# It all depends: A modern, type-theoretic, compositional dynamic semantics for projection and beyond

Scott Martin
http://coffeeblack.org/

Natural Language Understanding and Artificial Intelligence Laboratory
Nuance Communications

*Dynamic Semantics: Modern Type Theoretic and Category Theoretic Approaches*
Ohio State University, Columbus
October 24, 2015

# This talk

- I'll give an in-depth discussion of a compositional dynamic semantics I have been developing for the past few years, jointly with Carl Pollard

# This talk

- I'll give an in-depth discussion of a compositional dynamic semantics I have been developing for the past few years, jointly with Carl Pollard
- This semantics continues the tradition of dynamic semantics due to Muskens (1996), Beaver (2001) and de Groote (2006), and somewhat more distantly Heim (1982), Groenendijk and Stokhof (1991), and Chierchia (1995)

# This talk

- I'll give an in-depth discussion of a compositional dynamic semantics I have been developing for the past few years, jointly with Carl Pollard
- This semantics continues the tradition of dynamic semantics due to Muskens (1996), Beaver (2001) and de Groote (2006), and somewhat more distantly Heim (1982), Groenendijk and Stokhof (1991), and Chierchia (1995)
- I'll discuss the formal specifics of the framework, which is encoded in dependent type theory
- I'll also show how it can be straightforwardly hooked up to many grammar formalisms, and how it performs empirically on a range of phenomena of interest: anaphora, iterative adverbs, supplements, VP ellipsis, (pseudo)gapping

# What do you mean “It all depends”?

- The framework incorporates the most central intuition of dynamic semantics: utterances are dependent on a context for their interpretation

# What do you mean "It all depends"?

- The framework incorporates the most central intuition of dynamic semantics: utterances are dependent on a context for their interpretation
- But it also uses *dependent types* to enforce certain aspects of the formalization, although in a completely different way than the *propositions-as-types* perspective used in Dependent Type Semantics (which Daisuke and Ribeka will talk about tomorrow)

# What do you mean "It all depends"?

- The framework incorporates the most central intuition of dynamic semantics: utterances are dependent on a context for their interpretation
- But it also uses *dependent types* to enforce certain aspects of the formalization, although in a completely different way than the *propositions-as-types* perspective used in Dependent Type Semantics (which Daisuke and Ribeka will talk about tomorrow)
- I'll explain both of these notions of dependency in a minute

# Talk outline


Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# The framework in brief

- Dynamic Agnostic Semantics (DAS) has been under development in various guises since 2009 (Martin, 2012, 2013, 2015, in press; Kierstead and Martin, 2012; Martin and Pollard, 2012a,b, 2014)

# The framework in brief

- Dynamic Agnostic Semantics (DAS) has been under development in various guises since 2009 (Martin, 2012, 2013, 2015, in press; Kierstead and Martin, 2012; Martin and Pollard, 2012a,b, 2014)
- Its core ideas:

  Dynamic intuitions

  1. Utterances both depend upon and update their context of interpretation
  2. Indefinites don't quantify, but rather introduce *discourse referents* for later discussion

# The framework in brief

- Dynamic Agnostic Semantics (DAS) has been under development in various guises since 2009 (Martin, 2012, 2013, 2015, in press; Kierstead and Martin, 2012; Martin and Pollard, 2012a,b, 2014)
- Its core ideas:

  Dynamic intuitions
  1. Utterances both depend upon and update their context of interpretation
  2. Indefinites don't quantify, but rather introduce *discourse referents* for later discussion

  Compositionality dynamicism extends down to the lexical level; semantic composition occurs in a way familiar to those acquainted with the Montagovian tradition

# The framework in brief

- Dynamic Agnostic Semantics (DAS) has been under development in various guises since 2009 (Martin, 2012, 2013, 2015, in press; Kierstead and Martin, 2012; Martin and Pollard, 2012a,b, 2014)
- Its core ideas:

  Dynamic intuitions
  1. Utterances both depend upon and update their context of interpretation
  2. Indefinites don't quantify, but rather introduce *discourse referents* for later discussion

  Compositionality dynamicism extends down to the lexical level; semantic composition occurs in a way familiar to those acquainted with the Montagovian tradition

  Agnosticism the semantic underpinnings are not necessarily the Montagovian interpretation of possible worlds semantics, but may be hyperintensional

# Talk outline

Dynamic Agnostic Semantics
Agnostic Semantics
Going dynamic
Connecting it to a grammar

Road testing
Projective meaning
Anaphora
Supplements
VP ellipsis and related phenomena

Conclusions and future directions

# How is it agnostic?

The underlying static semantics is the Agnostic Hyperintensional Semantics (AHS) of Pollard (2008, 2015). In this semantics,

- The usual types e (entities), t (truth values), and w (worlds) are available

# How is it agnostic?

The underlying static semantics is the Agnostic Hyperintensional Semantics (AHS) of Pollard (2008, 2015). In this semantics,

- The usual types e (entities), t (truth values), and w (worlds) are available
- There is also the type p, of propositions, which can be defined in several ways:

  Intensional Montagovian $w \to t$
  Extensional Montagovian $t$

# How is it agnostic?

The underlying static semantics is the Agnostic Hyperintensional Semantics (AHS) of Pollard (2008, 2015). In this semantics,

- The usual types e (entities), t (truth values), and w (worlds) are available
- There is also the type p, of propositions, which can be defined in several ways:

  Intensional Montagovian $w \to t$
  Extensional Montagovian t
  Hyperintensional As its own type, with t as its extension

# How is it agnostic?

The underlying static semantics is the Agnostic Hyperintensional Semantics (AHS) of Pollard (2008, 2015). In this semantics,

- The usual types e (entities), t (truth values), and w (worlds) are available
- There is also the type p, of propositions, which can be defined in several ways:

  Intensional Montagovian $w \to t$
  Extensional Montagovian $t$
  Hyperintensional As its own type, with t as its extension
  Some other way

# Senses and their extensions

- The *sense types* are the types that can be formed from e and p using the constructors $\rightarrow$ and $\times$

# Senses and their extensions

- The *sense types* are the types that can be formed from e and p using the constructors $\rightarrow$ and $\times$
- The *extension types* associated with senses are

$$\begin{aligned}
\mathrm{Ext}(\mathrm{e}) &=_{\mathrm{def}} \mathrm{e} \\
\mathrm{Ext}(\mathrm{p}) &=_{\mathrm{def}} \mathrm{t} \\
\mathrm{Ext}(A \rightarrow B) &=_{\mathrm{def}} A \rightarrow \mathrm{Ext}(B) \\
\mathrm{Ext}(A \times B) &=_{\mathrm{def}} \mathrm{Ext}(A) \times \mathrm{Ext}(B)
\end{aligned}$$

- For example, the extension type of $\mathrm{e} \rightarrow \mathrm{p}$ (the sense of unary properties) is $\mathrm{e} \rightarrow \mathrm{t}$ (sets of entities)

# Senses and their extensions

- The *sense types* are the types that can be formed from e and p using the constructors $\to$ and $\times$
- The *extension types* associated with senses are

$$\begin{aligned} \mathrm{Ext}(\mathrm{e}) &=_{\mathrm{def}} \mathrm{e} \\ \mathrm{Ext}(\mathrm{p}) &=_{\mathrm{def}} \mathrm{t} \\ \mathrm{Ext}(A \to B) &=_{\mathrm{def}} A \to \mathrm{Ext}(B) \\ \mathrm{Ext}(A \times B) &=_{\mathrm{def}} \mathrm{Ext}(A) \times \mathrm{Ext}(B) \end{aligned}$$

- For example, the extension type of $\mathrm{e} \to \mathrm{p}$ (the sense of unary properties) is $\mathrm{e} \to \mathrm{t}$ (sets of entities)
- Then the agnosticism is maintained by adding an abstraction layer, the *extension functions* $@_A : A \to \mathrm{w} \to \mathrm{Ext}(A)$, for every sense type $A$
- So for any proposition $p$ and world $w$, $(p \mathbin{@_{\mathrm{p}}} w)$ in principle gives the truth value of $p$ at $w$

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \to \mathrm{p} \to \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \to \mathrm{p} \to \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world
- From these, we can derive that two propositions $p$ and $q$ entail each other just in case they are equivalent

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \to \mathrm{p} \to \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world
- From these, we can derive that two propositions $p$ and $q$ entail each other just in case they are equivalent
- However, mutual entailment between $p$ and $q$ does not require that $p = q$!

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \to \mathrm{p} \to \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world
- From these, we can derive that two propositions $p$ and $q$ entail each other just in case they are equivalent
- However, mutual entailment between $p$ and $q$ does not require that $p = q$!
- We can opt for Montagovian intensionality by defining the type p as $\mathrm{w} \to \mathrm{t}$ (sets of worlds), and the extension function $@_{\mathrm{p}}$ as set membership, i.e. as $\lambda_p \lambda_w.(p\, w)$

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \rightarrow \mathrm{p} \rightarrow \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world
- From these, we can derive that two propositions $p$ and $q$ entail each other just in case they are equivalent
- However, mutual entailment between $p$ and $q$ does not require that $p = q$!
- We can opt for Montagovian intensionality by defining the type p as $\mathrm{w} \rightarrow \mathrm{t}$ (sets of worlds), and the extension function $@_{\mathrm{p}}$ as set membership, i.e. as $\lambda_p \lambda_w.(p\, w)$
- But then equivalence and equality collapse together, and several unsavory, recalcitrant problems reappear (see Pollard, 2008, 2015; Plummer and Pollard, 2012)

# Entailment and equivalence

- Entailment is encoded by $\mathsf{entails} : \mathrm{p} \rightarrow \mathrm{p} \rightarrow \mathrm{t}$, so that $p$ $\mathsf{entails}$ $q$ iff for every world $w$, $q$ is true at $w$ provided $p$ is
- Propositional equivalence is defined so that $p \equiv q$ iff $p$ and $q$ have the same extension at every world
- From these, we can derive that two propositions $p$ and $q$ entail each other just in case they are equivalent
- However, mutual entailment between $p$ and $q$ does not require that $p = q$!
- We can opt for Montagovian intensionality by defining the type p as $\mathrm{w} \rightarrow \mathrm{t}$ (sets of worlds), and the extension function $@_{\mathrm{p}}$ as set membership, i.e. as $\lambda_p \lambda_w.(p\,w)$
- But then equivalence and equality collapse together, and several unsavory, recalcitrant problems reappear (see Pollard, 2008, 2015; Plummer and Pollard, 2012)
- We could also opt for hyperintensionality, and define worlds as maximal consistent sets of propositions

# Business as usual

- Fortunately, we can also avoid the choice entirely by staying agnostic on how $@_p$ is defined, and still go about our semantic business

# Business as usual

- Fortunately, we can also avoid the choice entirely by staying agnostic on how $@_p$ is defined, and still go about our semantic business
- All the necessary connectives and quantifiers can be defined at the sense level in terms of $@_p$: true, false, and, not, implies, or, forall, exists

# Business as usual

- Fortunately, we can also avoid the choice entirely by staying agnostic on how $@_p$ is defined, and still go about our semantic business
- All the necessary connectives and quantifiers can be defined at the sense level in terms of $@_p$: true, false, and, not, implies, or, forall, exists
- All the usual relations needed to model natural language can be encoded as AHS senses

  Unary properties cyclist : $e \to p$
  Binary relations love : $e \to e \to p$
  Ternary relations give : $e \to e \to e \to p$
  Propositional attitudes believe : $p \to e \to p$

# Business as usual

- Fortunately, we can also avoid the choice entirely by staying agnostic on how $@_p$ is defined, and still go about our semantic business
- All the necessary connectives and quantifiers can be defined at the sense level in terms of $@_p$: true, false, and, not, implies, or, forall, exists
- All the usual relations needed to model natural language can be encoded as AHS senses

  Unary properties cyclist : $e \to p$
  Binary relations love : $e \to e \to p$
  Ternary relations give : $e \to e \to e \to p$
  Propositional attitudes believe : $p \to e \to p$
- Standard treatments of modality can also be developed inside AHS, but I won't bother with the details here

# Talk outline


Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# How it all started

- In the summer of 2009, Carl Pollard, Craige Roberts, Elizabeth Smith and I started reading papers about dynamic semantics

# How it all started

- In the summer of 2009, Carl Pollard, Craige Roberts, Elizabeth Smith and I started reading papers about dynamic semantics
- The goal was to build a better mousetrap that could model projective meaning: anaphora, Potts's (2005) "CIs", etc.
- Everything on the market at that time seemed overly laden with definitions, too complicated at the type level, out of touch with the core intuitions, or too reliant on aspects of the model theory

# How it all started

- In the summer of 2009, Carl Pollard, Craige Roberts, Elizabeth Smith and I started reading papers about dynamic semantics
- The goal was to build a better mousetrap that could model projective meaning: anaphora, Potts's (2005) "CIs", etc.
- Everything on the market at that time seemed overly laden with definitions, too complicated at the type level, out of touch with the core intuitions, or too reliant on aspects of the model theory
- We wanted to mix the nice features of various dynamic semantics
  - Contexts as first-class objects that can be extended (de Groote, 2006)
  - Meanings explicitly modeled as functions that both consume and output contexts (Heim, 1982; Groenendijk and Stokhof, 1991; Muskens, 1996; Beaver, 2001; de Groote, 2006)
  - Fully compositional, with all the semantic work handled by lambdas (Muskens, 1996; Beaver, 2001; de Groote, 2006)
  - Systematic 'lifting' from static to dynamic semantics (Groenendijk and Stokhof, 1990; Chierchia, 1995)

# Contexts

- Many previous frameworks pass around sets or lists of entities, but true context dependence also requires (at least) access to entailments triggered previously

# Contexts

- Many previous frameworks pass around sets or lists of entities, but true context dependence also requires (at least) access to entailments triggered previously
- So, thinking of a discourse context as a set of propositions containing free variables for the unknown identities of the discourse referents, we came up with

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to \mathrm{p}$$

as the type of contexts

# Contexts

- Many previous frameworks pass around sets or lists of entities, but true context dependence also requires (at least) access to entailments triggered previously
- So, thinking of a discourse context as a set of propositions containing free variables for the unknown identities of the discourse referents, we came up with

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \rightarrow \mathrm{p}$$

  as the type of contexts
- That is, a context that is about $n$ discourse referents is modeled as a function from a vector of $n$ entities (its *arity*) to a proposition

# Contexts

- Many previous frameworks pass around sets or lists of entities, but true context dependence also requires (at least) access to entailments triggered previously
- So, thinking of a discourse context as a set of propositions containing free variables for the unknown identities of the discourse referents, we came up with

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \rightarrow \mathrm{p}$$

  as the type of contexts
- That is, a context that is about $n$ discourse referents is modeled as a function from a vector of $n$ entities (its *arity*) to a proposition
- Then the role of discourse referents is handled by the natural number indices into a context's input vector

# Contexts

- Many previous frameworks pass around sets or lists of entities, but true context dependence also requires (at least) access to entailments triggered previously
- So, thinking of a discourse context as a set of propositions containing free variables for the unknown identities of the discourse referents, we came up with

  $$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to \mathrm{p}$$

  as the type of contexts
- That is, a context that is about $n$ discourse referents is modeled as a function from a vector of $n$ entities (its *arity*) to a proposition
- Then the role of discourse referents is handled by the natural number indices into a context's input vector
- For example, the 2-context

  $$\lambda_{x,y}.(\mathsf{cyclist}\,x)\ \mathsf{and}\ (\mathsf{wheel}\,y)\ \mathsf{and}\ (\mathsf{break}\,y\,x) : \mathrm{c}_2$$

  would correspond to an utterance of *Some cyclist broke a wheel*

# Contents

- Keeping with a long-standing tradition in dynamic semantics of (implicitly or explicitly) modeling meanings as functions from contexts to contexts, our *contents*, at a first cut, have the type

$$c \to c$$

# Contents

- Keeping with a long-standing tradition in dynamic semantics of (implicitly or explicitly) modeling meanings as functions from contexts to contexts, our *contents*, at a first cut, have the type

$$\mathrm{c} \to \mathrm{c}$$

- But since contexts have arities, there's a subtlety: a content may *introduce* discourse referents, increasing the arity of the output context over the input

# Contents

- Keeping with a long-standing tradition in dynamic semantics of (implicitly or explicitly) modeling meanings as functions from contexts to contexts, our *contents*, at a first cut, have the type

$$c \to c$$

- But since contexts have arities, there's a subtlety: a content may *introduce* discourse referents, increasing the arity of the output context over the input
- Another subtlety: dynamic properties will need to take natural numbers (discourse referents) as arguments, but how can we ensure that the context of interpretation actually *has* such a referent?

# Enter dependent types

- To handle these issues, the type theory is extended to use *dependent types* parameterized by the natural numbers (type n)

# Enter dependent types

- To handle these issues, the type theory is extended to use *dependent types* parameterized by the natural numbers (type n)
- Here, *products* $\Pi_{x:A}.B$ generalize simple type-theoretic functions, and *sums* $\Sigma_{x:A}.B$ generalize simply-typed cartesian products
- So $\Pi_{x:A}.B$ is a function from $A$ to $B$ where the type $B$ may depend on the value of $x$, and $\Sigma_{x:A}.B$ is a pair where the second component's type $B$ may depend on the first component $x$

# Enter dependent types

- To handle these issues, the type theory is extended to use *dependent types* parameterized by the natural numbers (type n)
- Here, *products* $\Pi_{x:A}.B$ generalize simple type-theoretic functions, and *sums* $\Sigma_{x:A}.B$ generalize simply-typed cartesian products
- So $\Pi_{x:A}.B$ is a function from $A$ to $B$ where the type $B$ may depend on the value of $x$, and $\Sigma_{x:A}.B$ is a pair where the second component's type $B$ may depend on the first component $x$
- Then the simple type-theoretic constructors represent the special case where no dependency is present:

$$A \to B =_{\text{def}} \Pi_{x:A}.B \qquad (x \text{ not free in } B)$$

$$A \times B =_{\text{def}} \Sigma_{x:A}.B \qquad (x \text{ not free in } B)$$

# Dependently-typed contexts and contents

- With dependent types, we still write the type of contexts as $\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \rightarrow \mathrm{p}$

# Dependently-typed contexts and contents

- With dependent types, we still write the type of contexts as $\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to \mathrm{p}$
- But the type of contents is

$$\mathrm{k}_n =_{\mathrm{def}} \Pi_{c:\mathrm{c}_m}.\mathrm{c}_{m+n}$$

- So $\mathrm{k}_n$ is the type of contents that introduce $n$ referents (the *degree*)

# Dependently-typed contexts and contents

- With dependent types, we still write the type of contexts as $\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to \mathrm{p}$
- But the type of contents is

$$\mathrm{k}_n =_{\mathrm{def}} \Pi_{c:\mathrm{c}_m}.\mathrm{c}_{m+n}$$

- So $\mathrm{k}_n$ is the type of contents that introduce $n$ referents (the *degree*)
- The types of contexts with arity at least $n$ is encoded as

$$\mathrm{c}_{\geq n} =_{\mathrm{def}} \Sigma_{m:\mathrm{n}}.\mathrm{c}_{m+n} \ ,$$

and $\mathrm{c}_{>n} =_{\mathrm{def}} \mathrm{c}_{\geq n+1}$ the type of contexts whose arity is strictly greater than $n$

# Dependently-typed contexts and contents

- With dependent types, we still write the type of contexts as $\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to \mathrm{p}$
- But the type of contents is

$$\mathrm{k}_n =_{\mathrm{def}} \Pi_{c:\mathrm{c}_m}.\mathrm{c}_{m+n}$$

- So $\mathrm{k}_n$ is the type of contents that introduce $n$ referents (the *degree*)
- The types of contexts with arity at least $n$ is encoded as

$$\mathrm{c}_{\geq n} =_{\mathrm{def}} \Sigma_{m:\mathrm{n}}.\mathrm{c}_{m+n} ,$$

and $\mathrm{c}_{>n} =_{\mathrm{def}} \mathrm{c}_{\geq n+1}$ the type of contexts whose arity is strictly greater than $n$

- We also write the types of contexts of any arity and contents of any degree as follows:

$$\mathrm{c} =_{\mathrm{def}} \Sigma_{n:\mathrm{n}}.\mathrm{c}_n$$

$$\mathrm{k} =_{\mathrm{def}} \Sigma_{n:\mathrm{n}}.\mathrm{k}_n$$

# Dynamic properties with dependent types

- Defining *n*-ary static properties is straightforward:

$$\mathrm{p}_0 =_{\mathrm{def}} \mathrm{p}$$
$$\mathrm{p}_{n+1} =_{\mathrm{def}} \mathrm{e} \to \mathrm{p}_n$$

# Dynamic properties with dependent types

- Defining *n*-ary static properties is straightforward:

$$\mathrm{p}_0 =_{\mathrm{def}} \mathrm{p}$$
$$\mathrm{p}_{n+1} =_{\mathrm{def}} \mathrm{e} \to \mathrm{p}_n$$

- But for the dynamic case, we have to worry that there actually are slots in the input context for the discourse referent arguments

# Dynamic properties with dependent types

- Defining *n*-ary static properties is straightforward:

$$\mathrm{p}_0 =_{\mathrm{def}} \mathrm{p}$$
$$\mathrm{p}_{n+1} =_{\mathrm{def}} \mathrm{e} \to \mathrm{p}_n$$

- But for the dynamic case, we have to worry that there actually are slots in the input context for the discourse referent arguments
- With dependent types, we can state the required constraint:

$$\mathrm{d}_{0,i,j} =_{\mathrm{def}} \Pi_{c:\mathrm{c}_{>i}}.\mathrm{c}_{|c|+j}$$
$$\mathrm{d}_{n+1,i,j} =_{\mathrm{def}} \Pi_{m:\mathrm{n}}.\mathrm{d}_{n,(\max\, i\, m),j}$$

- So fully-saturated dynamic properties are contents with a constraint that the input context have the right number of discourse referents

# Dynamic properties with dependent types

- Defining *n*-ary static properties is straightforward:

$$\mathrm{p}_0 =_{\mathrm{def}} \mathrm{p}$$
$$\mathrm{p}_{n+1} =_{\mathrm{def}} \mathrm{e} \rightarrow \mathrm{p}_n$$

- But for the dynamic case, we have to worry that there actually are slots in the input context for the discourse referent arguments
- With dependent types, we can state the required constraint:

$$\mathrm{d}_{0,i,j} =_{\mathrm{def}} \Pi_{c:\mathrm{c}_{>i}}.\mathrm{c}_{|c|+j}$$
$$\mathrm{d}_{n+1,i,j} =_{\mathrm{def}} \Pi_{m:\mathrm{n}}.\mathrm{d}_{n,(\max\ i\, m),j}$$

- So fully-saturated dynamic properties are contents with a constraint that the input context have the right number of discourse referents
- For each $n$, there is also the disjoint union type $\mathrm{d}_n =_{\mathrm{def}} \Sigma_{i:\mathrm{n}}\Sigma_{j:\mathrm{n}}.\mathrm{d}_{n,i,j}$ over all the types $\mathrm{d}_{n,i,j}$

# Dynamicization

- The *dynamicizer* functions $\mathrm{dyn}_{n,i} : \mathrm{p}_n \to \mathrm{d}_{n,i,0}$ lift static properties to dynamic ones:

$$\mathrm{dyn}_{0,i} =_{\mathrm{def}} \lambda_{p:\mathrm{p}_0} \lambda_{c:\mathrm{c}_{>i}} \lambda_{\mathbf{x}^{|c|}} . p$$
$$\mathrm{dyn}_{n+1,i} =_{\mathrm{def}} \lambda_{R:\mathrm{p}_{n+1}} \lambda_{m:\mathrm{n}} . \mathrm{dyn}_{n,(\max\, i\, m)} \,(R\, x_m)$$

# Dynamicization

- The *dynamicizer* functions $\text{dyn}_{n,i} : \mathrm{p}_n \to \mathrm{d}_{n,i,0}$ lift static properties to dynamic ones:

$$\text{dyn}_{0,i} =_{\text{def}} \lambda_{p:\mathrm{p}_0} \lambda_{c:\mathrm{c}_{>i}} \lambda_{\mathbf{x}^{|c|}} . p$$
$$\text{dyn}_{n+1,i} =_{\text{def}} \lambda_{R:\mathrm{p}_{n+1}} \lambda_{m:\mathrm{n}} . \text{dyn}_{n,(\max\, i\, m)}\,(R\, x_m)$$

- Some examples:

$$(\text{dyn}_{0,0}\, \mathsf{rain}) = \lambda_{c:\mathrm{c}_{>0}} \lambda_{\mathbf{x}^{|c|}} . \mathsf{rain}$$
$$(\text{dyn}_{1,0}\, \mathsf{cyclist}) = \lambda_{n:\mathrm{n}} \lambda_{c:\mathrm{c}_{>n}} \lambda_{\mathbf{x}^{|c|}} . (\mathsf{cyclist}\, x_n)$$
$$(\text{dyn}_{2,0}\, \mathsf{break}) = \lambda_{m:\mathrm{n}} \lambda_{n:\mathrm{n}} \lambda_{c:\mathrm{c}_{>(\max\, m\, n)}} \lambda_{\mathbf{x}^{|c|}} . (\mathsf{break}\, x_m\, x_n)$$
$$(\text{dyn}_{3,0}\, \mathsf{give}) = \lambda_{k:\mathrm{n}} \lambda_{m:\mathrm{n}} \lambda_{n:\mathrm{n}} \lambda_{c:\mathrm{c}_{>(\max\, k\, m\, n)}} \lambda_{\mathbf{x}^{|c|}} . (\mathsf{give}\, x_k\, x_m\, x_n)$$

# Updates and context change

- Contents are distinguished from *updates*, which have the same type:
  $\mathrm{u}_n =_{\mathrm{def}} \mathrm{k}_n$

# Updates and context change

- Contents are distinguished from *updates*, which have the same type: $\mathrm{u}_n =_{\mathrm{def}} \mathrm{k}_n$
- A content $k$ is promoted to an update by the *context change* function $\mathsf{cc} : \mathrm{k}_n \to \mathrm{u}_n$:

$$\mathsf{cc} =_{\mathrm{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|k|}}.(c\,\mathbf{x}) \text{ and } (k\,c\,\mathbf{x},\mathbf{y})$$

- That is, the update $(\mathsf{cc}\,k)$ has the same content as $k$, but also incorporates the information from the input context

# Updates and context change

- Contents are distinguished from *updates*, which have the same type: $\mathrm{u}_n =_{\mathrm{def}} \mathrm{k}_n$
- A content $k$ is promoted to an update by the *context change* function $\mathsf{cc} : \mathrm{k}_n \to \mathrm{u}_n$:

$$\mathsf{cc} =_{\mathrm{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}, \mathbf{y}^{|k|}} . (c\, \mathbf{x}) \text{ and } (k\, c\, \mathbf{x}, \mathbf{y})$$

- That is, the update $(\mathsf{cc}\, k)$ has the same content as $k$, but also incorporates the information from the input context
- For example, defining RAIN as the content $(\mathrm{dyn}_{0,0}\ \mathsf{rain})$,

$$(\mathsf{cc}\, \mathrm{RAIN}) = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}} . (c\, \mathbf{x}) \text{ and rain}$$

# Updates and context change

- Contents are distinguished from *updates*, which have the same type: $\mathrm{u}_n =_{\mathrm{def}} \mathrm{k}_n$
- A content $k$ is promoted to an update by the *context change* function $\mathsf{cc} : \mathrm{k}_n \to \mathrm{u}_n$:

$$\mathsf{cc} =_{\mathrm{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}, \mathbf{y}^{|k|}}.(c\, \mathbf{x}) \text{ and } (k\, c\, \mathbf{x}, \mathbf{y})$$

- That is, the update $(\mathsf{cc}\, k)$ has the same content as $k$, but also incorporates the information from the input context
- For example, defining RAIN as the content $(\mathsf{dyn}_{0,0}\, \mathsf{rain})$,

$$(\mathsf{cc}\, \text{RAIN}) = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.(c\, \mathbf{x}) \text{ and rain}$$

- The cc function models the process of making an at-issue proposal, i.e., proffering a content (cf. Roberts, 2012b)

# Existential 'quantifier'

- A prerequisite for the dynamic existential is the *context extension* function, which extends a context with a new coordinate $y$:

$$(\cdot)^+ =_{\text{def}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|},y}. c\,\mathbf{x} \qquad (y \text{ not in } \mathbf{x} \text{ or free in } (c\,\mathbf{x}))$$

# Existential 'quantifier'

- A prerequisite for the dynamic existential is the *context extension* function, which extends a context with a new coordinate $y$:

$$(\cdot)^+ =_{\text{def}} \lambda_{c:\text{c}} \lambda_{\mathbf{x}^{|c|}, y} . c\, \mathbf{x} \qquad (y \text{ not in } \mathbf{x} \text{ or free in } (c\, \mathbf{x}))$$

- Then the existential $\text{EXISTS} : \Pi_{D:\text{d}_{1,i,j}} . \text{k}_{j+1}$ just adds a new discourse referent using $(\cdot)^+$, and passes it to its argument property:

$$\text{EXISTS} =_{\text{def}} \lambda_{D:\text{d}_{1,i,j}} \lambda_{c:\text{c}} . D\ |c|\ c^+$$

# Existential 'quantifier'

- A prerequisite for the dynamic existential is the *context extension* function, which extends a context with a new coordinate $y$:

$$(\cdot)^{+} =_{\text{def}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}, y}. c\, \mathbf{x} \qquad (y \text{ not in } \mathbf{x} \text{ or free in } (c\, \mathbf{x}))$$

- Then the existential $\textsc{exists} : \Pi_{D:\mathrm{d}_{1,i,j}}.\mathrm{k}_{j+1}$ just adds a new discourse referent using $(\cdot)^{+}$, and passes it to its argument property:

$$\textsc{exists} =_{\text{def}} \lambda_{D:\mathrm{d}_{1,i,j}} \lambda_{c:\mathrm{c}}. D\ |c|\ c^{+}$$

- For example, letting $\textsc{wheel} =_{\text{def}} (\mathsf{dyn}_{1,0}\, \mathsf{wheel})$, the meaning of *There's a wheel* would be

$$\begin{aligned} \textsc{exists}\, \textsc{wheel} &= \lambda_{c:\mathrm{c}}. \textsc{wheel}\ |c|\ c^{+} \\ &= \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}, y}. (\mathsf{wheel}\, (\mathbf{x}, y)_{|c|}) \\ &= \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}, y}. (\mathsf{wheel}\, y) \end{aligned}$$

# Dynamic conjunction

- As usual in dynamic semantics, conjunction is asymmetric, with the second conjunct interpreted 'after' the first conjunct has a chance to modify the input context

$$\text{AND} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|h|},\mathbf{z}^{|k|}}.(h\,c\,\mathbf{x},\mathbf{y})\ \mathsf{and}\ (k\,(\mathsf{cc}\,h\,c)\,\mathbf{x},\mathbf{y},\mathbf{z})$$

# Dynamic conjunction

- As usual in dynamic semantics, conjunction is asymmetric, with the second conjunct interpreted 'after' the first conjunct has a chance to modify the input context

$$\text{AND} =_{\text{def}} \lambda_{h:\text{k}}\lambda_{k:\text{k}}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|h|},\mathbf{z}^{|k|}}.(h\,c\,\mathbf{x},\mathbf{y})\ \mathsf{and}\ (k\,(\mathsf{cc}\,h\,c)\,\mathbf{x},\mathbf{y},\mathbf{z})$$

- The first conjunct $h$ is interpreted with respect to the input context

# Dynamic conjunction

- As usual in dynamic semantics, conjunction is asymmetric, with the second conjunct interpreted 'after' the first conjunct has a chance to modify the input context

$$\text{AND} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|h|},\mathbf{z}^{|k|}}.(h\,c\,\mathbf{x},\mathbf{y}) \text{ and } (k\,(\mathrm{cc}\,h\,c)\,\mathbf{x},\mathbf{y},\mathbf{z})$$

- The first conjunct $h$ is interpreted with respect to the input context
- But the second is interpreted in the context $(\mathrm{cc}\,h\,c)$ that results from updating the input context with $h$'s content

# Dynamic conjunction

- As usual in dynamic semantics, conjunction is asymmetric, with the second conjunct interpreted 'after' the first conjunct has a chance to modify the input context

$$\text{AND} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|h|},\mathbf{z}^{|k|}}.(h\, c\, \mathbf{x}, \mathbf{y})\ \mathsf{and}\ (k\, (\mathsf{cc}\, h\, c)\, \mathbf{x}, \mathbf{y}, \mathbf{z})$$

- The first conjunct $h$ is interpreted with respect to the input context
- But the second is interpreted in the context $(\mathsf{cc}\, h\, c)$ that results from updating the input context with $h$'s content
- So generating CYCLIST via $(\mathrm{dyn}_{1,0}\ \mathsf{cyclist})$, we get a model of *There's a cyclist and there's a wheel* as

$$\text{EXISTS CYCLIST AND EXISTS WHEEL}$$
$$= \lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},y,z}.(\mathsf{cyclist}\, y)\ \mathsf{and}\ (\mathsf{wheel}\, z)$$

# Dynamic negation

- Famously, dynamic negation traps discourse referents introduced in its scope, making them unavailable for future reference

# Dynamic negation

- Famously, dynamic negation traps discourse referents introduced in its scope, making them unavailable for future reference
- This is achieved by existentially binding the referents in the content passed to $\textsc{not} : \mathrm{k}_n \to \mathrm{k}_0$, reminiscent of Heim's (1982) "existential closure":

$$\textsc{not} =_{\text{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}} . \mathsf{not}\, \mathsf{exists}_{\mathbf{y}^{|k|}} . (k\, c\, \mathbf{x}, \mathbf{y})$$

# Dynamic negation

- Famously, dynamic negation traps discourse referents introduced in its scope, making them unavailable for future reference
- This is achieved by existentially binding the referents in the content passed to $\textsc{not} : \mathrm{k}_n \to \mathrm{k}_0$, reminiscent of Heim's (1982) "existential closure":

$$\textsc{not} =_{\text{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}} . \mathsf{not}\,\mathsf{exists}_{\mathbf{y}^{|k|}} . (k\, c\, \mathbf{x}, \mathbf{y})$$

- So for *It's not raining*, we get

$$\textsc{not}\,\textsc{rain} = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}} . \mathsf{not}\,\mathsf{rain}$$

- But the model of *There's no wheel* is

$$\textsc{not}\,(\textsc{exists}\,\textsc{wheel}) = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}} . \mathsf{not}\,\mathsf{exists}_{y} . (\mathsf{wheel}\; y)$$

# Dynamic negation

- Famously, dynamic negation traps discourse referents introduced in its scope, making them unavailable for future reference
- This is achieved by existentially binding the referents in the content passed to NOT : $\mathrm{k}_n \to \mathrm{k}_0$, reminiscent of Heim's (1982) "existential closure":

$$\text{NOT} =_{\text{def}} \lambda_{k:\mathrm{k}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.\mathsf{not}\,\mathsf{exists}_{\mathbf{y}^{|k|}}.(k\,c\,\mathbf{x},\mathbf{y})$$

- So for *It's not raining*, we get

$$\text{NOT RAIN} = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.\mathsf{not}\,\mathsf{rain}$$

- But the model of *There's no wheel* is

$$\text{NOT}\,(\text{EXISTS WHEEL}) = \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.\mathsf{not}\,\mathsf{exists}_y.(\mathsf{wheel}\,y)$$

- The importance of this definition will become apparent when we get to anaphoric accessibility

# Other dynamic connectives, quantifiers, and determiners

- With AND, EXISTS, and NOT, we can define other connectives:

$$\text{THAT} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}\lambda_{n:\mathrm{n}}.(D\,n)\ \text{AND}\ (E\,n)$$
$$\text{OR} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.\text{NOT}\,((\text{NOT}\,h)\ \text{AND}\ (\text{NOT}\,k))$$
$$\text{IMPLIES} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.(\text{NOT}\,h)\ \text{OR}\ (h\ \text{AND}\ k)$$

# Other dynamic connectives, quantifiers, and determiners

- With AND, EXISTS, and NOT, we can define other connectives:

$$\text{THAT} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}\lambda_{n:\mathrm{n}}.(D\,n)\ \text{AND}\ (E\,n)$$
$$\text{OR} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.\text{NOT}\,((\text{NOT}\,h)\ \text{AND}\ (\text{NOT}\,k))$$
$$\text{IMPLIES} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.(\text{NOT}\,h)\ \text{OR}\ (h\ \text{AND}\ k)$$

- Other quantifiers:

$$\text{FORALL} =_{\text{def}} \lambda_{D:\mathrm{d}_1}.\text{NOT}\ \text{EXISTS}_n.\text{NOT}\ (D\,n)$$

# Other dynamic connectives, quantifiers, and determiners

- With AND, EXISTS, and NOT, we can define other connectives:

$$\text{THAT} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}\lambda_{n:\mathrm{n}}.(D\,n)\ \text{AND}\ (E\,n)$$
$$\text{OR} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.\text{NOT}\,((\text{NOT}\,h)\ \text{AND}\ (\text{NOT}\,k))$$
$$\text{IMPLIES} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.(\text{NOT}\,h)\ \text{OR}\ (h\ \text{AND}\ k)$$

- Other quantifiers:

$$\text{FORALL} =_{\text{def}} \lambda_{D:\mathrm{d}_1}.\text{NOT}\,\text{EXISTS}_n.\text{NOT}\,(D\,n)$$

- And, in turn, dynamic versions of the static determiners:

$$\text{A} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}.\text{EXISTS}_n.(D\,n)\ \text{AND}\ (E\,n)$$
$$\text{NO} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}.\text{NOT}\,(\text{A}\,D\,E)$$
$$\text{EVERY} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}.\text{FORALL}_n.(D\,n)\ \text{IMPLIES}\ (E\,n)$$

# Weak readings and the proportion problem

- The definition of dynamic implication IMPLIES may seem a bit roundabout, but it is an implementation of Chierchia's (1995) *dynamic conservativity*, since the antecedent's content is copied into the consequent

$$\text{IMPLIES} =_{\text{def}} \lambda_{h:\mathrm{k}}\lambda_{k:\mathrm{k}}.(\text{NOT}\, h) \text{ OR } (h \text{ AND}\, k)$$

- The effect of this definition is that donkey sentences get the so-called *weak* reading by default, avoiding the *proportion problem* (which Ribeka will also talk about tomorrow)

# Weak readings and the proportion problem

- The definition of dynamic implication IMPLIES may seem a bit roundabout, but it is an implementation of Chierchia's (1995) *dynamic conservativity*, since the antecedent's content is copied into the consequent

$$\text{IMPLIES} =_{\text{def}} \lambda_{h:\text{k}}\lambda_{k:\text{k}}.(\text{NOT}\, h) \text{ OR } (h \text{ AND } k)$$

- The effect of this definition is that donkey sentences get the so-called *weak* reading by default, avoiding the *proportion problem* (which Ribeka will also talk about tomorrow)
- For example, the weak reading of

(1) Everyone with a quarter in their pocket put it in the meter.

does not require that everyone deposited all their change into the meter, only that everyone put *at least one* quarter into the meter

# Modeling discourse

- Going beyond the utterance level, updates are combined by the *parataxis* operation, which is just function composition written in the other order:

$$\circ =_{\text{def}} \lambda_{u:\mathrm{u}} \lambda_{v:\mathrm{u}} \lambda_{c:\mathrm{c}} . v\,(u\,c)$$

# Modeling discourse

- Going beyond the utterance level, updates are combined by the *parataxis* operation, which is just function composition written in the other order:

$$\circ =_{\text{def}} \lambda_{u:\mathrm{u}} \lambda_{v:\mathrm{u}} \lambda_{c:\mathrm{c}} . v\,(u\,c)$$

- So the model of the mini-discourse *It was raining. A cyclist left.* is the composed update

$$(\mathsf{cc}\ \text{RAIN}) \circ \mathsf{cc}\ (\text{A CYCLIST LEAVE})$$
$$= \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|},y} . (c\,\mathbf{x})\ \mathsf{and}\ \mathsf{rain}\ \mathsf{and}\ (\mathsf{cyclist}\,y)\ \mathsf{and}\ (\mathsf{leave}\,y)$$

# Some connections

- Modulo type constraints, this semantics shares with many others the idea of treating the meanings of declaratives as functions from contexts to contexts

# Some connections

- Modulo type constraints, this semantics shares with many others the idea of treating the meanings of declaratives as functions from contexts to contexts
- It can be seen as a rational reconstruction of Heim 1982, similarly to Beaver 2001 and Muskens 1996

# Some connections

- Modulo type constraints, this semantics shares with many others the idea of treating the meanings of declaratives as functions from contexts to contexts
- It can be seen as a rational reconstruction of Heim 1982, similarly to Beaver 2001 and Muskens 1996
- It is also similar to de Groote's (2006) dynamic semantics; the type of contexts is essentially the type of de Groote's *continuations*

# Some connections

- Modulo type constraints, this semantics shares with many others the idea of treating the meanings of declaratives as functions from contexts to contexts
- It can be seen as a rational reconstruction of Heim 1982, similarly to Beaver 2001 and Muskens 1996
- It is also similar to de Groote's (2006) dynamic semantics; the type of contexts is essentially the type of de Groote's *continuations*
- Also, as Carl Pollard once noted, de Groote's declaratives get the type

  $$\gamma \to (\gamma \to \mathrm{t}) \to \mathrm{t}\,,$$

  where $\gamma$ is the type of sets of entities
- But with $\gamma$ analogous to $\mathrm{e}^n$ and t analogous to p, this is just a permutation of

  $$(\mathrm{e}^n \to \mathrm{p}) \to (\mathrm{e}^m \to \mathrm{p})\,,$$

  which is the type of contents

# Talk outline


Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# Hooking DAS up to HTLCG

- Connecting DAS to a formalism like Hybrid Type-Logical Categorial Grammar (Kubota and Levine, to appear) is straightforward, and mostly consists of modifying the lexicon

# Hooking DAS up to HTLCG

- Connecting DAS to a formalism like Hybrid Type-Logical Categorial Grammar (Kubota and Levine, to appear) is straightforward, and mostly consists of modifying the lexicon
- First, all *n*-ary static properties need to be replaced by the dynamic counterparts, obtained by the lifting functions dyn
- For example, letting $\textsc{give} =_{\text{def}} (\text{dyn}_{3,0}\, \mathsf{give})$, one lexical entry for *gave* becomes

$$\text{gave} ; \textsc{give} ; \text{VP}/\text{NP}/\text{NP}$$

# Hooking DAS up to HTLCG

- Connecting DAS to a formalism like Hybrid Type-Logical Categorial Grammar (Kubota and Levine, to appear) is straightforward, and mostly consists of modifying the lexicon
- First, all *n*-ary static properties need to be replaced by the dynamic counterparts, obtained by the lifting functions dyn
- For example, letting $\textsc{give} =_{\text{def}} (\text{dyn}_{3,0}\, \mathsf{give})$, one lexical entry for *gave* becomes

$$\text{gave}\,;\textsc{give}\,;\text{VP}/\text{NP}/\text{NP}$$

- Then the semantic component of the determiner lexical entries need to be replaced by their dynamic counterparts, e.g., the lexical entry for *every* becomes

$$\lambda_\tau\lambda_\sigma.\sigma\,(\text{every}\circ\tau)\,;\textsc{every}\,;(\text{S}|(\text{S}|\text{NP}))/\text{N}$$

# Hooking DAS up to HTLCG

- Connecting DAS to a formalism like Hybrid Type-Logical Categorial Grammar (Kubota and Levine, to appear) is straightforward, and mostly consists of modifying the lexicon
- First, all *n*-ary static properties need to be replaced by the dynamic counterparts, obtained by the lifting functions dyn
- For example, letting $\textsc{give} =_{\text{def}} (\text{dyn}_{3,0}\, \mathsf{give})$, one lexical entry for *gave* becomes

$$\text{gave}\,;\textsc{give}\,;\text{VP}/\text{NP}/\text{NP}$$

- Then the semantic component of the determiner lexical entries need to be replaced by their dynamic counterparts, e.g., the lexical entry for *every* becomes

$$\lambda_\tau\lambda_\sigma.\sigma\,(\text{every}\circ\tau)\,;\textsc{every}\,;(\text{S}|(\text{S}|\text{NP}))/\text{N}$$

- None of the inference rules need to change, although the semantic variable in NP hypotheses now has type n, of discourse referents

# Basic dynamic HTLCG analysis

- The analysis of *A cyclist broke every wheel* just requires adding some more lexical entries:

$$\lambda_{\tau}\lambda_{\sigma}.\sigma\,(\mathrm{a} \circ \tau)\,;\,\textsc{a}\,;\,(\mathrm{S}|(\mathrm{S}|\mathrm{NP}))/\mathrm{N}$$

$$\mathrm{cyclist}\,;\,\textsc{cyclist}\,;\,\mathrm{N}$$

$$\mathrm{wheel}\,;\,\textsc{wheel}\,;\,\mathrm{N}$$

$$\lambda_{\varphi_1}\lambda_{\varphi_2}.\varphi_2 \circ \mathrm{broke} \circ \varphi_1\,;\,\textsc{break}\,;\,(\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}$$

# Basic dynamic HTLCG analysis

- The analysis of *A cyclist broke every wheel* just requires adding some more lexical entries:

$$\lambda_{\tau}\lambda_{\sigma}.\sigma\,(\mathrm{a}\circ\tau)\,;\textsc{a}\,;(\mathrm{S}|(\mathrm{S}|\mathrm{NP}))/\mathrm{N}$$

$$\mathrm{cyclist}\,;\textsc{cyclist}\,;\mathrm{N}$$

$$\mathrm{wheel}\,;\textsc{wheel}\,;\mathrm{N}$$

$$\lambda_{\varphi_1}\lambda_{\varphi_2}.\varphi_2\circ\mathrm{broke}\circ\varphi_1\,;\textsc{break}\,;(\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}$$

- From these (along with the entry for *every*), we can derive both of the following:

$$\mathrm{a}\circ\mathrm{cyclist}\circ\mathrm{broke}\circ\mathrm{every}\circ\mathrm{wheel}\,;$$
$$(\textsc{a cyclist})_n.(\textsc{every wheel})_m.\textsc{break}\,m\,n\,;\mathrm{S}$$

$$\mathrm{a}\circ\mathrm{cyclist}\circ\mathrm{broke}\circ\mathrm{every}\circ\mathrm{wheel}\,;$$
$$(\textsc{every wheel})_m.(\textsc{a cyclist})_n.\textsc{break}\,m\,n\,;\mathrm{S}$$

# Discourse-level rules

- HTLCG, like many frameworks, has been aimed primarily at sentence-level phenomena

# Discourse-level rules

- HTLCG, like many frameworks, has been aimed primarily at sentence-level phenomena
- Extending it to model discourse requires the addition of a new type D (of discourses), and a new inference rule

$$\frac{\varphi_1 \,;\, u \,;\, \mathrm{D} \qquad \varphi_2 \,;\, k \,;\, \mathrm{S}}{\varphi_1 \circ \varphi_2 \,;\, u \circ (\mathsf{cc}\, k) \,;\, \mathrm{D}}\ \text{Continue}$$

- This just says that you can concatenate the result of proffering a content $k$ to an ongoing discourse to create a new discourse

# Discourse-level rules

- HTLCG, like many frameworks, has been aimed primarily at sentence-level phenomena
- Extending it to model discourse requires the addition of a new type D (of discourses), and a new inference rule

$$\frac{\varphi_1 \,;\, u \,;\, \mathrm{D} \qquad \varphi_2 \,;\, k \,;\, \mathrm{S}}{\varphi_1 \circ \varphi_2 \,;\, u \circ (\mathsf{cc}\, k) \,;\, \mathrm{D}}\ \text{Continue}$$

- This just says that you can concatenate the result of proffering a content $k$ to an ongoing discourse to create a new discourse
- Positing the *empty discourse* $\epsilon \,;\, \lambda_{c:\mathrm{c}}.c \,;\, \mathrm{D}$, the Continue rule gives the following derived rule:

$$\frac{\varphi \,;\, k \,;\, \mathrm{S}}{\varphi \,;\, (\mathsf{cc}\, k) \,;\, \mathrm{D}}\ \text{Start}$$

- This rule allows any dynamic sentence meaning $\varphi \,;\, k \,;\, \mathrm{S}$ to be promoted to a discourse, proffering its content along the way

# Talk outline

Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# That's an empirical question

- We now have a fully compositional dynamic framework in hand that incorporates the core insights of dynamic semantics and is easy to hook up to a categorial grammar

# That's an empirical question

- We now have a fully compositional dynamic framework in hand that incorporates the core insights of dynamic semantics and is easy to hook up to a categorial grammar
- Let's see how well it does in modeling phenomena that have a strong context-dependent component

  Projective meaning (Simons et al., 2010; Tonhauser et al., 2013)

  Anaphora: must find its antecedent in prior discourse, modulo accessibility constraints and salience

  Supplements: sometimes constitute a separate discourse update, in addition to participating in anaphora

# That's an empirical question

- We now have a fully compositional dynamic framework in hand that incorporates the core insights of dynamic semantics and is easy to hook up to a categorial grammar
- Let's see how well it does in modeling phenomena that have a strong context-dependent component

Projective meaning (Simons et al., 2010; Tonhauser et al., 2013)

Anaphora: must find its antecedent in prior discourse, modulo accessibility constraints and salience

Supplements: sometimes constitute a separate discourse update, in addition to participating in anaphora

VP ellipsis and (pseudo)gapping (Kubota and Levine, 2014)

needs to find a suitable antecedent property in order to get the meaning right

# Talk outline

Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# Just projecting?

- In what sense is projective meaning projective?

# Just projecting?

- In what sense is projective meaning projective?
- Projection occurs when an implication survives embedding under semantic operators that normally modify entailments

# Just projecting?

- In what sense is projective meaning projective?
- Projection occurs when an implication survives embedding under semantic operators that normally modify entailments
- Anaphora is projective because the requirement that the utterance context contain a suitable antecedent doesn't go away when embedded:

(2) There was a big pothole around one of the corners on the descent. One cyclist in the group didn't see the pothole.

# Just projecting?

- In what sense is projective meaning projective?
- Projection occurs when an implication survives embedding under semantic operators that normally modify entailments
- Anaphora is projective because the requirement that the utterance context contain a suitable antecedent doesn't go away when embedded:

(2) There was a big pothole around one of the corners on the descent. One cyclist in the group didn't see the pothole.

- For supplements, projection occurs when the supplemental content doesn't interact with the operators targeting the main clause content

(3) It's not true that Lance, a cheating doper, won the Tour de France in 2011.

# Invoking the prior context

- Though minimally simplified, the example below shows how the semantics needs to be extended to handle anaphora:

(4) A $\text{cyclist}_i$ arrived. The $\text{cyclist}_i$ left.

# Invoking the prior context

- Though minimally simplified, the example below shows how the semantics needs to be extended to handle anaphora:

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- In order for the anaphoric link indicated by the subscripts to be established, the dynamic meaning of *The cyclist* needs to

# Invoking the prior context

- Though minimally simplified, the example below shows how the semantics needs to be extended to handle anaphora:

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- In order for the anaphoric link indicated by the subscripts to be established, the dynamic meaning of *The cyclist* needs to
  1. know which discourse referents in the input context are entailed to be cyclists, and

# Invoking the prior context

- Though minimally simplified, the example below shows how the semantics needs to be extended to handle anaphora:

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- In order for the anaphoric link indicated by the subscripts to be established, the dynamic meaning of *The cyclist* needs to
  1. know which discourse referents in the input context are entailed to be cyclists, and
  2. select the most salient one from among them.

# Invoking the prior context

- Though minimally simplified, the example below shows how the semantics needs to be extended to handle anaphora:

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- In order for the anaphoric link indicated by the subscripts to be established, the dynamic meaning of *The cyclist* needs to
  1. know which discourse referents in the input context are entailed to be cyclists, and
  2. select the most salient one from among them.
- So we need a notion of dynamic entailment

# Context entailment

- Dynamic entailment is based on entailment between contexts, which is encoded by

$$\textsf{c-entails} =_{\text{def}} \lambda_{c:\mathrm{c}} \lambda_{d:\mathrm{c}_{\geq |c|}} \forall_{\mathbf{x}^{|c|}} . (c\,\mathbf{x})\ \textsf{entails}\ \textsf{exists}_{\mathbf{y}^{|d|-|c|}} . (d\,\mathbf{x}, \mathbf{y})$$

# Context entailment

- Dynamic entailment is based on entailment between contexts, which is encoded by

$$\textsf{c-entails} =_{\text{def}} \lambda_{c:\text{c}}\lambda_{d:\text{c}_{\geq|c|}}\forall_{\mathbf{x}^{|c|}}.(c\,\mathbf{x})\ \textsf{entails}\ \textsf{exists}_{\mathbf{y}^{|d|-|c|}}.(d\,\mathbf{x},\mathbf{y})$$

- In words, context entailment between $c$ and some context $d$ of at least $c$'s arity holds if every way of instantiating $c$'s discourse referents yields a proposition that entails the proposition obtained by instantiating $d$ with those same referents, plus any extras

# Context entailment

- Dynamic entailment is based on entailment between contexts, which is encoded by

$$\textsf{c-entails} =_{\text{def}} \lambda_{c:\mathrm{c}} \lambda_{d:\mathrm{c}_{\geq|c|}} \forall_{\mathbf{x}^{|c|}}.(c\,\mathbf{x})\ \textsf{entails}\ \textsf{exists}_{\mathbf{y}^{|d|-|c|}}.(d\,\mathbf{x},\mathbf{y})$$

- In words, context entailment between $c$ and some context $d$ of at least $c$'s arity holds if every way of instantiating $c$'s discourse referents yields a proposition that entails the proposition obtained by instantiating $d$ with those same referents, plus any extras
- For example, instantiate the contexts $c$ and $d$ as follows:

$$c = \lambda_x.\textsf{person}\,x$$
$$d = \lambda_{x,y}.(\textsf{name}\,y)\ \textsf{and}\ (\textsf{have}\,y\,x)$$

Then assuming people always have names, we have $\vdash c\ \textsf{c-entails}\ d$, because

$$\vdash \forall_x.(\textsf{person}\,x)\ \textsf{entails}\ \textsf{exists}_y.(\textsf{name}\,y)\ \textsf{and}\ (\textsf{have}\,y\,x)$$

# Content entailment

- But for anaphora, we need to know when a context entails some content, e.g., when a context entails that one of its discourse referents is a cyclist
- Entailment between a context and a content can be checked via

$$\mathsf{k\text{-}entails} =_{\mathrm{def}} \lambda_{c:\mathrm{c}}\lambda_{k:\mathrm{k}}.c\ \mathsf{c\text{-}entails}\ (\mathsf{cc}\, k\, c)$$

# Content entailment

- But for anaphora, we need to know when a context entails some content, e.g., when a context entails that one of its discourse referents is a cyclist
- Entailment between a context and a content can be checked via

$$\textsf{k-entails} =_{\text{def}} \lambda_{c:\text{c}} \lambda_{k:\text{k}}. c \; \textsf{c-entails} \; (\textsf{cc} \, k \, c)$$

- That is, a context $c$ entails a content $k$ if $c$ contextually entails the context we get by updating $c$ with $(\textsf{cc} \, k)$

# Content entailment

- But for anaphora, we need to know when a context entails some content, e.g., when a context entails that one of its discourse referents is a cyclist
- Entailment between a context and a content can be checked via

$$\mathsf{k\text{-}entails} =_{\mathrm{def}} \lambda_{c:\mathrm{c}}\lambda_{k:\mathrm{k}}.c\ \mathsf{c\text{-}entails}\ (\mathsf{cc}\, k\, c)$$

- That is, a context $c$ entails a content $k$ if $c$ contextually entails the context we get by updating $c$ with $(\mathsf{cc}\, k)$
- Example: letting $\textsc{person} =_{\mathrm{def}} (\mathrm{dyn}_{1,0}\ \mathsf{person})$, then

$$\vdash \lambda_x.(\mathsf{cyclist}\, x)\ \mathsf{k\text{-}entails}\ (\textsc{person}\, 0)$$

because $\vdash \lambda_x.(\mathsf{cyclist}\, x)\ \mathsf{c\text{-}entails}\ \lambda_x.(\mathsf{person}\, x)$

# Generalized definiteness

- With a notion of dynamic entailment, we can define an operator that selects the discourse referent with a certain property

# Generalized definiteness

- With a notion of dynamic entailment, we can define an operator that selects the discourse referent with a certain property
- The *generalized definiteness* operator $\mathsf{the} : \mathrm{d}_1 \to \mathrm{c} \to \mathrm{n}$ does this:

$$\mathsf{the} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{c:\mathrm{c}} \imath_{n:\mathrm{n}}.(n < |c|) \wedge c \;\mathsf{k\text{-}entails}\; (D\, n)$$

- So the returns the discourse referent $n$ known to $c$ such that $c$ content-entails $(D\, n)$

# Generalized definiteness

- With a notion of dynamic entailment, we can define an operator that selects the discourse referent with a certain property
- The *generalized definiteness* operator $\mathsf{the} : \mathrm{d}_1 \to \mathrm{c} \to \mathrm{n}$ does this:

$$\mathsf{the} =_{\mathrm{def}} \lambda_{D:\mathrm{d}_1} \lambda_{c:\mathrm{c}} \imath_{n:\mathrm{n}}.(n < |c|) \wedge c \ \mathsf{k\text{-}entails}\ (D\, n)$$

- So the returns the discourse referent $n$ known to $c$ such that $c$ content-entails $(D\, n)$
- Here, $\imath : (\mathrm{n} \to \mathrm{t}) \to \mathrm{n}$ is one of the definite description operators that come with the logic (cf. Henkin, 1963)

# Generalized definiteness

- With a notion of dynamic entailment, we can define an operator that selects the discourse referent with a certain property
- The *generalized definiteness* operator the : $\mathrm{d}_1 \to \mathrm{c} \to \mathrm{n}$ does this:

$$\mathsf{the} =_{\mathrm{def}} \lambda_{D:\mathrm{d}_1} \lambda_{c:\mathrm{c}} \iota_{n:\mathrm{n}}.(n < |c|) \wedge c \text{ k-entails } (D\, n)$$

- So the returns the discourse referent $n$ known to $c$ such that $c$ content-entails $(D\, n)$
- Here, $\iota : (\mathrm{n} \to \mathrm{t}) \to \mathrm{n}$ is one of the definite description operators that come with the logic (cf. Henkin, 1963)
- Caveat: a large component of $\iota$ is simply assumed, namely the requirement of greatest salience
- For example, it's not enough to select the unique cowboy in the following:

(5) A cowboy walked in and sat down. Another cowboy came in, and that cowboy ordered a Mai Tai.

# The definite determiner

- The definite determiner is then based on the:

$$\textsc{the} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} \lambda_{c:\mathrm{c}}.E\,(\mathsf{the}\,D\,c)\,c$$

- This just takes two properties $D$ and $E$, passing to $E$ the uniquely most salient discourse referent in $c$ with the property $D$

# The definite determiner

- The definite determiner is then based on the:

$$\text{THE} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} \lambda_{c:\mathrm{c}}. E\left(\mathsf{the}\, D\, c\right) c$$

- This just takes two properties $D$ and $E$, passing to $E$ the uniquely most salient discourse referent in $c$ with the property $D$
- For example, the model of *The cyclist left* is

$$\begin{aligned} &\text{THE CYCLIST LEAVE} \\ &= \lambda_{c:\mathrm{c}}.\text{LEAVE}\left(\mathsf{the}\,\text{CYCLIST}\, c\right) c \\ &= \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.\mathsf{leave}\, x_{(\mathsf{the}\,\text{CYCLIST}\, c)} \end{aligned}$$

# The definite determiner

- The definite determiner is then based on the:

$$\text{THE} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} \lambda_{c:\mathrm{c}}.E\left(\mathsf{the}\, D\, c\right) c$$

- This just takes two properties $D$ and $E$, passing to $E$ the uniquely most salient discourse referent in $c$ with the property $D$
- For example, the model of *The cyclist left* is

$$\begin{aligned} &\text{THE CYCLIST LEAVE} \\ &= \lambda_{c:\mathrm{c}}.\text{LEAVE}\left(\mathsf{the}\,\text{CYCLIST}\,c\right) c \\ &= \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.\mathsf{leave}\, x_{(\mathsf{the}\,\text{CYCLIST}\,c)} \end{aligned}$$

- This content takes a context $c$ to return another context in which whichever discourse referent $c$ has at the index $(\mathsf{the}\,\text{CYCLIST}\,c)$ is asserted to have left

# Resolving a definite

- Returning to our previous example

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- With ARRIVE $=_{\text{def}}$ $(\text{dyn}_{1,0}\,\mathsf{arrive})$, the model of (4) is

(6) $\left(\mathsf{cc}\left(\text{A CYCLIST ARRIVE}\right)\right) \circ \mathsf{cc}\left(\text{THE CYCLIST LEAVE}\right)$

# Resolving a definite

- Returning to our previous example

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- With ARRIVE $=_{\text{def}}$ $(\text{dyn}_{1,0}\, \mathsf{arrive})$, the model of (4) is

(6) $(\mathsf{cc}\,(\text{A CYCLIST ARRIVE})) \circ \mathsf{cc}\,(\text{THE CYCLIST LEAVE})$

- Passing the empty context $\lambda_{\mathbf{x}^1}.\mathsf{true}$ to (6) shows how the anaphora resolution works for (4)

# Resolving a definite

- Returning to our previous example

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- With ARRIVE $=_{\text{def}}$ $(\text{dyn}_{1,0}\,\mathsf{arrive})$, the model of (4) is

(6) $(\mathsf{cc}\,(\text{A CYCLIST ARRIVE})) \circ \mathsf{cc}\,(\text{THE CYCLIST LEAVE})$

- Passing the empty context $\lambda_{\mathbf{x}^1}.\mathsf{true}$ to (6) shows how the anaphora resolution works for (4)
- The context passed to THE CYCLIST LEAVE is

$$((\mathsf{cc}\,(\text{A CYCLIST ARRIVE}))\,\lambda_{\mathbf{x}^1}.\mathsf{true})$$
$$= \lambda_x.\mathsf{true}\ \mathsf{and}\ (\mathsf{cyclist}\,x)\ \mathsf{and}\ (\mathsf{arrive}\,x)$$

# Resolving a definite

- Returning to our previous example

(4) A cyclist$_i$ arrived. The cyclist$_i$ left.

- With ARRIVE $=_{\text{def}}$ $(\mathsf{dyn}_{1,0}\,\mathsf{arrive})$, the model of (4) is

(6) $(\mathsf{cc}\,(\text{A CYCLIST ARRIVE}))\circ\mathsf{cc}\,(\text{THE CYCLIST LEAVE})$

- Passing the empty context $\lambda_{\mathbf{x}^1}.\mathsf{true}$ to (6) shows how the anaphora resolution works for (4)
- The context passed to THE CYCLIST LEAVE is

$$((\mathsf{cc}\,(\text{A CYCLIST ARRIVE}))\,\lambda_{\mathbf{x}^1}.\mathsf{true})$$
$$=\lambda_x.\mathsf{true}\ \mathsf{and}\ (\mathsf{cyclist}\,x)\ \mathsf{and}\ (\mathsf{arrive}\,x)$$

- And so THE CYCLIST is able to select the intended referent, giving

$$\lambda_x.\mathsf{true}\ \mathsf{and}\ (\mathsf{cyclist}\,x)\ \mathsf{and}\ (\mathsf{arrive}\,x)\ \mathsf{and}\ (\mathsf{leave}\,x)$$

as the context output by (4) interpreted in the empty context

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others
- Instead, following Geurts (1999) and Beaver (2001), we harness the definiteness machinery to give meanings to proper names

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others
- Instead, following Geurts (1999) and Beaver (2001), we harness the definiteness machinery to give meanings to proper names
- In DAS, a proper name is just a definite with a special property, namely, the property of being named thus-and-so

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others
- Instead, following Geurts (1999) and Beaver (2001), we harness the definiteness machinery to give meanings to proper names
- In DAS, a proper name is just a definite with a special property, namely, the property of being named thus-and-so
- For example, the proper name *Kim* gets the meaning

$$\text{KIM} =_{\text{def}} \text{THE NAMED-KIM}$$

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others
- Instead, following Geurts (1999) and Beaver (2001), we harness the definiteness machinery to give meanings to proper names
- In DAS, a proper name is just a definite with a special property, namely, the property of being named thus-and-so
- For example, the proper name *Kim* gets the meaning

$$\text{KIM} =_{\text{def}} \text{THE NAMED-KIM}$$

- Here, NAMED-KIM is the dynamic property of being named "Kim", derived from its static counterpart named-kim via $\text{dyn}_{1,0}$

# Proper names

- In this framework, proper names are not modeled by constants, as they are in some others
- Instead, following Geurts (1999) and Beaver (2001), we harness the definiteness machinery to give meanings to proper names
- In DAS, a proper name is just a definite with a special property, namely, the property of being named thus-and-so
- For example, the proper name *Kim* gets the meaning

$$\text{KIM} =_{\text{def}} \text{THE NAMED-KIM}$$

- Here, NAMED-KIM is the dynamic property of being named "Kim", derived from its static counterpart named-kim via $\text{dyn}_{1,0}$
- In other words, *Kim* is treated on a par with the definite *the one named Kim*

# Definitely general

- The definite determiner THE is designed for general-purpose use, because arbitrarily complex dynamic properties can serve as its restrictor

# Definitely general

- The definite determiner THE is designed for general-purpose use, because arbitrarily complex dynamic properties can serve as its restrictor
- For example, the restrictor in the following is a relative clause:

(7) The cyclist that broke a wheel left

# Definitely general

- The definite determiner THE is designed for general-purpose use, because arbitrarily complex dynamic properties can serve as its restrictor
- For example, the restrictor in the following is a relative clause:

(7) The cyclist that broke a wheel left

- A model of (7), in this framework, would be

$$\textsc{the}\ (\textsc{cyclist that}\ \lambda_n.(\textsc{a wheel})_m.\textsc{break}\ m\ n)\ \textsc{leave}$$

# Definitely general

- The definite determiner THE is designed for general-purpose use, because arbitrarily complex dynamic properties can serve as its restrictor
- For example, the restrictor in the following is a relative clause:

(7) The cyclist that broke a wheel left

- A model of (7), in this framework, would be

$$\textsc{the}\ (\textsc{cyclist that}\ \lambda_n.(\textsc{a wheel})_m.\textsc{break}\, m\, n)\ \textsc{leave}$$

- Because it is sensitive to entailments, THE can be extended to handle *bridging anaphora* by implementing Roberts's (2005) "weak familiarity", but I omit the details here (see Martin 2012, 2013)

# What about pronouns?

- It's tempting to encode the meanings of pronouns using the definiteness function $\mathsf{the}$

# What about pronouns?

- It's tempting to encode the meanings of pronouns using the definiteness function `the`
- But empirically, pronouns are different:

(8) A cop$_i$ pulled me over, and she$_i$ wrote me a ticket!

(9) # Some guy$_i$ pulled me over, and she$_i$ wrote me a ticket!

# What about pronouns?

- It's tempting to encode the meanings of pronouns using the definiteness function the
- But empirically, pronouns are different:

(8) A cop$_i$ pulled me over, and she$_i$ wrote me a ticket!

(9) # Some guy$_i$ pulled me over, and she$_i$ wrote me a ticket!

- The generalization is that pronouns don't *require* their antecedents to strictly entail their descriptive content, just that the antecedent is consistent with their content

# What about pronouns?

- It's tempting to encode the meanings of pronouns using the definiteness function the
- But empirically, pronouns are different:

(8) A $\text{cop}_i$ pulled me over, and $\text{she}_i$ wrote me a ticket!

(9) # Some $\text{guy}_i$ pulled me over, and $\text{she}_i$ wrote me a ticket!

- The generalization is that pronouns don't *require* their antecedents to strictly entail their descriptive content, just that the antecedent is consistent with their content
- So we also need a notion of contextual consistency

# Content consistency and pronominal definiteness

- Fortunately, consistency between a context and a content is easy to define in terms of k-entails:

$$\textsf{k-cons} =_{\text{def}} \lambda_{c:\text{c}}\lambda_{k:\text{k}}.\neg\left(c\ \textsf{k-entails}\ (\text{NOT}\ k)\right)$$

- As the definition shows, k-cons only requires that the context does not entail the *negation* of the content

# Content consistency and pronominal definiteness

- Fortunately, consistency between a context and a content is easy to define in terms of k-entails:

$$\text{k-cons} =_{\text{def}} \lambda_{c:\mathrm{c}}\lambda_{k:\mathrm{k}}.\neg\,(c\ \text{k-entails}\ (\mathrm{NOT}\,k))$$

- As the definition shows, k-cons only requires that the context does not entail the *negation* of the content
- Then there is a modified version of the for pronouns that uses k-cons instead of k-entails:

$$\text{pro} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{c:\mathrm{c}}\imath_{n:\mathrm{n}}.(n < |c|) \wedge c\ \text{k-cons}\ (D\,n)$$

- Similarly to the, this function selects the uniquely most salient discourse referent in the context that is consistent with the dynamic property $D$

# Generalized pronouns

- Pronouns are defined by a 'determiner' that works in a parallel way to THE but using pro:

$$\text{PRO} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} \lambda_{c:\mathrm{c}}.E\,(\mathsf{pro}\,D\,c)\,c$$

# Generalized pronouns

- Pronouns are defined by a 'determiner' that works in a parallel way to THE but using pro:

$$\text{PRO} =_{\text{def}} \lambda_{D:\text{d}_1} \lambda_{E:\text{d}_1} \lambda_{c:\text{c}}.E\,(\mathsf{pro}\,D\,c)\,c$$

- Just like THE, the pronoun 'determiner' takes two dynamic properties and passes to the second the uniquely mostly salient discourse referent that is consistent with the first

# Generalized pronouns

- Pronouns are defined by a 'determiner' that works in a parallel way to THE but using pro:

$$\text{PRO} =_{\text{def}} \lambda_{D:\text{d}_1}\lambda_{E:\text{d}_1}\lambda_{c:\text{c}}.E\,(\mathsf{pro}\,D\,c)\,c$$

- Just like THE, the pronoun 'determiner' takes two dynamic properties and passes to the second the uniquely mostly salient discourse referent that is consistent with the first
- So the and pro can be seen as analogous to de Groote's (2006) `sel` function, except that `sel` doesn't take entailments into account

# Generalized pronouns

- Pronouns are defined by a 'determiner' that works in a parallel way to THE but using pro:

$$\text{PRO} =_{\text{def}} \lambda_{D:\text{d}_1} \lambda_{E:\text{d}_1} \lambda_{c:\text{c}}.E\,(\mathsf{pro}\,D\,c)\,c$$

- Just like THE, the pronoun 'determiner' takes two dynamic properties and passes to the second the uniquely mostly salient discourse referent that is consistent with the first
- So the and pro can be seen as analogous to de Groote's (2006) `sel` function, except that `sel` doesn't take entailments into account
- Note that, in contrast to *the*, this 'determiner' is never pronounced in English!

# Pronouns defined

- It does figure in the definitions of pronouns, however:

$$\text{HE} =_{\text{def}} \text{PRO MALE}$$
$$\text{HIM} =_{\text{def}} \text{PRO MALE}$$
$$\text{SHE} =_{\text{def}} \text{PRO FEMALE}$$
$$\text{HER} =_{\text{def}} \text{PRO FEMALE}$$
$$\text{IT} =_{\text{def}} \text{PRO NONHUMAN}$$

Here MALE, FEMALE, and NONHUMAN are unary dynamic properties derived from their counterparts male, female, and nonhuman by $\text{dyn}_{1,0}$

# Pronouns defined

- It does figure in the definitions of pronouns, however:

$$\text{HE} =_{\text{def}} \text{PRO MALE}$$
$$\text{HIM} =_{\text{def}} \text{PRO MALE}$$
$$\text{SHE} =_{\text{def}} \text{PRO FEMALE}$$
$$\text{HER} =_{\text{def}} \text{PRO FEMALE}$$
$$\text{IT} =_{\text{def}} \text{PRO NONHUMAN}$$

  Here MALE, FEMALE, and NONHUMAN are unary dynamic properties derived from their counterparts male, female, and nonhuman by $\text{dyn}_{1,0}$

- For example, the content SHE ARRIVE expands to

$$\begin{aligned}\text{SHE ARRIVE} &= \lambda_{c:\mathbf{c}}.\text{ARRIVE}\left(\mathsf{pro}\ \text{FEMALE}\ c\right) c \\ &= \lambda_{c:\mathbf{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{arrive}\ x_{(\mathsf{pro}\ \text{FEMALE}\ c)}\end{aligned}$$

# Possessives

- We can also define possessive pronouns based on the definite and pronoun determiners
- For example, the dynamic meaning of *his* can be modeled as

$$\text{HIS} =_{\text{def}} \lambda_{D:\text{d}_1} \lambda_{E:\text{d}_1}.\text{THE}\,(D\ \text{THAT}\ \lambda_n.\text{HE}\,(\text{HAVE}\ n))\ E$$

(Here HAVE is generated by $\text{dyn}_{2,0}$ from have)

# Possessives

- We can also define possessive pronouns based on the definite and pronoun determiners
- For example, the dynamic meaning of *his* can be modeled as

$$\text{HIS} =_{\text{def}} \lambda_{D:\text{d}_1}\lambda_{E:\text{d}_1}.\text{THE}\left(D\ \text{THAT}\ \lambda_n.\text{HE}\left(\text{HAVE}\ n\right)\right) E$$

(Here HAVE is generated by $\text{dyn}_{2,0}$ from have)

- For example, the meaning of *his wheel*, under this treatment, is

$$\text{HIS WHEEL} = \lambda_{E:\text{d}_1}.\text{THE}\left(\text{WHEEL THAT}\ \lambda_n.\text{HE}\left(\text{HAVE}\ n\right)\right) E$$

# Possessives

- We can also define possessive pronouns based on the definite and pronoun determiners
- For example, the dynamic meaning of *his* can be modeled as

$$\text{HIS} =_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}.\text{THE}\,(D\;\text{THAT}\;\lambda_n.\text{HE}\,(\text{HAVE}\;n))\;E$$

(Here HAVE is generated by $\text{dyn}_{2,0}$ from have)
- For example, the meaning of *his wheel*, under this treatment, is

$$\text{HIS WHEEL} = \lambda_{E:\mathrm{d}_1}.\text{THE}\,(\text{WHEEL THAT}\;\lambda_n.\text{HE}\,(\text{HAVE}\;n))\;E$$

- A similar treatment can be given to the possessives HER and ITS, by replacing HE with SHE or IT, respectively, and analogously for other possessives

# The obligatory donkey sentence

- Defining BIKE via $\text{dyn}_{1,0}$ and OWN and RIDE via $\text{dyn}_{2,0}$, we can get a meaning for

  (10) Every cyclist who owns a bike$_i$ rides it$_i$.

  that does the right thing

# The obligatory donkey sentence

- Defining BIKE via $\mathrm{dyn}_{1,0}$ and OWN and RIDE via $\mathrm{dyn}_{2,0}$, we can get a meaning for

  (10) Every cyclist who owns a bike$_i$ rides it$_i$.

  that does the right thing
- The semantics is

$$\text{EVERY}\ (\text{CYCLIST THAT}\ \lambda_n.(\text{A BIKE})_m.\text{OWN}\ m\ n)\ \lambda_n.(\text{IT}_m.\text{RIDE}\ m\ n)$$

# The obligatory donkey sentence

- Defining BIKE via $\text{dyn}_{1,0}$ and OWN and RIDE via $\text{dyn}_{2,0}$, we can get a meaning for

  (10) Every cyclist who owns a $\text{bike}_i$ rides $\text{it}_i$.

  that does the right thing
- The semantics is

$$\text{EVERY}\ (\text{CYCLIST THAT}\ \lambda_n.(\text{A BIKE})_m.\text{OWN}\ m\ n)\ \lambda_n.(\text{IT}_m.\text{RIDE}\ m\ n)$$

- The anaphora works because the restrictor property generates the following:

$$\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},y}.(\mathsf{cyclist}\ x_n)\ \mathsf{and}\ (\mathsf{bike}\ y)\ \mathsf{and}\ (\mathsf{own}\ y\ x_n)$$

# The obligatory donkey sentence

- Defining BIKE via $\text{dyn}_{1,0}$ and OWN and RIDE via $\text{dyn}_{2,0}$, we can get a meaning for

  (10) Every cyclist who owns a bike$_i$ rides it$_i$.

  that does the right thing
- The semantics is

$$\text{EVERY}\ (\text{CYCLIST THAT}\ \lambda_n.(\text{A BIKE})_m.\text{OWN}\ m\ n)\ \lambda_n.(\text{IT}_m.\text{RIDE}\ m\ n)$$

- The anaphora works because the restrictor property generates the following:

$$\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|},y}.(\mathsf{cyclist}\ x_n)\ \mathsf{and}\ (\mathsf{bike}\ y)\ \mathsf{and}\ (\mathsf{own}\ y\ x_n)$$

- The pronoun in the scope can pick up the uniquely most salient nonhuman antecedent from its input context, namely, the bike $y$

# Anaphoric accessibility in discourse

- However, the bike referent in (10) isn't accessible outside the scope of EVERY, since EVERY is defined in terms of FORALL, which is in turn defined in terms of NOT

# Anaphoric accessibility in discourse

- However, the bike referent in (10) isn't accessible outside the scope of EVERY, since EVERY is defined in terms of FORALL, which is in turn defined in terms of NOT
- To illustrate this, we take the even simpler example

(11) # No cyclist$_i$ arrives. She$_i$ leaves.

# Anaphoric accessibility in discourse

- However, the bike referent in (10) isn't accessible outside the scope of EVERY, since EVERY is defined in terms of FORALL, which is in turn defined in terms of NOT
- To illustrate this, we take the even simpler example

(11) # No cyclist$_i$ arrives. She$_i$ leaves.

- The dynamic meaning of the first utterance of (11) is

$$\text{NO CYCLIST ARRIVE} = \lambda_{c:\mathbf{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{not}\ \mathsf{exists}_y.(\mathsf{cyclist}\ y)\ \mathsf{and}\ (\mathsf{arrive}\ y)$$

# Anaphoric accessibility in discourse

- However, the bike referent in (10) isn't accessible outside the scope of EVERY, since EVERY is defined in terms of FORALL, which is in turn defined in terms of NOT
- To illustrate this, we take the even simpler example

(11) # No cyclist$_i$ arrives. She$_i$ leaves.

- The dynamic meaning of the first utterance of (11) is

$$\text{NO CYCLIST ARRIVE} = \lambda_{c:\mathbf{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{not\ exists}_{y}.(\mathsf{cyclist}\ y)\ \mathsf{and}\ (\mathsf{arrive}\ y)$$

- Since the cyclist referent $y$ is existentially bound, it is trapped—no reference to it in subsequent discourse is possible

# Anaphoric accessibility in discourse

- However, the bike referent in (10) isn't accessible outside the scope of EVERY, since EVERY is defined in terms of FORALL, which is in turn defined in terms of NOT
- To illustrate this, we take the even simpler example

(11) # No cyclist$_i$ arrives. She$_i$ leaves.

- The dynamic meaning of the first utterance of (11) is

$$\text{NO CYCLIST ARRIVE} = \lambda_{c:\mathbf{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{not}\ \mathsf{exists}_y.(\mathsf{cyclist}\ y)\ \mathsf{and}\ (\mathsf{arrive}\ y)$$

- Since the cyclist referent $y$ is existentially bound, it is trapped—no reference to it in subsequent discourse is possible
- And this inaccessibility is inherited by all the connectives, quantifiers, and determiners defined in terms of dynamic negation: EVERY, FORALL, IMPLIES, NO, OR

# Iterative adverbs

- Iterative adverbs like *too* can also be analyzed under the rubric of anaphora:

$$\text{TOO} =_{\text{def}} \lambda_{D:\text{d}_1} \lambda_{n:\text{n}} \lambda_{c:\text{c}_{>n}} \lambda_{\mathbf{x}^{|c|}}.$$
$$D\,(\imath_{m:\text{n}}.m = n \wedge \exists_{k:\text{n}}.(c \;\textsf{k-entails}\; (D\,k)) \wedge \neg\,(k = m))\,c\,\mathbf{x}$$

- This definition effectively requires the discourse referent passed to $D$ to be distinct from one the context already knows about with that property

# Iterative adverbs

- Iterative adverbs like *too* can also be analyzed under the rubric of anaphora:

$$\textsc{too} =_{\text{def}} \lambda_{D:\mathrm{d}_1} \lambda_{n:\mathrm{n}} \lambda_{c:\mathrm{c}_{>n}} \lambda_{\mathbf{x}^{|c|}}. \\ D\left(\imath_{m:\mathrm{n}}.m = n \wedge \exists_{k:\mathrm{n}}.(c \;\textsf{k-entails}\; (D\,k)) \wedge \neg\,(k = m)\right) c\,\mathbf{x}$$

- This definition effectively requires the discourse referent passed to $D$ to be distinct from one the context already knows about with that property
- For example, a model of *Kim owns a bike, too* would be

$$\textsc{kim}\left(\textsc{too}\,\lambda_n.(\textsc{a bike})_m.\textsc{own}\,m\,n\right)$$

# Iterative adverbs

- Iterative adverbs like *too* can also be analyzed under the rubric of anaphora:

$$\textsc{too} =_{\text{def}} \lambda_{D:\text{d}_1} \lambda_{n:\text{n}} \lambda_{c:\text{c}_{>n}} \lambda_{\mathbf{x}^{|c|}}.$$
$$D\,(\imath_{m:\text{n}}.m = n \wedge \exists_{k:\text{n}}.(c \;\textsf{k-entails}\;(D\,k)) \wedge \neg\,(k = m))\,c\,\mathbf{x}$$

- This definition effectively requires the discourse referent passed to $D$ to be distinct from one the context already knows about with that property
- For example, a model of *Kim owns a bike, too* would be

$$\textsc{kim}\;(\textsc{too}\;\lambda_n.(\textsc{a bike})_m.\textsc{own}\;m\,n)$$

- Supposing $k$ is selected as the discourse referent entailed to be named "Kim", the definition of TOO requires that there be some other referent besides $k$ that is also entailed to own a bike

# The conventional (implicature) view of supplements

- Potts (2005) and many others have characterized supplements like the one in (3) as contributing to a separate meaning "dimension"

(3) It's not true that Lance, a cheating doper, won the Tour de France in 2011.

# The conventional (implicature) view of supplements

- Potts (2005) and many others have characterized supplements like the one in (3) as contributing to a separate meaning "dimension"

(3) It's not true that Lance, a cheating doper, won the Tour de France in 2011.

- Multidimensionality has been touted as giving a nice model for (3), because it allows the implication that Lance doped to survive even when the implication of winning is negated

# The conventional (implicature) view of supplements

- Potts (2005) and many others have characterized supplements like the one in (3) as contributing to a separate meaning "dimension"

(3) It's not true that Lance, a cheating doper, won the Tour de France in 2011.

- Multidimensionality has been touted as giving a nice model for (3), because it allows the implication that Lance doped to survive even when the implication of winning is negated
- AnderBois et al. (2010, 2015) pointed out a potential problem for multidimensional accounts, namely that run-of-the-mill anaphoric links are unproblematic

(12) $\text{Kim}_i$'s $\text{bike}_j$, which used to have $\text{reflectors}_k$ on $\text{it}_j$, was safe to ride until $\text{she}_i$ took $\text{them}_k$ off.

# The conventional (implicature) view of supplements

- Potts (2005) and many others have characterized supplements like the one in (3) as contributing to a separate meaning "dimension"

(3) It's not true that Lance, a cheating doper, won the Tour de France in 2011.

- Multidimensionality has been touted as giving a nice model for (3), because it allows the implication that Lance doped to survive even when the implication of winning is negated
- AnderBois et al. (2010, 2015) pointed out a potential problem for multidimensional accounts, namely that run-of-the-mill anaphoric links are unproblematic

(12) $\text{Kim}_i$'s $\text{bike}_j$, which used to have $\text{reflectors}_k$ on $\text{it}_j$, was safe to ride until $\text{she}_i$ took $\text{them}_k$ off.

- In my dissertation (Martin, 2013), I tried to reconcile anaphora and multidimensionality, but more recently I became unsure that a multidimensional semantics is right for supplements

# Problems with multidimensionality for supplements

- The main reason is that, contrary to claims often made about them, supplements can participate in scope interactions

(13) In **each** class, **several** students$_i$ failed the midterm exam, which they$_i$ had to retake later. (Amaral et al., 2007)

(14) It's **not** the case that a boxer, a famous one, lives in this street. (Nouwen, 2014)

(15) **If** tomorrow I call the chair, who in turn calls the dean, then we will be in deep trouble. (Schlenker, ms)

(16) **Every** famous boxer I know$_i$ has a devoted brother, who he$_i$ completely relied on back when he$_i$ was just an amateur.

(17) But there would **always** be some student, a photographer or a glassblower, who would simply have taken a piece of newspaper and folded it once and propped it up like a tent and let it go at that.

# Further problems with multidimensionality

- Potts and others have often claimed that supplements are not deniable because they can't ever be *at-issue*, since their content ends up in the non-at-issue dimension

(18) a. Edna, who is a fearless leader, started the descent.

b. # No, she isn't. She is a coward.

(Koev, 2012)

# Further problems with multidimensionality

- Potts and others have often claimed that supplements are not deniable because they can't ever be *at-issue*, since their content ends up in the non-at-issue dimension

(18) a. Edna, who is a fearless leader, started the descent.

b. # No, she isn't. She is a coward.

(Koev, 2012)

- But this pattern isn't general, because supplements get easier to deny when they're closer to the end of an utterance

(19) a. He told her about Luke, who loved to have his picture taken.

b. No, he didn't like that at all.

c. No, he told her about Noah.

(AnderBois et al., 2010)

# More dimensions, more problems

- Baked into the multidimensional program is the idea that inhabiting the non-at-issue dimension is an inherent property of supplements

# More dimensions, more problems

- Baked into the multidimensional program is the idea that inhabiting the non-at-issue dimension is an inherent property of supplements
- But this means that all supplement anchors are treated on a par, so that current multidimensional approaches don't distinguish between proper name and indefinite anchors

# More dimensions, more problems

- Baked into the multidimensional program is the idea that inhabiting the non-at-issue dimension is an inherent property of supplements
- But this means that all supplement anchors are treated on a par, so that current multidimensional approaches don't distinguish between proper name and indefinite anchors
- And so they don't explain the apparent difference between the following:

(20) It's not true that some cyclist, a cheating doper, won the Tour de France in 1918. There was no Tour that year.

(21) It's not true that Henri Pélissier, a cheating doper, won the Tour de France in 1918. There was no Tour that year.

- It is much easier to interpret the supplement in the scope of negation for (20) than it is for (21)

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers
- That way, a supplement inherits the scope of its anchor, and scope interactions are re-enabled without further stipulation

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers
- That way, a supplement inherits the scope of its anchor, and scope interactions are re-enabled without further stipulation
- Then the preference for supplement content to project is attributed to scope preferences, rather than being an inherent property

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers
- That way, a supplement inherits the scope of its anchor, and scope interactions are re-enabled without further stipulation
- Then the preference for supplement content to project is attributed to scope preferences, rather than being an inherent property
- Since the framework already handles anaphora, we also get anaphoric interactions between supplements and other content with no extra effort

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers
- That way, a supplement inherits the scope of its anchor, and scope interactions are re-enabled without further stipulation
- Then the preference for supplement content to project is attributed to scope preferences, rather than being an inherent property
- Since the framework already handles anaphora, we also get anaphoric interactions between supplements and other content with no extra effort
- As I'll discuss in a minute, the account also gives a nice model of how a supplement's deniability increases with its utterance-finality

# A new, unidimensional account

- The core idea is to get rid of the extra dimensions and model supplements as quantifier phrase modifiers
- That way, a supplement inherits the scope of its anchor, and scope interactions are re-enabled without further stipulation
- Then the preference for supplement content to project is attributed to scope preferences, rather than being an inherent property
- Since the framework already handles anaphora, we also get anaphoric interactions between supplements and other content with no extra effort
- As I'll discuss in a minute, the account also gives a nice model of how a supplement's deniability increases with its utterance-finality
- This account is discussed in detail in Martin 2015 and Martin in press

# A prerequisite

- As a preliminary, we need to define the operation of *predicativization*

# A prerequisite

- As a preliminary, we need to define the operation of *predicativization*
- The predicativizer turns a dynamic generalized quantifier (GQ) into a dynamic property:

$$\text{PRED} =_{\text{def}} \lambda_{Q:\text{d}_1 \to \text{k}} \lambda_{n:\text{n}}.Q_m.m \text{ EQUALS } n$$

# A prerequisite

- As a preliminary, we need to define the operation of *predicativization*
- The predicativizer turns a dynamic generalized quantifier (GQ) into a dynamic property:

$$\text{PRED} =_{\text{def}} \lambda_{Q:\text{d}_1 \to \text{k}} \lambda_{n:\text{n}}.Q_m.m \text{ EQUALS } n$$

- For example, a component of the meaning of *Lance is a cyclist* is the predicativized GQ *a cyclist*, derived as follows

$$\begin{aligned}&\text{PRED A CYCLIST}\\ &= \lambda_{n:\text{n}}.(\text{A CYCLIST})_m.m \text{ EQUALS } n\\ &= \lambda_{n:\text{n}}.\text{EXISTS}_m.(\text{CYCLIST } m) \text{ AND } (m \text{ EQUALS } n)\end{aligned}$$

# A prerequisite

- As a preliminary, we need to define the operation of *predicativization*
- The predicativizer turns a dynamic generalized quantifier (GQ) into a dynamic property:

$$\text{PRED} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{n:\mathrm{n}}.Q_m.m \text{ EQUALS } n$$

- For example, a component of the meaning of *Lance is a cyclist* is the predicativized GQ *a cyclist*, derived as follows

$$\begin{aligned} &\text{PRED A CYCLIST} \\ &= \lambda_{n:\mathrm{n}}.(\text{A CYCLIST})_m.m \text{ EQUALS } n \\ &= \lambda_{n:\mathrm{n}}.\text{EXISTS}_m.(\text{CYCLIST } m) \text{ AND } (m \text{ EQUALS } n) \end{aligned}$$

- Here EQUALS $=_{\text{def}} \lambda_{m:\mathrm{n}} \lambda_{n:\mathrm{n}} \lambda_{c:\mathrm{c}} \lambda_{\mathbf{x}^{|c|}}.x_m$ equals $x_n$, and equals is the intensional equality function

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\textsc{comma} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} . (Q\,D) \text{ AND } (\text{THE}\, D\, E)$$

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\textsc{comma} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} . (Q\,D) \ \textsc{and} \ (\textsc{the}\, D\, E)$$

- The comma intonation is a quantifier phrase modifier, taking a GQ and a predicativized GQ (the appositive) to another GQ

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\textsc{comma} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1} . (Q\, D) \text{ AND } (\text{THE}\, D\, E)$$

- The comma intonation is a quantifier phrase modifier, taking a GQ and a predicativized GQ (the appositive) to another GQ
- It first applies the GQ's quantificational force to the appositive

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\text{COMMA} =_{\text{def}} \lambda_{Q:\text{d}_1\to\text{k}}\lambda_{D:\text{d}_1}\lambda_{E:\text{d}_1}.(Q\,D)\ \text{AND}\ (\text{THE}\ D\ E)$$

- The comma intonation is a quantifier phrase modifier, taking a GQ and a predicativized GQ (the appositive) to another GQ
- It first applies the GQ's quantificational force to the appositive
- Next, it anaphorically selects the uniquely most salient DR with the appositive's property and passes it to the scope

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\text{COMMA} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1}.(Q\,D) \text{ AND } (\text{THE}\,D\,E)$$

- The comma intonation is a quantifier phrase modifier, taking a GQ and a predicativized GQ (the appositive) to another GQ
- It first applies the GQ's quantificational force to the appositive
- Next, it anaphorically selects the uniquely most salient DR with the appositive's property and passes it to the scope
- The result is conjoined into a GQ in which the appositive does double duty, effectively adding its content into the GQ's restrictor

# The entire analysis of supplements on one slide

- All the work of the analysis is handled by the *comma intonation*, defined as

$$\text{COMMA} =_{\text{def}} \lambda_{Q:\mathrm{d}_1 \to \mathrm{k}} \lambda_{D:\mathrm{d}_1} \lambda_{E:\mathrm{d}_1}.(Q\,D)\ \text{AND}\ (\text{THE}\,D\,E)$$

- The comma intonation is a quantifier phrase modifier, taking a GQ and a predicativized GQ (the appositive) to another GQ
- It first applies the GQ's quantificational force to the appositive
- Next, it anaphorically selects the uniquely most salient DR with the appositive's property and passes it to the scope
- The result is conjoined into a GQ in which the appositive does double duty, effectively adding its content into the GQ's restrictor
- And that's all, folks

# A projecting supplement

- A simple example of a supplement that projects:

(22) Lance, a doper, won the Tour de France.

# A projecting supplement

- A simple example of a supplement that projects:

(22) Lance, a doper, won the Tour de France.

- We get the analysis

$$\text{COMMA LANCE}\ (\text{PRED A DOPER})\ \text{WIN-TDF}$$
$$= (\text{LANCE}\ (\text{PRED A DOPER}))\ \text{AND THE}\ (\text{PRED A DOPER})\ \text{WIN-TDF}$$

# A projecting supplement

- A simple example of a supplement that projects:

(22) Lance, a doper, won the Tour de France.

- We get the analysis

$$\textsc{comma lance}\,(\textsc{pred a doper})\,\textsc{win-tdf}$$
$$= (\textsc{lance}\,(\textsc{pred a doper}))\ \textsc{and the}\,(\textsc{pred a doper})\,\textsc{win-tdf}$$

- So (22) is treated on a par with *Lance is a doper, and the one who's a doper won the Tour de France*

# Projection as conjoined update

- Projection arises for (22) because the supplement constitutes a separate update when (22) gets proffered

# Projection as conjoined update

- Projection arises for (22) because the supplement constitutes a separate update when (22) gets proffered
- That's because of a formal theorem linking conjoined update and parataxis (see Martin in press for proof):

$$\vdash \forall_{h:\mathrm{k}}\forall_{k:\mathrm{k}}.\mathsf{cc}\,(h \text{ AND } k) = (\mathsf{cc}\,h) \circ (\mathsf{cc}\,k)$$

(Recall that the context change function cc transforms a content into an update (i.e., an at-issue proposal), and represents the process of proffering a content for acceptance or rejection)

# Projection as conjoined update

- Projection arises for (22) because the supplement constitutes a separate update when (22) gets proffered
- That's because of a formal theorem linking conjoined update and parataxis (see Martin in press for proof):

$$\vdash \forall_{h:\mathrm{k}}\forall_{k:\mathrm{k}}.\mathsf{cc}\,(h\ \mathrm{AND}\ k) = (\mathsf{cc}\,h) \circ (\mathsf{cc}\,k)$$

  (Recall that the context change function cc transforms a content into an update (i.e., an at-issue proposal), and represents the process of proffering a content for acceptance or rejection)
- And so, when proffered, the analysis of (22) is equivalent to

$$(\mathsf{cc}\ \mathrm{LANCE}\ (\mathrm{PRED\ A\ DOPER})) \circ (\mathsf{cc}\ \mathrm{THE}\ (\mathrm{PRED\ A\ DOPER})\ \mathrm{WIN\text{-}TDF})$$

- This amounts to a two-utterance discourse with (1) the update that Lance dopes followed by (2) the update that he won

# Projection as conjoined update

- Projection arises for (22) because the supplement constitutes a separate update when (22) gets proffered
- That's because of a formal theorem linking conjoined update and parataxis (see Martin in press for proof):

$$\vdash \forall_{h:\mathrm{k}}\forall_{k:\mathrm{k}}.\mathsf{cc}\,(h \text{ AND } k) = (\mathsf{cc}\,h) \circ (\mathsf{cc}\,k)$$

(Recall that the context change function cc transforms a content into an update (i.e., an at-issue proposal), and represents the process of proffering a content for acceptance or rejection)

- And so, when proffered, the analysis of (22) is equivalent to

$$(\mathsf{cc}\,\text{LANCE}\,(\text{PRED A DOPER})) \circ (\mathsf{cc}\,\text{THE}\,(\text{PRED A DOPER})\,\text{WIN-TDF})$$

- This amounts to a two-utterance discourse with (1) the update that Lance dopes followed by (2) the update that he won
- More generally, this implies that whenever a supplement outscopes all other operators, it projects because it constitutes its own discourse update

# Projection from under negation

- This account models projection for the negated simplification of (21) below:

(23) It's not true that Henri, a doper, won the Tour de France.

# Projection from under negation

- This account models projection for the negated simplification of (21) below:

(23) It's not true that Henri, a doper, won the Tour de France.

- The system generates the following two representations of (23)

$$\textsc{comma henri}\,(\textsc{pred a doper})\,\lambda_n.\textsc{not}\,(\textsc{win-tdf}\,n)$$

$$\textsc{not}\,(\textsc{comma henri}\,(\textsc{pred a doper})\,\textsc{win-tdf})$$

# Projection from under negation

- This account models projection for the negated simplification of (21) below:

(23) It's not true that Henri, a doper, won the Tour de France.

- The system generates the following two representations of (23)

$$\text{COMMA HENRI}\,(\text{PRED A DOPER})\,\lambda_n.\text{NOT}\,(\text{WIN-TDF}\,n)$$
$$\text{NOT}\,(\text{COMMA HENRI}\,(\text{PRED A DOPER})\,\text{WIN-TDF})$$

- The first of these is the projective one, because it is equivalent, under proffering, to the two-update discourse

$$(\mathsf{cc}\,\text{HENRI}\,(\text{PRED A DOPER}))\circ$$
$$(\mathsf{cc}\,\text{THE}\,(\text{PRED A DOPER})\,\lambda_n.\text{NOT}\,(\text{WIN-TDF}\,n))$$

# Projection from under negation

- This account models projection for the negated simplification of (21) below:

(23) It's not true that Henri, a doper, won the Tour de France.

- The system generates the following two representations of (23)

$$\text{COMMA HENRI}\ (\text{PRED A DOPER})\ \lambda_n.\text{NOT}\ (\text{WIN-TDF}\ n)$$

$$\text{NOT}\ (\text{COMMA HENRI}\ (\text{PRED A DOPER})\ \text{WIN-TDF})$$

- The first of these is the projective one, because it is equivalent, under proffering, to the two-update discourse

$$(\text{cc HENRI}\ (\text{PRED A DOPER}))\ \circ$$
$$(\text{cc THE}\ (\text{PRED A DOPER})\ \lambda_n.\text{NOT}\ (\text{WIN-TDF}\ n))$$

- This reading is preferred, as desired, because of the general preference for proper names to scope widest (Kamp and Reyle, 1993; Bos, 2003)

# Non-projection from under negation

- Things are different for indefinites, however:

(24) It's not true that some cyclist, a doper, won the Tour de France.

- For this simplified variant of (20), two scopings are generated, as before

$$\textsc{comma}\ (\textsc{a cyclist})\ (\textsc{pred a doper})\ \lambda_n.\textsc{not}\ (\textsc{win-tdf}\ n)$$

$$\textsc{not}\ (\textsc{comma}\ (\textsc{a cyclist})\ (\textsc{pred a doper})\ \textsc{win-tdf})$$

# Non-projection from under negation

- Things are different for indefinites, however:

  (24) It's not true that some cyclist, a doper, won the Tour de France.

- For this simplified variant of (20), two scopings are generated, as before

  $$\text{COMMA}\ (\text{A CYCLIST})\ (\text{PRED A DOPER})\ \lambda_n.\text{NOT}\ (\text{WIN-TDF}\ n)$$
  $$\text{NOT}\ (\text{COMMA}\ (\text{A CYCLIST})\ (\text{PRED A DOPER})\ \text{WIN-TDF})$$

- In this case, there is no default preference for the indefinite to scope wide, and so we get a genuine ambiguity between the projective and non-projective readings

# Quantifier scope ambiguity and projection ambiguity

- For Nouwen's (2014) example

(25) Every boxer has a coach, who is famous.

the system also gives two analyses:

$$(\textsc{every boxer})_n.(\textsc{comma}\ (\textsc{a coach})\ \lambda_m.(\textsc{have}\ m\ n)\ \textsc{famous})$$
$$(\textsc{comma}\ (\textsc{a coach})\ \lambda_m.(\textsc{every boxer})_n.(\textsc{have}\ m\ n)\ \textsc{famous})$$

# Quantifier scope ambiguity and projection ambiguity

- For Nouwen's (2014) example

  (25) Every boxer has a coach, who is famous.

  the system also gives two analyses:

  $$(\text{EVERY BOXER})_n.(\text{COMMA } (\text{A COACH})\ \lambda_m.(\text{HAVE } m\, n)\ \text{FAMOUS})$$
  $$(\text{COMMA } (\text{A COACH})\ \lambda_m.(\text{EVERY BOXER})_n.(\text{HAVE } m\, n)\ \text{FAMOUS})$$

- The first, non-projective, reading of (25) is preferred because of the independent preference for surface scope

# Quantifier scope ambiguity and projection ambiguity

- For Nouwen's (2014) example

  (25) Every boxer has a coach, who is famous.

  the system also gives two analyses:

  $$(\text{EVERY BOXER})_n.(\text{COMMA } (\text{A COACH})\ \lambda_m.(\text{HAVE } m\, n)\ \text{FAMOUS})$$
  $$(\text{COMMA } (\text{A COACH})\ \lambda_m.(\text{EVERY BOXER})_n.(\text{HAVE } m\, n)\ \text{FAMOUS})$$

- The first, non-projective, reading of (25) is preferred because of the independent preference for surface scope
- But just as with normal quantifier scope ambiguity, the second, projective, reading is also available by selecting the inverse scope reading instead

# Ruling out quantificational anchors

- A pervasive pattern is that quantificational anchors are disallowed, as in

(26) # Every cyclist, a doper, won the Tour de France.

# Ruling out quantificational anchors

- A pervasive pattern is that quantificational anchors are disallowed, as in

(26) # Every cyclist, a doper, won the Tour de France.

- In this account, quantificational anchors are ruled out by the familiar mechanism of anaphoric accessibility

# Ruling out quantificational anchors

- A pervasive pattern is that quantificational anchors are disallowed, as in

(26) # Every cyclist, a doper, won the Tour de France.

- In this account, quantificational anchors are ruled out by the familiar mechanism of anaphoric accessibility
- That's because the analysis of (26), when proffered, is

$$(\mathsf{cc}\ \textsc{every cyclist}\ (\textsc{pred a doper}))\ \circ$$
$$(\mathsf{cc}\ \textsc{the}\ (\textsc{pred a doper})\ \textsc{win-tdf})$$

- Since the doping cyclist referent is trapped in the scope of every, it cannot be accessed by THE (PRED A DOPER) in the next update, as desired

# Exceptional binding and supplements I

- Carl Pollard (p.c.) once pointed out this example to me:

(27) No Tibetan Buddhist$_i$ thinks the Dalai Lama, his$_i$ spiritual mentor, would ever cave to Chinese pressure tactics.

- To see how the system analyzes (27), we first have to define a meaning for *think*

$$\textsc{think} =_{\text{def}} \lambda_{k:\text{k}}\lambda_{n:\text{n}}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{think}\,(k\,c\,\mathbf{x})\,x_n\ ,$$

# Exceptional binding and supplements I

- Carl Pollard (p.c.) once pointed out this example to me:

(27) No Tibetan Buddhist$_i$ thinks the Dalai Lama, his$_i$ spiritual mentor, would ever cave to Chinese pressure tactics.

- To see how the system analyzes (27), we first have to define a meaning for *think*

$$\textsc{think} =_{\text{def}} \lambda_{k:\mathrm{k}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.\mathsf{think}\,(k\,c\,\mathbf{x})\,x_n\ ,$$

- Then the preferred reading generated for (27) is

$$\begin{aligned}&\textsc{comma}\,(\textsc{the d-l})\,(\textsc{pred his mentor})\\&\quad\lambda_m.(\textsc{no t-b})_n.\textsc{think}\,(\textsc{cave}\,m)\,n\\&= (\textsc{the d-l}\,(\textsc{pred his mentor}))\ \textsc{and}\\&\quad(\textsc{the}\,(\textsc{pred his mentor}))_m.(\textsc{no t-b})_n.\textsc{think}\,(\textsc{cave}\,m)\,n\end{aligned}$$

# Exceptional binding and supplements II

- This reading, repeated below, is *almost* the right one

$$(\text{THE D-L}\ (\text{PRED HIS MENTOR}))\ \text{AND}$$
$$(\text{THE}\ (\text{PRED HIS MENTOR}))_m.(\text{NO T-B})_n.\text{THINK}\ (\text{CAVE}\ m)\ n$$

# Exceptional binding and supplements II

- This reading, repeated below, is *almost* the right one

  $$(\textsc{the d-l}\ (\textsc{pred his mentor}))\ \textsc{and}\ (\textsc{the}\ (\textsc{pred his mentor}))_m.(\textsc{no t-b})_n.\textsc{think}\ (\textsc{cave}\ m)\ n$$

- In addition to cataphora, HIS can't access its antecedent, the Tibetan Buddhist, because it's in the scope of NO

# Exceptional binding and supplements II

- This reading, repeated below, is *almost* the right one

  (THE D-L (PRED HIS MENTOR)) AND
  (THE (PRED HIS MENTOR))$_m$.(NO T-B)$_n$.THINK (CAVE $m$) $n$

- In addition to cataphora, HIS can't access its antecedent, the Tibetan Buddhist, because it's in the scope of NO
- But note the similarity between (27) and this example, an instance of Roberts's (1989) *telescoping*:

(28) Each degree candidate$_i$ walked to the stage. He$_i$ took his$_i$ diploma from the dean and returned to his$_i$ seat. (Roberts, 1989)

# Exceptional binding and supplements II

- This reading, repeated below, is *almost* the right one

$$(\text{THE D-L }(\text{PRED HIS MENTOR}))\text{ AND}$$
$$(\text{THE }(\text{PRED HIS MENTOR}))_m.(\text{NO T-B})_n.\text{THINK }(\text{CAVE } m)\ n$$

- In addition to cataphora, HIS can't access its antecedent, the Tibetan Buddhist, because it's in the scope of NO
- But note the similarity between (27) and this example, an instance of Roberts's (1989) *telescoping*:

(28) Each degree candidate$_i$ walked to the stage. He$_i$ took his$_i$ diploma from the dean and returned to his$_i$ seat. (Roberts, 1989)

- An analysis of exceptional binding like (28) has been implemented by Wang et al. (2006) via discourse relations, and could be here too

# Salience and supplement deniability

- The supplement in (30) is easier to deny than the one in (29):

(29) Some cyclist, a doper, met Lance.

(30) Some cyclist met Lance, a doper.

# Salience and supplement deniability

- The supplement in (30) is easier to deny than the one in (29):

(29) Some cyclist, a doper, met Lance.

(30) Some cyclist met Lance, a doper.

- In this account, supplement deniability is related to the fact that more recent utterances are more salient (Ginzburg, 2012)

# Salience and supplement deniability

- The supplement in (30) is easier to deny than the one in (29):

(29) Some cyclist, a doper, met Lance.

(30) Some cyclist met Lance, a doper.

- In this account, supplement deniability is related to the fact that more recent utterances are more salient (Ginzburg, 2012)
- In the analysis of (30), the supplement updates the discourse last, and is therefore more salient:

$$\text{COMMA LANCE}\ \lambda_m.(\text{A CYCLIST})_n.(\text{MEET}\ m\ n)\ (\text{PRED A DOPER})$$
$$= (\text{LANCE}_m.(\text{A CYCLIST})_n.\text{MEET}\ m\ n)\ \text{AND}$$
$$\text{THE}\ \lambda_m.(\text{A CYCLIST})_n.(\text{MEET}\ m\ n)\ (\text{PRED A DOPER})$$

- Under proffering, this is equivalent to the two-utterance discourse *Some cyclist met Lance. The one that some cyclist met is a doper.*

# Anaphora and presupposition I

- This account follows Simons et al. (2010), Martin (2013) and Tonhauser et al. (2013) in not lumping factives, aspectuals, and achievements in with anaphora

# Anaphora and presupposition I

- This account follows Simons et al. (2010), Martin (2013) and Tonhauser et al. (2013) in not lumping factives, aspectuals, and achievements in with anaphora
- That's because they don't seem to constrain the context the way anaphora does:

(31) It can't be that Kim is worried because she regrets leaving the stove on. Her stove is currently broken.

(32) Sandy can't participate in that smoking cessation program because she didn't quit smoking—actually, she never smoked in her life.

(33) Lance didn't win the Tour de France in 2011. He didn't even enter that year.

# Anaphora and presupposition I

- This account follows Simons et al. (2010), Martin (2013) and Tonhauser et al. (2013) in not lumping factives, aspectuals, and achievements in with anaphora
- That's because they don't seem to constrain the context the way anaphora does:

(31) It can't be that Kim is worried because she regrets leaving the stove on. Her stove is currently broken.

(32) Sandy can't participate in that smoking cessation program because she didn't quit smoking—actually, she never smoked in her life.

(33) Lance didn't win the Tour de France in 2011. He didn't even enter that year.

- Contrast with the completely bizarre

(34) # She might be here, but there's no suitable antecedent to resolve *she* to.

# Anaphora and presupposition II

- This approach's stance:
  - Factives, aspectuals, achievements, etc., sometimes strongly suggest an inference on the part of the hearer
  - But it would be incorrect for the semantics to force the inference

# Anaphora and presupposition II

- This approach's stance:
  - Factives, aspectuals, achievements, etc., sometimes strongly suggest an inference on the part of the hearer
  - But it would be incorrect for the semantics to force the inference
- And so this approach can be seen as strengthening van der Sandt's (1992) slogan that *presupposition is [an instance of] anaphora* to the claim that *presupposition and anaphora are synonyms*

# Anaphora and presupposition II

- This approach's stance:
  - Factives, aspectuals, achievements, etc., sometimes strongly suggest an inference on the part of the hearer
  - But it would be incorrect for the semantics to force the inference
- And so this approach can be seen as strengthening van der Sandt's (1992) slogan that *presupposition is [an instance of] anaphora* to the claim that *presupposition and anaphora are synonyms*
- In other words, the job of the semantics should be to say which entailments the contextual interpretation gives rise to, but factives, aspectuals, achievements, etc., don't have the same force as true entailments

# Talk outline


Dynamic Agnostic Semantics

- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing

- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# An anaphoric analysis of VP ellipsis, etc.

- Our workshop organizers have a really cool analysis of VP ellipsis and numerous instances of (pseudo)gapping (Kubota and Levine, 2014, ms)

# An anaphoric analysis of VP ellipsis, etc.

- Our workshop organizers have a really cool analysis of VP ellipsis and numerous instances of (pseudo)gapping (Kubota and Levine, 2014, ms)
- Its central feature is that it gets correct analyses for a whole bunch of related phenomena via a single operator (VP abbreviates NP\S):

$$\lambda_{\varphi}.\varphi\,;\lambda_{\mathcal{F}}.(\mathcal{F}\,P)\,;(\mathrm{VP}/\$)|((\mathrm{VP}/\$)/(\mathrm{VP}/\$))$$

- The occurrence of $P$ is anaphoric to a previously mentioned property, with some constraints on its suitability that I'll discuss in a minute

# An anaphoric analysis of VP ellipsis, etc.

- Our workshop organizers have a really cool analysis of VP ellipsis and numerous instances of (pseudo)gapping (Kubota and Levine, 2014, ms)
- Its central feature is that it gets correct analyses for a whole bunch of related phenomena via a single operator (VP abbreviates NP\S):

$$\lambda_{\varphi}.\varphi\,;\lambda_{\mathcal{F}}.(\mathcal{F}\,P)\,;(\mathrm{VP}/\$)|((\mathrm{VP}/\$)/(\mathrm{VP}/\$))$$

- The occurrence of *P* is anaphoric to a previously mentioned property, with some constraints on its suitability that I'll discuss in a minute
- Kubota and Levine's account is static, but here we'll fill in the dynamic details, point out some problems, and make some suggestions for improvement

# Some data

- The analysis is targeted at data like the following

(35) a. Kim sneezed. Sandy did (too).

b. Kim thought she sneezed. Sandy did (too).

c. Kim read every book before Sandy did.
(VP ellipsis)

(36) Kim can eat pizza and Sandy tacos. (Gapping)

(37) a. Kim should eat the banana. Sandy should the apple

b. You can't take the lining out of that coat. You can this one.

c. Although I didn't give Kim the book, I did Sandy.
(Pseudogapping)

# Redefining contexts

- The basic idea is to store dynamic properties in the context as they're used, so they're available for later anaphoric reference

# Redefining contexts

- The basic idea is to store dynamic properties in the context as they're used, so they're available for later anaphoric reference
- So we need to redefine the type of contexts to be

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \rightarrow (\mathrm{p} \times (\Sigma_m.\mathrm{d}_m) \rightarrow \mathrm{t})$$

- This is the type of functions from an $n$-ary entity vector to a pair consisting of (1) a proposition and (2) a set of dynamic properties (of any arity)

# Redefining contexts

- The basic idea is to store dynamic properties in the context as they're used, so they're available for later anaphoric reference
- So we need to redefine the type of contexts to be

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \rightarrow (\mathrm{p} \times (\Sigma_m.\mathrm{d}_m) \rightarrow \mathrm{t})$$

- This is the type of functions from an *n*-ary entity vector to a pair consisting of (1) a proposition and (2) a set of dynamic properties (of any arity)
- The new second component of the context will store the dynamic properties as they are encountered

# Redefining contexts

- The basic idea is to store dynamic properties in the context as they're used, so they're available for later anaphoric reference
- So we need to redefine the type of contexts to be

$$\mathrm{c}_n =_{\mathrm{def}} \mathrm{e}^n \to (\mathrm{p} \times (\Sigma_m.\mathrm{d}_m) \to \mathrm{t})$$

- This is the type of functions from an *n*-ary entity vector to a pair consisting of (1) a proposition and (2) a set of dynamic properties (of any arity)
- The new second component of the context will store the dynamic properties as they are encountered
- Two new functions give mnemonic access to the two components:

$$\mathsf{cont} =_{\mathrm{def}} \lambda_{c:\mathrm{c}}.\pi_1\, c$$
$$\mathsf{rels} =_{\mathrm{def}} \lambda_{c:\mathrm{c}}.\pi_2\, c$$

# Sets in type theory

- We also need to define some functions for accessing and extending the dynamic property sets in the context

$$\emptyset =_{\text{def}} \lambda_{D:\text{d}_n}.\mathsf{F}$$
$$\{\cdot\} =_{\text{def}} \lambda_{D:\text{d}_n}\lambda_{Q:\Sigma_m.\text{d}_m}.Q = \langle n, D\rangle$$
$$\in =_{\text{def}} \lambda_{D:\text{d}_n}\lambda_{S:(\Sigma_m.\text{d}_m)\to\text{t}}.(S\ \langle n, D\rangle)$$
$$\cup =_{\text{def}} \lambda_{S:(\Sigma_n.\text{d}_n)\to\text{t}}\lambda_{T:(\Sigma_n.\text{d}_n)\to\text{t}}\lambda_{D:\text{d}_k}.D \in S \vee D \in T$$

- Also, $\{D, E\}$ is shorthand for $\{D\} \cup \{E\}$, and outer brackets are often dropped

# Redefining the connectives, quantifiers, and entailment

- The dynamic connectives and quantifiers also need redefining, so that they keep track of the properties they inherit

$$\mathsf{cc} =_{\text{def}} \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|}, \mathbf{y}^{|k|}}.\, \langle \mathsf{cont}\,(c\,\mathbf{x})\ \mathsf{and}\ \mathsf{cont}\,(k\,c\,\mathbf{x}, \mathbf{y}),$$
$$\mathsf{rels}\,(c\,\mathbf{x}) \cup \mathsf{rels}\,(k\,c\,\mathbf{x}, \mathbf{y})\rangle$$

$$\text{EXISTS} =_{\text{def}} \lambda_D \lambda_c.\, \left\langle \mathsf{cont}\,(D\ |c|\ c^+), \mathsf{rels}\,(D\ |c|\ c^+)\right\rangle$$

$$\text{AND} =_{\text{def}} \lambda_h \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|}, \mathbf{y}^{|h|}, \mathbf{z}^{|k|}}.\, \langle \mathsf{cont}\,(h\,c\,\mathbf{x}, \mathbf{y})\ \mathsf{and}\ \mathsf{cont}\,(k\,(\mathsf{cc}\,h\,c)\,\mathbf{x}, \mathbf{y}, \mathbf{z}),$$
$$\mathsf{rels}\,(h\,c\,\mathbf{x}, \mathbf{y}) \cup \mathsf{rels}\,(k\,(\mathsf{cc}\,h\,c)\,\mathbf{x}, \mathbf{y}, \mathbf{z})\rangle$$

$$\text{NOT} =_{\text{def}} \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|}}.\langle \mathsf{not\ exists}_{\mathbf{y}^{|k|}}.\mathsf{cont}\,(k\,c\,\mathbf{x}, \mathbf{y}),$$
$$\lambda_D.\exists_{\mathbf{z}^{|k|}}.D \in \mathsf{rels}\,(k\,c\,\mathbf{y}, \mathbf{z})\rangle$$

# Redefining the connectives, quantifiers, and entailment

- The dynamic connectives and quantifiers also need redefining, so that they keep track of the properties they inherit

$$\mathsf{cc} =_{\text{def}} \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|k|}}.\ \langle \mathsf{cont}\,(c\,\mathbf{x})\ \mathsf{and}\ \mathsf{cont}\,(k\,c\,\mathbf{x},\mathbf{y}),$$
$$\mathsf{rels}\,(c\,\mathbf{x}) \cup \mathsf{rels}\,(k\,c\,\mathbf{x},\mathbf{y})\rangle$$

$$\text{EXISTS} =_{\text{def}} \lambda_D \lambda_c.\ \big\langle \mathsf{cont}\,(D\ |c|\ c^+), \mathsf{rels}\,(D\ |c|\ c^+)\big\rangle$$

$$\text{AND} =_{\text{def}} \lambda_h \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|},\mathbf{y}^{|h|},\mathbf{z}^{|k|}}.\ \langle \mathsf{cont}\,(h\,c\,\mathbf{x},\mathbf{y})\ \mathsf{and}\ \mathsf{cont}\,(k\,(\mathsf{cc}\,h\,c)\,\mathbf{x},\mathbf{y},\mathbf{z}),$$
$$\mathsf{rels}\,(h\,c\,\mathbf{x},\mathbf{y}) \cup \mathsf{rels}\,(k\,(\mathsf{cc}\,h\,c)\,\mathbf{x},\mathbf{y},\mathbf{z})\rangle$$

$$\text{NOT} =_{\text{def}} \lambda_k \lambda_c \lambda_{\mathbf{x}^{|c|}}.\langle \mathsf{not\ exists}_{\mathbf{y}^{|k|}}.\mathsf{cont}\,(k\,c\,\mathbf{x},\mathbf{y}),$$
$$\lambda_D.\exists_{\mathbf{z}^{|k|}}.D \in \mathsf{rels}\,(k\,c\,\mathbf{y},\mathbf{z})\rangle$$

- We (trivially) redefine contextual entailment as follows:

$$\mathsf{c\text{-}entails} =_{\text{def}} \lambda_{c:\mathrm{c}} \lambda_{d:\mathrm{c}_{\geq |c|}} \forall_{\mathbf{x}^{|c|}}.\mathsf{cont}\,(c\,\mathbf{x})\ \mathsf{entails\ exists}_{\mathbf{y}^{|d|-|c|}}.(d\,\mathbf{x},\mathbf{y})$$

# Redefining dynamicization

- We also need to redefine the dynamicizer functions

$$\mathrm{dyn}_{0,i} =_{\mathrm{def}} \lambda_{p:\mathrm{p}_0} \lambda_{c:\mathrm{c}_{>i}} \lambda_{\mathbf{x}^{|c|}} . \langle p, \emptyset \rangle$$

$$\mathrm{dyn}_{n+1,i} =_{\mathrm{def}} \lambda_{R:\mathrm{p}_{n+1}} \lambda_{m:\mathrm{n}} \lambda_{c:\mathrm{c}_{>(\max\, i\, m)}} \lambda_{\mathbf{x}^{|c|}} .$$

$$\Big\langle \mathsf{cont}\, (\mathrm{dyn}_{n,(\max\, i\, m)}\, (R\, x_m)\, c\, \mathbf{x}),$$

$$\Big\{ \lambda_k . \mathrm{dyn}_{n,k}\, (R\, x_k) \Big\} \cup \mathsf{rels}\, (\mathrm{dyn}_{n,(\max\, i\, m)}\, (R\, x_m)\, c\, \mathbf{x}) \Big\rangle$$

# Redefining dynamicization

- We also need to redefine the dynamicizer functions

$$\mathrm{dyn}_{0,i} =_{\mathrm{def}} \lambda_{p:\mathrm{p}_0}\lambda_{c:\mathrm{c}_{>i}}\lambda_{\mathbf{x}^{|c|}}.\, \langle p, \varnothing \rangle$$

$$\mathrm{dyn}_{n+1,i} =_{\mathrm{def}} \lambda_{R:\mathrm{p}_{n+1}}\lambda_{m:\mathrm{n}}\lambda_{c:\mathrm{c}_{>(\max\, i\, m)}}\lambda_{\mathbf{x}^{|c|}}.$$
$$\Big\langle \mathsf{cont}\,(\mathrm{dyn}_{n,(\max\, i\, m)}\,(R\,x_m)\,c\,\mathbf{x}),$$
$$\Big\{\lambda_k.\mathrm{dyn}_{n,k}\,(R\,x_k)\Big\} \cup \mathsf{rels}\,(\mathrm{dyn}_{n,(\max\, i\, m)}\,(R\,x_m)\,c\,\mathbf{x})\Big\rangle$$

- For example, these give dynamic properties that store themselves and any sub-properties

$$(\mathrm{dyn}_{1,0}\,\mathsf{sneeze}) = \lambda_n\lambda_c\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{sneeze}\,x_n),$$
$$\lambda_k\lambda_c\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{sneeze}\,x_k),\varnothing\rangle\rangle$$

$$(\mathrm{dyn}_{2,0}\,\mathsf{eat}) = \lambda_m\lambda_n\lambda_c\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{eat}\,x_m\,x_n),$$
$$\{\lambda_k\lambda_j\lambda_c\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{eat}\,x_k\,x_j),\ldots\rangle, \lambda_j\lambda_c\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{eat}\,x_m\,x_j),\ldots\rangle\}\rangle$$

# Redefining the anaphoric determiners

- Lastly, we need to redefine the anaphoric determiners THE and PRO to store their scope property

$$\text{THE} =_{\text{def}} \lambda_{D:\text{d}_1}\lambda_{E:\text{d}_1}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \mathsf{cont}\,(E\,(\mathsf{the}\,D\,c)\,c\,\mathbf{x}),$$
$$\{E\} \cup \mathsf{rels}\,(E\,(\mathsf{the}\,D\,c)\,c\,\mathbf{x})\rangle$$
$$\text{PRO} =_{\text{def}} \lambda_{D:\text{d}_1}\lambda_{E:\text{d}_1}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \mathsf{cont}\,(E\,(\mathsf{pro}\,D\,c)\,c\,\mathbf{x}),$$
$$\{E\} \cup \mathsf{rels}\,(E\,(\mathsf{pro}\,D\,c)\,c\,\mathbf{x})\rangle$$

# Redefining the anaphoric determiners

- Lastly, we need to redefine the anaphoric determiners THE and PRO to store their scope property

$$\begin{aligned}\text{THE} &=_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \mathsf{cont}\,(E\,(\mathsf{the}\,D\,c)\,c\,\mathbf{x}),\\ &\quad \{E\}\cup \mathsf{rels}\,(E\,(\mathsf{the}\,D\,c)\,c\,\mathbf{x})\rangle\\ \text{PRO} &=_{\text{def}} \lambda_{D:\mathrm{d}_1}\lambda_{E:\mathrm{d}_1}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \mathsf{cont}\,(E\,(\mathsf{pro}\,D\,c)\,c\,\mathbf{x}),\\ &\quad \{E\}\cup \mathsf{rels}\,(E\,(\mathsf{pro}\,D\,c)\,c\,\mathbf{x})\rangle\end{aligned}$$

- For example, *The cyclist leaves* gets the meaning

$$\text{THE CYCLIST LEAVE} = \lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.\left\langle (\mathsf{leave}\,x_{(\mathsf{the}\,\text{CYCLIST}\,c)}), \{\text{LEAVE}\}\right\rangle$$

# New ellipsis/gapping operator

- For $n > 0$, we define the ellipsis operators vpe

$$\mathsf{vpe}_1 =_{\mathrm{def}} \lambda_{F:\mathrm{d}_1\to\mathrm{d}_1}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_1}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\;n\;c\;\mathbf{x}$$
$$\mathsf{vpe}_2 =_{\mathrm{def}} \lambda_{F:\mathrm{d}_2\to\mathrm{d}_2}\lambda_{m:\mathrm{n}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_2}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\;m\;n\;c\;\mathbf{x}$$
$$\mathsf{vpe}_3 =_{\mathrm{def}} \lambda_{F:\mathrm{d}_3\to\mathrm{d}_3}\lambda_{k:\mathrm{n}}\lambda_{m:\mathrm{n}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_3}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\;k\;m\;n\;c\;\mathbf{x}$$
$$\vdots$$

- These operators all select the uniquely most salient property in the context with the matching arity

# New ellipsis/gapping operator

- For $n > 0$, we define the ellipsis operators vpe

$$\mathsf{vpe}_1 =_{\text{def}} \lambda_{F:\mathrm{d}_1\to\mathrm{d}_1}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_1}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\,n\,c\,\mathbf{x}$$
$$\mathsf{vpe}_2 =_{\text{def}} \lambda_{F:\mathrm{d}_2\to\mathrm{d}_2}\lambda_{m:\mathrm{n}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_2}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\,m\,n\,c\,\mathbf{x}$$
$$\mathsf{vpe}_3 =_{\text{def}} \lambda_{F:\mathrm{d}_3\to\mathrm{d}_3}\lambda_{k:\mathrm{n}}\lambda_{m:\mathrm{n}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.F\,(\imath_{D:\mathrm{d}_3}.D \in \mathsf{rels}\,(c\,\mathbf{x}))\,k\,m\,n\,c\,\mathbf{x}$$
$$\vdots$$

- These operators all select the uniquely most salient property in the context with the matching arity
- We can now redefine Kubota and Levine's operator for VP ellipsis and gapping as follows:

$$\lambda_\varphi.\varphi\,;\mathsf{vpe}_{|\$|+1}\,;(\mathrm{VP}/\$)|((\mathrm{VP}/\$)/(\mathrm{VP}/\$))$$

Here $|\$|$ is the number of argument categories in \$ (NP, PP, ...)

# VP ellipsis 1

- With the lexical entry for *did*

$$\text{did} \,;\, \lambda_{D:\mathrm{d}_1}.D \,;\, \mathrm{VP}/\mathrm{VP}$$

we can now analyze the following VP ellipsis example:

(38) Kim read every book and then Sandy did.

- The semantics gives two readings for (38)

$$(\textsc{every book})_m.(\textsc{kim}_n.(\textsc{read}\, m\, n)\ \textsc{and sandy}\ (\mathsf{vpe}_1\ \textsc{did}))$$
$$(\textsc{kim}_n.(\textsc{every book})_m.(\textsc{read}\, m\, n))\ \textsc{and sandy}\ (\mathsf{vpe}_1\ \textsc{did})$$

# VP ellipsis 1

- With the lexical entry for *did*

$$\text{did} \,;\, \lambda_{D:\mathrm{d}_1}.D \,;\, \mathrm{VP}/\mathrm{VP}$$

we can now analyze the following VP ellipsis example:

(38) Kim read every book and then Sandy did.

- The semantics gives two readings for (38)

$$(\text{EVERY BOOK})_m.(\text{KIM}_n.(\text{READ}\, m\, n)\ \text{AND SANDY}\ (\mathsf{vpe}_1\ \text{DID}))$$
$$(\text{KIM}_n.(\text{EVERY BOOK})_m.(\text{READ}\, m\, n))\ \text{AND SANDY}\ (\mathsf{vpe}_1\ \text{DID})$$

- For the first, $\mathsf{vpe}_1$ selects the property $\lambda_k.\text{READ}\, m\, k$, but for the second, it selects the property $\lambda_k.(\text{EVERY BOOK})_m.\text{READ}\, m\, k$

# VP ellipsis 2

- To analyze

(35b) Kim thought she sneezed. Sandy did (too).

- we redefine the meaning of *thinks* as

$$\text{THINK} =_{\text{def}} \lambda_{k:\text{k}} \lambda_{n:\text{n}} \lambda_{c:\text{c}} \lambda_{\mathbf{x}^{|c|}} . \langle (\mathsf{think}\,(\mathsf{cont}\,(k\,c\,\mathbf{x}))\,x_n), \mathsf{rels}\,(k\,c\,\mathbf{x}) \rangle$$

- Then the meaning of (35b) is the discourse

$$(\mathsf{cc}\,\text{KIM}\,(\text{THINK}\,(\text{SHE SNEEZE})) \circ (\mathsf{cc}\,\text{SANDY}\,(\mathsf{vpe}_1\,\text{DID}))$$

# VP ellipsis 2

- To analyze

  (35b) Kim thought she sneezed. Sandy did (too).

- we redefine the meaning of *thinks* as

$$\text{THINK} =_{\text{def}} \lambda_{k:\text{k}}\lambda_{n:\text{n}}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{think}\,(\mathsf{cont}\,(k\,c\,\mathbf{x}))\,x_n),\mathsf{rels}\,(k\,c\,\mathbf{x})\rangle$$

- Then the meaning of (35b) is the discourse

$$(\mathsf{cc}\,\text{KIM}\,(\text{THINK}\,(\text{SHE SNEEZE}))\circ(\mathsf{cc}\,\text{SANDY}\,(\mathsf{vpe}_1\,\text{DID}))$$

- Since the context passed to the second utterance contains the properties

$$\{\text{THINK}\,(\text{SHE SNEEZE}),\text{SNEEZE}\}$$

  the $\mathsf{vpe}_1$ operator needs to select the most salient one

# VP ellipsis 2

- To analyze

  (35b) Kim thought she sneezed. Sandy did (too).

- we redefine the meaning of *thinks* as

$$\text{THINK} =_{\text{def}} \lambda_{k:\mathrm{k}}\lambda_{n:\mathrm{n}}\lambda_{c:\mathrm{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle(\mathsf{think}\,(\mathsf{cont}\,(k\,c\,\mathbf{x}))\,x_n), \mathsf{rels}\,(k\,c\,\mathbf{x})\rangle$$

- Then the meaning of (35b) is the discourse

$$(\mathsf{cc}\,\text{KIM}\,(\text{THINK}\,(\text{SHE SNEEZE})) \circ (\mathsf{cc}\,\text{SANDY}\,(\mathsf{vpe}_1\,\text{DID}))$$

- Since the context passed to the second utterance contains the properties

$$\{\text{THINK}\,(\text{SHE SNEEZE}), \text{SNEEZE}\}$$

  the $\mathsf{vpe}_1$ operator needs to select the most salient one
- (Is there really an ambiguity? Or not?)

# VP ellipsis and salience I

- Assuming the $\mathsf{vpe}_1$ operator selects THINK (SHE SNEEZE) as the more salient property, we're still left with an ambiguity

(39) Kim$_i$ thought she$_{i/j}$ sneezed. Sandy$_k$ thought she$_{i/j/k}$ sneezed too.

# VP ellipsis and salience I

- Assuming the $\mathsf{vpe}_1$ operator selects THINK (SHE SNEEZE) as the more salient property, we're still left with an ambiguity

(39) $\text{Kim}_i$ thought $\text{she}_{i/j}$ sneezed. $\text{Sandy}_k$ thought $\text{she}_{i/j/k}$ sneezed too.

- We can use various devices to force one of the readings over the other, such as binding the first occurrence of the pronoun at the VP level

# VP ellipsis and salience I

- Assuming the $\mathsf{vpe}_1$ operator selects THINK (SHE SNEEZE) as the more salient property, we're still left with an ambiguity

  (39) Kim$_i$ thought she$_{i/j}$ sneezed. Sandy$_k$ thought she$_{i/j/k}$ sneezed too.

- We can use various devices to force one of the readings over the other, such as binding the first occurrence of the pronoun at the VP level
- As an alternative, I simply leave it up to the (unimplemented) salience mechanism to decide which antecedent is right for which occurrence

# VP ellipsis and salience II

As justification, consider (39) in the following contexts:

## Context

Kim and Sandy are wondering whether Megyn Kelly sneezed on air after Donald Trump assailed her with misogynistic comments.
(Kim / Megyn Kelly; Sandy / Megyn Kelly)

# VP ellipsis and salience II

As justification, consider (39) in the following contexts:

## Context

Kim and Sandy are wondering whether Megyn Kelly sneezed on air after Donald Trump assailed her with misogynistic comments.
(Kim / Megyn Kelly; Sandy / Megyn Kelly)

## Context

Kim and Sandy are discussing whether or not Kim sneezed during her testimony about Chelsea Clinton's potential ties to Hezbollah in the 37th House select committee on Benghazi.
(Kim / Kim; Sandy / Kim)

# VP ellipsis and salience II

As justification, consider (39) in the following contexts:

## Context

Kim and Sandy are wondering whether Megyn Kelly sneezed on air after Donald Trump assailed her with misogynistic comments.
(Kim / Megyn Kelly; Sandy / Megyn Kelly)

## Context

Kim and Sandy are discussing whether or not Kim sneezed during her testimony about Chelsea Clinton's potential ties to Hezbollah in the 37th House select committee on Benghazi.
(Kim / Kim; Sandy / Kim)

## Context

Kim and Sandy are arguing over which one of them had the worse time during last year's exceptionally tortuous allergy season.
(Kim / Kim; Sandy / Sandy)

# Pseudogapping

- We can analyze the pseudogapping example

  (37a) Kim ate the banana. Sandy should the apple.

- Giving a definition for the transitive verb version of *should* as

$$\text{SHOULD} =_{\text{def}} \lambda_{D:\text{d}_2}\lambda_{m:\text{n}}\lambda_{n:\text{n}}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \textsf{should cont}\,(D\,m\,n\,c\,\mathbf{x}),\ \{D\} \cup \textsf{rels}\,(D\,m\,n\,c\,\mathbf{x})\rangle$$

allows an analysis of (37a):

$$(\textsf{cc}\,\text{KIM}_n.(\text{THE BANANA})_m.\text{EAT}\,m\,n) \circ (\textsf{cc}\,\text{SANDY}_k.(\text{THE APPLE})_j.(\textsf{vpe}_2\,\text{SHOULD})\,j\,k)$$

# Pseudogapping

- We can analyze the pseudogapping example

(37a) Kim ate the banana. Sandy should the apple.

- Giving a definition for the transitive verb version of *should* as

$$\text{SHOULD} =_{\text{def}} \lambda_{D:\text{d}_2}\lambda_{m:\text{n}}\lambda_{n:\text{n}}\lambda_{c:\text{c}}\lambda_{\mathbf{x}^{|c|}}.\,\langle \mathsf{should\,cont}\,(D\,m\,n\,c\,\mathbf{x}), \{D\} \cup \mathsf{rels}\,(D\,m\,n\,c\,\mathbf{x})\rangle$$

allows an analysis of (37a):

$$(\mathsf{cc}\,\text{KIM}_n.(\text{THE BANANA})_m.\text{EAT}\,m\,n) \circ (\mathsf{cc}\,\text{SANDY}_k.(\text{THE APPLE})_j.(\mathsf{vpe}_2\,\text{SHOULD})\,j\,k)$$

- Since the input context to *Sandy should the apple* contains

$$\{(\text{THE BANANA})_m.(\text{EAT}\,m), \text{EAT}\}\ ,$$

$\mathsf{vpe}_2$ selects the only available binary dynamic property EAT, as desired

# The syntactic identity (meta)constraint

- So the arity requirement built into the vpe operators partially constrains which antecedent property can be chosen

# The syntactic identity (meta)constraint

- So the arity requirement built into the vpe operators partially constrains which antecedent property can be chosen
- But as Kubota and Levine point out, this can't be the whole story, since sometimes a syntactic match is required too

(43) * John spoke to Mary more often than Peter did for Anne.

- To rule out (43), Kubota and Levine constrain the anaphora resolution for their VP ellipsis / gapping operator so that anaphora isn't possible

# The syntactic identity (meta)constraint

- So the arity requirement built into the vpe operators partially constrains which antecedent property can be chosen
- But as Kubota and Levine point out, this can't be the whole story, since sometimes a syntactic match is required too

(43) * John spoke to Mary more often than Peter did for Anne.

- To rule out (43), Kubota and Levine constrain the anaphora resolution for their VP ellipsis / gapping operator so that anaphora isn't possible
- The reason is that the category $\mathrm{VP}/\mathrm{PP}_{to}$ of *spoke to* doesn't match the category $\mathrm{VP}/\mathrm{PP}_{for}$ of *spoke for*

# The syntactic identity (meta)constraint

- So the arity requirement built into the vpe operators partially constrains which antecedent property can be chosen
- But as Kubota and Levine point out, this can't be the whole story, since sometimes a syntactic match is required too

(43) * John spoke to Mary more often than Peter did for Anne.

- To rule out (43), Kubota and Levine constrain the anaphora resolution for their VP ellipsis / gapping operator so that anaphora isn't possible
- The reason is that the category $\mathrm{VP}/\mathrm{PP}_{to}$ of *spoke to* doesn't match the category $\mathrm{VP}/\mathrm{PP}_{for}$ of *spoke for*
- However, this constraint probably can't be encoded in the logic, since judgments like $\varphi; s; C$ are metalanguage statements
- So we may have to content ourselves with the syntactic match being a metaconstraint

# Talk outline

Dynamic Agnostic Semantics
- Agnostic Semantics
- Going dynamic
- Connecting it to a grammar

Road testing
- Projective meaning
  - Anaphora
  - Supplements
- VP ellipsis and related phenomena

Conclusions and future directions

# Summing up

- DAS is a modern, type-theoretic, compositional semantic framework that draws on the core insights of both dynamic semantics and the Montagovian tradition

# Summing up

- DAS is a modern, type-theoretic, compositional semantic framework that draws on the core insights of both dynamic semantics and the Montagovian tradition
- It can be undergirded by a wide range of static semantics: intensional Montagovian, extensional Montagovian, hyperintensional, etc.

# Summing up

- DAS is a modern, type-theoretic, compositional semantic framework that draws on the core insights of both dynamic semantics and the Montagovian tradition
- It can be undergirded by a wide range of static semantics: intensional Montagovian, extensional Montagovian, hyperintensional, etc.
- It is straightforward to hook up to your favorite grammar formalism, and it has a ton of empirical payoff: anaphora, supplements, VP ellipsis, (pseudo)gapping in addition to quantifier scope, discontinuous constituency, etc.
- (Note that Simon's talk may give an alternative perspective on supplements)

# Summing up

- DAS is a modern, type-theoretic, compositional semantic framework that draws on the core insights of both dynamic semantics and the Montagovian tradition
- It can be undergirded by a wide range of static semantics: intensional Montagovian, extensional Montagovian, hyperintensional, etc.
- It is straightforward to hook up to your favorite grammar formalism, and it has a ton of empirical payoff: anaphora, supplements, VP ellipsis, (pseudo)gapping in addition to quantifier scope, discontinuous constituency, etc.
- (Note that Simon's talk may give an alternative perspective on supplements)
- Via dependent types, it accomplishes what other frameworks do in the metalanguage, namely making sure the context has enough discourse referents for the purported interpretation

# Looking ahead

Some loose ends remain:

- The account of VP ellipsis / gapping requires a lot of bookkeeping, so it'd be nice to trim that down and also ensure that it gets other instances (e.g., discontinuous pseudogapping)

# Looking ahead

Some loose ends remain:

- The account of VP ellipsis / gapping requires a lot of bookkeeping, so it'd be nice to trim that down and also ensure that it gets other instances (e.g., discontinuous pseudogapping)
- Ideally, the context should model a *question under discussion* (QUD)—Murat will hopefully fill in some of the details for that

# Looking ahead

Some loose ends remain:

- The account of VP ellipsis / gapping requires a lot of bookkeeping, so it'd be nice to trim that down and also ensure that it gets other instances (e.g., discontinuous pseudogapping)
- Ideally, the context should model a *question under discussion* (QUD)—Murat will hopefully fill in some of the details for that
- Unlike supplements, expressives like *damn* may actually call for a multidimensional semantics, but no account of them in DAS has yet been developed (this would also probably require a model of multiple speakers/points of view)

# Looking ahead

Some loose ends remain:

- The account of VP ellipsis / gapping requires a lot of bookkeeping, so it'd be nice to trim that down and also ensure that it gets other instances (e.g., discontinuous pseudogapping)
- Ideally, the context should model a *question under discussion* (QUD)—Murat will hopefully fill in some of the details for that
- Unlike supplements, expressives like *damn* may actually call for a multidimensional semantics, but no account of them in DAS has yet been developed (this would also probably require a model of multiple speakers/points of view)
- It would be interesting to see how the *de dicto*/*de re* distinction plays out in a dynamic setting; maybe Colin will shed some light

# Looking ahead

Some loose ends remain:

- The account of VP ellipsis / gapping requires a lot of bookkeeping, so it'd be nice to trim that down and also ensure that it gets other instances (e.g., discontinuous pseudogapping)
- Ideally, the context should model a *question under discussion* (QUD)—Murat will hopefully fill in some of the details for that
- Unlike supplements, expressives like *damn* may actually call for a multidimensional semantics, but no account of them in DAS has yet been developed (this would also probably require a model of multiple speakers/points of view)
- It would be interesting to see how the *de dicto*/*de re* distinction plays out in a dynamic setting; maybe Colin will shed some light
- A comparison with the approaches using monads, which seem increasingly popular, is in order—I'm hoping Carl and Simon will provide some clues

# References I

P. Amaral, C. Roberts, and E. A. Smith. Review of *The Logic of Conventional Implicatures* by Chris Potts. *Linguistics and Philosophy*, 30 (6):707–749, 2007. doi: 10.1007/s10988-008-9025-2.

S. AnderBois, A. Brasoveanu, and R. Henderson. Crossing the appositive/at-issue meaning boundary. In *Semantics and Linguistic Theory (SALT) 20*, pages 328–346, 2010. URL `http://journals.linguisticsociety.org/proceedings/index.php/SALT/article/view/2551`.

S. AnderBois, A. Brasoveanu, and R. Henderson. At-issue proposals and appositive impositions in discourse. *Journal of Semantics*, 32(1): 93–138, 2015. doi: 10.1093/jos/fft014.

D. I. Beaver. *Presupposition and Assertion in Dynamic Semantics*. CSLI Publications, Stanford, CA, 2001.

# References II

J. Bos. Implementing the binding and accommodation theory for anaphora resolution and presupposition projection. *Computational Linguistics*, 29(2):179–210, 2003. doi: 10.1162/089120103322145306.

G. Chierchia. *The Dynamics of Meaning: Anaphora, Presupposition, and the Theory of Grammar*. University of Chicago Press, Chicago, IL & London, 1995. doi: 10.7208/chicago/9780226104515.001.0001.

P. de Groote. Towards a Montagovian account of dynamics. In *Semantics and Linguistic Theory (SALT) 16*, pages 1–16, 2006. URL `http://journals.linguisticsociety.org/proceedings/index.php/SALT/article/view/2952`.

P. de Groote and M.-J. Nederhof, editors. *Formal Grammar 15 and 16*, number 7395 in Lecture Notes in Computer Science, 2012. doi: 10.1007/978-3-642-32024-8.

B. Geurts. *Presuppositions and Pronouns*, volume 3 of *Current Research in the Semantics/Pragmatics Interface*. Elsevier, Oxford, 1999.

# References III

J. Ginzburg. *The Interactive Stance*. Oxford University Press, New York, NY & Oxford, 2012. doi: 10.1093/acprof:oso/9780199697922.001.0001.

J. Groenendijk and M. Stokhof. Dynamic Montague grammar. In L. Kálmán and L. Pólos, editors, *Papers from the Second Symposium on Logic and Language*. Akadémiai Kiadó, 1990.

J. Groenendijk and M. Stokhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14(1):39–100, 1991. doi: 10.1007/BF00628304.

I. Heim. *The Semantics of Definite and Indefinite Noun Phrases*. PhD thesis, University of Massachusetts, Amherst, 1982.

L. Henkin. A theory of propositional types. *Fundamenta Mathematicae*, 52(1):323–344, 1963.

# References IV

H. Kamp and U. Reyle. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic, and Discourse Representation Theory*. Number 42 in Studies in Linguistics and Philosophy. Springer, Dordrecht, 1993. doi: 10.1007/978-94-017-1616-1.

G. Kierstead and S. Martin. A multistratal account of the projective Tagalog evidential 'daw'. In *Semantics and Linguistic Theory (SALT) 22*, pages 326–346, 2012. URL `http://journals.linguisticsociety.org/proceedings/index.php/SALT/article/view/2653`.

T. Koev. On the information status of appositive relative clauses. In M. Aloni, V. Kimmelman, F. Roelofsen, G. W. Sassoon, K. Schulz, and M. Westera, editors, *Logic, Language and Meaning*, number 7218 in Lecture Notes in Computer Science, pages 401–410. Springer, Berlin & Heidelberg, 2012. doi: 10.1007/978-3-642-31482-7_41.

# References V

Y. Kubota and R. Levine. Pseudogapping as pseudo-VP ellipsis. In N. Asher and S. Soloviev, editors, *Logical Aspects of Computational Linguistics 2014*, pages 122–137, Heidelberg, 2014. Springer.

Y. Kubota and R. Levine. Pseudogapping as pseudo-VP ellipsis. University of Tsukuba and Ohio State University, available at `http://ling.auf.net/lingbuzz/002504`, ms.

Y. Kubota and R. Levine. Gapping as hypothetical reasoning. *Natural Language and Linguistic Theory*, to appear.

S. Martin. Weak familiarity and anaphoric accessibility in dynamic semantics. In de Groote and Nederhof (2012), pages 287–306. doi: 10.1007/978-3-642-32024-8_19.

S. Martin. *The Dynamics of Sense and Implicature*. PhD thesis, Ohio State University, Columbus, OH, 2013. URL `http://semanticsarchive.net/Archive/DNmMjllM/Martin-dissertation.pdf`.

# References VI

S. Martin. A unidimensional syntax-semantics interface for supplements. In *Workshop on Empirical Advances in Categorial Grammar*, Barcelona, 2015. European Summer School in Logic, Language and Information (ESSLLI) 27.

S. Martin. Supplemental update. *Semantics and Pragmatics*, in press. (`http://semanticsarchive.net/Archive/TMzYTZhY/suppdate.pdf`).

S. Martin and C. Pollard. A higher-order theory of presupposition. *Studia Logica*, 100(4):727–751, 2012a. doi: 10.1007/s11225-012-9427-6.

S. Martin and C. Pollard. Hyperintensional dynamic semantics: Analyzing definiteness with enriched contexts. In de Groote and Nederhof (2012), pages 114–129. doi: 10.1007/978-3-642-32024-8_8.

# References VII

S. Martin and C. Pollard. A dynamic categorial grammar. In G. Morrill, R. Muskens, R. Osswald, and F. Richter, editors, *Formal Grammar 19*, number 8612 in Lecture Notes in Computer Science, pages 138–154, Dordrecht, 2014. Springer. doi: 10.1007/978-3-662-44121-3_9.

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

R. Nouwen. A note on the projection of appositives. In E. McCready, K. Yabushita, and K. Yoshimoto, editors, *Formal Approaches to Semantics and Pragmatics: Japanese and Beyond*, number 95 in Studies in Linguistics and Philosophy, pages 205–222. Springer, Dordrecht, 2014. doi: 10.1007/978-94-017-8813-7_10.

# References VIII

A. Plummer and C. Pollard. Agnostic possible worlds semantics. In *Logical Aspects of Computational Linguistics (LACL) 7*, number 7351 in Lecture Notes in Computer Science, pages 201–212, Berlin & Heidelberg, 2012. Springer. doi: 10.1007/978-3-642-31262-5_14.

C. Pollard. Hyperintensions. *Journal of Logic and Computation*, 18(2): 257–282, 2008. doi: 10.1093/logcom/exm003.

C. Pollard. Agnostic hyperintensional semantics. *Synthese*, 192(3): 535–562, 2015. doi: 10.1007/s11229-013-0373-2.

C. Potts. *The Logic of Conventional Implicatures*. Number 7 in Oxford Studies in Theoretical Linguistics. Oxford University Press, New York, NY & Oxford, 2005.

C. Roberts. Modal subordination and pronominal anaphora in discourse. *Linguistics and Philosophy*, 12(6):683–721, 1989. doi: 10.1007/BF00632602.

# References IX

C. Roberts. Pronouns as definites. In M. Reimer and A. Bezuidenhout, editors, *Descriptions and Beyond*, pages 503–543. Oxford University Press, New York, NY & Oxford, 2005.

C. Roberts. Information structure: Afterword. *Semantics and Pragmatics*, 5(7):1–19, 2012a. doi: 10.3765/sp.5.7.

C. Roberts. Information structure in discourse: Towards an integrated formal theory of pragmatics. *Semantics and Pragmatics*, 5(6):1–69, 2012b. doi: 10.3765/sp.5.6. Accompanying afterword in Roberts 2012a.

P. Schlenker. Supplements without bidimensionalism. Unpublished manuscript, Institut Jean-Nicod and New York University. February, 2013. `www.semanticsarchive.net/Archive/jgwMjNmM/Supplements_without_Bidimensionalism.pdf.`, ms.

# References X

M. Simons, C. Roberts, D. Beaver, and J. Tonhauser. What projects and why. In *Semantics and Linguistic Theory (SALT) 20*, pages 309–327, 2010. URL http://journals.linguisticsociety.org/proceedings/index.php/SALT/article/view/2584.

J. Tonhauser, D. Beaver, C. Roberts, and M. Simons. Toward a taxonomy of projective content. *Language*, 89(1):66–109, 2013. doi: 10.1353/lan.2013.0001.

R. A. van der Sandt. Presupposition projection as anaphora resolution. *Journal of Semantics*, 9(4):333–377, 1992. doi: 10.1093/jos/9.4.333.

L. Wang, E. McCready, and N. Asher. Information dependency in quantificational subordination. In K. von Heusinger and K. Turner, editors, *Where Semantics Meets Pragmatics*, number 16 in Current Research in the Semantics/Pragmatics Interface, pages 267–306. Elsevier, Amsterdam, 2006.