Variable-free semantics and flexible grammars for anaphora

Simon Charlow
Rutgers University

Abstract Jacobson (1999) proposes an account of anaphora that eschews variables and variable assignments, instead treating pronouns as identity functions and extending functional application with operations that pass up and close off anaphoric dependencies. I review the central aspects of Jacobson's variable-free semantics, counterpoising it with standard accounts of anaphora. I discuss conceptual and empirical virtues of Jacobson's theory, and some shortcomings. I sketch an alternative compositional implementation of variable-free semantics that draws on certain design features of the standard account, connect this approach to the computer science concept of 'applicative functors' (and thereby to frameworks as varied as alternative semantics and continuations), and clarify which of the variable-free theory's properties should be regarded as proprietary, and which can be easily repurposed into a theory with variables.

1 Reviewing the standard account

Formal semantic theories characterize the kinds of meanings expressions can have and the procedure that compositionally assembles meanings for complex constituents from the meanings of their parts. A simple model in the Fregean vein is given below: (1) says that meanings can either be entities (type $\texttt{e}$), propositions (type $\texttt{t}$), or functions from meanings to meanings; (2) says that the meaning of a binary branching node is gotten by doing functional application on the meanings of its daughters (in this definition and the others to come, I leave the possibility of backwards application, $[\![\beta]\!][\![\alpha]\!]$, implicit).[1]

(1) $\tau ::= \texttt{e} \mid \texttt{t} \mid \tau \to \tau$

(2) $[\![\alpha\ \beta]\!] := [\![\alpha]\!][\![\beta]\!]$

Figure 1 represents this grammar as a simple deductive system. Rule (2) is characterized by the inference scheme on the left: given (an expression denoting) a function f of type $a \to b$ and (an expression denoting) an argument x of type a, the semantic value of the two together is $f\,x$, of type b. A sample derivation of *Amy saw Bob* is given on the right of Figure 1. Two notational points on the deductive presentation: First, to encourage the reader to reason using types rather than values, I set the latter in gray. Second, the derivations are typeset bottom-up; the resulting 'upside-down' proofs echo the syntactic structures that undergird them.

1 Here is a summary of the notational conventions used throughout this paper. First, types: '$a ::= b$' means that type a is being defined as b; '$a \to b$' names the type of functions from type a to type b; and '$x : a$' means that x has type a. Second, parentheses are dropped whenever possible, under the following conventions: types associate to the right, such that $a \to b \to c$ is equivalent to $a \to (b \to c)$; dually, application associates to the left, such that '$f\,x\,y$' is equivalent to '$(f\,(x))\,(y)$'. Finally, '$\lambda_x\,\Delta$' names the function f such that for all type-appropriate d, $f\,d = \Delta^{x:=d}$ (that is, Δ with d substituted for x).

$$\frac{f\,x : b}{f : a \to b \qquad x : a} \qquad\qquad \frac{\mathsf{saw\,b\,a} : \mathtt{t}}{\mathsf{a} : \mathtt{e} \qquad \dfrac{\mathsf{saw\,b} : \mathtt{e} \to \mathtt{t}}{\mathsf{saw} : \mathtt{e} \to \mathtt{e} \to \mathtt{t} \qquad \mathsf{b} : \mathtt{e}}}$$

Figure 1: Orienting a grammar around functional application, and a derivation of *Amy saw Bob*.

This theory can be used to theorize about a large fragment of natural language. But there's a lot it doesn't handle, including pronouns. No concrete member of type $\mathtt{e}$ is a suitable candidate for the meaning of a free prounoun, whose reference shifts with the context of utterance as in (3), or a bound pronoun, whose meaning is controlled by a higher operator as in (4).

(3) Amy saw **him**.

(4) Every philosopher$_i$ thinks **they**$_i$'re a genius.

In the standard treatment of pronouns (e.g., Heim & Kratzer 1998), meanings are uniformly determined relative to some way of valuing pronouns, generally an *assignment function*, with $[\![\mathrm{pro}_n]\!]^g := g_n$ (that is, whatever the assignment g maps the index n to) and binary-branching nodes interpreted via assignment-relative functional application:

(5) $[\![\alpha\ \beta]\!]^g := [\![\alpha]\!]^g\,[\![\beta]\!]^g$

A rule like (5) requires semantic composition to be done with a specific assignment g in view. Less commitally, we can see meanings *simpliciter* as functions from assignments to values (cf. Lewis 1980). Taking this perspective in lieu of (5), the rule in Figure 1 (left) is replaced with Figure 2 (left), where '$\tau^{\mathtt{g}}$' abbreviates the type $\mathtt{g} \to \tau$ (and '$\mathtt{g}$' is the type of assignments).[2] This lets us derive a type-$\mathtt{t}^{\mathtt{g}}$ meaning for *Amy saw him$_0$* in Figure 2 (right). (Notice in particular how the types "under the $\mathtt{g}$" compose exactly as in Figure 1.) To extract propositional content from the derived meaning $\lambda_g\,\mathsf{saw}\,g_0\,\mathsf{a} : \mathtt{t}^{\mathtt{g}}$, this meaning can be applied to a (contextually furnished) assignment supplying a value for the index 0.

$$\frac{\lambda_g\, f\, g\, (x\, g) : b^{\mathtt{g}}}{f : (a \to b)^{\mathtt{g}} \qquad x : a^{\mathtt{g}}} \qquad\qquad \frac{\lambda_g\,\mathsf{saw}\,g_0\,\mathsf{a} : \mathtt{t}^{\mathtt{g}}}{\lambda_g\,\mathsf{a} : \mathtt{e}^{\mathtt{g}} \qquad \dfrac{\lambda_g\,\mathsf{saw}\,g_0 : (\mathtt{e} \to \mathtt{t})^{\mathtt{g}}}{\lambda_g\,\mathsf{saw} : (\mathtt{e} \to \mathtt{e} \to \mathtt{t})^{\mathtt{g}} \qquad \lambda_g\, g_0 : \mathtt{e}^{\mathtt{g}}}}$$

Figure 2: Assignment-passing functional application, and a derivation of *Amy saw him$_0$*.

We round out this setup with the rule for binding in Figure 3 (left). When this rule applies, m is evaluated at a modified assignment mapping n to x, with x anchored to the functional abstract λ_x, over which $\mathcal{F}$ scopes. Figure 3 (right) shows how this rule allows pronouns to be bound, with a derivation of *everyone t$_0$ likes their$_0$ mom*.[3]

2 This rule corresponds closely to von Fintel & Heim's (2011) intensional interpretation function $[\![\cdot]\!]_{\mathfrak{c}}$. Related approaches to assignments include Montague 1970, Rooth 1985, Poesio 1996, and Sternefeld 1998, 2001.

3 This structure might be generated by covert movement (i.e., Quantifier Raising) of *everyone* or, more likely, overt movement out of a vP-internal subject position (e.g., Heim & Kratzer 1998: 218ff).

$$\frac{\lambda_g\, \mathcal{F}\, g\, (\lambda_x\, m\, g^{n:=x}) : \mathtt{t}^{\mathsf{g}}}{\mathcal{F} : ((\mathsf{e} \to \mathsf{t}) \to \mathsf{t})^{\mathsf{g}} \qquad m : \mathtt{t}^{\mathsf{g}}}\ \rhd_n \qquad \frac{\lambda_g\, \text{eo}\, (\lambda_x\, \text{likes}\, (\text{mom}\, x)\, x) : \mathtt{t}^{\mathsf{g}}}{\lambda_g\, \text{eo} : ((\mathsf{e} \to \mathsf{t}) \to \mathsf{t})^{\mathsf{g}} \quad \dfrac{\lambda_g\, \text{likes}\, (\text{mom}\, g_0)\, g_0 : \mathtt{t}^{\mathsf{g}}}{\vdots \quad \vdots \quad \vdots}}\ \rhd_0$$

Figure 3: A rule for binding, and an abbreviated derivation of *everyone* t_0 *likes their*$_0$ *mom.*

2 Variable-free semantics

2.1 Pronouns

Jacobson (1999) proposes a variable-free account of pronouns and binding that eschews indices and assignments, instead treating pronouns as identity functions on entities, as in (6).[4] I'll present the basics of Jacobson's system before reviewing empirical and conceptual motivations in Sections 3 and 4.

(6) $[\![\text{pro}]\!] := \lambda_x\, x : \mathsf{e}^{\mathsf{e}}$

The definition repurposes the superscripting convention for types introduced in Section 1: e^{e} is inhabited by functions from individuals to individuals. Importantly, though, we will follow Jacobson in grammatically distinguishing $a \to b$ and b^a, even though these types are potentially inhabited by the same functions. More on this soon.

2.2 Composition

There is an immediate compositional challenge: pronouns are type type e^{e} but occur in places where something of type e is expected. This is analogous to variable-full theories, where pronouns are type e^{g}. But whereas theories with variables deal with pronouns by 'generalizing to the worst case', making every lexical meaning assignment-relative and *replacing* functional application wholesale with the assignment-passing enrichment in Figure 2, Jacobson retains conservative lexical entries for non-pronominals, *supplementing* functional application with additional compositional rules.

Jacobson's central rules are G and Z, defined in Figure 4. G lets a function $f : a \to b$ apply to an 'incomplete' value $m : a^c$ by function-composing f with m, passing up the anaphoric dependency. Z similarly performs a kind of application to an 'incomplete' value $m : a^c$, but it does so by identifying the missing c with the second argument of $f : a \to c \to b$, closing off the anaphoric dependency. Note: I've given G and Z initially as binary compositional rules since I believe this presentation is more intuitive for beginners, but readers should keep in mind that on Jacobson's official proposal, G and Z are unary rules ('type-shifters') which apply directly to the function f (e.g., G maps $f : a \to b$ to $\mathsf{G}\, f : a^c \to b^c$). We will occasionally appeal to unary versions as we go.

4 Hepple (1990) was the first author to treat pronouns as identity functions. Other variable-free treatments of anaphora exist which I regrettably lack the space to discuss here. The reader is referred to works by Szabolcsi (1983, 1989, 1992, 2003), Steedman (1987, 2000), Dowty (1992, 2007), Dekker (1994), van Eijck (2001), Shan (2001, 2004), Jäger (2005), and de Groote (2006).

$$\dfrac{f : a \to b \qquad m : a^c}{\lambda_x f\,(m\,x) : b^c}\;\mathsf{G} \qquad\qquad \dfrac{f : a \to c \to b \qquad m : a^c}{\lambda_x f\,(m\,x)\,x : c \to b}\;\mathsf{Z}$$

Figure 4: The G and Z rules, which supplement functional application, Figure 1 (left).

A derivation of *every boy$_i$ thinks Amy saw him$_i$* illustrating the basic workings of Jacobson's system appears in Figure 5. Focus first on the embedded clause, *Amy saw him*. G allows saw to compose with the pronominal object, followed by the subject, resulting in a meaning of type $\mathtt{t}^{\mathtt{e}}$ (since G requires the constituent hosting the pronoun to play the role of argument to some function, we invoke Lift on the embedded subject (Partee 1986, Winter this volume) to turn it into such a function). Next, an application of Z is used to compose this clause with think, yielding a garden-variety predicate which identifies the values of the matrix subject and the embedded pronoun: $\lambda_x \mathsf{thinks}\,(\mathsf{saw}\,x\,\mathsf{a})\,x : \mathtt{e} \to \mathtt{t}$. The quantified subject merges via regular functional application, and we are done. The result represents the binding reading: for every boy x, x thinks Amy saw x.

$$\dfrac{\mathsf{eb} : (\mathtt{e} \to \mathtt{t}) \to \mathtt{t} \qquad \dfrac{\mathsf{thinks} : \mathtt{t} \to \mathtt{e} \to \mathtt{t} \qquad \dfrac{\dfrac{\mathsf{a} : \mathtt{e}}{\lambda_\kappa \kappa\,\mathsf{a} : (\mathtt{e} \to \mathtt{t}) \to \mathtt{t}}\;\mathsf{Lift} \qquad \dfrac{\mathsf{saw} : \mathtt{e} \to \mathtt{e} \to \mathtt{t} \qquad \lambda_x x : \mathtt{e}^{\mathtt{e}}}{\lambda_x \mathsf{saw}\,x : (\mathtt{e} \to \mathtt{t})^{\mathtt{e}}}\;\mathsf{G}}{\lambda_x \mathsf{saw}\,x\,\mathsf{a} : \mathtt{t}^{\mathtt{e}}}\;\mathsf{G}}{\lambda_x \mathsf{thinks}\,(\mathsf{saw}\,x\,\mathsf{a})\,x : \mathtt{e} \to \mathtt{t}}\;\mathsf{Z}}{\mathsf{eb}\,(\lambda_x \mathsf{thinks}\,(\mathsf{saw}\,x\,\mathsf{a})\,x) : \mathtt{t}}$$

Figure 5: A typical variable-free derivation: *every boy$_i$ thinks Amy saw him$_i$*.

Of course, pronouns do not *need* to be bound. In such cases, the anaphoric dependency is simply passed up all the way: G replaces Z in Figure 5, and another instance of G is used to fold in the matrix subject. The result, $\lambda_x \mathsf{eb}\,(\mathsf{thinks}\,(\mathsf{saw}\,x\,\mathsf{a})) : \mathtt{t}^{\mathtt{e}}$, is a function from individuals x to the proposition that every boy thinks Amy saw x, and can be applied by the speaker/hearer to a contextually salient individual. Though readers may find it odd to think of a sentence with an unbound pronoun as denoting a property of individuals, Jacobson argues that the standard account is not really in better shape, since in that theory *all* sentences denote properties, of assignments (as in Figure 2).

Now's a good time to see why we can't conflate $a \to b$ and b^a. If we did, the embedded VP's meaning, $\lambda_x \mathsf{saw}\,x : (\mathtt{e} \to \mathtt{t})^{\mathtt{e}}$, could apply directly to the embedded subject $\mathsf{a} : \mathtt{e}$, yielding $\mathsf{saw}\,\mathsf{a} : \mathtt{e} \to \mathtt{t}$ and disastrously scrambling subject and object. Distinguishing $a \to b$ and b^a prevents this undesirable outcome, guaranteeing that a superscripted type can only be passed up by G or discharged (bound) by Z.

2.3 Stepping back

The central features of Jacobson's theory are as follows. Pronouns lack indices, whence the 'variable-free' moniker. (Don't be misled by the presence of variables and variable binding in our metalanguage. 'Variable-free' is a claim about the *object language*.) Given the absence of indices, there can be no co-indexing relationships between pronouns and anything else. Instead, the 'co-construal' relationship between a pronoun and its binder is established *combinatorially*: G passes up anaphoric dependencies, and Z closes them off just before the binder is merged. Likewise, it is impossible to formulate rules that refer to, let alone constrain, the distribution of indices — e.g., Conditions A, B, and C of the Binding Theory, prohibitions on Weak Crossover, etc. Insofar as such phenomena are grammatical, they will need to be captured some other way. (There is reason for optimism, though. I discuss Weak Crossover in Sections 5.2 and 6.2.)

Jacobson's account is also *flexible* in a way the variable-full account is not. Whereas the latter theory treats all expressions as assignment-sensitive (often trivially so) and therefore lacks any non-assignment-passing modes of composition, Jacobson supplements functional application with additional modes of composition. We will see in Section 6 that theories with variables and assignments can avail themselves of a similar flexibility in lieu of the usual uniformity-based approach. This points the way to improvements in both variable-full and variable-free systems and lets us clearly see where their fundamental differences (and similarities) lie.

3 Some empirical results

This section fleshes out the bare-bones system of Section 2 with a basic account of relativization. It does this by extending our baseline variable-free theory with simple hypotheses about the meanings of gaps and relative pronouns. I use this slightly modified substrate to showcase some marquee empirical results of variable-free semantics.

3.1 Gaps, pied-piping, and *i*-within-*i*

We'll begin by implementing a simple account of extraction in relative clauses. Some basic examples are given in (7)–(9), where '[]' marks an extraction gap. Summarizing, (7) is a basic object relativization construction; (8) is subject relativization with binding of a pronoun; and (9) involves pied-piping of *-'s mom* by the relative pronoun *who*.

(7) The woman who John saw [] left.

(8) The woman who $[]_i$ married her_i childhood sweetheart left.

(9) The woman whose mom John saw [] left.

What's the semantics of gaps and relative pronouns? We make the typical assumption that the semantics of gaps is *pronominal*, and (somewhat less typically) extend this treatment to relative pronouns such as *who* (Jacobson 1998: 81). In the present setting, this means gap and relative pronoun denotations are identity functions, type e^{e}.[5]

This theory of gapped structures differs from Jacobson's in that it assumes (with, e.g., Shan & Barker (2006)) that gaps are null pronominal elements, while Jacobson

5 Since pronouns can't be replaced with gaps while preserving well-formedness (though gaps *can* be replaced with resumptive pronouns), gaps and pronouns should really be given different types. It's straightforward to do so (e.g., pronouns are type e^{e} and gaps are type e_{e}) and to generalize G/Z accordingly. (The fact that pronouns can replace gaps may suggest that e^{e} is a proper subtype of e_{e}, cf. Bernardi & Szabolcsi 2008.)

does not actually countenance gaps! Instead, in the tradition of Combinatory Categorial Grammar, Jacobson uses syntactic function composition to assemble gapped clauses, without gaps (cf., e.g., Lambek 1958, Ades & Steedman 1982, Szabolcsi 1989, 1992, Steedman 1987, 2000). Why admit gaps? First, it streamlines the presentation. Second, and more importantly, I wish to highlight that the key ideas and techniques of Jacobson 1999 are independent of many of the syntactic assumptions therein.

With our meanings for gaps and relative pronouns we have essentially all we need to account for (7)–(9). Figure 6 shows this in one fell swoop with a derivation of *whose mom* $[]_i$ *saw herself*$_i$, a relative clause with binding and pied-piping. The relative pronoun, gap, and reflexive pronoun all denote identity functions. The application of Z guarantees that the subject and object of the relative clause are covalued, and the applications of G pass up pronominal dependencies, as usual.[6] I additionally appeal to a Front rule which recognizes a complete relative clause and adjusts its type to that of a property, which allows the fronted phrase *whose mom* to be folded in (cf. Shan & Barker 2006: 114).

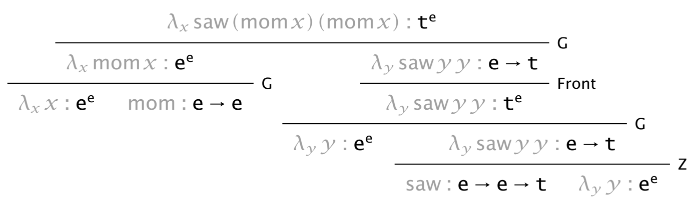

Figure 6: Deriving the pied-piped relative clause *whose mom* $[]_i$ *saw herself*$_i$.

The most notable feature of this derivation is that the meanings of gaps and relative pronouns, together with the G rule, generate an immediate account of pied-piping (as pointed out in Jacobson 1998). Thus, though the relative pronoun's movement brings along additional material, that material is in the end interpreted as if it were in situ (that is, in the gap). In other words, a form of semantic reconstruction in pied-piping is automatic, in contrast with many standard theories (see, e.g., Engdahl 1986, Nishigauchi 1990, Cresti 1995, von Stechow 1996, Sharvit 1997, Heck 2008, Cable 2010, Dayal 2016).

A final empirical point discussed by Jacobson (1994b, 2014) is so-called *i*-within-*i* effects in relational noun phrases. Examples (10) and (11) are ungrammatical, though the meaning that they are trying to express, that of (8), is coherent.

(10) *The wife$_i$ of her$_i$ childhood sweetheart left.

(11) *Her$_i$ childhood sweetheart's wife$_i$ left.

As Jacobson notes, Z provides a ready explanation of these facts. Z insists that its input be a two-place function $f : a \to c \to b$ (Figure 4). But relational nouns like *wife*, though semantically two-place, are syntactically one-place. (Jacobson points out that relational nouns are impossible in small clauses such as **with Calista wife of Bill*.) Thus, relational

6 Nothing I've said explains why a reflexive's antecedent must be a higher argument of the same predicate (but see Szabolcsi 1989, 1992 for a variable-free treatment of reflexives). I've used the reflexive here because Jacobson's theory does not actually generate a binding reading of *whose*$_i$ *mom [] saw her*$_i$. See Section 5.2.

nouns should in fact be of type $\mathtt{e} \to \mathtt{n}$ (with '$\mathtt{n}$' the type of nouns and noun phrases), rather than $\mathtt{e} \to \mathtt{e} \to \mathtt{t}$. $\mathtt{Z}$ cannot apply, and so binding is correctly ruled out. Parallel explanations appealing to the intransitivity of relational nouns are available in standard theories, but it is remarkable that *i*-within-*i* facts fall out of the variable-free system.

3.2 Functional gaps and pronouns

Functional readings, exemplified by (12) and (13), suggest that *wh* and nominal quantification sometimes ranges over functions (e.g., Engdahl 1980, 1986, Groenendijk & Stokhof 1983, Sharvit 1997, 1999, Winter 2004). Standard analyses of these examples have two features. First, the gapped constituents (underlined) have functional meanings. The question in (12) requests an $f : \mathtt{e} \to \mathtt{e}$ such that every Englishman x loves $f\,x$; and the subject of (13) denotes the (woman-valued) $f : \mathtt{e} \to \mathtt{e}$ such that every Englishman x loves $f\,x$. Second, the meaning of *his$_i$ mom* is $\lambda_x\, \mathtt{mom}\, x : \mathtt{e} \to \mathtt{e}$. In (12) this function is the answer to the question; in (13) it is equated with the meaning of the copular subject.

(12) Who does every Englishman$_i$ love []? His$_i$ mom.

(13) The woman every Englishman$_i$ loves [] is his$_i$ mom.

What explains functional gaps? Standard accounts following Engdahl 1980 posit that functional gaps are *complex variables*. In existing compositional implementations this requires stipulations and/or interpretive apparatus beyond what is strictly necessary to account for pronouns and binding in simple cases. The issue is that one expression, the gap, contributes a complex meaning $f\,x$ to the functional gapped clause, even though in non-functional readings gaps contribute a simplex meaning, with a single variable. It is unclear how a single meaning for gaps can do all this work (but cf. Charlow 2019).

A significant virtue of variable-free semantics, one emphasized by Jacobson (1999), is its ready compatibility with functional readings. To show this, I'll sketch a variable-free treatment of the functional readings of (12) and (13)'s gapped clauses.

So far we have treated variable-free pronoun meanings as identity functions with type $\mathtt{e}^{\mathtt{e}}$. We now generalize this treatment a bit. After all, an identity function is naturally polymorphic: its type is $a \to a$, for any type a (cf. Barker 2018; Pierce 2002 is a useful reference on polymorphism). A generalized pronominal type, then, would be a^a, again for any type a. This is in fact a tad *too* general, since pronouns should eventually deliver a type-$\mathtt{e}$ value to the semantics (rather than something ridiculous like, say, a relation). In other words, only a restricted sub-space of identity function types is suitable for pronouns. I therefore propose the following hierarchy of pronominal types:

- $\mathtt{e}^{\mathtt{e}}$ is a pronominal type.
- If b^a is a pronominal type, $(b^{\mathtt{e}})^{(a^{\mathtt{e}})}$ is a pronominal type.
- Nothing else is a pronominal type.

It should be emphasized that the semantics of pronouns/gaps is unitary and unchanged: a pronoun's meaning is $\lambda_x\, x$. But x may be of type $\mathtt{e}$, $\mathtt{e}^{\mathtt{e}}$, $(\mathtt{e}^{\mathtt{e}})^{\mathtt{e}}$, and so on.

The generalization of pronominal types is all that is required to derive functional meanings in gapped structures; G and Z do the rest. A schematic derivation of $Q\,V\,[\,]$ (e.g., *every Englishman loves [ ]*) is given in Figure 7. The gap meaning is $\lambda_f f : (\mathsf{e}^{\mathsf{e}})^{(\mathsf{e}^{\mathsf{e}})}$, an identity function over type-e^{e} meanings, and the derivation is a mundane series of applications of G and Z. One notable feature, foreshadowed in Section 2.2, is that Z is applied here as a *unary* rule (as per Jacobson's official proposal), turning V's type from $\mathsf{e} \to \mathsf{e} \to \mathsf{t}$ to the out-to-bind $\mathsf{e}^{\mathsf{e}} \to \mathsf{e} \to \mathsf{t}$, which then composes via G with the functional gap. The result is a property of functions f such that Q-many x's stand in the V relation to $f\,x$; the complex gap meaning $f\,x$ has been conjured combinatorially.

$$\dfrac{Q : (\mathsf{e} \to \mathsf{t}) \to \mathsf{t} \qquad \dfrac{\dfrac{V : \mathsf{e} \to \mathsf{e} \to \mathsf{t}}{\lambda_f \lambda_x V\,(f\,x)\,x : \mathsf{e}^{\mathsf{e}} \to \mathsf{e} \to \mathsf{t}}\,\mathsf{Z} \qquad \lambda_f f : (\mathsf{e}^{\mathsf{e}})^{(\mathsf{e}^{\mathsf{e}})}}{\lambda_f \lambda_x V\,(f\,x)\,x : (\mathsf{e} \to \mathsf{t})^{(\mathsf{e}^{\mathsf{e}})}}\,\mathsf{G}}{\lambda_f Q\,(\lambda_x V\,(f\,x)\,x) : \mathsf{t}^{(\mathsf{e}^{\mathsf{e}})}}\,\mathsf{G}$$

Figure 7: A schematic analysis of functional readings.

Finally, as noted, (12) and (13) suggest a symmetry between their functional gapped structures and *his*$_i$ *mom*: the former denotes a property of functions (Figure 7), and the latter denotes such a function. On standard theories, the second half of this symmetry requires us to bind the free variable *his*$_i$. While that's feasible, it's noteworthy that the variable-free account requires no such shift. If *-'s mom* denotes an $\mathsf{e} \to \mathsf{e}$ function from people to their mothers, the meaning of *his mom* (i.e., *he's mom*) is $\lambda_x\, \mathsf{mom}\, x : \mathsf{e}^{\mathsf{e}}$ (via G).

This variable-free analysis of functional gaps extends immediately to examples of functional pronouns, more commonly known as *paycheck pronouns* and exemplified in (14) (e.g., Karttunen 1969, Cooper 1979, Engdahl 1986, Jacobson 2000a). Here the second sentence means that every philosopher x spent $f\,x$, with f a function from x to x's paycheck. As in (12) and (13), the function in question corresponds to other material in the example, here *their paycheck*, which the functional pronoun is anaphoric to.

(14) Every linguist$_i$ deposited [their$_i$ paycheck]$_j$. Every philosopher$_k$ spent it$_j$.

Figure 7 doubles as an analysis of paycheck pronouns (Jacobson 2000a). Set Q to the meaning of *every philosopher* and V to the meaning of *spent*. G, Z, and the pronoun's general type yield a type-$\mathsf{t}^{(\mathsf{e}^{\mathsf{e}})}$ property of functions f such that every philosopher x spent $f\,x$. This property may be applied to a salient e^{e} function to yield a proposition. A natural candidate in (14) is $\lambda_x\, \mathsf{paycheck}\, x : \mathsf{e}^{\mathsf{e}}$, the meaning of *their paycheck*![7]

3.3 Pointers to other work

There is a great deal more work on variable-free semantics than I can hope to cover in this contribution. Jacobson herself develops and defends variable-free semantics in a

7 Jacobson assumes pronouns are type e^{e} and derive further instances of our pronominal type by generalizing G to to apply to functions of type b^a in addition to functions of type $a \to b$ (Jacobson 2014: 329). While this is elegant, allowing pronouns to be polymorphic incurs little cost and streamlines the presentation.

substantial body of work (1992, 1994a, 1994b, 1996, 1998, 2000a, 2000b, 2004, 2014, 2016, inter alia). While the empirical focus of these contributions is pronominal binding and adjacent phenomena, Jacobson has also argued that the variable-free architecture offers (or is consistent with) competitive accounts of verb phrase ellipsis and antecedent-contained deletion (1992, 1998, 2004, 2009; see also Szabolcsi 1992, 2013, Charlow 2008). Subsequent to Jacobson's main developments, continuations-based theories of anaphora and binding (e.g., Shan & Barker 2006, Barker & Shan 2008, 2014, Barker 2018) have been developed which extend variable-free semantics with general semantic mechanisms for scope-taking and fine-grained control of evaluation order. The reader is additionally referred to the contributions listed in footnote 4, some of which approach variable-free semantics from a very different set of foundations than Jacobson.

4 Some conceptual points

4.1 Direct Compositionality

Part of Jacobson's motivation for variable-free semantics is a methodological orientation towards *Direct Compositionality* ('DC'). The essence of DC is that surface structures directly receive model-theoretic interpretations, without mediating levels of representation such as LF — indeed, without any (explanatorily indispensable) notion of logical form whatsoever. The output of any structure-building operation is thus *immediately* paired with a corresponding interpretation, guaranteeing that the syntax and the semantics operate in tandem (e.g., Barker & Jacobson 2007). According to Jacobson, DC is, other things being equal, a simpler, more austere, and ultimately more explanatory view of the syntax-semantics interface than its alternatives.

While I find the latter point persuasive, it is important to note that there is no *necessary* connection between variable-free semantics and DC: theories with variables may be DC, and theories without them can fail to be. (Jacobson (2014: xviii–xix) acknowledges as much.) What does seem true is that variable-free semantics reduces the appeal of certain non-DC architectures. As noted in Section 2.3, rules that regulate the distribution of variables in LF cannot be formulated if there are no variables to begin with. And the operation of Quantifier Raising, in some respects the central motivation for LF, is incompatible in its standard formulation with variable-free semantics (since the expression it leaves behind is a variable). Overall it is probably fair to say that variable-free semantics makes Direct Compositionality more appealing at the margins.

4.2 Issues with variables

Compositionality aside, one might simply be skeptical that variables have an explanatory role to play; after all, natural language syntax certainly *looks* variable-free.[8] With Harris (2019), we might add that it can seem absurd to think language users to bear referential intentions towards assignments. The commitments of the variable-free theory appear

8 Recently, Kuhn (2016) has motivated a variable-tree treatment of spatial loci in American Sign Language (which had been argued to be overt manifestations of variables, cf. Schlenker 2011).

lighter. 'Open' propositions require mere referential intentions towards individuals — or towards sequences of individuals, in the general case (Sections 5.1 and 6.3).[9]

Variables can also create unintended technical issues. I might wish for assignments that can value variables of arbitrary types (for cross-categorial movement, semantic reconstruction, and so on; see Heim & Kratzer 1998: 213, Charlow 2019, and citations therein). However, Muskens (1995: 179ff) cautions that a type theory with assignments powerful enough to value variables of any type requires additional stipulations to avoid inconsistency. Again, the variable-free theory is on solid ground, since all it requires to value 'variables' of arbitrary types is the identity-functional type $a \to a$.

Finally, variables cause problems for processes like ellipsis that are licensed (as is standardly supposed) in virtue of syntactic or semantic identity. 'Rebinding' configurations like (15) are especially vexing (Evans 1988, Rooth 1992b, Takahashi & Fox 2005). The sentence allows a 'sloppy' reading, i.e., with Δ = likes him_j, and so Δ must be identical (either in form or meaning) with the antecedent VP *likes* $\textit{him}_i$. But, then again, it *can't* be: the overt him_i and the elided him_j are distinct variables.[10] (Additionally, since the sloppy pronoun him_j in (15) isn't bound by the subject, the antecedent and elided VPs cannot be treated as 'alphabetic variants', cf. Keenan 1971, Sag 1976.)

(15) John_i's mom likes him_i. Bill_j's mom does Δ too.

The solution proposed by Rooth (1992b) (and widely adopted in subsequent literature) is that the syntactic identity requirement on ellipsis licensing *allows variables to differ*. But this is, of course, suspicious: variables exist, and variables matter — until they don't.

5 Some problems

5.1 Generalized rules

The appealingly simple picture summarized in Figure 4 under-generates. Constructions like (16) are impossible to analyze: the subject is type $\mathtt{e}^{\mathtt{e}}$, and the VP is readily assigned type $(\mathtt{e} \to \mathtt{t})^{\mathtt{e}}$ (i.e., via $\mathtt{G}$). But that is as far as we can go: we have no ability to combine these two values to derive the expected final value $\lambda_y \lambda_x\, \mathtt{saw}\, y\, x : (\mathtt{t}^{\mathtt{e}})^{\mathtt{e}}$.

(16) He_i saw her_j.

(17) Every professor_i gave their_i opinion to the dean.

Example (17) highlights an issue for $\mathtt{Z}$. The formulation in Figure 4 allows a pronoun inside the intial argument of some function $f : a \to c \to b$ to be bound by the *very next* argument of f (i.e., the c). In (17), however, the pronoun inside the direct object is bound by the matrix subject, skipping over the indirect object (assuming the verb combines first with the direct object).

9 However, sequences of individuals are isomorphic to partial assignments, and so it is appropriate to take this argument with a grain of salt. See Section 6.3 for discussion of a related point.

10 Allowing the overt and elided pronouns to be 'accidentally' identical for the purpose of licensing the ellipsis creates enormous problems of its own, as discussed by Heim (1997: 217ff).

Jacobson (1999: 138ff) is aware of both issues and proposes generalizations of G and Z allowing these rules to 'skip over' arbitrary amounts of irrelevant material. I do not give the fully general versions of G and Z here, but Figure 8 provides the relevant (unary) instances of the general schemas (additional instances simply allow for more and more intervening material in place of the type *d*).

$$\frac{f : (a \to b)^d}{\lambda_y \lambda_m \lambda_x\, f\, y\, (m\, x) : (a^c \to b^c)^d}\ \mathsf{G} \qquad \frac{f : a \to d \to c \to b}{\lambda_m \lambda_y \lambda_x\, f\, (m\, x)\, y\, x : a^c \to d \to c \to b}\ \mathsf{Z}$$

Figure 8: Further instances of generalized G and Z rules.

In lieu of presenting detailed derivations of (16) and (17), I simply note that generalized G allows *saw her* to shift from type $(\mathtt{e} \to \mathtt{t})^{\mathtt{e}}$ to type $(\mathtt{e}^{\mathtt{e}} \to \mathtt{t}^{\mathtt{e}})^{\mathtt{e}}$, making room for the pronominal subject in the inner $\mathtt{e}^{\mathtt{e}}$. Meanwhile, generalized Z allows *gave* to shift from type $\mathtt{e} \to \mathtt{e} \to \mathtt{e} \to \mathtt{t}$ to type $\mathtt{e}^{\mathtt{e}} \to \mathtt{e} \to \mathtt{e} \to \mathtt{t}$, with the superscripted and right-most e's identified. So the generalized formulations of G and Z solve our issues.

There is a sense in which G need not actually be generalized, so long as we admit *nullary* occurrences of our compositional rules, i.e., we allow G/Z to apply to themselves. As the formulation of Jacobson (1999: 138) suggests, 'generalized' G is just G applied to itself (and further instances can be derived by iteratively G-ing G).[11]

The generalization of Z is more costly, and gives rise to an infinite family of distinct Z operations (since in principle there is no bound on the number of arguments betwen a^c and c). More technically, the generalization of Z is a species of 'ad hoc' polymorphism (in which an operation behaves *differently* depending on the types of its inputs) and contrasts with 'parametric' polymorphism (in which an operation behaves *uniformly* on inputs of different types; e.g., $\lambda_x\, x : a \to a$ is a parametric-polymorphic identity function). Other things being equal, theories without ad hoc polymorphism may be preferable to, and potentially more explanatory than, theories requiring it.

5.2 Binding between non-coarguments

Even with the generalization of Z, other under-generation issues remain. Importantly, (generalized) Z only allows binding relationships to be established between two coarguments of the same predicate (with the binder the 'higher' or later argument of that predicate, and the bind-ee contained in the 'lower' or earlier argument). Effectively, this means that binding needs surface c-command, much as proposed by Reinhart 1983.

This is a two-edged sword. On the one hand, as Jacobson (1999: 135ff) points out, it correctly rules out instances of Weak Crossover like (18). This is especially notable in view of the fact that traditional accounts of Weak Crossover (such as Reinhart's) are representational in nature. As mentioned back in Section 2.3, this is one reason to be optimistic that the variable-free program is consistent with grammatical explanations of phenomena that have previously received only representational accounts.

(18) *His$_i$ mom likes every boy$_i$.

11 That is, $\mathsf{G}\,\mathsf{G} : ((a \to b) \to a^c \to b^c) \to (a \to b)^d \to (a^c \to b^c)^d$. Thanks to Jeremy Kuhn for pointing this out.

A challenge for Jacobson's non-representational account of Weak Crossover and Reinhart's representational theory alike is that binding doesn't require surface c-command (Safir 2004, Barker 2005, 2012), and arguably doesn't require 'LF c-command' (i.e., scope) either (Kamp 1981, Heim 1982). Against surface c-command, we have possessor binding (19), inverse linking binding (20), and binding into adjuncts (21). Against 'LF c-command', we have cross-sentential anaphora (22) and donkey anaphora (23).

(19) Everyone$_i$'s mom likes them$_i$.

(20) Somebody from every city$_i$ likes it$_i$.

(21) We will sell no wine$_i$ before its$_i$ time.

(22) {A, exactly one} linguist$_i$ walked in the park. She$_i$ whistled.

(23) If there's [a train or a bus]$_i$ leaving Dallas, I hope you're on it$_i$.

It would arguably be too much to demand an account of (22) and (23) from a 'static' theory of binding such as Jacobson's (see the conclusion for more on this point). But the inconsistency of $\mathtt{Z}$ with cases such as (19), (20), and (21) suggests that the explanation of Weak Crossover and the sole reliance on $\mathtt{Z}$ to effect binding are untenable.[12]

5.3 On becoming variable-free

Variable-free semantics is both conceptually and empirically attractive. However, it isn't embedded in a general theory of which kinds of supplements to functional application ('type-shifts') are available in grammar, or related to existing modes of theorizing about other kinds of phenomena. Importantly, general techniques for *integrating* variable-free theorizing with extant semantic architectures have not been specified, such that it is unclear what tradeoffs or new possibilities might be associated with going variable-free.

Jacobson herself has considered the interaction of variable-free semantics with 'alternative semantics', a framework proposed Hamblin (1973) for questions and extended to focus by Rooth (1985, 1992a, 1996). In Roothian alternative semantics, constituents are associated with sets of alternative meanings, their 'focus values'. Jacobson (2000b, 2004) argues that focus values of constituents with pronouns may in some circumstances be sets of functions, type $a^{\mathtt{e}} \to \mathtt{t}$, and in others functions into sets, type $(a \to \mathtt{t})^{\mathtt{e}}$. However, she does not give general mechanisms deriving these results.

6 New foundations

6.1 The essence of the standard theory

Our strategy for improving variable-free semantics will be to crib some tools from the standard account based on assignments, while retaining the flexibility and modularity

12 Another potential challenge comes from neo-Davidsonian event semantics, especially varieties which sever one or more of the verb's arguments (e.g., Parsons 1990, Kratzer 1996, Champollion 2015). For illustration, suppose with Kratzer that a transitive verb is type $\mathtt{e} \to \mathtt{v} \to \mathtt{t}$ ($\mathtt{e}$ is the object and 'v' names the type of events). Since the subject argument is severed, $\mathtt{Z}$ cannot be used to effect binding between the subject and any pronouns contained in the verb's object (similarly in theories that sever all the verb's arguments).

of Jacobson's variable-free architecture (and, of course, her lexical semantics for pronouns!). Recall that the standard account is built on *uniformity*: lexical meanings are all assignment-relative, and meaning composition always passes around assignments.

Another approach is to abstract out and modularize the core features of the standard account. Instead of treating the lexical-semantic values of non-pronominals as uniformly, trivially dependent on an assignment, we invoke a function η which turns any x into a constant function from assignments into x (Figure 9, left). Instead of making semantic composition uniformly relative to an assignment, we'll help ourselves to a function $\circledast$ which performs assignment-friendly function application on demand (Figure 9, right).

$$\frac{\lambda_y\, x : a^r}{x : a}\,\eta \qquad\qquad \frac{\lambda_y\, m\, y\, (n\, y) : b^r}{m : (a \to b)^r \qquad n : a^r}\,\circledast$$

Figure 9: Abstracting out the two operations underlying environment-sensitivity.

The definitions replace the superscripted $\mathtt{g}$ with r, which can stand in for any type (for example, $\mathtt{e}$!). The reason to (parametrically) generalize η and $\circledast$ in this way is that their form has exactly nothing to do with assignment functions per se. Rather, these operations together capture the logic of *environment-sensitive* composition, a feature shared equally by variable-full and variable-free architectures: in theories with variables, the environment relative to which values are determined is an assignment; in variable-free theories, the environment relative to which values are determined is an individual.

Thus, the operations in Figure 10 offer a completely general way to do environment-sensitive composition, with or without variables. Variable-full and variable-free derivations of *she left* using η and $\circledast$ are provided in Figure 10. We use η to 'lift' environment-insensitive values, and $\circledast$ to combine environment-sensitive functions and arguments.

$$\frac{\lambda_g\, \mathsf{left}\, g_0 : \mathtt{t}^{\mathtt{g}}}{\lambda_g\, g_0 : \mathtt{e}^{\mathtt{g}} \quad \dfrac{\lambda_g\, \mathsf{left} : (\mathtt{e} \to \mathtt{t})^{\mathtt{g}}}{\mathsf{left} : \mathtt{e} \to \mathtt{t}}\,\eta}\,\circledast \qquad\qquad \frac{\lambda_x\, \mathsf{left}\, x : \mathtt{t}^{\mathtt{e}}}{\lambda_x\, x : \mathtt{e}^{\mathtt{e}} \quad \dfrac{\lambda_x\, \mathsf{left} : (\mathtt{e} \to \mathtt{t})^{\mathtt{e}}}{\mathsf{left} : \mathtt{e} \to \mathtt{t}}\,\eta}\,\circledast$$

Figure 10: Variable-full and variable-free composition of *she left* using η and $\circledast$.

These derivations highlight that η and $\circledast$ are a *decomposition* of $\mathtt{G}$: instead of directly using $\mathtt{G}$ to compose $\mathsf{left} : \mathtt{e} \to \mathtt{t}$ and $\lambda_x\, x : \mathtt{e}^{\mathtt{e}}$, we first use η and then $\circledast$. Additionally, we observe that the lexical and compositional uniformity of the standard account turn out to be negotiable: a semantics oriented around variables can be compositionally grounded using precisely the same flexibly applying operations as the variable-free system.[13]

6.2 Binding

The re-engineered variable-free semantics needs just one more piece: a way to effect binding. Whereas Jacobson's appeal to $\mathtt{Z}$ under-generates by linking binding too closely

13 Moreover, this flexibility allows a theory with variables to behave much like variable-free semantics with respect to functional readings. See Charlow 2019 for discussion.

to the semantic structure of the verb (and thereby to coargument-hood and surface c-command), the standard account does better. For example, the rule in Figure 3 easily generates possessor binding for LFs like *everyone t_0's mom likes them$_0$*.

In effect, the standard account improves on Z here because it links binding to scope rather than coargument-hood (as appropriate for a static semantics). This feature can be imported into the variable-free setting. The rule in Figure 11 gives a scopal expression $\mathcal{F} : (\mathtt{e} \to \mathtt{t}) \to \mathtt{t}$ an anaphoric charge, turning it into something which expects to scope over and plug two e-sized holes, one of them pronominal.[14]

$$\frac{\lambda_\kappa \mathcal{F}\,(\lambda_x\, \kappa\, x\, x) : (\mathtt{e} \to \mathtt{t}^{\mathtt{e}}) \to \mathtt{t}}{\mathcal{F} : (\mathtt{e} \to \mathtt{t}) \to \mathtt{t}} \triangleright$$

Figure 11: A rule for binding that gives the binder an anaphoric charge.

Variable-free treatments of scope (e.g., Hendriks 1993, Shan & Barker 2006, Szabolcsi 2011) go well beyond what I can cover here (but see Appendix A). Instead I take a shortcut, defining a descriptive 'rule' for scope in Figure 12, in which a scopal $\mathcal{F} : (a \to b) \to c$ is stowed away until an appropriate scope target is derived (cf. Cooper 1983, Moortgat 1997, Carpenter 1998). This 'rule' is a placeholder for an official account of quantifier scope, but in the meantime, it allows us to get a better sense for how binding operates.

$$\cfrac{\cfrac{\Delta : b}{\mathcal{F}\,(\lambda_x\, \Delta) : c}\Uparrow}{\begin{array}{c}\vdots \\ \hline x : a \\ \hline \mathcal{F} : (a \to b) \to c \;\Downarrow\end{array}}$$

Figure 12: A descriptive 'rule' for quantifier scope.

Consider the derivation of *everyone$_i$'s mom likes them$_i$* in Figure 13. The quantifier shifts via the $\triangleright$ rule, and is then stowed away. The η and $\circledast$ operations, together with the quantifier scope 'rule', allow us to construct the scope argument $\lambda_x \lambda_y\, \mathsf{likes}\, y\, (\mathsf{mom}\, x) : \mathtt{e} \to \mathtt{t}^{\mathtt{e}}$, which is fed to the retrieved, bind-shifted quantifier. The result is equivalent (via a couple of β-conversions) to $\mathsf{eo}\,(\lambda_x\, \mathsf{likes}\, x\, (\mathsf{mom}\, x))$.

Along with liberating binding from coargument-hood, the scope-oriented $\triangleright$ rule does not need to be generalized. Recall that generalized Z (Figure 8) was required in order to allow an arbitrary number of arguments to intervene between binder and bind-ee, as required by (17). There isn't any need for a comparable generalization here. Scope is necessary for binding; the amount of intervening material is irrelevant.

Let's take stock. Taking some cues from the standard account, we decomposed G into η and $\circledast$, and replaced Z with $\triangleright$. The payoffs of $\triangleright$ are immediate. Yoking binding to

14 The rule is identical to Shan & Barker's (2006) rule for binding, and like it draws on Szabolcsi's (1989, 1992) use of the duplicator combinator $\mathsf{W} := \lambda_\kappa \lambda_x\, \kappa\, x\, x$. Unlike Jacobson's variable-free semantics, Shan & Barker adopt a scopal treatment of *pronouns* as well, which requires a pronoun to take scope immediately under its binder (see also Dowty 2007). This may be problematic since binding is not subject to locality restrictions.

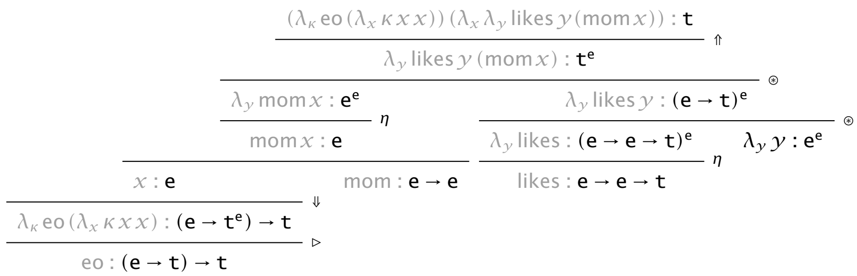

Figure 13: Binding via scope and ▷: *everyone$_i$'s mom likes them$_i$*.

scope remedies an important under-generation issue with Z, while obviating any need for a generalized binding rule. As detailed in the next two sections, the payoffs of η and $\circledast$ are subtler, but significant all the same: these operations instantiate a general, widely used abstraction in computer science, with properties particularly well suited to the compositional behavior of pronouns and their interactions with other facets of meaning.

Of course, we should be sure we haven't lost anything in the bargain. We haven't. For one, η and $\circledast$ together *entail* G, and ▷ generates strictly more binding configurations than Z (even so, *i*-within-*i* is ruled out, for basically the same reason as before: relational nouns lack subjects, and so there is nothing for ▷ to target). Examples that may raise concerns are pied-piping with binding (cf. Figure 6) and functional readings (cf. Figure 7), since the toy 'rule' for scope does not seem to allow gaps and ▷ to interact in the right ways. In reality there are no problems here. See Appendix A for a brief discussion.

6.3 On applicatives

Abstracting out η and $\circledast$ puts us in the presence of something known to computer scientists and functional programmers as an APPLICATIVE (FUNCTOR) (McBride & Paterson 2008). The essence of an applicative functor is an enriched type-space with η and $\circledast$ operations that support some correspondingly enriched notion of functional application.

More formally, an applicative functor is a type constructor F associated with mappings $\eta : a \to F\,a$ and $\circledast : F\,(a \to b) \to F\,a \to F\,b$.[15] η and $\circledast$ must satisfy several laws which guarantee that $\circledast$ embodies an enriched notion of functional application, and that η does nothing more than trivially inject values into the enriched type-space characterized by F. We pass over these here, but see McBride & Paterson 2008, Charlow 2019.

Applicative functors can be factored out of a great deal of existing semantic theory. I'll mention just two examples. Alternative semantics can be built from an applicative functor for sets, such that $\mathsf{S}\,a ::= a \to \mathtt{t}$, and with η and $\circledast$ as defined as follows:

(24) $\eta\,x := \{x\}$ $\qquad\qquad \eta : a \to \mathsf{S}\,a$

(25) $m \circledast n := \{f\,x \mid f \in m, x \in n\}$ $\qquad\qquad \circledast : \mathsf{S}\,(a \to b) \to \mathsf{S}\,a \to \mathsf{S}\,b$

15 Type constructors are mappings from types to types. You can think of them as ways of abbreviating complex types. For example, our superscripted-e notation is a type constructor: if a is a type, then so is $a^{\mathtt{e}}$.

Likewise, continuations-based theories of scope (Shan & Barker 2006, Barker & Shan 2014) are built on two combinators (Lift and Scope) that directly instantiate the applicative functor for continuations, such that $\mathsf{C}_r\, a ::= (a \to r) \to r$, and with η and $\circledast$ as follows:[16]

(26) $\eta\, x \coloneqq \lambda_\kappa\, \kappa\, x$ $\qquad\qquad \eta : a \to \mathsf{C}_r\, a$

(27) $m \circledast n \coloneqq \lambda_\kappa\, m\, (\lambda_f\, n\, (\lambda_x\, \kappa\, (f\, x)))$ $\qquad\qquad \circledast : \mathsf{C}_r\, (a \to b) \to \mathsf{C}_r\, a \to \mathsf{C}_r\, b$

Applicative functors enjoy an important property: they're closed under composition (e.g., McBride & Paterson 2008, Kiselyov 2015): if F and G are applicative, then $F \circ G$ is too. The recipe for assembling a composite $\circledast$ operation for $F \circ G$ is given in Figure 14 (to assemble a composite η, just apply G and F's η's in succession).[17]

$$\cfrac{\cfrac{\cfrac{\circledast_G : G\,(a \to b) \to G\,a \to G\,b}{F\,(G\,(a \to b) \to G\,a \to G\,b)}\,\eta_F \qquad F\,(G\,(a \to b))}{F\,(G\,a \to G\,b)}\,\circledast_F \qquad F\,(G\,a)}{F\,(G\,b)}\,\circledast_F$$

Figure 14: Composing two applicative functors F and G.

For example, using 'E' to stand in for the variable-free semanticist's type constructor (i.e., $\mathsf{E}\, a ::= a^{\mathsf{e}}$), $\mathsf{E} \circ \mathsf{S}$ yields environment-sensitive alternative sets, with $\eta\, x = \lambda_y\, \{x\}$ and $m \circledast n = \lambda_y\, \{f\, x \mid f \in m\, y, x \in n\, y\}$ (Kratzer & Shimoyama 2002, Charlow 2014, 2018). Conversely, $\mathsf{S} \circ \mathsf{E}$ gives alternative environment-sensitive meanings, with $\eta\, x = \{\lambda_y\, x\}$ and $m \circledast n = \{\lambda_y\, f\, y\, (x\, y) \mid f \in m, x \in n\}$ (Poesio 1996, Romero & Novel 2013). And composing E with *itself* — taking $\mathsf{E} \circ \mathsf{E}$ — yields an applicative for double-sensitivity $\eta\, x = \lambda_y\, \lambda_z\, x$ and $m \circledast n = \lambda_y\, \lambda_z\, m\, y\, z\, (n\, y\, z)$.

The 'compositionality' of applicative functors is significant in part because it guarantees that applicative pieces of grammar can be theorized about separately and modularly. Any set of potentially disparate analyses relying on applicatives automatically generates a 'composite' analysis in terms of composed applicatives. For example, the existence of applicative E and S within a grammar immediately generates the compositional resources needed to handle both environment-sensitive alternative sets ($\mathsf{E} \circ \mathsf{S}$), and sets of alternative environment-sensitive meanings ($\mathsf{S} \circ \mathsf{E}$) (cf. Section 5.3).

Moreover, composing the variable-free applicative functor with *itself* ($\mathsf{E} \circ \mathsf{E}$) provides an easy solution to the problem of multiple pronouns (cf. Section 5.1). This composite applicative functor embodies sensitivity to two type-e environments. A derivation of *he saw her* is provided in Figure 15. Here, I use '$\circledast\circledast$' for the composite $\circledast$ operation, and '$\circledast\eta$' as shorthand for applying η 'under the e' (i.e., in precisely the way G had allowed).[18]

16 In fact, linguistic uses of continuations rely on *indexed* or *parametrized* applicatives, a somewhat more general notion. See Wadler 1994 and Kobele 2018, as well as Appendix A, for discussions of related constructs.

17 It may appear from Figure 14 that we require a *nullary* instance of $\circledast_G$ (i.e., we allow our rules to operate on *themselves*). See Barker & Shan (2014: 118) and White et al. (2017) for alternative ways to get the same result.

18 Intriguingly, the continuations-based analysis of inverse scope (e.g., Shan & Barker 2006) involves composing C_r with itself, in a way that is remarkably parallel to the strategy for dealing with multiple pronouns.

$$\dfrac{\dfrac{\lambda_x x : \mathsf{e}^{\mathsf{e}}}{\lambda_x \lambda_y x : (\mathsf{e}^{\mathsf{e}})^{\mathsf{e}}}\,{\circledast\eta} \qquad \dfrac{\dfrac{\dfrac{\mathsf{saw} : \mathsf{e} \to \mathsf{e} \to \mathsf{t}}{\lambda_y \mathsf{saw} : (\mathsf{e} \to \mathsf{e} \to \mathsf{t})^{\mathsf{e}}}\,\eta \qquad \lambda_y y : \mathsf{e}^{\mathsf{e}}}{\lambda_y \mathsf{saw}\, y : (\mathsf{e} \to \mathsf{t})^{\mathsf{e}}}\,\circledast}{\lambda_x \lambda_y \mathsf{saw}\, y : ((\mathsf{e} \to \mathsf{t})^{\mathsf{e}})^{\mathsf{e}}}\,\eta}{\lambda_x \lambda_y \mathsf{saw}\, y\, x : (\mathsf{t}^{\mathsf{e}})^{\mathsf{e}}}\,{\circledast\circledast}$$

Figure 15: Deriving *he saw her* by composing the variable-free applicative functor with itself.

Consider now the familiar Curry-Uncurry isormorphisms in Figure 16: Curry converts a function from tuples into a higher-order function, and Uncurry turns a higher-order function back into a function on tuples. Uncurrying the two-place function derived by the variable-free theory for *he saw her* in Figure 15 gives $\lambda_{(x,y)}\, \mathsf{saw}\, y\, x = \lambda_p\, \mathsf{saw}\, p_2\, p_1$. Something very much like dependence on an assignment (dependence on a sequence) springs organically into being, albeit with a complete absence of object-language variables. In other words, not only can variable-free and variable-full approaches be treated with an identical set of applicative combinatory tools, and cognate rules for binding — the semantic values thereby generated turn out to be equivalent up to isomorphism.

$$\dfrac{f : c^{(a\times b)}}{\lambda_x \lambda_y f\,(x, y) : (c^b)^a}\,\mathsf{Curry} \qquad\qquad \dfrac{f : (c^b)^a}{\lambda_{(x,y)} f\, x\, y : c^{(a\times b)}}\,\mathsf{Uncurry}$$

Figure 16: The Curry-Uncurry isomorphisms.

Again, this does not imply any sort of ultimate equivalence between variable-free and -full theories. Having object-language variables allows the variable-full theorist to impose representational constraints that the variable-free theorist cannot even contemplate.

7 Conclusion

Our time's up. I hope you've gotten a glimpse of the elegance and viability of variable-free theorizing, along with a sense of which of its differences from standard theories are negotiable, and which aren't. For example, both sorts of theories can readily exploit flexible compositional architectures (though variable-full theories typically do not), and both sorts of theories can have a rule for binding oriented around scope (though Jacobson does not). The status of object-language variables, by contrast, is definitional and, therefore, non-negotiable. Nevertheless, the two kinds of theories can be stated in ways that make them seem much more alike than we might have initially supposed. This suggests in turn that one may not need to be a variable-free semanticist in order to reap the empirical insights of variable-free semantics (see Charlow 2019).

Where do we go from here? One area ripe for investigation is variable-free *dynamic* semantics, in order to explain data like (22) and (23). While there are existing 'variable-free' (or nearly so) dynamic systems (e.g., Dekker 1994, van Eijck 2001, Bittner 2001, de Groote 2006, Murray 2014, Charlow 2014), these theories are generally quite different in their aims and form from Jacobson's static variable-free system, and cannot be properly regarded as dynamic extensions of it. Preliminary work has been initiated by Shan (2001) and Szabolcsi (2003), with Shan's decomposition of canonical dynamic systems into 'adjoint' functors plus nondeterminism especially close to the outlook of Section 6.

A Scope without variables

This appendix presents a pared-down version of Shan & Barker's (2006) variable-free account of scope based on the continuations (indexed) applicative functor, shows how composing this applicative functor with the variable-free applicative of Section 6 allows scope to feed binding, and gives hints about how to derive pied-piping with binding (cf. Figure 6) and functional readings (cf. Figure 7).

Continuations-based theories of scope are built on two combinators (Lift and Scope) that instantiate the *indexed* applicative functor for continuations, with $\mathsf{C}_i^o\, a ::= (a \to i) \to o$, and with η and $\circledast$ as follows (note that the λ-terms are unchanged from (26) and (27)):

(28) $\eta\, x := \lambda_\kappa\, \kappa\, x$ $\qquad \eta : a \to \mathsf{C}_o^o\, a$

(29) $m \circledast n := \lambda_\kappa\, m\, (\lambda_f\, n\, (\lambda_x\, \kappa\, (f\, x)))$ $\qquad \circledast : \mathsf{C}_i^o\, (a \to b) \to \mathsf{C}_j^i\, a \to \mathsf{C}_j^o\, b$

Indexed applicatives have more general types than garden-variety applicatives, but they still obey the four applicative laws and compose freely with themselves and with other applicatives, following the general recipe in Figure 14. For example, the indexed continuations applicative may be composed with the variable-free applicative:

(30) $\eta\, x := \lambda_\kappa\, \kappa\, (\lambda_y\, x)$ $\qquad \eta : a \to \mathsf{C}_o^o\, a^{\mathsf{e}}$

(31) $m \circledast n := \lambda_\kappa\, m\, (\lambda_f\, n\, (\lambda_x\, \kappa\, (\lambda_y\, f\, y\, (x\, y))))$ $\qquad \circledast : \mathsf{C}_i^o\, (a \to b)^{\mathsf{e}} \to \mathsf{C}_j^i\, a^{\mathsf{e}} \to \mathsf{C}_j^o\, b^{\mathsf{e}}$

Our two applicatives allow us to give a binder scope over its bind-ee, as in Figure 17. Similar to Figure 15, '$\circledast\eta$' means we use the continuations $\circledast$ to apply the variable-free η under the C (and '$\circledast\circledast$' is the composite $\circledast$). As in other analyses based on continuations, a Lower rule ends a derivation by applying the resulting term to a trivial continuation, here an identity function of type $\mathtt{t}^{\mathsf{e}} \to \mathtt{t}^{\mathsf{e}}$. The result after Lower is equivalent via a β-reduction to $\mathsf{eo}\, (\lambda_x\, \mathsf{likes}\, x\, (\mathsf{mom}\, x))$, as desired.[19]

Using these applicatives, we can derive pied-piping with binding (cf. Figure 6) and functional readings (cf. Figure 7). Details are omitted, but breadcrumbs are provided. There are two kinds of derivations of pied-piping with binding; one identifies the gap and pronoun directly via the variable-free $\circledast$, and another is closer in form to Figure 17, but juggles the gap's anaphoric dependency as well.[20] Derivations of functional readings will likewise be similar in form to Figure 17, but additionally require us to project up the anaphoric dependency on an e^{e} function using (what else?) another applicative layer.

---

19 The form of Lower we rely on is actually inconsistent with Shan & Barker's (2006) account of Weak Crossover! If we wish to help ourselves to their account, we can follow them in only allowing Lower to apply to types of the form $\mathsf{C}_{\mathtt{t}}^{a}\, \mathtt{t}$. This will in turn necessitate an auxiliary mapping $\lambda_m\, \lambda_\kappa\, \lambda_x\, m\, x\, \kappa : (\mathsf{C}_i^o\, a)^{\mathsf{e}} \to \mathsf{C}_i^{o^{\mathsf{e}}}\, a$. Applied to constituents containing an unbound pronoun, this mapping allows binder and bind-ee to meet via C's $\circledast$ operation, rather than via Lower. See Shan & Barker 2006 for in-depth discussion of related matters.

20 The latter of these is probably to be preferred since, as per the considerations in fn. 5, gaps and pronouns should be analyzed as having different types (and so cannot be directly identified by $\circledast$).

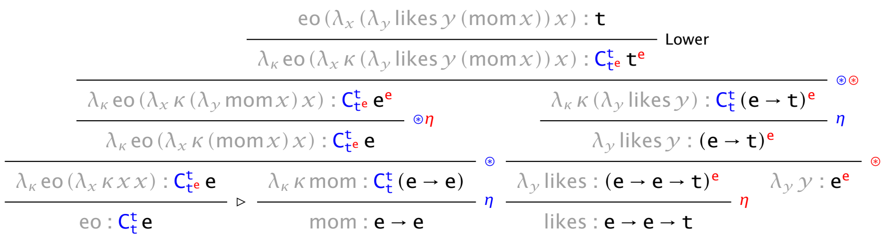

Figure 17: Deriving *everyone$_i$'s mom likes them$_i$* using two applicatives.

References

Ades, Anthony E. & Mark J. Steedman. 1982. On the order of words. *Linguistics and Philosophy* 4(4). 517–558. http://www.jstor.org/stable/25001073.

Barker, Chris. 2005. Remark on Jacobson 1999: Crossover as a local constraint. *Linguistics and Philosophy* 28(4). 447–472. https://doi.org/10.1007/s10988-004-5327-1.

Barker, Chris. 2012. Quantificational binding does not require c-command. *Linguistic Inquiry* 43(4). 614–633. https://doi.org/doi:10.1162/ling_a_00108.

Barker, Chris. 2018. Evaluation order, crossover, and reconstruction. In Manfred Krifka & Mathias Schenner (eds.), *Reconstruction effects in relative clauses* (Studia Grammatica 75), 357–386. Berlin, Boston: De Gruyter. https://doi.org/10.1515/9783050095158-011.

Barker, Chris & Pauline Jacobson. 2007. Introduction: Direct Compositionality. In Chris Barker & Pauline Jacobson (eds.), *Direct compositionality*, chap. 1, 1–19. Oxford: Oxford University Press.

Barker, Chris & Chung-chieh Shan. 2008. Donkey anaphora is in-scope binding. *Semantics and Pragmatics* 1(1). 1–46. https://doi.org/10.3765/sp.1.1.

Barker, Chris & Chung-chieh Shan. 2014. *Continuations and natural language*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199575015.001.0001.

Bernardi, Raffaella & Anna Szabolcsi. 2008. Optionality, scope, and licensing: An application of partially ordered categories. *Journal of Logic, Language and Information* 17(3). 237–283. https://doi.org/10.1007/s10849-008-9060-y.

Bittner, Maria. 2001. Surface composition as bridging. *Journal of Semantics* 18(2). 127–177. https://doi.org/10.1093/jos/18.2.127.

Cable, Seth. 2010. *The grammar of Q: Q-particles, wh-movement, and pied piping*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780195392265.001.0001.

Carpenter, Bob. 1998. *Type-logical semantics*. Cambridge, MA: MIT Press.

Champollion, Lucas. 2015. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy* 38(1). 31–66. https://doi.org/10.1007/s10988-014-9162-8.

Charlow, Simon. 2008. Free and bound pro-verbs: A unified treatment of anaphora. In Tova Friedman & Satoshi Ito (eds.), *Proceedings of Semantics and Linguistic Theory 18*, 194–211. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v18i0.2487.

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

Charlow, Simon. 2018. The scope of alternatives: Indefiniteness and islands. To appear in *Linguistics and Philosophy*. http://ling.auf.net/lingbuzz/003302.

Charlow, Simon. 2019. A modular theory of pronouns and binding. Unpublished ms., Rutgers University. http://ling.auf.net/lingbuzz/003720.

Cooper, Robin. 1979. The interpretation of pronouns. In Frank Heny & Helmut S. Schnelle (eds.), *Syntax and semantics, volume 10: Selections from the Third Groningen Round Table*, 61–92. New York: Academic Press.

Cooper, Robin. 1983. *Quantification and syntactic theory*. Dordrecht: Springer Science+Business Media. https://doi.org/10.1007/978-94-015-6932-3.

Cresti, Diana. 1995. Extraction and reconstruction. *Natural Language Semantics* 3(1). 79–122. https://doi.org/10.1007/BF01252885.

Dayal, Veneeta. 2016. *Questions*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199281268.001.0001.

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

Dowty, David. 1992. 'Variable-free' syntax, variable-binding syntax, the natural deduction Lambek calculus, and the crossover constraint. In Jonathan Mead (ed.), *Proceedings of the 11th West Coast Conference on Formal Linguistics*, 161–176. Stanford: CSLI Publications.

Dowty, David. 2007. Compositionality as an empirical problem. In Chris Barker & Pauline Jacobson (eds.), *Direct compositionality* (Oxford Studies in Theoretical Linguistics 14), chap. 2, 23–101. Oxford: Oxford University Press.

van Eijck, Jan. 2001. Incremental dynamics. *Journal of Logic, Language and Information* 10(3). 319–351. https://doi.org/10.1023/A:1011251627260.

Engdahl, Elisabet. 1980. *The syntax and semantics of questions in Swedish*. University of Massachusetts, Amherst Ph.D. thesis. https://scholarworks.umass.edu/dissertations_1/936/.

Engdahl, Elisabet. 1986. *Constituent questions*. Vol. 27 (Studies in Linguistics and Philosophy). Dordrecht: Reidel. https://doi.org/10.1007/978-94-009-5323-9.

Evans, Frederic. 1988. Binding into anaphoric verb phrases. In Joyce Powers & Kenneth de Jong (eds.), *Proceedings of the 5th Annual Eastern States Conference on Linguistics*, 122–129.

von Fintel, Kai & Irene Heim. 2011. Intensional semantics, Spring 2011 edition.

Groenendijk, Jeroen & Martin Stokhof. 1983. Interrogative quantifiers and Skolem functions. In Konrad Ehlich & Henk C. van Riemsdijk (eds.), *Connectedness in sentence, discourse and text: Proceedings of the Tilburg conference held on 25 and 26 January 1982* (Tilburg studies in language and literature 4), 71–110. Tilburg: Tilburg University, Department of Language & Literature.

de Groote, Philippe. 2006. Towards a Montagovian account of dynamics. In Masayuki Gibson & Jonathan Howell (eds.), *Proceedings of Semantics and Linguistic Theory 16*, 1–16. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v16i0.2952.

Hackl, Martin, Jorie Koster-Hale & Jason Varvoutis. 2012. Quantification and ACD: Evidence from real-time sentence processing. *Journal of Semantics* 29(2). 145–206. https://doi.org/10.1093/jos/ffr009.

Hamblin, C. L. 1973. Questions in Montague English. *Foundations of Language* 10(1). 41–53.

Harris, Daniel W. 2019. Semantics without semantic content. To appear in *Mind & Language*.

Heck, Fabian. 2008. *On pied-piping: Wh-movement and beyond*. Berlin, Boston: Mouton de Gruyter. https://doi.org/10.1515/9783110211467.

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

Heim, Irene. 1997. Predicates or formulas? Evidence from ellipsis. In Aaron Lawson (ed.), *Proceedings of Semantics and Linguistic Theory 7*, 197–221. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v7i0.2793.

Heim, Irene & Angelika Kratzer. 1998. *Semantics in generative grammar*. Oxford: Blackwell.

Hendriks, Herman. 1993. *Studied flexibility: Categories and types in syntax and semantics*. University of Amsterdam Ph.D. thesis.

Hepple, Mark. 1990. *The grammar and processing of order and dependency: a categorial approach*. University of Edinburgh Ph.D. thesis.

Jacobson, Pauline. 1992. Antecedent contained deletion in a variable-free semantics. In Chris Barker & David Dowty (eds.), *Proceedings of Semantics and Linguistic Theory 2* (OSU Working Papers in Linguistics 40), 193–213. https://doi.org/10.3765/salt.v2i0.3027.

Jacobson, Pauline. 1994a. Binding connectivity in copular sentences. In Mandy Harvey & Lynn Santelmann (eds.), *Proceedings of Semantics and Linguistic Theory 4*, 161–178. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v4i0.2456.

Jacobson, Pauline. 1994b. I-within-i effects in a variable-free semantics and a categorial syntax. In Paul Dekker & Martin Stokhof (eds.), *Proceedings of the Ninth Amsterdam Colloquium*, 349–369. University of Amsterdam.

Jacobson, Pauline. 1996. The locality of interpretation: The case of binding and coordination. In Teresa Galloway & Justin Spence (eds.), *Proceedings of Semantics and Linguistic Theory 6*, 111–135. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v6i0.2771.

Jacobson, Pauline. 1998. Antecedent contained deletion and pied-piping: Evidence for a variable-free semantics. In Devon Strolovitch & Aaron Lawson (eds.), *Proceedings of Semantics and Linguistic Theory 8*, 74–91. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v8i0.2800.

Jacobson, Pauline. 1999. Towards a variable-free semantics. *Linguistics and Philosophy* 22(2). 117–184. https://doi.org/10.1023/A:1005464228727.

Jacobson, Pauline. 2000a. Paycheck pronouns, Bach-Peters sentences, and variable-free semantics. *Natural Language Semantics* 8(2). 77–155. https://doi.org/10.1023/A:1026517717879.

Jacobson, Pauline. 2000b. Paychecks, stress, and variable-free semantics. In Brendan Jackson & Tanya Matthews (eds.), *Proceedings of Semantics and Linguistic Theory 10*, 65–82. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v10i0.3103.

Jacobson, Pauline. 2004. Kennedy's puzzle: What I'm named or who I am? In Kazuha Watanabe & Robert B. Young (eds.), *Proceedings of Semantics and Linguistic Theory 14*, 145–162. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v14i0.2911.

Jacobson, Pauline. 2009. Do representations matter or do meanings matter: The case of antecedent containment. In Erhard Hinrichs & John Nerbonne (eds.), *Theory and evidence in semantics*, 81–107. Stanford: CSLI Publications.

Jacobson, Pauline. 2014. *Compositional semantics: An introduction to the syntax/semantics interface*. Oxford: Oxford University Press.

Jacobson, Pauline. 2016. The short answer: Implications for direct compositionality (and vice versa). *Language* 92(2). 331–375. https://doi.org/10.1353/lan.2016.0038.

Jäger, Gerhard. 2005. *Anaphora and type logical grammar*. Dordrecht: Springer. https://doi.org/10.1007/1-4020-3905-0.

Kamp, Hans. 1981. A theory of truth and semantic interpretation. In Jeroen Groenendijk, Theo M. V. Janssen & Martin Stokhof (eds.), *Formal Methods in the Study of Language*, 277–322. Mathematical Centre Amsterdam.

Karttunen, Lauri. 1969. Pronouns and variables. In Robert I. Binnick, Alice Davison, Georgia M. Green & Jerry L. Morgan (eds.), *Papers from the Fifth Regional Meeting, Chicago Linguistic Society*, 108–116. Chicago: Department of Linguistics, University of Chicago.

Keenan, Edward L. 1971. Names, quantifiers, and the sloppy identity problem. *Research on Language & Social Interaction* 4(2). 211–232. https://doi.org/10.1080/08351817109370257.

Kiselyov, Oleg. 2015. Applicative abstract categorial grammars. In Makoto Kanazawa, Lawrence S. Moss & Valeria de Paiva (eds.), *NLCS'15. Third workshop on natural language and computer science*, vol. 32 (EPiC Series), 29–38.

Kobele, Gregory M. 2018. The Cooper storage idiom. *Journal of Logic, Language and Information* 27(2). 95–131. https://doi.org/10.1007/s10849-017-9263-1.

Kratzer, Angelika. 1996. Severing the external argument from its verb. In Johan Rooryck & Laurie Zaring (eds.), *Phrase structure and the lexicon*, 109–137. Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-015-8617-7_5.

Kratzer, Angelika & Junko Shimoyama. 2002. Indeterminate pronouns: The view from Japanese. In Yukio Otsu (ed.), *Proceedings of the Third Tokyo Conference on Psycholinguistics*, 1–25. Tokyo: Hituzi Syobo.

Kuhn, Jeremy. 2016. ASL loci: Variables or features? *Journal of Semantics* 33(3). 449–491. https://doi.org/10.1093/jos/ffv005.

Lambek, Joachim. 1958. The mathematics of sentence structure. *The American Mathematical Monthly* 65(3). 154–170. https://doi.org/10.1080/00029890.1958.11989160.

Lewis, Davis. 1980. Index, context, and content. In Stig Kanger & Sven Öhman (eds.), *Philosophy and grammar: Papers on the occasion of the quincentennial of Uppsala University*, 79–100. Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-009-9012-8_6.

McBride, Conor & Ross Paterson. 2008. Applicative programming with effects. *Journal of Functional Programming* 18(1). 1–13. https://doi.org/10.1017/S0956796807006326.

Montague, Richard. 1970. Universal grammar. *Theoria* 36(3). 373–398. https://doi.org/10.1111/j.1755-2567.1970.tb00434.x.

Moortgat, Michael. 1997. Categorial type logics. In Johan van Benthem & Alice ter Meulen (eds.), *Handbook of logic and language*, 93–177. Amsterdam: Elsevier. https://doi.org/10.1016/B978-044481714-3/50005-9.

Murray, Sarah E. 2014. Varieties of update. *Semantics and Pragmatics* 7(2). 1–53. https://doi.org/10.3765/sp.7.2.

Muskens, Reinhard. 1995. Tense and the logic of change. In Urs Egli, Peter E. Pause, Christoph Schwarze, Arnim von Stechow & Götz Wienold (eds.), *Lexical Knowledge in the Organization of Language*, 147–183. Amsterdam: John Benjamins. https://doi.org/10.1075/cilt.114.08mus.

Nishigauchi, Taisuke. 1990. *Quantification in the theory of grammar*. Dordrecht: Kluwer Academic Publishers. https://doi.org/10.1007/978-94-009-1972-3.

Parsons, Terence. 1990. *Events in the semantics of English*. Cambridge, MA: MIT Press.

Partee, Barbara H. 1986. Noun phrase interpretation and type-shifting principles. In Jeroen Groenendijk, Dick de Jongh & Martin Stokhof (eds.), *Studies in Discourse Representation Theory and the Theory of Generalized Quantifiers*, 115–143. Dordrecht: Foris.

Pierce, Benjamin C. 2002. *Types and programming languages*. Cambridge, MA: MIT Press.

Poesio, Massimo. 1996. Semantic ambiguity and perceived ambiguity. In Kees van Deemter & Stanley Peters (eds.), *Semantic Ambiguity and Underspecification* (CSLI Lecture Notes 55), 159–201. Stanford: CSLI Publications.

Reinhart, Tanya. 1983. *Anaphora and semantic interpretation*. Chicago: University of Chicago Press.

Romero, Maribel & Marc Novel. 2013. Variable binding and sets of alternatives. In Anamaria Fălăuș (ed.), *Alternatives in Semantics*, chap. 7, 174–208. London: Palgrave Macmillan UK. https://doi.org/10.1057/9781137317247_7.

Rooth, Mats. 1985. *Association with focus*. University of Massachusetts, Amherst Ph.D. thesis.

Rooth, Mats. 1992a. A theory of focus interpretation. *Natural Language Semantics* 1(1). 75–116. https://doi.org/10.1007/BF02342617.

Rooth, Mats. 1992b. Ellipsis redundancy and reduction redundancy. In Stephen Berman & Arild Hestvik (eds.), *Proceedings of the Stuttgart Workshop on Ellipsis,* no. 29 in Arbeitspapiere des SFB 340. Stuttgart: University of Stuttgart.

Rooth, Mats. 1996. Focus. In Shalom Lappin (ed.), *The Handbook of Contemporary Semantic Theory*, 271–298. Oxford: Blackwell.

Safir, Ken. 2004. *The syntax of (in)dependence*. Cambridge, MA: MIT Press. https://doi.org/10.7551/mitpress/6595.001.0001.

Sag, Ivan A. 1976. *Deletion and logical form*. Massachusetts Institute of Technology Ph.D. thesis. https://hdl.handle.net/1721.1/16401.

Schlenker, Philippe. 2011. Donkey anaphora: the view from sign language (ASL and LSF). *Linguistics and Philosophy* 34(4). 341–395. https://doi.org/10.1007/s10988-011-9098-1.

Shan, Chung-chieh. 2001. A variable-free dynamic semantics. In Robert van Rooy & Martin Stokhof (eds.), *Proceedings of the Thirteenth Amsterdam Colloquium*. University of Amsterdam.

Shan, Chung-chieh. 2004. Binding alongside Hamblin alternatives calls for variable-free semantics. In Kazuha Watanabe & Robert B. Young (eds.), *Proceedings of Semantics and Linguistic Theory 14*, 289–304. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v14i0.2901.

Shan, Chung-chieh & Chris Barker. 2006. Explaining crossover and superiority as left-to-right evaluation. *Linguistics and Philosophy* 29(1). 91–134. https://doi.org/10.1007/s10988-005-6580-7.

Sharvit, Yael. 1997. *The syntax and semantics of functional relative clauses*. Rutgers University Ph.D. thesis.

Sharvit, Yael. 1999. Functional relative clauses. *Linguistics and Philosophy* 22(5). 447–478. https://doi.org/10.1023/A:1005411720444.

von Stechow, Arnim. 1996. Against LF pied-piping. *Natural Language Semantics* 4(1). 57–110. https://doi.org/10.1007/BF00263537.

Steedman, Mark. 1987. Combinatory grammars and parasitic gaps. *Natural Language & Linguistic Theory* 5(3). 403–439. https://doi.org/10.1007/BF00134555.

Steedman, Mark. 2000. *The syntactic process*. Cambridge, MA: MIT Press.

Sternefeld, Wolfgang. 1998. *The semantics of reconstruction and connectivity*. Arbeitspapier 97, SFB 340. Universität Tübingen & Universität Stuttgart, Germany.

Sternefeld, Wolfgang. 2001. Semantic vs. syntactic reconstruction. In Christian Rohrer, Antje Roßdeutscher & Hans Kamp (eds.), *Linguistic Form and its Computation*, chap. 5, 145–182. Stanford: CSLI Publications.

Szabolcsi, Anna. 1983. ECP in categorial grammar. Unpublished ms., Max Planck Institute, Nijmegen.

Szabolcsi, Anna. 1989. Bound variables in syntax (are there any?) In Renate Bartsch, Johan van Benthem & Peter van Emde Boas (eds.), *Semantics and contextual expressions*, 295–318. Dordrecht: Foris.

Szabolcsi, Anna. 1992. Combinatory grammar and projection from the lexicon. In Ivan A. Sag & Anna Szabolcsi (eds.), *Lexical matters*, 241–268. Stanford: CSLI Publications.

Szabolcsi, Anna. 2003. Binding on the fly: Cross-sentential anaphora in variable-free semantics. In Geert-Jan M. Kruijff & Richard T. Oehrle (eds.), *Resource-sensitivity, binding and anaphora*, chap. 8, 215–227. Dordrecht: Kluwer Academic Publishers. https://doi.org/10.1007/978-94-010-0037-6_8.

Szabolcsi, Anna. 2011. Scope and binding. In Klaus von Heusinger, Claudia Maienborn & Paul Portner (eds.), *Semantics: An international handbook of natural language meaning*, vol. 2 (HSK 33), chap. 62, 1605–1641. Berlin: de Gruyter. https://doi.org/10.1515/9783110255072.1605.

Szabolcsi, Anna. 2013. Quantification and ACD: What is the evidence from real-time processing evidence for? A response to Hackl *et al.* (2012). *Journal of Semantics* 31(1). 135–145. https://doi.org/10.1093/jos/ffs025.

Takahashi, Shoichi & Danny Fox. 2005. MaxElide and the re-binding problem. In Effi Georgala & Jonathan Howell (eds.), *Proceedings of Semantics and Linguistic Theory 15*, 223–240. Ithaca, NY: Cornell University. https://doi.org/10.3765/salt.v15i0.3095.

Wadler, Philip. 1994. Monads and composable continuations. *LISP and Symbolic Computation* 7(1). 39–56. https://doi.org/10.1007/BF01019944.

White, Michael, Simon Charlow, Jordan Needle & Dylan Bumford. 2017. Parsing with dynamic continuized CCG. In *Proceedings of the 13th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+13)*, 71–83. Umeå, Sweden: Association for Computational Linguistics. http://aclweb.org/anthology/W17-6208.

Winter, Yoad. 2004. Functional quantification. *Research on Language and Computation* 2(3). 331–363. https://doi.org/10.1007/s11168-004-0909-1.