# Canonical Constituents and Non-canonical Coordination
# Simple Categorial Grammar Account

Oleg Kiselyov

University of Tsukuba, Japan
oleg@okmij.org

**Abstract.** A variation of the standard non-associative Lambek calculus with the slightly non-standard yet very traditional semantic interpretation turns out to straightforwardly and uniformly express the instances of non-canonical coordination while maintaining phrase structure constituents. Non-canonical coordination looks just as canonical on our analyses. Gapping, typically problematic in Categorial Grammar–based approaches, is analyzed just like the ordinary object coordination. Furthermore, the calculus uniformly treats quantification in any position, quantification ambiguity and islands. It lets us give what seems to be the simplest account for both narrow- and wide-scope quantification into coordinated phrases and of narrow- and wide-scope modal auxiliaries in gapping.
The calculus lets us express standard covert movements and anaphoric-like references (analogues of overt movements) in types – as well as describe how the context can block these movements.

## 1 Introduction

Non-canonical coordination and in particular gapping (2) are challenges to semantic theory [2, 3].

(1) John gave a book to Mary and a record to Sue.

(2) I gave Leslie a book and she a CD.

(3) John gave a present to Robin on Thursday and to Leslie on Friday.

(4) Mrs. J can't live in Boston and Mr. J in LA.

Further challenges are accounting for both narrow- and wide-scope reading of "a present" in (3) and for the two readings in (4), with the wide-scope "can't" and the wide-scope coordination.

While CCG treats the non-constituent coordination with no problems [6], it comes at the price of giving up on phrase structure constituents. The CCG analysis of gapping requires further conceptually problematic (and eventually, not fully adequate) postulates. Kubota and Levine [3] present a treasure trove of empirical material illustrating the complexities of coordination (from which we drew our examples). They develop a variant of type-logical categorial grammar with both directed and undirected implications and higher-order phonology. Here we demonstrate that the challenges of coordination and gapping can be met within the standard non-associative Lambek calculus **NL**, one of the basic categorial grammars. Our calculus represents arbitrary discontinuous constituents, including multiple discontinuities and the discontinuous displaced material. Hence our calculus seems both simpler and more expressive than other type-logical grammar approaches [5].

The main idea is a slightly non-standard semantic interpretation of **NL** derivations (the phonology remains standard). It opened the way to represent what looks like typical overt and covert movements. Being strictly within **NL**, we add no modes or structural rules to the calculus per se. No lexical items are ever moved. The antecedent of a sequent, describing the *relevant* phrase structure, can still be manipulated by the standard rules, given constants of particular types. These constants have the empty phonology but the clear semantic (computational) interpretation. One may say that the structural rules (even those that account for matching of a 'hole' in one part of the derivation with the referent in another) are all lexicalized and computational. Although our approach may be reminiscent of QR and dynamic semantics, we stay squarely within logic: the semantic interpretation is compositionally computed by combining closed formulas using the standard operations of higher-order logic. Unbound traces, free variables, any movement or rearrangement of lexical items are simply not possible in our approach.

After presenting the calculus in §2 we illustrate its various features in §3.1 on very simple examples of coordination. That section presents the two techniques that underlie the analyses of more complex non-constituent coordination and gapping in §3.3. The same techniques also account for quantification: in §4 we analyze not only the simple QNP in subject and object positions but also the quantification ambiguity and the scoping islands. Finally, §5 treats the seemingly anomalous wide scope of QNP and modals in sentences with non-canonical coordination, which proved most difficult to account for in the past. We answer the challenges posed at the beginning of this section.

All the examples in this abstract have been mechanically verified. The accompanying code `http://okmij.org/ftp/gengo/HOCCG.hs` is the verification record. The code also has more examples and their analyses.

## 2 Non-associative Lambek calculus and theory

After reminding of the non-associative Lambek calculus **NL**, we derive convenient rules and introduce types and constants to be used throughout the paper. The semantic interpretation is in §3.2.

Figure 1 is our presentation of **NL**. Types $A, B, C$ are built with the binary connectives / and \; antecedent structures $\Gamma, \Delta$ are built with $(-, -)$ and the empty structure $\bullet$. We stress that the antecedent structure is an ordered tree and hence $(A, \bullet)$ is different from just $A$. Labeled sequents have the form $\Gamma \vdash t : A$, to be read as the term $t$ having the type $A$ assuming $\Gamma$. The figure presents the standard introduction and elimination rules for the two binary connectives (slashes). To ease the notational burden, we write $t_1 t_2$ for both left and right applications

$$\frac{\Gamma \vdash t_1 : B/A \quad \Delta \vdash t_2 : A}{(\Gamma, \Delta) \vdash t_1 t_2 : B}\,/E \qquad \frac{\Gamma \vdash t_1 : A \quad \Delta \vdash t_2 : A\backslash B}{(\Gamma, \Delta) \vdash t_1 t_2 : B}\,\backslash E$$

$$\frac{(\Gamma, A) \vdash t : B}{\Gamma \vdash \leftthreetimes t : B/A}\,/I \qquad \frac{(A, \Gamma) \vdash t : B}{\Gamma \vdash \rightthreetimes t : A\backslash B}\,\backslash I$$

$$\frac{}{A \vdash x : A}\,Var$$

**Fig. 1.** The non-associative Lambek calculus **NL**

As basic (atomic) types we choose $NP$ and $S$ plus a few others used for quantification and to mark context boundaries. They will be introduced as needed. We also use a parallel set of types, which we write as $\overline{NP}$, $\overline{NP\backslash S}$, etc. As far as calculus is concerned, they are not special in any way: one may regard $\overline{A}$ as an abbreviation for $A\backslash\bot$ where $\bot$ is a dedicated atomic type.

The introduction rules $\backslash I$ and $/I$ almost always[1] appear in combination with the corresponding elimination rules. To emphasize the pattern and to save space in derivations we introduce the admissible cut-like rules HypL and HypR in Figure 2. HypL is $\backslash I$ immediately followed by $\backslash E$; HypR is similar.

The calculus per se has no structural rules. Still, the existing rules may manipulate the antecedent structure $\Gamma$ using constants of appropriate types. For example, consider the sequent $(\bullet, (\bullet, A)) \vdash t : B$. Applying $\backslash I$ twice we derive $A \vdash \rightthreetimes\rightthreetimes t : \bullet\backslash(\bullet\backslash B)$. The derivation shows that we are not really distinguishing types and structures: structures are types. We will call types that are structures or include structures (that is, contain $\bullet$ and

[1] A notable exception is quantification, see §4.

commas) full structure types. Assuming the constant $ooL$ with the type $\bullet \vdash ooL : B/(\bullet\backslash(\bullet\backslash B))$ gives us $(\bullet, A) \vdash ooL(\leftthreetimes \leftthreetimes t) : B$. In effect, $ooL$ transformed $(\bullet, (\bullet, A)) \vdash B$ into $(\bullet, A) \vdash t : B$. Since this and other similar structural transformations will appear very frequently, we will abbreviate the derivations through the essentially admissible rule Hyp [2] in Figure 2.

$$\frac{\Gamma \vdash t_1 : A \quad (A, \Delta) \vdash t_2 : B}{(\Gamma, \Delta) \vdash t_1 \cdot t_2 : B}\,HypL \qquad \frac{(\Delta, A) \vdash t_2 : B \quad \Gamma \vdash t_1 : A}{(\Delta, \Gamma) \vdash t_1 \cdot t_2 : B}\,HypR$$

$$\frac{\Delta\backslash A \vdash t_1 : \Gamma\backslash A \quad \mathcal{C}[\Delta] \vdash t_2 : A}{\mathcal{C}[\Gamma] \vdash t_1 \uparrow t_2 : A}\,Hyp$$

**Fig. 2.** Convenient derived rules (In the rule Hyp, $\Gamma$ must be a full structure type.)

The Hyp rule can replace a structure type $\Delta$ within the arbitrary context $\mathcal{C}[]$ of another structure. The replacement must be a full structure type. Although our calculus has no structural rules whatsoever, Hyp lets us rearrange, replace, etc. parts of the antecedent structure, *provided* there is a term $t$ of the suitable type for that operation. One may say that our structural rules are lexicalized and have a computational interpretation. For now we introduce the following schematic constants.

$$\begin{aligned} (\bullet, \bullet)\backslash A &\vdash oo : \bullet\backslash A \\ (\bullet, (\bullet, \Gamma))\backslash A &\vdash oL : (\bullet, \Gamma)\backslash A \\ ((\Gamma, \bullet), \bullet)\backslash A &\vdash oR : (\Gamma, \bullet)\backslash A \\ (C, \bullet)\backslash S &\vdash resetCtx : \bullet\backslash S \end{aligned}$$

The types of these terms spell out the structural transformation. We will later require all such terms, whose type is a full structure type, be phonetically silent. In $resetCtx$, $C$ is any context marker (such as those that mark the coordinated or subordinated clause). The corresponding structural rule drops the context marker when the context becomes degenerate.

§3.1 below illustrates the calculus on many simple examples. Throughout the paper, we will write $VP$ for $NP\backslash S$, $VT$ for $VP/NP$ and $PP$ for $VP\backslash VP$.

## 3 Coordination

This section describes analyses of coordination in our calculus, from canonical to non-canonical. We show two approaches. The first is less general

[2] We call this rule essentially admissible because it cannot transform $(\bullet, \bullet) \vdash A$ to $\bullet \vdash A$. Therefore, we will have many derivations and sequents that differ only in $(\bullet, \bullet)$ vs. $\bullet$. Since they are morally the same, it saves a lot of tedium to treat them as identical, assuming that $(\bullet, \bullet)$ can always be replaced by $\bullet$. We will use this assumption throughout.

and does not always apply, but it is more familiar and simpler to explain. It builds the intuition for the second, encompassing and general approach. The approaches are best explained on very simple examples below. §3.3 applies them to non-canonical coordination and gapping; §5 to scoping phenomena in coordination.

### 3.1 Two approaches to coordination

We start with the truly trivial example "John tripped and fell." VP coordination is the simplest and the most natural analysis, assuming the constant *and* : $(VP\backslash VP)/VP$. Here is another analysis, with coordination at type $S$ rather than $VP$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{NP \vdash x : NP} \quad \dfrac{}{\bullet \vdash \text{tripped} : VP}}{(NP,\bullet) \vdash (x\ \text{tripped}) : S}\backslash E}{} \qquad \dfrac{}{And \vdash \text{and} : (S\backslash S)/S} \qquad \dfrac{\dfrac{\dfrac{}{NP \vdash y : NP} \quad \dfrac{}{\bullet \vdash \text{fell} : VP}}{(NP,\bullet) \vdash (y\ \text{fell}) : S}\backslash E}{}}{((NP,\bullet),(And,(NP,\bullet))) \vdash (x\ \text{tripped})\ \text{and}\ (y\ \text{fell}) : S}$$

Here, $And$ is an atomic type, used to mark the coordination in the antecedent structure[3]. Clearly the derivation can be reconstructed from its conclusion. To save space, we will only write the conclusion. To proceed further, we assume the "structural constant" (schematic)

$$(\Gamma,(And,\Gamma))\backslash A \vdash and_C : \Gamma\backslash A$$

This constant has the full structure type and is hence phonetically silent. (All terms in the same font as *and* are silent.) The Hyp rule then gives

$$(NP,\bullet) \vdash and_C \uparrow (x\ \text{tripped})\ \text{and}\ (y\ \text{fell}) : S$$

We now can apply the HypL rule with $\bullet \vdash \text{John} : NP$ obtaining

$$(\bullet,\bullet) \vdash \text{John} \cdot and_C \uparrow (x\ \text{tripped})\ \text{and}\ (y\ \text{fell}) : S$$

Finally Hyp is used once again, with the *oo* constant to contract $(\bullet,\bullet)$ to just $\bullet$ producing the final conclusion. The phonology is standard; hypotheses, *and* and other constants in the same font are silent. The semantic interpretation is described in §3.2.

There is yet another analysis of the same trivial example. Although patently overkill in this case, it explains the most general technique to

[3] Incidentally, such a mark restricts the use of the structure constants such as $and_L$ and especially $and_D$ below – in effect restricting gapping to coordination.

be used for non-canonical coordination. The intuition comes from the apparent similarity of our analysis of quantification in §4 to the quantifier raising (QR). The key idea is to paraphrase "John tripped and fell" as "John tripped; he fell" with the silent 'pronoun'. The paraphrase is analyzed in the manner reminiscent of dynamic logic. We assume the axiom schema

(5) $\overline{A} \vdash \mathit{ref} : A/A$

where $\overline{A}$ is intended to signify that *ref provides* the value of the type $A$. We derive

$$((\overline{NP}, \bullet), \bullet) \vdash (\mathit{ref}\,\mathrm{John})\ \mathrm{tripped} : S$$

and, as before, $(NP, \bullet) \vdash x\ \mathrm{fell} : S$, whose coordination produces:

$$(((\overline{NP}, \bullet), \bullet), (And, (NP, \bullet))) \vdash (\mathit{ref}\,\mathrm{John})\ \mathrm{tripped\ and}\ (x\ \mathrm{fell}) : S$$

Matching the reference $x : NP$ with its referent $\overline{NP}$ is done by the following two structural constants.

$(((\overline{A}, \bullet), \mathit{\Gamma}), (And, (A, \mathit{\Delta})))\backslash S \vdash \mathit{and}_L : ((\bullet, \mathit{\Gamma}), (And, (\bullet, \mathit{\Delta})))\backslash S$
$((\mathit{\Gamma}, A), (And, (\mathit{\Delta}, (\overline{A}, \bullet))))\backslash S \vdash \mathit{and}_R : ((\mathit{\Gamma}, \bullet), (And, (\mathit{\Delta}, \bullet)))\backslash S$

The constant $\mathit{and}_L$ matches up the reference $A$ and the referent $(\overline{A}, \bullet)$ and replaces both with $\bullet$ – provided both occur at the left edge and the referent is in the left conjunct. The motivation for this choice comes from the HypL rule, which $\mathit{and}_L$ is meant generalize. If we can derive

$$t_1 \cdot \mathit{and}_C \uparrow (x\ t_2)\ \mathrm{and}\ (y\ t_3)\ \ : S$$

with the HypL rule, we should also be able to derive

$$\mathit{and}_L \uparrow ((\mathit{ref}\ t_1)\ t_2)\ \mathrm{and}\ (y\ t_3)\ \ : S$$

The latter also applies when $t_2$ and $t_3$ have different antecedents. The conclusion of our example is then

$$\bullet \vdash \mathit{and}_C \uparrow \mathit{and}_L \uparrow (\mathit{ref}\,\mathrm{John})\ \mathrm{tripped\ and}\ (x\ \mathrm{fell}) : S$$

(we shall elide the final step of reducing $(\bullet, \bullet)$ to $\bullet$ from now on). Again, the read-out is standard, keeping in mind that all italicized items are silent. The semantic interpretation is described in §3.2.

The subject coordination, as in "John and Mary left." is similar. Besides the natural $NP$ coordination (if we assume and : $(NP\backslash NP)/NP$), we can coordinate at type $S$. Our first approach leads to

$$(\bullet, VP) \vdash \mathit{and}_C \uparrow (\mathrm{John}\ x)\ \mathrm{and}\ (\mathrm{Mary}\ y) : S$$

followed by the application of the HypR rule with left : $VP$ on the right. The more general approach derives $(\bullet, VP) \vdash$ (John $x$) : $S$ with the $VP$ hole for the left conjunct and $(\bullet, (\overline{VP}, \bullet)) \vdash$ Mary (*ref* left) : $S$ for the right conjunct. The right-edge hole is matched up with the right-edge referent by $and_R$, which requires the referent to be in the right conjunct. Again, the motivation is to generalize the HypR rule, which comes from the $/I$ rule of the Lambek calculus.

Object coordination "John saw Bill and Mary." lets us introduce the final structural constant of our approach, which matches a hole with a referent in the medial position rather than at the edge. Our first approach invariably leads to

$$(NP, (TV, \bullet)) \vdash and_C \uparrow (x\ (y \text{ Bill})) \text{ and } (u\ (v \text{ Mary}))$$

which is the dead-end: since $TV$ is not at the edge of the antecedent structure, neither HypL nor HypR can eliminate it. The first approach, although simple, clearly has limitations – which should be obvious considering it is just the standard Lambek calculus. The latter too has trouble with eliminating hypotheses far from the edges.

The second approach produces

$$((\bullet, ((\overline{TV}, \bullet), \bullet)), (And, (\bullet, (TV, \bullet)))) \vdash$$
$$and_L \uparrow (\mathit{ref} \text{ John})\ ((\mathit{ref} \text{ see}) \text{ Bill}) \text{ and } (x\ (y \text{ Mary})) : S$$

Since the referent is now deep in the structure, neither $and_L$ nor $and_R$ can get to it. We have to use the more general constant

$$((\Gamma_1, ((\overline{TV}, \bullet), \Gamma_2)), (And, (\Delta_1, (TV, \Delta_2))))\backslash S \vdash$$
$$and_D : ((\Gamma_1, (VB, \Gamma_2)), (And, (\Delta_1, (VB, \Delta_2))))\backslash S$$

(it is actually a family for different shapes of contexts). Although this constant seems to give rise to an unrestricted structural rule, it is limited by the type of the hole. Since the hole is for a term deep inside a formula, that term must denote a relation, that is, have at least two arguments. The hole is thus restricted to be of the type $TV$, $VP/PP$ and similar. Kubota and Levine discuss in detail a similar restriction for their seemingly freely dischargeable hypotheses in [4, §3.2]. The context marker $VB$ tells that the verb has been gapped.

One may be concerned that plugging the hole is too loose a feature, letting us pick any word in the left conjunct and refer to it from the right conjunct, or vice versa. We could then derive "*John tripped and." (with the hole in the right conjunct referring to "tripped"). Such a derivation is not possible however. We can get as far as

$$(((\overline{NP}, \bullet), (\overline{VP}, \bullet)), (And, (NP, VP))) \vdash$$
$$(\mathit{ref} \text{ John})\ (\mathit{ref} \text{ tripped}) \text{ and } (x\ y) : S$$

Although we can plug the $NP$ hole at the left edge with $and_L$, after that we are stuck. Since the $VP$ hole is at the right edge in the antecedent structure of the conjunct, it can be eliminated only if we apply $and_R$ – which however requires the referent to be in the right conjunct rather than the left one. On the other hand, $VP$ does not denote a relation and $and_D$ does not apply either. (The latter also does not apply because it targets holes in the middle of the structure rather than at the edge.)

The prominent feature of our second approach is using the Var rule to introduce into the derivation a hypothetical $NP$, $VP$, $TV$ or other phrase. That hypothesis is eliminated by "matching it up" with the suitable referent in the other coordinated clause. This hypothesis is strongly reminiscent of a trace, or discontinuity (as in Morrill et al. [5]). It is also reminiscent of anaphora, especially of the sort used in Montague's PTQ. To be sure, this 'anaphora' differs notably from overt pronouns or even null pronouns. When targeted by the HypR rule, the hypothesis acts as a pure cataphora rather than anaphora. Mainly, the rules of resolving our 'pronoun', such as $and_R$, are quite rigid. They are syntactic rather than pragmatic, based on matching up two derivations, one with the hole and the other with the referent. The derivations should be sufficiently similar, 'parallel', for them to match. To avoid confusion, we just call our hypothetical phrase a hole, a sort of generalized discontinuity. Unlike Morrill, this hole may also occur at the edge.

### 3.2 Semantic (computational) interpretation

The phonological interpretation of a derivation – obtaining its yield – is standard. We read out the fringe (the leaves) of the derivation tree in order, ignoring silent items. This section expounds the conservative, traditional and yet novel semantic interpretation. It is this new interpretation that lets us use **NL** for analyzing phenomena like gapping, quantifier ambiguity and scope islands that were out of its reach before.

The semantic interpretation of a grammatical derivation maps it to a logical formula that represents its meaning. The mapping is compositional: the formula that represents the meaning of a derivation is built from the formulas for sub-derivations. In our interpretation, the meaning of every derivation, complete or incomplete, is represented by an always *closed* formula in the higher-order logic: the simply-typed lambda calculus with equality and two basic types $e$ and $t$. Although pairs are easy to express in lambda calculus (using Church encoding), for notational convenience we will treat pairs, pair type $(A, B)$ and the unit type $()$ as primitives. Another purely notational convenience is pattern-matching to access the components of a pair; for example, to project the first component we write $\lambda(x, y).x$. (The accompanying code uses the projection functions instead of pattern-matching, which are expressible in lambda-calculus.) We write underscore for the unused argument of the abstraction.

Our interpretation maps every **NL** sequent $\Gamma \vdash A$ to a closed formula of the type $\lceil \Gamma \backslash A \rceil$ where the homomorphic map from the **NL** types to the semantic types is given in Figure 3. (The interpretation of a sequent confirms that a structure is really treated as a type.)

$$
\begin{array}{ccc}
NP & \mapsto & e \\
S & \mapsto & t \\
A\backslash B & \mapsto & \lceil A \rceil \to (\lceil B \rceil, t) \\
B/A & \mapsto & \lceil A \rceil \to (\lceil B \rceil, t) \\
\bullet & \mapsto & () \\
(A, B) & \mapsto & (\lceil A \rceil, \lceil B \rceil) \\
\overline{A} & \mapsto & \lceil A \rceil \\
And & \mapsto & () \\
\multicolumn{3}{c}{\text{the same for all other context markers}} \\
\mathcal{U} & \mapsto & e \\
\mathcal{E} & \mapsto & e
\end{array}
$$

**Fig. 3.** Mapping **NL** types to semantic types

Intuitively, every sequent is treated as a computation which, when given the values representing assumptions will produce the term representing the conclusion. In particular, the conclusion of the complete derivation, the sequent $\bullet \vdash S$ is mapped to the formula of the type $() \to (t, t)$. It is the computation which, when applied to $()$ (the trivial assumption, the synonym for $\top$) will produce *two* truth values. Their conjunction represents the truth of the proposition expressed by the original sequent. The first truth value in the pair is traditional whereas the second represents side-conditions. This splitting off of the side conditions (which are composed separately by the inference rules) is the crucial feature of our interpretation. The type mapping in Figure 3 maps directional implications to functions from assumptions to the conclusion plus the side-conditions. Contextual markers such as *And*, *BV*, etc. have no semantic significance.

Each axiom (sequent) is mapped to a logical formula of the corresponding semantic type. Each rule of the calculus combines the formulas of its premises to build the formula in the conclusion. For example, the rule $\backslash I$ takes the sequent $(A, \Gamma) \vdash B$ (whose interpretation, to be called $f$, has the type $(\lceil A \rceil, \lceil \Gamma \rceil) \to (\lceil B \rceil, t))$ and derives the sequent

$\Gamma \vdash A\backslash B$. Its interpretation is the formula $\lambda g.((\lambda a.f(a,g)),\top)$ of the type $\lceil\Gamma\rceil \to (\lceil A\rceil \to (\lceil B\rceil, t), t)$. More interesting are the eliminations rules, for example, $\backslash E$. Recall, given the formula (to be called $x$) interpreting the sequent $\Gamma \vdash A$ and the formula $f$ for the sequent $\Delta \vdash A\backslash B$, the rule builds the interpretation of $(\Gamma,\Delta) \vdash B$:

$$\begin{aligned}\lambda(g,d).\ &\text{let } (f_v, f_s) = f\ d \text{ in}\\ &\text{let } (x_v, x_s) = x\ g \text{ in}\\ &\text{let } (b_v, b_s) = f_v\ x_v \text{ in}\\ &(b_v, b_s \wedge f_s \wedge x_s)\end{aligned}$$

(where let $x = e_1$ in $e_2$ is the abbreviation for $(\lambda x.e_2)e_1$.) Informally, whereas introduction rules correspond semantically to a $\lambda$-abstraction, elimination rules correspond to applications. In our interpretation, the elimination rules also combine the side-conditions.

As an illustration we show the semantic interpretation of two derivations from the previous section. The first is rather familiar

$$\bullet \vdash oo \uparrow \text{John} \cdot and_C \uparrow (x \text{ tripped}) \text{ and } (y \text{ fell}) : S$$

and its interpretation is unsurprising. The sequent $\bullet \vdash \text{John} : NP$ is assigned the logical formula $\lambda\_.(\mathsf{john},\top)$ where $\mathsf{john} : e$ is the domain constant. The semantic interpretation of the other axiom sequents is similar. The constant

$$(\Gamma,(And,\Gamma))\backslash A \vdash and_C : \Gamma\backslash A$$

corresponds to the formula

$$\lambda f.(\lambda d.f\ (d,((),d)),\top) : ((\lceil\Gamma\rceil,((),\lceil\Gamma\rceil)) \to (\lceil A\rceil,t)) \to (\lceil\Gamma\rceil \to (\lceil A\rceil,t))$$

All side-conditions are $\top$; the sequent for the sentence is interpreted as $\lambda\_.(\mathsf{tripped\ john} \wedge \mathsf{fell\ john},\top)$.

More interesting is the derivation with holes and referents:

$$\bullet \vdash and_C \uparrow and_L \uparrow (\mathit{ref}\,\text{John}) \text{ tripped and } (x \text{ fell}) : S$$

The referent maker $\overline{A} \vdash \mathit{ref} : (A/A)$ is interpreted as $\lambda\overline{x}.(\lambda x.(x, x = \overline{x}),\top)$ That is, *ref* John corresponds to a formula that receives an $\overline{NP}$ assumption (the value of the type $e$) and produces $\mathsf{john}$, what John by itself would have produced, *along with* the side-condition that the received assumption must be $\mathsf{john}$. The structured constant $and_L$ matches up the hole and the referent. Recall, it converts the structure of the type $((((\overline{A},B),\Gamma),(And,(A,\Delta)))$ into $(((B,\Gamma),(And,(\bullet,\Delta)))$ which no longer has the assumption $A$ in it (nor the matching $\overline{A}$). This assumption is eliminated 'classically' by assuming it with the existential quantifier.

$$
\begin{aligned}
&(\lambda f.\ (\lambda(((),g),((),((),d))).\\
&\quad \exists x.\text{let } (b_v,b_s) = f\ (((x,()),g),((),(x,d))) \text{ in } b_v \wedge b_s,\\
&\ \top),\top)
\end{aligned}
$$

The quantified variable is passed as the $A$ assumption and as the $\overline{A}$ assumption. The latter will later be converted to the side condition that the quantified variable must be equal to the referent. The final truth condition is represented by the formula $\exists x.(\mathsf{tripped\ john} \wedge \mathsf{fell}\ x) \wedge x = \mathsf{john}$ that shows how the hole in the conjunct $\mathsf{fell}\ x$ gets filled by the referent $\mathsf{john}$.

We have just demonstrated a dynamic-semantic–like approach that uses only the traditional means (rather than mutation, continuation and other powerful features) to accomplish the filling of a hole with a referent deep in the tree. Our side-condition plays the role of the constraint store, or of the current substitution in unification algorithms. We prefer to view our semantics in structural rather than dynamic terms, as matching up/unifying derivations (trees) rather than mutating the shared 'discourse context'.

### 3.3 Non-canonical coordination and gapping

We now apply the two methods from the previous section to analyses of non-canonical coordination. Our first example is "John liked and Mary hated Bill."

$$(and_C \uparrow \text{John (liked } x\text{) and (Mary (hated } y\text{)))} \cdot \text{Bill}$$

The last significant rule in the derivation is HypR. The very similar derivation can be given with the hole-referent approach. Unlike CCG, we do not treat 'John liked' as a constituent. In fact, the latter is not derivable in our calculus.

We stress that "*John liked Bill and Mary hated $\phi$" is not derivable: since Bill occurs at the right edge of the conjunct, it can only be targeted by $and_R$. However, that rule requires the referent to be in the right conjunct rather than the left conjunct.

Next is gapping, for example "Mary liked Chicago and Bill Detroit."

$$and_D \uparrow \text{Mary ((}ref\text{ liked) Chicago) and (Bill (}x\text{ Detroit))}$$

The analysis is almost identical to the object coordination analysis in §3.1. It may seem surprising that in our calculus such a complex phenomenon as gapping is analyzed just like the simple object coordination.

As presented, the analysis can still overgenerate, for example, erroneously predicting coordination with a lower clause:

*John bought a book and Bill knows that Sue a CD.

The problem can be easily eliminated with the mechanism that used in §4 for quantification islands. Kubota and Levine [4] however argue against the island conditions in gapping. They advocate that we should accept the above sentence *at the level of derivation (combinatorial grammar)* and rule it out on pragmatical grounds. Although Kubota and Levine's argument applies as it is in our case, we should stress that our calculus does offer a mechanism to express the requirement that the coordinated clauses must be 'parallel' (see the type of $and_C$ for example). We could specify, at the level of combinatorial grammar, exactly what it means.

## 4 Quantification

The techniques introduced to analyze non-canonical coordination turn out to work for quantification in any position, quantification ambiguity and scope islands.

We start with the deliberately simple example "John liked everyone", where 'everyone' is typed as $\mathcal{U} \vdash NP$ where $\mathcal{U}$ is an atomic type. From its simple $NP$ type, 'everyone' looks like an ordinary NP, letting us easily derive

$$(\bullet, (\bullet, \mathcal{U})) \vdash \text{John (liked everyone)} : S$$

'Everyone' however has the assumption $\mathcal{U}$, which has to be eventually discharged. The only way to do it in our system is to find a structural rule (structural constant) that moves $\mathcal{U}$ to the left edge of the antecedent structure, where it can be abstracted by the rule $\backslash I$. We do in fact have such a structural constant, which, in combination with the Hyp rule, converts $(\bullet, (\bullet, \mathcal{U}))$ to $(\mathcal{U}, (\bullet, \bullet))$. We call it $float_U$ (actually, it is a combination of constants, each responsible for smaller-step $\mathcal{U}$ 'movements'). Once $\mathcal{U}$ is floated to the left edge of the antecedent, the $\backslash I$ rule gives $\bullet \vdash \mathcal{U} \backslash S$. The final step applies $/E$ with the silent constant $\bullet \vdash \mathit{forall} : (S/(\mathcal{U}\backslash S))$ producing

$$\bullet \vdash \mathit{forall}\ (float_U \uparrow \text{(John (liked everyone))}) : S$$

The analysis wrote itself: each step is predetermined by the types and the available constants. It only works if $\mathcal{U}$ is allowed to float to the top of the assumption structure, which has been the case.

The semantic interpretation maps $\mathcal{U}$ (and the similar $\mathcal{E}$ below) to $e$ and interprets 'everyone' as essentially the identity function. The semantics of *forall* is $\lambda\_.(\lambda k.(\forall x.\text{let } (b_v, b_s) = k\ x \text{ in } b_s \Rightarrow b_v, \top), \top)$ The meaning of the phrase is hence given by the formula $(\lambda k.\forall x.kx)\ (\lambda x.\ \mathsf{like}\ x\ \mathsf{john})$.

Existential quantification uses the $\mathcal{E}$ hypothesis. Quantification in the subject and even medial positions are just as straightforward. The key is the ability to float $\mathcal{U}$ or $\mathcal{E}$ to the left edge of the antecedent.

Our approach treats the quantification ambiguity. Consider "Someone likes everyone". At an intermediate stage we obtain

$$(\mathcal{E}, (\bullet, \mathcal{U})) \vdash \text{someone (like everyone)} : S$$

Since $\mathcal{E}$ is already at the left edge, it can be abstracted by $\backslash I$ and discharged by the application of *exists*. The hypothesis $\mathcal{U}$ still remains; it can be floated and then discharged by *forall*. The resulting truth condition shows the inverse reading of the sentence. If $\mathcal{U}$ is floated first, the linear reading results.

We stress that the analysis of quantification crucially relies on the ability to float the hypotheses $\mathcal{U}$ or $\mathcal{E}$ to the left edge, permuting them with the other components of the structure. We can easily block such moves by introducing context markers, for example, *Clause*:

$$Clause \vdash \text{TheFactThat} : NP/S$$

We can then posit that $\mathcal{U}$ is not commutable with *Clause*, which explains why "That every boy left upset a teacher" is not ambiguous. The hypothesis $\mathcal{E}$ may still be allowed to commute with *Clause*, hence letting existentials take scope beyond the clause.

It is instructive to compare the present analyses of the quantification ambiguity and islands with the continuation analyses of [1]. The latter rely on the so called quantifier strength (the position of quantifiers in the continuation hierarchy) to explain how one quantifier may outscope another. Here we use essentially the structural rules (programmed as special lexical items to be applied by the Hyp rule). In the continuation analyses, the clause boundary acts as a 'delimiter' that collapses the hierarchy and hence prevents the quantifiers within from taking a wider scope. Essentially the same effect is achieved here by simply not permitting reassociation and commutation with *Clause*.

For the sake of the explanation we have used the very simple QNP "everyone" and "someone" with the unrealistically trivial restrictors. Our approach handles arbitrary restrictors. It turns out the 'side-conditions' in the semantic representation are exactly the restrictors of the quantification. For the lack of space we can only refer to the accompanying source code for details.

## 5 Anomalous scoping in non-canonical coordination

We now combine the analyses of quantification and coordination and apply them to the wide scoping of quantifiers in gapped sentences, for example: "I gave a present to Robin on Thursday and to Leslie on Friday."

We start with the straightforward derivation

$$(\bullet, \mathcal{E}) \vdash \text{gave (a present)} : VP/PP$$

and extend it to
$((\overline{NP}, \bullet), (((\overline{VP/PP}, (\bullet, \mathcal{E})), (\bullet, \bullet)), (\bullet, \bullet))) \vdash$
(*ref* John) (((*ref*(gave (a present))) (to Bill)) (on Monday))
The derivation for the second conjunct assumes the 'holes' $x_j : NP$ and $x_g : VP/PP$:

$$(NP, ((VP/PP, (\bullet, \bullet)), (\bullet, \bullet))) \vdash x_j \;((x_g \text{ (to Leslie)}) \text{ (on Friday)})$$

The NP hole $x_j$ is at the left edge of the antecedent and is filled with the referent "John", using Hyp and the constant $and_L$. The hole $x_g$ is filled with the referent (gave (apresent)), using Hyp and $and_D$. The type $VP/PP$ corresponds to a relation (between $NP$ and $PP$) and hence can be used with $and_D$. Finally, $\mathcal{E}$ in the remaining antecedent structure is floated to the left edge and eliminated with *exists*. The semantic interpretation shows the wide scope of "a present".

The also available narrow-scope reading is produced by using the hole of the type $\mathcal{E}\backslash(VP/PP)$, whose referent is obtained from (gave(a present)) by floating $\mathcal{E}$ to the right edge and abstracting with $\backslash I$. Exactly the same approach applies to narrow- and wide- scope modal auxiliaries such as "must" and "cannot".

## 6 Discussion and Conclusions

The familiar non-associative Lambek calculus with the conservative, traditional and yet novel semantic interpretation turns out capable of analyzing discontinuous constituency. Gapping, quantifier ambiguity and scope islands now fall within **NL**'s scope. The semantic interpretation is traditional in that it is compositional, assigning every (sub)derivation an always closed logical formula and using only the standard operations of the higher-order logic. The interpretation uses no dependent types, monads, effects or the continuation-passing style. The crucial feature is side-conditions to the semantic interpretation, to be combined with truth conditions at the end.

Our calculus shares many capabilities with the hybrid type-logical categorial grammar of Kubota and Levine K&L [2]. Both calculi analyze non-canonical coordination, gapping, and narrow and wide scopes of quantifiers and modal auxiliaries in coordinated structures. Our calculus uses no modes, type raising or higher-order phonology. Our coordination is always at type S and maintains phrase structure constituents.

The immediate future work is the treatment of summatives (with "total") and symmetric phrases (with the "same"). Our interpretation has the classical logic flavor, which is interesting to explore further.

**Acknowledgments.** I am very grateful to Yusuke Kubota for very helpful conversations and many suggestions.

*References*

[1] Kiselyov, O., Shan, C.c.: Continuation hierarchy and quantifier scope. In: McCready, E., Yabushita, K., Yoshimoto, K. (eds.) Formal Approaches to Semantics and Pragmatics, pp. 105–134. Studies in Linguistics and Philosophy, Springer Netherlands (2014), http://dx.doi.org/10.1007/978-94-017-8813-7_6
[2] Kubota, Y., Levine, R.: Gapping as like-category coordination. In: Béchet, D., Dikovsky, A. (eds.) Logical Aspects of Computational Linguistics: 7th International Conference. pp. 135–150. Springer (2012)
[3] Kubota, Y., Levine, R.: Empirical foundations for hybrid type-logical categorial grammar. the domain of phenomena (Aug 2013)
[4] Kubota, Y., Levine, R.: Gapping as hypothetical reasoning (2014)
[5] Morrill, G., Valentín, O., Fadda, M.: The displacement calculus. Journal of Logic, Language, and Information 20(1), 1–48 (2011), http://dx.doi.org/10.1007/s10849-010-9129-2
[6] Steedman, M.J.: Gapping as constituent coordination. Linguistics and Philosophy 13, 207–263 (1990)