Intensionality in Non-Constituent Conjunction

Richard T. Oehrle[1], Pacific Grove, CA 93950, rto@redshift.com

voor Frans Zwarts

1. Goals

The first goal of this paper is to document the existence of cases of non-constituent coordination in which the expected form of semantic distributivity across the coordinating connective fails. This failure challenges some widely accepted accounts of non-constituent coordination. The second goal of this paper is to consider how theories of non-constituent coordination in particular, and theories involving logical operators, in general, can be adjusted to accommodate such cases. The analysis of Partee & Rooth (1983) and Rooth & Partee (1982) paves the way.

2. Empirical Observations

Non-constituent coordination is an awkward name for an elegant structure in which two or more sequences of constituents of the same kind are coordinated. When the constituents are arguments of an intensional predicate, interesting things can happen. For example, consider (1) below:

(1) I owe Olivia two dollars or Aaron two euros.

In the interpretation of primary relevance here, this sentence is true of a situation in which I am under an obligation that I can discharge in one of two ways: either by giving Olivia two dollars or by giving Aaron two euros. For no good reason that I can think of, I will call this the 'primary reading'. There is an interpretation of secondary relevance as well (the 'secondary reading'), one consistent with the continuation 'but I've forgotten which':[2]

---

1 This paper is based in part on a presentation at the Seminar für Sprachwissenschaft at the Eberhard Karls Universität, Tübingen, April 26, 1999. I'm grateful to Uwe Mönnich for making this possible. Some of the research reported here was carried out while I benefited from the support of NSF grant SBR-9510706.

2 Rooth & Partee (1982) call this the 'wide-scope or' reading. We'll come back to their path-breaking work below.

(1') I owe Olivia two dollars or Aaron two euros, but I've forgotten which.

Distributing the verb *owe* across the disjunction in (1) yields (2):

(2) I owe Olivia two dollars or owe Aaron two euros.

This sentence does not describe the situation in which I am under an obligation that I can discharge in one of two ways. Rather, it describes a situation in which I'm either under an obligation that I can discharge by providing Olivia with two dollars or under an obligation that I can discharge by providing Aaron with two euros. Thus, syntactic distributivity at this level is semantically consistent with the secondary reading, but is not semantically consistent with the primary reading. It's worth noting that the failure of semantic distributivity observed with *or* does not seem to hold when we replace disjunctive *or* with conjunctive *and*. Examples (3) and (4) apply to the same range of circumstances (as far as I can tell):

(3) I owe Olivia two dollars and Aaron two euros.
(4) I owe Olivia two dollars and owe Aaron two euros.

In both cases, discharging my obligations requires that I pay Olivia two dollars and pay Aaron two euros. (Whether one of these cases involves a single obligation and the other multiple obligations seems to be moot: the individuation of obligations is not so clear.)

There are other intensional predicates with analogous behavior. For example, consider the dialogue below:

Me: Either I'll bring you back a CD, Olivia, or I'll bring you back a DVD, Aaron.
Kids: Is that a promise.
Me: I promise.

Describing this situation later, I might say (5):

(5) I promised Olivia a CD or Aaron a DVD.

But I can't describe the situation with (6):

(6) I promised Olivia a CD or promised Aaron a DVD.

Similar properties are found with intensional predicates in gapping (Oehrle, 1971, 1987).

(7) Mrs. J. can't live in one city and Mr. J. in another.

The most obvious interpretation of (7) is the one paraphrasable in (8):

(8) It can't be the case that Mrs. J. live in one city and Mr. J. live in another.

In other words, the most obvious interpretation of (7) is not available when the sequence of negated modal and predicate *can't live* is syntactically distributed across the conjunction, as in (9) below.

(9) Mrs. J can't live in one city and Mr. J can't live in another.

Apart from the intrinsic interest of this kind of empirically observable semantic behavior, it is of theoretical interest because it poses a challenge to theories of coordination that attempt to address both syntactic and semantic issues.

3. Some theories of coordination

The earliest generative theories of coordination were purely syntactic. Chomsky (1957) states a transformation that we reformulate below as a Gentzen-style structural rule:

$$\frac{\Gamma, A, \Delta \vdash s \qquad \Gamma, B, \Delta \vdash s}{\Gamma, A \text{ and } B, \Delta \vdash s}$$

In this rule, 'A' and 'B' are variables ranging over categories; 'Γ' and 'Δ' represent the left and right contexts of coordination, respectively. If the context variables are empty, this structural rule provides an analysis of sentential coordination. If the context variables are not empty, this structural rule offers an analysis of sentence-internal constituent coordination. Nothing in the rule prepares us for non-constituent coordination or for semantic problems of the kind we've just been discussing. I won't speculate here on how this rule might be amended to address these two problems.

A more promising attack on coordination has resulted from combining two threads of research---one motivated by semantic considerations involving the semantic polymorphism of coordinating particles (that is, *and* and *or*), the other motivated by the account of syntactic categories in flexible categorial grammar.

The key semantic idea of the first thread (following Gazdar, Keenan, Faltz, Partee, Rooth) is to regard the natural language coordinators represented by English *and* and *or* as being interpreted as lattice-theoretic meets and joins.[3] The set **2** of truth values consisting of points 0 (false) and 1 (true) and the order $0 \leq 1$ is a lattice, with meet as conjunction and join as disjunction. Leverage comes from the fact that if L is a lattice and S is a nonempty set, then the function set L^S consisting of the functions from S to L is a lattice with meet and join defined 'pointwise': if f and g are functions from S to L, then we define the meet $f \wedge g$ of f and g (with respect to the lattice structure of L^S) as that function that maps $s \in S$ to $f(s) \wedge g(s)$ (with respect to the lattice structure of L) for all $s \in S$. Similarly for joins. The λ-calculus offers notational clarity: $f \wedge g = \lambda s.(f(s) \wedge g(s))$ (with s a variable ranging over S, so that this is a function from S to L) and $f \vee g = \lambda s.(f(s) \wedge g(s))$. (Note that the meet and join on the left side of the equality signs are in the lattice structure of the function set L^S , while the meet and join inside the λ-terms are in the lattice structure of L.) If we interpret the syntactic type s (sentence) as semantic type t (truth values in the lattice **2**), and if we regard intransitive verb phrases syntactically as maps from subject noun-phrases to sentences and semantically as maps from the interpretation of subject noun-phrases to the lattice **2**, it follows that we can regard verb phrases semantically as a lattice, defined as just described by pointwise definition. Thus, if we can coordinate verb phrases directly syntactically with *and* and *or*, we can interpret the results as a lattice. Further, if we interpret verb phrases as a lattice, we can interpret functions into verb phrases as a lattice. Etc.

Example: we take *Kim:np:k* to be a noun phrase (with three dimensions: phonology/ orthography *Kim,* syntactic category *np,* and interpretation *k*) and *sings:np\s:sing* and *dances:np\s:dance* to be maps from noun phrases to sentences. We assume we can combine these lexical assumptions to make sentences that we represent as *Kim sings:s:sing(k)* and *Kim dances:s:dance(k)*. And suppose we can conjoin these sentences to form the sentence-level conjunction *Kim sings and Kim dances:s:sing(k) ∧ dance(k)* (with meet in the lattice **2** of truth values). Then we should be able to form the verb phrase *sings and dance:np\s:λx(sing(x) ∧ dance(x))*, using pointwise definition in the semantics. And then we can use these assumptions to provide a direct analysis of *Kim sings and*

3 A lattice is a partially-ordered set in which every pair *(a,b)* of items has a greatest lower bound $a \wedge b$ (called the *meet* of a and b) and a least upper bound $a \vee b$ (called the *join* of a and b). In the text, we frame the issue from the Partee/Rooth perspective, to simplify exposition and to connect with their very relevant immediate concerns.

dances, of category *s* and interpretation *λx(sing(x) ∧ dance(x))(k)*, which reduces by β-reduction to *sing(k)* ∧ *dance(k)*, which is the interpretation associated to the non-reduced sentence-level coordination *Kim sings and Kim dances*.

The key combinatorial idea of the second thread (pioneered by Steedman (1985) and Dowty (1988)) was to realize that flexible categorial grammar allows this general lattice-theoretic insight to be applied elegantly and directly to a wider class of syntactic structures than originally envisioned---structures involving so-called 'non-constituents'. For example, if we take the structure of *Kim called Andrea* and *Sandy texted Andrea* to involve the structural grouping *np · (v · np)*, then it's hard to explain the coordinatizability displayed by *Kim called and Sandy texted---Andrea*. But if we also recognize the possibility of the structural grouping *(np · v) · np*, then the perspective on the form of coordination known as Right Node Raising changes: Right Node Raising is just coordination (accompanied semantically by pointwise definition). Similarly, if we can flexibly associate the structure *v · (np· np)* with a verb phrase like *give Kim a cookie*, and if we can regard the sequence *np· np* as the functor, then we can generate conjunctive sentences like *Kim gave Sandy a quarter and Leslie a dime* directly and assign them an interpretation equivalent to the interpretation of *Kim gave Sandy a quarter and Kim gave Leslie a dime*, simply by invoking the correspondence between syntactic functional structure and its corresponding lattice-theoretic pointwise definition.

But with intensional predicates like *owe*, we have a problem: the syntax works just as for extensional predicates, but the semantic inferences are not always complete, because the approach via syntactic type-raising and pointwise definition requires that the interpretation split across the cases. In a simple (non-coordinating) case, we start with the combination *(A/B:f, B:b)*, reducible to *A:f(b)*; we then lift *B:b* to *(A/B)\A:λF(F(b))*, resulting in the combination (*A/B:f, (A/B)\A:λF(F(b)))*, reducible to *A:(λF(F(b)))(f)*, which normalizes to *A:f(b).* If *A* is interpreted semantically in a lattice, then *A/B* may be interpreted as a lattice (pointwise) and *(A/B)\A* as a lattice (pointwise) as well. Thus we can combine elements of *(A/B)\A* with *and* and *or* and interpret the results as pointwise meets and joins, respectively. If we start with two expressions of type *B---k:B:b* and *j:B:c*---we can raise them and coordinate them, interpreting the coordinations pointwise, to yield:

k and j: (A/B)\A:λF(F(b) ∧ F(c)) ; k or j: (A/B)\A:λF(F(b) ∨ F(c))

If we combine the second, say, with an expression of type *A/B* and interpretation *f*, in the resulting interpretation, β-conversion of (*λF(F(b)* ∨ *F(c)))(f)* yields *f(b)* ∨ *f(c)*, with ∨ outscoping *f*.

The situation does not change if we lift the functor $A/B:f$ to the type $A/((A/B)\backslash A):\lambda\Phi(\Phi(f))$. Combining this with *k or* $j:(A/B)\backslash A:\lambda F(F(b) \vee F(c))$ yields $A:(\lambda\Phi(\Phi(f)))(\lambda F(F(b) \vee F(c)))$ which converts to $(\lambda F(F(b) \vee F(c))(f)$, which normalizes to $f(b) \vee f(c)$, with ∨ outscoping *f*.

When the functor is more complex, such as the double object verb *owe*, the situation is no different. One way to see this is to take *B* as the product of the types assigned to *Olivia* and *two dollars* in *owe Olivia two dollars*. Then *owe* has type *A/B*, for some suitable type *A* and the exposition proceeds exactly as above. Or we can take *B* to be the composition of the lifted types of the types assigned to *Olivia* and *two dollars*, in which case, *A/B* may be replaced by a more complex functor type of the form *(A/C)/D* (with *B* now identifiable as $((A/C)/D)\backslash A$). Pursuing these lines does not bend the properties built into the type-raising/composition/pointwise definition account. In other words, when the verb is *owe*, this does not yield what we called earlier the primary reading. Why?

4. Taking stock

If we wish to pursue the elegant properties of the categorial solution---and of course, we do---the problem that we wish to solve has the following structure (where we describe the component *I owe* as the 'base component' and the component *Olivia a dollar or Aaron a euro* as the 'coordinated component'):

- the base component and coordinated component have compatible syntactic categories, combining to form a sentence *s*;
- the base component and coordinated component have compatible semantic types;
- there are two semantic interpretations to be accounted for.

According to this way of framing the problem, there are in principle a number of different solutions. On the first, there is a single category and a single type assigned to the base component and a single category and a single type assigned to the coordinated component, and the non-deterministic interaction of these assumptions yields both syntactic well-formedness and two interpretations. On the second, one component is assigned more than one syntactic-category / semantic-type pair, yielding both syntactic well-formedness and two interpretations. On the third, both components are assigned multiple syntactic-category / semantic-type pairs, yielding both syntactic well-formedness and two interpretations. It could get worse. But before seeing how bad things could get, let's see if we can find a solution along the lines of the first option: each component has a single syntactic-category / semantic-type specification, and their interaction yields both readings non-deterministically.

5. Disjunction in intensional contexts more generally

Rooth & Partee (1982) discuss a variety of interpretations of disjunction in intensional contexts. The interesting property for us is that in many intensional contexts (that is, based on an impressionistic sample), we find that distributivity of the predicate over the disjunction fails.

(10) a. Sandy wants an espresso or a cappuccino.
  b. Sandy wants an espresso or wants a cappuccino.

(11) a. Mary is looking for a maid or a cook. (Rooth & Partee)
  b. Mary is looking for a maid or looking for a cook.

(12) a. Alex seeks a uniorn or a mastodon.
  b. Alex seeks a unicorn or seeks a mastodon.

Rooth & Partee represent the primary reading (in our sense) of (11a) as

$$\text{look-for}'\,({}^{\wedge}\lambda P\,\exists x[(\text{maid}'(x) \vee \text{cook}'(x)) \wedge {}^{\vee}[P](x)])(m)$$

This representation accounts for the fact that the disjunction (and the existential quantifier ∃x) are inside the scope of the predicate look-for’. As the authors point out (page 355), insight into the properties of this construction depends on connecting it to a meaning postulate like Montague's meaning postulate 9 in *PTQ* (Montague (1974)).

6. Montague's *PTQ*: types and meaning postulates

Montague's *PTQ* system (Montague (1974)) is based on a recursively defined set of syntactic categories, a recursively defined set of semantic types, and a recursively defined mapping from syntactic categories to syntactic types. The syntactic category of transitive verbs, abbreviated TV in *PTQ*, is IV/T, where IV abbreviates *t/e* and T abbreviates *t/(t/e)*. Unpacking all this yields *(t/e)/(t/(t/e))*. The map *f* from syntactic categories maps syntactic *e* to semantic *e* (entities) and syntactic *t* to semantic *t* (truth values) and maps *A/B* to *<<s,f(B)>,f(A)>*. So we can approach the semantic type associated with the syntactic type *(t/e)/(t/(t/e))* in a series of steps.

f((t/e)/(t/(t/e)))   = <<s,f(t/(t/e))>,f(t/e)>
= <<s,<<s, f(t/e)>,f(t)>>,<<s,f(e)>,f(t)>>
= <<s,<<s,<<s,f(e)>,f(t)>>,t>>,<<s,e>,t>>
= <<s,<<s,<<s,e>,t>>, t>>,<<s,e>,t>>

In the *PTQ* system, a transitive verb like *find* or *seek* is associated with an intensional logic (IL) constant of this type and in an interpretation this constant is interpreted arbitrarily in an appropriate domain. This respects types, but it need not respect the internal structure of an instance of that type: if an extensional verb like *find* is combined with a quantifier like *every fish*, the interpretation of *find every fish* applies the interpretation of *find* to the intension of the interpretation of *every fish*. This interpretation maps the interpretation of the quantifier to the interpretation of a verb phrase, but because it's arbitrary, it might just happen (details aside) that the arbitrarily assigned interpretation of phrases of the form *find every ξ* corresponds to the meaning we assign to 'find a ξ' and that the arbitrarily assigned interpretation of phrases of the form f*ind some ξ* corresponds to the meaning we assign to 'find every ξ'. In other words, the arbitrarily assigned interpretations that respect high-level type structure are not sensitive to interpretive fine structure, even where this fine structure has a logical character. Montague's meaning postulates resolve this problem, at least on first blush.

In the case of extensional transitive verbs, the appropriate meaning postulate takes the form shown below(using functional notation, rather than Montague's relational notation, with the types of the individual components following):

$$\exists S \; \forall x \; \forall P \; \Box \; [(\delta P)(x) \leftrightarrow {}^{\vee}P({}^{\wedge}\lambda y \, {}^{\vee}(({}^{\vee}S \, y)({}^{\vee}x)))]$$

expression	*type*
S	*<s,<e, <e,t>>*
x	*<s,e>*
y	*<s,e>*
P	*<s,<<s.<<s.e>,t>>,t>>*
δ	*<<s,<<s,<<s,e>,t>>,t>>,<<s.e>,t>>*

On the righthand side of the meaning postulate, the quantifier *P* takes scope over an expression in which the translation δ has been replaced by a corresponding two-place relation-intension S. The critical point is that on the righthand side, the quantifier's fine-structure is respected and not buried unexpressed within a predicate, as it is on the lefthand side.

In the intensional case, Montague offers a meaning postulate for *seek*, which takes the following form (using the relational notation of the original).[4]

> □ [**seek′**(x,*P*) ↔ **try-to′**(x, ˆ[**find′** (*P*)])], where **seek′**, **try-to′**, **find′** translate **seek**, **try-to**, **find**, respectively.

On the lefthand side of the arrow, the quantifier *P* is stranded inside the arbitrarily assigned interpretation of **seek′**, just as in the extensional case above. On the righthand side, it falls within the scope of the extensional **find′**, and as such will be subject to the meaning postulate for extensional verbs above, which will allow it to take scope over the corresponding lower-typed predicate corresponding to **find′**, in a way allowing its fine-structure to be expressed and thus respected.

Partee & Rooth (1983) show that if the pointwise definition of coordination is accepted for Boolean types (*t* is a Boolean type and if *b* is a Boolean type, then <*a,b*> is a Boolean type, for all types *a*), then Montague's type assignment system faces problems. Compare the following pairs involving conjunctions of extensional and intensional verbs (based on Partee & Rooth's examples (12) and (14)).

(13) a. John caught and ate a fish.
b. John caught a fish and ate a fish.

(14) a. John wants and needs two secretaries.
b. John wants two secretaries and needs two secretaries.

(13b) can be true of a situation in which different fish are caught and eaten. In (13a), only one fish is directly involved. The intensional interpretations of (14a) and (14b), however, seem to be equivalent. These facts follow if we interpret 'caught and ate' as λxλy(((*caught′* x)y) ∧ ((*ate′* x) y)) and interpret 'wants and needs' as λ*P*λy(((*wants′ P*) y) ∧ ((*needs′ P)* y)). With respect to meaning postulates, the type-lowering of the extensional case removes the necessity of a meaning postulate for extensional verbs (with respect to the object position); the type proposed for intensional verbs allows the necessary meaning postulate for intensional verbs to play out inside the scope of the coordinating conjunction. (It's worth noting that compiling the meaning postulate into the interpretation of extensional verbs, for example by interpreting 'caught' as λ*P*λy *P* λx(((*caught*⁎*′* x)y))$ (where *caught*⁎*′*

4 The variables 'x' and '*P'* should be universally quantified.

is extensional in both arguments), does not achieve the desired result: conjoining two such interpretations distributes the object interpretation separately to the conjuncts.)

Earlier, I noted that Rooth & Partee point out the importance of connecting the interpretive properties of representations like the one below to appropriate meaning postulates.

$$\text{look-for}'(\hat{}\lambda P\, \exists x[(\text{maid}'(x) \vee \text{cook}'(x)) \wedge \check{}[P](x)])(m)$$

The reason for this (if there was ever any doubt) should now be clear: the representation itself doesn't provide for the expression of the fine structure of the quantifier argument---an expression that a meaning postulate like Montague's meaning postulate for **seek** provides. And the different ways that this can be done for different cases has a significant impact on semantic inference.

7. Consequences for the analysis of *owe*[5]

Consider now the form of an appropriate meaning postulate for *owe*. Let's start informally with the following (where *R* is a binary relation symbol ranging over a suitable family of possessive and similar binary relations,[6] and we've suppressed initial universal quantifiers in the interests of readability):

a owes b c $\leftrightarrow$ a is under an obligation until a cause R(b, c)

Our goal is a representation that will support the inference:

a owes (b c or b′ c′) $\leftrightarrow$ a is under an obligation until a cause (R(b, c) or R(b′, c′))

To get the primary reading, the functor *owe* has to act on quantifiers and assign them scope inside the scope of *until* on the right. We reformulate accordingly.

a owes Q1 Q2 $\leftrightarrow$ *a is under an obligation until a cause (Q1′λx Q2′λy (R(x,y))*

---

5 For a broader historical perspective, see the Postscript at the end of this paper.

6 The use of a variable in this representation results in a *semantic structure* in the sense of Oehrle (1978). We are not interested as much in paraphrasing or defining exactly the semantic properties associated with an expression or class of expressions as in characterizing the structure of semantic relations in its interpretation.

And it has to be able to do this in a way that combines *Q1'*and *Q2'* into a single package so that such a package or a coordination of such packages can be passed into the scope of *cause* (on the right) as a unit.

There are two formal ways to do this in principle (that I'm aware of): either by composing *Q1* and *Q2* into a single binary quantifier or by treating *Q1* and *Q2* as a product Q1 ⊗ Q2, which combines the grammatical resources into a coordinated package at the observable level, and which can be passed down to an unobservable predicate where it is semantically unpacked. The solution proposed below draws on both approaches.

8. A Solution: composition across a product

In the solution considered here, we make the assumption that *Q2'* will act on the second argument of the binary relation *R*, yielding *λx Q2'λy (R(x,y)*, and *Q1'* will act on this result. We thus assume that *Q2* can be associated with the syntactic category *(np → (np → s)) → (np → s)* and semantic type *<<e,<e,t>>,<e,t>>* and that *Q1* is of category *(np → s) → s* and type *<<e,t>,t>.*[7]

---

7 To implement this solution, we have to make some assumptions---some of which are already on view in the paragraph above. We work in the framework of the associative, commutative Lambek-Van Benthem calculus (van Benthem, 1988). One possible justification for this is that information about order and grouping can be abstracted away from the syntactic combinatory system and reconstructed in an accompanying phonological/orthographical dimension: see Oehrle (1994), de Groote (2001), Muskens (2003), Levine & Kubota (2014), for example. If this is not enough, another possible justification is even more basic: if we can't solve this problem with an associative, commutative system, we will never solve it in a more restricted system. The type system, then, has a binary product ⊗ and a binary implicational type constructor → adjoint to the product. Sequents consist of a single type in the succedent (to the right of the sequent symbol ├), with non-empty antecedents built up by an associative, commutative product '·' (whose properties we silently compile away). We present proofs in the sequent-style natural deduction framework (Troelstra (1992), Troelstra & Schwichtenberg (1996)), which contains introduction and elimination rules for each type constructor in the natural deduction style and yet displays the structured sequent antecedent in a way that makes it possible to include explicit structural rules, as in Gentzen's sequent calculus. To reduce the width of proofs, we will abbreviate the type *np* as *n* and types of the form *a → b* as *(ab)* or *(a b)* (as in van Benthem (1983)). On the semantic side,

We can construct sentences of the form *z owes Q1 Q2* as follows, where the lexical assumptions appear in the axiom leaves and semantic terms are indexed numerically and listed below the proof.

```
                                    n:12 ├ n:16   (n(ns)):17 ├ (n(ns)):18
                                    ------------------------------------------- →E
                                n:8 ├ n:14     n:12 · (n(ns)) ├ (ns):15
                                ---------------------------------------------- →E
                                    n:12 · (n(ns)) · n:8 ├ s:13
                                    ------------------------------ → I
                       Q2 ├ ((ns)s):10             (n(ns)) · n ├ (ns):11
                       ------------------------------------------------------------ →E
                                Q2 · (n(ns)) · n:8 ├ s:9
                                -------------------------- → I
               Q1 ├ ((ns)s):6              Q2 · (n(ns)) ├ (ns):7
               ------------------------------------------------------------ →E
                       Q1 · Q2 · (n(ns)) ├ s:5
                       ---------------------------- → I
       owe ├ ((((n(ns)s)(ns)):3        Q1 · Q2 ├ ((n(ns))s):4
       ------------------------------------------------------------------ →E
   z ├ n                  owe · Q1 · Q2 ├ (ns):2
   --------------------------------------------------------- →E
          z · owe · Q1 · Q2 ├ s:1
          -------------------------------- λ>
          z · owe · Q1 · Q2 ├ s:0
```

our types are extensional but can occur within the scope of an intensional predicate or operator---a situation which blocks extensional inferences (such as existential generalization). This means that β-conversion does not preserve interpretation (as discussed more generally in Gamut (1991, pp. 131ff.). We may as well assume that model-theoretic interpretation is defined only on normal forms.

index	*λ-term*
0	*z is under an obligation until z cause (Q1' λv (Q2' λu (Rv, u)))*
1	*(((λΞ(λ x(x is under an obligation until x cause Ξ(R))))(λT(Q1'(λv(Q2'(λu((Tu)v)))))))(z))*
2	*((λΞ(λ x(x is under an obligation until x cause Ξ(R)))(λT(Q1'(λv(Q2'(λu((Tu)v))))))))*
3	*(λΞ(λ x(x is under an obligation until x cause Ξ(R)))*
4	*λT(Q1'(λv(Q2'(λu((Tu)v)))))*
5	*Q1'(λv(Q2'(λu((Tu)v))))*
6	*Q1'*
7	*λv(Q2'(λu((Tu)v)))*
8	*v*
9	*Q2'(λu((Tu)v))*
10	*Q2'*
11	*λu((Tu)v)*
12	*u*
13	*((Tu)v)*
14	*v*
15	*Tu*
16	*u*
17	*T*
18	*T*

There are really two critical assumptions here. First, based on the assumed meaning postulate, *owe* combines with a binary quantifier in a way that applies the interpretation of the binary quantifier to a possessive relation inside the scope of *cause.* That is, we have (where *Ξ* is a variable ranging over the interpretation of binary quantifiers):

owe' = λΞ(λ x(x is under an obligation until x cause Ξ(R)))

Second, we can combine two unary quantifiers (each of type *(np → s)* → *s*, to form a binary quantifier.

Q1 · Q2 ⊢ ((np → (np →s)) →s):λT(Q1'(λv(Q2'(λu((Tu)v)))))

When we combine *owe* with the two quantifiers in an elimination step, we apply the interpretation of *owe* to the interpretation of the quantifiers, which then normalizes in two β-conversion steps to yield a one-place predicate of type *<e,t>*.

((λΞ(λ x(x is under an obligation until x cause Ξ(R)))(λT(Q1'(λv(Q2'(λu((Tu)v))))))   *λ>*
(λ x(x is under an obligation until x cause (λT(Q1'(λv(Q2'(λu((Tu)v))))))(R)))   *λ>*
(λ x(x is under an obligation until x cause (Q1'(λv(Q2'(λu((Tu)v))))))

Now, we can easily modify the proof above so that the conclusion contains the product of *Q1 ⊗ Q2* of *Q1* and *Q2*: we simply add an ⊗ E elimination step and adjust the Curry-Howard term accordingly.

```
Q1├ Q1:T1     Q2├ Q2:T2                           [as in previous proof]
------------------------------------ ⊗I            --------------------------------------------- → E
Q1 · Q2 ├ Q1 ⊗ Q2: 〈T1,T2〉                             z · owe · Q1 · Q2├ s:1
------------------------------------------------------------------------------------------------ ⊗E
                        z · owe · Q1 ⊗ Q2├ s:1[T1/Q1',T2/Q2']
                        ---------------------------------------------------------- λ>
                        z · owe · Q1 ⊗ Q2├ s:0[T1/Q1',T2/Q2']
```

For the Curry-Howard term associated with the conclusion of the *⊗E* step, in the term associated with the conclusion of this step, we substitute the first and second projections *T1* and *T2* for the corresponding terms *Q1'* and *Q2'* associated with *Q1* and *Q2* in the subproof in the upper right (borrowed wholesale from the previous proof). We display below the results of substituting *T1* and *T2* for *Q1'* and *Q2',* respectively. On the semantic side, this is just book-keeping and its consequences for the last two steps of the proof are shown below.

index   *λ-term[T1/Q1',T2/Q2']*
0   *z is under an obligation until z cause (T1 λv (T2 λu (Rv, u)))*
1   *(((λΞ(λ x(x is under an obligation until x cause Ξ(R))))(λT(T1(λv(T2(λu((Tu)v)))))))(z))*

On the syntactic side, the consequences of these assumptions are not just bookkeeping. The term associated with the product *Q1 ⊗ Q2* is *((np → (np →s)) →s),* which is a Boolean category and thus conjoinable with a pointwise interpretation. Replacing *owe'* with a representation that compiles in the proposed meaning postulate yields the following for *Ricardo owes Q1 Q2*, where as above, *R* ranges over a suitable family of possessive relations.

Ricardo is under an obligation until cause ((λT Q1' λx Q2' λ y ((Ty)x))(R))(Ricardo) λ>
Ricardo is under an obligation until cause (Q1' λx Q2' λ y ((Ry)x))(Ricardo)

For *Ricardo owes Olivia a dollar,* we have:

Ricardo is under an obligation until cause ((λT (λS(S(Olivia')(λx ∃z (λ y ((Ty)x) ∧ dollar' (y))(z)(R))(Ricardo) λ>
Ricardo is under an obligation until cause ∃z (((Rz)Olivia') ∧ dollar' (z))(Ricardo)

For *Ricardo owes Q1a Q2a or Q1b Q2b*, we may take the category of *Q1a Q2a or Q2a Q2b* to be *((np → (np →s)) →s)*, with interpretation defined pointwise:

Q1a Q2a or Q1b Q2b ├ *((np → (np →s)) →s):λT(Q1a'(λv(Q2a'(λu((Tu)v)))) ∨ (Q1b'(λv(Q2b'(λu((Tu)v))))))*

Thus, to get the primary interpretation of *Ricardo owes Q1a Q2a or Q1b Q2b*, we plug this formula into our pattern to get:

Rcrdo is under an obligation until cause (λT(Q1a'(λv(Q2a'(λu((Tu)v)))) ∨ (Q1b'(λv(Q2b'(λu((Tu)v))))))(R))(Rcrdo) λ>
Ricardo is under an obligation until cause(Q1a'(λv(Q2a'(λu((Ru)v)))) ∨ (Q1b'(λv(Q2b'(λu((Ru)v))))(Ricardo)

To get the secondary interpretation ('... but I don't know which'), type-raise *Q1 ⊗ Q2* and coordinate pointwise in the way pioneered by Partee, Rooth (for 'constitutuent' arguments) and Dowty (for 'non-constituent' arguments). To make this easier, write the category of *owe* as *φ → ψ*, where *φ* is *(np → (np → s)) → s)* and *ψ* is *(np → s)*. Then *Q1 ⊗ Q2* (of type *φ)* lifts to a category of semantic type *(φ → ψ) → ψ*, with interpretation below (where '*Θ* ' is a variable ranging over the semantic type or types compatible with the syntactic category *φ → ψ*).

λΘ(Θ(λT(Q1'(λv(Q2'(λu((Tu)v)))))))

Applying this function to our interpretation for *owe* (that is, to the term indexed by 3 in our earlier proof) yields

(λΘ(Θ(λT(Q1'(λv(Q2'(λu((Tu)v))))))))((λΞ(λ x(x is under an obligation until x cause Ξ(R)))),

which normalizes to

$(\lambda x(x \text{ is under an obligation until } x \text{ cause } (Q1'(\lambda v(Q2'(\lambda u((Ru)v)))))),$

just as in the non-lifted case discussed above. Since the lifted case is a Boolean type, we may coordinate instances of it, as shown for the disjunctive case below.

$\lambda\Theta(\Theta(\lambda T(Q1a'(\lambda v(Q2a'(\lambda u((Tu)v)))))) \vee \Theta(\lambda T(Q1b'(\lambda v(Q2b'(\lambda u((Tu)v)))))))$

Applying this function to our interpretation for *owe* distributes this interpretation across the disjunction in the conversion step involving Θ. This is not particularly surprising: it just shows (again) that the type-lifting/pointwise approach is distributive with respect to the category/type that is lifted over. Note that the quantifiers $Q1a', Q2a', Q1b', Q2b'$ are still interpreted inside the scope of Θ on this approach. If Θ is the interpretation of *owe*, this means that the quantifiers will still be interpreted inside the scope of the temporally intensional operator *until*.

To get wider scope interpretations, we use the fact that the following sequent is provable: $owe \cdot np \cdot np \vdash np \rightarrow s$

This is also not surprising, since we have already seen that $owe \cdot Q1 \cdot Q2 \vdash np \rightarrow s$ is provable and that $np \vdash Q$ (for $Q = (np \rightarrow s) \rightarrow np$). (This is a cut-inference, as represented here, but the cut is eliminable.) This means that we can also type *owe*} as $(np \rightarrow (np \rightarrow (np \rightarrow s)$ or as $np \cdot np \rightarrow (np \rightarrow s)$. Any of these *np* argument positions can be the target of a unary quantifier with wide scope. And also not so surprising is the further fact that $owe \cdot np$ is coherent in Ajdukiewicz's sense and can be typed $(np \rightarrow s) \rightarrow s) \rightarrow (np \rightarrow s)$: that is, it can combine with a unary quantifier to form a predicate. We leave the proof as an exercise. The significance of this fact is that the category assigned to *owe* here provides a basis both for the analysis of coordinated double objects (as in *owe Olivia a dollar or Aaron a euro* and for the analysis of coordinated verb and first objects (as in *owe Olivia or promise Aaron a treat*). Similar considerations lead to an analysis of cases like *owe Olivia a dollar today or two dollars tomorrow*.[8]

8 But no such analysis completely characterizes what we owe to Frans.

9. Conclusion

We began with a problem about distributivity in non-constituent conjunction. Our solution to this problem draws on the insights of Partee & Rooth (1983) and Rooth & Partee (1982) and on existing properties of resource-sensitive deduction, using a novel, higher-order type to characterize the argument structure of the intensional verb *owe*---a type that comes clearly into view once an appropriate meaning postulate for *owe* is formulated. In the exposition above, we emphasized the necessity of meaning postulates in Montague's *PTQ* system, if the fine structure of the logical properties of quantifiers and other operators with logical properties are to be respected. This role for meaning postulates has many other applications.

Postscript

In the course of formating the final version of this paper (porting from elegant LaTeX to … Word), I came across some slides by Barbara Partee (2013), in which she states (slide 10) that Montague "gives a similar paraphrase analysis of Buridan's examples with ***owe***" in his paper 'On the Nature of Certain Philosophical Entities' (Montague, 1974, 148-187). This got my attention! Here is the relevant passage (pp. 166 ff.):

> "It was not Quine who first called attention to the kind of difficulty we have just examined. Indeed, Buridan pointed out that the argument
>
> (12) 'Jones owes Smith a horse; therefore there is a horse which Jones owes Smith',
>
> "though intuitively invalid, would appear to be validated by formal criteria. The solution is just as before We notice that 'owes' amounts roughly to 'is obliged to give'; more precisely, we may regard '...owes___to---' as abbreviating '--- is such that … is obliged to give ___ to him'. [where 'him' is evidently bound to a variable introduced by '---' *rto*] For example, if *c*, *d*, *e* are individual constants, then
>
> (13) $\ulcorner c$ owes d to $e \urcorner$
>
> "is taken as
>
> $\vee u\,[u = e \wedge \text{Obliged}\,[c, \tilde{v}\,\text{Gives}\,[v, d, u]]]$,

"where Obliged and Gives are predicate constants of types <-1, 1> and <-1,-1,-1>; Gives [*x, y*, *z*] is read ⌈*x* gives *y* to *z*⌉ and Obliged [*x*, *P*y] is read ⌈*x* is obliged *P*⌉ or ⌈*x* has the obligation *P*⌉. It would not be quite correct to take (13) as

⌈*c* is obliged to give *d* to *e*⌉,

that is,

Obliged [*c*, *ṽ* Gives [*v*, *d*, *e*]],

because examination of examples would reveal that in (13) *e* (though not of course *d*) is 'purely referential'. [footnote omitted-*rto*]. The argument (12) may then be represented as

∨*u* (*u* = Smith ∧ Obliged [Jones, *ṽ* ∨x (Horse [x] ∧ Gives [*v*, *x*, *u*])]); therefore ∨*x* (Horse [*x*] ∧ ∨*u* (*u* = Smith ∧ Obliged [Jones, Gives [*v*, *x*, *u*])],

which is easily shown invalid.

"The solution proposed in these two cases is substantially to reject 'seeks' and 'owes' as predicate constants, and to insist on circumlocution when we might be tempted to use those verbs. We may wonder whether it is possible to approximate English more closely within our intensional language...."

Indeed. Montague's proposed postulate for *owe* is not so different from the one proposed above. Both require a more explicit account of how intensional constructions can depend on temporal properties.

10. References

Chomsky, N. (1957). Syntactic Structures. Mouton, The Hague.

de Groote, P. (2001). Towards abstract categorial grammars. *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings of the Conference*, 148--155.

Dowty, D. (1988). Type raising, functional composition, and non-constituent conjunction. In Oehrle, Bach, and Wheeler, eds., 1988, 153--197.

Gamut, L. T. F. (1991). Logic, Language, and Meaning Volume 2: Intensional Logic and

Logical Grammar. University of Chicago Press, Chicago.
Levine, R. and Kubota, Y. (2014). Gapping as hypothetical reasoning. to appear.
Montague, R. (1974a). On the nature of certain philosophical entities. *Formal Philosophy: Selected Papers of Richard Montague*, 148--187. Yale University Press, New Haven.
Montague, R. (1974b). The proper treatment of quantification in ordinary english. *Formal Philosophy: Selected Papers of Richard Montague*, 247--270. Yale University Press, New Haven. Richmond H. Thomason, ed.
Muskens, R. (2003). Language, lambdas, and logic. In Kruijff, G.-J. and Oehrle, R., editors, Resource Sensitivity in Binding and Anaphora, 23--54. Kluwer, Dordrecht.
Oehrle, R. T. (1971). On gapping. unpublished term paper, Department of Linguistics, MIT.
Oehrle, R. T. (1978). Semantic import of an investigation of the English double object constructions. In W. U. Dressler and W. Mied, editors, Proceedings of the XIIth International Congress of Linguists, Innsbruck, 214--217.
Oehrle, R. T. (1987). Boolean properties in the analysis of gapping. In Syntax and Semantics 20: Discontinuous Constituency, 201--240. Academic Press, Orlando.
Oehrle, R. T. (1994). Term-labeled categorial type systems. Linguistics and Philosophy, 17: 633--678.
Oehrle, R. T., Bach, E., and Wheeler, D. W., editors (1988). Categorial Grammars and Natural Language Structures. D. Reidel, Dordrecht.
Partee, Barbara. (2013). http://people.umass.edu/scable/LING720-A13/Handouts/Partee-Presentation.pdf. Accessed 12 April 2014.
Partee, B. H. and Rooth, M. (1983). Generalized conjunction and type ambiguity. In Bäuerle, R., Schwarze, C., and von Stechow, A., editors, *Meaning, Use, and Interpretation of Language*, Foundations of Communication. De Gruyter, Berlin.
Rooth, M. and Partee, B. H. (1982). Conjunction, type ambiguity, and wide scope ''or''. In Flickinger, D. P., Macken, M., and Wiegand, N., editors, *Proceedings of the First West Coast Conference on Formal Linguistics*, 353—362. Stanford University, Linguistics Department.
Steedman, M. (1985). Dependency and coordination in the grammar of Dutch and English. *Language*, 61:523--568.
Troelstra, A. S. (1992). *Lectures on Linear Logic*. Center for the Study of Language and Information, Stanford, California.
Troelstra, A. S. and Schwichtenberg, H. (1996). *Basic Proof Theory*, Cambridge Tracts in Theoretical Computer Science vol 43. Cambridge University Press, Cambridge.
van Benthem, J. (1986). Essays in Logical Semantics. D. Reidel, Dordrecht.
van Benthem, J. (1988). The Lambek calculus. In Oehrle, Bach, and Wheeler, eds., 1988, 35--68.