A model-theoretic framework for grammaticality judgements

Denys Duchier, Jean-Philippe Prost, Thi-Bich-Hanh Dao

A Model-Theoretic Framework for Grammaticality Judgements

Denys Duchier, Jean-Philippe Prost, and Thi-Bich-Hanh Dao

LIFO, Université d'Orléans

Abstract. Although the observation of grammaticality judgements is well acknowledged, their formal representation faces problems of different kinds: linguistic, psycholinguistic, logical, computational. In this paper we focus on addressing some of the logical and computational aspects, relegating the linguistic and psycholinguistic ones in the parameter space. We introduce a model-theoretic interpretation of Property Grammars, which lets us formulate numerical accounts of grammaticality judgements. Such a representation allows for both clear-cut binary judgements, and graded judgements. We discriminate between problems of Intersective Gradience (*i.e.*, concerned with choosing the syntactic category of a model among a set of candidates) and problems of Subsective Gradience (*i.e.*, concerned with estimating the degree of grammatical acceptability of a model). Intersective Gradience is addressed as an optimisation problem, while Subsective Gradience is addressed as an approximation problem.

1 Introduction

Model-Theoretic Syntax (MTS) fundamentally differs from proof-theoretic syntax (or Generative-Enumerative Syntax—GES—as coined by Pullum and Scholz [1]) in the way of representing language: while GES focuses on describing a procedure to generate by enumeration the set of all the legal strings in the language, MTS abstracts away from any specific procedure and focuses on describing individual syntactic properties of language. While the syntactic representation of a string is, in GES, the mere trace of the generative procedure, in MTS it is a model for the grammar, with no information as to how such a model might be obtained. The requirement to be a *model for the grammar* is to satisfy the set of all unordered grammatical constraints.

When compared with GES, the consequences in terms of coverage of linguistic phenomena is significant. Pullum and Scholz have shown that a number of phenomena, which are not accounted for by GES, are well covered in MTS frameworks. Most noticeably, *quasi-expressions*[1] and graded grammatical-

[1] The term *quasi-expression* was coined by Pullum and Scholz [1] in order to refer to those utterances of a natural language, which are not completely well-formed, yet show some form of syntactic structure and properties. In contrast, *expressions* refer to well-formed utterances, that is, utterances which strictly meet all the grammatical requirements. We adopt here the same terminology; we will use *utterance* to refer to either an expression or a quasi-expression.

ity judgements are only covered by MTS. Yet there exists no logical formulation for such graded grammaticality judgements, although they are made theoretically possible by MTS. This paper proposes such a formulation, based on the model of gradience implemented by Prost [2].

Our contribution is 3-fold: first and foremost, we offer precise model-theoretic semantics for property grammars; we then extend it to permit *loose* models for deviant utterances; and finally we use this formal apparatus to devise scoring functions that can be tuned to agree well with natural comparative judgements of grammaticality.

While Prost [2] proposed a framework for gradience and a parsing algorithm for possibly deviant utterances, his formalization was not entirely satisfactory; among other things, his models were not trees, but technical devices suggested by his algorithmic approach to parsing. Our proposal takes a rather different angle; our models are trees of syntactic categories; our formalization is fully worked out and was designed for easy conversion to constraint programming.

The notions of gradience that underly our approach are described in section 2; property grammars are introduced in section 3; their strong semantics are developed in section 4; their loose semantics in section 5; section 6 presents the postulates that inform our modelization of acceptability judgements, and section 7 provides its quantitative formalization.

2 Gradience

Aarts [3] proposes to discriminate the problems concerned with gradience in two different families: those concerned with *Intersective Gradience* (IG), and those concerned with *Subsective Gradience* (SG). In reference to Set Theory, IG refers to the problem of choosing which category an item belongs to among a set of candidates, while SG refers to the problem of estimating to what extent an item is prototypical within the category it belongs to. Applied here, we regard the choice of a model for an utterance (*i.e.* expression or quasi-expression) as a problem of IG, while the estimation of a degree of grammatical acceptability for a model is regarded as a problem of SG.

For example, Fig 1 illustrates a case of IG with a set of possible parses for a quasi-expression. In that case the preferred model is the first one. The main reason is that, unlike the other ones, it is rooted with the category S.

Fig 2 illustrates different sentences ordered by decreasing grammatical acceptability. Each given judgement corresponds to a (human) estimate of how acceptable it is compared with the reference expression 1.

Fig 3 gives models for quasi-expressions 2 (QE2) and 5 (QE5) from Fig.2. We observe that the model for QE2 is rooted by S, while the one for QE5 is rooted by Star (wildcard category). QE5, unlike QE2, is essentially and crucially missing a VP. QE5 is also unexpectedly terminated with a P. QE2, on the other hand, is only missing a determiner for introducing *rapport*, since it is a requirement in French for a noun to be introduced by a determiner. For all these reasons, the model for QE5 is judged more ungrammatical than the one for QE2.

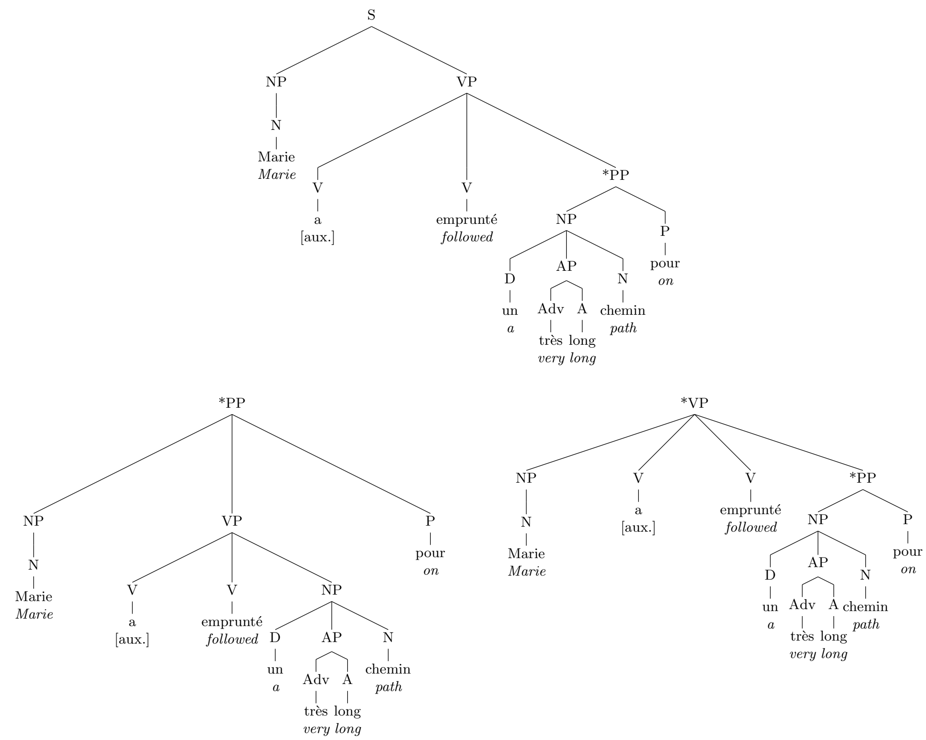

Fig. 1. Intersective Gradience: possible models for the French quasi-expression *Marie a emprunté un très long chemin pour*

1. Les employés ont rendu un rapport très complet à leur employeur [100%]
   The employees have sent a report very complete to their employer
2. Les employés ont rendu rapport très complet à leur employeur [92.5%]
   The employees have sent report very complete to their employer
3. Les employés ont rendu un rapport très complet à [67.5%]
   The employees have sent a report very complete to
4. Les employés un rapport très complet à leur employeur [32.5%]
   The employees a report very complete to their employer
5. Les employés un rapport très complet à [5%]
   The employees a report very complete to their employer

Fig. 2. Sentences of decreasing acceptability

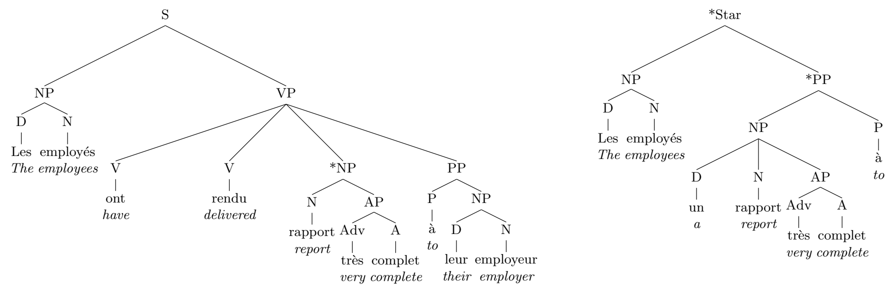

Fig. 3. Models for the quasi-expressions 2. and 5. from Fig.2

We will come back shortly to the precise meaning of *model*. For the moment, let us just say that a model is a syntactic representation of an utterance. Intuitively, the syntactic representation of an expression can easily be grasped, but it is more problematic in case of a quasi-expression. What we propose in that case, is to *approximate* models, then to choose the optimal one(s). The numeric criterion to be optimised may take different forms ; we decide to maximise the proportion of grammatical constraints satisfied by the model. Once the problem of IG is solved, we can then make a grammaticality judgement on that model and estimate a degree of acceptability for it. We propose that that estimate be based on different psycholinguistic hypotheses regarding factors of influence in a grammaticality judgement. We propose a formulation for each of them, and for combining them into a single score for the model.

3 Property Grammars

The framework for gradience which we propose is formulated in terms of Property Grammars [4]. Property Grammars are appealing for modeling deviant utterances because they break down the notion of grammaticality into many small constraints (properties) which may be independently violated.

Property Grammars are perhaps best understood as the transposition of phrase structure grammars from the GES perspective into the MTS perspective. Let's consider a phrase structure grammar expressed as a collection of rules. For our purpose, we assume that there is exactly one rule per non-terminal, and that rule bodies may be disjunctive to allow alternate realizations of the same non-terminal. In the GES perspective, such a grammar is interpreted as a generator of strings.

It is important to recognize that the same grammar can be interpreted in the MTS perspective: its models are all the syntax trees whose roots are labeled with the axiom category and such that every rule is satisfied at every node. For example, we say that the rule `NP` $\rightarrow$ `D N` is satisfied at a node if either the node is not labeled with `NP`, or it has exactly two children, the first one labeled with `D` and the second one labeled with `N`.

In this manner, rules have become constraints and a phrase structure grammar can be given model-theoretical semantics by interpretation over syntax tree structures. However these constraints remain very coarse-grained: for example, the rule $\texttt{NP} \rightarrow \texttt{D N}$ simultaneously stipulates that for a `NP`, there must be (1) a `D` child and (2) only one, (3) a `N` child and (4) only one, (5) nothing else and (6) that the `D` child must precede the `N` child.

Property grammars explode rules into such finer-grained constraints called *properties*. They have the form $A : \psi$ meaning in an A, the constraint ψ applies to its children (its constituents). The usual types of properties are:

obligation	$A : \triangle B$	at least one B child
uniqueness	$A : B!$	at most one B child
linearity	$A : B \prec C$	a B child precedes a C child
requirement	$A : B \Rightarrow C$	if there is a B child, then also a C child
exclusion	$A : B \not\Leftrightarrow C$	B and C children are mutually exclusive
constituency	$A : S?$	the category of any child must be one in S

For the rule $\texttt{NP} \rightarrow \texttt{D N}$ studied above, stipulation (1) would be expressed by a property of *obligation* $\texttt{NP} : \triangle\texttt{D}$, similarly stipulation (3) by $\texttt{NP} : \triangle\texttt{N}$, stipulation (2) by a property of uniqueness $\texttt{NP} : \texttt{D}!$, similarly stipulation (4) by $\texttt{NP} : \texttt{N}!$, stipulation (5) by a property of *constituency* $\texttt{NP} : \{\texttt{D}, \texttt{N}\}?$, and stipulation (6) by a property of *linearity* $\texttt{NP} : \texttt{D} \prec \texttt{N}$.

In other publications, property grammars are usually displayed as a collection of boxes of properties. For example, Table 1 contains the property grammar for French that is used in [2]. The present article deviates from the usual presentation in four ways. First, in the interest of brevity, we do not account for features though this would pose no formal problems. Consequently, second: we omit the *dependency* property. Third, we make the constituency property explicit. Fourth, our notation is different: the `S` box is transcribed as the following set of property literals: $\texttt{S} : \triangle\texttt{VP}$, $\texttt{S} : \texttt{NP}!$, $\texttt{S} : \texttt{VP}!$, and $\texttt{S} : \texttt{NP} \prec \texttt{VP}$.

4 Strong semantics

Property grammars. Let $\mathcal{L}$ be a finite set of labels representing syntactic categories. We write $\mathcal{P}_\mathcal{L}$ for the set of all possible property literals over $\mathcal{L}$ formed $\forall c_0, c_1, c_2 \in \mathcal{L}$ in any of the following 6 ways:

$$c_0 : c_1 \prec c_2, \quad c_0 : \triangle c_1, \quad c_0 : c_1!, \quad c_0 : c_1 \Rightarrow c_2, \quad c_0 : c_1 \not\Leftrightarrow c_2, \quad c_0 : s_1?$$

Let $\mathcal{S}$ be a set of elements called *words*. A lexicon is a subset of $\mathcal{L} \times \mathcal{S}$.[2] A *property grammar* G is a pair (P_G, L_G) where P_G is a set of properties (a subset of $\mathcal{P}_\mathcal{L}$) and L_G is a lexicon.

[2] We restricted ourselves to the simplest definition sufficient for this presentation.

S (Utterance)	
obligation	: $\triangle \texttt{VP}$
uniqueness	: $\texttt{NP!}$
	: $\texttt{VP!}$
linearity	: $\texttt{NP} \prec \texttt{VP}$
dependency	: $\texttt{NP} \rightsquigarrow \texttt{VP}$

AP (Adjective Phrase)	
obligation	: $\triangle(\texttt{A} \vee \texttt{V}_{[\text{past part}]})$
uniqueness	: $\texttt{A!}$
	: $\texttt{V}_{[\text{past part}]}!$
	: $\texttt{Adv!}$
linearity	: $\texttt{A} \prec \texttt{PP}$
	: $\texttt{Adv} \prec \texttt{A}$
exclusion	: $\texttt{A} \not\Leftrightarrow \texttt{V}_{[\text{past part}]}$

PP (Propositional Phrase)	
obligation	: $\triangle \texttt{P}$
uniqueness	: $\texttt{P!}$
	: $\texttt{NP!}$
linearity	: $\texttt{P} \prec \texttt{NP}$
	: $\texttt{P} \prec \texttt{VP}$
requirement	: $\texttt{P} \Rightarrow \texttt{NP}$
dependency	: $\texttt{P} \rightsquigarrow \texttt{NP}$

NP (Noun Phrase)	
obligation	: $\triangle(\texttt{N} \vee \texttt{Pro})$
uniqueness	: $\texttt{D!}$
	: $\texttt{N!}$
	: $\texttt{PP!}$
	: $\texttt{Pro!}$
linearity	: $\texttt{D} \prec \texttt{N}$
	: $\texttt{D} \prec \texttt{Pro}$
	: $\texttt{D} \prec \texttt{AP}$
	: $\texttt{N} \prec \texttt{PP}$
requirement	: $\texttt{N} \Rightarrow \texttt{D}$
	: $\texttt{AP} \Rightarrow \texttt{N}$
exclusion	: $\texttt{N} \not\Leftrightarrow \texttt{Pro}$
dependency	: $\texttt{N}\begin{bmatrix}\text{GEND} & \boxed{1} \\ \text{NUM} & \boxed{2}\end{bmatrix} \rightsquigarrow \texttt{D}\begin{bmatrix}\text{GEND} & \boxed{1} \\ \text{NUM} & \boxed{2}\end{bmatrix}$

VP (Verb Phrase)	
obligation	: $\triangle \texttt{V}$
uniqueness	: $\texttt{V}_{[\text{main past part}]}!$
	: $\texttt{NP!}$
	: $\texttt{PP!}$
linearity	: $\texttt{V} \prec \texttt{NP}$
	: $\texttt{V} \prec \texttt{Adv}$
	: $\texttt{V} \prec \texttt{PP}$
requirement	: $\texttt{V}_{[\text{past part}]} \Rightarrow \texttt{V}_{[\text{aux}]}$
exclusion	: $\texttt{Pro}_{[\text{acc}]} \not\Leftrightarrow \texttt{NP}$
	: $\texttt{Pro}_{[\text{dat}]} \not\Leftrightarrow \texttt{Pro}_{[\text{acc}]}$
dependency	: $\texttt{V}\begin{bmatrix}\text{PERS} & \boxed{1} \\ \text{NUM} & \boxed{2}\end{bmatrix} \rightsquigarrow \texttt{Pro}\begin{bmatrix}\text{TYPE} & \text{pers} \\ \text{CASE} & \text{nom} \\ \text{PERS} & \boxed{1} \\ \text{NUM} & \boxed{2}\end{bmatrix}$

Table 1. Example property grammar for French

Class of models. The strong semantics of property grammars are given by interpretation over the class of syntax tree structures defined below.

We write $\mathbb{N}_0$ for $\mathbb{N} \setminus \{0\}$. A *tree domain* D is a finite subset of $\mathbb{N}_0^*$ which is closed for prefixes and for left-siblings; in other words it satisfies:

$$\begin{array}{lr} \forall \pi, \pi' \in \mathbb{N}_0^* & \pi\pi' \in D \;\Rightarrow\; \pi \in D \\ \forall \pi \in \mathbb{N}_0^*,\ \forall i, j \in \mathbb{N}_0 & i < j \wedge \pi j \in D \;\Rightarrow\; \pi i \in D \end{array}$$

A syntax tree $\tau = (D_\tau, L_\tau, R_\tau)$ consists of a tree domain D_τ, a labeling function $L_\tau : D_\tau \to \mathcal{L}$ assigning a category to each node, and a function $R_\tau : D_\tau \to \mathcal{S}^*$ assigning to each node its surface realization.

For convenience, we define the arity function $A_\tau : D_\tau \to \mathbb{N}$ as follows, $\forall \pi \in D_\tau$:

$$A_\tau(\pi) = \max \{0\} \cup \{i \in \mathbb{N}_0 \mid \pi i \in D_\tau\}$$

Instances. A property grammar G stipulates a set of properties. For example the property $c_0 : c_1 \prec c_2$ is intended to mean that, for a non-leaf node of category c_0, and any two daughters of this node labeled respectively with categories c_1 and c_2, then the one labeled with c_1 must precede the one labeled with c_2. Clearly, for each node of category c_0, this property must be checked for every pair of daughters of said node. Thus, we arrive at the notion of instances of a property.

An instance of a property is a pair of a property and a tuple of nodes (paths) to which it is applied. We define the property instances of a grammar G on a syntax tree τ as follows:

$$\begin{aligned}
\mathcal{I}_\tau[\![G]\!] &= \cup\{\mathcal{I}_\tau[\![p]\!] \mid \forall p \in P_G\} \\
\mathcal{I}_\tau[\![c_0 : c_1 \prec c_2]\!] &= \{(c_0 : c_1 \prec c_2)@\langle \pi, \pi i, \pi j\rangle \mid \forall \pi, \pi i, \pi j \in D_\tau,\ i \neq j\} \\
\mathcal{I}_\tau[\![c_0 : \triangle c_1]\!] &= \{(c_0 : \triangle c_1)@\langle \pi\rangle \mid \forall \pi \in D_\tau\} \\
\mathcal{I}_\tau[\![c_0 : c_1 !]\!] &= \{(c_0 : c_1 !)@\langle \pi, \pi i, \pi j\rangle \mid \forall \pi, \pi i, \pi j \in D_\tau,\ i \neq j\} \\
\mathcal{I}_\tau[\![c_0 : c_1 \Rightarrow c_2]\!] &= \{(c_0 : c_1 \Rightarrow c_2)@\langle \pi, \pi i\rangle \mid \forall \pi, \pi i \in D_\tau\} \\
\mathcal{I}_\tau[\![c_0 : c_1 \not\Leftrightarrow c_2]\!] &= \{(c_0 : c_1 \not\Leftrightarrow c_2)@\langle \pi, \pi i, \pi j\rangle \mid \forall \pi, \pi i, \pi j \in D_\tau,\ i \neq j\} \\
\mathcal{I}_\tau[\![c_0 : s_1 ?]\!] &= \{(c_0 : s_1 ?)@\langle \pi, \pi i\rangle \mid \forall \pi, \pi i \in D_\tau\}
\end{aligned}$$

Pertinence. Since we created instances of all properties in P_G for all nodes in τ, we must distinguish properties which are truly pertinent at a node from those which are not. For this purpose, we define the predicate P_τ over instances as follows:

$$\begin{aligned}
P_\tau((c_0 : c_1 \prec c_2)@\langle \pi, \pi i, \pi j\rangle) &\equiv L_\tau(\pi) = c_0 \ \wedge\ L_\tau(\pi i) = c_1 \ \wedge\ L_\tau(\pi j) = c_2 \\
P_\tau((c_0 : \triangle c_1)@\langle \pi\rangle) &\equiv L_\tau(\pi) = c_0 \\
P_\tau((c_0 : c_1 !)@\langle \pi, \pi i, \pi j\rangle) &\equiv L_\tau(\pi) = c_0 \ \wedge\ L_\tau(\pi i) = c_1 \ \wedge\ L_\tau(\pi j) = c_1 \\
P_\tau((c_0 : c_1 \Rightarrow c_2)@\langle \pi, \pi i\rangle) &\equiv L_\tau(\pi) = c_0 \ \wedge\ L_\tau(\pi i) = c_1 \\
P_\tau((c_0 : c_1 \not\Leftrightarrow c_2)@\langle \pi, \pi i, \pi j\rangle) &\equiv L_\tau(\pi) = c_0 \ \wedge\ (L_\tau(\pi i) = c_1 \ \vee\ L_\tau(\pi j) = c_2) \\
P_\tau((c_0 : s_1 ?)@\langle \pi, \pi i\rangle) &\equiv L_\tau(\pi) = c_0
\end{aligned}$$

Satisfaction. When an instance is pertinent, it should also (preferably) be satisfied. For this purpose, we define the predicate S_τ over instances as follows:

$$\begin{aligned}
S_\tau((c_0 : c_1 \prec c_2)@\langle \pi, \pi i, \pi j\rangle) &\equiv i < j \\
S_\tau((c_0 : \triangle c_1)@\langle \pi\rangle) &\equiv \vee\{L_\tau(\pi i) = c_1 \mid 1 \leq i \leq A_\tau(\pi)\} \\
S_\tau((c_0 : c_1 !)@\langle \pi, \pi i, \pi j\rangle) &\equiv i = j \\
S_\tau((c_0 : c_1 \Rightarrow c_2)@\langle \pi, \pi i\rangle) &\equiv \vee\{L_\tau(\pi j) = c_2 \mid 1 \leq j \leq A_\tau(\pi)\} \\
S_\tau((c_0 : c_1 \not\Leftrightarrow c_2)@\langle \pi, \pi i, \pi j\rangle) &\equiv L_\tau(\pi i) \neq c_1 \ \vee\ L_\tau(\pi j) \neq c_2 \\
S_\tau((c_0 : s_1 ?)@\langle \pi, \pi i\rangle) &\equiv L_\tau(\pi i) \in s_1
\end{aligned}$$

We write $I^0_{G,\tau}$ for the set of pertinent instances of G in τ, $I^+_{G,\tau}$ for its subset that is satisfied, and $I^-_{G,\tau}$ for its subset that is violated:

$$\begin{aligned}
I^0_{G,\tau} &= \{r \in \mathcal{I}_\tau[\![G]\!] \mid P_\tau(r)\} \\
I^+_{G,\tau} &= \{r \in I^0_{G,\tau} \mid S_\tau(r)\} \\
I^-_{G,\tau} &= \{r \in I^0_{G,\tau} \mid \neg S_\tau(r)\}
\end{aligned}$$

Admissibility. A syntax tree τ is admissible as a candidate model for grammar G iff it satisfies the projection property,[3] i.e. $\forall \pi \in D_\tau$:

$$\begin{aligned} A_\tau(\pi) = 0 \quad &\Rightarrow \quad \langle L_\tau(\pi), R_\tau(\pi) \rangle \in L_G \\ A_\tau(\pi) \neq 0 \quad &\Rightarrow \quad R_\tau(\pi) = \sum_{i=1}^{i=A_\tau(\pi)} R_\tau(\pi i) \end{aligned}$$

where $\sum$ represents here the concatenation of sequences. In other words: leaf nodes must conform to the lexicon, and interior nodes pass upward the ordered realizations of their daughters. We write $\mathcal{A}_G$ for the set of admissible syntax trees for grammar G.

Strong models. A syntax tree τ is a strong model of a property grammar G iff it is admissible and $I^-_{G,\tau} = \emptyset$. We write $\tau : \sigma \models G$ iff τ is a strong model of G with realization σ, i.e. such that $R_\tau(\varepsilon) = \sigma$.

5 Loose Semantics

Since property grammars are intended to also account for deviant utterances, we must define alternate semantics that accommodate deviations from the *strong* interpretation. The *loose semantics* will allow some property instances to be violated, but will seek syntax trees which maximize the overall *fitness* for a specific utterance.

Admissibility. A syntax tree τ is loosely admissible for utterance σ iff it is admissible and its realization is $\sigma = R_\tau(\epsilon)$. We write $\mathcal{A}_{G,\sigma}$ for the loosely admissible syntax trees for utterance σ:

$$\mathcal{A}_{G,\sigma} = \{\tau \in \mathcal{A}_G \mid R_\tau(\epsilon) = \sigma\}$$

Following Prost [2], we define fitness as the ratio of satisfied pertinent instances over the total number of pertinent instances:

$$F_{G,\tau} = I^+_{G,\tau} / I^0_{G,\tau}$$

The loose models for an utterance σ are all loosely admissible models for utterance σ that maximize fitness:

$$\tau : \sigma \mathrel{|\!\!\approx} G \quad \text{iff} \quad \tau \in \underset{\tau' \in \mathcal{A}_{G,\sigma}}{\operatorname{argmax}}(F_{G,\tau'})$$

[3] It should be noted that Prost [2] additionally requires that all constituency properties be satisfied.

6 Modeling Judgements of Acceptability

We now turn to the issue of modeling natural judgements of acceptability. We hypothesize that an estimate for acceptability can be predicted by quantitative factors derivable from the loose model of an utterance. To that end, we must decide what factors and how to combine them. The answers we propose are informed by the 5 postulates outlined below.

Postulates 1, 2, and 3 are substantiated by empirical evidence and work in the fields of Linguistics and Psycholinguistics, but postulates 4 and 5 are speculative. While the factors we consider here are all syntactic in nature, it is clear that a complete model for human judgements of acceptability should also draw on other dimensions of language (semantics, pragmatics, . . .).

Postulate 1 (Failure Cumulativity) *Gradience is impacted by constraint failures; that is, an utterance's acceptability is impacted by the number of constraints it violates.*

This factor is probably the most intuitive, and is the most commonly found in the literature [5, 6]. It corresponds to Keller's *cumulativity effect*, substantiated by empirical evidence.

Postulate 2 (Success Cumulativity) *Gradience is impacted by constraint successes; that is, an utterance acceptability is impacted by the number of constraints it satisfies.*

Different works suggest that acceptability judgements can also be affected by successful constraints [7–9, 3]. The underlying intuition is that failures alone are not sufficient to account for acceptability, hence the postulate that some form of interaction between satisfied and violated constraints contributes to the judgements. It significantly differs from other accounts of syntactic gradience, which only rely on constraint failures (*e.g.* Keller's LOT, or Schröder's WCDG).

Postulate 3 (Constraint Weighting) *Acceptability is impacted to a different extent according to which constraint is satisfied or violated.*

Here we postulate that constraints are weighted according to their influence on acceptability. This intuition is commonly shared in the literature[4] [8–13, 5, 14, 15], and supported by empirical evidence. In this paper, we make the simplifying assumption that a constraint weighs the same whether it is satisfied or violated, but we could just as well accommodate different weights.

Strategies for assigning weights may be guided by different considerations of *scope* and *granularity*. Scope: should weights be assigned to constraint individually or by constraint type. Granularity: should weights be decided globally for the grammar, or separately for each syntactic category.

[4] Note that these constraint weights take different meanings in different works. Many of them, for instance, have a statistical component that we do not have here.

Scope and granularity can then be combined in different ways: all constraints of the same type at the grammar level, or all constraints of the same type at the construction level, or individual constraints at the construction level, or individual constraints at the grammar level. The difference between the last two possibilities assumes that the same constraint may occur in the specification of more than one construction.

Arguably, the finer the granularity and the narrower the scope, the more flexibility and accuracy we get but at a significant cost in maintenance. This high cost is confirmed by Schröder [6], who opted for weights being assigned to each individual constraint at the grammar level. Prost [2] opted for a compromise, where the weighting scheme is restricted to the constraint types at the grammar level, which means that all constraints of the same type in the grammar are assigned the same weight. For example, all the Linearity constraints (*i.e.* word order) are weighted 20, all the Obligation constraints (*i.e.* heads) are weighted 10, and so on.

While the strategy of weight assignment is of considerable methodological import, for the purpose of the present logical formulation it is sufficient to suppose given a function that maps each constraint to its weight.

Postulate 4 (Constructional complexity) *Acceptability is impacted by the* complexity *of the constituent structure.*

How to precisely measure and capture the complexity of an utterance is an open question, which we do not claim to fully answer. In fact, this factor of influence probably ought to be investigated in itself, and split into more fine-grained postulates with respect to acceptability and syntactic gradience. Different works from Gibson [16, 12] could be used as a starting point for new postulates in this regard. Here we simply measure the complexity of the category a constituent belongs to as the number of constraints specifying this category in the grammar. This postulate aims to address, among others, the risk of disproportionate convergence. The underlying idea is to balance the number of violations with the number of specified constraints: without such a precaution a violation in a rather simple construction, such as AP, would be proportionally much more costly than a violation in a rather complex construction, such as NP.

Postulate 5 (Propagation) *Acceptability is propagated through the dominance relationships; that is, an utterance's acceptability depends on its nested constituents' acceptability.*

Here it is simply postulated that the nested constituents' acceptability is recursively propagated to their dominant constituent.

7 Formalizing Judgements of Acceptability

Following the previous discussion, we define a *weighted property grammar* G as a triple (P_G, L_G, ω_G) where (P_G, L_G) is a property grammar and $\omega_G : P_G \to \mathbb{R}$ is a function assigning a weight to each property.

Since our approach relies on quantitative measurements of satisfactions and violations, it must be formulated in terms of property instances. Furthermore, postulates 4 and 5 require the computation of local quantitative factors at each node in the model. For this reason, we need to identify, in the set of all property instances the subset which applies at a given node.

For each property instance r, we write $\mathsf{at}(r)$ for the node where it applies; and we define it by cases $\forall p \in \mathcal{P}_{\mathcal{L}},\ \forall \pi_0, \pi_1, \pi_2 \in \mathbb{N}_0^*$ as follows:

$$\mathsf{at}(p@\langle \pi_0 \rangle) = \pi_0 \qquad \mathsf{at}(p@\langle \pi_0, \pi_1 \rangle) = \pi_0 \qquad \mathsf{at}(p@\langle \pi_0, \pi_1, \pi_2 \rangle) = \pi_0$$

If B is a set of instances, then $B|_\pi$ is the subset of B of all instances applying at node π:

$$B|_\pi = \{r \in B \mid \mathsf{at}(r) = \pi\}$$

We now define the sets of instances pertinent, satisfied, and violated at node π:

$$I^0_{G,\tau,\pi} = I^0_{G,\tau}|_\pi \qquad I^+_{G,\tau,\pi} = I^+_{G,\tau}|_\pi \qquad I^-_{G,\tau,\pi} = I^-_{G,\tau}|_\pi$$

which allow us to express the cummulative weights of pertinent, satisfied, and violated instances at node π:

$$W^0_{G,\tau,\pi} = \sum \{\omega_G(x) \mid \forall x@y \in I^0_{G,\tau,\pi}\}$$
$$W^+_{G,\tau,\pi} = \sum \{\omega_G(x) \mid \forall x@y \in I^+_{G,\tau,\pi}\}$$
$$W^-_{G,\tau,\pi} = \sum \{\omega_G(x) \mid \forall x@y \in I^-_{G,\tau,\pi}\}$$

Following Prost [2], we define at each node π the quality index $W_{G,\tau,\pi}$, the satisfaction ratio $\rho^+_{G,\tau,\pi}$, and the violation ratio $\rho^-_{G,\tau,\pi}$:

$$W_{G,\tau,\pi} = \frac{W^+_{G,\tau,\pi} - W^-_{G,\tau,\pi}}{W^+_{G,\tau,\pi} + W^-_{G,\tau,\pi}} \qquad \rho^+_{G,\tau,\pi} = \frac{|I^+_{G,\tau,\pi}|}{|I^0_{G,\tau,\pi}|} \qquad \rho^-_{G,\tau,\pi} = \frac{|I^-_{G,\tau,\pi}|}{|I^0_{G,\tau,\pi}|}$$

According to postulate 4, we must take into account the complexity of a construction: is it specified by many properties or by few? For each node π, we look up the set of properties $T_{G,\tau,\pi}$ that are used in the grammar to specify the category $L_\tau(\pi)$ of π:

$$T_{G,\tau,\pi} = \{c : C \in P_G \mid L_\tau(\pi) = c\}$$

and we use it to define a completeness index $C_{G,\tau,\pi}$:

$$C_{G,\tau,\pi} = \frac{|I^0_{G,\tau,\pi}|}{|T_{G,\tau,\pi}|}$$

According to postulate 5, these quantities must be combined recursively to compute the overall rating of a model. Several rating functions have been investigated: we describe the *index of grammaticality* and the *index of coherence*.

7.1 Index of grammaticality

This scoring function is based on a local compound factor called the *index of precision* computed as follows:

$$P_{G,\tau,\pi} = kW_{G,\tau,\pi} + l\rho^{+}_{G,\tau,\pi} + mC_{G,\tau,\pi}$$

where the parameters (k, l, m) are used to tune the model. The index of grammaticality at node π is then defined inductively thus:

$$g_{G,\tau,\pi} = \begin{cases} P_{G,\tau,\pi} \cdot \frac{1}{A_\tau(\pi)} \sum_{i=1}^{A_\tau(\pi)} g_{G,\tau,\pi i} & \text{if } A_\tau(\pi) \neq 0 \\ 1 & \text{if } A_\tau(\pi) = 0 \end{cases}$$

The overall score of a loose model τ is the score $g_{G,\tau,\varepsilon}$ of its root node.

7.2 Index of coherence

This scoring function is based on a local compound factor called the *index of anti-precision* computed as follows:

$$A_{G,\tau,\pi} = kW_{G,\tau,\pi} - l\rho^{-}_{G,\tau,\pi} + mC_{G,\tau,\pi}$$

where the parameters (k, l, m) are used to tune the model. The index of coherence at node π is then defined inductively thus:

$$\gamma_{G,\tau,\pi} = \begin{cases} A_{G,\tau,\pi} \cdot \frac{1}{A_\tau(\pi)} \sum_{i=1}^{A_\tau(\pi)} \gamma_{G,\tau,\pi i} & \text{if } A_\tau(\pi) \neq 0 \\ 1 & \text{if } A_\tau(\pi) = 0 \end{cases}$$

The overall score of a loose model τ is the score $\gamma_{G,\tau,\varepsilon}$ of its root node.

7.3 Experimental Validation and Perspectives

An interesting aspect of the framework presented here is that it makes it possible to formally devise scoring functions such as those for the indexes of Grammaticality or Coherence. It is interesting because it opens the door to reasonning with graded grammaticality, in relying on numerical tools which can be validated empirically. As far as Grammaticality and Coherence are concerned, they find their origin in psycholinguistic postulates, but other kinds of justifications may just as well yield different formulations.

Of course, assigning a score to an utterance is only meaningful if supporting evidence can be put forward for validation. In the present case, the relevance of these automatic scores has been validated experimentally in Blache *et al.* [9] then in Prost [2], in measuring to what extent the scores correlate to human judgements. Human judgements of acceptability were gathered for both grammatical and ungrammatical sentences, as part of psycholinguistic experiments (reported in [9]) using the Magnitude Estimation protocol [17]. The corpus in use was artificially constructed, each sentence matching one of the 20 different error patterns. The experiment involves 44 annotators, all native speakers of French, and with no particular knowledge of linguistics. Each annotator was asked to rate, for each sentence, how much better (or worse) it was compared with a reference sentence. The figures obtained were normalised across annotators for every sentence, providing a score of human judgement for each of them.

The validation of the automatic scoring was then performed in calculating the correlation between the mean scores (automatic judgements on one hand, and human judgements on the other hand) per error pattern. The outcome is a Pearson's correlation coefficient $\rho = 0.5425$ for the Coherence score, and $\rho = 0.4857$ for the Grammaticality Index.

Interestingly as well, the same scoring functions are also opened to the modelling of graded judgements of grammaticality, as opposed to ungrammaticality. The scale of scores calculated through the Grammaticality or Coherence Indexes being open-ended, two distinct expressions can be assigned two distinct scores. We could then wonder whether, and to what extent, these scores are comparable. If comparable, then they could be interpreted as modelling the linguistic complexity of an expression. For example, we might like scores to capture that an expression with embedded relative clauses is more complex to comprehend than a simple Subject-Verb-Object construction. Unfortunately, the psycholinguistic experiment in [9] was not designed to that end (very few different constructions are in fact used for the reference sentences). Validating that hypothesis is, therefore, not possible at this stage, and must be kept for further works.

Finally, our formalization was designed for easy conversion to constraint programming and we are currently developing a prototype solver to compare its practical performance with the baseline established by Prost's parser.

8 Conclusion

While formal grammars typically limit their scope to well-formed utterances, we wish to extend their formal reach into the area of graded grammaticality. The first goal is to permit the analysis of well-formed and ill-formed utterances alike. The second more ambitious goal is to devise accurate models of natural judgements of acceptability.

In this paper, we have shown that property grammars are uniquely suited for that purpose. We contributed precise model-theoretic semantics for property grammars. Then, we relaxed these semantics to permit the loose models required by deviant utterances. Finally, we showed how formally to devise quantitative scoring functions on loose models.

Prost [2] has shown that these scoring functions can be tuned to agree well with human judgements.

Acknowledgement

This work was partly funded by the French *Agence Nationale de la Recherche*, project ANR-07-MDCO-03 (CRoTAL).

References

1. Pullum, G., Scholz, B.: On the Distinction Between Model-Theoretic and Generative-Enumerative Syntactic Frameworks. In de Groote, P., Morrill, G., Rétoré, C., eds.: Logical Aspects of Computational Linguistics: 4th International Conference. Number 2099 in Lecture Notes in Artificial Intelligence, Berlin, Springer Verlag (2001) 17–43
2. Prost, J.P.: Modelling Syntactic Gradience with Loose Constraint-based Parsing. Cotutelle Ph.D. Thesis, Macquarie University, Sydney, Australia, and Université de Provence, Aix-en-Provence, France (December 2008)
3. Aarts, B.: Syntactic gradience: the nature of grammatical indeterminacy. Oxford University Press (2007)
4. Blache, P.: Constraints Graphs and Linguistic Theory. (source unknown) (2000)
5. Keller, F.: Gradience in Grammar - Experimental and Computational Aspects of Degrees of Grammaticality. PhD thesis, University of Edinburgh (2000)
6. Schröder, I.: Natural Language Parsing with Graded Constraints. PhD thesis, Universität Hamburg (2002)
7. Aarts, B.: Modelling Linguistic Gradience. Studies in Language **28**(1) (2004) 1–49
8. Blache, P., Prost, J.P.: Gradience, Constructions and Constraint Systems. In Christiansen, H., Skadhauge, P.R., Villadsen, J., eds.: Constraint Solving and Language Processing - CSLP 2004. Volume 3438 of Lecture Notes in Artificial Intelligence (LNAI). Springer, Roskilde, Denmark (September 2004 2005) 74–89 (revised and extended version).
9. Blache, P., Hemforth, B., Rauzy, S.: Acceptability prediction by means of grammaticality quantification. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, Association for Computational Linguistics (July 2006) 57–64
10. Bresnan, J., Nikitina, T.: On the Gradience of the Dative Alternation. Draft (May 7 2003)
11. Foth, K.: Writing Weighted Constraints for Large Dependency Grammars. Proceedings of Recent Advances in Dependency Grammars, COLING-Workshop (2004)
12. Gibson, E.: The Dependency Locality Theory: A Distance-Based Theory of Linguistic Complexity. In Marantz, A., Miyashita, Y., O'Neil, W., eds.: Image, Language, Brain. Cambridge, Massachussetts, MIT Press (2000) 95–126
13. Heinecke, J., Kunze, J., Menzel, W., Shröder, I.: Eliminative Parsing with Graded Constraints. In: Proceedings 7th International Conferenceon Computational Linguistics, 36th Annual Meeting of the ACL. Volume Coling–ACL '98., Montreal, Canada (1998) pp. 526–530
14. Sorace, A., Keller, F.: Gradience in linguistic data. Lingua **115**(11) (2005) 1497–1524
15. VanRullen, T.: Vers une analyse syntaxique à granularité variable. PhD thesis, Université de Provence, Informatique (2005)
16. Gibson, E., Schütze, C.T., Salomon, A.: The relationship between the frequency and the processing complexity of linguistic structure. Journal of Psycholinguistic Research **25**(1) (1996) 59–92
17. Bard, E., Robertson, D., Sorace, A.: Magnitude Estimation of Linguistic Acceptability. Language **72**(1) (1996) 32–68