

# Crossover and accessibility in dynamic semantics\*

Patrick D. Elliott  
Massachusetts Institute of Technology

July 13, 2020

[Chierchia](#) develops a theory of Weak Crossover effects couched in terms of dynamic semantics, the central idea being that discourse referents are *only* introduced in thematic positions. Intrasentential anaphora proceeds via standard dynamic mechanisms, thus grounding wco in an independently motivated notion of *accessibility*. While conceptually appealing, I show that [Chierchia](#)'s account is hamstrung by a central design feature of first-generation dynamic theories — the *external staticity* of negation and related operators. This gives rise to the (I argue, erroneous) prediction that any expression that can outscope negation for the purposes of intersentential anaphora can obviate Weak Crossover. To address this issue, I suggest that we need a dynamic semantics in which operators such as negation *selectively* destroy *indeterminate* discourse referents, but leave *determinate* discourse referents unscathed. I develop a version of monadic dynamic semantics ([Charlow 2014, 2019](#)) which accomplishes just this, by tweaking the semantics of indefinites. Perhaps surprisingly, the resulting system is in some respects simpler than first-generation dynamic theories — negation will end up being classical. As an added bonus, the resulting system validates double-negation elimination — a recalcitrant problem for first-generation theories.

## 1 Introduction

In a recent article, [Chierchia \(2020\)](#) purports to explain Weak Crossover (wco) as a theorem of Dynamic Semantics (DS) and restrictions governing introduction of Discourse Referents (DRs). [Chierchia](#)'s account of wco is striking in its elegance, resting as it does on independently motivated mechanisms for analyzing anaphoric dependencies, generalized to the intersentential domain. In this short paper, I'll demonstrate that [Chierchia](#)'s account of wco has (at least) one major flaw — it predicts that quantificational scope *never* feeds dynamic scope, and therefore

---

\*I'm extremely grateful to Keny Chatain, Simon Charlow, Filipe Hisao Kobayashi and Yasu Sudo for their patient and thoughtful feedback, and of course Gennaro Chierchia whose work continues to be a source of inspiration. I'd also like to thank participants in the Spring 2020 MIT *topics in semantics* seminar, and my presentation of [Chierchia \(2020\)](#) at the MIT *LF Reading Group*.

that scoping a Quantificational Phrase (QP) over an externally static operator, such as negation, should fail to feed DR introduction. This is shown to make a number of bad predictions, casting the enterprise of explaining wco in terms of DS in serious doubt. Nevertheless, the idea is worth salvaging, and in the following I'll show that it *can* be salvaged. This will necessitate rethinking the dynamic plumbing underlying the account, and this will have consequences for both the semantics of indefinites and negation in DS. Concretely, I'll argue that, contrary to received wisdom, negation is in fact externally dynamic, but only with respect to DRS introduced by *definites*. As well, as reviving Chierchia's theory as a plausible account of wco (the main goal of this paper), the resulting theory will also be shown to independently address a recalcitrant problem for DS — double negation.

wco (Postal 1971, and subsequent work) has long been a thorn in the side of theories of the syntax-semantics interface, demonstrating poorly understood restrictions on the interaction between scope and binding. An instantiation of the wco paradigm is given in (1). In order to understand the problem, consider that any theory of scope needs to explain why QPs interpreted in places other than where they're pronounced, in order to account for, e.g., inverse scope readings. A popular approach to this question in the generative literature is Quantifier Raising (QR) (May 1977). If such an operation is independently necessary, then why can't scoping the QP feed binding in (1b)?<sup>1</sup>

- (1) a. Every boy<sup>1</sup> loves his<sub>1</sub> mother.  
 b. \* His<sub>1</sub> mother loves every boy<sup>1</sup>.

It's tempting to conclude that scope simply can't feed binding, but the empirical landscape is more nuanced than this, as has long been recognized (see, e.g., Ruys 2000). For example, QPs can bind out subjects (2a), and into right-adjoined VP modified (2b).

- (2) a. Every boy<sup>1</sup>'s mother loves him<sub>1</sub>.  
 b. Mattie petted every kitten<sup>1</sup> against it<sub>1</sub>'s will.

Chierchia's account side-steps this issue by adopting an account on which anaphora proceeds from left-to-right, and potentially to non-c-commanding positions — Dynamic Semantics (DS). The central move is to claim that the binder is always the *thematic* position of the QP, but, by dint of how anaphora is licensed in DS, the thematic position of the binder must be to the left of the pronoun.

In order to understand how Chierchia's theory works in detail, and ultimately in order to understand its pitfalls, it will be useful to first go through some basic design features of DS.

---

<sup>1</sup>Of course, there are also many other approaches to scope-taking which *don't* involve movement, such as continuation semantics (Barker 2002, Barker & Shan 2014) — the reasoning outlined here applies equally to such theories.

## 2 The dynamics of crossover

### 2.1 Dynamic semantics: the basics

DS (Heim 1982, Groenendijk & Stokhof 1991, a.o.) purports to account for linear asymmetries in the domain of anaphora, as exemplified by the following contrast.<sup>2</sup>

- (3) a. A woman<sup>1</sup> walked in. She<sub>1</sub> sat down.  
b. \* She<sub>1</sub> sat down. A woman<sup>1</sup> walked in.

Assuming that the indefinite takes scope within its containing sentence, first-generation dynamic theories account for the contrast in (3) by building an Input/Output (I/O) asymmetry into semantic composition. In first generation dynamic theories, such as Groenendijk & Stokhof's Dynamic Predicate Logic (DPL), the asymmetry is precompiled into the entry for the sentential sequencing operator. In order to see how this works, it will be useful to start laying out some details.<sup>3</sup> In DS, sentences denote *dynamic propositions* — functions from assignments to sets of assignments. We'll abbreviate the type of a dynamic proposition as  $\top$  (4).

- (4) Dynamic propositional type (def.)<sup>4</sup>  
 $\top := g \rightarrow \{g\}$

Indefinites induce, in the dynamic parlance, *random assignment* — that is to say that the sentence takes an input assignment  $g$  and outputs a set of modified assignments. This is shown in (5).<sup>5</sup>

- (5) Indefinites in DS  
 $\llbracket \text{A woman}^1 \text{ walked in} \rrbracket = \lambda g . \{g^{[1 \rightarrow x]} \mid \text{woman } x \wedge \text{walked-in } x\}$   $\top$

We can illustrate the effect of random assignment by considering the application of (5) to  $g_\emptyset$ .<sup>6</sup>

- (6) Context: *The women are Shirley, Britta, and Annie.*  
 $\llbracket \text{a woman}^1 \text{ walked in} \rrbracket g_\emptyset = \{[1 \rightarrow \text{shirley}], [1 \rightarrow \text{britta}], [1 \rightarrow \text{annie}]\}$   $\{g\}$

<sup>2</sup>Throughout this paper, we use superscript indices for the binder, and matching subscript indices for the expression(s) it binds.

<sup>3</sup>The system outlined here is roughly equivalent to DPL.

<sup>4</sup>We write  $\{a\}$  for the type of a set of values of type  $a$ .

<sup>5</sup>Technically speaking, we'll assume that assignments are *partial* — that is to say, given a stock of variables  $\mathbb{N}$ , an assignment is a function whose domain is a subset of  $\mathbb{N}$ , and whose codomain is the set of individuals. This means that a given assignment  $g$  may be *undefined* for a variable  $n$ .

We write  $g^{[n \rightarrow x]}$  is the assignment that is just like  $g$ , other than mapping  $n$  to  $x$  — we assume that it is defined iff  $g_n$  is *undefined* (Heim's 1991 *novelty condition*).

<sup>6</sup>In order to understand the contribution of a dynamic proposition, it can be useful to apply it to the unique initial assignment  $g_\emptyset$ , whose domain is the empty set. The initial assignment represents an idealized state in which no DRS have been introduced.

Pronouns, on the other hand, induce a simple form of environment-sensitivity, as shown in (7). The consequences of feeding a sentence with a pronoun (a) an assignment which passes the test, and (b) an assignment which fails the test are shown in the two respective examples in (8).

(7) Pronouns in DS

$$\llbracket \text{She}_1 \text{ sat down} \rrbracket = \lambda g . \begin{cases} \{g\} & \text{sat-down } g_1 \\ \emptyset & \text{otherwise} \end{cases} \quad \top$$

(8) Context: *only Shirley sat down*

$$\begin{aligned} \text{a. } & \llbracket \text{She}_1 \text{ sat down} \rrbracket [1 \rightarrow \text{shirley}] = \{[1 \rightarrow \text{shirley}]\} & \{g\} \\ \text{b. } & \llbracket \text{She}_1 \text{ sat down} \rrbracket [1 \rightarrow \text{annie}] = \emptyset & \{g\} \end{aligned}$$

Note that a direct consequence of the definitions given here is that a sentence with a pronoun is never defined for the initial assignment (Heim 1991's 1991 *familiarity* condition). In order for pronouns to co-vary with indefinites in previous sentences, it is standard to define a dynamic sequencing operator ( $;$ ), which threads dynamic propositions together by feeding the outputs of the first sentence into the second, pointwise, and gathering up the results.

(9) Dynamic sequencing (def.)

$$p ; q := \lambda g . \bigcup_{g' \in p g} \{g'' \mid g'' \in q g'\} \quad ; : \top \rightarrow \top \rightarrow \top$$

It follows from the definition of dynamic sequencing that the input of the second sentence can be sensitive to the output of the first, but not vice versa. We can see how this works by sketching the result of interpreting (3a) via dynamic sequencing, and applying the result to the initial assignment.

(10) Context: *Three women walked in and sat down: Shirley, Britta and Annie*

$$\begin{aligned} \llbracket (3a) \rrbracket &= (\llbracket \text{A woman}^1 \text{ walked in} \rrbracket ; \llbracket \text{she}_1 \text{ sat down} \rrbracket) g_\emptyset & \{g\} \\ &= \bigcup_{g' \in \llbracket \text{a woman}^1 \text{ walked in} \rrbracket g_\emptyset} \llbracket \text{she}_1 \text{ sat down} \rrbracket g' \\ &= \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow \text{shirley}] \\ &\quad \cup \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow \text{britta}] \\ &\quad \cup \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow \text{annie}] \\ &= \{[1 \rightarrow \text{shirley}], [1 \rightarrow \text{britta}], [1 \rightarrow \text{annie}]\} \end{aligned}$$

In the dynamic parlance we say that, by dint of how dynamic sequencing is defined, in a sentence of the form  $p ; q$ ,  $p$  is *accessible* to  $q$ , but not vice-versa. Chierchia (2020) accounts for wco by generalizing this notion of accessibility to sentence-internal positions. Another variant of the wco paradigm given below. The generalization, at first blush, is that a pronoun may only be bound by an expression  $X$ , if the thematic position of  $X$  is to the left of the pronoun.

- (11) a. Which girl<sub>1</sub> \_\_\_ loves her<sub>1</sub> father?  
 b. \* Which girl<sub>1</sub> does her<sub>1</sub> father love \_\_\_?

Chierchia’s account of wco is founded on the assumption that semantic composition is dynamic *all the way down*, and therefore exhibits left-to-right asymmetries sentence internally. Concretely, in Chierchia’s system, *theme* positions will be accessible to *agent* positions, but not vice versa. Where Chierchia departs significantly from DS is in the conjecture that DRS may be introduced *only* at thematic positions; really, what is doing the binding in (11) is the trace position, and coupled with a sentence-internal notion of accessibility, wco follows straightforwardly.

In DS, there are certain *externally static* expressions, such as negation, which are known to induce inaccessibility. As we’ll see, Chierchia’s account will face a major obstacle once we examine in detail how such expressions interact with the basic theory of crossover described above. Before laying out the issue, we’ll first turn to a consideration of Chierchia’s system in more detail.

## 2.2 Chierchia’s dynamic event semantics

Chierchia assumes a version of DS roughly equivalent to the one outlined in the previous section. In order to get the notion of accessibility to do some work in the sentence-internal domain, Chierchia adopts a view of semantic composition that is fundamentally *conjunctive* — *neo-Davidsonian event semantics* (Castañeda 1967). According to neo-Davidsonian event semantics, verbs uniformly denote (monadic) predicates of events, participants are related to events via *thematic functions*,<sup>7</sup> and sentences are existential statements about events. This is illustrated for an non-argument-taking predicate in (12a) and a predicate with more complex argument structure in (12b).

- (12) a.  $\llbracket \text{it rained} \rrbracket = \exists e[\text{rain } e]$   
 b.  $\llbracket \text{Shirley gave Jeff the book} \rrbracket = \exists e \left[ \begin{array}{l} \text{agent } e = \text{shirley} \\ \wedge \text{ theme } e = \text{the-book} \\ \wedge \text{ goal } e = \text{jeff} \end{array} \right]$

Following Chierchia, we can translate this basic setup into a dynamic setting fairly straightforwardly. In order to do so, we’ll first define a helper function *dynamic lift* which shifts a truth value into the corresponding dynamic *test*.<sup>8</sup>

- (13) Dynamic lift (def.)  

$$t^\uparrow := \lambda g. \begin{cases} \{g\} & t \\ \emptyset & \text{otherwise} \end{cases} \quad \uparrow: \mathbb{T} \rightarrow \mathbb{T}$$

<sup>7</sup>The literature sometimes diverges as to whether thematic roles are modelled as relations or functions. This distinction will be immaterial for our purposes.

<sup>8</sup>A test is a dynamic proposition that, given as input assignment  $g$ , either returns  $\{g\}$  or  $\emptyset$ .

N.b. that our decomposition differs somewhat from the exact presentation in Chierchia (2020), but the results are equivalent.

With dynamic lift in hand, predicates of events can be shifted into functions from individuals to dynamic sentential meanings using the following recipe:<sup>9</sup>

(16) Dynamically lifting a verbal predicate

$$\begin{aligned} \llbracket \text{rain} \rrbracket^\uparrow &:= \lambda e . (\llbracket \text{rain} \rrbracket e)^\uparrow && v \rightarrow T \\ &= \lambda e g . \begin{cases} \{g\} & \text{rain } e \\ \emptyset & \text{otherwise} \end{cases} \end{aligned}$$

In a static setting, thematic functions take individuals and return event predicates. These can be lifted into the current setting using an analogous recipe:<sup>10</sup>

(17) Dynamically lifting a thematic function

$$\begin{aligned} \text{agent}^\uparrow &:= \lambda x e . (\text{agent } x e)^\uparrow && e \rightarrow v \rightarrow T \\ &= \lambda g x e . \begin{cases} \{g\} & \text{agent } e = x \\ \emptyset & \text{otherwise} \end{cases} \end{aligned}$$

We also need a composition rule for gluing together dynamic event predicates — *dynamic predicate modification*. This is simply an instantiation of dynamic sequencing, generalized in an analogous way to Partee & Rooth’s (1983) generalized conjunction.

(18) Predicate modification in dynamic event semantics (def.)

$$f ; g := \lambda e . f e ; g e \quad (v \rightarrow T) \rightarrow (v \rightarrow T) \rightarrow v \rightarrow T$$

Finally, we need a way of closing off the event variable, in order to get back a sentential meaning. In event semantics, this is typically done via an operation of existential closure. A dynamic version of this operation is defined below.

---

<sup>9</sup>This recipe takes advantage of the fact that  $((\rightarrow) v)$  is a *functor* in category theoretic terms, which we can characterize via the following type constructor:

(14)  $\text{Reader}_v$  (def.)

$$R_v a := v \rightarrow a$$

Since  $R_v$  is a functor, we can define a structure-preserving operation that maps a function into the functorial structure introduced by  $R_v$ .

(15)  $\text{map } f := \lambda m e . f (m e) \quad \text{map} : (a \rightarrow b) \rightarrow (v \rightarrow a) \rightarrow (v \rightarrow b)$

Our recipe for converting a verbal predicate into a dynamic verbal predicate, simply involves mapping dynamic lift into the verbal predicate.

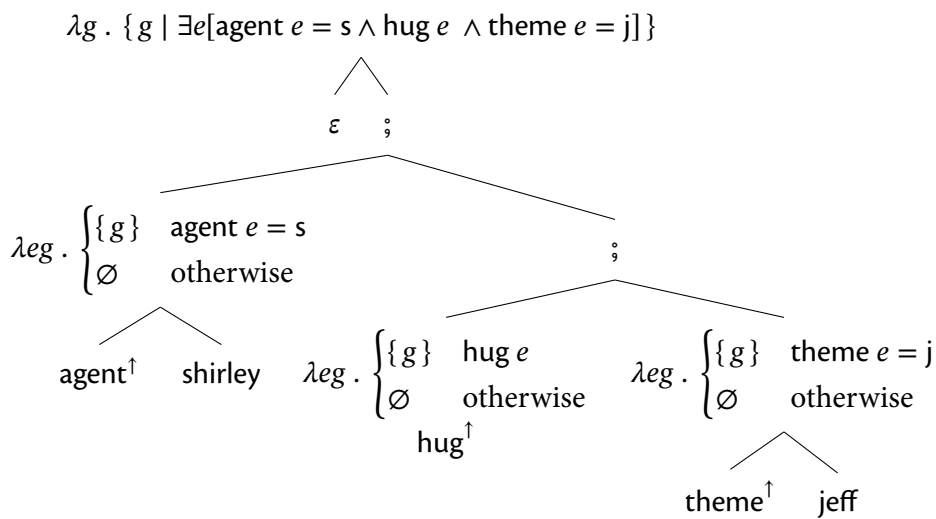
<sup>10</sup>Similarly, this recipe takes advantage of the fact that in addition to  $((\rightarrow) v)$ ,  $((\rightarrow) e)$  is also a functor. A reconstruction of the details is left to the reader.

(19) Existential closure in dynamic event semantics

$$\varepsilon m := \lambda g . \bigcup_{e \in D_v} m e g \qquad \varepsilon : (v \rightarrow T) \rightarrow T$$

Before we discuss expressions with dynamic import — indefinites, pronouns, etc. — let’s first consider an example derivation, to illustrate how semantic composition proceeds in Chierchia’s dynamic event semantics. The verbal predicate is lifted into a *dynamic* predicate, and similarly, lifted thematic functions applied to their arguments return dynamic predicates. The verb and its arguments thus compose via dynamic predicate modification, and the resulting dynamic predicate is existentially closed, returning a dynamic proposition. Since no DRS are introduced here, the result is a test.

Figure (1): “Shirley hugged Jeff”



We’ve shown how Chierchia systematically lifts neo-Davidsonian event semantics into a dynamic setting. It should already be reasonably clear that, given how the system is set up, accessibility relations between thematic positions automatically fall out which mirror their linear order, i.e., the agent phrase is accessible to the phrase, but not vice versa. In order to illustrate the account of wco it will only be necessary to discuss the semantics of indefinites and pronouns. Departing slightly from Chierchia here, are defined such that they may compose directly with thematic functions.

(20)  $\text{pron}_n := \lambda r . \lambda e . \lambda g . r g_n e g \qquad (e \rightarrow v \rightarrow T) \rightarrow v \rightarrow T$

Indefinites on the other hand denote functions from dynamic predicates to dynamic propositions. One important thing to note here is that they are *not* attributed any inherent anaphoric potential. By dint of its type,<sup>11</sup> it is clear that the indefinite is a scope-taker; they may not compose with thematic functions *in-situ*, but rather must be scoped out.

<sup>11</sup>Any expression of type  $(a \rightarrow b) \rightarrow b$  may take scope at a type  $b$  node, leaving behind a trace of type  $a$ .



$$(21) \quad \llbracket \text{some boy} \rrbracket := \lambda k g . \bigcup_{\text{boy } x} k x g \quad (e \rightarrow T) \rightarrow T$$

So just how do DRS get introduced on this set-up, if not by indefinites? The most substantial contribution of Chierchia’s paper is the Dynamic Predication Principle (DPP). A direct consequence of the DPP is that only verbs, and (more relevantly for our purposes) thematic functions may introduce DRS.<sup>12</sup>

$$(22) \quad \text{The Dynamic Predication Principle (DPP)} \\ \text{DRS can only be introduced by predicates.} \quad (\text{Chierchia 2020: p. 32})$$

This is cashed out in the formal system via a superscript operator that applies to a thematic function, introducing a *determinate* DR relative to its individual argument. Note that it’s easy to define a version of the superscript operator that may apply to an individual or a quantifier, and therefore it’s a crucial claim of the theory that such operators don’t exist.

$$(23) \quad \text{DR introduction (def.)} \\ r^n := \lambda x e g . r x e g^{[n \rightarrow x]} \quad (e \rightarrow v \rightarrow T) \rightarrow e \rightarrow v \rightarrow T$$

Some terminological clarification will be useful at this point: we say that a dynamic proposition *introduces a DR*  $n$  if it is undefined for any input defined at  $n$ , and its outputs are defined at  $n$ . We say that  $n$  is a *determinate* DR if every assignment in the output agrees on  $n$  (assuming they are defined at  $n$ ), and  $n$  is an *indeterminate* DR if the output assignments *disagree* at  $n$  (again, assuming they are defined at  $n$ ). In classical DS, indefinites introduce indeterminate DRS by triggering random assignment. In Chierchia’s system,  $x^n$  is guaranteed to introduce a determinate DR, since output assignments map  $n$  to the same individual:  $x$ .

We’re now in a position to understand Chierchia’s theory of wco. Due to the DPP, only thematic positions may introduce DRS — scope never feeds DR-introduction. Anaphora therefore always respects the accessibility hierarchy initially established in the thematic core of the sentence. The classic dynamic entry for an indefinite is, in essence, decomposed into two components — the base position is responsible for introducing the DR, and at the scope site, the indefinite induces non-determinism by feeding in a set of alternatives. A derivation involving successful binding is given in figure (2), and a wco derivation is given in figure (3). In the latter instance, binding is unsuccessful due to the accessibility relations set up in the thematic core of the sentence, which may not subsequently be disrupted.

<sup>12</sup>In order to account for, e.g., interactions A-movement and wco, the generalization in (22) ultimately must be extended in an apparently *ad-hoc* fashion. Since we’re not concerned with A-movement here, the original version of the generalization will be sufficient for our purposes.

I’m currently working on an alternative to Chierchia’s theory of wco (Elliott in progress), which aims to account for the A/A’ distinction for the purposes of wco in a more principled fashion — the idea, in brief, is that whereas A’-movement involves *bona fide* scope and phonological displacement, A-movement is cashed out as null operator movement — A-moved expressions are in fact base-generated in their landing sites. Once married with a dynamic account of wco, the distinction between A/A’ movement falls out automatically; as an added bonus, *improper movement* becomes unrepresentable.



Figure (2): A girl  $\_1$  loves her $_1$  mother

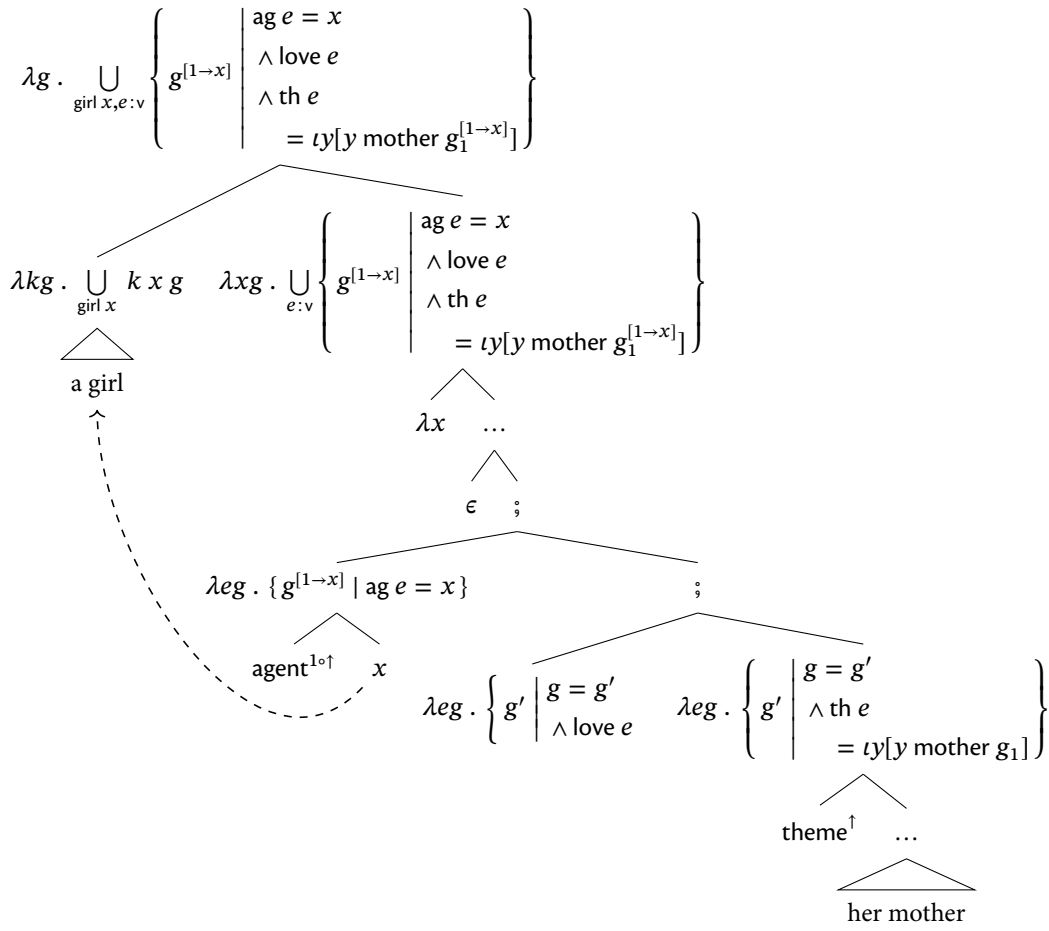
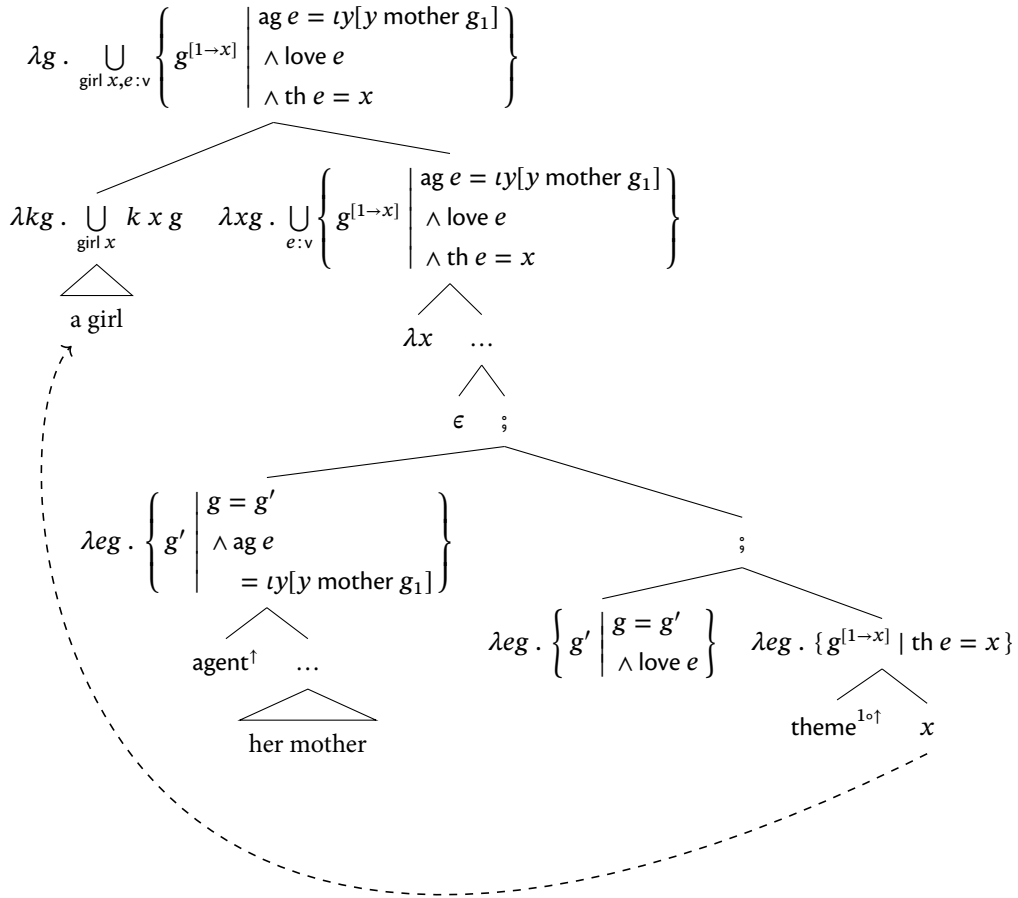


Figure (3): A girl her<sub>1</sub> mother loves \_\_<sup>1</sup>



One aspect of Chierchia’s account which is crucial, yet somewhat glossed over in the paper, is that, assuming a QR perspective, a QP cannot bind its trace via dynamic mechanisms, as this is incompatible with the DPP. Rather, there must be a syntactically distinct stock of variables responsible for trace-binding.

Putting this conceptual issue to one side, Chierchia’s explanation for wco is rather neat, and arguably worth preserving; taking as it does independently motivated mechanisms for anaphora, and generalizing them to the subsentential domain. As discussed in detail in the paper, this account has the further advantage of straightforwardly accounting for binding into adjuncts, without recourse to shell-like descending structures.

(24) John [[loves every cat<sup>1</sup>] [against it<sub>1</sub>'s will]].

Chierchia 2020: 43

As discussed by Chierchia this follows straightforwardly from the system as laid out here, on the assumption that the adjunct is a right-adjoined dynamic event predicate. Since composition proceeds via discourse sequencing, it follows that the *theme* is accessible to adjunct for the purposes of anaphora.

In the next section, we'll introduce the central problem, as we see it, for Chierchia's theory of wco, which arises by dint of how negation functions in DS.

### 3 An accessibility issue

The problem for Chierchia's account is a simple one — it can be observed that certain operators, such as negation, render their arguments inaccessible for the purposes of anaphora. This is illustrated in (25) — we use a Negative Polarity Item (NPI) to ensure that the expression takes scope below negation.

(25) \*It's not true that any man<sup>1</sup> walked in. He<sub>1</sub> sat down.  $\neg > \exists$

If an indefinite takes scope *above* negation, however, anaphora is possible, even if its surface position is below negation. This is illustrated in (26) — in fact, anaphora *disambiguates* in favor of the wide scope reading. This would seem to suggest that the indefinite may introduce a DR at its scope site, rather than at its thematic position, in conflict with the DPP.

(26) It's not true that a man<sup>1</sup> walked in. He<sub>1</sub> sat down.  $\exists > \neg$

This however subverts Chierchia's explanation for wco — there are two possible responses, if the indefinite is permitted to introduce a DR at its scope site, then we correctly account for the interaction with negation, but predict that such expressions should always obviate wco. Chierchia suggests something along these lines for specific indefinites, but we'll argue that it can't be correct in the general case. Alternatively, one could maintain the DPP, in which case the interaction with negation remains puzzling. We'll ultimately argue that a version of Chierchia account of wco can be salvaged, but this requires us to rethink the basic dynamic foundations upon which the system is built.

In order to understand exactly why this issue arises, we'll consider the status of negation in DS in somewhat more detail.

#### 3.1 Negation as a destructive operation

In order to account for the paradigm in (25) and (26), the standard move in DS is to adopt an *externally static* entry for negation — i.e., to define negation in such a way that it is guaranteed to

return a test.<sup>13</sup> The definition in (53) accomplishes this. Even if a DR is introduced in  $m$ , not  $m$  can only ever return  $\{g\}$  or  $\emptyset$ , for a given input  $g$ .

(27) Externally static negation (def.)

$$\text{not } m := \lambda g . \begin{cases} \{g\} & m \ g = \emptyset \\ \emptyset & \text{otherwise} \end{cases} \quad \mathbb{T} \rightarrow \mathbb{T}$$

If we consider the Logical Form (LF) for a sentence involving an indefinite taking wide quantificational scope over negation, as in (28), we can see exactly how the problem will arise — the thematic function is responsible for introducing the DR, and since the thematic function is within the scope of negation, it in effect doesn't matter where the indefinite takes scope with respect to negation — the DR is always going to get wiped out.

(28) Some man  $\lambda x$  [not [ $x^1$  walked in]]

$$\llbracket \text{some man} \rrbracket (\lambda x g . \{g' \mid g' = g \wedge \{g^{[1 \rightarrow x]} \mid \text{walked-in } x\} = \emptyset \})$$

Since negation is fundamentally a *destructive* operation in DS, we can't tweak the entry for the indefinite in order to “resurrect” the DR introduced by the thematic function. The only option is to assume that indefinites can introduce DRS at their scope sites. We can modify the denotation of an indefinite as follows in order to cash this out (this is simply the standard semantics for indefinites in first-generation dynamic theories such as DPL):

$$(29) \llbracket \text{some man}^1 \rrbracket := \lambda k g . \bigcup_{\text{man } x} k \ x \ g^{[1 \rightarrow x]} \quad (\mathbb{e} \rightarrow \mathbb{T}) \rightarrow \mathbb{T}$$

Naturally, this gives rise to the prediction that, if an indefinite can outscope negation, it obviates wco. This prediction has some initial plausibility since, as pointed out by Chierchia (2020: 49), specific indefinites, which generally take wide scope over negation, seem to ameliorate wco, as illustrated by the examples below (judgments due to Chierchia 2020):

(30) **His<sub>1</sub>** father hates **{ a boy<sup>1</sup> I know | a friend<sup>1</sup> of mine | a certain boy<sup>1</sup> }**

This response will however not work in the general case. Much like indefinites, *wh-expressions* are externally dynamic, as illustrated by the sentence in (31):

(31) I know [**which boy<sup>1</sup>** Mary invited to the party and why **he<sub>1</sub>** was invited].

Furthermore, scoping a *wh-expression* over negation feeds anaphora, as illustrated by example (32). In order to account for this example, we'd need to pursue the same strategy as for indefinites, and allow *wh-expressions* to introduce DRS. It's fairly clear however that this makes extremely bad predictions for wco, essentially predicting that *wh-expressions* should never give rise to wco effects.

<sup>13</sup>Negation is not the only expression which is taken to be externally static — non-existential quantifiers, also, are typically taken to be externally static. We'll come back to those later.

(32) I know **which boy<sup>1</sup>** Mary didn't **invite \_\_\_ to the party**, and why **he<sub>1</sub>** was excluded.

The problem here does not exclusively effect the account of wco for expressions such as indefinites and *wh*-expressions. For other scope takers too, just so long as the thematic position is in the scope of negation, any DRS introduced will be rendered inaccessible.<sup>14</sup> We can construct an example in which this predicts the impossibility of a universal binding into an adjunct, just so long as the adjunct takes scope above negation. (33) has a salient reading which can be paraphrased as follows: *Each puppy is such that, after it peed on the carpet, John decided not to keep it..* This involves negation in an embedded clause, with the adverbial attaching high. The universal, in turn, takes scope over both the embedding verb and the adverbial.

(33) John [decided [not **to keep each of these puppies<sup>1</sup>**] [after **it<sub>1</sub>** peed on the carpet]].  
 $\forall > \text{decide} > \neg$

Surprisingly, **Chierchia** predicts that anaphora should in fact be impossible here. This is because a DR is introduced in the position corresponding to the *theme* of *keep*, rather than at the scope site of the universal. This DR should be wiped out by embedded negation prior to attachment of the adverbial clause.

Note that this problem isn't specific to negation — we can create examples illustrating the same point using other externally static operators, such as (non-indefinite) determiners. Like negation, determiners fail to destroy DRS introduced in a lower thematic position. Rather, the *dynamic* scope of the indefinite is at least as high as its quantificational scope.

(34) Most girls **danced with a boy<sup>1</sup>**. **He<sub>1</sub>** was pleased.  $\exists > \text{most}$

## 4 Rethinking external staticity

In this section, we'll suggest a way to fix these bad predictions while retaining **Chierchia's** basic account of wco by tweaking the semantics of negation. The idea will be as follows: negation, and similar operators, are externally static with respect to *indeterminate* DRS, but externally dynamic with respect to *determinate* DRS.<sup>15</sup> To illustrate, consider the meaning of the sentence "A girl<sup>1</sup> likes Jeff<sup>2</sup>" in DS, where both Determiner Phrases (DPS) introduce DRS (n.b., we're abstracting away from events and other complications here). If there are three girls who like Jeff, namely *Shirley*, *Britta*, and *Annie*, then the dynamic proposition expressed by the sentence is as follows:

<sup>14</sup>Thanks to Filipe Hisao Kobayashi (p.c.) for drawing my attention to examples such as this.

<sup>15</sup>This approach was in part inspired by unpublished work on negation in dynamic semantics by Simon Charlow, who kindly shared this work with me. The resulting system however will focus on the semantic contribution of *indefinites* rather than negation per se.

(35) Context: *Three girls like Jeff*  
 $\lambda g . \{ g^{[1 \rightarrow \text{shirley}, 2 \rightarrow \text{jeff}]}, g^{[1 \rightarrow \text{britta}, 2 \rightarrow \text{jeff}]}, g^{[1 \rightarrow \text{annie}, 2 \rightarrow \text{jeff}]} \}$

Classical dynamic negation is destructive — applying it in this instance would eliminate all DRS, including 2. Rather, we want negation to eliminate all *indeterminate* DRS (here 1), and but retain determinate DRS (here 2). We want the effect of applying negation to be as follows:

(36)  $\lambda g . \{ g^{[2 \rightarrow \text{jeff}]} \}$

It’s not clear how to achieve this in a first-generation DS. The reason for this is as follows — in classical DS, we only distinguish between assignments that are in the output set, and those that are not. If there are no girls that like Jeff, then “A girl<sup>1</sup> likes Jeff<sup>2</sup>” will simply return the empty set, for any input assignment  $g$ . There is no way to get back the information associated with the DR 2. In order to address this issue, we will need to adopt a more expressive dynamic system — the monadic DS of Charlow (2014, 2019). Before turning to the refined system, I give an additional, independent argument that we want a dynamic semantics in which determinate DRS are left unscathed by operators such as negation.

#### 4.1 From classical DS to monadic DS

We’ll now shift gears and introduce a system with somewhat more expressive power than classical DS — this is because classical DS only distinguishes between assignments that are in the output set, and those that are not. We will need a three-way distinction between assignments that are paired with *true*, in the output set, those that are paired with *false*, and those absent from the output set. Charlow’s (2014) monadic DS delivers exactly these design features, and n.b., the expressive power afforded is independently motivated, providing as it does a highly compelling account of exceptional scope phenomena.<sup>16</sup>

In monadic DS, rather than defining the type of a dynamic proposition directly, we provide a recipe for mapping an “ordinary” type to its dynamic counterpart. Technically speaking, this makes use of the `State.Set` monad, but the formal niceties will be largely unimportant for our purposes. What will be important here is that, in monadic DS, we keep track of ordinary semantics values, while simultaneously engaging in the familiar acrobatics of DS.

(37) The `State.Set` type constructor (def.)  
 $S a := g \rightarrow \{ (a, g) \}$

The type of a dynamic proposition is just what we get if we apply the recipe in (37) to type  $t$  — the type of a truth value. In monadic DS, therefore, a dynamic proposition maps an input assignment to a set of truth-value-assignment pairs. This is illustrated in (38).

---

<sup>16</sup>Unlike other theories of exceptional scope based on pseudo-scope mechanisms such as *choice functions*, Charlow’s theory accounts for the *binder roof constraint* (Brasoveanu & Farkas 2011).

- (38)  $\top := g \rightarrow \{g\}$  Type of a dynamic proposition in classical DS  
 $\text{St} := g \rightarrow \{(t, g)\}$  Type of a dynamic proposition in monadic DS

In monadic DS, the counterpart of a *test* returns either  $\{(\top, g)\}$  or  $\{(\perp, g)\}$  for a given input  $g$ . This is illustrated for a simple sentence in (39). If the conditions are satisfied (namely, if *John left*), the test returns the (singleton set containing) *true-tagged* input assignment. If the conditions aren't satisfied, it returns the false-tagged input assignment — a classical test, by way of contrast, would simply return the empty set in the latter instance.

$$(39) \quad \llbracket \text{John left} \rrbracket = \lambda g . \{(j \text{ left}, g)\} =: \lambda g . \begin{cases} \{(\top, g)\} & \text{John left} \\ \{(\perp, g)\} & \text{otherwise} \end{cases} \quad \text{St}$$

We can retrieve the classical dynamic proposition from a monadic proposition by simply filtering out all the true tagged assignments — for illustrative purposes, we define a helper function  $\mathcal{C}$  to do just this in (40). We retrieve the classical test from the monadic test in (41).

$$(40) \quad \text{Converting from a monadic DS proposition to a classical DS proposition:} \\ \mathcal{C} p = \lambda g . \{g \mid (g, \top) \in (p g)\} \quad \text{St} \rightarrow \top$$

$$(41) \quad \mathcal{C} (\lambda g . \{(j \text{ left}, g)\}) = \lambda g . \begin{cases} \{g\} & \text{John left} \\ \emptyset & \text{otherwise} \end{cases} \quad \top$$

In monadic DS, just as in classical DS, a sentence with an indefinite, such as “A boy left”, induces an indeterminate output. Unlike in classical DS, we keep track of both the *verifiers* (i.e., the boys that left) and the *falsifiers* (i.e., the boys that didn't leave). This is illustrated in (42): true-tagged assignments map 1 to a boy who left, and false-tagged assignments map 1 to a boy who didn't leave. In classical DS, on the other hand, we only retain assignments corresponding to verifiers. Again, applying  $\mathcal{C}$  to the meaning in (42) returns the corresponding meaning in classical DS. This is shown in (43).

$$(42) \quad \llbracket \text{a boy}^1 \text{ left} \rrbracket = \lambda g . \{(\text{left } x, g^{[1 \rightarrow x]}) \mid \text{boy } x\} \quad \text{St} \\ =: \lambda g . \bigcup_{\text{boy } x} \begin{cases} \{(\top, g^{[1 \rightarrow x]})\} & \text{left } x \\ \{(\perp, g^{[1 \rightarrow x]})\} & \text{otherwise} \end{cases}$$

$$(43) \quad \mathcal{C} (\lambda g . \{(\text{left } x, g^{[1 \rightarrow x]}) \mid \text{boy } x\}) = \lambda g . \{g^{[1 \rightarrow x]} \mid \text{boy } x \wedge \text{left } x\} \quad \top$$

Just as in classical DS, in monadic DS sentences with pronouns induce input sensitivity. We leave it as an exercise to the reader to retrieve the corresponding classical dynamic proposition.



$$(44) \quad \llbracket \text{he}_1 \text{ was upset} \rrbracket = \lambda g . \{ (\text{upset } g_1, g) \} \quad \text{S t}$$

Dynamic sequencing behaves just as in classical DS, only in addition to passing outputs from the first conjunct into the second, it conjoins the contained truth values.<sup>17</sup>

$$(45) \quad m ; n := \lambda g . \bigcup_{(t, g') \in m g} \{ (t \wedge u, g'') \mid (u, g'') \in n g' \} \quad \text{S t} \rightarrow \text{S t} \rightarrow \text{S t}$$

Let's now consider a concrete example of cross-sentential anaphora in monadic DS, to get a feel for how things work. Note that we keep track of both the *verifiers* (here: the women who walked in and sat down), and the *falsifiers* (here: the woman who didn't walk in but sat down).

$$(46) \quad \begin{aligned} &\text{Context: } \textit{Only Britta and Shirley walked in, but Britta, Shirley, and Annie all sat down.} \\ &(\llbracket \text{a woman}^1 \text{ walked in} \rrbracket ; \llbracket \text{she}_1 \text{ sat down} \rrbracket) g_\emptyset \\ &= \bigcup_{(g', t) \in \llbracket \text{a woman}^1 \text{ walked in} \rrbracket g_\emptyset} \{ (t \wedge u, g'') \mid (u, g'') \in \llbracket \text{she}_1 \text{ sat down} \rrbracket g' \} \\ &= \{ (\top \wedge u, g'') \mid (u, g'') \in \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow s] \} \\ &\quad \cup \{ (\top \wedge u, g'') \mid (u, g'') \in \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow b] \} \\ &\quad \cup \{ (\perp \wedge u, g'') \mid (u, g'') \in \llbracket \text{she}_1 \text{ sat down} \rrbracket [1 \rightarrow a] \} \\ &= \{ (\top, [1 \rightarrow s]), (\top, [1 \rightarrow b]), (\perp, [1 \rightarrow a]) \} \end{aligned}$$

## 4.2 Refining the semantics of indefinites

Now that we have a better understanding of monadic DS at the sentential level, we will need to rethink our assumptions concerning the meanings of sentences with indefinites. First, let's zoom in to the subsentential level, and consider the standard semantics for indefinites in monadic DS, after [Charlow \(2014, 2019\)](#).<sup>18</sup>

<sup>17</sup>In a monadic setting, it isn't in fact necessary to define sequencing directly – it arises through the interaction of classical conjunction and exceptional scope mechanisms. See [Charlow \(2019\)](#) for details. We do so here only be illustrative purposes.

<sup>18</sup>As discussed in detail by [Charlow](#), the meaning in (49) can be decomposed into the basic meaning for the indefinite, given in (47), and monadic bind ( $\star$ ), defined in (48):

$$(47) \quad \lambda g . \{ (x, g^{[1 \rightarrow x]}) \mid \text{boy } x \} \quad \text{S e}$$

$$(48) \quad m^\star := \lambda k g . \bigcup_{(x, g') \in m g} k x g' \quad (\text{a} \rightarrow \text{S b}) \rightarrow \text{S b}$$

We abstract away from this here, but this decomposition plays a *crucial* role in [Charlow's](#) monadic DS, since the availability of bind directly accounts for the exceptional scoping abilities of indefinites.

$$(49) \text{ Indefinites in monadic } \mathcal{D}\mathcal{S} \text{ (first attempt)} \quad (e \rightarrow \mathcal{S} t) \rightarrow \mathcal{S} t$$

$$\llbracket \text{a boy}^1 \rrbracket = \lambda k g . \bigcup_{\text{boy } x} k x g^{[1 \rightarrow x]}$$

Indefinites compose with a scope and return a function from an input assignment, to a (potentially) heterogeneous set of true- and false-tagged assignments. This is illustrated in (50).

$$(50) \text{ Context: } \textit{Of the boys, only Troy and Abed left}$$

$$\llbracket \text{a boy}^1 \text{ left} \rrbracket = \llbracket \text{a boy}^1 \rrbracket (\lambda x g . \{ (\text{left } x, g) \}) \quad \mathcal{S} t$$

$$= \lambda g . \{ (\top, g^{1 \rightarrow \text{troy}}), (\top, g^{[1 \rightarrow \text{abed}]}), (\perp, g^{[1 \rightarrow \text{jeff}]}) \}$$

Recall that we ultimately want negation to be externally dynamic, at least selectively. In monadic  $\mathcal{D}\mathcal{S}$ , there's a very straightforward way of achieving this by simply lifting propositional negation into a dynamic setting. The definition is given in (53) — externally dynamic negation simply flips the contained truth-values in the output set.<sup>19</sup>

$$(53) \text{ Externally dynamic negation (def.)} \quad \mathcal{S} t \rightarrow \mathcal{S} t$$

$$\text{not } m := \lambda g . \{ (\neg t, g') \mid (t, g') \in m g \}$$

Perhaps we can simply adopt the semantics for negation in (53) and be done with it. Unfortunately, this will make bad predictions in tandem with the semantics for indefinites given in (49). This is because the entry for negation in (53) is externally dynamic with respect to *all* discourse referents. In effect, it simply allows the dynamic scope of an indefinite to pass straight through it. This is illustrated below:

$$(54) \text{ not } \llbracket \text{someone}^1 \text{ left} \rrbracket = \text{not } (\lambda g . \{ (\text{left } x, g^{[1 \rightarrow x]}) \mid \text{person } x \})$$

$$= \lambda g . \{ (\neg (\text{left } x), g^{[1 \rightarrow x]}) \mid \text{person } x \}$$

This won't do at all — as we've seen, if an indefinite takes scope below negation, negation rooofs its potential to bind. In order to avoid this result, the strategy we'll pursue here is one of tweaking the semantics of indefinites, while maintaining that negation is simply externally dynamic. Intuitively, what is going wrong here is that a sentence with an indefinite may give rise to a heterogeneous output set, consisting of both true-tagged and false-tagged assignments.

<sup>19</sup>Why might we want to define negation in this way? There's actually a strong conceptual motivation for doing so. It turns out that the entry in (53) is exactly what we get if we *map* plain old propositional negation into a dynamic setting — since  $\mathcal{S}$  is a functor, the recipe for achieving this mapping can be defined straightforwardly for any function:

$$(51) \text{ map } f := \lambda m \lambda g . \{ (f x, g') \mid (x, g') \in m g \} \quad \text{map} : (a \rightarrow b) \rightarrow \mathcal{S} a \rightarrow \mathcal{S} b$$

$$(52) \text{ map } (\neg) =: (53)$$

A criticism often leveled at  $\mathcal{D}\mathcal{S}$  is that it precompiles the desired behavior directly into the meanings of the logical connectives (see, e.g., [Schlenker 2009](#) for discussion). Here however, all we've done is lifted classical negation into a dynamic setting via a totally general and logically well-behaved operation.

Rather, we need to ensure that a sentence with an indefinite only *ever* outputs either (a) a set of true-tagged assignments, or (b) a set of false-tagged assignments. Towards this aim, we'll first a helper functions, which filters out the false-tagged assignments.

(55) Positive extension operator (def)

$$p^+ := \{(\top, g') \mid (\top, g') \in p\} \quad + : \{g, t\} \rightarrow \{g, t\}$$

We can now revise our semantics for indefinites in terms of (a) **Charlow's** semantics for indefinites, and (b) the positive extension operator. The idea behind the new entry will be as follows: if the output set contains a true-tagged assignment, return its positive extension, and otherwise, return the (singleton set of the) *grand intersection* of the false-tagged assignments, and tag it false. The reader may wonder at this point why we don't just return the false-tagged assignments. This will in fact be crucial for explaining why only *determinate* DRS may survive, as we'll see shortly. .

(56) Indefinites in monadic DS (second attempt)

$$\llbracket \text{a boy}^n \rrbracket := \lambda k g . \begin{cases} (p g)^+ & (p g)^+ \neq \emptyset \\ \{(\perp, \bigcap \{g' \mid (g', \perp) \in (p g)\})\} & \text{otherwise} \end{cases} \quad p := \bigcup_{\text{boy } x} k x g^{[n \rightarrow x]}$$

We need to say something here about what it means to intersect two assignments, as this is a somewhat unfamiliar notion. We can take assignment functions to be equivalent to their graphs, i.e., sets of variable-value pairs. This is illustrated below for an assignment mapping the variables  $\{1, 2, 3\}$  to boys.

$$\left[ \begin{array}{l} 1 \rightarrow \text{jeff} \\ 2 \rightarrow \text{troy} \\ 3 \rightarrow \text{abed} \end{array} \right] =: \left\{ \begin{array}{l} (1, \text{jeff}), \\ (2, \text{troy}), \\ (3, \text{abed}) \end{array} \right\}$$

Intersecting two assignments therefore amounts to set intersection of their graphs. This will have the desired effect of preserving determinate DRS, while eliminating indeterminate DRS. Suppose we have a set of assignments which agree on 1 and 2, but differ on 3 — Intersecting their graphs returns an assignment that is undefined at 3, but which preserves information about the determinate DRS.

$$\bigcap \left\{ \left\{ \begin{array}{l} (1, \text{troy}) \\ (2, \text{abed}) \\ (3, \text{shirley}) \end{array} \right\}, \left\{ \begin{array}{l} (1, \text{troy}) \\ (2, \text{abed}) \\ (3, \text{britta}) \end{array} \right\}, \left\{ \begin{array}{l} (1, \text{troy}) \\ (2, \text{abed}) \\ (3, \text{annie}) \end{array} \right\} \right\} = \left\{ \begin{array}{l} (1, \text{troy}) \\ (2, \text{abed}) \end{array} \right\}$$

### 4.3 Deriving the interaction with negation

We've done a lot of work in order to define indefinites in a way which seems rather ad hoc, but as we'll see, this new definition will have number of advantages. Originally, the problem with

Chierchia’s account of wco arose due to the assumption that negation is fundamentally a *destructive* operation, taken from first-generation dynamic theories. In our new, monadic setting, we can simply make use of externally dynamic operation, as defined in (53), and repeated below.

$$(57) \quad \text{Externally dynamic negation (def.)} \\ \text{not } m := \lambda g . \{(\neg t, g') \mid (t, g') \in m g\} \quad S t \rightarrow S t$$

With our refined semantics for indefinites, this simplistic treatment of negation will pay dividends. On the one hand, due to the way that indefinites have been (re-)defined, we will retain the basic results of classical DS — namely, negation will still render any DRS introduced by an indefinite inaccessible<sup>20</sup> On the other hand, determinate DRS will survive. As we’ll see later, an added bonus of this system will be an account of double-negation — a recalcitrant problem for classical DS.

First off, let’s make sure that we get the basic static truth-conditions correct. We can retrieve a classical notion of truth relative to assignment from a dynamic proposition by checking whether the output contains at least one true-tagged assignment.<sup>21</sup> Now, consider the meaning of “A boy<sup>1</sup> left”, where Troy and Abed left, but Jeff didn’t. Since the positive extension is non-empty, this sentence will *only* output true-tagged assignments:

$$(59) \quad \lambda g . \{(\top, g^{[1 \rightarrow \text{troy}]}), (\top, g^{[1 \rightarrow \text{abed}]})\}$$

Negating the sentence will simply flip the contained truth values, so the sentence will come out false, as desired.

$$(60) \quad \text{Context: } \textit{Of the boys, only Troy and Abed left.} \\ \text{not } ([\text{a boy left}]) = \lambda g . \{(\perp, g^{[1 \rightarrow \text{troy}]}), (\perp, g^{[1 \rightarrow \text{abed}]})\}$$

Now, consider what happens if the facts are such that *no boy left*. First, let’s consider the output set induced by Charlow’s semantics for the indefinite.

$$(61) \quad \lambda g . \{(\perp, g^{[1 \rightarrow \text{j}]}), (\perp, g^{[1 \rightarrow \text{t}]}), (\perp, g^{[1 \rightarrow \text{a}]})\}$$

In order to compute the meaning of the sentence “A boy<sup>1</sup> left”, we first filter out the true-tagged assignments — since there are none, we just get back the same meaning. Based on how the indefinite is defined in our new setting, the output of “A boy<sup>1</sup> left” should be the grand intersection of the false-tagged assignments. Since the output assignments disagree only in 1, the output simply contains  $g$  (in other words, we end up with a false-tagged test).

<sup>20</sup>This statement has an important caveat – if the indefinite introduces a *determinate* DR, it will survive, as we’ll see.

<sup>21</sup>We can define this formally as an operation that existentially closes the output set.

$$(58) \quad \text{clo } m := \lambda g . \exists g' [(T, g') \in m g] \quad \text{clo} : S t \rightarrow g \rightarrow t$$

- (62) Context: *No boys left.*  
 $\llbracket \text{a boy}^1 \text{ left} \rrbracket := \lambda g . (\perp, g)$

Applying negation simply flips the contained truth-value — the result is a true-tagged test. We therefore retain a core result of  $\text{DS}$ , that negation roofs the dynamic scope of indefinites.

- (63) Context: *No boys left.*  
 $\text{not } \llbracket \text{a boy}^1 \text{ left} \rrbracket = \lambda g . (\top, g)$

This is all well and good, but so far we haven’t done anything above and beyond classical  $\text{DS}$ . The interesting part comes when alongside an indefinite, we also have some determinate  $\text{DR}$  introduced. Consider the sentence “A boy<sup>1</sup> likes Annie<sup>2</sup>” in a context where no boy in fact likes Annie. Again, let’s start out with the meaning predicted by Charlow’s semantics, and just take the false-tagged assignments:

- (64)  $\lambda g . \{ (\perp, g^{[1 \rightarrow \text{jeff}, 2 \rightarrow \text{annie}]}) , (\perp, g^{[1 \rightarrow \text{troy}, 2 \rightarrow \text{annie}]}) , (\perp, g^{[1 \rightarrow \text{abed}, 2 \rightarrow \text{annie}]}) \}$

To compute the meaning of “A boy<sup>1</sup> likes Annie<sup>2</sup>” in this context, recall that we first check whether the positive extension is empty. Since it is, we take the grand intersection of the false-tagged assignments. Since all the output assignments map 2 to Annie, but disagree on 1, the unique such assignment is one that is just like  $g$ , but maps 2 to Annie.

- (65) Context: *No boys like Annie*  
 $\llbracket \text{A boy}^1 \text{ likes Annie}^2 \rrbracket = \lambda g . \{ (\perp, g^{[2 \rightarrow \text{a}]} ) \}$

Negating the resulting meaning simply flips the resulting meaning, and the result is no longer a test! We correctly predict that determinate  $\text{DRS}$  can survive negation. We can think of negation as being a dynamically *translucent* operator — it allows only determinate  $\text{DRS}$  to survive — namely, only those that all output assignments agree upon.

- (66) Context: *No boys like Annie*  
 $\text{not } \llbracket \text{A boy}^1 \text{ likes Annie}^2 \rrbracket = \lambda g . \{ (\top, g^{[2 \rightarrow \text{a}]} ) \}$

#### 4.4 Applying the results to crossover

On the dynamic regime adopted here, we can now resolve the problem for Chierchia’s account of wco. For the purposes of illustration, in the previous section we attributed  $\text{DR}$ -introduction to indefinites directly. Since we want to maintain Chierchia’s account of wco, the revised entry for indefinites is identical to the definition in (56), only it doesn’t introduce a discourse reference (this, as on Chierchia’s account, will be the preserve of the thematic position).

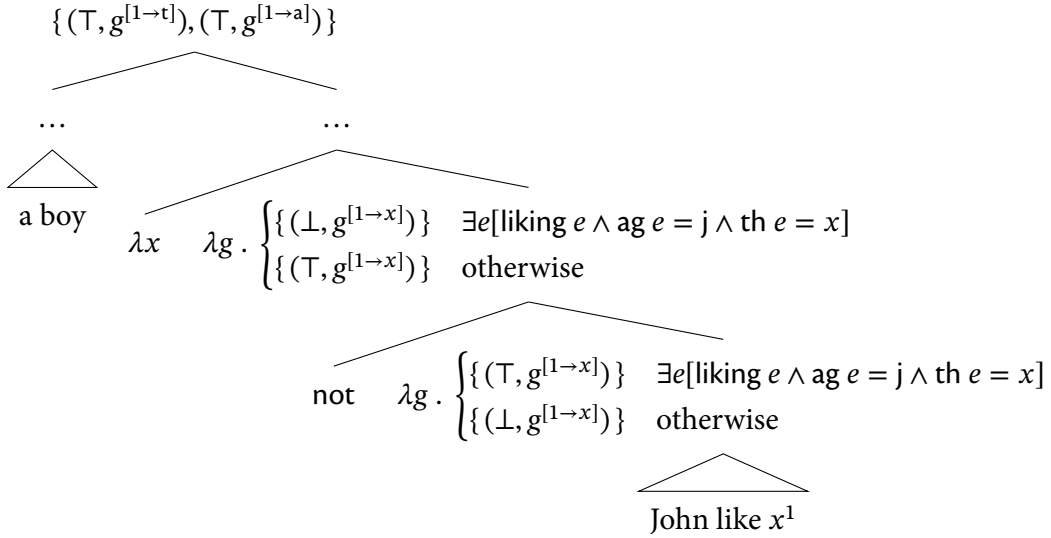
(67) Indefinites in monadic dynamics (final ver.)

$$\llbracket \text{a boy} \rrbracket := \lambda k g . \begin{cases} (p g)^+ & (p g)^+ \neq \emptyset \\ \{(\perp, \bigcap \{g' \mid (g', \perp) \in (p g)\})\} & \text{otherwise} \end{cases}$$

$$p := \bigcup_{\text{boy } x} k x g$$

Recall that examples where indefinites take scope over negation were problematic for Chierchia's account. The insight here is that, if the indefinite takes wide scope, at the point in the derivation at which negation composes, the DR introduced by the thematic position is *determinate*, and therefore survives. We can therefore retain Chierchia's account of wco just so long as we couch it in terms of the refined dynamic system outlined previously. The task of lifting Chierchia's dynamic event semantics is a largely mechanical one, and therefore the details are elided.<sup>22</sup>

Figure (4): A boy John doesn't like \_\_\_<sup>1</sup>  
(context: *John likes Troy and Abed but not Jeff*)



#### 4.5 The problem of double negation

An independent motivation for the variation on monadic DS outlined here is a principles account of an otherwise recalcitrant problem for first generation dynamic theories — double negation

<sup>22</sup>More concretely, We can simply replace instances of *dynamic lift* with a more general operation *monadic return*, defined in (68).

(68) Monadic return (def.)

$$a^{\eta} := \lambda g . \{(a, g)\}$$

$$a \rightarrow S a$$

(see, e.g., [Krahmer & Muskens 1995](#), [Gotham 2019](#) for discussion). As is well known, classical DS with equipped with externally static negation cannot handle anaphora in discourses such as the following:<sup>23</sup>

(69) John doesn't own no car<sup>1</sup>; it<sub>1</sub>'s in the garage.

Perhaps unexpectedly, an independent motivation for the semantics of indefinites and negation outlined here is that it provides a straightforward account of (69). This is because, according to the revised semantics for dynamic negation adopted here, all it does is flip the contained truth-values in the output set. Recall furthermore that, indefinites return the set of true-tagged semantics, if non-empty. This means that “John owns a car<sup>1</sup>” returns the following, if the facts are such that John owns exactly one car – the ferrari (f):

(70)  $\lambda g . \{(\top, g^{[1 \rightarrow f]})\}$

If we apply externally static negation to this meaning *twice*, all it will be will flip the contained truth value twice, giving back...an identical meaning:

(71)  $\text{not} (\text{not} (\lambda g . \{(\top, g^{[1 \rightarrow f]})\})) = \text{not} (\lambda g . \{(\perp, g^{[1 \rightarrow f]})\}) = \lambda g . \{(\top, g^{[1 \rightarrow f]})\}$

One potential worry for this account is that, as discussed by [Gotham \(2019\)](#), anaphora from under double negation seems to give rise to a uniqueness inference.

(72) a. John owns a shirt<sup>1</sup>. It<sub>1</sub>'s hanging up. The rest are in the closet.  
 b. John doesn't own no shirt<sup>1</sup>. It<sub>1</sub>'s hanging up. ??The rest are in the closet.

I don't have much to add here, other than to note that there seem to be exceptions to uniqueness, involving maximal reference, as illustrated in (73). Indeed, [Gotham](#) does not have a principled explanation for the uniqueness effect, and it seems reasonable to conclude that this phenomenon is still poorly understood.

(73) John doesn't own no shirt<sup>1</sup>. They're in the closet.

## 5 Conclusion

An important contribution of monadic DS which we've glossed over in this paper is that it provides a way to generalize function-argument application to values with dynamic side-effects, as in (74). Application in monadic DS passes output assignments from the first value  $m$  into the second value  $n$  pointwise, and does Function Application (FA) of the contained ordinary-semantic values.

---

<sup>23</sup>We assume here that, at LF negative indefinites are decomposed into a negative and an existential component.



$$(74) \text{ Application in monadic DS} \quad m \otimes n := \lambda g . \bigcup_{(f, g') \in m g} \{(f x, g'') \mid (x, g'') \in n g'\} \quad S(a \rightarrow b) \rightarrow S a \rightarrow S b$$

As discussed at the beginning of this paper, one of Chierchia’s motivations for dynamicizing event semantics was to allow for a system in which the machinery of dynamic semantics does some work intrasententially. Adopting monadic DS wholesale arguably renders this move redundant. We’ve refrained from the more substantial enterprise of recasting Chierchia’s ideas in an event-free monadic setting, but this would be helpful in order to understand the core properties of this approach to wco.

Another issue with Chierchia’s theory which we haven’t addressed here is its reliance on a syntactic distinction between pronouns and *traces*. Arguably, this is an artifact of its adoption of the QR approach to scope. In ongoing work (Elliott in progress), I attempt to recast the basic ideas of Chierchia’s theory in a dynamic semantics where quantificational scope is modeled via *continuations* (Barker 2002, Barker & Shan 2014).

In this paper our aim has been rather more conservative — we’ve argued that Chierchia’s theory, as it stands, makes bad predictions regarding interactions with externally static operators. This motivated a significant rethinking of the underlying dynamic plumbing — ultimately, we argued for the expressive power of monadic DS, where we systematically keep track of both verifiers and falsifiers. The core the account lay in the revised semantics for indefinites which, unlike in standard monadic DS, either return true-tagged assignments *only*, or the grand-intersection of the false-tagged assignments. This revised semantics allowed us to simply lift propositional negation into our dynamic setting, resulting in an operator with all the desired properties — namely, it roofs dynamic scope with respect to indeterminate DRS, but allows determinate DR to survive unscathed. This move, as an added bonus, provided us with a straightforward account of double negation.

The properties of the resulting system deserve a more thorough exploration than we have had space for here. Namely, it remains to be seen whether there is any independent motivation for allowing operations such as negation to leave *determinate* DRS unscathed. Despite the fact that the analysis was motivated by fairly narrow concerns — namely, a technical problem with Chierchia’s theory of wco — we believe that the resulting discussion has broader lessons for any theory of wco couched in DS, and is suggestive of the properties we ultimately want the dynamic semantics characterizing natural language to have.

## References

- Barker, Chris. 2002. Continuations and the Nature of Quantification. *Natural Language Semantics* 10(3). 211–242.
- Barker, Chris & Chung-chieh Shan. 2014. *Continuations and natural language* (Oxford Studies in Theoretical Linguistics 53). Oxford University Press. 228 pp.
- Brasoveanu, Adrian & Donka F. Farkas. 2011. How indefinites choose their scope. *Linguistics and Philosophy* 34(1). 1–55.
- Castañeda, Hector-Neri. 1967. Comments. In Nicholas Resher (ed.), *The logic of decision and action*, 104–112. Pittsburgh: University of Pittsburgh Press.

- Charlow, Simon. 2014. *On the Semantics of Exceptional Scope*. Dissertation.
- Charlow, Simon. 2019. Static and dynamic exceptional scope. [lingbuzz/004650](https://lingbuzz/004650).
- Chierchia, Gennaro. 2020. Origins of weak crossover - when dynamic semantics meets event semantics. *Natural Language Semantics* (28). 23–76.
- Gotham, Matthew. 2019. Double negation, excluded middle and accessibility in dynamic semantics. In Julian J. Schlöder, Dean McHugh & Floris Roelofsen (eds.), *Proceedings of the 22nd Amsterdam Colloquium*, 142–151.
- Groenendijk, Jeroen & Martin Stokhof. 1991. Dynamic predicate logic. *Linguistics and Philosophy* 14(1). 39–100.
- Heim, Irene. 1982. *The semantics of definite and indefinite noun phrases*. University of Massachusetts - Amherst dissertation.
- Heim, Irene. 1991. Artikel und definitheit. In Armin von Stechow & Dieter Wunderlich (eds.), *Semantik: Ein internationales Handbuch der zeitgenössischen Forschung*, 487–535. de Gruyter Mouton.
- Krahmer, Emiel & Reinhard Muskens. 1995. Negation and Disjunction in Discourse Representation Theory. *Journal of Semantics* 12(4). 357–376.
- May, Robert. 1977. *The grammar of quantification*. Massachusetts Institute of Technology dissertation.
- Partee, Barbara & Mats Rooth. 1983. Generalized conjunction and type ambiguity. In *Meaning, use, and interpretation of language*, Reprint 2012, 361–383. Berlin, Boston: De Gruyter.
- Postal, Paul Martin. 1971. *Cross-over phenomena* (Transatlantic Series in Linguistics). New York: Holt, Rinehart and Winston. 262 pp.
- Ruys, E. G. 2000. Weak Crossover as a Scope Phenomenon. *Linguistic Inquiry* 31(3). 513–539.
- Schlenker, Philippe. 2009. Local contexts. *Semantics and Pragmatics* 2.