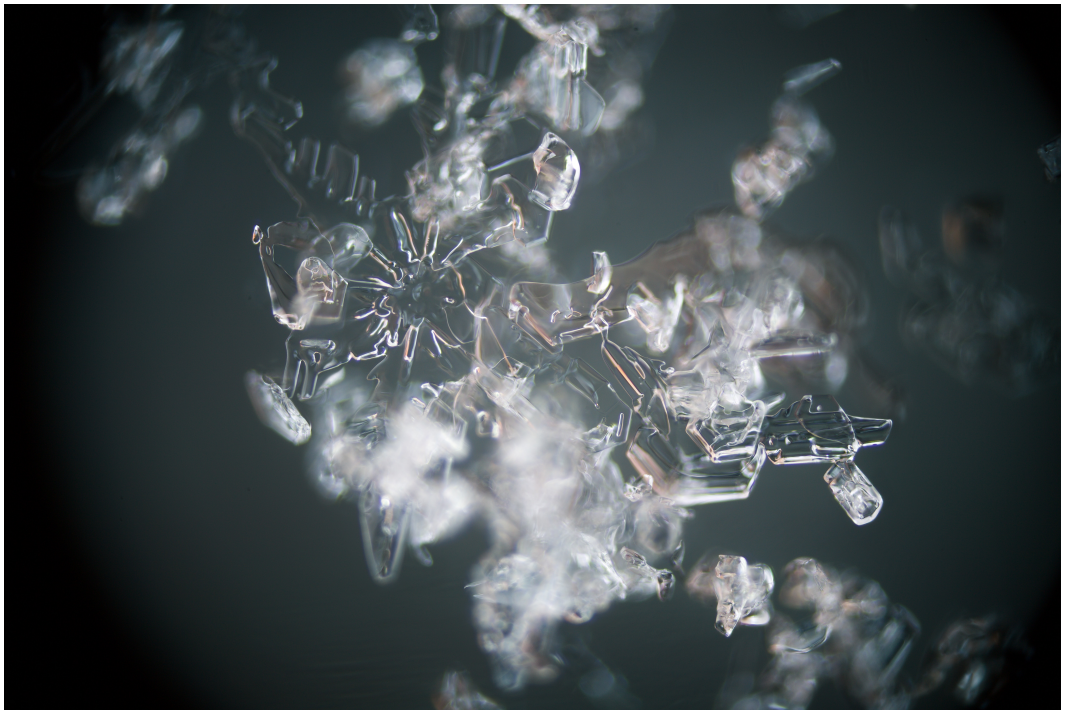


Figure 1: Frosted snow (photo by Andrii Ganzevych on Unsplash)

Explanatory granularity

However, as the descriptive granularity for a phenomenon increases, its explanatory granularity must also increase, because the newly revealed details become new explicanda. In other words, **descriptive granularity and explanatory granularity must match.**

Explanatory granularity

However, as the descriptive granularity for a phenomenon increases, its explanatory granularity must also increase, because the newly revealed details become new explicanda. In other words, **descriptive granularity and explanatory granularity must match.**

In fact not only descriptions and their explanations but also different aspects of a single explanation must match in granularity.

Example: syntax and semantics

Montague (1974): There is an algebraic homomorphism from syntax to semantics.

Partee (2004): The meaning of an expression is a function of the meanings of its parts and of the way they are syntactically combined. (Frege's principle)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] =?, [\![\text{Voice}]\!] =?, [\![\text{Fin}]\!] =?$
  (an easier task; see Ramchand & Svenonius 2014)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] =?, [\![\text{Voice}]\!] =?, [\![\text{Fin}]\!] =?$
  (an easier task; see Ramchand & Svenonius 2014)
- we must decide how to compose roots
  $[\![\sqrt{\text{DOG}}]\!] =?, [\![\sqrt{\text{RUN}}]\!] =?, [\![\text{n}, \sqrt{\text{DOG}}]\!] =?$
  (a less easy task; see Kelly 2013, Song 2019)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] = ?, [\![\text{Voice}]\!] = ?, [\![\text{Fin}]\!] = ?$
  (an easier task; see Ramchand & Svenonius 2014)
- we must decide how to compose roots
  $[\![\sqrt{\text{DOG}}]\!] = ?, [\![\sqrt{\text{RUN}}]\!] = ?, [\![\text{n}, \sqrt{\text{DOG}}]\!] = ?$
  (a less easy task; see Kelly 2013, Song 2019)

This is essentially a model extension. And if we view syntactic derivations as formal proofs (following Chomsky 1965, 2007), then this extension must not damage the well-formedness of the proof system (with respect to soundness, completeness, etc.).[2]

[2] A syntactic derivation system is *sound* if it can only derive/prove semantically valid sentences and *complete* if all semantically valid sentences can be derived/proved in it.

Example: syntax and semantics

Moreover, the model extension must meet the "interface condition" (Chomsky 2004) on a conceptual level; that is, we cannot assign denotations to terms merely based on model-theoretic needs but should also ask: What is the conceptual interpretability?

Example: syntax and semantics

Moreover, the model extension must meet the "interface condition" (Chomsky 2004) on a conceptual level; that is, we cannot assign denotations to terms merely based on model-theoretic needs but should also ask: What is the conceptual interpretability?

Compare categorial grammar and minimalism:

- Categorial grammar: an intransitive verb is a category $\frac{S}{\mathrm{NP}}$ that when combined with NP yields S ($\frac{S}{\mathrm{NP}} \times \mathrm{NP} \rightarrow S$)
- Minimalism: an intransitive verb can be assigned a denotation t^e (a type that when combined with an individual yields a truth value; $\mathrm{t}^e \times e \rightarrow \mathrm{t}$), but its interface interpretation is more than that (e.g., Panagiotidis' "extending-in-time")[3]

[3] e (for individuals) and t (for truth values) are the two basic semantic types in model-theoretic (aka "formal") semantics, and the function type t^e describes a one-place predicate (e.g., an intransitive verb).

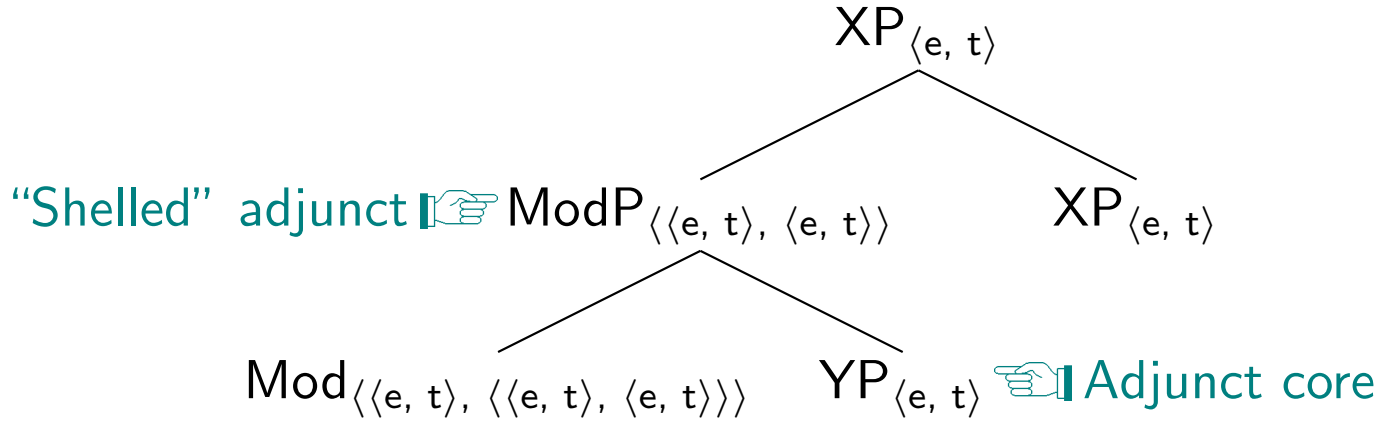
Functional categories

Just as purely morphosyntactic considerations may lead to "uninterpretable" categories (e.g., Chomsky's 1995 criticism of Agr), so purely model-theoretic considerations may lead to "proofs" of conceptually vague or void categories.

Functional categories

Just as purely morphosyntactic considerations may lead to "uninterpretable" categories (e.g., Chomsky's 1995 criticism of Agr), so purely model-theoretic considerations may lead to "proofs" of conceptually vague or void categories.

e.g., Rubin's (2003) semantic definition of Mod (an "adjunct shell")



How different is this definition from that of intransitive verbs as "a category that when combined with NP yields S"? (Not really.)

This style of definition defines what a category **does** but not what it **is**.

Roots

Now let us turn to roots. Syntacticians (especially DMers) have decomposed content words like *dog* and *run* into roots and categorizers, but this has not had much influence on semanticists, who mostly still treat bare nouns and verbs as (typed) predicates.

- $[\![\text{dog}]\!] = \lambda x.\text{DOG}(x)$
- $[\![\text{run}]\!] = \lambda e.\text{RUN}(e)$

Roots

Lexical decomposition in syntax corresponds to predicate decomposition in semantics, but the latter has not reached the root level.

- Jones buttered the toast slowly in the bathroom with a knife.
- $\exists e.\text{BUTTER}(e) \wedge \text{AGENT}(e) = \text{Jones} \wedge \text{THEME}(e) = \text{TOAST} \wedge \text{SLOWLY}(e) \wedge \text{LOCATION}(e) = \text{BATHROOM} \wedge \text{INSTRUMENT}(e) = \text{KNIFE}$ (Landman 2000)

Roots

Lexical decomposition in syntax corresponds to predicate decomposition in semantics, but the latter has not reached the root level.

- Jones buttered the toast slowly in the bathroom with a knife.
- $\exists e.\text{BUTTER}(e) \wedge \text{AGENT}(e) = \text{Jones} \wedge \text{THEME}(e) = \text{TOAST} \wedge \text{SLOWLY}(e) \wedge \text{LOCATION}(e) = \text{BATHROOM} \wedge \text{INSTRUMENT}(e) = \text{KNIFE}$ (Landman 2000)

If DMers want to develop a comprehensive theory of lexical decomposition, they need a semantics with matching granularity for the sake of Frege's principle.

Summary 1

In sum, both exemplary paradigms of high-granularity syntax (cartography and distributed morphology) require some effort to achieve an adequate semantics with matching granularity.

Summary 1

In sum, both exemplary paradigms of high-granularity syntax (cartography and distributed morphology) require some effort to achieve an adequate semantics with matching granularity.

Both cartography and DM are about **tree growth**, but the same sort of concern arises in **feature expansion** as well.

Features

How do individual features together describe lexical items? The usual format is a record-like data structure (a list of attribute-value pairs).

- *he*: [PER:3RD, NUM:SG, GEN:MASC]
- *runs*: [CATEGORY:V, PER:3RD, NUM:SG]

Features

How do individual features together describe lexical items? The usual format is a record-like data structure (a list of attribute-value pairs).

- *he*: [PER:3RD, NUM:SG, GEN:MASC]
- *runs*: [CATEGORY:V, PER:3RD, NUM:SG]

How are such featural descriptions interpreted? The usual mode of composition is conjunction.

- $[\![\text{PER:3RD, NUM:SG, GEN:MASC}]\!] = [\![\text{PER:3RD}]\!] \wedge [\![\text{NUM:SG}]\!] \wedge [\![\text{GEN:MASC}]\!]$
- $[\![\text{CATEGORY:V, PER:3RD, NUM:SG}]\!] =$
  $[\![\text{CATEGORY:V}]\!] \wedge [\![\text{PER:3RD}]\!] \wedge [\![\text{NUM:SG}]\!]$

Features

But this mode of composition has two prerequisites:

1. All features within a [ ] must be of the same type (e.g., first-order predicate), because conjunction requires type matching.
2. All features within a [ ] must have parallel status (i.e., no hierarchical structure), because conjunction is commutative.

Features

But this mode of composition has two prerequisites:

1. All features within a [ ] must be of the same type (e.g., first-order predicate), because conjunction requires type matching.
2. All features within a [ ] must have parallel status (i.e., no hierarchical structure), because conjunction is commutative.

This is at odds with the kind of tree structure familiar in Chomskyan syntax, especially with the branch that argues for "syntax (aka Merge) all the way down." It is more suitable for the unification-based framework.

Two types of granularity lifting

Record-and-conjunction-based feature expansion is of a different type of granularity lifting from that in cartography/DM. I call them **paradigmatic granularity** and **syntagmatic granularity** respectively.

- Paradigmatic: redefine a feature by a list of features
  e.g., [SG] → [+ATOMIC, −AUGMENTED]
- Syntagmatic: redefine a category by a sequence of categories
  e.g., I → Agr-T, C → Top-Foc-Fin, V → Init-Proc-Res

Problem

Feature integration does not always fall in a record-and-conjunction style.

- Some features are not designed to denote (first-order) predicates
- Sometimes the ordering of features matters (i.e., noncommutative)

Problem

Feature integration does not always fall in a record-and-conjunction style.

- Some features are not designed to denote (first-order) predicates
- Sometimes the ordering of features matters (i.e., noncommutative)

For example, Harbour's (2016) lattice-theoretic person features:

- $[\![+\textsc{author}(\pi)]\!] = [\![\textsc{author}]\!] \oplus [\![\pi]\!] = \{\mathfrak{a} \sqcup \mathfrak{b} : \mathfrak{a} \in \mathcal{L}_{\mathfrak{au}}, \mathfrak{b} \in \mathcal{L}_{\pi}\}$
- $[\![-\textsc{author}(\pi)]\!] = [\![\textsc{author}]\!] \ominus [\![\pi]\!] = \{\mathfrak{b} \setminus \mathfrak{max}(\mathcal{L}_{\mathfrak{au}}) : \mathfrak{b} \in \mathcal{L}_{\pi}\}$

where person features denote lattice-theoretic structures and $+/-$ denote actions on those structures.

Summary 2

So feature expansion (i.e., paradigmatic granularity lifting) also requires some effort in achieving a semantics with matching granularity, because

- there's no fixed denotation pattern for features
- there's no fixed mode of composition for features.

Interim summary

Granularity $\triangleq$ level of detailedness

Two types of granularity in general: descriptive and explanatory.

Two types of granularity matching: (i) descriptive-explanatory, (ii) different aspects of an explanatory theory (group).

It is not easy to maintain granularity matching, as illustrated by the various thorny issues arising from the syntax-semantics example. . . which may partly explain why both syntacticians and semanticists prefer focusing on one side of the story.

Progress

Status quo of granularity

1. Granularity is omnipresent in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.

Status quo of granularity

1. Granularity is omnipresent in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.
2. Granularity is flexible in generative syntax: Different analyses may assume different levels of granularity.
   e.g., standard minimalism vs. cartography

Status quo of granularity

1. Granularity is omnipresent in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.
2. Granularity is flexible in generative syntax: Different analyses may assume different levels of granularity.
   e.g., standard minimalism vs. cartography
3. Granularity is usually left implicit in generative syntax: It is sometimes a matter of trend and sometimes to-each-his-own.
   e.g., C-Agr-T-Asp-v-V vs. C-T-Asp-Voice-Appl-V

Status quo of granularity

4. Analyses of different granularity levels often overlap in labels, though the granularity difference means that identical labels may not have identical definitions.
   e.g., "v" in C-T-v-V $\neq$ "v" in C-Agr-T-Asp-Voice-v-Appl-V
   (some functionalities of the left-hand v are relocated to Voice/Appl in the right-hand sequence)

Status quo of granularity

4. Analyses of different granularity levels often overlap in labels, though the granularity difference means that identical labels may not have identical definitions.
   e.g., "v" in C-T-v-V $\neq$ "v" in C-Agr-T-Asp-Voice-v-Appl-V
   (some functionalities of the left-hand v are relocated to Voice/Appl in the right-hand sequence)
5. Given 1–4, granularity mismatches may lead to fundamental misunderstandings of terms/concepts, and a lack of granularity awareness is detrimental to theoretical integration.
   e.g., if my "v" isn't your "v" how do we know we are arguing about the same thing? (see, e.g., D'Alessandro, Franco & Gallego 2017)

But these are just methodological considerations. . . or are they?

Progress

A new conception

There is more to granularity than methodology:

Generalized granularity

A granularity level Γ can be viewed as the ambient categorial context for a derivational analysis $\mathcal{A}$.

$$\Gamma \vdash \mathcal{A}$$ [4]

[4] I use this notation because this conception of granularity is somewhat similar to the *typing context* in type theory.

A new conception

There is more to granularity than methodology:

Generalized granularity

A granularity level Γ can be viewed as the ambient categorial context for a derivational analysis A.

$$\Gamma \vdash A$$ [4]

In other words, a granularity level in the abstract sense is just a *background categorial setting*. A granularity level is **completely defined** by the **syntactic categories** it consists of as well as their **individual definitions**.

This abstraction encompasses syntagmatic and paradigmatic granularity.

[4] I use this notation because this conception of granularity is somewhat similar to the *typing context* in type theory.

Typing context (Nederpelt & Geuvers 2014)

In type theory, a *typing context* (or simply *context*) is a (possibly empty) list of typing statements for variables.
e.g., $\Gamma \triangleq x_1 : \alpha,\ x_2 : \alpha \to \beta,\ x_3 : (\beta \to \alpha) \to \beta$

A *judgment* $\Gamma \vdash M : \sigma$ is derivable iff M has type σ in context Γ.
e.g., given the above context, $\Gamma \vdash x_2 x_1 : \beta$ is derivable

$$\frac{\Gamma \vdash x_1 : \alpha \qquad \Gamma \vdash x_2 : \alpha \to \beta}{\Gamma \vdash x_2 x_1 : \beta}$$

When one says "M has type σ" a context is always assumed.

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Examples:

- $\Gamma \vdash x_2 x_1 : \beta$ ($x_2 x_1 : \beta$ is derivable in context Γ)
- $\Gamma \vdash$ *I saw a dog* :
  $[_{CP}\ C\ [_{TP}\ [_{DP}\ I\ ]\ [_{TP}\ see_i\text{-}T_{Past}\ [_{vP}\ v\ [_{VP}\ [_{V}\ t_i\ ]\ [_{DP}\ a\ dog\ ]]]]]]$
  (this structure is derivable in granularity level $\Gamma \triangleq \{C, T, v, V, D \ldots\}$)

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Examples:

- $\Gamma \vdash x_2 x_1 : \beta$ ($x_2 x_1 : \beta$ is derivable in context Γ)
- $\Gamma \vdash$ *I saw a dog* :
  $[_{CP}\ C\ [_{TP}\ [_{DP}\ I\ ]\ [_{TP}\ see_i\text{-}T_{Past}\ [_{vP}\ v\ [_{VP}\ [_{V}\ t_i\ ]\ [_{DP}\ a\ dog\ ]]]]]]$
  (this structure is derivable in granularity level $\Gamma \triangleq \{C, T, v, V, D \ldots\}$)

Difference: no term variables in granularity level (Why?)

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Examples:

- $\Gamma \vdash x_2 x_1 : \beta$ ($x_2 x_1 : \beta$ is derivable in context Γ)
- $\Gamma \vdash$ *I saw a dog* :
  $[_{CP}\ C\ [_{TP}\ [_{DP}\ I\ ]\ [_{TP}\ see_i\text{-}T_{Past}\ [_{vP}\ v\ [_{VP}\ [_{V}\ t_i\ ]\ [_{DP}\ a\ dog\ ]]]]]]$
  (this structure is derivable in granularity level $\Gamma \triangleq \{C, T, v, V, D \ldots\}$)

Difference: no term variables in granularity level (Why?)

Because of empty categories!

- There are more abstract categories (i.e., types) than overt vocabulary items (i.e., terms) in current generative syntax.

Granularity level vs. lexical array

Due to the existence of empty categories, a fully type-theoretic syntax is impossible in current minimalism (though it is possible and has been implemented in categorial grammar).

Granularity level vs. lexical array

Due to the existence of empty categories, a fully type-theoretic syntax is impossible in current minimalism (though it is possible and has been implemented in categorial grammar).

The closest concept to a bona fide typing context we can get in minimalism is the *numeration* or *lexical array*.

- e.g., {*I*:D, *saw*:V-T, *a*:D, *dog*:N}

But even that is complicated by empty categories (e.g., v)!

Granularity level vs. lexical array

Due to the existence of empty categories, a fully type-theoretic syntax is impossible in current minimalism (though it is possible and has been implemented in categorial grammar).

The closest concept to a bona fide typing context we can get in minimalism is the *numeration* or *lexical array*.

- e.g., {*I*:D, *saw*:V-T, *a*:D, *dog*:N}

But even that is complicated by empty categories (e.g., v)!

A granularity level is **not** a lexical array; it is an inventory of types instead.

- A lexical array only serves a single derivation.
- A granularity level in theory underlies an infinite number of derivations.

Granularity redefined

Recall first attempt: the level of detailedness in description or analysis.

Granularity (second attempt)

A granularity level is a set of well-defined syntactic categories that can serve as the "typing context" of an entire syntactic derivation system.

Granularity redefined

Recall first attempt: the level of detailedness in description or analysis.

Granularity (second attempt)

A granularity level is a set of well-defined syntactic categories that can serve as the "typing context" of an entire syntactic derivation system.

In other words, a granularity level is just a **functioning inventory** of syntactic categories (where *functioning* means that a usable derivation system can be built based on the given inventory). For example:

- The inventory used in GB.
- The inventory used in standard minimalist program (Chomsky 1995).
- The inventory used in cartography.

Granularity redefined

Recall first attempt: the level of detailedness in description or analysis.

Granularity (second attempt)

A granularity level is a set of well-defined syntactic categories that can serve as the "typing context" of an entire syntactic derivation system.

In other words, a granularity level is just a **functioning inventory** of syntactic categories (where *functioning* means that a usable derivation system can be built based on the given inventory). For example:

- The inventory used in GB.
- The inventory used in standard minimalist program (Chomsky 1995).
- The inventory used in cartography.

Remarks on new definition

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

Remarks on new definition

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

So, not only GB, standard MP, and cartography but also various categorial inventories of intermediate sizes can define granularity levels.

- Top-Foc-Fin-Mood-T-Asp-v-V...
- C-T-Asp-Voice-v-Appl-V...
- C-T-Init-Proc-Res...

Remarks on new definition

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

So, not only GB, standard MP, and cartography but also various categorial inventories of intermediate sizes can define granularity levels.

- Top-Foc-Fin-Mood-T-Asp-v-V...
- C-T-Asp-Voice-v-Appl-V...
- C-T-Init-Proc-Res...

And syntacticians can indeed freely choose whichever granularity level they like as the background categorial setting of their analyses.
e.g., someone studying the C-domain may choose to only split CP, and someone studying the V-domain may choose to only split VP.

Progress

Takeaway

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in analytical granularity.

Takeaway

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in analytical granularity.
2. Two ways to define granularity: (1) level of detailedness, (2) inventory of syntactic categories.

Takeaway

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in analytical granularity.
2. Two ways to define granularity: (1) level of detailedness, (2) inventory of syntactic categories.
3. Granularity is a multifaceted notion: (1) descriptive vs. explanatory, (2) syntagmatic vs. paradigmatic.

Takeaway

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in analytical granularity.
2. Two ways to define granularity: (1) level of detailedness, (2) inventory of syntactic categories.
3. Granularity is a multifaceted notion: (1) descriptive vs. explanatory, (2) syntagmatic vs. paradigmatic.
4. Granularity is omnipresent, flexible, and usually left implicit in generative syntax. Its role is analogous (though not exactly parallel) to that of a typing context in type theory.

The bigger picture

Our generalized notion of granularity is a formal one. While this suffices for derivational purposes, we may legitimately ask: Is there more to granularity in generative syntax, especially in the minimalist program?

The bigger picture

Our generalized notion of granularity is a formal one. While this suffices for derivational purposes, we may legitimately ask: Is there more to granularity in generative syntax, especially in the minimalist program?

What role does granularity play in human language as a natural object?

The bigger picture

Our generalized notion of granularity is a formal one. While this suffices for derivational purposes, we may legitimately ask: Is there more to granularity in generative syntax, especially in the minimalist program?

What role does granularity play in human language as a natural object?

Two directions to consider:

1. How is granularity related to typology?
2. How is granularity related to I-language?

The formal definition and the big-picture questions make granularity-oriented thinking a nontrivial move in the minimalist program.


Thank you!

Selected bibliography


Harbour, Daniel
Impossible Persons.
MIT Press, 2016.


Panagiotidis, Phoevos
Categorial Features.
CUP, 2015.


Biberauer, Theresa & Ian Roberts
Rethinking formal hierarchies: A proposed unification.
Cambridge Occasional Papers in Linguistics 7, 2015.


Song, Chenchen
On the formal flexibility of syntactic categories.
University of Cambridge PhD dissertation, 2019.

(Email me for other references.)

Appendix: More about granularity

The following slides elaborate on the two big-picture directions:

- Relation to typology (granularity level space)
- Relation to I-language (mental granularity)

Granularity and typology

Granularity levels and languages define each other

In theory there are as many granularity levels as there are potential natural language varieties.

- Each language variety has a categorial inventory.
- Each categorial inventory defines a granularity level.
- So each language variety defines a granularity level. (And vice versa.)

Granularity levels and languages define each other

In theory there are as many granularity levels as there are potential natural language varieties.

- Each language variety has a categorial inventory.
- Each categorial inventory defines a granularity level.
- So each language variety defines a granularity level. (And vice versa.)

Granularity level space (GLS)

The totality of all possible granularity levels for human language syntax can be conceived as a set, called the *granularity level space*.

Granularity levels and languages define each other

In theory there are as many granularity levels as there are potential natural language varieties.

- Each language variety has a categorial inventory.
- Each categorial inventory defines a granularity level.
- So each language variety defines a granularity level. (And vice versa.)

Granularity level space (GLS)

The totality of all possible granularity levels for human language syntax can be conceived as a set, called the *granularity level space*.

This is more precisely a *partially ordered set*.[5] Given two granularity levels Γ and Γ', if Γ is less fine-grained than or equal to Γ' we can write $\Gamma \leqslant \Gamma'$.

- so the standard MP granularity $\leqslant$ the cartographic granularity

[5] A partially ordered set or poset is a set with a partial order; i.e., a binary relation that is reflexive, transitive, and antisymmetric.

Metagranularity

Depending on how serious we are about distinguishing separate language varieties—e.g.,

- Do East London English and Central London English count as two varieties?
- Do Donald Trump's English and Barack Obama's English count as two varieties?—

we can perhaps view the notion of granularity itself through the lens of granularity (call this "metagranularity").

Metagranularity

Depending on how serious we are about distinguishing separate language varieties—e.g.,

- Do East London English and Central London English count as two varieties?
- Do Donald Trump's English and Barack Obama's English count as two varieties?—

we can perhaps view the notion of granularity itself through the lens of granularity (call this "metagranularity").

- At a coarser metagranularity level, East London English and Central London English (or even English as a whole) count as a single language.
- At a (much) finer metagranularity level Trump and Obama have their respective idiolects.

Metagranularity

There is a sequence of metagranularity levels:

- At the finest level each idiolect is a language variety.
- At the coarsest level (for us) Humanese is just one language variety.

> *According to Chomsky, a visiting Martian scientist would surely conclude that aside from their mutually unintelligible vocabularies, Earthlings speak a single language.*
>
> —Pinker (1994:232)

> *To put it whimsically, the Martian language might not be so different from human language after all.*
>
> —Chomsky (2018, METI symposium; source: cnet.com)

Granularity and I-language

Granularity in I-language

So far we have mainly looked at granularity variation from an analyst's perspective. Does granularity also have a place in I-language, where speakers do not have the kind of cross-granularity perspective that linguists have?

Granularity in I-language

So far we have mainly looked at granularity variation from an analyst's perspective. Does granularity also have a place in I-language, where speakers do not have the kind of cross-granularity perspective that linguists have?

Yes. At a specific point (or period) in time, a speaker's I-language only has a particular granularity level, because it only has a particular inventory of categories.

The granularity level of a speaker's I-language may change **over time**, especially during its development/maturing process.

Granularity in I-language

The change of granularity in I-language is usually **increasing**.

Consider Biberauer & Roberts' (2015) category subtyping hierarchy:

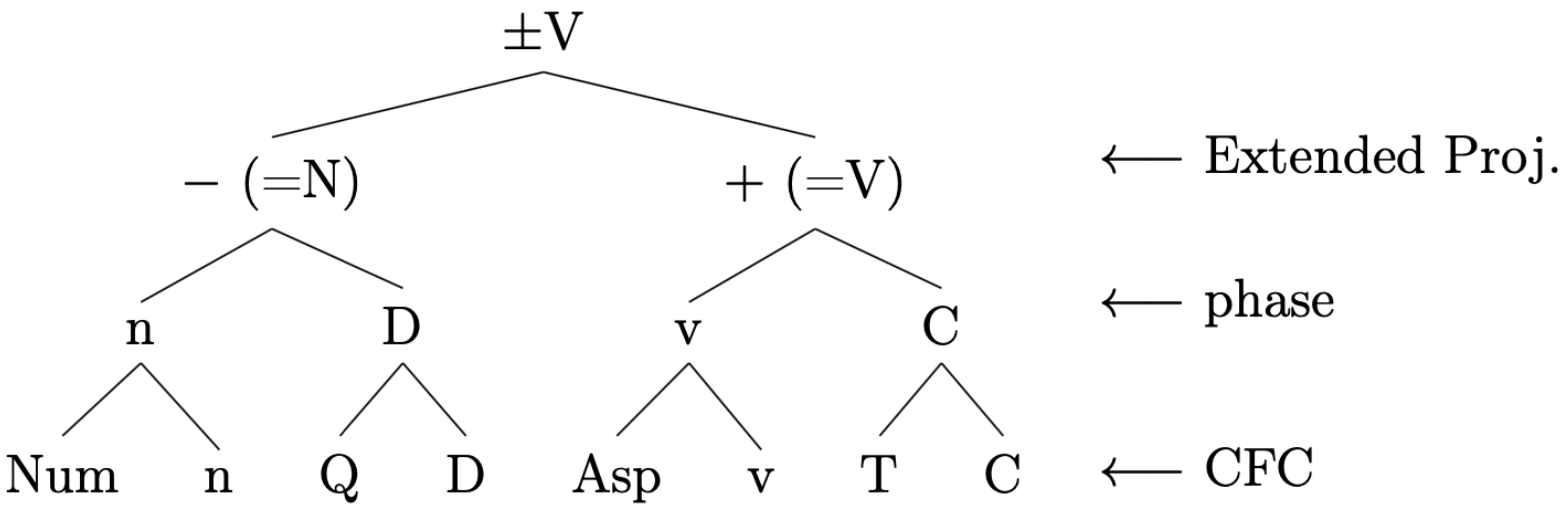
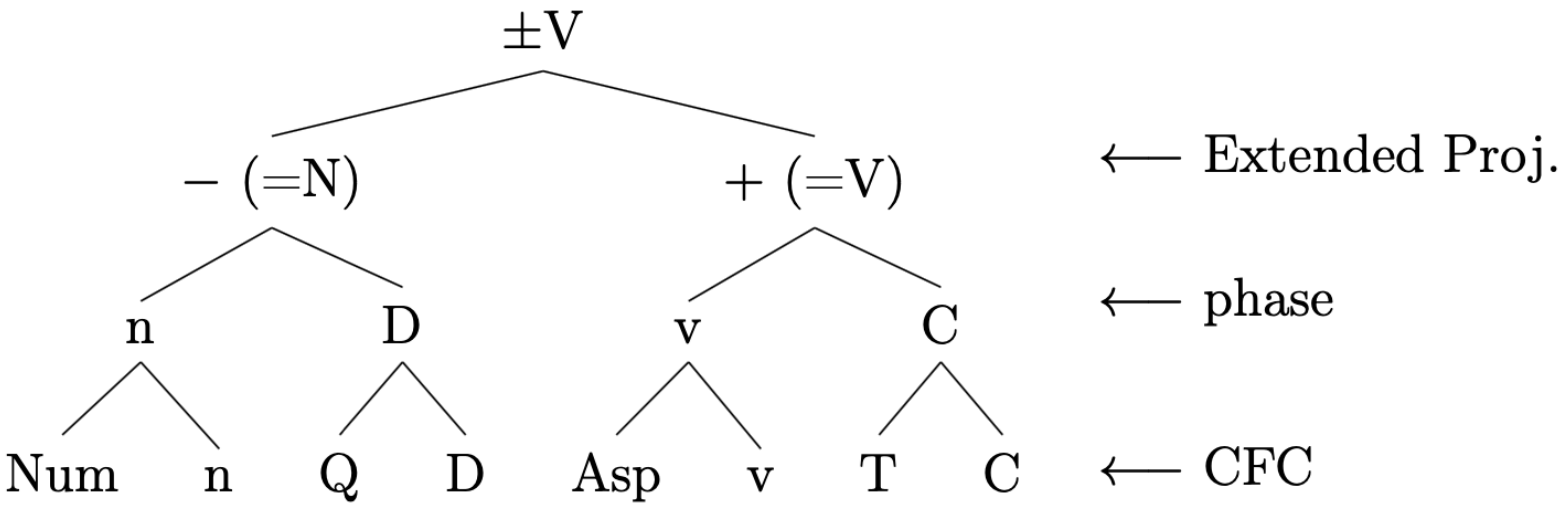


Figure 2: B&R's (2015) different levels of "magnification" for syntactic analyses

Granularity in I-language



In our partial order notation:

$$\Gamma_{EP} \leqslant \Gamma_{Ph} \leqslant \Gamma_{CFC}$$

Each Γ corresponds to a row in B&R's hierarchy, so (an abstracted form of) the B&R hierarchy is a corner in the GLS.

Broad vs. narrow granularity

It makes sense to talk about granularity in an I-language discourse even though speakers lack the linguist's viewpoint, because the generalized notion of granularity level no longer depends on such a viewpoint; it is just a synonym for categorial inventory instead.

The term granularity merely highlights the fact that the various categorial inventories for natural languages are interconnected to one another.

Broad vs. narrow granularity

It makes sense to talk about granularity in an I-language discourse even though speakers lack the linguist's viewpoint, because the generalized notion of granularity level no longer depends on such a viewpoint; it is just a synonym for categorial inventory instead.

The term granularity merely highlights the fact that the various categorial inventories for natural languages are interconnected to one another.

To avoid misunderstanding, I call this generalized granularity **granularity in the broad sense**; accordingly, the originally conceived, comparison-based granularity is **granularity in the narrow sense**.

Granularity in I-language

Individual speakers are naturally equipped with granularity levels in the broad sense, while linguists can reason about all sorts of granularity levels, either broad or narrow.

We can call granularity in the speaker's mind **mental granularity** and granularity in linguists' practice **analytical granularity**.

Granularity in I-language

Individual speakers are naturally equipped with granularity levels in the broad sense, while linguists can reason about all sorts of granularity levels, either broad or narrow.

We can call granularity in the speaker's mind **mental granularity** and granularity in linguists' practice **analytical granularity**.

The granularity level space contains both mental and analytical granularity levels, because both are just sets of syntactic categories.

(See Song 2019 for the mathematical structure of the GLS.)