Staged Updates

Simon Charlow (*Rutgers*)

NY Phil Lang Workshop November 6, 2023

simoncharlow.com/nyplw

Overview

Dynamic semantics is the best game in town for understanding anaphora. But:

- Standard dynamic systems don't allow involutive negation (DNE); this is bad
- Several theories, but little consensus, or understanding of how they relate

This talk: many existing, prominent theories of DNE share a common core:

- They are all, in one way or another, systems of staged updates
- This captures what's important to capture about DNE in dynamic semantics
- Other differences between the theories are superficial, or incidental
- This allows us to theory-build without worrying about implementation details, and easily translate insights between theories

Dynamic semantics and negation

Unexpected anaphora

Indefinites can antecede pronouns they don't (can't) scope over:

1. Polly$_i$ walked in the park. She$_i$ whistled.
2. A linguist$_i$ walked in the park. She$_i$ whistled.
3. If a man$_i$ is from Omaha, he$_i$ isn't from Lincoln.

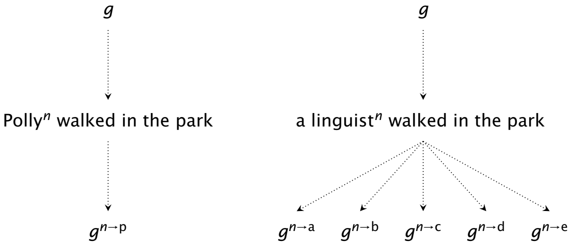
The meanings of (2) and (3) aren't captured by the natural compositional translations:

- $\exists x(Lx \wedge Px) \wedge Wx$
- $\exists x(Mx \wedge Ox) \Rightarrow \neg Lx$

Dynamic semantics explains (2) and (3) by assimilating them to (1).

Geach 1962, Lewis 1975, Karttunen 1976, Heim 1982, many more

Dynamic semantics



Heim 1982, Barwise 1987, Rooth 1987, Groenendijk & Stokhof 1991a, Muskens 1996

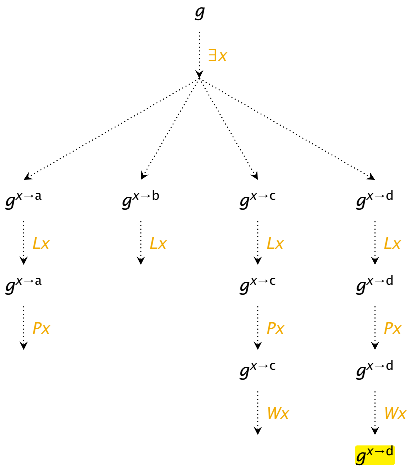
Dynamic Predicate Logic

$$
\begin{aligned}
g[Fx_1, \ldots, x_n] &:= \{g\} \text{ if } (\llbracket x_1 \rrbracket^g, \ldots, \llbracket x_n \rrbracket^g) \in \llbracket F \rrbracket^g \text{ else } \varnothing \\
g[\exists x] &:= \{g^{x \to d} \mid d \in D\} \\
g[\phi \wedge \psi] &:= \{h \in k[\psi] \mid k \in g[\phi]\} \\
g[\neg \phi] &:= \{g\} \text{ if } g[\phi] = \varnothing \text{ else } \varnothing
\end{aligned}
$$

- I'll generally abbreviate DPL '$\exists x \wedge Fx$' as '$\exists x Fx$'
- We can easily go intensional by making our states (g, w) pairs
- This is a *relational* presentation of DPL, that is, $[\phi] : \mathtt{i} \to \{\mathtt{i}\}$. We can equivalently give DPL updates as *functions* from sets of points to sets of points, $\{\mathtt{i}\} \to \{\mathtt{i}\}$.

Groenendijk & Stokhof 1991a,b, Dekker 1994, Muskens 1996, ...

Cross-sentential anaphora: $\exists x(Lx \wedge Px) \wedge Wx$



Donkey anaphora

Implication is defined in the usual material way:

$$\phi \Rightarrow \psi := \neg(\phi \wedge \neg\psi)$$

Donkey anaphora follows. We get a “strong” reading, requiring every *F* to be *G*:

$$\begin{aligned} \exists x Fx \Rightarrow Gx &= \neg(\exists x Fx \wedge \neg Gx) \\ &= \neg\exists x(Fx \wedge \neg Gx) \end{aligned}$$

We’ll get to negation on the next slide, but the key point here is that existentials are able to expand their scope rightward.

We can get a weak reading if $\phi \Rightarrow \psi := \neg\phi \vee (\phi \wedge \psi)$, given a suitable semantics for disjunction

Negation and failures of DNE

When is something **true** in DPL? When it has **any** outputs:

$$g[F(x_1, \ldots, x_n)] := \{g\} \textit{ if } ([\![x_1]\!]^g, \ldots, [\![x_n]\!]^g) \in [\![F]\!]^g \textit{ else } \varnothing$$

When is something **false**? When it has **no** outputs. So when $\neg\phi$ is true, all we can return is the unchanged input:

$$g[\neg\phi] := \{g\} \textit{ if } g[\phi] = \varnothing \textit{ else } \varnothing$$

One negation is anaphorically opaque, and two negations are too: $\neg\neg\phi$ returns its input unchanged (conditional on ϕ being true). So $\neg\neg\phi \neq \phi$, and in particular:

$$\neg\neg\exists xFx \neq \exists xFx$$

Negation in the wild

Though the opaque semantics for negation works well for sentences with just one negation, it wrongly rules out anaphora to doubly-negated indefinites:

4. I don't have a car. #It's parked outside.
5. It isn't the case that I don't have a car. It's in my garage.
6. It's not the case that Sue doesn't have a child. She's at boarding school.
7. It's not the case that Sue isn't a parent. #She's at boarding school.

While (5)–(7) might be explained by giving the indefinite widest scope in its clause ($\exists x \neg\neg Fx = \exists x Fx$), Partee disjunctions show that this is not a general solution:

8. Either there's no bathroom here, or it's in a very funny place.
9. Either Sue doesn't have a child, or she's at boarding school.
10. #Either Sue isn't a parent, or she's at boarding school.

Child/parent examples lifted from Mandelkern 2022

On disjunction

It would be kind of natural to define disjunction in terms of negation:

$$\phi \vee \psi := \neg(\neg\phi \wedge \neg\psi)$$

But disjunction is both internally and externally dynamic, and this entry is neither:

11. If a farmer owns a donkey or a sheep, she hugs it.
12. If I run into Harry or I see Sam, I wave to him.
13. Either Mary doesn't have a car, or it's a VW.

Putting these together, we'd want something like the following to work, but the failure of DNE means that it doesn't:

$$g[\phi \vee \psi] := g[\phi] \cup g[\underbrace{\neg\phi \wedge}_{?} \psi]$$

Groenendijk & Stokhof 1991a, Stone 1992, Charlow 2014

Dynamic approaches to DNE

Krahmer & Muskens 1995

The idea (familiar from partial logic): sentences have a positive **extension** and a negative **anti-extension**. For simple existential and atomic ϕ:

$$[\phi]^{+} = [\ \ \phi]_{\mathsf{DPL}}$$
$$[\phi]^{-} = [\neg\phi]_{\mathsf{DPL}}$$

Then negation just swaps $+$ and $-$, from which it follows that 2 negations cancel:

$$[\neg\phi]^{+} := [\phi]^{-} \qquad [\neg\neg\phi]^{+} = [\neg\phi]^{-} = [\phi]^{+}$$
$$[\neg\phi]^{-} := [\phi]^{+} \qquad [\neg\neg\phi]^{-} = [\neg\phi]^{+} = [\phi]^{-}$$

Connectives and Partee disjunctions

Conjunction and implication are more complex but follow a simple pattern:

$$\left.\begin{array}{l} [\phi * \psi]^+ := [\phi]^+ *_{\mathrm{DPL}} [\psi]^+ \\ [\phi * \psi]^- := \neg_{\mathrm{DPL}} [\phi * \psi]^+ \end{array}\right\} * \in \{\wedge, \Rightarrow\}$$

Disjunction evaluates the right disjunct against the negation of the left, from which the possibility of anaphora in Partee disjunctions follows:

$$\begin{array}{ll} [\phi \vee \psi]^+ := [\phi]^- \Rightarrow_{\mathrm{DPL}} [\psi]^+ & \qquad [\neg\exists x Bx \vee Fx]^+ = [\neg\exists x Bx]^- \Rightarrow_{\mathrm{DPL}} [Fx]^+ \\ [\phi \vee \psi]^- := \neg_{\mathrm{DPL}} [\phi \vee \psi]^+ & \qquad = [\ \ \exists x Bx]^+ \Rightarrow_{\mathrm{DPL}} [Fx]^+ \end{array}$$

Krahmer & Muskens's disjunction is strong ('any bathroom here is in a funny place'). This choice is incidental, as they discuss in a footnote.

Mandelkern 2022

A seemingly quite different approach, weaving together several ideas:

- Sentences have classical truth conditions
- Sentences also have non-classical **bounds**, where anaphora lives

Existential sentences have classical truth conditions, but non-classical bounds:

- In bounds, existentials are **variables** (Heim 1982)
- They don't **introduce** discourse referents as such, but **restrict** our attention to certain ways of valuing variables.

Negation targets truth conditions, and is therefore classical.

The witness bound: decomposing dynamic semantics

$\exists x F x$

- **TRUE** at $(g, w) \longleftrightarrow F_w \neq \emptyset$
- **SATT** at $(c, g, w) \longleftrightarrow F_w \neq \emptyset \rightarrow g(x) \in F_w$
- **TRUE** and **SATT** at $(c, g, w) \longleftrightarrow g(x) \in F_w$

$F\mathrm{p}_x$

- **TRUE** at $(g, w) \longleftrightarrow g(x) \in F_w$
- **SATT** at $(c, g, w) \longleftrightarrow \forall (h, v) \in c : h(x)$ is defined

Conjunction and cross-sentential anaphora

$\phi \wedge \psi$

- **TRUE** at $(g, w) \longleftrightarrow \phi$ and ψ are **TRUE** at (g, w)
- **SATT** at $(c, g, w) \longleftrightarrow \phi$ is **SATT** at (c, g, w) and ψ is **SATT** at (c^{ϕ}, g, w)
- $c^{\phi} := \{(g, w) \in c \mid \phi$ is **TRUE** and **SATT** at $(c, g, w)\}$

$\exists x Fx \wedge G\mathrm{p}_x$

- **TRUE** at $(g, w) \longleftrightarrow F_w \neq \emptyset$ and $g(x) \in G_w$
- **SATT** at $(c, g, w) \longleftrightarrow F_w \neq \emptyset \rightarrow g(x) \in F_w$ and $G\mathrm{p}_x$ is **SATT** at $(c^{\exists x Fx}, g, w)$
- **TRUE** and **SATT** at $(c, g, w) \longleftrightarrow g(x) \in F_w \cap G_w$

Donkey anaphora

$\exists xFx \Rightarrow G\mathrm{p}_x$

- **TRUE** at $(g, w) \longleftrightarrow F_w \neq \varnothing \rightarrow g(x) \in G_w$
- **SATT** at $(c, g, w) \longleftrightarrow F_w \neq \varnothing \rightarrow g(x) \in F_w$ and $G\mathrm{p}_x$ is **SATT** at $(c^{\exists xFx}, g, w)$
- **TRUE** and **SATT** at $(c, g, w) \longleftrightarrow F_w \neq \varnothing \rightarrow g(x) \in F_w \cap G_w$

This is a “weak” meaning: it can hold at (c, g, w) with $F_w \neq \varnothing$ even if $F_w \nsubseteq G_w$. All we require is that at **this** g, x is F_w and G_w.

Negation and disjunction

$\neg\exists xFx$

- **TRUE** at $(g, w) \longleftrightarrow F_w = \varnothing$
- **SATT** at $(c, g, w) \longleftrightarrow \exists xFx$ is **SATT** at (c, g, w)

$\neg\exists xBx \vee F\mathrm{p}_x$

- **TRUE** at $(g, w) \longleftrightarrow B_w \neq \varnothing \rightarrow g(x) \in F_w$
- **SATT** at $(c, g, w) \longleftrightarrow B_w \neq \varnothing \rightarrow g(x) \in B_w$ and $F\mathrm{p}_x$ is **SATT** at $(c^{\neg\neg\exists xBx}, g, w)$
- **TRUE** and **SATT** at $(c, g, w) \longleftrightarrow B_w \neq \varnothing \rightarrow g(x) \in B_w \cap F_w$

Again, this is a "weak" meaning.

An unusual prediction

These sentences all seem to mean that nobody brought any of their umbrellas:

14. Nobody who has an umbrella brought it.
15. It isn't true that anyone with an umbrella brought it.
16. It isn't true that anyone has an umbrella and brought it.

$\neg(\exists x Ux \wedge B\mathrm{p}_x)$

- **TRUE** at $(g, w) \longleftrightarrow U_w \neq \varnothing \to g(x) \notin B_w$
- **SATT** at $(c, g, w) \longleftrightarrow U_w \neq \varnothing \to g(x) \in U_w$ and $G\mathrm{p}_x$ is **SATT** at $(c^{\exists x Ux}, g, w)$
- **TRUE** and **SATT** at $(c, g, w) \longleftrightarrow U_w \neq \varnothing \to g(x) \in U_w \setminus B_w$

Weaker than expected? This can hold at (c, g, w) with $U_w \neq \varnothing$ even if $U_w \cap B_w \neq \varnothing$.
Unlike DPL, where $\neg(\exists x Ux \wedge Bx) \equiv \neg\exists x(Ux \wedge Bx)$.

Benjamin Spector, p.c.

Indefinites as (half) variables

Mandelkern's system systematically predicts weak readings. This is due to the Heim 1982-style treatment of indefinites as variables (in **SATT** conditions).

Another characteristic consequence of treating indefinites as variables is the need to avoid variable reuse. Indefinites cannot be co-construed with other indefinites:

17. A linguist$_x$ walked in the park. A birdwatcher$_{y/*x}$ whistled.

These indexings must be ruled out with a syntactic or semantic **novelty** constraint.

Actually, I think Mandelkern's system is about the only one following Heim that *can* get by with a simple semantic novelty constraint, and his system avoids notorious complications with the existential closure operation that otherwise plague variabilist theories of indefinites (Reinhart 1997).

Staged updates

Warm up

Types can be manipulated up to isomorphism using familiar algebraic identities. Function types are exponentials, and product types are, well, products.

$$|a \to b| = |b|^{|a|} \qquad |a \times b| = |a| \times |b|$$

Some simple examples and their isomorphisms (ϕ is a meaning not a formula):

$$\begin{aligned} a \to (b \to c) &= (c^b)^a & \quad curry(f) &:= \lambda a.\lambda b.f(a,b) \\ &\simeq c^{a \times b} & uncurry(g) &:= \lambda(a,b).g(a)(b) \end{aligned}$$

$$\begin{aligned} a \to (b \times c) &= (b \times c)^a & \quad push(f) &:= (\lambda i.\pi_1(f(i)), \lambda i.\pi_2(f(i))) \\ &\simeq b^a \times c^a & pull(\phi,\psi) &:= \lambda i.(\phi(i), \psi(i)) \end{aligned}$$

In general, the laws of products and exponentiation (over positive integers) are a sound way to reason about the interaction of product and function types!

Some benighted souls use types to remember how products and exponents work

Mandelkern's semantics in intension

We can see Mandelkern's interpretation function as determining two truth values at a context, assignment, and world:

$$[\![\phi]\!]^{c,g,w} : \underbrace{\mathtt{t}}_{\textbf{TRUE}} \times \underbrace{\mathtt{t}}_{\textbf{SATT}}$$

So $[\![\phi]\!]$ delivers *intensions* of type $(\mathtt{c} \times \mathtt{g} \times \mathtt{w}) \to (\mathtt{t} \times \mathtt{t})$. These are isomorphic to **pairs of update functions**, quite like Krahmer & Muskens!

$$\begin{aligned}
(\mathtt{c} \times \mathtt{g} \times \mathtt{w}) \to (\mathtt{t} \times \mathtt{t}) &= (\mathtt{t} \times \mathtt{t})^{\mathtt{c}\times\mathtt{g}\times\mathtt{w}} && \equiv \\
&\simeq \mathtt{t}^{\mathtt{c}\times\mathtt{g}\times\mathtt{w}} \times \mathtt{t}^{\mathtt{c}\times\mathtt{g}\times\mathtt{w}} && \textit{push} \\
&\simeq (\mathtt{t}^{\mathtt{g}\times\mathtt{w}})^{\mathtt{c}} \times (\mathtt{t}^{\mathtt{g}\times\mathtt{w}})^{\mathtt{c}} && \textit{curry} \\
&\simeq \mathtt{c}^{\mathtt{c}} \times \mathtt{c}^{\mathtt{c}} && \equiv \\
&= (\mathtt{c} \to \mathtt{c}) \times (\mathtt{c} \to \mathtt{c}) && \equiv
\end{aligned}$$

Along the same lines, Yalcin 2007's domain semantics, where truth values are determined relative to a world and a domain (set) of worlds can be viewed as a kind of update semantics (Veltman 1996)

Staged updates

This is not quite fair, as only **SATT** conditions are sensitive to c.

$$\underbrace{(c \to c)}_{\text{constant functions}} \times (c \to c) \simeq c \times (c \to c)$$

$c \times (c \to c)$ is a type for **staged updates**:

- c is a **grounding context** embodying truth-conditional content that characterizes or restricts the sorts of points we may actually run the update at
- $c \to c$ is a **proto-update** which remains neutral on truth or falsity (this burden is shouldered by c), but knows how to update the context in either event

Staged updates seem like a good *general* model for DNE. Negation can target the grounding context classically, leaving the proto-update untouched.

- Negation's effects will be felt when the staged update is **run**

Back to bilateral systems

Krahmer & Muskens 1995 provide two interpretation functions, $[\cdot]^+$ and $[\cdot]^-$, both taking sentences into a DPL meaning, type $\mathtt{i} \to \mathtt{c}$ (writing $\mathtt{c}$ for $\{\mathtt{i}\}$).

This is equivalent to having one interpretation function that maps us into a pair of DPL meanings, i.e., $[\phi] : (\mathtt{i} \to \mathtt{c}) \times (\mathtt{i} \to \mathtt{c})$, or equivalently, $\mathtt{i} \to \mathtt{c}^2$.

$$
\begin{aligned}
(\mathtt{i} \to \mathtt{c}) \times (\mathtt{i} \to \mathtt{c}) &= \mathtt{i} \to (\mathtt{c} \times \mathtt{c}) && \textit{push} \\
&= \mathtt{i} \to \mathtt{c}^2 && \equiv
\end{aligned}
$$

In fact, $\mathtt{c}^2$ is more power than we need! Krahmer & Muskens's bilateral meanings are always **contraries**. Full bilateralism tolerates unrelated meanings (e.g., Potts 2005).

A more austere setting

The de facto restriction on $\mathtt{c}^2$ means we can replace $(\mathtt{i} \to \mathtt{c}) \times (\mathtt{i} \to \mathtt{c}) \simeq \mathtt{i} \to \mathtt{c}^2$ with a more austere type for staged updates, $\mathtt{c} \times (\mathtt{i} \to \mathtt{c})$:

$$\begin{aligned}
&\mathit{toStage} && : ((\mathtt{i} \to \mathtt{c}) \times (\mathtt{i} \to \mathtt{c})) \to (\mathtt{c} \times (\mathtt{i} \to \mathtt{c}))\\
&\mathit{toStage}(\phi, \psi) && := (\{i \in G \times W \mid \phi(i) \neq \varnothing\}, \underbrace{\phi \cup \psi}_{\lambda i.\phi(i) \cup \psi(i)})\\
&\mathit{toBilat} && : (\mathtt{c} \times (\mathtt{i} \to \mathtt{c})) \to ((\mathtt{i} \to \mathtt{c}) \times (\mathtt{i} \to \mathtt{c}))\\
&\mathit{toBilat}(c, \varphi) && := (\lambda i.\{j \in \phi(i) \mid i \in c\}, \lambda i.\{j \in \phi(i) \mid i \notin c\})
\end{aligned}$$

Proposition.

For any ϕ, $\mathit{toBilat}(\mathit{toStage}([\phi]^+, [\phi]^-)) = ([\phi]^+, [\phi]^-)$ $\dashv$

Reiterating: staged updates seem like a good *general* model for DNE. We can recast Krahmer & Muskens's bilateral system as an isomorphic staged update system.

- In fact, *toStage* yields essentially the system of Gotham 2019.

One more iso

The type of staged Krahmer & Muskens updates might as well be $\mathtt{i} \to (2 \times \mathtt{c})$:

$$\begin{aligned}
\mathtt{c} \times (\mathtt{i} \to \mathtt{c}) &= \mathtt{c} \times \mathtt{c}^{\mathtt{i}} && \equiv \\
&\simeq \{\mathtt{i}\} \times \mathtt{c}^{\mathtt{i}} && \textit{def} \\
&\simeq 2^{\mathtt{i}} \times \mathtt{c}^{\mathtt{i}} && \textit{char} \\
&\simeq \mathtt{i} \to (2 \times \mathtt{c}) && \textit{pull}
\end{aligned}$$

The required upgrade, from the DPL type $\mathtt{i} \to \mathtt{c}$ to the DNE-friendly type $\mathtt{i} \to (2 \times \mathtt{c})$, is modest. In fact, the two are substantially closer than $\mathtt{i} \to (2 \times \mathtt{c})$ and $\mathtt{i} \to \mathtt{c}^2$!

Summing up

Projecting Mandelkern's intensions through a few simple isomorphisms reveals a system that explains DNE in terms of staged updates.

Krahmer & Muskens's bilateral semantics was revealed to run on staged updates, too. This system is a minimal extension of DPL, basically identical to Gotham 2019.

These systems seem to share something deep in their DNA.

An abstract view of staged updates

Enrichment

Any semantics that validates DNE and characteristically dynamic patterns of anaphora will constitute a proper **enrichment** of an underlying dynamic(?) system.

Then a **minimal interface** for DNE should include 3 mappings satisfying 3 laws:

$\uparrow : \mathit{Dyn} \to \mathit{Dyn}^\star$	**STAGE**	$\phi^{\uparrow\downarrow} = \phi$	$(\phi^{\downarrow\uparrow} \neq \phi)$
$\downarrow : \mathit{Dyn}^\star \to \mathit{Dyn}$	**RUN**	$\sim\sim\Phi = \Phi$	
$\sim : \mathit{Dyn}^\star \to \mathit{Dyn}^\star$	**NEGATE**	$(\sim\Phi)^\downarrow = \neg\Phi^\downarrow$	

Note: these are mappings between **semantic** domains. To keep things readable, I'm abusing notation and omitting interpretation brackets. E.g., I write $\phi^{\uparrow\downarrow}$ not $[\phi]^{\uparrow\downarrow}$.

The laws require that $\uparrow$ is an **embedding** of *Dyn* in *Dyn**, and that $\sim$ is involutive, and a *Dyn** representation of *Dyn*'s negation.

“Minimal” “interface”?

I mean “interface” in the sense programmers use: as a **specification** that abstractly characterizes a group of operations for some task:

- An interface says something about how these operations should look and behave
- But an interface isn’t a concrete implementation. Many implementations may constitute appropriate realizations, or **instances**.
- Java calls them interfaces; Haskell calls them type classes

And by “minimal”, I mean that the interface identifies a suite of operations and uses them to tightly characterize the behavior we want to model, and nothing else.

- I’ll argue that our minimal interface for DNE can be used to do exactly that
- All other aspects of a concrete implementation are **incidental**

Upgrading your favorite dynamic semantics

With the exception of disjunction (which evaluates Ψ relative to $\sim\Phi$), $\uparrow$ extends to the full language in predictable way, guided by the types:

$$\begin{aligned}
F^{\uparrow} x_1, \ldots, x_n &:= (F\, x_1, \ldots, x_n)^{\uparrow} \\
\exists^{\uparrow} x \Phi &:= (\exists x \Phi^{\downarrow})^{\uparrow} \\
\Phi *^{\uparrow} \Psi &:= (\Phi^{\downarrow} * \Psi^{\downarrow})^{\uparrow} \qquad * \in \{\wedge, \Rightarrow\} \\
\Phi \vee^{\uparrow} \Psi &:= \underbrace{(\Phi^{\downarrow} \vee (\sim\Phi \wedge^{\uparrow} \Psi)^{\downarrow})^{\uparrow}}_{\text{a weak meaning; for strong, replace } \wedge^{\uparrow} \text{ with strong } \Rightarrow^{\uparrow}}
\end{aligned}$$

This semantics inherits whatever “regular” dynamic properties the non-DNE basis has:

- Weak/strong $\Rightarrow$ stays weak/strong
- $\vee$ may (not) be externally dynamic
- Generalized quantifiers are straightforward to upgrade, preserving meaning

Upgrading your favorite dynamic semantics

With the exception of disjunction (which evaluates Ψ relative to $\sim\Phi$), $\uparrow$ extends to the full language in predictable way, guided by the types:

$$\begin{aligned}
F^{\uparrow} x_1, \ldots, x_n &:= (F x_1, \ldots, x_n)^{\uparrow} \\
\exists^{\uparrow} x \Phi &:= (\exists x \Phi^{\downarrow})^{\uparrow} \\
\Phi *^{\uparrow} \Psi &:= (\Phi^{\downarrow} * \Psi^{\downarrow})^{\uparrow} \qquad * \in \{\wedge, \Rightarrow\} \\
\Phi \vee^{\uparrow} \Psi &:= \underbrace{(\Phi^{\downarrow} \vee (\sim\Phi \wedge^{\uparrow} \Psi)^{\downarrow})^{\uparrow}}_{\text{a weak meaning; for strong, replace } \wedge^{\uparrow} \text{ with strong } \Rightarrow^{\uparrow}}
\end{aligned}$$

This semantics inherits whatever "regular" dynamic properties the non-DNE basis has:

- Weak/strong $\Rightarrow$ stays weak/strong
- $\vee$ may (not) be externally dynamic
- Generalized quantifiers are straightforward to upgrade, preserving meaning

```haskell
class DN repr where
  up   :: DPL -> repr               -- down (up m) == m
  down :: repr -> DPL               -- neg (neg m) == m
  neg  :: repr -> repr              -- down (neg m) == DPL.neg (down m)

eq :: DN repr => Var -> Var -> repr
eq x y = up (DPL.eq x y)

ex :: DN repr => Var -> repr -> repr
ex v phi = up (DPL.ex v (down phi))

upCon :: DN repr => (DPL -> DPL -> DPL) -> repr -> repr -> repr
upCon f l r = up (f (down l) (down r))

(/\) :: DN repr => repr -> repr -> repr
(/\) = upCon (DPL./\)

(\/) :: DN repr => repr -> repr -> repr
l \/ r = up (down l DPL.\/ down (neg l /\ r))
```

Some simple equivalences

To save $\uparrow$'s and $\downarrow$'s and stay sane, we write ϕ for the interpretation of ϕ in *Dyn**:

$$
\begin{aligned}
\exists x Fx &= \exists^{\uparrow} x\, Fx \\
&= (\exists x (Fx)^{\downarrow})^{\uparrow} \\
&= (\exists x (Fx)^{\uparrow\downarrow})^{\uparrow} \\
&= (\exists x Fx)^{\uparrow}
\end{aligned}
\qquad
\begin{aligned}
\exists x Fx \wedge Gx &= ((\exists x Fx)^{\downarrow} \wedge (Gx)^{\downarrow})^{\uparrow} \\
&= ((\exists x Fx)^{\uparrow\downarrow} \wedge (Gx)^{\downarrow})^{\uparrow} \\
&= ((\exists x Fx)^{\uparrow\downarrow} \wedge (Gx)^{\uparrow\downarrow})^{\uparrow} \\
&= (\exists x Fx \wedge Gx)^{\uparrow}
\end{aligned}
$$

Proposition.

For ϕ without negation or disjunction, $\phi = \phi^{\uparrow}$ $\dashv$

DNE and Partee disjunctions

DNE follows immediately from the spec: $\neg\neg\phi = {\sim}{\sim}\phi = \phi$.

Partee disjunctions follow from the spec and the definition for disjunction. Given that $\phi \vee \psi = (\phi^{\downarrow} \vee (\neg\phi \wedge \psi)^{\downarrow})^{\uparrow}$, we calculate:

$$\begin{aligned}
\neg\exists xBx \vee Fx &= ((\neg\exists xBx)^{\downarrow} \vee (\neg\neg\exists xBx \wedge Fx)^{\downarrow})^{\uparrow} \\
&= ((\neg\exists xBx)^{\downarrow} \vee (\exists xBx \wedge Fx)^{\downarrow})^{\uparrow} \\
&= (\neg(\exists xBx)^{\downarrow} \vee (\exists xBx \wedge Fx)^{\downarrow})^{\uparrow} \\
&\vdots \\
&= (\neg\exists xBx \vee (\exists xBx \wedge Fx))^{\uparrow}
\end{aligned}$$

Example instance: Krahmer & Muskens 1995

$$\begin{array}{llll} \uparrow : \mathit{Dyn} \to \mathit{Dyn}^\star & \textbf{STAGE} & \phi^{\uparrow\downarrow} = \phi & (\phi^{\downarrow\uparrow} \neq \phi) \\ \downarrow : \mathit{Dyn}^\star \to \mathit{Dyn} & \textbf{RUN} & {\sim}{\sim}\Phi = \Phi & \\ \sim : \mathit{Dyn}^\star \to \mathit{Dyn}^\star & \textbf{NEGATE} & ({\sim}\Phi)^\downarrow = \neg\Phi^\downarrow & \end{array}$$

We take $\mathit{Dyn} ::= \overbrace{\mathtt{i} \to \mathtt{c}}^{\mathit{DPL}}$, and $\mathit{Dyn}^\star ::= \mathit{Dyn} \times \mathit{Dyn}$, with the following concrete values:

- $\phi^\uparrow := (\phi, \neg\phi)$ **STAGE**
- $(\phi, \psi)^\downarrow := \phi$ **RUN**
- ${\sim}(\phi, \psi) := (\psi, \phi)$ **NEGATE**

Aside from the (negotiably weak) semantics for disjunction, under this $\mathit{Dyn}^\star$:

$$\phi = ([\phi]^+, [\phi]^-)$$

Haskell

```haskell
data Squ = Squ {positive :: I -> [I], negative :: I -> [I]}

instance DN Squ where
  up :: DPL -> Squ
  up phi = Squ phi (DPL.neg phi)
  --
  down :: Squ -> DPL
  down (Squ a _) = a
  --
  neg :: Squ -> Squ
  neg (Squ a b) = Squ b a
```

Example instance: staged DPL updates

$$
\begin{array}{llll}
\uparrow : \mathit{Dyn} \to \mathit{Dyn}^\star & \textbf{STAGE} & \phi^{\uparrow\downarrow} = \phi & (\phi^{\downarrow\uparrow} \neq \phi) \\
\downarrow : \mathit{Dyn}^\star \to \mathit{Dyn} & \textbf{RUN} & {\sim}{\sim}\Phi = \Phi & \\
\sim : \mathit{Dyn}^\star \to \mathit{Dyn}^\star & \textbf{NEGATE} & ({\sim}\Phi)^{\downarrow} = \neg\Phi^{\downarrow} &
\end{array}
$$

We take $\mathit{Dyn} ::= \overbrace{\mathtt{i} \to \mathtt{c}}^{\mathit{DPL}}$, and $\mathit{Dyn}^\star ::= \mathtt{c} \times \mathit{Dyn}$, with the following concrete values:

- $\phi^{\uparrow} := (\{i \in G \times W \mid \phi(i) \neq \varnothing\}, \phi \cup \neg\phi)$ **STAGE**
- $(c, \phi)^{\downarrow} := \lambda i.\{j \in \phi(i) \mid i \in c\}$ **RUN**
- ${\sim}(c, \phi) := (G \times W \setminus c, \phi)$ **NEGATE**

Recalling that $(\phi \cup \psi)(i) := \phi(i) \cup \psi(i)$. This is essentially Gotham 2019.

Equivalently, we could take $\mathit{Dyn}^\star ::= \mathtt{i} \to (\mathtt{t} \times \mathtt{c})$

```haskell
data Stage = Stage {static :: I -> T, protoDyn :: I -> [I]}

instance DN Stage where
  up :: DPL -> Stage
  up phi = Stage (\i -> DPL.trueD i phi) (phi <> negD phi)
  --                                    [phi <> psi = \i -> phi i ++ psi i]
  down :: Stage -> DPL
  down (Stage tc f) i = [j | tc i, j <- f i]
  --
  neg :: Stage -> Stage
  neg (Stage tc f) = Stage (\i -> not (tc i)) f

stageSqu :: Stage -> Squ
stageSqu (Stage tc f) = Squ a b where
  a i = [j | tc i, j <- f i]
  b i = [j | not $ tc i, j <- f i]

squStage :: Squ -> Stage                -- stageSqu (squStage (a,b)) = (a,b)
squStage (Squ a b) = Stage tc f where -- squStage (stageSqu (c,u)) = (c,u)
  tc i = DPL.trueD i a
  f = a <> b
```

A different basis: indefinites as variables

Or assume a Heim-style update semantics for a first-order language in lieu of DPL:

$$\begin{aligned} c[Fx] &:= \{i \in c \mid [\![Fx_1, \ldots, x_n]\!]^i\} \\ c[\exists x\, \phi] &:= \{i \in c[\phi] \mid [\![\exists x\, \phi]\!]^i\} \\ c[\phi \wedge \psi] &:= c[\phi][\psi] \\ c[\neg\phi] &:= c \setminus c[\phi] \end{aligned}$$

Extended, if you like, with a Mandelkern-style clause for pronouns:

$$c[F\mathrm{p}_x] := c[Fx] \textit{ if } \forall (g, w) \in c : g(x) \textit{ is defined else undefined}$$

Example instance: Mandelkern 2022

$$
\begin{array}{llll}
\uparrow : \mathit{Dyn} \to \mathit{Dyn}^\star & \textbf{STAGE} & \phi^{\uparrow\downarrow} = \phi & (\phi^{\downarrow\uparrow} \neq \phi) \\
\downarrow : \mathit{Dyn}^\star \to \mathit{Dyn} & \textbf{RUN} & {\sim}{\sim}\Phi = \Phi & \\
\sim : \mathit{Dyn}^\star \to \mathit{Dyn}^\star & \textbf{NEGATE} & ({\sim}\Phi)^{\downarrow} = \neg\Phi^{\downarrow} &
\end{array}
$$

We take $\mathit{Dyn} ::= \overbrace{\mathrm{c} \to \mathrm{c}}^{\textit{Heim}}$, and $\mathit{Dyn}^\star ::= \mathrm{c} \times \mathit{Dyn}$, with the following concrete values:

- $\phi^{\uparrow} := (\{(g, w) \in G \times W \mid \phi\{(g, w)\} \supsetneq \varnothing\}, \phi \cup \neg\phi)$ **STAGE**
- $(c, \phi)^{\downarrow} := \lambda c'.\, \phi(c) \cap \phi(c')$ **RUN**
- ${\sim}(c, \phi) := (G \times W \setminus c, \phi)$ **NEGATE**

And now we have a system very much like Mandelkern 2022 in how it predicts weak readings across-the-board (due to the Heim-style variable treatment of indefinites).

The *Dyn* in $\mathit{Dyn}^\star ::= \mathrm{c} \times \mathit{Dyn}$ is partial, unlike Mandelkern's system. I think this is the most convenient way to set things up, not an essential feature of $\mathit{Dyn}^\star$.

Dynamic?

Mandelkern (2022) writes:

> *There are various precise criteria of dynamicness [e.g., van Benthem 1986, Rothschild & Yalcin 2016] . . . . An interesting question, which I leave for further work, is where my system falls vis-à-vis those criteria, and more generally whether we should think of it as a dynamic theory or not.*

These criteria are formulated for update systems, and so cannot be directly applied to Mandelkern's semantics. But they can be applied to our update-theoretic basis.

- That basis is dynamic, in the sense that it isn't reducible to an intersective system, as it fails to be **eliminative**. There are c for which $c[F\mathrm{p}_x] \nsubseteq c$ (because undefined).
- *Dyn** is a proper enrichment of *Dyn*, and so arguably at least as dynamic.

While eliminativity failures due to p_x terms may be negotiable, ensuring *novelty* for indefinites is necessary, and is bound to lead to eliminativity failures anyhow.

The connection to exceptional scope

Our instance for staged DPL updates takes $\mathit{Dyn} ::= \overbrace{\mathtt{i} \to \mathtt{c}}^{\mathit{DPL}}$, and $\mathit{Dyn}^\star ::= \mathtt{c} \times \mathit{Dyn}$

- We observed that this $\mathit{Dyn}^\star$ is isomorphic to $\mathtt{i} \to (2 \times \mathtt{c})$
- Since $\mathtt{c} ::= \{\mathtt{i}\}$, and $2 \simeq \mathtt{t}$, we have $\mathtt{i} \to (\mathtt{t} \times \{\mathtt{i}\})$

In other work (Charlow 2023) I've argued for dynamic sentence meanings of type $\mathtt{i} \to \{\mathtt{t} \times \mathtt{i}\}$. This looks very similar, but is actually **richer** ($\mathtt{i}$'s with different $\mathtt{t}$'s).

The benefit of $\mathtt{i} \to \{\mathtt{t} \times \mathtt{i}\}$ is a unified explanation of indefinites' exceptional binding scope and **exceptional quantificational scope**:

18. If a rich relative of mine dies, I'll inherit a house.

I speculated that $\mathtt{i} \to \{\mathtt{t} \times \mathtt{i}\}$ was richer than DPL in a way that might help with DNE.

One last instance

$$\begin{array}{llll} \uparrow : \mathit{Dyn} \to \mathit{Dyn}^\star & \textbf{STAGE} & \phi^{\uparrow\downarrow} = \phi & (\phi^{\downarrow\uparrow} \neq \phi) \\ \downarrow : \mathit{Dyn}^\star \to \mathit{Dyn} & \textbf{RUN} & {\sim}{\sim}\Phi = \Phi & \\ \sim : \mathit{Dyn}^\star \to \mathit{Dyn}^\star & \textbf{NEGATE} & ({\sim}\Phi)^{\downarrow} = \neg\Phi^{\downarrow} & \end{array}$$

We take $\mathit{Dyn} ::= \overbrace{\mathtt{i} \to \mathtt{c}}^{\mathit{DPL}}$, and $\mathit{Dyn}^\star ::= \mathtt{i} \to \{\mathtt{t} \times \mathtt{i}\}$, with the following concrete values:

- $\phi^{\uparrow} := \lambda i.\{(b,j) \mid j \in \phi(i)\}$ *if* b *else* $\{(b,i)\}$ *where* $b := \phi(i) \neq \varnothing$ **STAGE**
- $m^{\downarrow} := \lambda i.\{j \mid (\mathbb{T},j) \in m(i)\}$ **RUN**
- ${\sim}m := \lambda i.\{(\neg b,j) \mid (b,j) \in m(i)\}$ **NEGATE**

Overall, we get a modular understanding of which bits of extra structure help where:

$$\underbrace{\{\mathtt{i}\}}_{\mathit{Static}} \ll \overbrace{\mathtt{i} \to \{\mathtt{i}\}}^{\mathit{Dynamic}} \ll \underbrace{\mathtt{i} \to (\mathtt{t} \times \{\mathtt{i}\})}_{\mathit{DNE}} \ll \overbrace{\mathtt{i} \to \{\mathtt{t} \times \mathtt{i}\}}^{\mathit{Exc\ scope}}$$

There are substantial overlaps between this instance and yet another proposal for DNE (Elliott 2020).

Wrapping up

Many accounts of dynamic DNE share a common core:

- They are staged update systems, whose key structural features can be captured with a single abstract characterization.
- Differences are incidental, and in many cases superficial and negotiable.

Maybe this is a little sad, but in a deeper sense it's exhilarating and wonderful! When careful researchers wind up in the same place time and again, we should see it as a highly natural place to be. We have many votes for one approach to DNE.

I haven't discussed the important and very interesting proposals of Hofmann 2019, van den Berg 1996, or Benjamin Spector. Let's see whether and how they fit the proposal here, and if not, what other insights to negation in language they embody.

Barwise, Jon. 1987. Noun phrases, generalized quantifiers, and anaphora. In Peter Gärdenfors (ed.), *Generalized Quantifiers*, 1–29. Dordrecht: Reidel. https://doi.org/10.1007/978-94-009-3381-1_1.

van Benthem, Johan. 1986. *Essays in logical semantics*. Dordrecht: Reidel. https://doi.org/10.1007/978-94-009-4540-1.

van den Berg, Martin H. 1996. *Some aspects of the internal structure of discourse: The dynamics of nominal anaphora*. University of Amsterdam Ph.D. thesis.

Charlow, Simon. 2014. *On the semantics of exceptional scope*. New York University Ph.D. thesis. https://semanticsarchive.net/Archive/2JmMWRjY/.

Charlow, Simon. 2023. Static and dynamic exceptional scope. In press at *Journal of Semantics*. https://ling.auf.net/lingbuzz/004650.

Dekker, Paul. 1994. Predicate logic with anaphora. In Mandy Harvey & Lynn Santelmann (eds.), *Proceedings of Semantics and Linguistic Theory 4*, 79–95. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v4i0.2459.

Elliott, Patrick D. 2020. Towards a principled logic of anaphora. Unpublished ms. https://ling.auf.net/lingbuzz/005562.

Geach, Peter. 1962. *Reference and Generality*. Ithaca, NY: Cornell University Press.

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. https://archive.illc.uva.nl/AC/AC2019/uploaded_files/inlineitem/1AC2019_Proceedings.pdf#page=149.

Groenendijk, Jeroen & Martin Stokhof. 1991a. Dynamic predicate logic. *Linguistics and Philosophy* 14(1). 39–100. https://doi.org/10.1007/BF00628304.

Groenendijk, Jeroen & Martin Stokhof. 1991b. Two theories of dynamic semantics. In Jan van Eijck (ed.), *Logics in AI: European workshop JELIA '90 Amsterdam, The Netherlands, September 10–14, 1990 proceedings*, 55–64. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0018433.

Heim, Irene. 1982. *The semantics of definite and indefinite noun phrases*. University of Massachusetts, Amherst Ph.D. thesis. https://semanticsarchive.net/Archive/Tk0ZmYyY/.

Hofmann, Lisa. 2019. The anaphoric potential of indefinites under negation and disjunction. In Julian J. Schlöder, Dean McHugh & Floris Roelofsen (eds.), *Proceedings of the 22nd Amsterdam Colloquium*, 181–190. https://archive.illc.uva.nl/AC/AC2019/uploaded_files/inlineitem/1AC2019_Proceedings.pdf#page=188.

Karttunen, Lauri. 1976. Discourse referents. In James D. McCawley (ed.), *Syntax and Semantics, volume 7: Notes from the Linguistic Underground*, 363–385. New York: Academic Press.

Krahmer, Emiel & Reinhard Muskens. 1995. Negation and disjunction in Discourse Representation Theory. *Journal of Semantics* 12(4). 357–376. https://doi.org/10.1093/jos/12.4.357.

Lewis, David. 1975. Adverbs of quantification. In Edward L. Keenan (ed.), *Formal Semantics of Natural Language*, 3–15. Cambridge: Cambridge University Press. https://doi.org/10.1017/cbo9780511897696.003.

Mandelkern, Matthew. 2022. Witnesses. *Linguistics and Philosophy* 45(5). 1091–1117. https://doi.org/10.1007/s10988-021-09343-w.

Muskens, Reinhard. 1996. Combining Montague semantics and discourse representation. *Linguistics and Philosophy* 19(2). 143–186. https://doi.org/10.1007/BF00635836.

Potts, Christopher. 2005. *The logic of conventional implicatures*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199273829.001.0001.

Reinhart, Tanya. 1997. Quantifier scope: How labor is divided between QR and choice functions. *Linguistics and Philosophy* 20(4). 335–397. https://doi.org/10.1023/A:1005349801431.

Rooth, Mats. 1987. Noun phrase interpretation in Montague grammar, File Change Semantics, and situation semantics. In Peter Gärdenfors (ed.), *Generalized Quantifiers*, 237–269. Dordrecht: Reidel. https://doi.org/10.1007/978-94-009-3381-1_9.

Rothschild, Daniel & Seth Yalcin. 2016. Three notions of dynamicness in language. *Linguistics and Philosophy* 39(4). 333–355. https://doi.org/10.1007/s10988-016-9188-1.

Stone, Matthew. 1992. *Or* and anaphora. In Chris Barker & David Dowty (eds.), *Proceedings of Semantics and Linguistic Theory 2* (OSU Working Papers in Linguistics 40), 367–385. https://doi.org/10.3765/salt.v2i0.3037.

Veltman, Frank. 1996. Defaults in update semantics. *Journal of Philosophical Logic* 25(3). 221–261. https://doi.org/10.1007/BF00248150.

Yalcin, Seth. 2007. Epistemic modals. *Mind* 116(464). 983–1026. https://doi.org/10.1093/mind/fzm983.