

This is an EARLY ACCESS version of

Lasersohn, Peter. 2026. Coordination of unmatched clause types and dynamic look-ahead. *Semantics and Pragmatics* 19(11). <https://doi.org/10.3765/sp.19.11>.

This version will be replaced with the final typeset version in due course. Note that page numbers will change, so cite with caution.

Coordination of unmatched clause types and dynamic look-ahead

Peter Lasersohn
University of Illinois

Declarative, interrogative, imperative, and exclamative sentences can all be coordinated with each other, even though they are often analyzed as belonging to different logical types. All four clause types can also appear as the consequent of a conditional. These facts are expected in a dynamic semantic theory in which each clause type is associated with an update operation to features of the discourse context, including a common ground, a questions-under-discussion list, a set of to-do lists, and a set of “affective trigger” registries. An adequate treatment of disjunction and conditionals in such a framework requires a representation of each context in terms of its own possible subsequent developments. Coordination and conditionals involving unmatched clause types are much more restricted in subordinate clauses than in main clauses, suggesting that dynamic effects of the kind posited here are limited to main clause contexts.

1. Introduction

Semantic analyses of conjunction and disjunction standardly require that any expressions which are conjoined or disjoined with one another must be of the same logical type. In the most basic case, the conjuncts or disjuncts are all of type t , but they might also all be of any of various other types, provided they are all of the *same* type. This seems reasonable, because conjunctions and disjunctions of mismatched types are typically ungrammatical, and uninterpretable using the ordinary apparatus of truth functions and type-shifting operations.

- (1) a. $[[_t \text{John is happy}] \text{ and } [_t \text{Mary is angry}]]$.
 b. $\text{John } [[_{\langle e, t \rangle} \text{loves Mary}] \text{ and } [_{\langle e, t \rangle} \text{hates Bill}]]$.
 c. $\text{John } [[_{\langle e, \langle e, t \rangle \rangle} \text{loves}] \text{ or } [_{\langle e, \langle e, t \rangle \rangle} \text{hates}]] \text{ Mary}$.
 (2) a. $*[[_{\langle e, t \rangle} \text{loves Mary}] \text{ and } [_t \text{Mary is angry}]]$.
 b. $*\text{John } [[_{\langle e, t \rangle} \text{loves Mary}] \text{ or } [_{\langle e, \langle e, t \rangle \rangle} \text{hates}]]$.
 c. $*[[_t \text{John loves Mary}] \text{ and } [_{\langle e, t \rangle} \text{hates Bill}]]$.

However, there is a major class of apparent exceptions: Declarative, interrogative, imperative, and exclamative clauses can all be conjoined or disjoined with one another, even though they are usually analyzed as belonging to different logical types:

- (3) a. *John left town, and after all, why wouldn't he?* (declarative-and-interrogative)
 b. *John is crazy, or is he just clever?* (declarative-or-interrogative)
 c. *Wash the dishes and I will dry.* (imperative-and-declarative)
 d. *Leave now or I will shoot!* (imperative-or-declarative)
 e. *John's dissertation defense is tomorrow, and is he ever nervous!* (declarative-and-exclamative)

- f. *Mary will be on time, or what a disaster we'll have on our hands!* (declarative-or-exclamative)
- g. *Stay in one place, and where is your little brother?* (imperative-and-interrogative)
- h. *Try to make sense, or are you incapable of that?* (imperative-or-interrogative)
- i. *How thoughtful he is, and isn't he handsome?* (exclamative-and-interrogative)
- j. *What a fool I've been, or am I just human?* (exclamative-or-interrogative)
- k. *Look in this box, and what a nice surprise you will find!* (imperative-and-exclamative)
- l. *Be alert, or what terrible consequences you will suffer!* (imperative-or-exclamative)

Declarative clauses are standardly held to be of type t . In the approach of Groenendijk and Stokhof 1982, which we will adopt here, interrogative clauses are analyzed as being of type $\langle s, t \rangle$.¹ A popular and intuitive idea about imperative clauses, advocated for example by Hausser 1980, Portner 2004, is to treat them as expressing properties,² hence of type $\langle e, t \rangle$ or something similar. A prominent approach to exclamative clauses advocated by Rett 2011 would treat them as predicates of degrees, hence as type $\langle d, t \rangle$ or something similar, where d is the type of degrees.³ Nonetheless, all these clause types⁴ can be conjoined and disjoined with one another, as the examples in (3) show.

A related observation may be made with respect to conditional sentences. In these, the antecedent clause must be declarative, but the consequent clause may be of any clause type:

- (4)
- a. *If you are ready, we can proceed.* (if-declarative-declarative)
 - b. *If John is so smart, then why can't he answer the question?* (if-declarative-interrogative)
 - c. *If he tries to escape, kill him!* (if-declarative-imperative)
 - d. *If John were here, what a good time we could have!* (if-declarative-exclamative)

How can we analyze *and*, *or*, and *if* in a way which makes sense of all these constructions? I will argue that an adequate analysis requires a dynamic semantic theory, in which the meanings of sentences are explained in terms of their effects on the contexts in which they are used. Beyond this, I will argue that the semantics of these connectives,

¹ Their intensions are therefore functions in type $\langle s, \langle s, t \rangle \rangle$. The intension of an interrogative clause maps each possible world onto a proposition which is true in that world. Moreover, these propositions form a partition on the set of worlds, so that in each world, exactly one of them is true. Each cell in the partition represents a complete answer to the question which the interrogative clauses expresses. For example, *Was John at the party?* has as its intension the function $\lambda w \lambda w' [\text{John was at the party in } w \leftrightarrow \text{John was at the party in } w']$; *Who was at the party?* has as its intension the function $\lambda w \lambda w' [\lambda x [x \text{ was at the party in } w] = \lambda x [x \text{ was at the party in } w']]$.

² See Kaufmann 2012 for a prominent analysis treating imperatives as type-theoretically more like declaratives.

³ See Zanuttini and Portner 2003, Villalba 2008, Rett 2011, and the references cited in these works for a variety of proposals regarding the logical type of exclamatives.

⁴ I use the term *clause type* for the classification of a clause as declarative, interrogative, imperative, or exclamative, and *logical type* for the classification of any expression (or its semantic values) as e , t , $\langle e, t \rangle$, etc. Unfortunately, the term *type* is too well established to permit an alternative in either system of classification.

particularly *or* and *if*, requires a formulation which does not simply constrain the transition from one context to the next, but which “looks ahead” to constrain later transitions.

The result will be that examples involving *or*, such as (3b, d, f, h, j, l) place the discourse temporarily into a “limbo” state, in some sense indeterminate between the result of updating with one disjunct and the result of updating with the other. This indeterminacy may be resolved at a later point in the discourse. Conditionals such as those in (4) place the discourse in a similarly indeterminate state, between the result of updating with the information that the antecedent is false, and the result of updating with the information that it is true, followed by updating with the consequent clause.

A roughly similar framework in which connectives constrain later alterations to the context was developed for independent reasons by Cohen and Krifka 2014, Krifka 2015.⁵ The present paper may be seen as providing additional support for this general strategy. However, we will depart from Cohen and Krifka’s analyses in several ways, perhaps most significantly in the treatment of disjunction. A thoroughgoing comparison of the two frameworks is beyond the scope of this paper, but I will present some reasons for using a disjunction operation which is not permitted in Cohen and Krifka’s system.

2. Scope of the analysis

Mixed clause-type conjunctions, disjunctions, and conditionals have been analyzed before; but to my knowledge, all such analyses focus on a specific, limited range of combinations: For example, see Clark 1993, Kaufmann 2012, Skelac 2016, von Stechow and Iatridou 2017, Starr 2018, and Keshet and Medeiros 2019 for coordination of imperatives and declaratives; see Hulstijn 1997, Velissaritou 2000, Isaacs and Rawlins 2008, and Sano and Hara 2014 for conditional interrogatives; see Kaufmann 2012, Charlow 2014, and Condoravdi and Lauer 2017 for conditional imperatives.

None of these earlier analyses attempts to account for the broad generalization that coordination of main clauses is productive across the full range of combinations of clause types, and that the choice of clause type for the consequent of a conditional is similarly unlimited. In general, the tendency has been to approach mixed clause-type constructions individually as special cases, calling for special explanation.

It has sometimes been pointed out that adopting a dynamic semantic framework makes it possible to treat one or another of these special cases in a uniform manner with examples involving coordination or conditionalization of declaratives (Isaacs and Rawlins 2008, Starr 2018), but the need for a general theory of sentential connectives and a general theory of clause types which will interact to predict that coordination is effectively “blind” to clause type seems not to have been directly addressed, or even much remarked on.

In addition, many of these earlier analyses focus on special, exceptional readings of the constructions involved, especially the conditional-like “pseudo-imperative” reading of conjunctions of imperatives with declaratives. This reading is illustrated in examples like

⁵ A meaningful comparison might also be made to Swanson 2022.

(5), which is paraphrasable roughly as “If you take one more step, you will fall in.” On this reading, (5) does not serve as a command or request to take one more step, followed by an assertion that the addressee will fall in:

(5) *Take one more step and you’ll fall in.*

This contrasts examples like (6), which has a prominent reading on which it does serve as a command or request to dress nicely, followed by an assertion that the speaker and others will pick up the addressee at 8:00:

(6) *Dress nicely, and we’ll pick you up at 8:00.*

Pseudo-imperative readings form their own complex topic, and are best considered together with other examples where conjunction receives something like a conditional interpretation, such as (7):

(7) *John starts to sing, and we’ll all have to cover our ears.*

Such an interpretation is by no means peculiar to examples where the conjuncts are of unmatched clause types; see Keshet 2013 for more examples and discussion.

Because a conditional interpretation for conjunction is neither an automatic concomitant of the imperative-*and*-declarative construction, nor limited to it, let us set it and any similarly exceptional readings aside, in order to focus on readings where the constituent clauses seem to retain their usual force, and the connectives their usual meanings. At least in the case of conjunction and disjunction, these “ordinary” readings seem to have received less attention in the literature than the exceptional ones, even though traditional analyses of the connectives do not account for them.

As an anonymous referee pointed out, several of the examples in (3) become unacceptable if the order of the conjuncts or disjuncts is reversed. The referee suggests more specifically that in mixed clause-type coordination, interrogatives and exclamatives must always come after other clause types.

However, this is simply incorrect. Here are some naturally occurring counterexamples:

- (8) a. *What does PAF mean and DON’t tell me to go to FAQ!*⁶
 b. *What a difference a day makes, and the difference is you.*⁷
 c. *How about giving it to me, eh? Or maybe I’ll just take it.*⁸

⁶ From an online geocaching forum, <https://forums.geocaching.com/GC/index.php?/topic/208413-what-does-paf-mean-and-dont-tell-me-to-go-to-faq/>, retrieved 8/10/2025.

⁷ From the popular song ‘What a Difference a Day Makes’, English lyrics by Stanley Adams (1934).

⁸ From *The Trials of Brother Bell : Two Novels, Repentance Vale and The Struggle Outside*, p. 44, by Raymond Fraser (2010).

- d. *It's obvious that those two were added to get 1,341', but what are they, and I don't see how you can come up with a total that's more than the entire parcel's street frontage.*⁹
- e. *Is he ever and he seems taken with Daisy.*¹⁰
- f. *What a scary thing it is to put your creative project out there into the world to be criticised, but come compadres and criticise, I would love to hear what you have to say.*¹¹
- g. *What a lucky guy John Krasinski is. Or maybe George Clooney, Emily Blunt, Stanley Tucci, Ryan Reynolds, Steve Carell and Bradley Cooper are the lucky ones getting to hug John.*¹²

The prevalence of such examples is somewhat obscured by English orthographic conventions, which call for a question mark at the end of an interrogative clause (or series of coordinated interrogative clauses), but also prohibit question marks from appearing sentence-medially, with similar principles governing exclamation points and exclamatives. The result is that when an interrogative or exclamative appears as the first conjunct or disjunct, the whole coordinate structure is often written as two sentences, as in (8c). But there seems little reason not to consider such examples as single sentences theoretically, and in informal contexts, where the pressures of standard orthography are weaker, we find that they are often written as single sentences, as in (8a, d).¹³

It is true that in many examples, the order of the conjuncts and disjuncts cannot be reversed. Typically, however, we find that the reverse order is permitted with some adjustment of anaphoric or presuppositional items, placement of *just*¹⁴ or of uncertainty markers such as *maybe*, or with wording changes to avoid implausible implicatures regarding temporal ordering or rhetorical strategy. The following examples parallel the those in (3), but with the clause types in the opposite order:

- (9) a. *Why wouldn't John leave town? And after all, maybe he did.*
 b. *Is John so clever? Or maybe he is just crazy.*
 c. *I will wash the dishes and you dry.*¹⁵

⁹ From 'Emails Between BID Consultant Tara Devine and the Los Angeles City Clerk's Office', <https://archive.org/details/EmailsBetweenTaraDevineAndLACityClerk/010114BM/page/n1/mode/2up>, retrieved 9/9/2025.

¹⁰ From *Worlds Beyond Tomorrow, Part III*, Ch. 2 (unpaginated), by Ellen Elizabeth Dudley (2021).

¹¹ From *Structural Pandemonium*, Ch. 5 (unpaginated) by Victoria Fatiregun (2020).

¹² From an Instagram reel: <https://www.instagram.com/reels/DFGNiKXNt8M/>, retrieved 4/28/2026.

¹³ Even if we analyze examples like (8c) as two independent sentences, we still must account for the meaning of the connective, and how it relates what comes before it to the sentence immediately following. Many of the same theoretical issues will inevitably arise as in an analysis that treats such examples as compound sentences, so making this move would do little to solve the general problems addressed in this paper.

¹⁴ For an analysis of *just* which, like the analysis of unmatched clause-type coordination offered here, is stated in terms of constraints on the future development of discourse, see Warstadt 2020.

¹⁵ The intended reading here is one where *you dry* is imperative with an overt subject, not one where it is declarative with *will wash* deleted by gapping.

- d. *I will shoot you! Or, just leave now!*
- e. *Is John ever nervous! And his dissertation defense is tomorrow!*
- f. *What a disaster we'll have on our hands! Or just maybe, Mary will show up on time.*
- g. *Where is your little brother? And stay in one place!*
- h. *Are you incapable of making sense? Or at least give it a try!*
- i. *Isn't he handsome? And how thoughtful he is!*
- j. *Who will take responsibility to clean up this mess? Or what a lot of explaining we'll have to do!*
- k. *What a nice surprise you will find in this box, and be sure to look at it carefully!*
- l. *What terrible consequences you will suffer! Or, just keep alert.*

The examples in (3) and (9) together show that all possible orderings of all possible combinations of clause types are acceptable. This suggests that the irreversibility of conjuncts and disjuncts in certain examples does not have anything specifically to do with the interaction between the semantics of the connectives and the semantics of clause types, but rather with independent factors having to do with anaphora, presupposition, implicature, and similar phenomena. I will not attempt to account for these here, in order to concentrate on our central topic: how to reconcile the apparent differences in logical type among the various clause types with the promiscuity with which they coordinate with one another.

3. Dynamic conjunction and unmatched clause types

A “dynamic” analysis of the connectives offers a promising approach to the phenomenon of mixed clause-type coordination and conditionals. The viability of this approach is most easily apparent in the case of conjunction, so in this section we briefly review the treatment of major clause types in dynamic semantics, then the analysis of conjunction, to show that dynamic conjunction can give intuitively appropriate results when linking clauses of different logical types. In Section 4, we will see some problems which this initial version of dynamic semantics faces in dealing with disjunction, and make some revisions in response to these problems. We will then return to the analysis of disjunction in Section 5.

3.1. Clause types in dynamic semantics

A variety of different dynamic semantic theories have been proposed, making different claims about the interpretations of the various clause types. Rather than reviewing all of our options in a dynamic framework, here we outline a simple “generic” version of dynamic semantics, drawing primarily on Heim 1982, Roberts 2012, and Portner 2004 without following any of them in detail.

We assume that an important part of any context of utterance is the *common ground* (CG) of the discourse participants, in roughly the sense of Stalnaker 1978 — the set of propositions which these participants collectively treat as shared background knowledge. Accordingly, we may represent each context c as an ordered n -tuple $\langle CG_c, \dots \rangle$, where CG_c is the common ground of c and the three dots represent other features to be discussed below.

Asserting a proposition involves adding it to the CG,¹⁶ so the result of updating a context by the assertion of p may tentatively be defined as in (10), where the three dots are filled in the same way on both sides of the equals-sign:

$$(10) \langle CG_c, \dots \rangle + p = \langle CG_c \cup \{p\}, \dots \rangle$$

In a static semantic theory, we might set up our compositional rules to assign a proposition to each declarative sentence; we could then say that the effect of uttering a sentence φ whose content is proposition p in context c is to update c to $c + p$. In a dynamic semantic theory, we would instead set up our compositional semantic rules to directly assign a context change potential — a function from contexts to contexts — to each declarative sentence. Specifically, if a sentence φ was assigned proposition p in the static semantics, we would set up the dynamic theory to assign φ the function which mapped any context c onto $c + p$:

$$(11) \text{ If } \llbracket \varphi \rrbracket_{\text{static}} = p, \text{ then } \llbracket \varphi \rrbracket_{\text{dynamic}} = \lambda c[c + p]$$

Then the context which results from using a declarative sentence φ in context c may be obtained by simple function application. Numbering the contexts for successive utterances in a discourse by increments of 1:

$$(12) \text{ If } \varphi \text{ is uttered in } c_n, \text{ then } c_{n+1} = \llbracket \varphi \rrbracket_{\text{dynamic}}(c_n).$$

If a proposition is added to the CG which contradicts information already in the CG, the resulting context is defective and must be repaired. We will not formulate repair operations here, but it will be important to bear in mind that “normal,” non-repair-provoking assertions may not contradict the CG.

Following Roberts 2012, let us also assume that any context of utterance will include a set of *questions under discussion* (QUD), which plays a role in the act of asking a question which is somewhat analogous to the role played by the CG in the act of asserting a proposition.¹⁷ We may now represent a context c as $\langle CG_c, QUD_c, \dots \rangle$.¹⁸ Asking a question Q in c adds it to QUD_c , so updating a context by asking Q may be defined as in (13):

$$(13) \langle CG_c, QUD_c, \dots \rangle + Q = \langle CG_c, QUD_c \cup \{Q\}, \dots \rangle$$

¹⁶ It will not remain in the CG if other discourse participants object. Some semanticists might therefore regard it as preferable to analyze assertion merely as a *proposal* to add a proposition to the CG, rather than as a direct addition. For our current purposes, the choice between these conceptions of assertion will not matter much.

¹⁷ Roberts 2012 obtains important results by structuring the QUD as a pushdown stack, with the most specific questions at the top of the stack. We omit imposing a stack structure here purely in the interest of keeping the presentation simple and easy to follow; a more detailed development should probably retain something closer to Roberts’ original approach.

¹⁸ Arguably, which questions are under discussion is given as part of the common ground; so in principle, the QUD need not be represented as a separate element from the CG. However, notating the QUD and CG separately allows for a more perspicuous presentation, so I will treat them as separate elements here. Similar comments apply to other contextual parameters such as the TDL and ATR, to be discussed below.

In a static semantic theory, we might set up our compositional rules to assign a question to each interrogative sentence; we could then say that the effect of uttering a sentence φ whose content is question Q in context c is to update c to $c + Q$. In a dynamic semantic theory, we would instead set up our compositional semantic rules to assign a context change potential to each interrogative sentence, just as we did with declaratives. If a sentence φ was assigned question Q in the static semantics, the dynamic theory would assign φ the function which maps any context c onto $c + Q$:

$$(14) \text{ If } \llbracket \varphi \rrbracket_{\text{static}} = Q, \text{ then } \llbracket \varphi \rrbracket_{\text{dynamic}} = \lambda c[c + Q]$$

As a question in the QUD is answered, it is removed. To assure this, we should revise our rule for updating a context with a proposition. If adding a proposition to the CG results in a new CG that entails an answer to any of the questions in the QUD, that question is removed:¹⁹

$$(15) \langle CG_c, QUD_c, \dots \rangle + p = \langle CG_c \cup \{p\}, QUD_c - \{Q \in QUD_c \mid \exists w CG_c \cup \{p\} \models Q(w)\}, \dots \rangle$$

Corresponding to the CG for assertions and the QUD for questions, each context will include a function TDL assigning a *to-do list* (TDL) to each participant in the discourse, as suggested in Portner 2004. Each context should also specify an *addressee*, *speaker*, and a *time*. We may now represent a context c as $\langle CG_c, QUD_c, TDL_c, ADDR_c, SPEAKER_c, TIME_c, \dots \rangle$, where TDL_c is a function assigning a set of properties $TDL_c(x)$ to each discourse participant x , $ADDR_c$ and $SPEAKER_c$ are discourse participants, and $TIME_c$ is a time. Issuing an order, request or suggestion to have some property involves adding that property to the addressee's TDL. To formulate this operation, first let us define some notation.

$$(16) \text{ For every discourse participant } x \text{ and property } P, \text{ let } TDL_c[x + P] \text{ be that function } F \text{ such that: } F(x) = TDL_c(x) \cup \{P\} \text{ and for every participant } y \neq x, F(y) = TDL_c(y).$$

$TDL_c[x + P]$ is the function just like TDL_c except that it includes P in x 's to-do-list. Now we may define updating a context by ordering P as in (17):

$$(17) \langle CG_c, QUD_c, TDL_c, ADDR_c, \dots \rangle + P = \langle CG_c, QUD_c, TDL_c[ADDR_c + P], ADDR_c, \dots \rangle$$

In a static semantic theory, we might set up our semantic rules to assign a property to each imperative sentence; we could say that uttering a sentence whose content is property P in context c is to update c to $c + P$. In a dynamic semantic theory, we would treat this function instead as the semantic value of the sentence itself:

$$(18) \text{ If } \llbracket \varphi \rrbracket_{\text{static}} = P, \text{ then } \llbracket \varphi \rrbracket_{\text{dynamic}} = \lambda c[c + P]$$

Just as a question is removed from the QUD once an answer to it is in the CG, a property is removed from a participant's to-do list once it is in the CG that that participant

¹⁹ If p is a proposition and Γ is a set of propositions, $\Gamma \models p$ iff $\forall w[\forall q[q \in \Gamma \rightarrow q(w) = 1] \rightarrow p(w) = 1]$.

has the property. This requires another revision to our rule for updating a context with a proposition, for which some more notation will be useful:

- (19) For every discourse participant x and set of properties \mathcal{P} , let $TDL_c[x - \mathcal{P}]$ be that function F such that $F(x) = TDL_c(x) - \mathcal{P}$, and for all $y \neq x$, $F(y) = TDL_c(y)$.

$TDL_c[x - \mathcal{P}]$ is the function just like TDL_c except that it omits all the properties in \mathcal{P} from x 's to-do-list. Now:

- (20) $\langle CG_c, QUD_c, TDL_c, ADDR_c, \dots \rangle + p = \langle CG_c \cup \{p\}, QUD_c - \{Q \in QUD_c \mid \exists w CG_c \cup \{p\} \models Q(w)\}, TDL_c[ADDR_c - \{P \in TDL_c(ADDR_c) \mid CG_c \cup \{p\} \models \lambda w [P(w)(x)]\}], \dots \rangle$

What if it becomes clear that a discourse participant does not and will not have a property which is on their TDL? As in cases where it becomes clear that a proposition in the CG is false, this provokes repair. Typically the repair will involve removing the property from the participant's TDL, and may also involve adding propositions to the effect that this participant will incur social consequences for failing to meet an obligation, depending on the specifics of the circumstances.

Exclamative sentences have not received the attention in dynamic semantics that declaratives, interrogatives, or imperatives have, but since in all these other cases, the context is treated as including a set of (what in a static theory would be) possible sentence contents — the CG, QUD, and TDLs — and the utterance of a sentence is treated as adding (what in a static theory would be) its content to the appropriate set, it seems appropriate to suggest that each context should also include a set of (what in a static theory would be) possible exclamative contents, and that using an exclamative sentence adds (what in a static theory would be) its content to this set.

In the static analysis of Rett 2011, exclamatives are predicates of degrees; if we adopt this general approach, they will be (extensionally) of type $\langle d, t \rangle$, with contents/intensions in type $\langle s, \langle d, t \rangle \rangle$. For example, (21a) will have the content shown in (22a), and (21b) will have the content shown in (22b):

- (21) a. What a snob John is!
 b. Is John ever nosy!
- (22) a. $\lambda w \lambda d [\text{John is a snob to degree } d \text{ in } w]$
 b. $\lambda w \lambda d [\text{John is nosy to degree } d \text{ in } w]$

We should expect, therefore, that in a dynamic semantics, contexts should include a set of functions in type $\langle s, \langle d, t \rangle \rangle$ — a set of degree properties — and that when a speaker uses an exclamative sentence, its content is added to this set.

How does this relate to the kinds of speech act one can perform directly by uttering an exclamative sentence? Rett's account of the speech act of exclaiming does not make use of such a set, and therefore does not give us an explicit answer. She suggests that exclaiming is an expression of surprise, and that using an exclamative sentence is more specifically an expression "that $\exists d'$ such that s_c had not expected that $D(d')$ ", where s_c is the

speaker and D is the content of the exclamative sentence. For example, an utterance of (21a) would be an expression that there is some degree d' such that the speaker had not expected John to be a snob to d' . But given all this, why would discourse participants collectively track a set of degree properties, and why would such an expression of surprise add a new property to the set?

I suspect the answer is simply that it is socially advantageous to maintain sensitivity to our interlocutors' affective states. Exclamatory utterances are the expression of sudden changes in affective state — typically (though perhaps not exclusively²⁰) as a result of surprise. Being a good conversational partner requires not only an awareness of these sudden changes, but an ongoing knowledge of what kinds of stimuli can trigger them. I suggest, therefore, that conversational participants collectively maintain an “affective trigger registry,” or ATR, for each interlocutor, just as they maintain a TDL, QUD, and CG.

More formally, we may now represent each context c as a tuple $\langle CG_c, QUD_c, TDL_c, ATR_c, ADDR_c, SPEAKER_c, TIME_c, \dots \rangle$, where ATR_c is a function assigning a set to each discourse participant. Intuitively, we may think of $ATR_c(x)$ as the set of things which the discourse participants collectively recognize as affective triggers for x — things which produce notable changes in x 's emotion or mood, when x becomes aware of them. Degree properties — the contents of exclamative sentences, in a static analysis — can be members of $ATR_c(x)$, for a given x . For example, if Mary says “Is John ever tall!”, we may add the property of being a degree to which John is tall to $ATR_c(\text{Mary})$. By the same token, if Mary is silent, but we observe by her facial expression that she is shocked at how tall John is, we may add this same property.

For the purposes of interpreting exclamative sentences, we may regard a participant's ATR simply as a set of degree properties. However, I see no reason to limit the set to members in this type. Someone may have a strong reaction to objects in practically any type; it would be useful to allow propositions, entities, etc. as members, in order to account for exclamatory utterances of expressions other than exclamative sentences, such as the examples in (23):

- (23) a. Wow! John is a movie star now!
 b. (Gasp) Freddie Krueger!

Ultimately, it might be necessary to represent contexts in a way which tracks not just what things are affective triggers for participants, but what specific kinds of affect each one triggers — shock, fear, admiration, delight, etc. But a simpler representation which simply keeps a registry of the triggers will serve us well enough for the purpose of illustration in this paper.

Adapting our notation for TDLs to ATRs, and using D as a variable over degree properties, we may now define update with a degree property as in (24a), and define dynamic contents for exclamative sentences in terms of static contents as in (24b):

²⁰ See the discussions in Zanuttini and Portner 2003 and Rett 2011 of examples like *What a nice house you've got!*

- (24) a. $\langle CG_c, QUD_c, TDL_c, ATR_c, ADDR_c, SPEAKER_c, \dots \rangle + D =$
 $\langle CG_c, QUD_c, TDL_c, ATR_c[SPEAKER_c + D], ADDR_c, SPEAKER_c, \dots \rangle$
 b. If $\llbracket \varphi \rrbracket_{static} = D$, then $\llbracket \varphi \rrbracket_{dynamic} = \lambda c[c + D]$

3.2. Dynamic conjunction of unmatched clause types

If a sentence has as its semantic value a function from contexts to contexts as sketched for the dynamic options in Section 3.1, we may follow Heim 1982, and interpret conjunction of sentences as simple function composition:²¹

$$(25) \llbracket and \rrbracket = \lambda p[\lambda q[\lambda c[p(q(c))]]]$$

This gives the kind of result illustrated in (26):

$$(26) \llbracket John\ arrived\ and\ Mary\ left \rrbracket = \llbracket and \rrbracket(\llbracket Mary\ left \rrbracket)(\llbracket John\ arrived \rrbracket) \\ = \lambda c[\llbracket Mary\ left \rrbracket(\llbracket John\ arrived \rrbracket)(c)]$$

Suppose that $\llbracket John\ arrived \rrbracket(c) = c'$, and that $\llbracket Mary\ left \rrbracket(c') = c''$. Then $\llbracket John\ arrived\ and\ Mary\ left \rrbracket(c) = c''$. Given the analysis sketched in Section 3.1, c' differs from c in the addition of the proposition that John arrived to the CG, and c'' differs from c' in the addition of the proposition that Mary left to the CG.

This interpretation for conjunction affords intuitive results for conjunctions of unmatched clause types. An utterance of *John peered into the box, and what do you think he saw?* in a context c is expected to result in a new context c' which differs from c in the addition of the proposition that John peered into the box to the CG, followed by the addition of the question of what the addressee thinks John saw to the QUD. An utterance of *Wash the dishes, and I will dry* in c is expected to result in a context c' differing from c in the addition of the property of washing the dishes to the addressee's TDL, followed by the addition of the proposition that the speaker will dry the dishes to the CG. An utterance of *Stay in one place, and where is your little brother?* in c is expected to result in a context c' differing from c in the addition of the property of staying in one place to the addressee's TDL, followed by the addition of the question of where the addressee's little brother is to the QUD.

4. Look-ahead restrictions

Unfortunately, the analysis just outlined does not extend straightforwardly to examples involving disjunction rather than conjunction. We would not want to adopt a rule that said something like "To update a context c with $(\varphi\ or\ \psi)$, simply update it with φ or update it with ψ ," for example. This would leave it indeterminate what the output context will be — it will be either the result of updating with φ or the result of updating with ψ , but nothing tells us which.

Perhaps we could maintain such a rule by claiming that it is up to the addressee(s) to choose which way to update the context. This would give reasonably intuitive results for certain examples: *Leave now or I will shoot!* would give the addressee the choice of putting

²¹ We assume that *and* forms a syntactic unit with its right-hand conjunct. In (25), p and q are variables ranging over functions from contexts to contexts.

the *leave now* property on their TDL or adding the proposition that the speaker will shoot to the CG; *John is crazy, or is he just clever?* would give the addressee the option of adding the proposition that John is crazy to the CG, or adding the question whether John is clever to the QUD. But especially in examples where the disjuncts are both declarative, the readings assigned are clearly wrong. A speaker who uttered *John is in the house, or Mary is in the yard* would not be leaving it to the discretion of the addressee to choose between the proposition that John is in the house and the proposition that Mary is in the yard, and add the chosen proposition to the CG.

We can obtain better results by reformulating our update rules so that they don't simply add items to the CG, QUD, TDLs, or ATRs, but impose constraints on what can be added later in the discourse. For example, adding a proposition of the form $[p \vee q]$ to the CG constrains it so that you cannot later add anything that entails $[\neg p \wedge \neg q]$ (without resulting in contradiction and provoking repair). Additions to the QUD, TDLs, or ATRs will also constrain what can be added later.

There are various ways we might try to formalize this idea, but let's concentrate on one in particular. Suppose that as a discourse progresses, participants keep track not just of the current CG, QUD, TDLs, ATRs, etc., but of a set of possible future developments of these. Each future course of development for these sets may be represented as a sequence of tuples, where each tuple is of the kind we have been using to represent a pragmatic context — of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME, \dots \rangle$. But since the idea now is that discourse participants keep track of a set of sequences of such tuples, we may regard this set of sequences as the context, rather than any one tuple.

Then it would make sense to define update operations which map sets of such sequences onto sets of such sequences, rather than (just) operations mapping tuples to tuples. The idea would be that each update leads the discourse participants to revise their collective expectations of what might happen later in the discourse.

On this view, which sequences should be in the set representing a context? Each one should represent a possible course of future development of the CG, QUD, TDLs, ATRs, etc.; but all kinds of changes to these may occur during a conversation, and perhaps we should leave out sequences that include certain kinds of transition from one tuple to another. Some transitions represent corrections, repairs, backtracking, or other moves which exemplify “releases” from the constraints imposed by earlier moves, rather than conforming to them. Let's not include such sequences, but only those which result from successive ordinary, non-repair utterances, and/or similar non-repair alterations brought about by non-linguistic means, such as new information becoming visually salient in the context.²² If some tuple in a sequence is defective in a way which would provoke repair — for example if it contains a contradictory CG, or a CG including the information that one of the discourse participants cannot acquire one of the properties in their TDL, or a CG

²² We may think of these non-linguistic alterations as involving the same basic operations as linguistic utterances, such as those in (13), (17), (20), and (24a) — though these will be revised shortly.

including information which renders a member of the QUD unanswerable — then let us suppose the sequence terminates at that point, with no subsequent members.

Just as we omit from our representation of contexts those sequences which involve “release” from prior constraints, we omit those sequences in which the discourse participants do not draw appropriate inferences or perform licensed updates. If a proposition is added to the CG, anything entailed by the combination of that proposition with the existing CG should be added too. In practice, of course, discourse participants may fail to recognize an entailment as such, or may not bother to calculate complex or pragmatically irrelevant entailments; and in such cases a proposition might not be added to the CG even though it is entailed by what is already there. But it would be a perverse group of conversational partners who publicly recognized a proposition as following from their own background assumptions, yet failed to treat that proposition as likewise assumed; so it seems clear that the rules by which we guide our construction of discourse contexts are ones which call for the CG to be closed under entailment, even if our execution of those rules is often lax and imperfect. The sequences in our representation of context are ones which conform to these rules, regardless of whether the subsequent discourse is as consistent in its conformity.

Similarly, we may regard a question as “entailing” various more specific sub-questions; and if a question is under discussion, its entailed sub-questions are in principle under discussion as well. For example, if the pragmatically relevant individuals in context are John, Mary and Bill, and we are considering the question of who was at the party, this requires consideration of the more specific question of whether John was at the party. Note here that the true complete answer to *Who was at the party?* will in any world entail the true answer to *Was John at the party?* More generally, we can say that (where a question is a function in type $\langle s, \langle s, t \rangle \rangle$), one question Q entails another question Q' iff for every w , $Q(w)$ entails $Q'(w)$ — that is, if for all w, w' , if $Q(w)(w') = 1$ then $Q'(w)(w') = 1$.²³

In some cases, it may be that one question Q does not by itself entail another question Q' , but that Q in combination with the CG does entail Q' , — that is, the true complete answer to Q , in combination with the CG, in any world entails the true complete answer to Q' . Such contextual entailments²⁴ should also be added to the QUD when the entailing question is. For example, if it is given in the CG that John is the only student, then adding the question whether a student was at the party to the QUD should result in the question whether John was at the party also being added.

²³ The characterization of entailment for questions here follows Groenendijk and Stokhof 1984. Note that in some cases, one question may be intuitively “more specific” than another without being entailed by it. For example, (i) is intuitively “more specific” than (ii), but (ii) does not entail (i), because in those worlds where the true answer to (ii) is “yes,” this does not entail an answer to (i).

(i) Are there any dachshunds in the kennel?

(ii) Are there any dogs in the kennel?

²⁴ The term *contextual entailment*, as applied to questions, and the general strategy for defining it, are taken from Roberts 2012.

As with the addition of propositions to the CG, discourse participants may in practice add a question to the QUD without adding all its contextual entailments, simply because they fail to recognize them as such, or don't bother to calculate the more complex entailments; but again, it seems even so that the rules by which we guide our construction of discourse contexts are ones which call for the addition of entailments of questions added to the QUD. Therefore, in our representation of contexts, the sequences will be ones in which the QUD at each stage is closed under entailment.

Likewise, if a property is added to a discourse participant's TDL, we may consider any more general property as in principle added too. If John has been requested to cook dinner, he has been requested to cook a meal, so the property of cooking a meal should be on his TDL no less than the property of cooking dinner. This does not mean that cooking lunch would discharge his duties, of course. More generally, we may say that one property P entails another property P' iff for all w, x , if $P(w)(x) = 1$, then $P'(w)(x) = 1$. Our representation of contexts should include only those sequences where, at any point in the sequence, if some property P is in the TDL of some individual x , and P contextually entails P' , then P' is also in x 's TDL.

This pattern does not appear to hold for the degree properties in a participant's ATR, however. We may say that one degree property D entails another D' iff for all w, d , if $D(w)(d) = 1$, then $D'(w)(d) = 1$. But we should allow $\lambda w \lambda d [\text{John is tall to } d \text{ in } w]$ to be in Mary's ATR, for example, without $\lambda w \lambda d \exists x [x \text{ is tall to } d \text{ in } w]$ also being Mary's ATR; Mary might be shocked at how tall John is, without being shocked that there is someone that tall — for example if John is her grandson whom she has not seen in a long time.

To make these ideas more explicit, let us set up some notation. Where Γ is any set of propositions:²⁵

(27) If p is a proposition, $\Gamma \models p$ iff for all w : If $q(w) = 1$ for all $q \in \Gamma$, then $p(w) = 1$.

(28) a. If $A \subseteq \mathbf{D}_{\langle s, \langle s, t \rangle \rangle}$ and $Q \in \mathbf{D}_{\langle s, \langle s, t \rangle \rangle}$, then $A \models_{\Gamma} Q$ iff for all worlds w :
 $\Gamma \cup \{Q'(w) \mid Q' \in A\} \models Q(w)$.

b. If $A \subseteq \mathbf{D}_{\langle s, \langle e, t \rangle \rangle}$ and $P \in \mathbf{D}_{\langle s, \langle e, t \rangle \rangle}$, then $A \models_{\Gamma} P$ iff for all individuals x :
 $\Gamma \cup \{\lambda w [P'(w)(x)] \mid P' \in A\} \models \lambda w [P(w)(x)]$.

We may read ' $A \models_{\Gamma} x$ ' as "A entails x in combination with Γ ."²⁶

We should revise our update operations so that they don't just add a single proposition, question, or property to the CG, QUD, or a participant's TDL, but also then close all these under contextual entailment. The closure operation may be defined as in (29):

²⁵ The definition in (28a) follows the definition of "contextual entailment" for questions in Roberts 2012, with some minor adjustments.

²⁶ When the members of A and/or Γ are listed out, we omit the curly braces, for example writing " $Q_1, Q_2 \models_p Q_3$ " rather than " $\{Q_1, Q_2\} \models_{\{p\}} Q_3$ ".

(29) $Close(\langle CG, QUD, TDL, \dots \rangle) = \langle CG \cup \{p \mid CG \models p\}, QUD \cup \{Q \mid QUD \models_{CG} Q\}, \lambda x[TDL(x) \cup \{P \mid TDL(x) \models_{CG} P\}], \dots \rangle$

Our update rules in (20), (13), and (17) may now be replaced with rules that apply the closure operation to the outputs of our old rules:

(30) a. $\langle CG_c, QUD_c, TDL_c, ADDR_c, \dots \rangle + p = Close(\langle CG_c \cup \{p\}, QUD_c - \{Q \in QUD_c \mid \exists q \in Q . \bigcap (CG_c \cup \{p\}) \subseteq q\}, TDL_c[ADDR_c - \{P \in TDL_c(ADDR_c) \mid \bigcap (CG_c \cup \{p\}) \subseteq [\lambda w . P(w)(x)]\}], \dots \rangle)$
 b. $\langle CG_c, QUD_c, \dots \rangle + Q = Close(\langle CG_c, QUD_c \cup \{Q\}, \dots \rangle)$
 c. $\langle CG_c, QUD_c, TDL_c, ADDR_c, \dots \rangle + P = Close(\langle CG_c, QUD_c, TDL_c[ADDR_c + P], ADDR_c, \dots \rangle)$

Now let us assume that each tuple in a sequence in a context must derive from the previous tuple by an application of one of the operations in (30a, b, c), (24a), and/or an alteration to the speaker or addressee, or advancement of the time. If a sequence meets this condition, we call it a *development*:²⁷

(31) A sequence σ of tuples of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME, \dots \rangle$ is a *development* iff for all natural numbers n such that $n, n+1 \in dom(\sigma)$, either:
 a. For some proposition p , $\sigma_{n+1} = \sigma_n + p$, or
 b. For some question Q , $\sigma_{n+1} = \sigma_n + Q$, or
 c. For some individual property P , $\sigma_{n+1} = \sigma_n + P$, or
 d. For some degree property D , $\sigma_{n+1} = \sigma_n + D$, or
 e. $CG_{\sigma_n} = CG_{\sigma_{n+1}}$ and $QUD_{\sigma_n} = QUD_{\sigma_{n+1}}$ and $TDL_{\sigma_n} = TDL_{\sigma_{n+1}}$, and $ATR_{\sigma_n} = ATR_{\sigma_{n+1}}$, and $TIME_{\sigma_n} \leq TIME_{\sigma_{n+1}}$.

In principle, each tuple should be derivable from a *minimal* tuple by successive application of these same operations, where a minimal tuple is one whose CG, QUD, TDLs, and ATRs are all trivial, which is to say that the CG contains nothing but the tautology $\lambda w[w = w]$, the QUD includes only the always-open $\lambda w \lambda w'[w = w']$, the TDL for each participant includes only the universal property $\lambda w \lambda x[x = x \text{ in } w]$, and the ATR for each participant is empty. A development from a minimal tuple may be called a “cold-start” development:

(32) $Minimal(\kappa)$ iff $CG_\kappa = \{\lambda w[w = w]\}$, $QUD_\kappa = \{\lambda w \lambda w'[w = w']\}$, for all x : $TDL_\kappa(x) = \{\lambda w \lambda x[x = x \text{ in } w]\}$, and for all x : $ATR_\kappa(x) = \emptyset$.

(33) A sequence σ of tuples of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME, \dots \rangle$ is a *cold-start development* iff σ is a development and $Minimal(\sigma_0)$.

Of course, in actual conversation we do not start with (or reach) a trivial CG, QUD, TDLs, or ATRs, so the sequences in an actual context will not be cold-start developments. However, they should “start in the middle” of cold-start developments:

²⁷ I assume a *sequence* is a function whose domain is a set S of natural numbers meeting the condition that (for all n, m) if $n \in S$ and $m < n$, then $m \in S$. If σ is a sequence and n is a natural number, we may write “ σ_n ” for the $n+1^{\text{th}}$ member of σ (that is, for $\sigma(n)$). We continue to use angle bracket notation, so ‘ $\langle a, b, c, \dots \rangle$ ’ represents a function which assigns 0 the value a , 1 the value b , 2 the value c , etc.

- (34) For all contexts c , for all $\sigma \in c$, there is a cold-start development δ and natural number i such that for all $n \in \text{dom}(\sigma)$: $\sigma_n = \delta_{n+i}$.

Finally, all the sequences in a context must align their times:

- (35) For all contexts c , for all $\sigma, \sigma' \in c$, for all n such that $n \in \text{dom}(\sigma)$ and $n \in \text{dom}(\sigma')$:
 $\text{TIME}_{\sigma_n} = \text{TIME}_{\sigma'_n}$.

Simple update operations at the level of this revised notion of context can be straightforwardly defined in terms of operations at the level of tuples (our old contexts). We simply filter out all those developments whose second member is not the result of updating the first member with the operation in question, then “behead” each of the remaining developments — that is, remove its first member:

- (36) For any sequence σ , $\text{TAIL}(\sigma)$ is that sequence such that for all natural numbers n such that $n+1$ is in the domain of σ : $\text{TAIL}(\sigma)_n = \sigma_{n+1}$.
- (37) If c is a context, and x is a proposition, question, or property:
 $c + x = \{\text{TAIL}(\sigma) \mid \sigma \in c \wedge \sigma_1 = \sigma_0 + x\}$

We continue to analyze sentences as denoting functions from contexts to contexts, though our new contexts are essentially sets of sequences of our old contexts. For simple sentences of the various clause types, these functions will relate to propositions, questions, and properties (the contents of sentences in a static semantics) in a very much analogous way to what we saw in (11), (14), and (18) above. Letting c be a variable ranging over contexts and letting x range over propositions, questions, and properties:

- (38) If $\llbracket \varphi \rrbracket_{\text{static}} = x$, then $\llbracket \varphi \rrbracket_{\text{dynamic}} = \lambda c [c + x]$

How should we understand the contexts in this kind of system? For example, what propositions do the discourse participants take as uncontroversial background, if the context is represented as a set of sequences, and each element of each sequence has its own common ground, rather than as a single tuple with a single common ground? What questions do they take to be under discussion? What properties do they take to be on each participant’s to-do list?

The fact that we expect answers to these questions means that for any of our “new” contexts, we should be able to construct a corresponding “old” context. (The reverse need not be the case — that is, we should not expect that for every “old” context, there will be exactly one corresponding “new” context, for reasons we shall see shortly.) Let us call this tuple the *reduct* of the context. The reduct of a context c should have as its CG just those propositions which are in the CG of the first tuple of every development in c . Its QUD should include just those questions which are in the QUD of the first tuple of every development in c . Its TDL function should assign each discourse participant x all those properties P such that the first tuple of every development in c includes P in x ’s TDL. Its ATR function should assign each participant x all those items y such that the first tuple of every development in c

includes y in x 's ATR. Its time should be the time of the head (first element) of all the developments in c (which will be the same for all, in accordance with (35)), and its addressee and speaker should be the addressee and speaker at that time (which we assume to be uniquely identifiable). More formally:

$$(39) \text{ reduct}(c) = \langle \{p \in \mathbf{D}_{\langle s, t \rangle} \mid \forall \sigma[\sigma \in c \rightarrow p \in CG_{\sigma_0}]\}, \\ \{Q \in \mathbf{D}_{\langle \langle s, t \rangle, t \rangle} \mid \forall \sigma[\sigma \in c \rightarrow Q \in QUD_{\sigma_0}]\}, \\ F, \\ G, \\ ADDR_{\sigma'_0}, \\ SPEAKER_{\sigma'_0}, \\ TIME_{\sigma'_0, \dots} \rangle,$$

where:

- a. for every discourse participant x and every $P \in \mathbf{D}_{\langle s, \langle e, t \rangle \rangle}$:
 $P \in F(x)$ iff for every $\sigma \in c$, $P \in TDL_{\sigma_0}(x)$;
- b. for every discourse participant x and every $D \in \mathbf{D}_{\langle s, \langle d, t \rangle \rangle}$:
 $D \in G(x)$ iff for every $\sigma \in c$, $D \in ATR_{\sigma_0}(x)$;
- c. σ' is any arbitrary development in c .

The reduct of c is a representation of what the discourse participants collectively take in c as definitely in the common ground, questions under discussion, to-do lists, etc.

It may be tempting to think of $\text{reduct}(c)$ as the “real” context, but this would be misleading. As we will see in the next section, disjunction may update contexts in ways that are not reflected in the reduct, but only in the full set of sequences c . In particular, disjunction may result in a context in which the heads of some but not all the sequences contain a particular item in their CG, QUD, or the TDL or ATR or a participant. This item will therefore not be in the CG, QUD, TDL or ATR of the reduct. This technique will allow us to represent contexts which are in “a state of limbo” — where the decision whether to assume something as uncontroversial background, or as a question under discussion, or as an action to be performed, or as an affective trigger, is temporarily deferred, so that its current status is in some sense indeterminate. For example, if some but not all of the sequences in c have proposition p in the CG of their first element, but not proposition q , and the rest all have q but not p , this represents a situation where the discourse participants collectively assume that one of p, q is true, but have not yet decided which. The common ground is “in limbo” between these two possibilities. Because all the sequences in c have first elements whose CG either includes p or includes q , none of the subsequent elements in any of these sequences will contain $[\neg p \wedge \neg q]$. In this way, the full set of sequences c incorporates constraints on later development of the discourse, even though these are not represented in $\text{reduct}(c)$.

5. Disjunction

Representing contexts as in Section 4 requires no revisions to the semantics for conjunction in (25), repeated here as (40):

$$(40) \llbracket and \rrbracket = \lambda p[\lambda q[\lambda c[p(q(c))]]]$$

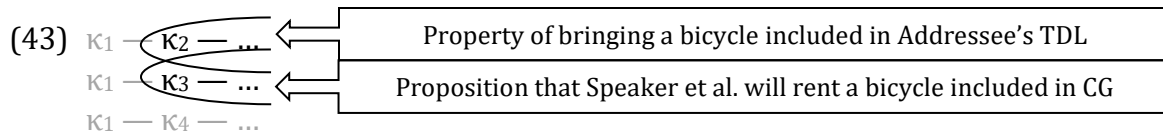
The advantage of the revisions in Section 4 comes in the treatment of disjunction. We may now interpret this via simple set union:

$$(41) \llbracket or \rrbracket = \lambda p[\lambda q[\lambda c[p(c) \cup q(c)]]]$$

This interpretation gives the result that (42), for example, will update the context to include just those developments which form the tails of developments previously in the context, and in whose first tuples either the property of bringing one's own bicycle is in the addressee's TDL or the proposition that a group including the speaker will rent one for \$10 per hour is in the CG.

(42) Bring your own bicycle, or we will rent you one for \$10 per hour.

This can be visualized as in (43), where the entire diagram represents the input context, $\kappa_1, \dots, \kappa_4$ are tuples of a CG, QUD, etc., and the enclosed portions together represent the output context.



Note that the developments in a context — for example the output context in (43) — do not all need to begin with the same tuple. This has the effect of constraining future updates in ways that are not reflected in the reduct of the context, and shows again that our new way of representing contexts distinguishes them more finely than simply using their reducts. Let c be the output context of (43). It might be that κ_2 does not have the proposition p that the speaker and others will rent the addressee a bicycle in its CG, and κ_3 does not have the property P of bringing one's own bicycle in the addressee's TDL. As a result, the reduct of c will not have p in its CG, and will not have the P in the TDL of its addressee x . That is, p is not collectively treated as a shared background assumption by the discourse participants, and P is not collectively treated as something the addressee is slated to do; the context is “in a state of limbo” between these possibilities. Nonetheless, any update to c (setting aside repairs) will consist only of sequences beginning with tuples with p in their CG, or with P in x 's TDL. Other contexts with the same reduct might not be constrained in this way, so we have gained some theoretical power by moving to this more complex representation of context.

Other examples work in a similar way. Updating a context with *John is crazy, or is he just clever?* will eliminate all those developments whose transition from the first to the second element neither adds the proposition that John is crazy in the CG nor adds the question whether John is clever in the QUD, and pass forward the tails of the remaining sequences. Updating with *John is in the house or Mary is in the yard* will pass forward just

those developments forming tails of developments in the previous context, and in whose first tuples the CG includes the proposition that John is in the house or the proposition that Mary is in the yard.

Let's consider some examples in more detail. The simplest case is probably the use of a sentence consisting of smaller disjoined declarative sentences. As we just noted, updating with *John is in the house or Mary is in the yard* will result in a set of developments in whose first tuples the CG includes the proposition that John is in the house (that is, the function $\lambda w[\text{John is in the house in } w]$) or the proposition that Mary is in the yard ($\lambda w[\text{Mary is in the yard in } w]$). By (30a), the proposition that John is in the house or Mary is in the yard will be included in all these tuples. As desired, if we later update with the proposition that John is not in the house and Mary is not in the yard, the common ground of the first tuple in every development in context will contain a contradictory common ground. All the developments terminate at that point, provoking repair.

Examples in which one or more disjuncts are interrogative or imperative present some additional complications, since we must consider not just how the use of such sentences adds questions to the QUD or properties to the addressee's TDL, but also how these questions or properties may be later discharged and removed from the QUD or TDL. Suppose a speaker uses the disjunction of two imperatives:

(44) Wash the dishes, or take out the trash!

Under the analysis developed so far, an utterance of (44) will update the context to include just those developments which form the tails of developments previously in the context, and in whose first tuples either the property W of washing the dishes or the property T of taking out the trash in the addressee's TDL. By (30c), these TDLs will also contain the disjunctive property $W \vee T$ which someone has if they have one or both of the properties of washing the dishes and taking out the trash. If, later in the discourse, the information is received that the addressee has washed the dishes, W is removed from addressee's TDL in those developments where it was included, as is $W \vee T$; but nothing would seem to remove the T or $W \vee T$ from the addressee's TDL in those developments where T was included. Is this a problem?

I think it is *not* a problem. After an utterance of (44) but before the addressee has washed the dishes, all the tuples in the context will include $W \vee T$ in the addressee's TDL, so this property is in the addressee's TDL in the reduct of the context and may be regarded as an obligation of the addressee. (But neither W nor T will be in the addressee's TDL in this reduct, so the addressee is not specifically obliged to perform W , and not specifically obliged to perform T .) After the information has been added to the CG that the addressee has washed the dishes, W and $W \vee T$ are no longer the addressee's to-do list on the branches that previously included W , but T and $W \vee T$ are included on those branches that previously included T . Since neither T nor $W \vee T$ is included on all branches, neither property is now in the addressee's TDL in the reduct, so neither is an obligation of the addressee — the obligation imposed by the utterance of (44) has been discharged. Keeping T and $W \vee T$ on the addressee's to-do list on these branches does no harm in the sense of

misrepresenting the addressee's obligations, and is actually helpful for dealing with the case where the proposition that the addressee has washed the dishes gets added to the CG, but later removed via correction. We will not attempt a formalization of the correction operation here, but it is reasonable to expect that it will not only remove this proposition from the CG, but also restore W and $W \vee T$ to the addressee's TDL on those branches where these had been removed. If this happens, the reduct will revert to having $W \vee T$ in the addressee's TDL, and because T was never removed from those branches where it appears, the addressee may still meet the obligation by taking out the trash.

Example (45), a disjunction of interrogatives, shows some similarities and some differences from the previous example:

(45) When will John arrive at the party, or will he be there at all?

An utterance of (45) will update the context to include just those developments which form the tails of developments previously in the context, and in whose first tuples either the question Q of when John will arrive at the party or the question R of whether he will be there at all is in the QUD. If someone later answers, "At eight o'clock," this answers both Q and R , so in all the developments in the succeeding context, whichever of these was in the QUD will be removed.

What if someone answers "He won't be there at all"? This directly answers R , so R will be removed from the QUDs of those developments where it was present. It does not answer Q , so this will remain in the QUDs of the developments where it was present. Since it is not in the QUDs of *all* developments in the context, it is not authentically a question under discussion. But if a later repair takes place which removes the proposition that John will not be at the party from the CG and restores R to the branches from which it was removed, then a subsequent answer of "At eight o'clock" will be appropriate — as expected if Q were not removed with the initial answer to R .

What if someone answers "He'll be there," but does not give a time? Just as in the previous case, this answers R , and removes it from those developments where it was present. But in contrast to "He won't be there at all," simply answering "He'll be there" does not seem acceptable, unless it is continued with some additional explanation such as "but I don't know when." This, I think, is for Gricean reasons: by using (45) instead of simply asking "Will John be at the party?" the speaker of (45) signaled an interest in the time when John would arrive, if he were to arrive at all, so a response of "He'll be there," without providing a time, would be a violation of the first Maxim of Quantity (albeit one which could be mitigated by explaining that the violation could not be avoided without violating the second Maxim of Quality).

Let us now consider the discharge of questions and orders introduced via disjunctions of unmatched clause types, as in (42), repeated here as (46):

(46) Bring your own bicycle, or we will rent you one for \$10 per hour.

As already mentioned, an utterance of this sentence will pass forward just those developments which form tails of developments in the previous context, and whose first

tuples have the property of bringing one's own bicycle in the addressee's TDL, or have the proposition that (a group containing) the speaker will rent the addressee a bicycle for \$10 per hour in the CG. Since neither of these will be in the addressee's initial TDL or the initial CG of *all* developments in the resulting context, neither will be in the addressee's TDL of the reduct or in the CG of the reduct; the context is not one in which the participants collectively treat the proposition that the speaker will rent the addressee a bicycle as shared knowledge, or the property of bringing one's own bicycle as one of addressee's things to do. Rather, the context is in a limbo state, indeterminate between the possibility of having this proposition as shared knowledge and having this property on the addressee's agenda.

If the addressee then brings their own bicycle (and this is clear to the discourse participants), only those developments are passed forward which form tails of developments previously in the context, and whose first tuples include the proposition that the addressee has brought their own bicycle in the CG and do *not* include the property of bringing one's own bicycle in the addressee's TDL. If (as would usually be the case) it is also in the background that one would not both bring one's own bicycle and rent one, then the set of developments which pass forward would be further restricted to exclude those whose first tuples have the proposition that the speaker will rent the addressee a bicycle in their CGs.

If, on the other hand, it becomes clear to the discourse participants that the addressee has *not* brought their own bicycle, and that the opportunity to do so has passed, then only those tuples will be passed forward whose heads include *this* information in their CGs. Of these, those tuples whose heads have the property of bringing one's own bicycle on the addressee's TDL will terminate at that point, since having the information in a CG that a participant cannot acquire a property in their TDL is a repair-provoking defect. If the speaker then follows through and rents the addressee a bicycle for \$10 per hour, developments which do not have the information that the speaker has done so in their CGs will be filtered out, and the discourse may proceed normally without repair. But if it becomes apparent that the speaker will not follow through in this way, all remaining developments in the context will have contradictory CGs in their heads, and will therefore terminate; some sort of repair will become necessary.

6. Roots, Reducts, and Disjunctions of Speech Acts

Our technique of defining connectives so that they constrain future developments of the context is similar to a technique developed for independent reasons in Cohen and Krifka 2014, Krifka 2015. Corresponding to our tuples $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME, \dots \rangle$, the Krifka/Cohen framework employs the notion of a *commitment state* — a set of propositions representing the commitments of the discourse participants. Corresponding to our notion of a context (a set of sequences of such tuples), Krifka and Cohen employ a *commitment space*:²⁸

²⁸ The definition in (47) follows the formulation in Krifka 2015. Cohen and Krifka 2014 use a slightly different (but equivalent) formulation.

(47) C is a commitment space if C is a set of commitment states, $\bigcap C \neq \emptyset$ and $\bigcap C \in C$.

Like our one of our contexts, a commitment space C represents the set of licit future developments of the discourse. At each step in a possible future development (corrections aside), the new commitment state will be a superset of the previous one. Hence the intersection $\bigcap C$ of all the commitment states in C will be the current commitment state, before any updates have taken place. Krifka and Cohen term this state the *root* of C . The definition in (47) requires that all commitment spaces have a root.

This rootedness requirement is, perhaps, the most important difference between the Krifka/Cohen system and the one presented here. An analogous requirement in our formalism would constrain the sequences in a context to all begin with the same tuple. But as we have seen, it is useful to allow contexts which do not meet this requirement, particularly in the analysis of disjunctive examples like (42), (44), and (45). It is precisely the lack of such a requirement which allows the representation of “limbo” contexts, as developed in Section 5, above.

Despite their rootedness requirement, Cohen and Krifka adopt a semantics for “disjunction of speech acts” which is directly analogous to the dynamic disjunction proposed here. Because this operation frequently results in unrooted structures, which they consider unsuitable as commitment spaces, they regard this disjunction operation as highly restricted, and unavailable except in the few cases where a rooted structure is the result. In all other cases, only a more traditional, static disjunction operation is available as the interpretation of *or*. This seems to me to be problematic, given examples like (42), (44), (45), and the disjunctive examples in (3). Such examples show that we need a disjunction operation which scopes over transitions from context to context — or as Krifka and Cohen put it, over speech acts.

Part of Krifka and Cohen’s motivation for treating speech act disjunction as highly restricted is to preserve the explanation given in Krifka 2001 for why interrogative examples like (48a) do not have a reading paraphrasable as in (48b), despite examples like (49a) having a “pair-list” reading paraphrasable as in (49b):

- (48) a. What dish did some guest make?
 b. For some guest x : What dish did x make?
- (49) a. What dish did every guest make?
 b. For every guest x : What dish did x make?

Existential quantification relates systematically to disjunction: if the guests are John, Mary, and Bill, then *Some guest made pudding* is equivalent to *John made pudding or Mary made pudding or Bill made pudding*. If disjunction over speech acts is not generally available, then existential quantification into speech acts will presumably also not be generally available. Assuming the quantification in examples like (48)–(48) takes scope over speech acts, a restriction against speech act disjunction will also prevent a reading of (48a) paraphrasable as (48b).

It is beyond the scope of this paper to give an analysis of pair-list readings for quantified questions, or of the contrast between examples like (48a) and (49a). Even so, it may be worth considering such examples briefly and pointing out a possible strategy for analysis. As we do this, we must bear in mind that in the framework developed here, it is the reduct of a context, not a root, which represents the common ground, questions under discussion, to-do lists, and affective trigger registries of the discourse participants at any particular point in the discourse. As outlined in Section 4, the reduct of a context includes just that information which is encoded in the first element of every sequence in the context.

Universal quantification relates to conjunction just as existential quantification relates to disjunction, so *Every guest made pudding* is equivalent to *John made pudding and Mary made pudding and Bill made pudding* (again under the assumption that John, Mary and Bill are the guests). Similarly, a reading of (49a) where the universal quantifier has wide scope is equivalent to (50):

(50) What dish did John make and what dish did Mary make and what dish did Bill make?

If we interpret *and* as in (40), then updating a context with (50) in our present framework will result in a new context represented as a set of sequences, all of whose first members include the question of what dish John made, the question of what dish Mary made, and the question of what dish Bill made in their QUDs. Because all three questions are in the QUDs of the first members of all the sequences, all three questions will also be in the QUD of the reduct of this context. This seems a correct characterization of the pair-list reading of (49a).

Similarly, a hypothetical reading of (48a) paraphrasable as in (48b) would be equivalent to (51):

(51) What dish did John make, or what dish did Mary make, or what dish did Bill make?

This example is not any better than (48a) But if we interpret *or* as in (41), then updating a context with this disjunctive question should result in a set of sequences, some of which have (52a) in the QUD of their first element, some of which (52b) in the QUD of their first element, and some of which have (52c) in the QUD of their first element:

(52) a. What dish did John make?
 b. What dish did Mary make?
 c. What dish did Bill make?

None of these questions appear in QUDs of the first elements in *all* the sequences, so none of them is in the QUD of the reduct — none of them is authentically a question under discussion. We thus do not obtain a pair-list reading for (48a).

What *does* get added to the QUD of the reduct in response to the disjunctive question? If anything, it would have to be a question which is contextually entailed by each of the three questions in (52). As outlined in Section 4, we consider a question *Q1* to contextually entail a question *Q2* iff in every world *w*, the true, complete answer to *Q1* in *w* (in combination with the common ground) entails the true, complete answer to *Q2* in *w*. It follows that what is added will be a question whose true answer in any world *w* is the

disjunction of the true answers to (52a–c) in w , along with all the contextual entailments of that question.

For example, suppose that in w_1 , John made pudding, Mary made fish, and Bill made risotto; in w_2 , John made beets, Mary made spam, and Bill didn't make anything; and in w_3 , John made beef, Mary made bread, and Bill made salad. The prediction is that if someone were to ask (48a) or (51), we should answer as in (53):

- (53) a. Correct answer in w_1 : John made pudding or Mary made fish or Bill made risotto.
 b. Correct answer in w_2 : John made beets or Mary made spam or Bill didn't make anything.
 c. Correct answer in w_3 : John made beef or Mary made bread or Bill made salad.

I do not think either (48a) or (51) is naturally interpretable as asking such a question. I am not prepared at this time to present and defend an analysis of why not. But one strategy that might be worth pursuing would be to note that one could know that (53a), for example, was the correct answer to (48a) only if one knew that John made pudding *and* Mary made fish *and* Bill made risotto. But if one knew this, it would normally be a violation of Grice's 1975 first Maxim of Quantity to assert the weaker disjunctive proposition in (53a): that John made pudding *or* Mary made fish *or* Bill made risotto. Because the putative readings of (48a) and (51) call for an answer which is systematically anomalous, we may regard these readings themselves as anomalous, and ruled out on a pragmatic basis.

Whether this line of analysis is tenable is a matter for further research.²⁹ In any case, we should be cautious not to conclude too quickly that the pattern in (48)–(49) demonstrates the unavailability of speech act disjunction, or a general requirement for rooted structures in representing contexts, especially in light of examples like (42), (44), (45), and the disjunctive examples in (3).

7. Subordination

As somewhat of a latecomer to the dynamic semantics bandwagon, I feel I should address an argument which for a long time seemed to me to make dynamic approaches to semantic theory problematic. This argument was based on examples involving subordinate clauses, especially subordinate clauses serving as complements to predicates of mental attitude. Here, I will illustrate the argument using the attitude verb *believe* as an example:

- (54) a. A sentence of the form " α *believes* φ " is true iff the referent of α stands in the belief relation to the content of φ .
 b. The objects of belief (that is, the things to which we stand in the belief relation) are propositions.

²⁹ The literature on disjoined questions and the relative scope of disjunction and an interrogative operator is relevant here; see Hoeks and Roelofsen 2019 and the works cited there. My intuitions about the interpretation of Hoeks and Roelofsen's examples are not at all sharp.

- c. Therefore, in a sentence of the form “ α believes φ ”, the content of φ is a proposition.
- d. But propositions are the contents which are assigned to declarative sentences in “static” semantic theories, not the contents assigned in dynamic semantic theories.

I suspect that many proponents of dynamic semantic theories would respond to this argument by attacking (54b), and trying to motivate the claim that the objects of belief are context change potentials rather than propositions.³⁰ I would like to suggest instead that we can retain (54b) — indeed all of (54) — by limiting our application of dynamic semantic techniques to main clauses.³¹

Perhaps this move will raise concerns about compositionality. It does require that *The world is round* in (55a), for example, has a different content from *the world is round* in (55):

- (55) a. The world is round.
- b. John believes the world is round.

But this is, I think, not a high price to pay. We must draw some sort of syntactic distinction between main and subordinate clauses, in order to account for grammatical constructions which are limited to appearing in main clauses (or those which are limited to appearing in subordinate clauses)³²; and if *The world is round* in (55a) and *the world is round* in (55b) are not instances of the same syntactic item, the motivation for treating them as expressing the same content is much reduced. It should perhaps be noted that subordinate interrogative clauses in English have a systematically different word order from main interrogative clauses, lacking subject-verb inversion as shown in (56).³³ Main and subordinate declarative clauses do not show as dramatic a difference in English, but they do in as closely related a language as German, as shown in (57):

- (56) a. John wonders whether the world is round.
- b. Is the world round?
- (57) a. Die Welt ist rund.
- the world is round

³⁰ Another option for meeting the argument in (54) would be to retain the idea that clauses have static objects such as propositions as their contents/intensions, but have dynamic effects through some other kind of semantic value or dimension of meaning. However, this strategy reintroduces the problem with which we began: What kind of contents can we assign to mixed-clause-type coordinate structures, given that the contents of their conjuncts or disjuncts appear not to match in logical type?

³¹ By “main” clauses, I mean those which share the same patterns of internal syntax as free-standing sentences. Main clauses are also sometimes called “root” clauses, though this terminology is misleading in that not all such clauses correspond to the root node of a free-standing sentence. Main clauses *can* appear as proper parts of larger sentences. In a free-standing sentence of the form “ φ and ψ ”, for example, φ and ψ are both main clauses, not subordinate clauses. Limiting dynamic techniques to main clauses does *not* mean excluding such techniques from the compositional truth definition.

³² See the classic discussion in Emonds 1976, for example.

³³ English also allows subject-verb inversion in a few declarative constructions, but here too it seems noticeably more natural in main than subordinate clauses:

- (i) a. Only rarely does John drink alcohol.
- b. ?Mary believes that only rarely does John drink alcohol.

- b. John glaubt, daß die Welt rund ist.
John believes that the word round is

It is debatable whether subordinate exclamative clauses occur in English,³⁴ but if they do, they are limited to *Wh*-exclamatives; inversion exclamatives occur only in main clauses:³⁵

- (58) a. What a tall building that is!
b. John is amazed what a tall building that is.
c. Is that building ever tall!
d. *John is amazed is that building ever tall.

It is also a matter of debate whether subordinate imperative clauses occur in English. Plausibly, the bracketed portion of (59) provides an example.³⁶ However, I am unaware of major differences in internal syntax between main and subordinate imperatives.

(59) John_i said [eat his_i share of the chicken].

Let us suppose, then, that subordinate but not main clauses are headed by “staticizing” operators, which produce static denotations from dynamic ones.³⁷

- (60) a. $[[STAT-DECL]] = \lambda \mathbf{p}[\lambda w[\forall c \forall \sigma \forall q[[\sigma \in c \wedge Minimal(\sigma_0) \wedge \forall \sigma'[\sigma' \in \mathbf{p}(c) \rightarrow q \in CG_{\sigma'_0}]] \rightarrow q(w) = 1]]]$
b. $[[STAT-INTERROG]] = \lambda \mathbf{p}[\lambda w[\lambda w'[\forall c \forall \sigma \forall Q[[\sigma \in c \wedge Minimal(\sigma_0) \wedge \forall \sigma'[\sigma' \in \mathbf{p}(c) \rightarrow Q \in QUD_{\sigma'_0}]] \rightarrow Q(w)(w') = 1]]]]]$
c. $[[STAT-IMP]] = \lambda \mathbf{p}[\lambda w[\lambda x[\forall c \forall \sigma \forall P[[\sigma \in c \wedge Minimal(\sigma_0) \wedge \forall \sigma'[\sigma' \in \mathbf{p}(c) \rightarrow P \in TDL_{\sigma'_0}(ADDRESSEE_{\sigma'_0})]] \rightarrow P(w)(x) = 1]]]]]$
d. $[[STAT-EXCL]] = \lambda \mathbf{p}[\lambda w[\lambda d[\forall c \forall \sigma \forall D[[\sigma \in c \wedge Minimal(\sigma_0) \wedge \forall \sigma'[\sigma' \in \mathbf{p}(c) \rightarrow D \in ATR_{\sigma'_0}(SPEAKER_{\sigma'_0})]] \rightarrow D(w)(d) = 1]]]]]$

These operators are perhaps most easily understood by example. Suppose *STAT-DECL* takes the sentence *John is a dentist* as its argument. Normally, this sentence denotes a function \mathbf{p} from contexts to contexts, mapping each context c onto a new context $\mathbf{p}(c)$. Here, c and $\mathbf{p}(c)$ are both sets of sequences, and $\mathbf{p}(c)$ differs from c in that each of its sequences is the tail of a sequence in c , such that the head of that tail has the proposition that John is a dentist in its CG, plus all its contextual entailments. If the head of a sequence σ , namely σ_0 , is *Minimal*, that means that σ_0 has a trivial CG, containing just the tautology $\lambda w[w = w]$. Adding the proposition that John is a dentist and all its contextual entailments to this trivial

³⁴ See the discussion in Rett 2011.

³⁵ This is presumably due to a general prohibition on subject-verb inversion structures in subordinate clauses; recall Footnote 33, above.

³⁶ Example (59) is from Roberts 2023. See Crnič and Trinh 2009, Roberts 2023 for arguments supporting the view that such examples involve subordinate imperatives. Kaufmann 2012, Ch. 6 provides a helpful look at the issue of subordinate imperatives across a range of languages.

³⁷ We might also consider an analysis in which main but not subordinate clauses are headed by “dynamizing” operators which derive dynamic denotations from static ones — perhaps a Force operator in the style of Rizzi 1997 and related literature. However, the approach developed here, with staticizing operators in subordinate clauses, will facilitate the treatment of conditionals in the next section.

CG results in the set containing *nothing but* the proposition that John is a dentist and all its entailments. So when c contains a sequence σ whose head is minimal, the only propositions in the CGs of the heads of *all* the sequences σ' in $\mathbf{p}(c)$ are the proposition that John is a dentist and all its entailments. The worlds in which all these entailments are true are precisely the worlds where the proposition that John is a dentist is true. So $\lambda w[\forall c \forall \sigma \forall q[[\sigma \in c \wedge \text{Minimal}(\sigma_0) \wedge \forall \sigma'[\sigma' \in \llbracket \text{John is a dentist} \rrbracket(c) \rightarrow q \in \text{CG}_{\sigma'_0}] \rightarrow q(w) = 1]]$ is just $\lambda w[\llbracket \text{John is a dentist in } w \rrbracket]$ — that is, the proposition that John is a dentist.

Similar reasoning will show that if \mathbf{p} is $\llbracket \text{Is John a dentist?} \rrbracket$, so that it maps each context c onto the set of tails of elements of c whose head includes the question whether John is a dentist — that is, the function $[\lambda w . [\lambda w' . \llbracket \text{John is a dentist in } w \leftrightarrow \text{John is a dentist in } w' \rrbracket]]$ — in its QUD, then $\llbracket \text{STAT-INTERROG} \rrbracket$ will map \mathbf{p} onto this same question. Likewise, if \mathbf{p} is $\llbracket \text{Wash the dishes!} \rrbracket$, so that it maps each context c onto the set of tails of elements of c whose head includes the function $\lambda w \lambda x[x \text{ washes the dishes in } w]$ to the addressee's TDL, then $\llbracket \text{STAT-IMP} \rrbracket$ will map \mathbf{p} onto this same property of individuals. If \mathbf{p} is $\llbracket \text{What a tall building that is!} \rrbracket$, so that it maps each context c onto the set of tails of elements of c whose head includes the function $\lambda w \lambda d[\llbracket \text{That building is } d \text{ tall in } w \rrbracket]$ in the speaker's ATR, then $\llbracket \text{STAT-EXCL} \rrbracket$ will map \mathbf{p} onto this same degree property.

The operators in (60) have the interesting property of distributing over the dynamic connectives, converting them to type-theoretic generalizations of corresponding truth-functional connectives in the process. Let \wedge and \vee be the binary operations on $\{0, 1\}$ defined by the standard truth tables; then define \sqcap , \sqcup inductively as in Partee and Rooth 1983, so that, $x \sqcap y = x \wedge y$ and $x \sqcup y = x \vee y$ whenever x, y are truth values, and $[f \sqcap g](x) = [f(x) \sqcap g(x)]$ and $[f \sqcup g](x) = [f(x) \sqcup g(x)]$ when f, g are functions of a conjoinable type. It then follows that:³⁸

- (61) a. $\llbracket \text{STAT-DECL} \rrbracket(\llbracket \text{and} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-DECL} \rrbracket(\mathbf{p}) \sqcap \llbracket \text{STAT-DECL} \rrbracket(\mathbf{q})$
 b. $\llbracket \text{STAT-INTERROG} \rrbracket(\llbracket \text{and} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-INTERROG} \rrbracket(\mathbf{p}) \sqcap \llbracket \text{STAT-INTERROG} \rrbracket(\mathbf{q})$
 c. $\llbracket \text{STAT-IMP} \rrbracket(\llbracket \text{and} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-IMP} \rrbracket(\mathbf{p}) \sqcap \llbracket \text{STAT-IMP} \rrbracket(\mathbf{q})$
 d. $\llbracket \text{STAT-EXCL} \rrbracket(\llbracket \text{and} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-EXCL} \rrbracket(\mathbf{p}) \sqcap \llbracket \text{STAT-EXCL} \rrbracket(\mathbf{q})$
- (62) a. $\llbracket \text{STAT-DECL} \rrbracket(\llbracket \text{or} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-DECL} \rrbracket(\mathbf{p}) \sqcup \llbracket \text{STAT-DECL} \rrbracket(\mathbf{q})$
 b. $\llbracket \text{STAT-INTERROG} \rrbracket(\llbracket \text{or} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-INTERROG} \rrbracket(\mathbf{p}) \sqcup \llbracket \text{STAT-INTERROG} \rrbracket(\mathbf{q})$
 c. $\llbracket \text{STAT-IMP} \rrbracket(\llbracket \text{or} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-IMP} \rrbracket(\mathbf{p}) \sqcup \llbracket \text{STAT-IMP} \rrbracket(\mathbf{q})$
 d. $\llbracket \text{STAT-EXCL} \rrbracket(\llbracket \text{or} \rrbracket(\mathbf{p})(\mathbf{q})) = \llbracket \text{STAT-EXCL} \rrbracket(\mathbf{p}) \sqcup \llbracket \text{STAT-EXCL} \rrbracket(\mathbf{q})$

These equalities imply a systematic relation between the dynamic conjunction and disjunction operations developed here and the traditional static operations. This allows us

³⁸ To see (61a), note that $\llbracket \text{STAT-DECL} \rrbracket(\llbracket \text{and} \rrbracket(\mathbf{p})(\mathbf{q}))$ is the proposition entailing all the propositions in the common ground that results from updating a trivial common ground with \mathbf{q} , then with \mathbf{p} . This is precisely the conjunction of the proposition which entails all the propositions in the common ground which results from updating a trivial common ground with \mathbf{p} and the proposition which entails all the propositions in the common ground which results from updating a trivial common ground with \mathbf{q} , that is $\llbracket \text{STAT-DECL} \rrbracket(\mathbf{p}) \sqcap \llbracket \text{STAT-DECL} \rrbracket(\mathbf{q})$. The cases of (61b–d) and (62a–d) are analogous.

to define a staticizing operation that derives the standard truth-functional connectives from their dynamic counterparts:

- (63) Where C is an operation mapping pairs of context change potentials to context change potentials, let $STAT(C)$ be that function $f: \{0, 1\} \times \{0, 1\} \rightarrow \{0, 1\}$ such that for all $x, y, z \in \{0, 1\}$: $STAT(C)(x, y) = z$ iff for all worlds w and context change potentials \mathbf{p}, \mathbf{q} , if $\llbracket STAT-DECL \rrbracket(\mathbf{p})(w) = x$ and $\llbracket STAT-DECL \rrbracket(\mathbf{q})(w) = y$, then $\llbracket STAT-DECL \rrbracket(C(\mathbf{p})(\mathbf{q}))(w) = z$ (undefined otherwise).

Here, C is a function of the type appropriate for the (dynamic) denotation of a sentential connective. $\llbracket STAT-DECL \rrbracket(\mathbf{p})$ and $\llbracket STAT-DECL \rrbracket(\mathbf{q})$ are the static propositions p, q which get added to the common ground by updating with \mathbf{p} and with \mathbf{q} , respectively. If $C = \llbracket and \rrbracket$, then $\llbracket STAT-DECL \rrbracket(C(\mathbf{p})(\mathbf{q}))$ is their conjunction $p \sqcap q$, and if $c = \llbracket or \rrbracket$, it is their disjunction $p \sqcup q$. Since for all worlds w , if $p(w) = 1$ and $q(w) = 1$, then $[p \sqcap q](w) = 1$, it follows from (63) that $STAT(\llbracket and \rrbracket)(1, 1) = 1$. Similarly, $STAT(\llbracket and \rrbracket)(1, 0) = 0$, $STAT(\llbracket and \rrbracket)(0, 1) = 0$, and $STAT(\llbracket and \rrbracket)(0, 0) = 0$. In other words, $STAT(\llbracket and \rrbracket)$ is the ordinary truth-functional conjunction operation \wedge . Likewise, $STAT(\llbracket or \rrbracket)$ is truth-functional disjunction \vee . These may be generalized across types to obtain \sqcap and \sqcup as above.

I propose that $STAT$ is freely available to shift the interpretations of *and* and *or* so that they may be interpreted as \wedge and \vee as needed. In particular, these interpretations must be used when *and* and *or* are used to link subordinate clauses, since subordinate clauses were posited above to have static interpretations.

We can now treat predicates like *believe* or *say* as expressing relations to propositions, predicates like *wonder* or *ask* as expressing relations to questions (functions from worlds to propositions), etc., in standard fashion.

Treating subordinate clauses as semantically static may seem an odd and problematic position, since subordinate clauses of different clause types may be coordinated with one another, and our adoption of dynamic semantics was motivated specifically by the need to account for coordinations of clauses of different clause types. But coordination of subordinate clauses of different clause types is much more limited than coordination of main clauses of different clause types, and in fact seems restricted to cases where we have independent reason to assign semantic values of the same logical type to clauses of different clause types.

For example, assuming that *know* denotes a relation between individuals and propositions, we have reason from examples like (64a) to assign the declarative clause *that Mary stole a car* a proposition as a semantic value (specifically, its static intension), and we also have reason from examples like (64b) to assign the interrogative clause *who Mary stole it from* a proposition as a semantic value (in this case, its static extension). Since both the declarative and the interrogative have propositions as semantic values (albeit the extension in one case and the intension in the other), the two clauses can be conjoined, with the result also having a propositional semantic value, which can then serve as argument to *know* as in (64c):

- (64) a. John knows that Mary stole a car.
 b. John knows who Mary stole it from.
 c. John knows that Mary stole a car, and who she stole it from.

The conjunction in (64c) may be treated as \sqcap , obtained from $\llbracket and \rrbracket$ by application of *STAT* (together with standard generalization across types to apply to functions of type $\langle s, t \rangle$); no dynamics need enter the picture.³⁹

In contrast, when there is no independent reason to assign semantic values of the same logical type to subordinate clauses of different clause types, coordination is impossible. For example, *wonder* represents a relation to a question, not a relation to a proposition; it takes arguments of type $\langle s, \langle s, t \rangle \rangle$, not of type $\langle s, t \rangle$. The contrast between (65a) and (65b) suggests that interrogative clauses have semantic values of type $\langle s, \langle s, t \rangle \rangle$ (namely, their static intensions), but declarative clauses do not.

- (65) a. *John wonders that Mary stole a car.
 b. John wonders who she stole it from.

Accordingly, declarative and interrogative clauses cannot be conjoined in this context:

- (66) *John wonders that Mary stole a car, and who she stole it from.

8. Conditionals

In English conditional sentences, the antecedent clause is a subordinate clause. The consequent clause may be either main or subordinate. English conditionals differ in this respect from examples with *and* or *or*, in which the linked clauses are either both main or both subordinate. As already noted, the antecedent clause in a conditional is also always declarative, but the consequent may be of any clause type. The conditional sentence as a whole usually inherits the clause type of the consequent clause. However, it should be noted that the consequent may be a coordinate structure, in which the conjuncts are of different clause types:

- (67) a. If John doesn't show up on time, I will be angry, and what a fuss I will raise!
 b. If the dishes are dirty, then wash them, and I will dry.
 c. If John doesn't pay up, I will pay him a visit, or would you rather do that yourself?

All this suggests that our analysis of the word *if* should treat it as expressing a relation between propositions and context change potentials. A natural option is to give *if* the denotation in (68):⁴⁰

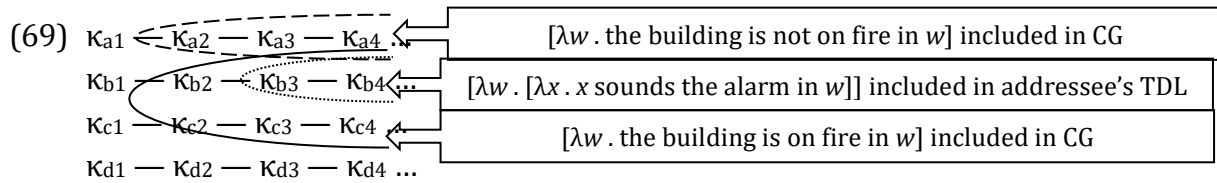
- (68) $\llbracket if \rrbracket = [\lambda p . [\lambda q . [\lambda c . (c + \neg p) \cup q(c + p)]]]$

³⁹ Depending on our analysis of internal clause structure, we might also consider using \sqcap and \sqcup to interpret simple examples of phrasal conjunction, such as *over or under the table*, even in main clauses. Alternatively, I see no obstacle to analyzing *and* and *or* as dynamic in such examples, and nothing in the present analysis seems to force a choice between static and dynamic analyses.

⁴⁰ Here, $\neg p$ is $\lambda w[p(w) = 0]$, and $+$ is as in (37), above.

Put more sloppily but perhaps more comprehensibly, “if p , q ” updates the context by filtering out future developments, leaving only those which either result from adding $\neg p$ to their CGs, or result from adding p to their CGs, followed by whatever update is performed by q .

As an example: Suppose in context c , the speaker says, “If the building is on fire, sound the alarm!” We represent c as a set of developments $\{\langle \kappa_{a1}, \kappa_{a2}, \kappa_{a3}, \kappa_{a4}, \dots \rangle, \langle \kappa_{b1}, \kappa_{b2}, \kappa_{b3}, \kappa_{b4}, \dots \rangle, \langle \kappa_{c1}, \kappa_{c2}, \kappa_{c3}, \kappa_{c4}, \dots \rangle, \dots\}$. The elements of these sequences are all tuples of a CG, QUD, TDL-function, etc. Suppose that κ_{a2} differs from κ_{a1} in having the proposition that the building is *not* on fire in its CG. In addition, κ_{b2} differs from κ_{b1} in having the proposition that the building *is* on fire in its CG, and κ_{b3} differs from κ_{b2} in having the property of sounding the alarm in the addressee’s TDL. In passing from κ_{c1} to κ_{c2} , the proposition that the building is on fire is added to the CG, but nothing is added to the addressee’s TDL. And in the development starting with κ_{d1} , neither the proposition that the building is on fire, nor the proposition that the building is not on fire, is added to the CG. All this can be visualized as in (69):



The output context is the union of the areas enclosed by the dashed and dotted lines: $\{\langle \kappa_{a2}, \kappa_{a3}, \kappa_{a4}, \dots \rangle, \langle \kappa_{b3}, \kappa_{b4}, \dots \rangle, \dots\}$. Intuitively, the output context is “in limbo” between accepting it as given that the building is not on fire, and accepting it as given that it *is* on fire, with sounding the alarm on the addressee’s to-do list. This limbo state may be resolved later in the discourse, as definite information is received whether the building is on fire or not; if it is, sounding the alarm will definitely be on the addressee’s to-do list.

Conditionals may appear as subordinate clauses, so we should examine the interaction between (68) and (60). First let us consider an example where the consequent clause (hence the conditional structure as a whole) is declarative, such as (70):

(70) If this is John’s footprint, he is the murderer.

According to (60), applying *STAT-DECL* to the context change potential denoted by (70) will result in that proposition which is true in a world w iff, when you update a trivial CG with (70), every proposition which is in all the resulting CGs is true in w . According to (68), updating with (70) adds the proposition $\neg p$ that this is *not* John’s footprint to some of these CGs, and adds the proposition $[p \wedge q]$ that this *is* John’s footprint and he is the murderer to the rest. In order to be in *all* these CGs, a proposition would have to be entailed both by $\neg p$ and by $[p \wedge q]$. The most specific such proposition is $[\neg p \vee [p \wedge q]]$. This is straightforwardly equivalent to the material conditional $[p \rightarrow q]$. It is easy to see that this proposition entails all propositions which are entailments both of $\neg p$ and of $[p \wedge q]$,⁴¹ so $[p \rightarrow q]$ is the result of

⁴¹ If it did not, there would have to be a proposition r entailed by $\neg p$ and entailed by $[p \wedge q]$, but not entailed by $[p \rightarrow q]$. Since r is not entailed by $[p \rightarrow q]$, there must be a world w where $[p \rightarrow q]$ is true but r is

applying *STAT-DECL* to (70). The end result is that in an example like *Mary believes that if this is John's footprint, he is the murderer*, it is the material conditional [*this is John's footprint* \rightarrow *he is the murderer*] which serves as the object of belief.

Whether one sees this as a desirable result will of course depend on one's views on the issue of whether the material conditional provides an adequate model for the semantics of English *if*. If one prefers another analysis of natural language conditionals, it may be possible to replace (68) with a different rule which "dynamizes" one's preferred analysis; but exploring the wide range of alternatives here would take us too far afield.

What about examples involving a subordinate interrogative clause with a conditional structure, as in *John wonders who the murderer is, if Mary is innocent*? Here, we must apply the *STAT-INTERROG* operator defined in (60b) to the context change potential denoted by (71):

(71) If Mary is innocent, who is the murderer?

The result will be that question which relates two worlds w, w' iff, when you update a trivial CG and QUD with (71), every question which is in all the resulting QUDs relates w to w' — that is to say, applying *STAT-INTERROG* to the context change potential denoted by (71) results in that question Q which contextually entails every question which is in all the QUDs resulting from the update of a trivial CG and QUD by (71).

By (68), updating a context with (71) passes forward those sequences whose heads either include the proposition that Mary is not innocent in their CGs, or include the proposition that she *is* innocent in their CGs and the question of who the murderer is in their QUDs. Since this question is included in the QUDs of the heads of only some of these sequences, one may well wonder whether there will be any questions which are in all the QUDs resulting from the update of a trivial CH and QUD by (71), hence whether there will be any question Q which entails all such questions as required.

The answer is yes, because our update operations in (30) close CGs, QUDs, and TDLs under contextual entailment. Contextual entailment for questions is defined in (28)a). Intuitively, where QUD is a set of questions and CG is a set of propositions, QUD entails a question Q relative to CG (QUD $\models_{CG} Q$) iff the true complete answer to Q follows from the true complete answers to all the questions in QUD, together with all the propositions in CG. Under this definition, even when QUD is trivial, it can entail a non-trivial question Q relative to CG, provided CG is not also trivial.

Specifically, a large set of "conditional questions" will be contextually entailed. Let us define conditional questions as in (72):⁴²

false. But any world where [$p \rightarrow q$] is true is one where either $\neg p$ is true, or one where [$p \wedge q$] is true, and both of these entail r .

⁴² Perhaps some readers will wonder why the dynamic account of *if* given in (68) is necessary, if an operator like \rightarrow is available, which allows construction of static questions expressed with declarative antecedents and interrogative consequents. I would remind such readers of examples like those in (67), where the consequent is a coordinate structure with unmatched clause types as the conjuncts.

(72) Where p is a proposition and Q is a question,
 $[p \rightarrow Q] = \lambda w \lambda w' [p(w) = 1 \rightarrow Q(w)(w') = 1]$

For example, where p is $\lambda w [\text{Mary is innocent in } w]$ (the proposition that Mary is innocent), and Q is $\lambda w \lambda w' [\lambda x [x \text{ is the murderer in } w] = \lambda x [x \text{ is the murderer in } w']]$ (the question of who the murderer is), $[p \rightarrow Q]$ will be the question in (73):

(73) $\lambda w \lambda w' [\text{Mary is innocent in } w \rightarrow \lambda x [x \text{ is the murderer in } w] = \lambda x [x \text{ is the murderer in } w']]$

This function maps any world w onto the set of worlds which have the same murderer as w does, *provided* Mary is innocent in w . If Mary is not innocent in w , it maps w onto the tautology.

Now, suppose that the CG contains just the proposition that Mary is not innocent (and its entailments), and QUD contains just the trival question $\lambda w \lambda w' [w = w']$. Is it the case that $\text{QUD} \models_{\text{CG}} (73)$? Yes, because any world where all the propositions in CG are true will be one in which the antecedent of the conditional in (73) is false, hence one in which the whole conditional is true, regardless of the nature of w' .

In the alternative case where CG contains the proposition that Mary *is* innocent and QUD contains the question of who the murderer is, it will still hold that $\text{QUD} \models_{\text{CG}} (73)$. Since in this case QUD contains the question of who the murderer is ($\lambda w \lambda w' [\lambda x [x \text{ is the murderer in } w] = \lambda x [x \text{ is the murderer in } w']]$), and any pair of worlds satisfying this question will also render the consequent of the conditional in (73) true, thereby satisfying (73) as a whole, we can see that any pair of worlds satisfying all the questions in QUD will also satisfy (73).

We have now seen that updating a trivial context with (71) will result in a context in which the head of every sequence has (73) in its QUD. To show that (73) is the result of applying *STAT-INTERROG* to the context change potential denoted by (71), we need now only show that (73) entails all other questions which are in those same QUDs. Suppose updating a minimal sequence with (71) results in a sequence whose head has some question Q in its QUD. This means that either $\lambda w \lambda w' [w = w'] \models_{\lambda w [\text{Mary is not innocent in } w]} Q$ or $\lambda w \lambda w' [\lambda x [x \text{ is the murderer in } w] = \lambda x [x \text{ is the murderer in } w']] \models_{\lambda w [\text{Mary is innocent in } w]} Q$. If it were the case that Q is *not* entailed by (73), there would have to be a pair of worlds which satisfy (73) but do not satisfy Q . But any pair of worlds w, w' satisfying (73) will either be such that Mary is not innocent in w , or such that Mary is innocent in w and $\lambda x [x \text{ is the murderer in } w] = \lambda x [x \text{ is the murderer in } w']$. In either case, w, w' must satisfy Q . Therefore (73) is the result of applying *STAT-INTERROG* to the context change potential denoted by (71).

9. Conclusion

Coordinate and conditional structures built by connecting unmatched clause types can be interpreted using a dynamic semantic theory with look-ahead. This is consistent with the view that the objects of attitudes are objects of the kind assigned to sentences in static semantic theories.

Many more issues remain to be explored in the kind of semantic theory advocated here, especially in connection with negation,⁴³ quantification, and anaphora; but such exploration must be a topic for a different paper.

References

- Charlow, Nate (2014) Logic and Semantics for Imperatives. *Journal of Philosophical Logic* 43.4.617–664. <https://doi.org/10.1007/s10992-013-9284-4>
- Clark, Billy (1993) Relevance and “Pseudo-Imperatives”. *Linguistics and Philosophy* 16.1.79–121. <https://doi.org/10.1007/BF00984723>
- Cohen, Ariel and Manfred Krifka (2014) Superlative Quantifiers and Meta-Speech Acts. *Linguistics and Philosophy* 37.1.41–90. <https://doi.org/10.1007/s10988-014-9144-x>
- Condoravdi, Cleo and Sven Lauer (2017) Conditional Imperatives and Endorsement. *NELS 47: Proceedings of the Forty-Seventh Annual Meeting of the North East Linguistic Society*, ed. by A. Lamont and K. Tetzloff, vol. 1, 185–204. Graduate Linguistics Student Association.
- Crnič, Luka and Tue Trinh (2009) Embedding Imperatives in English. *Proceedings of Sinn Und Bedeutung 13*, ed. by A. Riester and T. Solstad, vol. 1, 109–123. Online Publikationsverbund der Universität Stuttgart.
- Emonds, Joseph E. (1976) *A Transformational Approach to English Syntax : Root, Structure-Preserving, and Local Transformations*. Academic Press.
- von Fintel, Kai and Sabine Iatridou (2017) A Modest Proposal for the Meaning of Imperatives. *Modality across Syntactic Categories*, ed. by A. Arregui, M. L. Rivero and A. Salanova, 288–319. Oxford University Press. <https://doi.org/10.1093/acprof:oso/9780198718208.003.0013>
- Grice, H.P. (1975) Logic and Conversation. *Speech Acts*, ed. by P. Cole and J. L. Morgan, 41–58. Academic Press. https://doi.org/10.1163/9789004368811_003
- Groenendijk, Jeroen and Martin Stokhof (1982) Semantic Analysis of Wh-Complements. *Linguistics and Philosophy* 5.2.175–234. <https://doi.org/10.1007/BF00351052>
- Groenendijk, Jeroen and Martin Stokhof (1984) *Studies on the Semantics of Questions and the Pragmatics of Answers*. University of Amsterdam dissertation.
- Hausser, Roland R. (1980) Surface Compositionality and the Semantics of Mood. *Speech Act Theory and Pragmatics*, ed. by J. R. Searle, F. Kiefer and M. Bierwisch, 71–95. Springer Netherlands. https://doi.org/10.1007/978-94-009-8964-1_4
- Heim, Irene (1982) *The Semantics of Definite and Indefinite Noun Phrases*. Graduate Linguistics Students Association, University of Massachusetts.

⁴³ Because negation applies to a single clause (or smaller phrase) rather than linking whole clauses, the issue of unmatched sentence types does not arise with negation. Arguments of the kind offered in this paper for dynamic interpretations of the connectives therefore do not motivate a dynamic interpretation for negation, and an interpretation which is dynamic in the way discussed in this paper does not seem generally available. Perhaps such an interpretation might be posited to account for “denegation,” as in Cohen and Krifka 2014; but if so, it must be suitably constrained to prevent *Do not close the door!*, for example, from being interpreted as “You will not be ordered to close the door.” (Thanks to an anonymous referee for this example.)

- Hoeks, Morwenna and Floris Roelofsen (2019) Coordinating Questions: The Scope Puzzle. *Proceedings of Salt 29*, ed. by K. Blake, F. Davis, K. Lamp and J. Rhyne, 562–581. Linguistic Society of America. <https://doi.org/10.3765/salt.v29i0.4632>
- Hulstijn, Joris (1997) Structured Information States : Raising and Resolving Issues. *Proceedings of Mundial '97*, ed. by A. Benz and G. Jäger, 99–117. University of Munich.
- Isaacs, James and Kyle Rawlins (2008) Conditional Questions. *Journal of Semantics* 25.3.269–319. <https://doi.org/10.1093/jos/ffn003>
- Kaufmann, Magdalena (2012) *Interpreting Imperatives*. Springer. <https://doi.org/10.1007/978-94-007-2269-9>
- Keshet, Ezra (2013) Focus on Conditional Conjunction. *Journal of Semantics* 30.2.211–256. <https://doi.org/10.1093/jos/ffs011>
- Keshet, Ezra and David Medeiros (2019) Imperatives under Coordination. *Natural Language & Linguistic Theory* 37.3.869–914. <https://doi.org/10.1007/s11049-018-9427-y>
- Krifka, Manfred (2001) Quantifying into Question Acts. *Natural Language Semantics* 9.1.1–40. <https://doi.org/10.1023/A:1017903702063>
- Krifka, Manfred (2015) Bias in Commitment Space Semantics: Declarative Questions, Negated Questions, and Question Tags. *Proceedings of Semantics and Linguistic Theory 25*, ed. by S. D'Antonio, M. Moroney and C. R. Little, 328–345. Linguistic Society of America. <https://doi.org/10.3765/salt.v25i0.3078>
- Partee, Barbara H. and Mats Rooth (1983) Generalized Quantifiers and Type Ambiguity. *Meaning, Use, and Interpretation of Language*, ed. by R. Bäuerle, C. Schwarze and A. v. Stechow, 361–383. Walter de Gruyter. <https://doi.org/10.1515/9783110852820.361>
- Portner, Paul (2004) The Semantics of Imperatives within a Theory of Clause Types. *Proceedings from Semantics and Linguistic Theory Xiv*, ed. by R. Young, 235–252. CLC Publications. <https://doi.org/10.3765/salt.v14i0.2907>
- Rett, Jessica (2011) Exclamatives, Degrees and Speech Acts. *Linguistics and Philosophy* 34.5.411–442. <https://doi.org/10.1007/s10988-011-9103-8>
- Rizzi, Luigi (1997) The Fine Structure of the Left Periphery. *Elements of Grammar : Handbook in Generative Syntax*, ed. by L. Haegeman, 281–337. Kluwer. https://doi.org/10.1007/978-94-011-5420-8_7
- Roberts, Craige (2012) Information Structure: Towards an Integrated Formal Theory of Pragmatics. *Semantics and Pragmatics* 5.6.1–69. <https://doi.org/10.3765/sp.5.6>
- Roberts, Craige (2023) Imperatives in a Dynamic Pragmatics. *Semantics and Pragmatics* 16.7. <https://doi.org/10.3765/sp.16.7>
- Sano, Katsuhiko and Yurie Hara (2014) Conditional Independence and Biscuit Conditional Questions in Dynamic Semantics. *Proceedings of SALT 24*, ed. by T. Snider, S. D'Antonio and M. Weigand, 84–101. Linguistic Society of America. <https://doi.org/10.3765/salt.v24i0.2417>
- Skelac, Ines (2016) Pseudo-Imperatives: The Special Case of Natural Language Conjunction and Disjunction. *Research Journal of English Language and Literature* 4.4.602–607.
- Stalnaker, Robert (1978) Assertion. *Pragmatics*, ed. by P. Cole, 315–332. Academic Press. https://doi.org/10.1163/9789004368873_013

- Starr, William (2018) Conjoining Imperatives and Declaratives. *Proceedings of Sinn Und Bedeutung 21*, ed. by R. Truswell, C. Cummins, C. Heycock, B. Rabern and H. Rohde, vol. 2, 1159–1176. University of Konstanz.
- Swanson, Eric (2022) Channels for Common Ground. *Philosophy and Phenomenological Research* 104.1.171–185. <https://doi.org/10.1111/phpr.12741>
- Velissaratou, Sophia (2000) *Conditional Questions and Which-Interrogatives*. University of Amsterdam dissertation.
- Villalba, Xavier (2008) Exclamatives: A Thematic Guide with Many Questions and Few Answers. *Catalan Journal of Linguistics* 7.9–40. <https://doi.org/10.5565/rev/catjl.131>
- Warstadt, Alex (2020) “Just” Don’t Ask: Exclusives and Potential Questions. *Proceedings of Sinn Und Bedeutung 24*, ed. by M. Franke, N. Kompa, M. Liu, J. L. Mueller and J. Schwab, vol. 2, 373–390. Osnabrück University.
- Zanuttini, Raffaella and Paul Portner (2003) Exclamative Clauses: At the Syntax-Semantics Interface. *Language* 79.1.39–81. <https://doi.org/10.1353/lan.2003.0105>