Contents

Chapter 1

Assertions and Questions

Assertions and questions can be seen as the two sides of the same coin, as they form the two core building blocks of any given conversation. Question typically request information, while assertions typically provide information. (1a) for instance, is a question that requests information about the country where Jo grew up (presupposing there is one such country). (1b) can be seen as a good (assertive) answer to this question, providing the piece of information that Jo grew up in France. Semanticists have observed that the pairs formed by questions and answers are restricted: some are obviously good, while some others are (sometimes surprisingly) odd. So, questions and answers have to be somewhat *congruent*. For instance, (1c) cannot be seen as a suitable answer to (1a), even if it seems to indicate something about Jo's nationality.

(1) a. In which country did Jo grow up?
b. –Jo grew up in France.
c. # –Jo speaks French natively.

This Chapter motivates and lays the foundation of the main contribution of this dissertation: a constrained machinery "retro-engineering" questions out of assertions, allowing to capture intricate patterns in the domain of pragmatic oddness, that were not previously seen as an issue of question-answer congruence. This Chapter is organized as follows. Section 1.1 provides a broad overview of the semantics of assertions, and discusses to what extent they can meaningfully contribute to a conversation. Section 1.2 turns to the semantics and pragmatics of questions and highlights how questions relate to alternative assertions, and their possible answers. Section 1.3 bridges Sections 1.1 and 1.2, by discussing how questions further constrain which assertions should matter in a given conversation. It also points out a few cases in which question-answer is (seemingly) unhelpful. Section 1.5 constitutes a more technical appendix sketching how the semantics of questions is standardly derived. This whole Chapter heavily builds on the section of von Fintel and Heim (2023) dedicated to Questions.

1.1 Assertions provide information in the form of propositions

1.1.1 Extension and intension of assertions

When studying the semantics of natural language expressions, one usually starts with assertions, because they appear intuitively simpler. We will use the simple assertion in (1b), as a running example. At the most basic level, assertions are truth-conditional, i.e. their meaning corresponds to the set of conditions under which they hold. For instance, *Jo grew up in France* will be true if and only if whoever *Jo* is, grew up in whatever geographical entity *France* is. The *extension* of an assertion is therefore of type $\mathtt{t}$, the type of truth-values.

Additionally, the truth-conditions of a sentence are parametrized by (at least) a world variable.[1]. So, *Jo grew up in France* will be true as evaluated against a world w_0 if and only if whoever *Jo* is in w_0, grew up in w_0 in whatever geographical entity *France* is in w_0. One can then abstract over this world-parameter, and define the *intension* of an assertion as a function from worlds to truth-values. Such functions are called *propositions*, and have type $\langle \mathtt{s}, \mathtt{t} \rangle$, where $\mathtt{s}$ is the type of world-variables. So, the intension, or propositional content of *Jo grew up in France*, will be a function mapping any world variable w, to true if and only if, whoever *Jo* is in w, grew up in w in whatever geographical entity *France* is in w. This is formalized (with some simplifications) in (2).

(2) $$[\![\text{ Jo grew up in France }]\!] = \lambda w.\ \text{Jo grew up in France in } w \\ : \langle \mathtt{s}, \mathtt{t} \rangle$$

Propositions can receive an alternative, equivalent interpretation in terms of sets, based on the idea that any function with domain D and range R is just a (potentially infinite) set of pairs of elements in $D \times R$. A proposition is then simply the set of worlds in which it holds. This interpretation of propositions will be heavily used throughout the dissertation, and is outlined in (3).

(3) $$[\![\text{ Jo grew up in France }]\!] = \lambda w.\ \text{Jo grew up in France in } w \\ \simeq \{ w \mid \text{ Jo grew up in France in } w \ \}$$

[1] Other parameters can also be relevant, like times, and assignments. But we choose to keep things simple here.

1.1.2 Assertions in conversation

Propositions either denote functions of type $\langle \mathtt{s}, \mathtt{t} \rangle$, or subsets of the set of elements of type $\mathtt{s}$. Should all elements of type $\mathtt{s}$ be considered when evaluating such functions, or computing such subsets? It is commonly assumed that the worlds under consideration at any point of a conversation, are the ones that are compatible with the premises of the said conversation (Stalnaker, 1974, 1978). For instance, if two people have a discussion about *France*, it is often reasonable to assume that they agree on what geographical area *France* encompasses, and more generally about the topology of Earth. Moreover, they agree that they agree on this; and agree that they agree that they agree on this; etc. Propositions subject to this recursive, mutual, tacit agreement pattern, form what is called a Common Ground (henceforth **CG**, (Stalnaker, 1978)). Each conversation has its own CG, as defined in (4). The set of worlds in which all the propositions of the CG hold, is called the Context Set (henceforth **CS**). The CS associated with a conversation is therefore a subset of the set of all possible worlds; and can also be seen (under the set interpretation of propositions) as the grand intersection of the propositions in the CG. This is defined in (5).

(4) *Common Ground* (***CG***). Let $\mathcal{C}$ be a conversation between participants $\{P_1, ..., P_k\}$. Let $K(x, p)$ is a proposition meaning that individual x knows p, and p is a proposition. The Common Ground of $\mathcal{C}$ is the set of propositions that are recursively taken for granted by all the participants in $\mathcal{C}$:
$p \in CG(\mathcal{C}) \iff \forall n \in \mathbb{N}^*.\ \forall\{k_1, ...k_n\} \in [1; k]^n.\ K(P_{k_1}, K(P_{k_2}, ...K(P_{k_n}, p)...)$

(5) *Context Set* (***CS***). Let $\mathcal{C}$ be a conversation between participants $\{P_1, ..., P_k\}$. Let CG $CG(\mathcal{C})$ be the Common Ground of this conversation. Under a set interpretation of propositions, the resulting Context Set $CS(\mathcal{C})$ is the set of worlds verifying all propositions of the CG, i.e.:
$CS(\mathcal{C}) = \bigcap\{p \mid p \in CG(\mathcal{C})\}$.

The concepts of CG and CS help delineate which worlds to focus on when evaluating an assertion in context, and determining to what extent this assertion is informative. If uttering an assertion is akin to *adding* it to the CG, then, it also amounts to *intersecting* this assertion with the CS.

(6) *Updating the Common Ground.* Let $\mathcal{C}$ be a conversation, and $CG(\mathcal{C})$ its Common Ground. If a sentence S denoting p is uttered, then p is added to $CG(\mathcal{C})$ to form a new Common Ground $CG'(\mathcal{C})$:
$CG'(\mathcal{C}) = CG(\mathcal{C}) \cup \{p\}$

(7) *Updating the Context Set.* Let $\mathcal{C}$ be a conversation and $CS(\mathcal{C})$ its Context Set. If a sentence S denoting p is uttered, then a new Context Set $CS'(\mathcal{C})$ is derived by intersecting $CS(\mathcal{C})$ with p:
$CS'(\mathcal{C}) = CS(\mathcal{C}) \cap p$

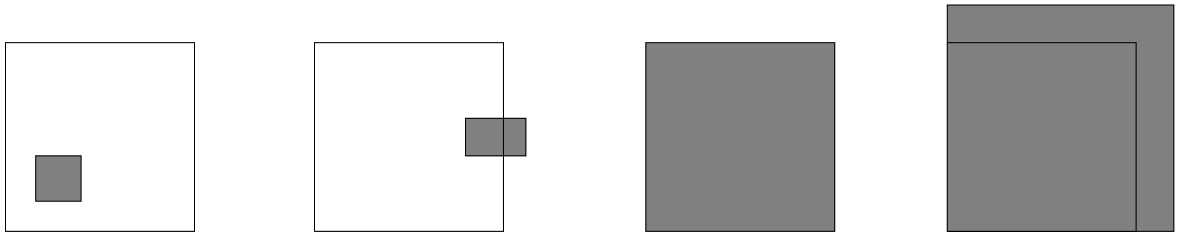
(8) *Link between the two updates.* (7) can be derived from (6) and the definition of the CG in (5):
$$\begin{aligned} CS'(\mathcal{C}) &= \bigcap\{q \mid q \in CG'(\mathcal{C})\} \\ &= \bigcap\{q \mid q \in CG(\mathcal{C}) \cup \{p\}\} \\ &= \bigcap\{q \mid q \in CG(\mathcal{C})\} \cap p \\ &= CS(\mathcal{C}) \cap p \end{aligned}$$

Note that updating the CG will always create a bigger set, because the CG is simply a collection of propositions. For instance, if *Jo grew up in Paris* is already in the CG, then, adding the proposition denoted by *Jo grew up in France* to the CG will mechanically expand it. Updating the CS however, does not always lead to a different, smaller CS. For instance, taking for granted that Paris is in France (i.e., all the *Paris*-worlds of the CS are *France*-worlds), and assuming that *Jo grew up in Paris* is already common ground, intersecting the CS with the proposition that *Jo lives in France* will not have any effect. This seems to capture the idea that a proposition like *Jo lives in France* is *uninformative* once it is already known by all participants that *Jo lives in Paris*.

More generally, if it is Common Ground that p, and a sentence S denoting p^- s.t. $p \vDash p^-$ is uttered, then S will feel uninformative. An informative assertion should lead to a non-vacuous update of the CS, i.e. it should properly *shrink* the CS. This is spelled out in (9).

(9) *Informativity (propositional view).* A sentence S denoting a proposition p is informative in a conversation $\mathcal{C}$, iff $CS(\mathcal{C}) \cap p \subset CS(\mathcal{C})$.

In that framework, an assertion provides information in the sense that it reduces the set of live possibilities, and allows to better guess which world is the "real" one. Figure 1.1 illustrates how an asserted proposition can be informative or uninformative, depending on its set-theoretic relationship to the CS.



(a) Informative update (p strictly contained in the CS)

(b) Informative update (p strictly overlaps with the CS)

(c) Uninformative update (p equal to the CS)

(d) Uninformative update (p containing the CS)

Figure 1.1: A few examples of informative and uninformative updates of the CS. The big squares represent the CS. The grey shapes refer to p, the proposition added to the CG (and intersected with the CS to update it).

1.1.3 Dynamic Semantics

So far, we have mainly considered "simplex" assertions that did not make use of operators, connectives or quantifiers. But what about sentences like those in (10)? How should they interact with the Context Set?

(10) a. Jo did not grow up in France.
b. Jo grew up in France or Belgium.
c. Jo grew up in France and Ed in Belgium.

The simplest way to deal with these sentences, would be to compute their intension (the proposition they denote) based on the semantics of negation, disjunction, and conjunction, and then, intersect the resulting proposition with the Context Set. We will call this approach the naive "bulk" CS update. There is evidence, coming from the behavior of presuppositions, that this might not be the way to go, and that complex assertions should be added to the Context Set "bit by bit" (Heim, 1982, 1983; Hei, 1983).

To see this, let us consider the pair in (11). The sentences in (11) are conjunctive and only vary in the order of their conjuncts. Additionally, one of their conjuncts contains the presupposition trigger *too*, associated with the predicate *grew up in France*. In the felicitous variant (11a), *too* occurs in the second conjunct; in the the infelicitous variant (11a), *too* occurs in the first conjunct. Intuitively, *X too VP* imposes that whatever predicate *VP* denotes be true of at least one individual different from the one *X* denotes. This presupposition can be seen as a precondition on the Context Set (as defined prior to the update step). In the case of (11a) and (11a), *Ed too grew up in France* then imposes that the Context Set at the time of the update entail that somebody other than Ed (e.g., Jo) grew up in France.

(11) a. Jo grew up in France, and Ed too grew up in France.

b. # Ed too grew up in France, and Jo grew up in France.

Let us attempt a naive "bulk" CS update with sentences (11a)/(11b). The first step is to compute (11a)/(11b)'s presuppositions and (propositional) assertions. The CS, as defined prior to the utterance of (11a)/(11b), then gets updated, provided that it verifies (11a)/(11b)'s presupposition. Let us start with (11a) and (11b)'s presuppositional component. We can assume that the presupposition that somebody other than Ed grew up in France projects from inside the conjunctive operator. Under this assumption, both (11a) and (11b) end up imposing that the CS prior to their utterance entail that somebody other than Ed grew up in France. This will in principle *not* be verified. So, the naive "bulk" Context Set update correctly predicts the infelicity of (11b), but, also, incorrectly predicts (11a) to be odd. Assuming the presupposition does not project does not address the issue. Under this assumption, both (11a) and (11b) end up being presuppositionless, and the naive "bulk" CS update correctly predicts (11a)'s felicity, but also incorrectly predicts (11b) to be just as felicitous. So, regardless of how presupposition should exactly behave in complex sentences, the asymmetry between (11a) and (11b) does not seem to be captured by the naive "bulk" CS update.

The linear asymmetry in (11) in fact suggests an alternative, "bit by bit" update strategy for complex sentences like conjunctions. If each conjunct were to update the CS one at a time, following the linear order of the sentence, then, the first conjunct of (11a) would create an updated CS that would incorporate the information that *Jo grew up in France*, and as such verify the presupposition of (11a)'s second conjunct (that somebody other than Ed grew up in France). This would allow (11a)'s second conjunct to be subsequently intersected to the CS, and would predict the whole conjunction in (11a) to be felicitous. By contrast, (11b)'s first conjunct would still be problematic in this framework, because its presupposition would not be satisfied by the original CS.

In this toy example, a presupposition was used as a diagnostic to better determine the nature of the CS update triggered by a conjunctive sentence. The conclusion is that the update should be dynamic: the two conjuncts should be intersected with the CS one by one, in the order in which they appear. This should apply to presuppositionless sentences as well; and is summarized in (12).

(12) *Conjunctive update of the CS.* Let $\mathcal{C}$ be a conversation and $CS(\mathcal{C})$ its Context Set. If a sentence S of the form $X \wedge Y$, with $[\![X]\!] = p$ and $[\![Y]\!] = q$ is uttered, then a new Context Set $CS''(\mathcal{C})$ is derived by, first intersecting $CS(\mathcal{C})$ with p to create $CS'(\mathcal{C})$, and second, intersecting $CS'(\mathcal{C})$ with q to create $CS''(\mathcal{C})$:
$CS''(\mathcal{C}) = (CS(\mathcal{C}) \cap p) \cap q = CS'(\mathcal{C}) \cap q$

The potential presuppositions of X and Y are tested on the CS at the time of their respective update, i.e. on $CS(\mathcal{C})$ and $CS'(\mathcal{C})$ respectively.

Dynamic Semantics is a framework that proposes to extend this view to other kinds of complex sentences, e.g. disjunctive and conditional sentences. In Dynamic Semantics, sentences give rise to different kinds of CS updates, depending on how they are constructed. More fundamentally, Dynamic Semantics proposes a shift of perspective when it comes to the meaning of assertions: assertions no longer denote propositions, instead they denote proposals to update the CS in specific ways. In that sense, assertions can be seen as functions from an input CS, to an output CS–sometimes called Context-Change Potentials (**CCP**). CCPs for disjunctive and conditional sentences are spelled out in (13) and (14) respectively.

(13) *Disjunctive update of the CS.* Let $\mathcal{C}$ be a conversation and $CS(\mathcal{C})$ its Context Set. If a sentence S of the form $X \vee Y$, with $[\![X]\!] = p$ and $[\![Y]\!] = q$ is uttered, then a new Context Set $CS'(\mathcal{C})$ is derived by intersecting $CS(\mathcal{C})$ with $p \cup q$:
$CS'(\mathcal{C}) = CS(\mathcal{C}) \cap (p \cup q)$
The potential presuppositions of X and Y are tested on, respectively, $CS(\mathcal{C})$ and $CS(\mathcal{C}) \cap \neg p$.[2]

(14) *Conditional update of the CS.* Let $\mathcal{C}$ be a conversation and $CS(\mathcal{C})$ its Context Set. If a sentence S of the form *if* X *then* Y, with $[\![X]\!] = p$ and $[\![Y]\!] = q$ is uttered, then a new Context Set $CS''(\mathcal{C})$ is derived by, first intersecting $CS(\mathcal{C})$ with p to create $CS'(\mathcal{C})$, and second, intersecting $CS'(\mathcal{C})$ with q to create $CS''(\mathcal{C})$:
$CS''(\mathcal{C}) = (CS(\mathcal{C}) \cap p) \cap q = CS'(\mathcal{C}) \cap q$
The potential presuppositions of X and Y are tested on the CS at the time of their respective update, i.e. on $CS(\mathcal{C})$ and $CS'(\mathcal{C})$ respectively.

This incremental view of assertions leads to a revised, incremental definition of informativity, given in (15).

(15) *Informativity (CCP view).* A sentence S is informative in a conversation $\mathcal{C}$, iff all the updates of $CS(\mathcal{C})$ it gives rise to are non-vacuous.

In sum, assertions can be seen as proposals to update (shrink) the CS. The specific update they give rise to is compositionally derived, and incrementally performed, following the structure of the sentence. We will use a similar approach in Chapter **??** when defining

[2]There is a debate on whether or not disjunctions should behave symmetrically w.r.t. the presupposition(s) carried by their disjuncts. An alternative, symmetric way to evaluate X and Y's potential presuppositions, would be to test them against $CS(\mathcal{C}) \cap \neg q$ and $CS(\mathcal{C}) \cap \neg p$ respectively.

questions *evoked* by assertions. But this first requires to define what questions mean. This is what we do in the next section, in which we show that questions influence, not the size, but rather, the topology of the CS.

1.2 Questions indicate which kind of information is worth providing

1.2.1 Questions as answerhood conditions

Participants in a conversation utter assertions to shrink the CS, and hopefully, jointly figure out which world they are in. But this allows for very unnatural interactions like (16), taking the forms of sequences of intuitively unrelated sentences–as long as each of them denotes propositions shrinking the CS!

(16) –Jo grew up in France.
–I like cheese.
–Al is arriving tomorrow.

This is where questions enter the game. Intuitively, a question indicates an interest in *which* proposition(s) hold, among a restricted set. The proposition at stake are typically possible answers to the question. Questions therefore denote sets of sets of worlds (equivalent to a type $\langle\langle \mathtt{s}, \mathtt{t}\rangle, \mathtt{t}\rangle$), and constrain which kind of (informative) propositions can be uttered as a follow-up. For instance, a polar question such as *Is it raining?* will typically request information of the form *It is raining*, or *It is not raining*, see (17).

(17) –Is it raining?
–Yes, it is raining. / No, it is not raining.

The question *Is it raining?* can thus be represented as a set made of two propositions, namely, the proposition that *it is raining*, and the proposition that *it is not raining*.

(18)
$$\begin{aligned}[\![\text{ Is it raining? }]\!] &= \{ [\![\text{ It is raining }]\!], [\![\text{ It is not raining }]\!] \} \\ &= \{\lambda w.\ \text{it is raining in } w,\ \lambda w.\ \text{it is not raining in } w\} \\ &: \langle\langle \mathtt{s}, \mathtt{t}\rangle, \mathtt{t}\rangle\end{aligned}$$

In the case of the question *is it raining?*, the set of possible answers is fairly simple: it only contains two elements.These two elements cover the space of all possibilities,[3] and

[3]This is the case assuming there is no vagueness-induced "grey area", i.e. any salient situation is either a *raining*-situation, or a *not raining*-situation

are *exclusive*: if it's the case that it's raining (at a salient place, at a salient time) in w, then, it's not the case that it is not raining (at the same place, at the same time), in w. We will see in the next section that this configuration amounts to a partition of the CS. A definition of exclusivity under the set interpretation of propositions is given in (19).

(19) *Exclusive propositions.* $p : \langle \mathtt{s}, \mathtt{t} \rangle$ and $q : \langle \mathtt{s}, \mathtt{t} \rangle$ are exclusive if $p \cap q = \emptyset$.

But questions may not always intuitively request information about exclusive propositions. For instance, a *wh*-question like *Which students passed the class?* expects answers that convey a subset of students who passed the class, see (20). But there are many possible, overlapping subsets of students, so, the corresponding propositions will be overlapping as well. For instance, the proposition that *Jo passed the class*, denotes the set of worlds in which Jo passed the class, and this set happens to contain the set of worlds where both Jo and Al passed the class. It also overlaps with the set of worlds in which Al passed the class.

(20) Which students passed the class?
–Jo did.
–Al did.
–Jo and Al did.

We will call propositions like *Jo passed the class*, and *Jo and Al passed the class*, alternatives associated to the question *Which students passed the class?* Alternatives may be overlapping; and, as we will see, can be obtained from the original question by substituting its *wh*-component (e.g., *which students*), with relevant, same-type material (e.g., students or groups of students).[4]

(21) Question : $[\![$ Which students passed the class? $]\!]$
Alternatives: $\{[\![$ Jo passed $]\!]$, $[\![$ Al passed $]\!]$, $[\![$ Jo and Al passed $]\!]$ ... $\}$

Why would this overlap between alternative answers be an issue in modeling the meaning of questions? The fact that entailing or merely overlapping propositions should be considered equally good answers does not capture the idea that more specific propositions constitute more exhaustive answers than less specific ones. For instance, answering that *Jo passed*, in theory leaves the fate of the other students undecided–for instance, it does

[4] It is worth mentioning that the set $\{\lambda w.$ it is raining in $w,\ \lambda w.$ it is not raining in $w\}$ does not strictly speaking correspond to the set of alternatives raised by *Is it raining?* Section 1.5 further specifies how alternatives get compositionally derived, and predicts that *Is it raining?* should only give rise to one alternative: $\lambda w.$ it is raining in w. The set $\{\lambda w.$ it is raining in $w,\ \lambda w.$ it is not raining in $w\}$ is derived from this singleton alternative *via* the "pragmatic" process presented in (24), in the next Section.

not settle if *Al passed*, or not. Answering that *Jo and Al passed* by contrast, settles Al's fate, in addition to Jo's. Ideally, an answer to *Which students passed?* should explicitly address whether *each* student of the class passed, or not. That would be an exhaustive answer.

1.2.2 Questions as partitions of the Context Set

We have just discussed that, at the semantic level, questions characterize the conditions under which they are answered, i.e. denote a set of potentially overlapping propositions. But, just like we did with assertions, the effect of this semantics on the Context Set has to be defined. There is in fact a deterministic way to change a set of overlapping propositions P (i.e. a set of subsets of the CS), into a set of exclusive subsets of the CS (called *cells*, for reasons made clear in (24)). To do so, one can group in the same cell the worlds of the Context Set that all "agree" on all propositions in P. This "agreement" property amounts to the same-cell relation in (22). This relation is reflexive, symmetric and transitive, i.e. is an equivalence relation (see proof in (23)). From this, we can conclude that the set of subsets of the CS induced by P, obtained by grouping worlds of the CS according to the same-cell relation, forms a partition of the Context Set (see proof in (24)).[5] So, on top of being exclusive, cells are non-empty and together cover the CS. We assume that the process changing the set of alternative propositions raised by a question, to a partition of the CS, belongs to pragmatics. So, questions *denote* sets of alternative propositions, and this set *pragmatically induces* a partition structure on the CS.

(22) *Same-cell relation* $\equiv_P$. Let P be a set of propositions, i.e. a set of subsets of the Context Set ($P \in \mathcal{P}(\mathcal{P}(CS))$, with $\mathcal{P}$ the powerset operation). Let w and w' be two worlds of the Context Set. $w \equiv_P w'$ iff, $\forall p \in P.\ p(w) = p(w')$.

(23) $\equiv_P$ is an equivalence relation, no matter what P is. Let $\forall P \in \mathcal{P}(\mathcal{P}(CS))$.

- $\equiv_P$ is reflexive: $\forall w \in CS.\ \forall p \in P.\ p(w) = p(w)$.
- $\equiv_P$ is symmetric. Let $\forall (w, w') \in CS^2$.
  $\forall p \in P.\ p(w) = p(w')$ iff $\forall p \in P.\ p(w') = p(w)$.
- $\equiv_P$ is transitive. Let $\forall (w, w', w'') \in CS^3$.
  We assume $\forall p \in P.\ p(w) = p(w')$ and $\forall p \in P.\ p(w') = p(w'')$.
  Let $\forall p \in P$. We have $p(w) = p(w')$ and $p(w') = p(w'')$, so $p(w') = p(w'')$.
  So, $\forall p \in P.\ p(w) = p(w'')$

[5]Cells as we defined them are also called equivalence classes. It's a general property that equivalence classes induced by an equivalence relation on a certain set on which this relation is defined, will create a partition of the set.

(24) *Partition of the CS induced by P.*[6] Let P be a set of propositions. The partition induced by P in the Context Set is the set of subsets of the CS (cells): $\mathfrak{P}_{P,CS} = \{\{w' \mid w' \in CS \wedge w' \equiv_P w\} \mid w \in CS\}$. This set partitions the CS.

- No cell c of $\mathfrak{P}_{P,CS}$ is empty. Let $c \in \mathfrak{P}_{P,CS}$. There is a $w \in CS$ s.t. $c = \{w' \mid w' \in CS \wedge w' \equiv_P w\}$. Then at least $w \in c$, because $w \equiv_P w$.
- Cells cover the CS. Let $w \in CS$. $\mathfrak{P}_{P,CS}$ contains a cell $c = \{w' \mid w' \in CS \wedge w' \equiv_P w\}$. Then $w \in c$ because $w \equiv_P w$.
- Cells are disjoint. Let $(c, c') \in \mathfrak{P}_{P,CS}$, s.t. $c \cap c' \neq \emptyset$. We show $c = c'$. c and c' have resp. the form $c = \{w'' \mid w'' \in CS \wedge w'' \equiv_P w\}$ and $c = \{w'' \mid w'' \in CS \wedge w'' \equiv_P w'\}$, for $(w, w') \in CS^2$. Let $w''' \in c \cap c'$. Then $w''' \equiv_P w$ and $w''' \equiv_P w'$, and so by symmetry and transitivity, $w \equiv_P w'$, and $c = c'$.

It is easy to show that, in the polar example (17), the subsets of the CS defined by *It is raining* and *It is not raining*, which we said were intuitive answers to the question, form a partition of the CS. Section 1.5 will in fact show that polar questions of the form p? *denote* the singleton set formed by p, and *induce* a 2-cell partition of the form $\{p, \neg p\}$.

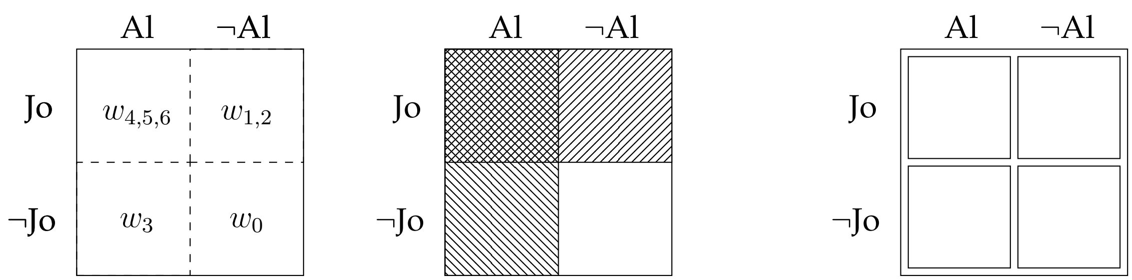
Let us now see how the above definitions apply to a *wh*-question like *Which students passed?* in (20). Let's assume there are only two salient students, Jo and Al. We assume that the alternatives the question raises (labeled P), are the proposition that *Jo passed*, and the proposition that *Al passed*. We assume that the CS contains six possible worlds, which vary according to whether Jo, Al, both, or none passed the class. The worlds may vary in other respects, that are not relevant to us here. The alternatives and cells associated with this question are given in (25). The alternative set P then corresponds to two subsets of the CS, which do not cover it. In particular, the world in which nobody passed (w_0) is included in none of the two alternatives. Moreover, the two subsets are overlapping: both *Jo passed* and *Al passed* contain w_4, w_5, and w_6. Now turning to the cells induced by P on the CS, we notice that there are four of them, which correspond to worlds where nobody, only Jo, only Al, or both Jo and Al passed the class. Such cells cover the CS, are disjoint, and non-empty, so correctly form a partition of the CS. They also fully specify, for *both* Jo and Al, if they passed the class; and as such constitute exhaustive answers to the original question.

(25) Question : *Which students passed the class?*
Context Set: $\{w_0, w_1, w_2, w_3, w_4, w_5, w_6\}$, s.t.:

[6] *Fox (2018) proposes an alternative way to derive a partition of the CS from a set of alternative propositions, leveraging the covert operator exh*

- Nobody passed in w_0;
- Only Jo passed in w_1 and w_2;
- Only Al passed in w_3;
- Both Jo and Al passed in w_4, w_5, and w_6.

Alternatives (P): $\{[\![\text{Jo passed}]\!], [\![\text{Al passed}]\!]\} = \{\{w_1, w_2, w_4, w_5, w_6\}, \{w_3, w_4, w_5, w_6\}\}$
Cells induced by $\equiv_P$: $\{\{w_0\}, \{w_1, w_2\}, \{w_3\}, \{w_4, w_5, w_6\}\}$



(a) Distribution of w_0...w_6 in the CS.

(b) Alternatives: defines *Al passed* and defines *Jo passed*

(c) Partition induced by and . Cells correspond to the subsets of Figure 1.2b featuring the same overall pattern.

Figure 1.2: Partitioning of the CS defined in (25) according to the alternatives *Jo passed* and *Al passed*. The CS is organized as follows: counter-clockwise, quadrant I is made of *Jo but not Al*-worlds; quadrant II, *Jo and Al*, quadrant III, *Al but not Jo*, and quadrant IV, *neither Jo nor Al*.

To summarize, at the pragmatic level questions are partitions of the Context Set, as formalized in (26).[7] The cells of such partitions constitute maximal answers to the questions. Unions of two or more cells constitute non-maximal answers, as defined in (27).

(26) *Standard semantics for questions* (*Jäger, 1996; Hulstijn, 1997; Groenendijk and Stokhof, 1984; Groenendijk, 1999*). Given a conversation $\mathcal{C}$ and a Context Set $CS(\mathcal{C})$, a question on $CS(\mathcal{C})$ is a partition of $CS(\mathcal{C})$, i.e. a set of subsets of $CS(\mathcal{C})$ ("cells") $\{c_1, ..., c_k\}$ s.t.:

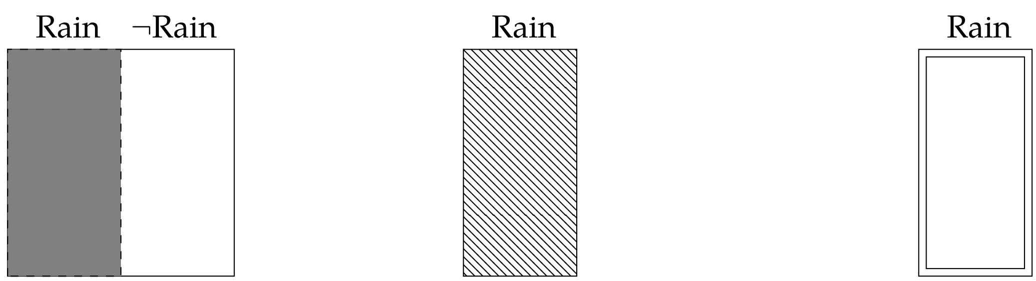
- "No empty cell": $\forall i \in [1;k].\ c_i \neq \emptyset$
- "Full cover": $\bigcup_{i\in[1;k]} c_i = CS(\mathcal{C})$
- "Pairwise disjointness": $\forall (i,j) \in [1;k]^2.\ i \neq j \Rightarrow c_i \cap c_j = \emptyset$

[7]It is important to note that questions may be taken to have a partition *semantics*. But we do not cover this here.

(27) *Maximal and non-maximal answers to a question.* Given a conversation $\mathcal{C}$, a Context Set $CS(\mathcal{C})$, and a question Q forming a partition $\{c_1, ..., c_k\}$ of $CS(\mathcal{C})$:

- Any $c \in \{c_1, ..., c_k\}$ constitutes a maximal answer to Q;
- Any c' s.t. $\exists C \subseteq \{c_1, ..., c_k\}. \, |C| > 1 \land c' = \bigcup C$ is a non-maximal answer to Q.

Just like we did with assertions, let us clarify further what it means to be a good question. We have established that the idea of a partition is a good candidate to model the effect of questions on a given CS. But what if the CS is already such that the partition induced by the question's alternatives is just made of one big cell? Such a configuration suggests that the question is already *settled*, meaning, the CS already makes one maximal answer trivial. For instance, if it is already common ground between the conversation's participants that *it is raining* (at the salient place and time) in (17), then, the question *Is it raining?* appears completely trivial. This is illustrated in Figure 1.3 and generalized in (28).



(a) Interecting the proposition that *It's raining* ( ) with a CS that is agnostic about the weather.

(b) Alternatives on the restricted CS: defines *It's raining*.

(c) Partition induced by on the CS. Cells correspond to the subsets of Figure 1.3b featuring the same overall pattern (1 such subset).

Figure 1.3: Updating the CS with the proposition that *It's raining*, and then computing the partition induced by *Is it raining* on the resulting "shrunk" CS. The outcome is a single-cell partition, i.e., the question has a trivial pragmatics.

(28) *Trivial question.* Let $\mathcal{C}$ be a conversation, $CS(\mathcal{C})$ its associated Context Set, and Q a question. Q is trivial given $CS(\mathcal{C})$ iff the partition induced by Q on $CS(\mathcal{C})$ is made of a singleton cell, i.e. has cardinal 1.

We now have a basic notion of what it mean to be a good assertion, given a CS, and a good question, given a CS. A good assertion has to be informative, i.e. properly shrink the CS (as per (9)/(15)). A good question has to induce a non-trivial, multiple-cell partition on the CS (as per (28)). But being a good question or a good assertion, does not *only* depend on the state of the CS! In particular, good assertions also have to be good answers to good questions. This principle, dubbed *Question-Answer Congruence*, is given in (29).

(29) *Question-Answer Congruence* (*Katzir and Singh, 2015*). A felicitous assertion has to be a good answer to a good question.

The next Section presents what can be seen as a partial implementation of this principle, in the form of a general principle dubbed RELEVANCE. It also points out the limitations of this principle.

1.3 Assertions as good answers to questions

1.3.1 Relevance mediate questions and assertions

Now that we precisified what assertions and questions are, it becomes possible to (at least partially) define what a good assertion should be, given a question. The principles we introduce in this Section are based on the general concept of RELEVANCE. They will eventually rule out informative but "unnatural" sequences of assertions like (16), but also, more generally, a wide range of odd question-answer pairs.

Following much previous literature (van Kuppevelt, 1995a,b; Roberts, 1996, 2012; Ginzburg, 1996; Büring, 2003), we call the question against which assertions are evaluated, *Question under Discussion* (henceforth **QuD**). QuDs are typically seen as partitions of the CS. In (27), we defined cells an unions of cells as respectively maximal and non-maximal answers to a question. Very broadly, RELEVANCE constrains what a proposition should do to the cells of the QuD. Let us now unpack this with an example.

If for instance the QuD is about which country Jo grew up in (as in (30)), the CS will be partitioned according to propositions of the form *Jo grew up in c*, with *c* a country. Utterances such as (30a) or (30b), both seem relevant to that kind of QuD, and both constitute answers to the QuD–maximal, or not. By contrast, utterances such as (30c), (30d) or (30e), do not appear relevant, and do *not* constitute answers to the QuD: there are native and non-native French speakers in virtually all countries; same holds for wine-lovers and wine-haters; as for (30e) it seems completely independent from the subject matter.[8] These various configurations are sketched in Figure 1.4.

(30) QuD: In which country did Jo grow up?

a. Jo grew up in France.

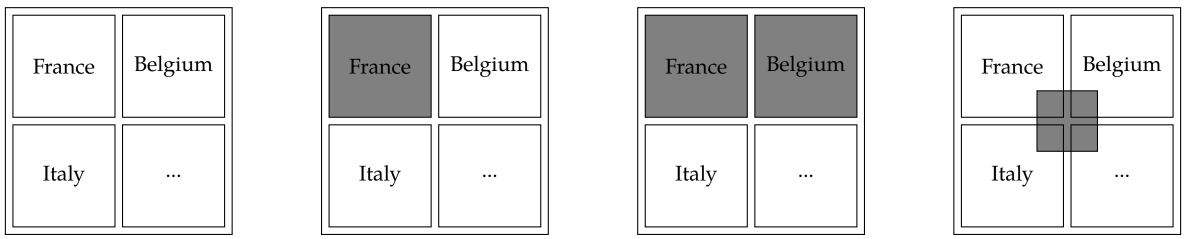
[8] It is interesting to note that (30c) and (30d) can be more easily coerced into relevance than (30e). For instance with (30c), one might consider that France is the country which, in proportion, comprises the most native French speakers, and so (30c) may be understood as *It is likely that Jo grew up in France*–which constitutes a modalized answer to the QuD. This kind of reasoning is harder (if not impossible) to perform when facing an utterance like (30e).

b. Jo grew up in France or Belgium.

c. ?? Jo speaks French natively.

d. ?? Jo enjoys wine.

e. # The cat went outside.



(a) QuD for *In which city did Jo grow up?* (b) Utterance: *Jo grew up in France.* (c) Utterance: *Jo grew up in France or Belgium.* (d) Utterance: (30c), (30d) or (30e).

Figure 1.4: QuD-utterance configurations for a QuD like *In which country did Jo grow up*, and possible follow-up utterance.

From this, we can conclude that a proposition is "relevant" to a question, if it constitutes a maximal or a non-maximal answer to the question. This is similar in spirit to the notion of *Aboutness* developed Lewis (1988), according to which a proposition p is about a subject matter (in modern terms, a QuD), if and only if the truth value of that proposition supervenes on that subject matter (i.e. p should not introduce truth-conditional distinctions between cellmates, i.e. p does not "cut across" cells). This is rephrased in (31). A typical Lewis-relevant configuration is exemplified in Figure 1.6a. It is also worth mentioning that (31) deems propositions contradicting the CS irrelevant (see Figure 1.6b), but in principle does not rule out propositions covering the whole CS (see Figure 1.6c), i.e. uninformative propositions as per (9).

(31) Lewis's Relevance (rephrased in the QuD framework). Let $\mathcal{C}$ be a conversation, Q a QuD defined as a partition of $CS(\mathcal{C})$. Let p be a proposition. p is Lewis-relevant to Q, iff $\exists C \subseteq Q.\ p \cap CS(\mathcal{C}) = C$

But, coming back to the QuD *In which country did Jo grow up?*, what about an utterance of the form *Jo grew up in Paris?* Although overinformative (the QuD was only asking about countries, not cities!), this utterance appears relevant, because it allows to infer that Jo grew up in France, and not, say, Belgium. This kind of configuration is sketched in Figure 1.5.

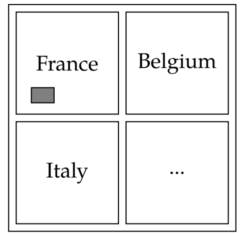
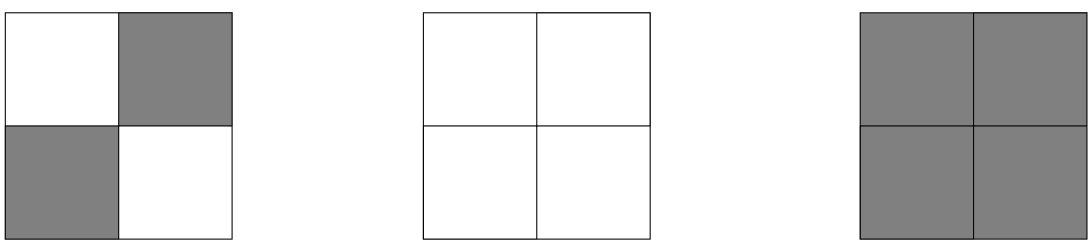


Figure 1.5: QuD-utterance configuration for a QuD like *In which country did Jo grow up?* and an utterance like *Jo grew up in Paris*.

The view of relevance, developed by Roberts (2012), captures this intuition, by stating that a relevant proposition has to rule out at least one maximal answer conveyed by the QuD. In other words, a relevant proposition has to be incompatible with at least one cell of the QuD. This is summarized in (32). This definition makes uninformative propositions irrelevant (see Figure 1.6c), but allows certain propositions that do not coincide with the grand union of a subset of the QuD's cells, to be relevant (see Figures 1.6d and 1.6e). In other words, relevant propositions in the sense of Roberts may introduce truth-conditional distinctions between cellmates–as long as they rule out a cell. A particular case is that of propositions like *Jo grew up in Paris*, when the QuD is about countries, which strictly entail a specific cell of the QuD, i.e. are strictly contained in one single cell (see Figure 1.6d).

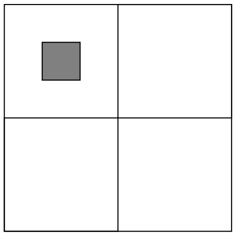
(32) Roberts's Relevance (Roberts, 2012). Let $\mathcal{C}$ be a conversation, Q a (non-trivial) QuD defined as a partition of $CS(\mathcal{C})$. Let p be a proposition. p is Roberts-relevant to Q, if $\exists c \in Q.\, p \cap c = \emptyset$.



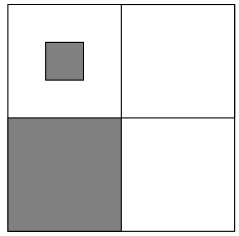
(a) Informative, Lewis-relevant, Roberts-relevant

(b) Informative, not Lewis-relevant, Roberts-relevant

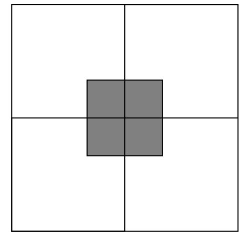
(c) Uninformative, Lewis-relevant, not Roberts-relevant.



(d) Informative, not Lewis-relevant, Roberts-relevant



(e) Informative, not Lewis-relevant, Roberts-relevant



(f) Informative, not Lewis-relevant, not Roberts-relevant

In sum, the concept of RELEVANCE (whether it follows Lewis's or Roberts's implementation) allows to rule-out a wide range of QuD-utterance pairs, by stating that propositions should properly relate to an existing question. We will not discuss which approach between Lewis's and Roberts's is best here, and will propose an incremental variant of this core concept in Chapter **??**, to deal with certain complex, out-of-the blue sentences. The next two section outline a few limitations of relevance.

1.3.2 A few conceptual shortcomings of RELEVANCE

Regardless on which view of RELEVANCE is adopted, relevant propositions can be added to the Common Ground, and as such, trigger an update of the CS. This, in turn, updates the QuD, which must remain a partition of the CS. It is easy to show, given how partition are "induced" on a set (see definition (24)), that the updated QuD on the smaller CS corresponds to the previous QuD, whose cells are pointwise intersected with the newly added proposition, and such that empty cells are filtered. This is formalized in (33).

(33) *Updating the partitioned Context Set.* Let $\mathcal{C}$ be a conversation, $CS(\mathcal{C})$ its Context Set, and let Q be a partition of $CS(\mathcal{C})$. If a sentence S denoting p is uttered and relevant given Q (as per (31) or (32)), then a new Context Set $CS'(\mathcal{C})$ is derived by intersecting $CS(\mathcal{C})$ with p, and this new context set is partitioned by Q', s.t.:
$Q' = \{c' \mid \exists c \in Q.\ c' = c \cap p \wedge c' \neq \emptyset\}$

There are two shortcomings to the current framework. First, adding a proposition to the CG "mechanically" leads to an update of the CS and of the QuD, but does not directly affect the *structure* of this QuD: even if some cells should shrink, the *limits* of each cell remain the same. This goes against the intuition that sometimes, sentences give rise rise to brand new QuDs, as exemplified by the exchange in (34).

(34) –Is it raining?
–Yes, I think so. I just so Ed come in with this very pretty umbrella.
(Likely follow-up: Where did Ed find this umbrella?)

Second, and relatedly, one can wonder what is supposed to happen in the case of out-of-the-blue sentences, i.e. sentences for which there is no explicit QuD. In such cases, it is generally assumed that a reasonable QuD is somehow inferred. But, given the fact that a QuD is merely a partition of the current CS, there exists many options. This dissertation will focus on how exactly QuDs are inferred, what additional constraints hold between an assertion and a QuD, and what the consequences are for pragmatic theory.

1.3.3 Relevance and the packaging of information

We start by showing that the felicity of disjunctions and conditionals is sensitive to *overt* QuDs – but in different ways. We take this as evidence that out-of-the-blue disjunctions and conditionals accommodate different kinds of implicit QuDs.

If a context contrasting *Paris* and *France but not Paris* is set as in (35), (**??**) and (**??**) improve (see Haslinger (2023) for similar effects on disjunctions and conjunctions). This is strange: even if the context and question made *Paris* (but no other French city) a relevant alternative to *France*, *exh* would remain IW in the consequent of (**??**): *if Jo did not grow up in Paris, she grew up in France but not Paris*, is equivalent to *if Jo did not grow up in Paris, she grew up in France*. In other words, *exh* (as constrained by IW) cannot leverage the contextually provided alternatives to make (**??**) escape SR in (35). The same applies to (**??**).

(35) *Context: French accents vary across countries and between Paris the rest of France.*
Al: I'm wondering where Jo learned French.
Lu: I'm not completely sure but... (**??**) ✓(**??**) ✓

This suggests that a purely LF-based view of redundancy such as SR, may be insufficient to capture the interaction between HCs and how their context of utterance packages information. Rather, it seems that the context of (35) makes a specific partition of the CS salient, and that this partition can be used to make otherwise infelicitous assertions accommodate a different question than the one they would evoke out-of-the-blue.

Additionally, conditionals and disjunctions seem to accommodate distinct QuDs. To show this, we use the construction *depending on Q, p* (Karttunen (1977); Kaufmann (2016)), where Q is a question and p a proposition. This construction has been argued to force the partition conveyed by Q to match specific live issues raised by p. We understand such "live issues" as the maximal true answers of the QuD evoked by p. The contrast between (36a) and (36b) then suggests that the *France* and *Belgium* answers can be matched against Q in the disjunctive, but not in the conditional case. This in turn means that a disjunction introduces a QuD making both disjuncts maximal true answers, while a conditional does not do the same with its consequent and the negation of its antecedent.

(36) Depending on [how her accent sounds like]$_Q$...

a. Jo grew up in France **or** in Belgium. $\mathbf{p}\vee\mathbf{q}$

b. ?? **if** Jo didn't grow up in France she grew up in Belgium. $\neg\mathbf{p}\rightarrow\mathbf{q}$

c. ? **if** Jo didn't grow up in France, she grew up in Belgium **or** in Québec. $\neg\mathbf{p}\rightarrow(\mathbf{q}\vee\mathbf{r})$

d. ?? **if** Jo didn't grow up in France **or** Belgium, she grew up in Québec. $\neg(\mathbf{p}\vee\mathbf{q})\rightarrow\mathbf{r}$

The existence of an improvement between (36b) and (36c), and the absence of a similar improvement in between (36b) and (36d), also implies that the answers targeted by *depending on Q*, when *p* is conditional, are the ones made available by the consequent of *p* (which is appropriately disjunctive in (36c), but not (36d)).

More generally, this predicts "connectivity effects" in disjunctions-of-conditionals, in that the antecedents and consequents respectively have to address similar QuDs; and no such effect in conditionals-of-disjunctions, in that disjuncts coming from the antecedent and consequent may be inquisitively unrelated.

1.4 Roadmap of the dissertation

Specifically, we will claim that instead of being a "good" answer to *some* QuD, an out-of-the-blue sentence must be a good answer to a *good* QuD, following insight by Katzir and Singh (2015). We will show that operationalizing this principle allows to account for a wider range of oddness phenomena, that previous approaches struggled to capture under the same umbrella.

"Good" QuDs are determined from the shape of the assertive sentence itself. This is pushing the idea that assertions evoke alternatives one step further, in the sense that sentences will be taken to evoke questions (themselves derived from alternatives). These evoked questions will have a structure that consists in a generalization of the partition structure, namely, they will take the form of parse tree of the CS. We will additionally claim the process deriving good questions from good answers, is subject to constraints that go beyond relevance, and cover concepts such as redundancy. These constraints will make way for a "lifted" view of pragmatic oddness, under which an assertion is not odd *per se*, but rather, is odd due to its interaction with the QuDs it evokes. Before presenting the core components of our model, we will briefly present two recent accounts of oddness based on similar ideas.

1.5 Appendix: computing questions from propositions

So far, we have described what could be a reasonable model for questions, in the form of partitions of the CS. But this was done without explaining how exactly such partitions are derived from the Logical Form of questions. This sketches how this is done, while further clarifying the distinction between propositions, alternatives, and questions. We will show that questions are standardly derived from closely related propositions, by abstracting over specific variables.

We will use the question *In which country did Jo grow up?* as an example. The LF associated with this question is given in Figure 1.7.

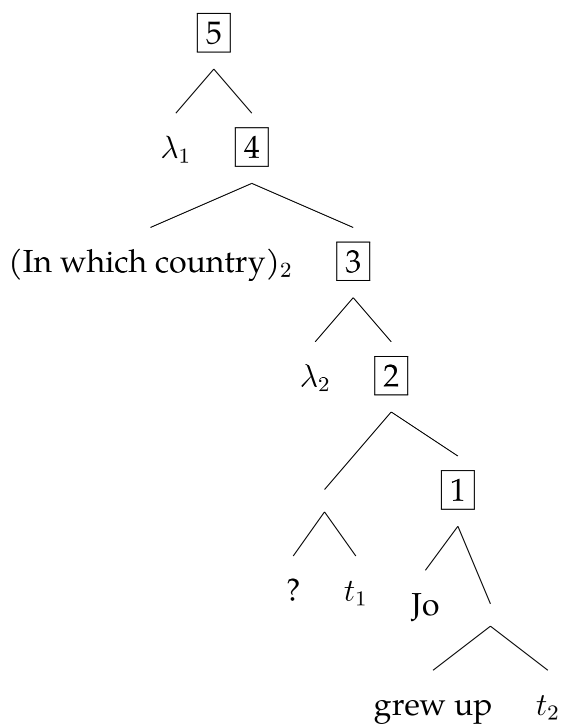


Figure 1.7: LF of the question *In which country did Jo grow up?*

This question involves a *wh*-phrase (*in which country*), which syntactically originates in an adjunct of *grow up*. It is assumed that the *wh*-phrase leaves a trace t_2 in this position. The semantics assigned to the *wh*-phrase is existential, and akin to *some country*. Specifically, *in which country* takes a predicate of type $\langle \mathtt{e}, \mathtt{t} \rangle$ as argument, and returns the quantified statement that *some country* verifies the predicate.

(37) $[\![\text{In which country}]\!]^w = \lambda P.\ \exists l.\ l$ is a country in $w \wedge P(l) = 1$

The *wh*-phrase outscopes another "proto-question" operator (Karttunen, 1977). This operator takes two propositions (here, the trace t_1 and the proposition that *Jo grew up in* t_2), and simply equates them.

(38) $[\![?]\!]^w = \lambda p.\ \lambda q.\ p = q$

Applying this operator successively to t_1 and the intension of $\boxed{1}$, yields the following.

(39) $\boxed{1} = [\![\text{Jo grew up } t_2]\!]^w = 1$ iff Jo grew up in t_2 in w

(40) $\boxed{2} = [\![?\ t_1 \text{ Jo grew up } t_2]\!]^w = 1$ iff $t_1 = \lambda w'$. Jo grew up in t_2 in w'

Abstraction then applies to $\boxed{2}$, binds t_2 and yields a predicate that can then serve as an argument of the *wh*-phrase. The *wh*-phrase then turns this predicate into an existentially quantified expression targeting the element being questioned (here, a country).

(41) $\boxed{3} = [\![\ \lambda_2\ ?\ t_1 \text{ Jo grew up } t_2]\!]^w = \lambda l.\ t_1 = \lambda w'$. Jo grew up in l in w'

(42) $\boxed{4} = [\![$ In which country ... Jo grew up $t_2]\!]^w$
$= \exists l.\ l$ is a country in $w \wedge t_1 = \lambda w'$. Jo grew up in l in w'

Lastly, a t_1 gets bound to produce a set of propositions, namely, the set of propositions that coincide with the proposition that *Jo grew up in l*, for some country l.

(43) $\boxed{5} = [\![\ \lambda_1$ In which country ... Jo grew up $t_2]\!]^w$
$= \lambda p.\ \exists l.\ l$ is a country in $w \wedge p = \lambda w'$. Jo grew up in l in w'
$\simeq \{p \mid \exists l.\ l$ is a country in $w \wedge p = \lambda w'$. Jo grew up in l in $w'\}$

This example showed that the semantics of a question is derived from that of its "assertive counterpart", where the *wh*-phrase is replaced by a quantified variable. Combined with the proto-question operator and λ-abstraction, this allows to generate a set of propositions, which only vary in terms of the variable being questioned. This set of propositions (alternatives) can then be used to induce a partition of the CS, as per (**??**).

Chapter 2

Accommodating QuDs: Qtrees

Abstract. This Chapter introduces a model of questions that is more sophisticated than standardly assumed (cf. Chapter 1). Questions are defined as recursive partitions, or parse trees of the Context Set. This model is shown to capture fine-grained information about how questions relate to each other in terms of specificity, and what it means to answer a question. The Chapter then describes how such questions can be "retro-engineered" from assertions, in a compositional way–relatively similarly to the *Dynamic Semantics* framework. Lastly, we suggest ways in which this more fine-grained model of questions can eventually make more fine-grained predictions in the domain of pragmatic oddness.

2.1 Making sense

2.1.1 Oddness despite relevance and informativeness

In Chapter 1, we have seen that assertive sentences should be informative, i.e. lead to an incremental shrinkage of the Context Set (**CS**) (Stalnaker, 1978; Heim, 1982). We have also seen that they should be relevant, i.e. shrink the CS in a way consistent with what the Question under Discussion (**QuD**) is (Lewis, 1988; Roberts, 2012). But sometimes, it is unclear what the QuD should be, and how relevance could help. The follow-up sentences in (44) exemplify this.

(44) –Have you seen Jo today?
–No I haven't...

a. # I heard she is at a conference in Paris or France.

b. Either she is sick, or if she's not sick, she is at a conference.

c. # Either she is sick, or if she's not at a conference, she is sick.

In these answers, the explicit question *Have you seen Jo today?* is first settled, so that there is not explicit QuD left to be addressed. Yet, it appears that the follow-up assertions address a different question, along the lines of *Where is Jo?*. But how exactly this question gets derived from the sentences at stake, remains unclear. Additionally, not all follow-up sentences appear felicitous. (44a) instantiates a Hurford Disjunction (Hurford, 1974), i.e., at the descriptive level, a disjunction whose disjuncts are in a relation of contextual entailment (*Paris* $\vdash$ *France*). This disjunction is informative: it says that *Jo is at a conference in France*. It is also relevant to the question that gets intuitively inferred from the exchange: it identifies an event and a country where Jo is at the moment. Yet, (44a) is sharply odd. There are in fact many successful accounts of (44a)'s oddness, building on constraints independent of the QuD and Relevance.

But we will see in Chapter **??** that such accounts fall short in explaining the contrast between (44b) and (44c). These two follow-up sentences are both informative: assuming implication is material, they both mean that Jo is sick or at a conference. They also appear intuitively relevant to a QuD along the lines of *Where is Jo?*. Yet, (44b) makes perfect sense, while (44c) does not seem to make any sense. The goal is then to devise a pragmatic model of these sentences in which (i) they package information differently, and (ii) unlike (44a) and (44c), (44b), packages information in a way that is pragmatically optimal.

2.1.2 Overview and motivation of the Chapter

The machinery we introduce in this Chapter aims to account the above datapoints (among others), by relating the oddness of the follow-ups in (44a) and (44c) to the QuD(s) inferred from them. The fundamental principle we want to operationalize is *Question-Answer Congruence* (henceforth **QAC**), repeated in (45).

(45) *Question-Answer Congruence* (***QAC***, *Katzir and Singh*, *2015*). A felicitous assertion has to be a good answer to a good question.

Chapter 1 showed that Relevance could rule out a wide range of question-answer pairs, and as such could constitute a partial implementation of QAC. The dissertation will show that, under a certain interpretation of "good answer" and "good question", many more cases of pragmatic oddness can be understood as an accross-the-board failure of QAC.

In ths Chapter, we will lay out the groundwork for this analysis, by first introducing a more sophisticated model of questions, based on recursive partitions or trees, instead of mere partitions of the CS. This model is building on Büring (2003); Ippolito (2019); Zhang

(2022), among many others. Additionally, we will suggest that questions can be evoked by assertions in a compositional way, such that more complex sentence tend to give rise to more structurally complex questions, and also, such that sentences involving different operators (specifically, disjunctions and conditionals), give rise to different kinds of questions. In this model, each sentence may be associated with multiple potential questions. In line with QAC, a sentence which cannot be felicitously paired with *any* question will be deemed odd. This can happen if *all* the pairs formed by a sentence and a question it evokes, are themselves ill-formed. This Chapter will focus on what it takes to get there: how questions should be modeled, and how question-answer pairs are generated from simple and more complex sentences. Chapters **??** and **??** will introduce specific constraints on question-answer pairs allowing to capture data like (44).

This kind of machinery is independently motivated by the idea that sentences are never uttered in and of themselves; their purpose is to answer a question, overt or not, and to induce further questions Roberts (1996). A pragmatic model of assertion therefore needs to integrate what sentences mean, but also what kind of information structure they evoke. Unlike inquisitive semantics (Mascarenhas, 2008; Ciardelli, 2009; Groenendijk and Roelofsen, 2009; Ciardelli et al., 2018), which proposes an *unified* view of questions and assertions at the semantic level, what we propose here is a form of inquisitive *pragmatics*: sentences are still assigned "standard" extensional/intensional meaning, but also have an inquisitive contribution at the pragmatic level. In fact, the current machinery may be closer in spirit to Dynamic Semantics (Heim, 1983; Hei, 1983), where different operators give rise to different incremental updates of the Context Set. Under our view, different operators will give rise to different *parses* of the Context Set, at the inquisitive level. This will eventually allow to capture a contrast between (44c) and (44b). On top of this model, Chapters **??** and **??** will introduce constraints on pairs formed by sentences and their accommodated questions, allowing to capture the oddness of e.g. (44a) and (44c). Chapter **??** will discuss (44b), (44c), and variants thereof, arguing that some, but crucially not all variants, appear redundant once inferred QuDs are taken into consideration. Chapter **??** will also cover the case of (44a), and some of its variants.

We now proceed to define questions, not just as partition, but rather, as parse trees of the Context Set, that we will call Qtrees.

2.2 Structure of Question Trees

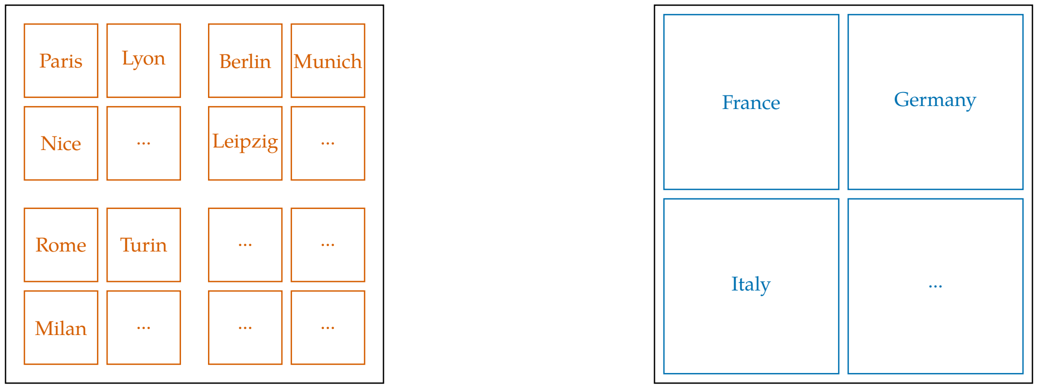
2.2.1 From partitions to recursive partitions, to parse trees

Building on the standard model presented in Chapter 1, we introduce a more sophisticated view of the pragmatics of questions. This model will incorporate the idea that questions have internal structure, and specifically, are hierarchically organized. A question such as (46a) for instance, appears more *fine-grained*, than a question like (46b). Alternatively, whatever proposition identifies a cell in (46a), also identifies a cell in (46a). This will be accounted for, as part of the internal structure of questions.

(46) a. In which city did Jo grow up?
b. In which country did Jo grow up?

First, let us observe these intuitions about question specificity are *not* readily cashed out by standard partitions or alternative set associated with questions. In the case of (46a) and (46b), these two notions coincide–*modulo* intersection with the CS. In partition talk, (46a) induces a by-city partition of the CS (see (47a) and Figure 2.1a), while (46b) induces a by-country partition (see (47b) and Figure 2.1). But nothing in (46a)'s partition signals that each of its cells is properly contained in a cell of (46b)'s partition. This property can de derived from the two structures, but is not readily *encoded* by them.

(47) a. $[\![$ In which city did Jo grow up?$]\!]^w =$
$\{p \mid \exists l.\ l$ is a city $\wedge\ p = \lambda w'.$ Jo grew up in l in $w'\}$
b. $[\![$ In which country did Jo grow up?$]\!]^w =$
$\{p \mid \exists l.\ l$ is a country $\wedge\ p = \lambda w'.$ Jo grew up in l in $w'\}$



(a) By-city partition associated with (47a). Cells are ordered on a grid for clarity only.

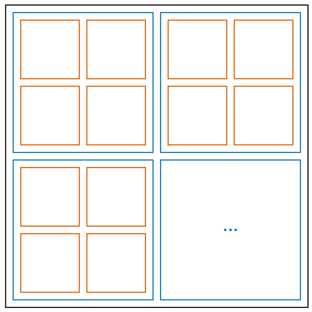
(b) By-country partition associated with (47b). Cells are ordered on a grid for clarity only.

Figure 2.1: Standard partitions induced by a fine-grained (47a) and a coarser-grained question (47b).

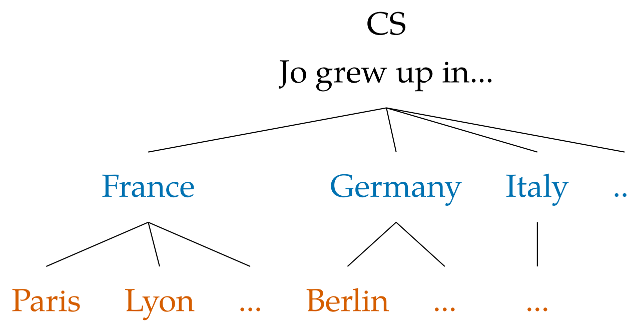
Intuitively, adding "brackets" to (47a) grouping together propositions talking about cities belonging to the same country, would help capture the desired property. This is done in (48). (48) then defines a set of sets of propositions.

(48) $[\![$ In which city did Jo grow up?$]\!]^w =$
$\{\{p \mid \exists l.\ l$ is a city in $l' \wedge\ p = \lambda w'.$ Jo grew up in l in $w'\} \mid l'$ is a country$\}$

Grouping together cells within bigger sets (which are cells themselves), amounts to building a *recursive* partition of the CS. In our example, the "outer" partition is by-country, and the "inner" partition, is by-city. Graphically, this is equivalent to adding the "blue rectangles" from Figure 2.1, to Figure 2.1a. This operation is performed in Figure 2.2a. The tree in Figure 2.2b is yet another, more readable way to represent the same thing. In this tree, each node refers to a proposition of the form *Jo grew up in l*, l denoting a city or a country. Each node is understood as intersected with the CS, which corresponds to the root of the tree. Nodes appearing at the same level (forming a "layer"),partition the CS. Deeper layers, correspond to finer-grained partitions.



(a) Recursive partition view



(b) Tree view

Figure 2.2: Alternative representations of the CS corresponding to the nested sets of (48).

(49) formally defines the bijective mapping between recursive sets of propositions (dubbed *inductive propositions*) like (48), and tree structures like Figure 2.2b.

(49) *Set-to-tree bijection.* To define this bijection, we first define inductive propositions, and their propositional content. S is an inductive proposition if either:

- S is a set of worlds (i.e. a proposition);
- S is a set of inductive propositions.

The propositional content of an inductive proposition is then defined as:

- If S is a proposition: S;
- If S is a set of inductive propositions: the grand union of the propositional contents of S's elements.

Any inductive proposition S is in a bijection with a tree structure whose nodes are propositions, and defined as:

- If S is a proposition: the tree node denoting S;
- If S is a set of inductive propositions: the tree whose root denotes S's propositional content, and whose children are the tree structures induced by each of S's elements.

We now generalize these observations, which, in fact, are not new. Building on Büring (2003); Riester (2019); Onea (2016); Ippolito (2019); Zhang (2022) (among others), we take questions to denote *parse trees* of the CS, i.e. structures that hierarchically organize the worlds of the CS. Such trees (abbreviated **Qtrees**) are defined in (50).

(50) *Structure of Question-trees* (***Qtrees***). Qtrees are rooted trees whose nodes are all subsets of the CS and s.t.:

- Their root generally[1] refers to the CS;
- Any intermediate node is a proposition, which is partitioned by the set of its children.

Before investigating the interpretation and the structural properties of this more sophisticated model of questions, the next Section covers a few core concepts from graph theory that will be useful in the rest of the Chapter and beyond.

2.2.2 A brief refresher on graph theory (and a few useful concepts for Qtrees)

(50) defines questions as rooted trees. Here, we precisify what it means to be a tree, and moreover, what is means to be a *rooted* tree (51). Let us unpack (51), and, first, disregard the "rooted" aspect of it. A tree is a kind of graph. A graph is a way to represent a binary relation, which by default will be symmetric (*directed* graphs implement asymmetric relations). Elements in the domain of the relation are modeled as nodes, and unordered pairs of nodes are connected with and edge, iff they verify the relation. A graph therefore amounts to a set of nodes, and a set of edges between these nodes. This is summarized in (52), and illustrated in Figure 2.3.

(51) *Rooted tree.* A rooted tree is a graph that is connected and acyclic, and features a distinguished node called root.

(52) (*Undirected*) *Graph.* An undirected graph (or just a graph) is defined by a set of nodes $\mathcal{N}$ and by a set of edges $\mathcal{E}$ between elements of $\mathcal{N}$. Edges are defined as unordered pairs of nodes: $\mathcal{E} \subseteq \{\{N_1, N_2\} \mid (N_1, N_2) \in \mathcal{N}^2\}$

[1] In the case of sentences carrying presuppositions, the root will be assumed to correspond to the intersection between the CS and the sentence's presupposition. In fact, the whole Qtree will be nodewise intersected with the presupposition. This will be put to use in Chapters **??** and **??**. But the examples we will see before this, will all involve Qtree rooted in the CS.

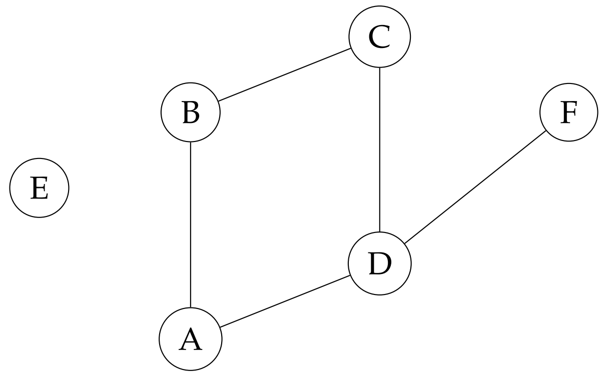


Figure 2.3: An (undirected) graph $G = (\mathcal{N}, \mathcal{E})$, with $\mathcal{N} = \{A, B, C, D, E, F\}$ and $\mathcal{E} = \{\{A, B\}, \{A, D\}, \{B, C\}, \{C, D\}, \{D, F\}\}$.

In graphs, sequences of adjacent edges form paths. For instance, in Figure 2.3, the sequence $[\{A, B\}, \{A, D\}, \{D, F\}]$ forms a path, between node A and node F. This is generalized in (53).

(53) *Path.* Let $G = (\mathcal{N}, \mathcal{E})$ be a graph. Let $(N_1, N_2) \in \mathcal{N}^2$ be two nodes of G. There is a path in G between N_1 and N_2 (abbreviated $N_1 \overset{G}{\rightsquigarrow} N_2$) iff N_1 and N_2 can be connected by a series of edges in G, i.e. $\exists (e_1, ...e_k) \in \mathcal{E}^k.\ N_1 \in e_1 \wedge N_2 \in e_k \wedge \forall i \in [1; k-1].\ |e_i \cap e_{i+1}| = 1$, where $|.|$ is the cardinality operator.

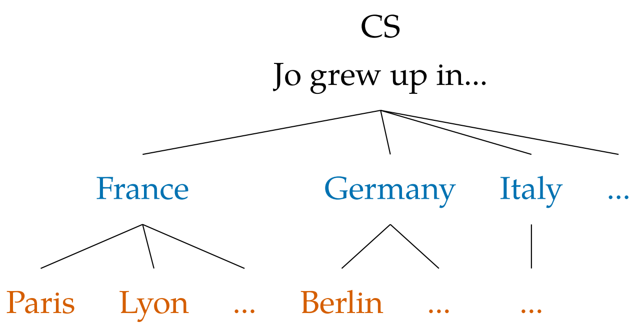
In Figure 2.3, it is easy to see that nodes A, B, C, D and F are all connected to each other by at least one path (in fact, infinitely many of them). Node E on the other hand, is isolated. So, Figure 2.3 represents a graph that is *not* connected. If E were removed from the set of nodes, and the edges remained the same, the resulting graph would be connected. This concept of connectivity is generalized in (54). If a graph is a tree, then, it is connected.

(54) *Connectivity.* Let $G = (\mathcal{N}, \mathcal{E})$ be a graph. G is connected, iff there is a path in G between any pair of nodes in $\mathcal{N}$, i.e. $\forall (N_1, N_2) \in \mathcal{N}^2.\ N_1 \overset{G}{\rightsquigarrow} N_2$.

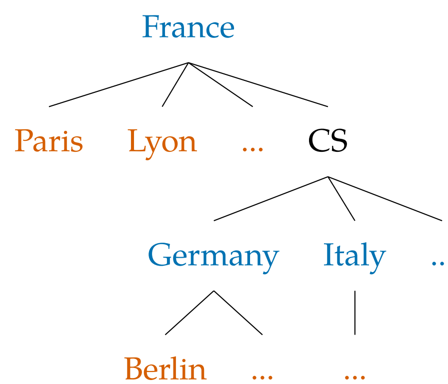
Another thing to note about Figure 2.3, is that nodes A, B, C, and D form a "cycle", there is a path that starts at one of these nodes (e.g., C), and ends at this very same node, *via* B, A, and D. Because of this cycle, there are infinitely many paths between A, B, C, and D, and also between each of these nodes, and F. Removing the edge between, say, A and B, would break the cycle (yet, interestingly, maintain connectivity between A, B, C, and D). The resulting graph would be acyclic. The general definition of an acyclic graph, is given in (55). If a graph is a tree, then, it is acyclic. Moreover, connectivity and acyclicity, are necessary and sufficient for a graph to be a tree.

(55) *Acyclicity.* Let $G = (\mathcal{N}, \mathcal{E})$ be a graph. G is acyclic, iff no node N of $\mathcal{N}$ is s.t. there is a path starting and ending at N in G, i.e. $\neg\exists N \in \mathcal{N}.\ N \overset{G}{\rightsquigarrow} N$.

We now have a definition of what kind of data structure a tree is. But why do we need Qtrees to be "rooted" then? To understand why, let us go back to the tree in Figure 2.2b, repeated in Figure 2.4a below. The way this tree is represented on paper, is somehow misleading. Recall that a tree is just an undirected graph, with a few extra properties constraining its edges. If Figure 2.4a were not "rooted", nothing would prevent us to represent it in the form of Figure 2.4b: the nodes and edges are strictly the same, but in Figure 2.4b, *France* "appears" to be the root of the tree, because it is represented at the top. To avoid this confusion, the fact that the CS node should be "at the top" is made part of the representation of the tree–which then becomes a *rooted* tree. So, a rooted tree is just a tree, plus one distinguished node that serves as root.



(a) Tree view of (46a)



(b) Alternative tree view of (46a), assuming Qtrees were not rooted

The notion of a distinguished root in fact allows to define a few interesting properties on trees, that will be used throughout the dissertation. First, once a tree is rooted, it is possible to define a measure of distance between each node of the tree, and the root. This is the concept of depth defined in (56a). In Figure 2.4a for instance, the CS has depth 0, *Germany* depth 1, and *Lyon* depth 2. This also allows to define the global "size" of the tree, in the form of its maximal depth; see (56b). Figure 2.4a for instance, is a tree of depth 2.

(56) a. *Depth of a node in a rooted tree.* Let $T = (\mathcal{N}, \mathcal{E}, R)$ be a rooted tree, with root R. Let $N \in \mathcal{N}$. The depth of N in T ($d(N, T)$) corresponds to the length of the minimal path between R and N if $N \neq R$,[2] and is set to 0 if $N = R$.

[2]This path can be determined using a simple Depth-First Search algorithm starting from the root.

b. *Depth of a rooted tree.* Let $T = (\mathcal{N}, \mathcal{E}, R)$ be a rooted tree, with root R. The depth of T $(d(T))$ is the maximal depth of a node in T: $d(T) = max_{N \in \mathcal{N}}(d(N, T))$.

Lastly, we will extensively use the concept of *layer*, that we define as a the maximal set of same-depth nodes in a rooted tree; see (57). Figure 2.4a features a country-layer at depth 1, and a city-layer at depth 2. Layers therefore reflect an intuitive notion of granularity.

(57) *Depth-k layer of a rooted tree.* Let $T = (\mathcal{N}, \mathcal{E}, R)$ be a rooted tree, with root R. Let k be an integer s.t. $0 \geq k < d(T)$. The depth-k layer of T is the set of nodes in $\mathcal{N}$ whose depth is k, i.e. $\{N \in \mathcal{N} \mid d(N, T) = k\}$.

Now that we have defined the core structure of Qtrees and a few related properties and metrics, we proceed to assign an interpretation to this kind of data structure.

2.2.3 Interpreting Qtrees

It is easy to see that the tree in Figure 2.2/2.4a , repeated in Figure 2.5, is a Qtree according to (50). We use this Qtree as an example, and assign an interpretation to nodes and paths in such structures.

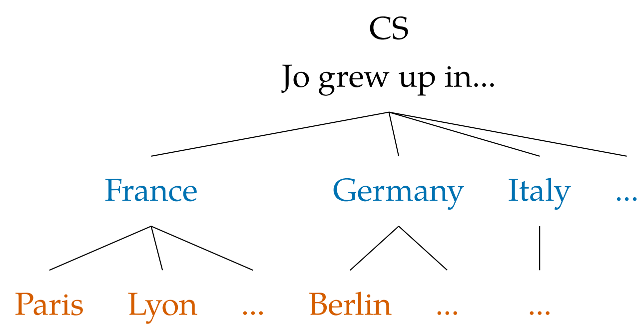


Figure 2.5: "Intuitive" Qtree for *Which city did Jo grow up in?*

Nodes, like *France*, *Paris*, or CS in Figure 2.5, can be assigned a "static" and a recursive interpretation. The static interpretation amounts to seeing a node just as what is it: an entity denoting a proposition that forms a subset of the CS. This is defined in (58).

(58) *"Static" interpretation of tree nodes.* Let $T = (\mathcal{N}, \mathcal{E}, R)$ be a rooted tree. Let $N \in \mathcal{N}$ be a node of T. N's "static" interpretation is simply N's denotation. If T is a Qtree, then N's static interpretation, is the proposition that N denotes.

Under this interpretation, the root, which denotes the whole CS, defines a tautology.[3]. Leaves, like *Paris*, *Lyon*, *Berlin* in Figure 2.5, correspond to the "smallest" cells of the recursive partition that the Qtree defines. They can therefore be seen as maximal answer to the underlying question, e.g., *In which city did Jo grow up?*. Intermediate nodes like *France* or *Germany* in Figure 2.5, form cells of "intermediate" size, and can always be seen as unions of leaves. They therefore correspond to non-maximal answers. Because Qtrees can be made of many layers, they induce a hierarchy between non-maximal answers: an non-maximal answer p is "more maximal" than another non-maximal answer q, iff the node denoting p is located deeper in the Qtree than the node denoting q. This is formalized in (59).

(59) *Answer granularity.* T be a Qtree and (N_1, N_2) be two nodes in T. N_1's static interpretation constitutes a finer-grained answer than N_2's static interpretation iff $d(N_1, T) > d(N_2, T)$. By definition, leaves of T denotes the finest-grained answers to the question T represents.

Nodes can also receive a recursive interpretation that incorporates everything the node dominates. Under this interpretation, a node N in a Qtree is not only what N denotes; it is the whole subtree ($\sim$subquestion) rooted in N. For instance, the recursive interpretation of the *France*-node in Figure 2.5, would be that of the subtree of Figure 2.5 rooted in *France*. This subtree amount to the question *In which city did Jo grow up?*, granted that *Jo lives in France*, since its root correspond to the CS, intersected with the proposition that *Jo lives in France*. This is generalized in (60). (61) further specifies the relation between a node's recursive interpretation in a Qtree, and the effect of a CS update on this Qtree.

(60) *Recursive interpretation of tree nodes*. Let $T = (\mathcal{N}, \mathcal{E}, R)$ be a rooted tree. Let $N \in \mathcal{N}$ be a node of T. N's recursive interpretation corresponds to:

- N's static interpretation if N is a leaf.
- The nodewise static interpretation of the subtree of T rooted in N, otherwise.

If T is a Qtree, then N's recursive interpretation will be the Qtree rooted in N, defined on a "local" CS updated with N's static interpretation.

(61) *Recursive interpretation and CS update.* Let T be a Qtree. Let N be a node of T. N's recursive interpretation corresponds to the nodewise intersection of T with the proposition N denotes, removing empty nodes and trivial edges.

[3]That is, an uninformative proposition that is Lewis-relevant but not Roberts-relevant, as defined in Chapter 1

refinement

Under the static interpretation, a path between the root and

Starting with nodes, they can assigned the following interpretation. The root denotes a tautology over the CS, and any other node, a possible answer to the global question denoted by the tree. Intermediate nodes can generally be seen as non-maximal answers, while leaves can be seen as maximal answers. By construction, the leaves of such trees form a partition of the CS, and as such denote "standard" questions. Any subtree rooted in a node N can be understood as conditional question taking for granted the proposition denoted by N. Finally, a path from the root to any node N can be seen as a strategy of inquiry (or a sequence of conditional questions) leading to the answer denoted by N.

answer granularity

This will eventually allow questions evoked by simple sentences to be "fused", or "chained/stacked" with each other, in order to generate questions evoked by more complex sentences. This will also allow to capture the intuition that logically related sentences may "package" information differently, and therefore exhibit different degrees of oddness.

2.2.4 Flagging Qtrees

We assume that out-of-the-blue LFs trigger a Qtree accommodation process that "retro-engineers" a Qtree from the sentence's structure.[4] When evoking a Qtree, a given LFs is assumed to "flag" specific nodes on the tree as maximal true answers. These nodes, that we dub *verifying nodes*, are typically the leaves of the Qtree which are subsets (i.e. entail) the proposition denoted by the LF. Those verifying nodes, just like the structure of the Qtree, are compositionally derived. Moreover, an accommodated Qtree should allow the sentence evoking it to properly answer it; that is why we assume that any well-formed Qtree derived from a sentence should come with a non-empty set of verifying nodes.(see (62)). More generally, we assume that oddness results from the fact that a given sentence, through its LF, cannot give rise to any well-formed Qtree. This is summarized in (63) and (64).

(62) *Empty labeling of verifying nodes.* If a sentence S evokes a Qtree T but does not flag any node as verifying on T, then T is deemed odd given S.

[4] Here, we do not cover the case of assertive sentences that are direct answers to an overt QuD. There is in fact an interesting line of work showing that overt QuDs can influence pragmatic oddness, especially when it comes to matters of redundancy (Haslinger, 2023).

(63) *Oddness of a Qtree, given a sentence.* If a sentence S evokes a Qtree T and the pair (S, T) is REDUNDANT (tbd) or induces a vacuous labeling of verifying nodes, then T is deemed odd given S.

(64) *Oddness of a sentence.* A sentence S is odd if any Qtree T it evokes is odd given S.

Before defining Qtrees for simplex LFs, let us clarify that all Qtrees will be defined and derived *modulo* a reduction function, which, given a tree T removes any empty nodes and trivial edges from T.

(65) *Qtree reduction*. If T is a tree whose nodes are sets, and endowed with a set of distinguished (e.g. verifying) nodes, a reduction of T is obtained by:

- Removing all empty nodes (and resulting dangling edges) from T;
- Removing all trivial links from T, in the following way:
  - if N has N' as only child, and neither N nor N' are verifying, replace the edge $N - N'$ by N;
  - if N has N' as only child, and either N or N' is verifying, replace the edge $N - N'$ by N, where N is labeled as verifying.

2.3 Compositional Qtrees: base case

We can then define the questions evoked by a proposition p as the partitions evoked either by p alone ($P = \{p\}$), or by p and relevant focus alternatives to p ($P = \mathcal{A}_p$) (Rooth, 1992). If p is not settled in the CS, the former kind of partition has the form $\{p, \neg p\}$ and amounts to the question of *whether p*. If $\mathcal{A}_p$ contains mutually exclusive, possible propositions covering the CS, then the partition induced by $\mathcal{A}_p$ on the CS is simply $\mathcal{A}_p$, and can be interpreted as a *wh*-question inquiring about p's focus material.

We assume that a simplex LF denoting a proposition p can give rise to two types of Qtree:[5] a "polar-question" depth-1 Qtree whose leaves are the p and $\neg p$ worlds respectively; and a "*wh*-question" depth-1 Qtree whose leaves are p and relevant, mutually exclusive alternatives to p. Moreover, verifying nodes are defined on such trees as the leaves entailing p.

(66) *Qtrees for simplex LFs* (to be extended in Chapter **??**). Let X be a simplex LF denoting p, not settled in the CS. Let $\mathcal{A}_{p,X}$ be a set of relevant focus alternatives to p

[5]This is a simplification; Chapter **??** will assume that even simplex LFs can give rise to layered Qtrees, whose layers are ordered by some notion of granularity. But this assumption is not relevant here, because we implicitely assume p and q are same-granularity alternatives.

(based on X). Let $\mathcal{A}^p_{p,X} \subseteq \mathcal{A}_{p,X}$ be the set of alternatives from $\mathcal{A}_{p,X}$ sharing same granularity with p. We assume for simplicity that $\mathcal{A}^p_{p,X}$ already partitions the CS. A Qtree for X is either:

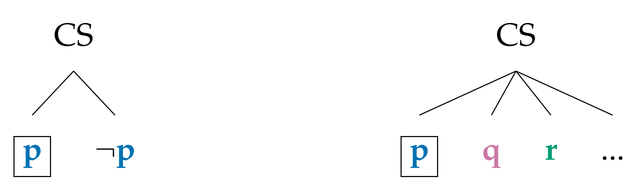
(i) A depth-1 Qtree whose leaves denote $\textsc{Partition}(\text{CS}, \{p\}) = \{p, \neg p\}$

(ii) A depth-1 Qtree whose leaves denote $\textsc{Partition}(\text{CS}, \mathcal{A}^p_{p,X}) = \mathcal{A}^p_{p,X}$.

In any case, the set of $\boxed{\text{verifying nodes}}$ for these Qtrees (signaled by $\boxed{\text{boxes}}$ in all figures) is defined as the set of their leaves that entail p.

This predicts a simplex LF denoting a tautology (e.g. a proposition entailed by the CS) to only be compatible with one Qtree, namely the Qtree whose root and unique (verifying) leaf is the whole CS. This also predicts a simplex LF denoting a contradiction (e.g. a proposition contradicted by the CS) to be compatible with Qtrees whose leaves are non-contradictory alternatives to the prejacent proposition, and whose set of verifying nodes is empty. In other words, contradictions cannot answer any suitable question and as such should be odd (see condition (62)).

Looking back at (**??**-**??**), where S_p = *Ido is at SuB* denotes p and S_q = *Ido is in Cambridge* denotes q, it is reasonable to think S_p and S_q are exclusive mutual alternatives. Other similar alternatives may be S_r = *Ido is in Paris*, S_s = *Ido is in Chicago* etc. As a result, the Qtrees compatible with S_p and S_q are given in Figures 2.6 and 2.7. Figures 2.6a and 2.7a, derived from principle (66.i), respectively model polar questions of the form: *Is Ido at SuB? Is Ido in Cambridge?* Figures 2.6b and 2.7b, derived from principle (66.ii), model a *wh*-question of the form: *Where is Ido?*.[6] At that point, it is worth observing that the "*wh*" Qtrees raised by S_p and S_q have similar structures (ignoring verifying nodes); while the corresponding "polar" Qtrees do not.



(a) "Polar" Qtree obtained from principle (66.i)

(b) "*Wh*" Qtree obtained from principle (66.ii)

Figure 2.6: Qtrees for $S_{\mathbf{p}}$ = *Ido is at SuB*. $\boxed{\text{Boxed}}$ nodes are verifying.

[6]Chapter **??** will argue that such questions are too vague to be included in Qtrees; a *where* question has to be decomposed into a series of stacked *which*-questions of increasing degrees of granularity from the top down. But assuming a *where*-question is enough for our purposes in this Chapter.

(a) "Polar" Qtree obtained from principle (66.i)

(b) "*Wh*" Qtree obtained from principle (66.ii)

Figure 2.7: Qtrees for S_{q} = *Ido is in Cambridge*. $\boxed{\text{Boxed}}$ nodes are verifying.

2.4 Compositional Qtrees: inductive step

2.4.1 Questions evoked by negated LFs

We assume negated LFs evoke questions that are structurally similar to those evoked by their non-negated counterpart. The only difference resides in the set of verifying nodes, which is flipped by negation. This is formalized in (67).[7]

(67) *Qtrees for negated LFs.* A Qtree T' for $\neg X$ is obtained from a Qtree T for X by:

- retaining T's structure;
- for every node N in the set of T's verifying nodes ($\mathbb{N}^+(T)$), add the nodes that are at the same level as N, but do not belong to $\mathbb{N}^+(T)$, to the set of T''s verifying nodes ($\mathbb{N}^+(T')$). In other words, $\mathbb{N}^+(T') = \{N' | N' \notin \mathbb{N}^+(T) \wedge \exists N \in \mathbb{N}^+(T).\ d(N', T') = d(N, T)\}$, where $d(N, T)$ denotes the depth of a node N in a tree T.[8]

Note that, if all verifying nodes are leaves, the above definition is simplified: $\mathbb{N}^+(T') = \{N' | N' \notin \mathbb{N}^+(T) \wedge \mathit{leaf}(N')\}$. Determining the verifying nodes of a Qtree T' evoked by $\neg X$, amounts to "swapping" the verifying node of its structural counterpart T evoked by X. Qtrees for $\neg S_p$ and $\neg S_q$, derived using this simple recipe, are given in Figures 2.8 and 2.9.

[7]This approach is perhaps a bit naive; uttering p vs. $\neg p$, does not seem to preferentially answer the same kind of question, i.e. evoke the same kind of Qtree structure. More specifically, it seems that uttering negative statements in general conveys the idea that the original question was a polar question of the form *whether p?* – more than a *wh* kind of question. We discuss this more in the context of Chapter **??**.

[8]Because T and T' have same structure, the tree-argument is irrelevant to determine node depth in that particular case: $\forall N.\ d(N, T') = d(N, T)$. We keep it because, in the general case, node-depth depends on tree structure.

(a) Derived from Tree 2.6a (b) Derived from Tree 2.6b

Figure 2.8: Qtrees for $\neg S_{\mathrm{p}}$ = *Ido is **not** at SuB*. $\boxed{\text{Boxed}}$ nodes are verifying.

(a) Derived from Tree 2.7a (b) Derived from Tree 2.7b

Figure 2.9: Qtrees for $\neg S_{\mathrm{q}}$ = *Ido is **not** in Cambridge*. $\boxed{\text{Boxed}}$ nodes are verifying.

2.4.2 Questions evoked by conditional LFs

Influential work in psychology (Wason, 1968), showed that, when asked to verify the truth of a conditional statement, participants tend to massively overlook the eventualities falsifying the antecedent.[9] Building on this finding, and insights from the recent linguistic literature (Aloni, 2022), we assume conditional LFs preferentially evoke questions pertaining to their consequent, *in the domain(s) of the CS where the antecedent holds*. The antecedent of a conditional LF therefore plays the role of a question "restrictor", rather than a question "generator".

This assumption introduces an asymmetry between antecedent and consequent. Together with a QuD-driven notion of redundancy, it will eventually explain why structurally redundant sentences like (**??**) whose re-occurring material is in the antecedent of a conditional, can escape a violation of redundancy: in such structures, the seemingly redundant antecedent acts as a QuD restrictor and is thus not treated as verifying in the resulting Qtree structure. This in turn will make the resulting Qtree distinct from alternative Qtrees evoked by structurally simpler LFs. In other words our view on conditional Qtrees will allow (**??**) to have a distinctive, efficient inquisitive contribution.

(68) and (69) define conditional Qtrees as Qtrees evoked by the antecedent of the conditional, but whose verifying nodes get replaced by their intersection with a Qtree evoked

[9]It is however interesting to note that this result is sensitive to the Question under Discussion provided by the experimental paradigm: when the QuD pertains to detecting violations of social contracts, individuals appear more classically logical Cosmides (1989).

by the consequent (*modulo* reduction). This process is assumed to filter out the outputs that do not qualify at Qtrees. The core idea behind this operation is that conditionals do not make antecedent and consequent QuDs at issue at the same time; rather, they introduce a hierarchy between these two objects, by raising the consequent QuD only in the cells of the CS (as defined by the antecedent QuD), where the antecedent holds. Yet another way to phrase this is by saying that, through the process of Qtree-conditionalization, the consequent Qtree gets *restricted* by the antecedent Qtree.

(68) *Qtrees for conditional LFs.* A Qtree T for $X \rightarrow Y$ is obtained from a Qtree T_X for X and a Qtree T_Y for Y by:

- replacing each node N of T_X that is in $\mathbb{N}^+(T_X)$ with $N \cap T_Y$ (see (69));
- returning the result only if it is a Qtree.

In other words, $Qtrees(X \rightarrow Y) = \{T_X \cup \bigcup_{N \in \mathbb{N}^+(T_X)} (N \cap T_Y) | (T_X, T_Y) \in Qtrees(X) \times Qtrees(Y) \wedge T_X \cup \bigcup_{N \in \mathbb{N}^+(T_X)} (N \cap T_Y)$verifies (50)$\}$, and $\mathbb{N}^+(T_X \rightarrow T_Y) = \{N \cap N' | (N, N') \in \mathbb{N}^+(T_X) \times \mathbb{N}^+(T_Y) \wedge N \cap N' \neq \emptyset\}$.

(69) *Node-Qtree intersection.* If N is a node (set of worlds) and T a Qtree, $N \cap T_Y$ is defined as T_Y, where each node gets intersected with N and empty nodes as well as trivial ("only child") links get removed (in line with (65)); and where T_Y's verifying nodes are preserved.

The Node-Qtree intersection operation is schematized in (2.10).

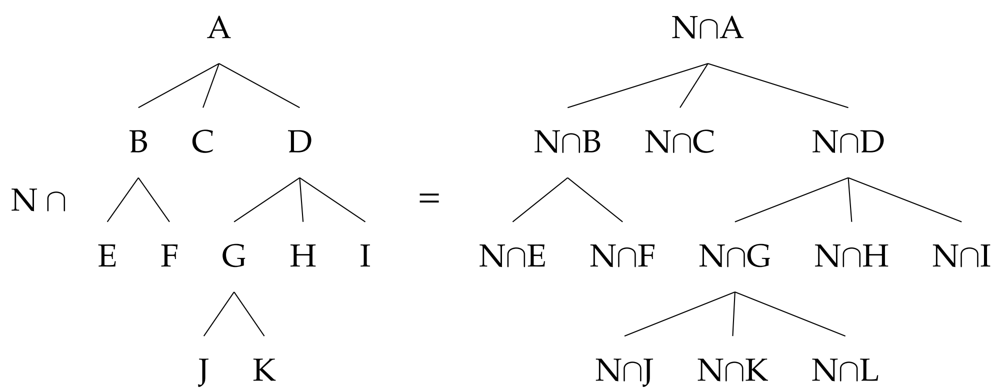


Figure 2.10: Node-Qtree intersection.

There are two additional things to note about this operation. First, the verifying nodes of a conditional Qtree are inherited from its input *consequent* Qtree; meaning, verifying nodes contributed by the antecedent Qtree are *disregarded*. This is in line with the idea that, when checking the truth of natural language conditionals, speaker tend to overlook the possible the falsity of the antecedent. The whole operation is schematized in Figure 2.11.

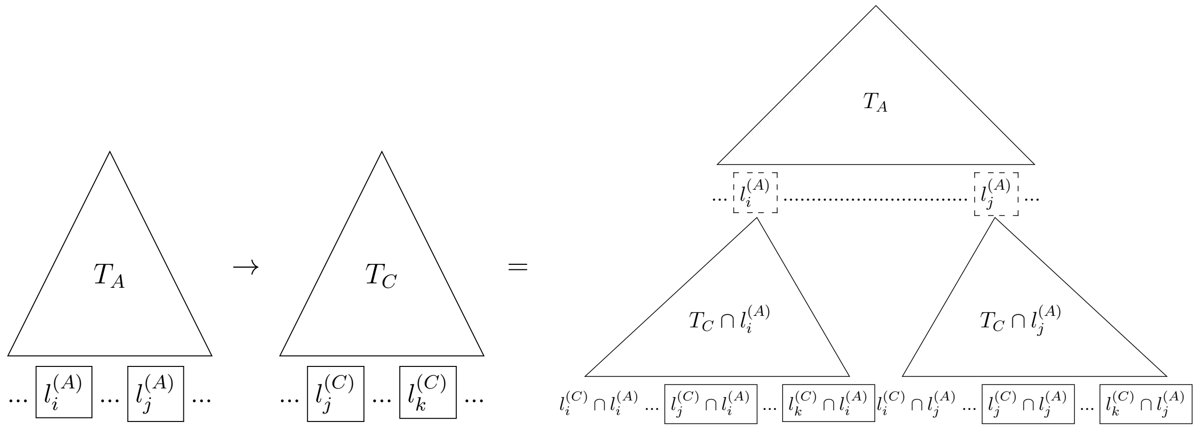


Figure 2.11: Schema of the derivation of a conditional Qtree. Nodes in dashed boxes refer to the nodes that were verifying in the input antecedent Qtree, but are no longer verifying in the output conditional Qtree. Nodes in $\boxed{\text{solid boxes}}$ refer to the nodes that were verifying in the input consequent Qtree, and are thus still verifying in the output conditional Qtree.

Second, the Node-Qtree intersection operation ($N \cap T$), which is part of the conditional Qtree formation process, is "vacuous" iff N entails a specific leaf in T. We call the operation $N \cap T$ vacuous if it outputs N; the status of N as verifying still depends on T's verifying nodes. This is exemplified in Figure (2.12) assuming the node N intersecting the Qtree entails (i.e. is a subset of) the leaf labeled L in T. What happens is the following. The definition of a Qtree (see (50)) states that each intermediate node is partitioned by the set of its children. A corollary of this definition, is that, given a leaf L, all the nodes present on the path from L to the root will be supersets of L, while all the other nodes will have no overlap with L. So, if $N \subseteq L$, N will be a subset of all the nodes in L's path to the root, and have no overlap with the other nodes, as well. Performing $N \cap T$ will thus initially yield a tree with same structure as the input Qtree T, but with nodes equal to N along the path between the root and L's original position, and empty nodes in all other positions. The Qtree reduction process devised in (65) then removes all these empty nodes, and collapses the path made of N-nodes into one single node, namely, N. The whole operation therefore returns the input node N. Because the status of being a verifying node percolates when reduction takes place, as per (69), the output N will be verifying iff L was in T.

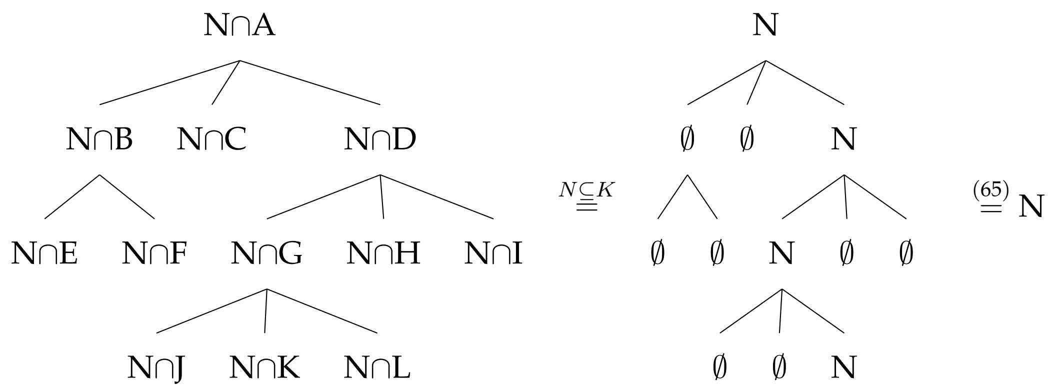
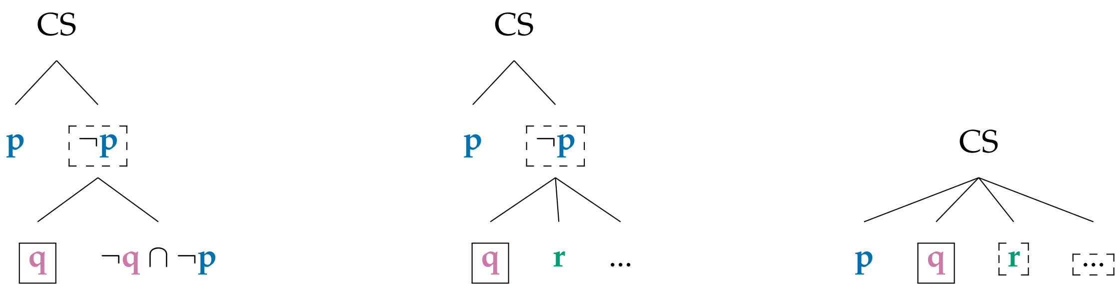


Figure 2.12: Vacuous Node-Qtree intersection if N entails a leaf of T, e.g. K.

The whole conditional Qtree formation process will then be vacuous if each verifying leaf in the antecedent Qtree entails a specific leaf of the consequent Qtree. Moreover, if each verifying leaf in the antecedent Qtree entails a specific *non-verifying* leaf of the consequent Qtree, the output Qtree will be structurally identical to the antecedent Qtree but, will be left with *no* verifying node. Such a tree will be deemed ill-formed as per principle (62).

We are now equipped to build conditional Qtrees from the sentences S_p=*Ido is at SuB*, $\neg S_p$=*Ido is not at SuB*, S_q=*Ido is in Cambridge*, and $\neg S_q$=*Ido is not in Cambridge*, whose Qtrees where computed in the previous Sections. This is done for $\neg S_p \rightarrow S_q$ in Figure 2.13, using Qtrees for $\neg S_p$ from Figure 2.8 and Qtrees for S_q from Figure 2.7. Figure 2.14, does the same for $\neg S_q \rightarrow S_p$, just swapping the roles of p and q. It is worth noting that the Qtrees in Figure (2.13c) and (2.14c) are structurally identical to the antecedent Qtree used to build them. Such Qtrees are thus examples of a vacuous application of the Node-Qtree intersection operation. Their verifying nodes are however different from those of their antecedent Qtree, since, by definition, they are inherited from their consequent Qtree.

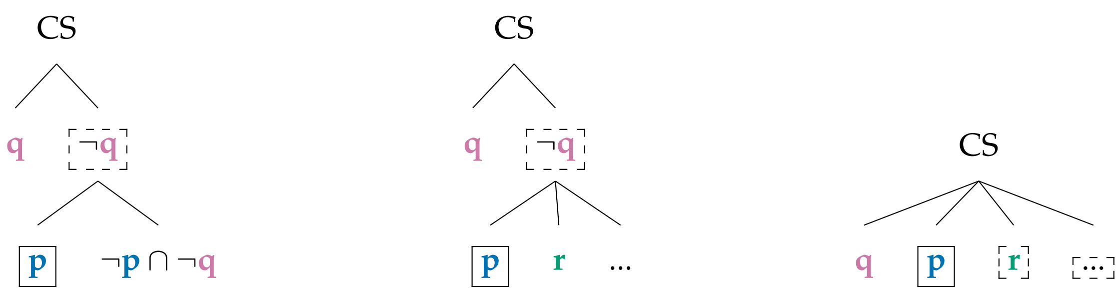


(a) Figure (2.8a) → Figure (2.7a)

(b) Figure (2.8a) → Figure (2.7b)

(c) Figure (2.8b) → Figure (2.7a)/(2.7b)[10]

Figure 2.13: Qtrees for $\neg S_{\text{p}} \to S_{\text{q}}$ = *If Ido is not at SuB then he is in Cambridge*. Nodes in dashed boxes refer to the nodes that were verifying in the input antecedent Qtree, but are no longer verifying in the output conditional Qtree. Nodes in $\boxed{\text{solid boxes}}$ refer to the nodes that were verifying in the input consequent Qtree, and are thus still verifying in the output conditional Qtree.



(a) Figure (2.9a) → Figure (2.6a)

(b) Figure (2.9a) → Figure (2.6b)

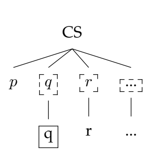
(c) Figure (2.9b) → Figure (2.6a)/(2.6b)

Figure 2.14: Qtrees for $\neg S_{\text{q}} \to S_{\text{p}}$ = *If Ido is not in Cambridge then he is at SuB*; obtained *mutatis mutandis* from Figure 2.13

At that point, we can already observe that Qtrees built from $\neg S_p \to S_q$, do not flag the p-node as verifying, since this corresponds to falsifying the antecedent of the conditional, a strategy that is typically overlooked. This feature of the model will be crucial in deriving the felicity of (**??**): because p is not treated as verifying in the Qtrees in Figure 2.13, it will

[10] This Qtree is derived *via* intersection and reduction as defined in (69). The Qtree derived *before* reduction is given in (i). Reduction on this Qtree collapses the two q-nodes and makes the resulting node verifying; collapses the two r-nodes and makes the resulting node non-verifying; and so on for all other nodes different from the p-node.

(i)



be possible to disjoin them with a Qtree for S_p, without producing redundant Qtrees as output. To clarify this intuition, we proceed to defining disjunction over Qtrees.

2.4.3 Questions evoked by disjunctive LFs

Building on Simons (2001); Zhang (2022), we assume disjunctive LFs evoke questions pertaining to both disjuncts *in parallel*. In other words, disjuncts should mutually address each-other's questions. This is modeled in (70), by assuming that disjunctions return all possible unions of the Qtrees evoked by both disjuncts, filtering out the outputs that do not qualify at Qtrees.

(70) *Qtrees for disjunctive LFs.* A Qtree T for $X \vee Y$ is obtained from a Qtree T_X for X and a Qtree T_Y for Y by:

- unioning the nodes, edges, and verifying nodes of T_X and T_Y;
- returning the output only if it is a Qtree.

In other words, $Qtrees(X \vee Y) = \{T_X \cup T_Y | T_X \cup T_Y \text{verifies (50)} \wedge (T_X, T_Y) \in Qtrees(X) \times Qtrees(Y)\}$

A prediction of this definition is that two Qtrees sharing the same CS can be properly disjoined only iff they appear structurally parallel up to a certain level, and any further partitionings they independently introduce do not "clash" with each other.[11]

The only possible Qtree for $S_p \vee S_q$ / $S_q \vee S_p$ is given in Figure 2.15. It is obtained from Qtrees 2.6b and 2.7b, which, as previously observed, have similar structures. As intuitively expected, it is a Qtree that inquires about p and about q *at the same time* (i.e., as part of the same subquestion), since the p and q nodes appear at the same level of the tree. Other possible unions of Qtrees are shown in Figure 2.16 but appear ill-formed, because the leaves of such Qtrees do not properly partition the CS. Following a similar line of

[11] We assume two Q-trees T and T' feature a bracketing clash iff there is $N \in T$ and $N' \in T'$ s.t. $N = N'$ but the sets of children of N and N' differ. We show that if T and T' exhibit such a clash, their disjunction is not a Q-tree. Let's call C and C' the sets of nodes of resp. T and T' that induce a bracketing clash; by assumption, C and C' are s.t. $C \neq C'$, and have mothers N and N' s.t. $N = N'$. Because $\vee$ achieves graph-union, $T \vee T'$ will have a node N with $C \cup C'$ as children, and because $C \neq C'$, $C \cup C' \supset C, C'$. Given that both C and C' are partitions of N, $C \cup C'$ cannot be a partition of N. Conversely, if two Q-trees T and T' sharing the same CS as root are s.t. their union $T \cup T'$ is not a Qtree, it must be because T and T' had a bracketing clash. Indeed, under those assumptions, $T \cup T'$ not being a Qtree means one node N in $T \cup T'$ is not partitioned by its children. Given N is in $T \cup T'$, N is also in T, T', or both. If N was only in, say, T, then it means N's children are also only in T, but then, T itself would have had a node not partitionned by its children, contrary to the assumption T is a Qtree. The same holds *mutatis mutandis* for T', so, N must come from *both* T and T'. Let us call C and C' the partitioning introduced by N in resp. T and T'. The fact C, C', but not $C \cup C'$ partition N entails $C \neq C'$, i.e. T and T' feature a bracketing clash.

reasoning, one can use the Qtrees in Figures 2.6b (for p) and 2.15 (for $p \vee q$), to derive the only possible Qtree for (**??**) $= p \vee (p \vee q)$. Because Qtrees 2.6b and 2.15 have same structure, and are s.t. the former Qtree's set of verifying nodes is a subset of the latter Qtree's set of verifying nodes, Qtree union simply returns Qtree 2.15 as output. So Qtree 2.15 turns out to be compatible with both $p \vee q$ and (**??**).

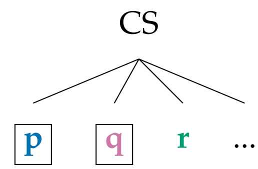
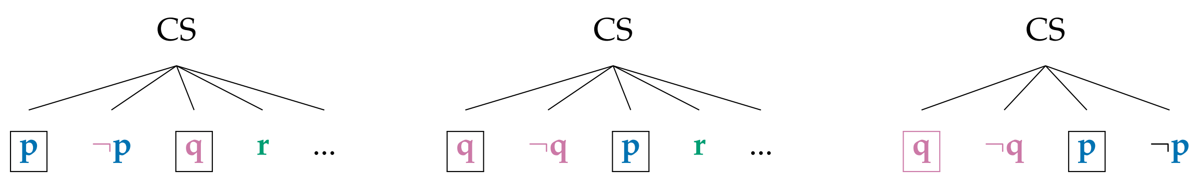


Figure 2.15: Only well-formed Qtree evoked by $S_{\mathbf{p}} \vee S_{\mathbf{q}}$ = *Ido is at SuB or in Cambridge*, obtained from Qtrees 2.6b and 2.7b. **This Qtree is also the only Qtree compatible with (??).**

Figure (2.15) already gives us a hint as to why (**??**) is degraded: there is an expression, namely $p \vee q$, that is strictly simpler than (**??**) (and in fact, a proper simplification of (**??**)), that accommodates the exact same Qtree (including verifying nodes). In that sense, (**??**) appears suboptimal. This will be formalized in the next Section.



(a) Figure 2.6a $\vee$ Figure 2.7b (b) Figure 2.6b $\vee$ Figure 2.7a (c) Figure 2.6a $\vee$ Figure 2.7a

Figure 2.16: Ill-formed Qtrees resulting from the union of the Qtrees in Figures 2.6 and 2.7. Red nodes are nodes that should be removed for the leaves to form a proper partition of the CS.

Chapter 3

Comparison of the Qtree model to earlier similar approaches

Abstract. This Chapter consists in a literature review and compares the model of questions introduced in Chapter 2 to earlier approaches accounting for oddness phenomena *via* theories of questions or alternatives. It is shown that these earlier models differ from the current framework in three possible ways: (i) the core model is technically very similar, but at the conceptual level assertions are not taken to evoke full-fledged questions (Ippolito, 2019), or (ii) the machinery proposed *is* based on evoked QuDs but not fully compositional (Zhang, 2022), or (iii) question semantics is taken to fully *replace* standard propositional content (the Inquisitive Semantics framework).

STRESSS OPTIONALITY

3.1 Inquisitive Semantics

inquisiyive semantics says that sentences are more or less questions, they raise issues. but paradox:sentences and qs are the same kind of thing, but then, sentences get impoverished to be made diff from qs eventually

3.2 Ippolito's contribution

Ippolito (2019) proposes a model of alternatives that is very close in its implementation to the Qtree model proposed in the first half of this Chapter. Under Ippolito's view, the way alternatives are structured is seen as a source of oddness. But, as a whole, the account will be shown to differ from ours in at least three respects: first, sentences are not taken to evoke full-fledged questions (a mainly conceptual difference); second, it leaves unexplained when, and how, sets of alternatives can be combined, cross-sententially and in biclausal sentences; third, under this view oddness arises from a purely structural constraint (the *Specificity Constraint*), that appears independent from familiar competition-based pragmatic principles. The current section will present the account and outline the first two differences. Chapter **??** will further clarify the third difference, by introducing a new, competition-based REDUNDANCY constraint on LF-Qtree pairs.

3.2.1 The data

Ippolito (2019)'s goal was to provide a unified analysis of a number of seemingly independent instances of pragmatic oddness, taking the form of Sobel sequences (71), sequences of superlatives (72), and Hurford Disjunctions (73).

(71) a. If the USA had thrown their nuclear weapons into the sea, there would have been war. But if all the nuclear powers had thrown their weapons into the sea, there would have been peace.

b. # If all the nuclear powers had thrown their nuclear weapons into the sea, there would have been peace. But if the USA had thrown their weapons into the sea, there would have been war.

(72) a. The closest gas stations are crummy; but the closest Shell stations are great.

b. # The closest Shell stations are great; but the closest gas stations are crummy.

(73) a. John ate some of the cookies or all of them.

b. # John ate all of the cookies or some of them.

These three families of sentences share commonalities. In all three configurations, two sentences or fragments are being contrasted using connectives like *but* and *or*. For instance, in the Sobel case (71a), *If the USA had thrown their nuclear weapons into the sea, there would have been war* gets contrasted with *If all the nuclear powers had thrown their nuclear weapons into the sea, there would have been peace*. Additionally, in all three cases, the two sentences being contrasted exhibit some degree of parallelism, in the sense that they each contain a subconstituent C/C^+, such that $[\![C^+]\!] \vdash [\![C]\!]$. For instance, *all the nuclear powers had thrown their nuclear weapons into the sea*, entails that *the USA had thrown their nuclear weapons into the sea*. Lastly, all configurations are such that the a. examples, which start with the sentences containing the "weaker" C, appear more felicitous than the b. examples, which start with the sentences containing the "stronger" C^+.

3.2.2 Structured Sets of Alternatives

To account for these asymmetries, Ippolito (2019) submits that the alternatives evoked by assertive sentences form "structured sets" (henceforth **SSA**). Such sets are defined in (74). The kind of structures generated by this definition are in essence recursive partitions of the CS, or Qtrees, as defined in (50).[1]

(74) *Structured Set of Alternatives* (*SSA*) (*Ippolito, 2019*). $T_{\mathcal{A}}$ is a well-formed structured set of alternatives iff the following conditions are met:

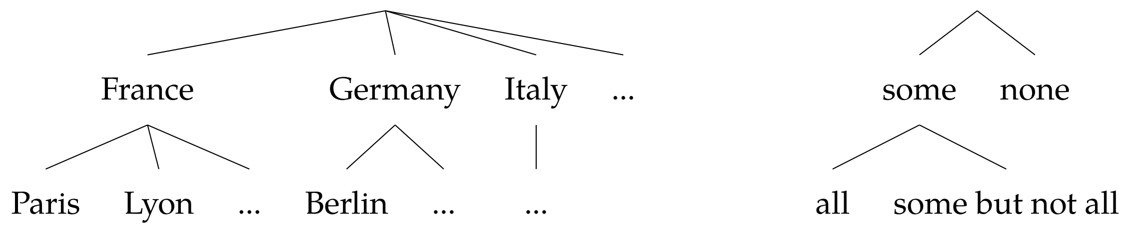
- Strength: for any two alternatives α, $\beta \in \mathcal{A}$, β is the daugther of α in $T_{\mathcal{A}}$ just in case $[\![\beta]\!] \subset [\![\alpha]\!]$.
- Disjointness: for any two alternatives β_1, $\beta_2 \in \mathcal{A}$, if β_1 and β_2 are sisters in $T_{\mathcal{A}}$, then $[\![\beta_1]\!] \cap [\![\beta_2]\!] = \emptyset$
- Exhaustivity: for any alternative α with daughters $\beta_1, ...\beta_n$, in $T_{\mathcal{A}}$, $[\![\beta_1]\!] \cup [\![\beta_2]\!] \cup ... \cup [\![\beta_n]\!] = [\![\alpha]\!]$

Alternatives evoked by an assertion are modeled following Rooth (1992), i.e. assumed to be obtained by substituting the original sentence's focused material by any expression of the same type. This is spelled out in (75).

(75) *Focus alternatives* (*Rooth, 1992*). Let S be a sentence containing a focused element α. The set of focus alternatives to $[\![S]\!]$ is the set of propositions $[\![S']\!]$, where S' is obtained from S by substituting α with any element of the same type as α.

Figure 3.1 illustrates SSAs for simple sentences containing scalar and non-scalar alternatives. It is worth noting that sentences associated with different degrees of granularity (e.g. *Jo grew up in Pairs* vs. *Jo grew up in France*) are not expected to give rise to different SSAs, as shown in Figure 3.1a. Same holds for scalar sentences in an entailment relation (e.g. *Jo ate some of the cookies* vs. *Jo ate all of the cookies*).

[1] This is what at least is argued in Ippolito (2019). It is worth mentioning however, that the definition in (74) does not in itself guarantee that any Structured Set of Alternatives should form a tree. Instead, it guarantees that any branching of the form $[_\alpha \beta_1 ... \beta_n]$ is s.t. $(\beta_i)_{i \in [1;n]}$ partitions α. But nothing in principle guarantees the connectedness of the structure: if specific alternatives happen to be "missing" (for relevance/QuD-related reasons, or perhaps due to a missing lexicalization), then, the resulting Structured Set of Alternatives may end up being a forest, instead of a single tree.



(a) SSA associated with *Jo grew up in* $Paris_F$/$France_F$/$Italy_F$ etc.

(b) SSA associated with *Jo ate* $some_F$/all_F/$none_F$ *of the cookies*.

Figure 3.1: SSAs for simple focused sentences.

Additionally, alternatives are assumed to be constrained by "the" QuD. This constitutes the first, conceptual difference with our account introduced earlier in this Chapter: under Ippolito's view, assertions are not assumed to help determine "the" QuD; instead, they are assumed to evoke alternatives, which are themselves constrained by "the" QuD. In other words, SSAs are not expected to help determine what "the" QuD is–they are partially derived from it. This is far from an esoteric perspective, and appears in line with much past literature. What we want to propose instead, is the reverse perspective: assertions and their alternatives are the primitive, and help *derive* potential QuDs (along with contrasts in pragmatic oddness).

3.2.3 The Specificity Constraint

Ippolito (2019) then proposes that oddness arises from certain SSA configurations. In particular, sequences of sentences belonging to the same SSA are subject to a Specificity Constraint (henceforth **SC**), spelled out in (76). The SC states that the two alternatives in the sequence, should be dominated by the same number of nodes in their common SSA. This is equivalent to saying that two alternatives being contrasted should match in terms of their degree of specificity, or granularity.

(76) *Specificity Condition* (*Ippolito, 2019*). A sequence $\Sigma =< [_{S_i} ...\alpha_F...], [_{S_j} ...\beta_F...] >$, s.t. both S_i and S_j are answers to the same QuD and β is in the structured set of alternatives evoked by α ($T_{\mathcal{A}_\alpha}$), is felicitous if either:

- α or β is the only node on its branch in $T_{\mathcal{A}_\alpha}$, or
- α and β are dominated by the same number of nodes in $T_{\mathcal{A}_\alpha}$.

A sentence like (73b) then violates the SC, because its "*all*" and its "*some*" disjunct are respectively dominated by 2, and 1 node in the corresponding SSA from Figure 3.1b. The

SC therefore correctly predicts (73b) to be odd. But, because (73a) only differs from (73b) in how the disjuncts are ordered, the SC also incorrectly predicts (73a) to be odd–at least in the absence of any additional assumptions.

The felicity of (73a) is captured in Ippolito (2019)'s framework based on the familiar idea that violations of the SC can be avoided by strengthening the weaker alternative (Gazdar, 1979; Singh, 2008b,a; Chierchia et al., 2009; Fox, 2018). To retain the *contrast* between (73b) and (73a), it is assumed that covert strengthening is governed by an economy condition, which disallows it in (73b). This is shown to generalize to the a. and b. sequences in (71-72).

Even though the SC appears like a reasonable constraint, the deep reason why contrast alternatives with different degrees of specificity should be disallowed, remains relatively mysterious. In particular, the account does not directly relate the SC to general pragmatic principles based on competition *between* sentences: the SC is a constraint that is only sensitive to the SSA associated with the target sentence, independently of the sentence's competitors and their own SSAs. In that respect, it remains close to Hurford's original constraint. Moreover, the constraint amounts to counting the number of parent nodes for each contrasted alternative, and as such is not sensitive to the relative positions of the two alternatives within their common SSA. This perspective might be slightly reductive, and would not capture the observation that oddness gets stronger if the two alternatives are in a dominance relation, as shown by gradience of the judgments in (77).

(77) a. # Jo grew up in Paris or France. Different specificity, dominance
b. ? Jo grew up in Paris or Germany. Different specificity,no dominance
c. Jo grew up in France or Germany. Same specificity,no dominance

In Chapter **??**, we will propose a constraint akin in effect to the SC, but that will constitute a more direct extension of earlier Redundancy-based constraints used to capture Hurford Disjunctions. We will show how it applies to basic (non-scalar) Hurford Disjunctions and extends to another challenging family of intuitively redundant sentences. Chapters **??** and **??** will discuss the particular case of scalar Hurford Disjunctions like (73), and extend the account to scalar Sobel sequences.

3.3 Zhang's

Bibliography

(1983). On the projection problem for presuppositions. In Barlow, M., Flickinger, D. P., and Wescoat, M. T., editors, *Proceedings of the Second West Coast Conference on Formal Linguistics*, pages 114–126, Stanford, CA. Stanford University Department of Lingustics.

Aloni, M. (2022). Logic and conversation: The case of free choice. *Semantics and Pragmatics*, 15(5):1–60.

Büring, D. (2003). On D-Trees, Beans, and B-Accents. *Linguistics and Philosophy*, 26(5):511–545.

Chierchia, G., Fox, D., and Spector, B. (2009). Hurford's constraint and the theory of scalar implicatures: Evidence for embedded implicatures. In Egré, P., , and Magri, G., editors, *Presuppositions and implicatures*, pages 47–62, Cambridge, MA. MIT Press.

Ciardelli, I. (2009). Inquisitive semantics and intermediate logics. Master's thesis, University of Amsterdam.

Ciardelli, I., Groenendijk, J., and Roelofsen, F. (2018). *Inquisitive Semantics*. Oxford University Press.

Cosmides, L. (1989). The logic of social exchange: Has natural selection shaped how humans reason? studies with the wason selection task. *Cognition*, 31(3):187–276.

Fox, D. (2007). *Free Choice and the Theory of Scalar Implicatures*, page 71–120. Palgrave Macmillan UK.

Fox, D. (2018). Partition by exhaustification: comments on dayal 1996. *ZAS Papers in Linguistics*, 60:403–434.

Gazdar, G. (1979). Pragmatics, Implicature, Presuposition and Logical Form. *Critica*, 12(35):113–122.

Ginzburg, J. (1996). Dynamics and the semantics of dialogue. In Seligman, J. and Westerstahl, D., editors, *Language, Logic, and Computation*, volume 1, Stanford, CA: CSLI. University of Chicago Press.

Grice, H. P. (1975). Logic and conversation. In Davidson, D., editor, *The logic of grammar*, pages 64–75. Dickenson Pub. Co.

Grice, H. P. (1989). *Studies in the Way of Words*. Harvard University Press, Cambridge, MA.

Groenendijk, J. (1999). The logic of interrogation: Classical version. *Semantics and Linguistic Theory*, 9:109.

Groenendijk, J. and Roelofsen, F. (2009). Inquisitive Semantics and Pragmatics. In *Workshop on Language, Communication and Rational Agency*.

Groenendijk, J. and Stokhof, M. (1984). *Studies in the semantics of questions and the pragmatics of answers*. PhD thesis, University of Amsterdam. (Unpublished doctoral dissertation).

Hamblin, C. L. (1958). Questions. *Australasian Journal of Philosophy*, 36(3):159–168.

Hamblin, C. L. (1973). Questions in montague english. *Foundations of Language*, 10(1):41–53.

Haslinger, N. (2023). *Pragmatic constraints on imprecision and homogeneity*. PhD thesis, Georg-August-Universität Göttingen.

Heim, I. (1982). *The Semantics of Definite and Indefinite Noun Phrases*. PhD thesis, UMass Amherst.

Heim, I. (1983). *File Change Semantics and the Familiarity Theory of Definiteness*, page 164–189. DE GRUYTER.

Hulstijn, J. (1997). Structured information states: Raising and resolving issues.

Hurford, J. R. (1974). Exclusive or Inclusive Disjunction. *Foundations of Language*, 11(3):409–411.

Hénot-Mortier, A. (2023). Alternatives are blind to some but not all kinds of context: the view from Hurford Disjunctions. *Proceedings of Sinn und Bedeutung*, 27:291–308.

Hénot-Mortier, A. (2024a). "One tool to rule them all"? An integrated model of the QuD for Hurford sentences. In *Proceedings of the 29th Sinn und Bedeutung*. Draft (`https://adelemortier.github.io/files/SuB_2024_TMT_paper.pdf`), to appear.

Hénot-Mortier, A. (2024b). Redundancy under Discussion. In *Proceedings of the 29th Sinn un Bedeutung*. Draft (`https://adelemortier.github.io/files/SuB_2024_main_session_paper.pdf`), to appear.

Ippolito, M. (2019). Varieties of sobel sequences. *Linguistics and Philosophy*, 43(6):633–671.

Jäger, G. (1996). Only updates: On the dynamics of the focus particle only. In *Proceedings of the 10th Amsterdam Colloquium*.

Kalomoiros, A. (2024). An approach to Hurford Conditionals. *Semantics and Linguistic Theory*, page 724–743.

Karttunen, L. (1977). Syntax and semantics of questions. *Linguistics and Philosophy*, 1(1):3–44.

Katzir, R. and Singh, R. (2014). Hurford disjunctions: embedded exhaustification and structural economy. In *Proceedings of Sinn und Bedeutung*, volume 18, pages 201–216.

Katzir, R. and Singh, R. (2015). Economy of structure and information: Oddness, questions, and answers. *Proceedings of Sinn und Bedeutung*, 19:322–339.

Kaufmann, M. (2016). Free Choice is a Form of Dependence. *Natural Language Semantics*, 24(3):247–290.

Kroch, A. (1972). Lexical and inferred meanings for some time adverbs. *Quarterly Progress Report of the Research Laboratory of Electronics 104*.

Lewis, D. (1988). Relevant Implication. *Theoria*, 54(3):161–174.

Mandelkern, M. and Romoli, J. (2018). Hurford Conditionals. *Journal of Semantics*, 35(2):357–367.

Marty, P. and Romoli, J. (2022). Varieties of Hurford disjunctions. *Semantics and Pragmatics*, 15(3):1–25.

Mascarenhas, S. (2008). Inquisitive semantics and logic. Master's thesis, University of Amsterdam. Manuscript, University of Amsterdam.

Mayr, C. and Romoli, J. (2016). A puzzle for theories of redundancy: Exhaustification, incrementality, and the notion of local context. *Semantics and Pragmatics*, 9(7).

Meyer, M.-C. (2013). *Ignorance and Grammar*. PhD thesis, Massachusetts Institute of Technology.

Onea, E. (2016). *Potential Questions at the Semantics-Pragmatics Interface*. BRILL.

Riester, A. (2019). *Constructing QUD Trees*, page 164–193. BRILL.

Roberts, C. (1996). Information structure in discourse: Towards an integrated formal theory of pragmatics. In Yoon, J. H. and Kathol, A., editors, *Ohio State University Working Papers in Linguistics*, volume 49. Ohio State University.

Roberts, C. (2012). Information structure in discourse: Towards an integrated formal theory of pragmatics. *Semantics & Pragmatics*, 5:1–69.

Rooth, M. (1992). A theory of focus interpretation. *Natural Language Semantics*, 1(1):75–116.

Schlenker, P. (2009). Local contexts. *Semantics and Pragmatics*, 2.

Simons, M. (2001). Disjunction and Alternativeness. *Linguistics and Philosophy*, 24(5):597–619.

Singh, R. (2008a). *Modularity and locality in interpretation*. PhD thesis, MIT.

Singh, R. (2008b). On the interpretation of disjunction: asymmetric, incremental, and eager for inconsistency. *Linguistics and Philosophy*, 31(2):245–260.

Spector, B., Fox, D., and Chierchia, G. (2009). Hurford's Constraint and the Theory of Scalar Implicatures: Evidence for embedded implicatures. In Égré, P. and Magri, G., editors, *Presuppositions and Implicatures: Proceedings of the MIT-Paris Workshop*.

Stalnaker, R. (1974). Pragmatic Presuppositions. In Munitz, M. K. and Unger, P. K., editors, *Semantics and Philosophy*, page 197–213. New York University Press, New York.

Stalnaker, R. (1978). Assertion. *Syntax and Semantics*, 9:315–332.

Stalnaker, R. (1999). Pragmatic presuppositions. In Stalnaker, R., editor, *Context and Content: Essays on Intentionality in Speech and Thought*, pages 47–62. Oxford University Press UK.

Tomioka, S. (2021). Scalar Implicature, Hurford's Constraint, Contrastiveness and How They All Come Together. *Frontiers in Communication*, 5.

van Kuppevelt, J. (1995a). Discourse structure, topicality and questioning. *Journal of Linguistics*, 31(1):109–147.

van Kuppevelt, J. (1995b). Main structure and side structure in discourse. *Linguistics*, 33(4):809–833.

von Fintel, K. and Heim, I. (2023). *Intensional Semantics*.

Wason, P. C. (1968). Reasoning about a rule. *Quarterly Journal of Experimental Psychology*, 20(3):273–281.

Zhang, Y. (2022). *New Perspectives on Inquisitive Semantics*. PhD thesis, University of Maryland.

Zhang, Y. (2024). A questions-under-discussion-based account of redundancy. *Unpublished manuscript*.