Coordination of Unmatched Clause Types and Dynamic Look-Ahead

Peter Lasersohn
University of Illinois
Unpublished draft, 3/26/2024

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 (the sort of expression that can have a truth value), but they might also all be of type $\langle e, t\rangle$ (one-place predicates), or of type $\langle e, \langle e, t\rangle\rangle$ (two-place predicates), or of any of various other types, provided they are all of the *same* type.[1] 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. [[$_t$ *John is happy*] *and* [$_t$ *Mary is angry*]].
b. *John* [[$_{\langle e, t\rangle}$ *loves Mary*] *and* [$_{\langle e, t\rangle}$ *hates Bill*]].
c. *John* [[$_{\langle e, \langle e, t\rangle\rangle}$ *loves*] *or* [$_{\langle e, \langle e, t\rangle\rangle}$ *hates*]] *Mary.*

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

[1] See Partee and Rooth (1983) for a particularly influential analysis incorporating this constraint.

l. *Be alert, or what terrible consequences you will suffer!* (imperative-*or*-exclamative)

Following Frege (1892), declarative clauses are standardly held to denote truth values, which is to say they are of type t. Many treatments of interrogative clauses treat them as denoting sets of propositions (e.g. Hamblin (1973), Karttunen (1977)) — that is, as being of type $\langle\langle s, t\rangle, t\rangle$; in the related approach of Groenendijk and Stokhof (1982), which we will adopt here, they are analyzed as denoting propositions, hence of type $\langle s, t\rangle$. (Their intensions are therefore functions in type $\langle s, \langle s, t\rangle\rangle$.[2]) Imperative clauses have not received as much attention in the formal semantic literature as declaratives and interrogatives, but a natural and reasonable option, advocated for example by Hausser (1980), Portner (2004), is to treat them as expressing properties,[3] hence of type $\langle e, t\rangle$ or something similar. Exclamatives are even less well-studied, and a variety of logical types have been proposed for them; a prominent approach advocated by Rett (2011) would treat them as predicates of degrees, hence as type $\langle d, t\rangle$ or something similar,[4] where d is the type of degrees.[5] Nonetheless, all these clause types[6] 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)

---

[2] The function serving as 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 $\lambda w\lambda w'$[John was at the party in $w \leftrightarrow$ John was at the party in w']; *Who was at the party?* has as its intension the function $\lambda w\lambda w'[\lambda x[x$ was at the party in $w] = \lambda x[x$ was at the party in $w']]$.

[3] Perhaps it would be more intuitive to use the term *actions*, rather than *properties*, for the contents of imperative clauses; but we may regard actions (in the relevant sense) simply as properties which are under the agentive control of the things they hold of. Note that properties like keeping still must count as actions under this characterization. For obvious pragmatic reasons, non-agentive properties would rarely serve as the contents of imperatives, so we might not need to posit a *grammatical* limitation of the contents of imperatives to actions.

[4] It is not completely clear from Rett's presentation whether she adopts an intension/extension distinction. If so, exclamatives would have extensions in type $\langle d, t\rangle$ and intensions in type $\langle s, \langle d, t\rangle\rangle$; if not, they would have a single semantic value in type $\langle s, \langle d, t\rangle\rangle$.

[5] See Zanuttini and Portner (2003), Villalba (2008), Rett (2011), and the references cited in these works for a variety of proposals concerning the logical type of exclamatives.

[6] I will 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. It is unfortunate that the term *type* is too well established to permit an alternative in either system of classification.

How can we analyze *and*, *or*, and *if* in a way which makes sense of these constructions?

2. Some problematic strategies

A wide variety of possible strategies for developing an analysis suggest themselves. We cannot possibly review them all here, but perhaps the one I wish to pursue may be seen as more plausible if we first note some problems with some of the others.

2.1. Analyze connectives as always joining propositions, which can be the objects of various speech acts

One very simple approach would be to analyze declarative, interrogative, imperative, and exclamative clauses as *not* differing from one another in denotation, but only in illocutionary force. We would regard all four clause types as being of type *t* — that is, as having truth values as their denotations/extensions, and propositions as their contents/intensions. One can perform a variety of different acts with a proposition as their object: assert that the proposition is true, ask whether it is true, command someone to make it true, express surprise at its being true, etc. We would regard the declarative, interrogative, imperative, and exclamative sentence structures as signaling which of these acts the sentence is used to perform, at least at some basic or default level which excludes explicit performatives and indirect speech acts. But since the denotation would be of the same type — namely *t* — no matter which act is performed, we could analyze conjunctions, disjunctions, and conditionals of unmatched clause types using a simple truth table analysis.

This is a poor strategy for several reasons. First, it does not extend naturally to *Wh*-questions. *Who is happy?* does not present a proposition for confirmation or disconfirmation the way *Is John happy?* does. Second, it does not extend naturally to subordinate clauses. The bracketed clause in (5)a. is declarative, but a speaker who uttered this sentence would not thereby assert the bracketed clause. The bracketed clause in (5)b. is interrogative, but a speaker who uttered this sentence would not thereby ask whether the bracketed clause were true.

(5) a. *John imagined* [*that Mary stole the money*].
    b. *John wondered* [*whether Mary stole the money*].

On the assumption that verbs like *know* represent relations between their subjects and the contents of their complement clauses, examples like those in (6) are particularly problematic for the analysis in question, since they differ in truth conditions, but are identical in form except that (6)b. contains an interrogative clause where (6)a. contains its corresponding declarative:

(6) a. *John knows that Mary stole the money.*
    b. *John knows whether Mary stole the money.*

Finally, even though this analysis portrays declarative, interrogative, and imperative clauses as conjoinable with one another, it does not seem to do so in a way that helps in explaining what the resulting structure means or how it is used. The analysis tells us that (7), for example, expresses the proposition that the addressee washes the dishes and the speaker will dry them, but this proposition seems to play no further role in the analysis.

(7) *Wash the dishes, and I will dry.*

Each of the conjuncts expresses a proposition which serves as the object of a different speech act — the proposition expressed by the first conjunct as the object of a request and the proposition expressed by the second conjunct as the object of an assertion. But what act is performed of which the conjunction of these two propositions is the object?

2.2. Analyze non-declarative clauses as expressing the proposition that a question, order/request, or exclamation is being performed

A different strategy would be to adopt some version of the "higher performative" analysis (Lewis (1972), Ross (1970), Sadock (1974)), in which non-declarative clauses derive from underlying structures roughly of the forms "I ask you whether…," "I order/request you to…," or "I exclaim to you that…" Since these forms are declarative, we would analyze all clause types as being of type *t*, and interpret conjunctions and disjunctions using standard truth tables.

The higher performative analysis has not been popular for some time, for well-known reasons.[7] Setting these concerns aside, this approach does give reasonably intuitive results for some examples, especially those with *and* or *if*:

(8) a. *Wash the dishes and I will dry* ⇨
I request you to wash the dishes, and I will dry them.
b. *John left town, and after all, why wouldn't he?* ⇨
John left town, and I ask you why, after all, he wouldn't.
c. *If John is so smart, then why can't he answer the question?* ⇨
If John is so smart, then I ask you why he can't answer the question.
d. *If he tries to escape, kill him!* ⇨
If he tries to escape, then I order you to kill him.

However, it does not do so well with other examples, especially those involving *or*:

(9) a. *Leave now or I will shoot!* ⇨
Either I order you to leave now, or I will shoot.
b. *John is crazy, or is he just clever?* ⇨
Either John is crazy, or I ask you whether he is just clever.

If Bill utters (9)a., is he ordering the addressee to leave, or is he asserting that he will shoot the addressee? According to the analysis, he must be doing one or the other, though the sentence leaves it unclear which. But intuitively, what he is doing is ordering the addressee to leave, and asserting that if the addressee does not comply with this order, he will shoot. If Bill utters (9)b., is he asserting that John is crazy, or is he asking whether John is clever? According to the analysis, he must be doing one or the other, though the sentence again

[7] For a summary, see Levinson (1983) §5.2–5.4.

leaves it unclear which. But intuitively, he first asserts that John is crazy, then retracts or qualifies that assertion by asking whether he is just clever instead.

In addition, the analysis in question faces difficulties in accounting for examples where conjunctions, disjunctions, or conditionals involving unmatched clause types appear as subordinate clauses:

(10) a. *Mary wondered why John couldn't answer the question, if he was really so smart.*
b. *John knew that Mary had bought some beets, and who she bought them from.*
c. *Bill wasn't sure that Mary would care, or why.*

Typically, the higher performative analysis is applied only to main clauses, not to subordinate clauses, so examples like these should be unexpected. If we try extending the analysis so that subordinate clauses are represented in the same way as main clauses, we derive incorrect results. The examples in (10) do not have the meanings indicated in (11):

(11) a. *Mary wondered* [*John is so smart* → *I ask you why John couldn't answer the question*].
b. *John knew* [*Mary bought some beets* ∧ *I ask you who she bought them from*].
c. *Bill wasn't sure that* [*Mary would care* ∨ *I ask you why she would care*].

Indeed, it is unclear what (11)a. even means, since *wonder* requires an interrogative complement.

2.3. Type-shift one of the clauses so that they match

Under this approach, we would analyze declarative, interrogative, and imperative clauses as normally having denotations of different logical types, but posit rules assigning them secondary "type-shifted" denotations of other types when needed. Specifically, when one clause is conjoined with another of a different clause type, one of the two would be assigned a denotation of a different logical type than usual, so that the two clauses matched in logical type despite the difference in clause type.

Whatever type-shifting rules we adopt, we would want them to work with standard type-theoretic generalizations of conjunction, disjunction, and implication. These give the following effects:

(12) Where φ and ψ are expressions of type $\langle s, t \rangle$:
a. $[\![\varphi \textit{ and } \psi]\!]^{M,w,g} = \lambda w'[[\![\varphi]\!]^{M,w,g}(w') = 1 \;\&\; [\![\psi]\!]^{M,w,g}(w') = 1]$
b. $[\![\varphi \textit{ or } \psi]\!]^{M,w,g} = \lambda w'[[\![\varphi]\!]^{M,w,g}(w') = 1 \vee [\![\psi]\!]^{M,w,g}(w') = 1]$
c. $[\![\textit{if } \varphi \textit{ then } \psi]\!]^{M,w,g} = \lambda w'[[\![\varphi]\!]^{M,w,g}(w') = 1 \rightarrow [\![\psi]\!]^{M,w,g}(w') = 1]$

(13) Where α and β are expressions of type $\langle e, t \rangle$:
a. $[\![\alpha \textit{ and } \beta]\!]^{M,w,g} = \lambda x_e[[\![\alpha]\!]^{M,w,g}(x) = 1 \;\&\; [\![\beta]\!]^{M,w,g}(x) = 1]$
b. $[\![\alpha \textit{ or } \beta]\!]^{M,w,g} = \lambda x_e[[\![\alpha]\!]^{M,w,g}(x) = 1 \vee [\![\beta]\!]^{M,w,g}(x) = 1]$
c. $[\![\textit{if } \alpha \textit{ then } \beta]\!]^{M,w,g} = \lambda x_e[[\![\alpha]\!]^{M,w,g}(x) = 1 \rightarrow [\![\beta]\!]^{M,w,g}(x) = 1]$

(14) Where α and β are expressions of type $\langle d, t \rangle$:
a. $[\![\alpha \textit{ and } \beta]\!]^{M,w,g} = \lambda x_d[[\![\alpha]\!]^{M,w,g}(x) = 1 \;\&\; [\![\beta]\!]^{M,w,g}(x) = 1]$
b. $[\![\alpha \textit{ or } \beta]\!]^{M,w,g} = \lambda x_d[[\![\alpha]\!]^{M,w,g}(x) = 1 \vee [\![\beta]\!]^{M,w,g}(x) = 1]$

c. $[\![\textit{if}\ \alpha\ \textit{then}\ \beta]\!]^{M,w,g} = \lambda x_d[[\![\alpha]\!]^{M,w,g}(x) = 1 \rightarrow [\![\beta]\!]^{M,w,g}(x) = 1]$

As an example, we might posit a rule which assigns a declarative sentence (normally of type t) a secondary denotation of type $\langle s, t\rangle$ so that it could conjoin with an interrogative. A couple of fairly natural rules along these lines suggest themselves: One would let the secondary denotation be the proposition which the declarative normally expresses as its *in*tension. Another would let the secondary denotation be the function mapping each world w either onto this proposition or its negation, depending on which of these is true in w — that is, the function which would serve as the denotation of the corresponding *whether*-interrogative. In either case — indeed in *any* analysis along these lines — we encounter a similar problem to one we saw in Section 2.1, above: Intuitively, when clauses of different clause types are conjoined, the content of each conjunct serves as the object of a different kind of (direct) speech act. Normally, a declarative conjunct is used to make an assertion, an interrogative is used to ask a question, etc. But if we shift one of the conjuncts so that its content is of a different logical type, it will no longer be an appropriate object for the same kind of speech act, and we should not expect that such a speech act will be performed in using the sentence. For example, if we analyze the declarative conjunct in (15) as shifting from type t to $\langle s, t\rangle$ (so that its content/intension is function from worlds to propositions instead of a proposition), then we should have no reason to expect that a speaker who utters (15) would be using the first conjunct, *John peered into the box*, to make an assertion.

(15) John peered into the box, and what do you think he saw?

But this is wrong. Intuitively, a speaker who utters (15) first asserts that John peered into the box, then asks the addressee what they think he saw.

We might consider shifting the second conjunct, *what do you think he saw?*, instead, so that it had a denotation like a declarative; but this seems no more promising — this clause is not used to assert a proposition.

The problem is even worse in considering conditional examples like (4)b.–d., repeated here as (16):

(16) a. *If John is so smart, then why can't he answer the question?*
b. *If he tries to escape, kill him!*
c. *If John were here, what a good time we could have!*

There is no question of shifting the antecedent clause, since conditionals require declarative antecedents. The only option, then, would be to shift the consequent clause to match the logical type of the antecedent — meaning that in all these examples, the consequent clause should function like a declarative. But none of them intuitively function in this way.

3. Dynamic conjunction and unmatched clause types

The analyses sketched in Section 2 all incorporated a strategy of adjusting the treatment of the various clause types in order to somehow allow a traditional, truth-functional analysis of the connectives. In light of the difficulties these analyses faced, it seems worth exploring the opposite strategy: adopting a less traditional analysis of the connectives.

A promising version of this strategy would adopt "dynamic" analyses of the connectives, rather than static, truth-functional analyses. 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, in order 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. In Section 5, we will make some revisions in response to these problems, then develop an analysis of disjunction in Section 6.

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 complete detail.

We assume that an important part of any context of utterance is the *common ground* (CG) of the discourse participants, in something close to the sense developed in Stalnaker (1978) — the set of propositions which these participants treat as mutually shared background knowledge. Accordingly, we may represent each context c as an ordered n-tuple $\langle CG_c,\ldots\rangle$, where CG_c is the common ground of c. Asserting a proposition involves adding it to the common ground,[8] so updating a context by the assertion of p may tentatively be defined as in (17), where the three dots are filled in the same way on both sides of the equals-sign:

(17) $\langle CG_c,\ldots\rangle + p = \langle CG_c \cup \{p\},\ldots\rangle$

In a static semantic theory, we might set up our compositional rules to assign a proposition (possibly identified with a set of worlds, or its characteristic function) 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" — that is, 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 $c + p$:

[8] Of course, it will not remain in the common ground 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 common ground, rather than as a direct addition. For our current purposes, the choice between these conceptions of assertion will not matter much.

(18) If $[\![\varphi]\!]_{static} = p$, then $[\![\varphi]\!]_{dynamic} = \lambda c[c + p]$

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:

(19) If φ is uttered in c_n, then $c_{n+1} = [\![\varphi]\!]_{dynamic}(c_n)$.

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

Following Roberts (2012), let us also assume that any context of utterance will include a set of *questions under discussion* (QUD), which relates to the act of asking a question a somewhat analogous role that the common ground serves for the act of asserting a proposition.[9] We may now represent a context c as $\langle CG_c, QUD_c, \ldots\rangle$. Asking a question Q in c adds it to QUD_c, so updating a context by asking Q may be defined as in (20):

(20) $\langle CG_c, QUD_c, \ldots\rangle + Q = \langle CG_c, QUD_c \cup \{Q\}, \ldots\rangle$

In a static semantic theory, we might set up our compositional rules to assign a question (possibly identified with a function from worlds to propositions) 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, we would set up the dynamic theory to assign φ the function which mapped any context c onto $c + Q$:

(21) If $[\![\varphi]\!]_{static} = Q$, then $[\![\varphi]\!]_{dynamic} = \lambda c[c + Q]$

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 common ground results in a new common ground that entails an answer to any of the questions in the QUD, that question is removed:[10]

(22) $\langle CG_c, QUD_c, \ldots\rangle + p = \langle CG_c \cup \{p\}, QUD_c - \{Q \in QUD_c \mid \exists w\, CG_c \cup \{p\} \vDash Q(w)\}, \ldots\rangle$

---

[9] 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.

[10] The turnstile represents entailment. If p is a proposition and Γ is a set of propositions, $\Gamma \vDash p$ iff $\forall w[\forall q[q \in \Gamma \rightarrow q(w) = 1] \rightarrow p(w) = 1]]$.

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*, to whom any utterance in that context is addressed, as well as a *speaker* who performs the utterance, and a *time* at which the utterance takes place. We may now represent a context c as $\langle CG_c, QUD_c, TDL_c, ADDR_c, SPEAKER_c, TIME_c, \ldots\rangle$, 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[11] to have some property (possibly identified with a function mapping each world to a set of individuals, or its characteristic function) involves adding that property to the addressee's TDL. To formulate this operation, first let us define some notation.

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

Intuitively, $TDL_c[x + P]$ is the function just like TDL_c except that it includes P in x's to-do-list. Now we may define the updating a context by ordering P as in (20):

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

In a static semantic theory, we might set up our compositional semantic rules to assign a property (possibly identified with a function from worlds to sets) 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:

(25) If $[\![\varphi]\!]_{static} = P$, then $[\![\varphi]\!]_{dynamic} = \lambda c[c + P]$

Just as a question is removed from the QUD once an answer to it is in the common ground, a property is removed from a participant's to-do list once it is in the common ground that that participant has the property. This requires one more revision to our rule for updating a context with a proposition, for which some more notation will be useful:

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

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

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

[11] Any of these speech acts can be directly performed by using an imperative sentence. I assume they differ not in whether the ordered/requested/suggested property is added to the addressee's TDL, but in the social consequences which the addressee would incur by failing to subsequently acquire the property (thereby discharging it from their TDL). Of course these speech acts also differ in the felicity conditions governing their appropriate use, including the social relations between the speaker and addressee, etc.

What if it becomes clear that a discourse participant does not and will not have a property which is on their TDL? Much 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 even imperatives have, but since in all these other cases, the context is treated as including a set of possible sentence contents (the CG, QUD, and TDLs for the discourse participants), and the utterance of a sentence is treated as adding its content to the appropriate set, it seems appropriate to suggest that each context should also include a set of possible exclamative contents, and that using an exclamative sentence adds its content to this set. In the static analysis of Rett (2011), exclamatives are predicates of degrees; if we adopt this general approach and assume a standard intension/extension distinction, this would treat them as (extensionally) of type $\langle d, t\rangle$, with contents/intensions in type $\langle s, \langle d, t\rangle\rangle$. For example, (28)a. should have the content shown in (29)a., and (28)b. should have the content shown in (29)b.:

(28) a. What a snob John is!
b. Is John ever nosy!

(29) a. $\lambda w \lambda d$[John is a snob to degree d in w]
b. $\lambda w \lambda d$[John is nosy to degree d in w]

We should expect, therefore, that in a dynamic semantics, contexts should include a set of functions in type $\langle s, \langle d, t\rangle\rangle$ — that is to say, 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.[12] For example, an utterance of (28)a. would be an expression that there is some degree d' such that the speaker had not expected John to be a snob to d'.[13] But given all this, why would discourse participants

[12] Rett suppresses the world argument of D in her notation. Presumably, what the speaker had not expected is the proposition $\lambda w[D(w)(d')]$.

[13] Examples like (28)a. imply not only that there is a degree of snobbery which the speaker had not expected John to meet, but also that John is a snob to that degree, a fact which Rett's analysis does not seem to account for. She explains similar implications in other examples by modeling the act of exclaiming in terms of a "force" operator, whose argument is constrained to be a true proposition. However, in examples using exclamative sentences like (28)a., she treats the argument of this operator as an open proposition, not a closed proposition, so it is unclear how this constraint can be met.

collectively track a set of degree properties, and why would such an expression of surprise add a new property to the set?[14]

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 exclusively[15]) 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 to-do list, set of questions under discussion, and common ground.

More formally, we may now represent each context c as a tuple $\langle CG_c, QUD_c, TDL_c, ATR_c, ADDR_c, SPEAKER_c, TIME_c, \ldots\rangle$, 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 collectively recognize as affective triggers for x — the set of 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(Mary).[16] 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 (30):

(30) a. Wow! John is a movie star now!

---

[14] Rett deliberately develops her account of the speech act of exclaiming so that it is broad enough to encompass the exclamatory use of declarative sentences, as in (i):
(i) Wow, John won the race!
Formulating her exclamatory force operator so that it takes a proposition as its argument allows Rett to include such declarative examples in her analysis. Appealing to this same operator in the analysis of utterances of exclamative sentences forms a large part of the motivation for the wording alluded to above, "expression *that*…had not expected *that*" [emphasis added] which seems to treat both the object of expression and the object of surprise as propositional. In my view, this undermines Rett's own strategy of assigning exclamative sentences non-propositional contents; it would be better to treat the speech acts which can be directly performed by using an exclamative sentence as having a degree property as their object. Examples like (i) might then be treated as involving some other speech act, or as only indirect acts of exclamation, performed via a direct act of assertion.

[15] See the discussions in Zanuttini and Portner (2003) and Rett (2011) of examples like *What a nice house you've got!*

[16] Note that it must be the degree property which is added, and not the degree itself. Presumably, Mary would not be shocked by 7′2″ simply as an abstract degree of height, but might be shocked by it *qua* John's height.

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 illustration purposes in this paper.

Adapting our notation for to-do lists to affective trigger registries, and using D as a variable over degree properties, we may now define update with a degree property as in (31)a., and define dynamic contents for exclamative sentences in terms of static contents as in (31)b.:

(31) a. $\langle CG_c, QUD_c, TDL_c, ATR_c, ADDR_c, SPEAKER_c, \ldots\rangle + D =$
$\langle CG_c, QUD_c, TDL_c, ATR_c[SPEAKER_c + D], ADDR_c, SPEAKER_c, \ldots\rangle$
b. If $[\![\varphi]\!]_{static} = D$, then $[\![\varphi]\!]_{dynamic} = \lambda c[c + D]$

3.2. Dynamic conjunction of unmatched clause types

If a sentence has as its semantic value a function from contexts to contexts as just sketched for the dynamic options in Section 3.1, we may follow Heim (1982) and much subsequent literature, and interpret conjunction of sentences as simple function composition: [17]

(32) $[\![\textit{and}]\!] = \lambda \mathbf{p}[\lambda \mathbf{q}[\lambda c[\mathbf{p}(\mathbf{q}(c))]]]$

This gives the kind of result illustrated in (33):

(33) $[\![\textit{John arrived and Mary left}]\!] = [\![\textit{and}]\!]([\![\textit{Mary left}]\!])([\![\textit{John arrived}]\!])$
$= \lambda c[[\![\textit{Mary left}]\!]([\![\textit{John arrived}]\!](c))]$

Suppose that $[\![\textit{John arrived}]\!](c) = c'$, and that $[\![\textit{Mary left}]\!](c') = c''$. Then $[\![\textit{John arrived and Mary left}]\!](c) = c''$. Assuming the kind of analysis of declaratives sketched in Section 3.1, we may assume that c' differs from c in the addition of the proposition that John arrived to the CG, and that c'' differs from c' in the addition of the proposition that Mary left to the CG (though of course many more details would need to be made explicit in order to derive this result formally). The result of uttering *John arrived and Mary left* in context c, then, is to add the proposition that John arrived to the common ground, then add the proposition that Mary left.

This kind of interpretation for conjunction affords reasonable and intuitive results for conjunctions of unmatched clause types. For example, 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'

[17] We assume that *and* forms a syntactic unit with its right-hand conjunct. In (32), **p** and **q** are variables ranging over functions from contexts to contexts.

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. A problem with disjunction

Unfortunately, the analysis just outlined does not extend very obviously to examples involving disjunction rather than conjunction. We would not want to adopt a rule said something like "To update a context *c* with (φ *or* ψ), 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 the *leave now* property on their to-do list or adding the proposition that the speaker will shoot to the common ground; *John is crazy, or is he just clever?* would give the addressee the option of adding the proposition that John is crazy to the common ground, or adding the question whether John is clever to the set of questions under discussion. But in other examples, especially 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 common ground.

5. Look-ahead restrictions

We can obtain better results by reformulating our update rules so that they don't simply add items to the CG, QUD, or TDLs, but impose constraints on what can be added later in the discourse. For example, adding a proposition of the form $[\varphi \vee \psi]$ to the common ground constrains it so that you cannot later add anything that entails $[\neg\varphi \;\&\; \neg\psi]$ (without resulting in contradiction and provoking subtractions/corrections). Additions to the QUD or to-do lists 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, 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 — that is, a tuple of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME,\ldots\rangle$. 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, etc.; but all kinds of changes to these may occur in the course of a conversation, and perhaps we should leave out sequences that include certain kinds of transition from one tuple to another. In particular, some transitions represent corrections, repairs, backtracking, or other moves which in some sense exemplify "releases" from the constraints imposed by earlier moves, rather than conforming to them. Let us 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.[18] 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 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 common ground, anything entailed by the combination of that proposition with the existing common ground 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 common ground 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 common ground 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 general question is under discussion, its entailed sub-questions are in

[18] We may think of these non-linguistic alterations as involving the same basic operations as linguistic utterances, such as those in (20), (24), (27), and (31)a. — though these will be revised shortly.

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. We should 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, the proposition $Q(w)$ entails the proposition $Q'(w)$ — which is to say, for all w, w', if $Q(w)(w') = 1$ then $Q'(w)(w') = 1$.[19]

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 common ground does entail Q', — that is, the true complete answer to Q, in combination with the common ground, in any world entails the true complete answer to Q'. Such contextual entailments[20] should also be added to the QUD when the entailing question is. For example, if it is given in the common ground 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.

As with the addition of propositions to the common ground, 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.

By the same token, if a property is added to a discourse participant's to-do list, 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 to-do list 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 affective trigger registry, 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 intuitively, we should allow

[19] The characterization of entailment for questions here follows Groenendijk and Stokhof (1984). It should be noted that in some cases, one question may be intuitively "more specific" than another without being entailed by it according to this definition. 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" (that is, the proposition $\{w \mid$ there are dogs in the kennel in $w\}$), this does not entail an answer to (i).

(i) Are there any dachshunds in the kennel?

(ii) Are there any dogs in the kennel?

[20] The term *contextual entailment*, as applied to questions, and the general strategy for defining it, are taken from Roberts (2012).

$\lambda w\lambda d$[John is tall to d in w] to be in Mary's ATR, for example, without $\lambda w\lambda d\exists x$[$x$ is tall to d 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 work these ideas out in more detail, let us first set up some notation. Where Γ is any set of propositions: [21]

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

(35) 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 $A \vDash_\Gamma Q$ iff for all worlds w: $\Gamma \cup \{Q'(w) \mid Q' \in A\} \vDash Q$.

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 $A \vDash_\Gamma B$ iff for all individuals x: $\Gamma \cup \{\lambda w[P'(w)(x)] \mid P' \in A\} \vDash P$.

We may read '$A \vDash_\Gamma x$' as "A entails x in combination with Γ."[22]

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 (36):

(36) $Close(\langle CG, QUD, TDL,\ldots\rangle) = \langle CG \cup \{p \mid CG \vDash p\}, QUD \cup \{Q \mid QUD \vDash_{CG} Q\}, \lambda x[TDL(x) \cup \{P \mid TDL(x) \vDash_{CG} P\},\ldots\rangle$

Our update rules in (27), (20), and (24) may now be replaced with rules that apply the closure operation to the outputs of our old rules:

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

b. $\langle CG_c, QUD_c,\ldots\rangle + Q = Close(\langle CG_c, QUD_c \cup \{Q\},\ldots\rangle)$

c. $\langle CG_c, QUD_c, TDL_c, ADDR_c,\ldots\rangle + P = Close(\langle CG_c, QUD_c, TDL_c[ADDR_c + P], ADDR_c,\ldots\rangle)$

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 in (37)a., b., c., (31)a., 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*:[23]

(38) A sequence σ of tuples of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME,\ldots\rangle$ is a *development* iff for all natural numbers n such that $n, n{+}1 \in dom(\sigma)$, either:

[21] The definition in (35)a. follows the definition of "contextual entailment" for questions in Roberts (2012), with some minor adjustments.

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

[23] I assume a *sequence* is a function whose domain is a set S of natural numbers meeting the condition that (for all natural numbers 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 '$\langle a, b, c,\ldots\rangle$' represents a function which assigns 0 the value a, 1 the value b, 2 the value c, etc.

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 must be derivable from a *minimal* tuple by successive application of these same operations, where a minimal tuple is one whose common ground, questions under discussion, to-do lists, and affective trigger registries are all trivial, which is to say that the common ground contains nothing but the tautology $\lambda w[w = w]$, the questions under discussion include only the always-open $\lambda w \lambda w'[w = w']$, the to-do list for each participant includes only the universal property $\lambda w \lambda x[x = x \text{ in } w]$, and the affective trigger registry for each participant is empty. A development from a minimal tuple may be called a "cold-start" development:

(39) $Minimal(\kappa)$ 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: $ATR_\kappa(x) = \emptyset$.
(40) A sequence σ of tuples of the form $\langle CG, QUD, TDL, ATR, ADDR, SPEAKER, TIME, \ldots \rangle$ 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 common ground, questions-under-discussion set, to-do lists, or affective trigger registries, so the sequences in an actual context will not be cold-start developments. However, they should "start in the middle" of cold-start developments:

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

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

(42) For all contexts c, for all $\sigma, \sigma' \in c$, for all n such that $n \in dom(\sigma)$ and $n \in dom(\sigma')$: $TIME_{\sigma_n} = TIME_{\sigma'_n}$.

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:

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

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 (18), (21), and (25) above. Letting c be a variable ranging over contexts and letting x range over propositions, questions, and properties:

(45) If $[\![\varphi]\!]_{static} = x$, then $[\![\varphi]\!]_{dynamic} = \lambda c[c + x]$

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 common ground just those propositions which are in the common ground of the first tuple of every development in c. Its questions under discussion should include just those questions which are in the first tuple of every development in c. Its to-do-list 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 to-do list. Its affective trigger registry 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 affective trigger registry. 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 (42)), and its addressee and speaker should be the addressee and speaker at that time (which we assume to be uniquely identifiable). More formally:

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

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.

6. Disjunction

Representing contexts as in Section 5 requires no revisions to the semantics for conjunction in (32), repeated here as (47):[24]

(47) $[\![and]\!] = \lambda\mathbf{p}[\lambda\mathbf{q}[\lambda c[\mathbf{p}(\mathbf{q}(c))]]]$

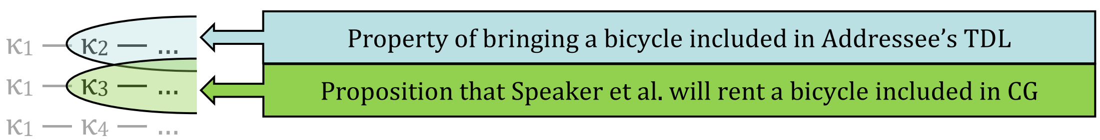
The advantage of the revisions in Section 5 comes in the treatment of disjunction. We may now interpret this via simple set union:

(48) $[\![or]\!] = \lambda\mathbf{p}[\lambda\mathbf{q}[\lambda c[\mathbf{p}(c) \cup \mathbf{q}(c)]]]$

This interpretation gives the result that (49), 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 to-do list or the proposition that a group including the speaker will rent one for \$10 per hour is in the common ground.

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

This can be visualized as in (50), where the entire diagram represents the input context, $\kappa_1,\ldots,\kappa_4$ are tuples of a CG, QUD, etc., and the blue- and green-shaded portions together represent the output context.

(50)


Note that the developments in a context — for example the output context in (50) — 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 that our new way of representing contexts distinguishes them more finely than simply using their reducts. Let c be the output context of (50). 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 tuples with p in their CG, or with P in

[24] As above, **p** and **q** here are variables ranging over functions from contexts to contexts.

x's TDL, or with the proposition that x has left in their CG. 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 common ground nor adds the question whether John is clever in the questions under discussion, 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 common ground 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 common ground includes the proposition that John is in the house (that is, the function λw[John is in the house in w]) or the proposition that Mary is in the yard (λw[Mary is in the yard in w]). By (37)a., 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 the 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:

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

Under the analysis developed so far, an utterance of (51) 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 to-do list. By (37)c., these to-do lists will also contain the disjunctive property $W \vee T$ which someone has if they have 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 (51) but before the addressee has washed the dishes, all the tuples in the context will include $W \vee T$ in the addressee's to-do list, so this property is in the addressee's to-do list 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 to-do list in this reduct, so the addressee is not specifically obliged to perform W, and not specifically required 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 (51) 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 (52), a disjunction of interrogatives, shows some similarities and some differences from the previous example:

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

An utterance of (52) 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 (52) instead of simply asking "Will John be at the party?" the speaker of (52) 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 (49), repeated here as (53):

(53) 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.

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.
d. The objects of belief (that is, the things to which we stand in the belief relation) are propositions.
e. Therefore, in a sentence of the form "α *believes* φ", the content of φ is a proposition.
f. 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 (1)d., and trying to motivate the claim that the objects of belief are context change potentials rather than propositions. I would like to suggest instead that we can retain (1)d. — indeed all of (54) — by limiting our application of dynamic semantic techniques to main clauses.[25]

Perhaps this move will raise concerns about compositionality. It does require that *The world is round* in (55)a., 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)[26]; and if *The world is round* in (55)a. and *the world is round* in (55)b. 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).[27] Main and

[25] 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 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.

[26] See the classic discussion in Emonds (1976), for example.

[27] 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.

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
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,[28] but if they do, they are limited to *Wh*-exclamatives; inversion exclamatives occur only in main clauses:[29]

(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.

Subordinate imperative clauses do not occur in English.

Let us suppose, then, that subordinate but not main clauses are headed by "staticizing" operators, which produce static denotations from dynamic ones.[30]

(59) a. $[\![STAT\text{-}DECL]\!] = \lambda\mathbf{p}[\lambda w[\forall c\forall\sigma\forall q[[\sigma \in c \;\&\; Minimal(\sigma_0) \;\&\; \forall\sigma'[\sigma' \in \mathbf{p}(c) \rightarrow q \in CG_{\sigma'_0}]] \rightarrow q(w) = 1]]]]$
b. $[\![STAT\text{-}INTERROG]\!] = \lambda\mathbf{p}[\lambda w[\lambda w'[\forall c\forall\sigma\forall Q[[Minimal(\sigma_0) \;\&\; \forall\sigma'[\sigma' \in \mathbf{p}(c) \rightarrow Q \in QUD_{\sigma'_0}]] \rightarrow Q(w)(w') = 1]]]]$
c. $[\![STAT\text{-}EXCL]\!] = \lambda\mathbf{p}[\lambda w[\lambda d[\forall c\forall\sigma\forall D[[Minimal(\sigma_0) \;\&\; \forall\sigma'[\sigma' \in \mathbf{p}(c) \rightarrow \exists x D \in ATR_{\sigma'_0}]] \rightarrow D(w)(d) = 1]]]]$

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 CG results in the set containing *nothing but* the proposition that John is a dentist and all its

[28] See the discussion in Rett (2011).
[29] This is presumably due to a general prohibition on subject-verb inversion structures in subordinate clauses; recall Footnote 27, above.
[30] 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. However, the approach developed here, with staticizing operators in subordinate clauses, will facilitate the treatment of conditionals in the next section.

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 $\lambda w[\forall c \forall \sigma \forall q[[\sigma \in c \ \& \ Minimal(\sigma_0) \ \& \ \forall \sigma'[\sigma' \in [\![John\ is\ a\ dentist]\!](c) \rightarrow q \in CG_{\sigma'_0}]] \rightarrow q(w) = 1]]]$ is just λw[John is a dentist in w] — that is, the proposition that John is a dentist.

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

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 it was already pointed out (in Section 2.2, above) that subordinate clauses of different clause types may appear 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 that coordination of main clauses of different clause types, and in fact seems limited 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 (60)a. 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 (60)b. to assign the interrogative clause *who Mary stole it from* a proposition as a semantic value (in this case, its static *ex*tension). 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 (60)c.:

(60) 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 (60)c. may be treated as ordinary truth-functional conjunction (type-shifted to apply to functions of type $\langle s, t\rangle$); no dynamics need enter the picture — at least not as a solution to the problem of unmatched clause types.[31]

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 (61)a. and (61)b. 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.

(61) 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:

(62) *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:

(63) 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 (64):[32]

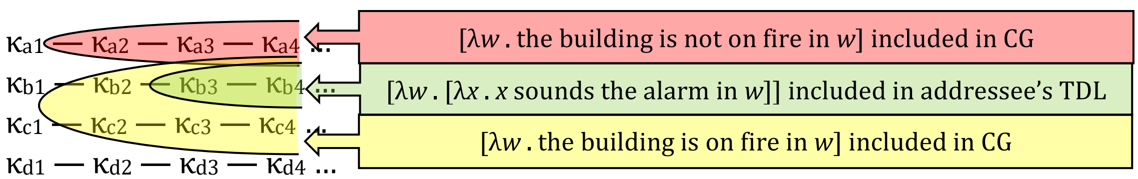
(64) $[\![if]\!] = [\lambda p\ .\ [\lambda \mathbf{q}\ .\ [\lambda c\ .\ (c + \neg p) \cup \mathbf{q}(c + p)]]]$

Put more sloppily but perhaps more comprehensibly, "*if* p, $\mathbf{q}$" 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 $\mathbf{q}$.

[31] I set aside here the issue of how to treat cases of discourse anaphora which cross the boundaries of subordinate clauses, such as the "Hob/Nob" examples of Geach (1967).

[32] Here, $\neg p$ is $\lambda w[p(w) = 0]$, and + is as in (44), above.

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 $\{\langle \kappa_{a1}, \kappa_{a2}, \kappa_{a3}, \kappa_{a4}, \ldots\rangle, \langle \kappa_{b1}, \kappa_{b2}, \kappa_{b3}, \kappa_{b4}, \ldots\rangle, \langle \kappa_{c1}, \kappa_{c2}, \kappa_{c3}, \kappa_{c4}, \ldots\rangle, \ldots\}$. 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 (65):

(65)


The output context is the union of the red- and the green-shaded areas: $\{\langle \kappa_{a2}, \kappa_{a3}, \kappa_{a4}, \ldots\rangle, \langle \kappa_{b3}, \kappa_{b4}, \ldots\rangle, \ldots\}$. 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 (64) and (59). First let us consider an example where the consequent clause (hence the conditional structure as a whole) is declarative, such as (66):

(66) If this is John's footprint, he is the murderer.

According to (59), applying *STAT-DECL* to the context change potential denoted by (66) will result in that proposition which is true in a world *w* iff, when you update a trivial CG with (66), every proposition which is in all the resulting CGs is true in *w*. According to (64), updating with (66) adds the proposition $\neg p$ that this is *not* John's footprint to some of these CGs, and adds the proposition $[p \;\&\; q]$ 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 $[p \;\&\; q]$. The most specific such proposition is $[\neg p \lor [p \;\&\; q]]$. 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 $[p \;\&\; q]$,[33] so $[p \rightarrow q]$ is the result of applying *STAT-DECL* to (66). The end result is that in an example like

[33] If it did not, there would have to be a proposition *r* entailed by $\neg p$ and entailed by $[p \;\&\; q]$, 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 false. But any world where $[p \rightarrow q]$ is true is one where either $\neg p$ is true, or one where $[p \;\&\; q]$ is true, and both of these entail *r*.

Mary believes that if this is John's footprint, he is the murderer, it is the material conditional [*this is John's footprint* → *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 (64) 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 (59)b. to the context change potential denoted by (67):

(67) 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 (67), 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 (67) 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 (67).

By (64), updating a context with (67) 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 (67), hence whether there will be any question Q which entails all such questions as required.

The answer is yes, because our update operations in (37) close CGs, QUDs, and TDLs under contextual entailment. Contextual entailment for questions is defined in (35)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 $\vDash_{CG}$ 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 GC. 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 (68):[34]

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

[34] Perhaps some readers will wonder why the dynamic account of *if* given in (64) is necessary, if an operator like $\rightarrowtail$ 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 (63), where the consequent is a coordinate structure with unmatched clause types as the conjuncts.

For example, where p is λw[Mary is innocent in w] (the proposition that Mary is innocent), and Q is $\lambda w \lambda w'[\lambda x[x$ is the murderer in $w] = \lambda x[x$ is the murderer in $w']]$ (the question of who the murderer is), $[p \rightarrowtail Q]$ will be the question in (69):

(69) $\lambda w \lambda w'$[Mary is innocent in $w \rightarrow \lambda x[x$ is the murderer in $w] = \lambda x[x$ 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 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 QUD $\vDash_{CG}$ (69)? Yes, because any world where all the propositions in CG are true will be one in which the antecedent of the conditional in (69) 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 QUD $\vDash_{CG}$ (69). Since in this case QUD contains the question of who the murderer is ($\lambda w \lambda w'[\lambda x[x$ is the murderer in $w] = \lambda x[x$ is the murderer in $w']]$), and any pair of worlds satisfying this question will also render the consequent of the conditional in (69) true, thereby satisfying (69) as a whole, we can see that any pair of worlds satisfying all the questions in QUD will also satisfy (69).

We have now seen that updating a trivial context with (67) will result in a context in which the head of every sequence has (69) in its QUD. To show that (69) is the result of applying *STAT-INTERROG* to the context change potential denoted by (67), we need now only show that (69) entails all other questions which are in those same QUDs. Suppose updating a minimal sequence with (67) results in a sequence whose head has some question Q in its QUD. This means that either $\lambda w \lambda w'[w = w'] \vDash_{\lambda w[\text{Mary is not innocent in } w]} Q$ or $\lambda w \lambda w'[\lambda x[x$ is the murderer in $w] = \lambda x[x$ is the murderer in $w']] \vDash_{\lambda w[\text{Mary is innocent in } w]} Q$. If it were the case that Q is *not* entailed by (69), there would have to be a pair of worlds which satisfy (69) but do not satisfy Q. But any pair of worlds w, w' satisfying (69) will either be such that Mary is not innocent in w, or such that Mary is innocent in w and $\lambda x[x$ is the murderer in $w] = \lambda x[x$ is the murderer in $w']$. In either case, w, w' must satisfy Q. Therefore (69) is the result of applying *STAT-INTERROG* to the context change potential denoted by (67).

9. Conclusion

Coordinate and conditional structures built by connected 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 quantification and anaphora; but such exploration must be a topic for a different paper.

References

Emonds, Joseph E. (1976) *A Transformational Approach to English Syntax: Root, Structure-Preserving, and Local Transformations*. Academic Press

Frege, Gottlob (1892) 'Über Sinn und Bedeutung', *Zeitschrift für Philosophie und philosophische Kritik* 100.1.25–50

Geach, Peter Thomas (1967) 'Intentional Identity', *Journal of Philosophy* 64.20.627–632

Groenendijk, Jeroen and Martin Stokhof (1982) 'Semantic Analysis of Wh-Complements', *Linguistics and Philosophy* 5.2.175–234

Groenendijk, Jeroen and Martin Stokhof (1984) *Studies on the Semantics of Questions and the Pragmatics of Answers*, University of Amsterdam dissertation.

Hamblin, C.L. (1973) 'Questions in Montague English', *Foundations of Language* 10.1.41–53

Hausser, Roland R. (1980) 'Surface Compositionality and the Semantics of Mood', *Speech Act Theory and Pragmatics*, Searle, J. R., et al. (ed.), pp. 71–95. Springer Netherlands

Heim, Irene (1982) *The Semantics of Definite and Indefinite Noun Phrases*. Graduate Linguistics Students Association, University of Massachusetts

Karttunen, Lauri (1977) 'Syntax and Semantics of Questions', *Linguistics and Philosophy* 1.1.3–44

Levinson, Stephen C. (1983) *Pragmatics*. Cambridge University Press

Lewis, David K. (1972) 'General Semantics', *Semantics of Natural Language*, Davidson, D. and G. H. Harman (ed.), pp. 169–218. D. Reidel Publishing Co

Partee, Barbara H. and Mats Rooth (1983) 'Generalized Quantifiers and Type Ambiguity', *Meaning, Use, and Interpretation of Language*, Bäuerle, R., et al. (ed.), pp. 361–383. Walter de Gruyter

Portner, Paul (2004) 'The Semantics of Imperatives within a Theory of Clause Types', *Proceedings from Semantics and Linguistic Theory Xiv*, Young, R. (ed.), pp. 235–252. CLC Publications

Rett, Jessica (2011) 'Exclamatives, Degrees and Speech Acts', *Linguistics and Philosophy* 34.5.411–442

Roberts, Craige (2012) 'Information Structure: Towards an Integrated Formal Theory of Pragmatics', *Semantics and Pragmatics* 5.6.1–69

Ross, John Robert (1970) 'On Declarative Sentences', *Readings in English Transformational Grammar*, Jacobs, R. A. and P. S. Rosenbaum (ed.), pp. 222–272. Georgetown University Press

Sadock, Jerrold M. (1974) *Toward a Linguistic Theory of Speech Acts*. Academic Press

Stalnaker, Robert (1978) 'Assertion', *Pragmatics*, Cole, P. (ed.), pp. 315–332. Academic Press

Villalba, Xavier (2008) 'Exclamatives: A Thematic Guide with Many Questions and Few Answers', *Catalan Journal of Linguistics* 7.9–40

Zanuttini, Raffaella and Paul Portner (2003) 'Exclamative Clauses: At the Syntax-Semantics Interface', *Language* 79.1.39–81