A Frobenius Algebraic Analysis for Parasitic Gaps

Michael Moortgat
Utrecht University∗
m.j.moortgat@uu.nl

Mehrnoush Sadrzadeh
University College London†
m.sadrzadeh@ucl.ac.uk

Gijs Wijnholds
Utrecht University
g.j.wijnholds@uu.nl

Abstract

The interpretation of parasitic gaps is an ostensible case of non-linearity in natural language composition. Existing categorial analyses, both in the typelogical and in the combinatory traditions, rely on explicit forms of syntactic copying. We identify two types of parasitic gapping where the duplication of semantic content can be confined to the lexicon. Parasitic gaps in adjuncts are analysed as forms of generalized coordination with a polymorphic type schema for the head of the adjunct phrase. For parasitic gaps affecting arguments of the same predicate, the polymorphism is associated with the lexical item that introduces the primary gap. Our analysis is formulated in terms of Lambek calculus extended with structural control modalities. A compositional translation relates syntactic types and derivations to the interpreting compact closed category of finite dimensional vector spaces and linear maps with Frobenius algebras over it. When interpreted over the necessary semantic spaces, the Frobenius algebras provide the tools to model the proposed instances of lexical polymorphism.

∗The research of the alphabetically first and third author is supported by NWO grant 360-89-070 “A composition calculus for vector-based semantic modelling with a localization for Dutch”.
†The alphabetically second author acknowledges the support of Royal Society International Exchange Award IE161631.
1 Introduction

Natural languages present many patterns, at the sentence and at the discourse level, where an overt syntactic element provides the semantic content for one or more occurrences of elements that are not physically realized, or that have no meaning of their own; examples are long-distance dependencies in ‘movement’ constructions, ellipsis phenomena, anaphora. Parasitic gaps\(^1\) are a challenging case in point.

As the name suggests, a parasitic gap is felicitous only in the presence of a primary gap. The examples in (1) illustrate some relevant patterns.

\[
\begin{align*}
  a & \quad \text{papers that Bob rejected} \quad \text{(immediately)} \\
  b & \quad \text{Bob left without closing the window} \\
  c & \quad \text{*window that Bob left without closing} \\
  d & \quad \text{papers that Bob rejected without reading} \quad \text{(carefully)} \\
  e & \quad \text{security breach that a report about} \quad \text{in the NYT made} \quad \text{public}
\end{align*}
\]

The case of object relativisation in (a) has a single gap (indicated by _) for the unexpressed direct object of rejected. In categorial type logics, gaps have the status of hypotheses, introduced by a higher-order type. In Lambek’s [6] Syntactic Calculus, for example, the relative pronoun would be typed as \((n\setminus n)/(s/np)\). The complete relative clause then acts as a noun postmodifier \(n\setminus n\). The relative clause body Bob rejected _\(_\) is typed as \(s/np\), which means it needs a noun phrase hypothesis in order to compose a full sentence. Because the hypothesis occupies the direct object position, it is impossible to physically realize that object, as the ungrammaticality of *papers that Bob rejected the proposal shows. The Lambek type requires the hypothetical np to occur at the right periphery of the relative clause body — a restriction that we will lift in Section 2 to allow for phrase-internal hypotheses.

The relative clause in (d) has two gaps: the primary one is for the object of rejected as in (a); the secondary, parasitic gap (marked by \(_\)\(_\)) is the unexpressed object of reading. The parasitic gap occurs here in an adjunct: the verb phrase modifier without closing _. Such an adjunct by itself, is an island for extraction: the ungrammatical (c) shows that it is impossible for the relative pronoun to establish communication with a np hypothesis occurring within the adjunct phrase. Compare (c) with the gapless (b) which has the complete adjunct without closing (the window)\(_\)\(_n\).

Example (e) represents a different type of parasitic gapping where both the primary and the parasitic gap regard co-arguments of the same verb: the primary gap is the direct object of made public, the secondary gap occurs in the subject argument of this predicate.

We illustrated the adjunct and co-argument types of parasitic gapping in (1) with relative clause examples. Primary gaps can also be triggered in main or subordinate constituent

\(^1\)We refer to [2] for general background, and proposed analyses in a variety of grammatical frameworks.
question constructions, as in (2a, b), where *which papers* will carry the higher-order type initiating hypothetical reasoning. In the ‘passive infinitive’ case (2c), the higher-order type is associated with the adjective *hard*, which in this context could be typed as \( ap/(to\_inf/np) \). The adjective then selects for an incomplete to-infinitive missing a \( np \) hypothesis, the direct object in (2c). As with the relative clause example (1a), putting a physically realized \( np \) in the position occupied by the hypothesis leads to ungrammaticality. Again, as in (1), the primary gaps here open the possibility for parasitic gaps dependent on them. See (2d, e, f), where we also show some variation in the forms the adjunct can take.

\[
\begin{align*}
  a & \text{ which papers did Bob reject (immediately)} \\
  b & \text{ I know which papers Bob will reject (immediately)} \\
  c & \text{ this paper is hard to understand / *the proposal} \\
  d & \text{ which papers did Bob accept despite not liking / \( p \) (really)} \\
  e & \text{ I know which papers Bob will reject before even reading / \( p \) (cursorily)} \\
  f & \text{ this paper is easy to explain well after studying / \( p \) (thoroughly)}
\end{align*}
\]

To account for the duplication of semantic content in parasitic gap constructions, existing categorial analyses rely on explicit forms of syntactic copying. The CCG analysis of [19] rests on (a directional version of) the \( S \) combinator of Combinatory Logic; the type-logical account of [13, 14] adapts the ! modality of Linear Logic to implement a restricted form of the structural rule of Contraction. These syntactic devices are hard to control: the CCG version of the \( S \) rule is subject to non-logical side conditions; the attempts to properly constrain Contraction easily lead to undecidability as shown in [4].

Our aim in this paper is to explore lexical polymorphism as an alternative to syntactic copying. Lexical polymorphism is already an indispensable ingredient of the categorial toolbox, allowing for the analysis of generalized coordination in terms of a type schema for the polymorphic coordinators *and, but, etc.* Treating the adjunct phrases of (1d) and (2d, e, f) as forms of subordinating conjunction, we propose to similarly handle the adjunct type of parasitic gaps by means of a polymorphic type schema for the heads *without, despite, after, etc.* In the co-argument type of parasitic gapping (1e), a conjunctive interpretation is absent. In this case, a polymorphic type schema for the relative pronoun *that* allows us to generalize from the single gap instance (1b) to the multi-gap case (1e). To obtain the derived relative pronoun type from the basic assignment, we can rely on the same mechanisms that relate the basic type for *without* etc to the derived type needed for the parasitic gap examples.

Our analysis builds on the categorical Frobenius algebraic compositional distributional semantics of [16, 17], combined with a multimodal extension of Lambek calculus as the syntactic front end, as in [9]. Our analysis provides further evidence that Frobenius algebra is a powerful tool to model the internal dynamics of lexical semantics.
2 Syntax

2.1 The logic $\text{NL}\diamond$

The syntactic front end for our analysis is the type logic $\text{NL}\diamond$ of [10] which extends Lambek’s pure logic of residuation [7] with modalities for structural control. The formula language is given by the following grammar ($p$ atomic):

$$A, B ::= p \mid A \otimes B \mid A/B \mid A\backslash B \mid \diamond A \mid \square A$$  \hspace{1cm} (3)

In $\text{NL}\diamond$, types are assigned to phrases, not to strings as in the more familiar Syntactic Calculus of [6], or its pregroup version [8]. The tensor product $\otimes$ then is a non-associative, non-commutative operation for putting phrases together; it has adjoints $/$ and $\backslash$ expressing right and left incompleteness with respect to phrasal composition, as captured by the residuation inferences (4). In addition to the binary family $/\,\otimes\,\backslash$, the extended language has unary control modalities $\diamond, \square$ which again form a residuated pair with the inferences in (5).

$$A \rightarrow C/B \text{ iff } A \otimes B \rightarrow C \text{ iff } B \rightarrow A\backslash C$$  \hspace{1cm} (4)

$$\diamond A \rightarrow B \text{ iff } A \rightarrow \square B$$  \hspace{1cm} (5)

The modalities serve a double purpose, either licensing reordering or restructuring that would otherwise be forbidden, or blocking structural operations that otherwise would be applicable. To license rightward extraction, as found in English long-range dependencies, we use the postulates in [6]. Postulate $\alpha_\diamond$ is a controlled form of associativity: the $\diamond$ marking licenses a rotation of the tensor formula tree that leaves the order of the components $A, B, \diamond C$ unaffected. Postulate $\sigma_\diamond$ implements a form of controlled commutativity: here the internal structure of the tensor formula tree is unaffected, but the components $B$ and $\diamond C$ are exchanged.

$$\alpha_\diamond : (A \otimes B) \otimes \diamond C \rightarrow A \otimes (B \otimes \diamond C)$$

$$\sigma_\diamond : (A \otimes B) \otimes \diamond C \rightarrow (A \otimes \diamond C) \otimes B$$  \hspace{1cm} (6)

To block these structural operations from applying, we use a pair of modalities $\diamond, \square$. Phrases that qualify as syntactic islands are marked off by $\diamond$. The modal island demarcation makes sure that the input conditions for $\alpha_\diamond, \sigma_\diamond$ do not arise. The island markers $\diamond, \square$ have no associated structural rules; their logical behaviour is fully characterized by (5).

$\text{NL}\diamond$ derivations will be represented using the axiomatisation of Figure 1 due to Došen [3]. This axiomatisation takes (Co)Evaluation as primitive arrows, and recursively generalizes these by means of Monotonicity. It is routine to show that the residuation inferences of (4) and (5) become derivable rules given the axiomatisation of Figure 1. To streamline derivations, we will make use of the derived residuation steps. Also, we will freely use (Co)Evaluation and the structural postulates (6) in their rule form, by composing them with Transitivity ($\circ$).
\[
\begin{array}{c}
1_A : A \to A \\
f : A \to B \quad g : B \to C \\
f \circ g : A \to C \\
f : A \to B \quad g : C \to D \\
f \otimes g : A \otimes C \to B \otimes D \\
f / g : A / D \to B / C \\
\Box f : \Box A \to \Box B \\
\diamond f : \diamond A \to \diamond B \\
ev_{A,B} : A \otimes A / B \to B \\
\text{co-ev}^\backslash_{A,B} : B \backslash (A \otimes B) \to A \\
\sigma_{\Box} : (A \otimes B) \otimes \Box C \to (A \otimes C) \otimes B \\
\alpha_{\diamond} : (A \otimes B) \otimes \diamond C \to A \otimes (B \otimes \diamond C) \\
\end{array}
\]

Figure 1: Došen style axiomatisation of NL\(\Box\).

2.2 Graphical calculus for NL\(\diamond\)

Wijnholds [20] gives a coherent diagrammatic language for the non-associative Lambek Calculus NL; the generalisation to NL with control modalities is straightforward, see Figure 2. In short, each connective is assigned two links that either compose or decompose a type built with that connective. Links (and diagrams) can be put together granted that their in- and outputs coincide. This system has a full recursive definition, and is shown to be sound and complete (i.e. coherent) with respect to the categorical formulation of the Lambek Calculus, given a suitable set of graphical equalities (not discussed in the current paper).

As an illustration, we present the derivation of the simple relative clause example (1a) in symbolic and diagrammatic form. For this case of non-subject relativisation, the relative pronoun that is typed as a functor that produces a noun modifier \(n\backslash n\) in combination with a sentence that contains an unexpressed np hypothesis (Bob rejected immediately). The subtype for the gap is the modally decorated formula \(\Diamond \Box np\). The \(\Diamond\) marking allows it to cross phrase boundaries on its way to the phrase-internal position adjacent to the transitive

Supplementary Note: Subject relative clauses, e.g. paper that irritates Bob, do not involve any structural reasoning. The relative pronoun for subject relatives can be typed simply as \((n\backslash n)/(np)\).
Figure 2: Došen style axiomisation of \textbf{NL}_\otimes with diagrams. Monotonicity and (co)evaluation laws for / are fully symmetrical to the given diagrams for \.

\begin{itemize}
\item \textbf{Identity}
\item \textbf{Composition}
\item \textbf{⊗ Monotonicity}
\item \textbf{\ Monotonicity}
\item \textbf{\ Evaluation}
\item \textbf{\ Co-evaluation}
\item \textbf{♦ Monotonicity}
\item \textbf{□ Evaluation}
\item \textbf{□ Co-evaluation}
\item \textbf{Controlled associativity} α
\item \textbf{Controlled commutativity} σ
\end{itemize}
verb rejected. At that point, the licensing $\Diamond$ has done its work, and can be disposed of by means of the $ev^{\Box}$ axiom $\Diamond np \rightarrow np$, which provides the np object required by the transitive verb rejected. For legibility, we use words instead of their types for the lexical assumptions in the derivation below. The steps labeled $\ell$ indicate the lexical look-up.

$$
\frac{\text{rejected}}{\ell} \frac{\Diamond \Box np \rightarrow np}{ev^{\Box}} \frac{\text{immediately}}{\ell} \frac{(np \backslash s) \backslash (np \backslash s)}{ev^\backslash}
$$

$$
\frac{\text{Bob} \times ((\text{rejected} \times \Diamond \Box np) \times \text{immediately})}{\ell} \frac{\rightarrow s}{ev^\backslash}
\frac{\text{Bob} \times ((\text{rejected} \times \text{immediately}) \times \Diamond \Box np) \rightarrow s}{\sigma \diamond}
\frac{\rightarrow s \text{ res}}{ev^\backslash}
\frac{\text{Bob} \times ((\text{rejected} \times \text{immediately})) \rightarrow s / \Diamond np}{\ell}
\frac{\rightarrow \text{ paper} \times (\text{that} \times (\text{Bob} \times (\text{rejected} \times \text{immediately})))}{ev^\backslash}
\frac{\rightarrow n \backslash n}{\ell}
\frac{\text{paper} \times (\text{that} \times (\text{Bob} \times (\text{rejected} \times \text{immediately}))))}{\rightarrow n}
$$

(7)

Figure 3: Diagrammatic form of Paper that Bob rejected immediately.

In the diagrammatic form of Fig 3 the $\Diamond \Box np$ gap hypothesis is indicated by the corresponding links. The leading $\Diamond$ link licenses the crossing over to the object position of rejected by means of the $\sigma \diamond$ postulate of Fig 2. In what follows, we use diagrams for $\text{NL}_\omega$ derivations because this format pictures the information flow in a simple and intuitive way.
2.3 Typing Parasitic Gaps

**Lexical polymorphism: generalized coordination** As our account of parasitic gaps in adjuncts treats the adjuncts as a form of subordinate conjunction, we briefly review how lexical polymorphism is used in the analysis of generalized coordination.

Chameleon words such as *and, but* cannot easily be typed monomorphically: given an initial type and interpretation, say $(s \setminus s)/s$ for sentence coordination, we’d like to be able to obtain derived types and interpretations for the coordination of (in)transitive verbs, as in (8b, c), or for non-constituent coordination cases such as (8d).

```
a  (Alice sings)$_s$ and (Bob dances)$_s$
b  Alice (sings and dances)$_{np\setminus s}$
c  Bob (criticized and rejected)$_{(np\setminus s)/np}$ the paper
```

Deriving the ($b$–$d$) types from an initial $(s \setminus s)/s$ assignment, however, goes beyond linearity. The attempt in (9) to derive verb phrase coordination from sentence coordination requires a copying step to strongly distribute the final $np$ abstraction over the two conjuncts.

```
\[
\begin{align*}
\[(np\setminus np\setminus s) \otimes ((s\setminus s)/s \otimes (np\setminus np\setminus s)) & \longrightarrow s \\
(np\setminus s \otimes (np\setminus s)/s \otimes np\setminus s) & \longrightarrow s \\
(s\setminus s)/s & \longrightarrow ((np\setminus s)(np\setminus s))/([np\setminus s])
\end{align*}
\]
Copy!
```

Partee and Rooth’s [15] work on generalized coordination offers a method for replacing syntactic copying by lexical polymorphism. Coordinating expressions *and, but* get a polymorphic type assignment $(X\setminus X)/X$ where $X$ is a conjoinable type. The set of conjoinable types $ CType$ forms a subset of the general set of types $ Type$. $ CType$ is defined inductively:

- $ s \in CType$
- $ A\setminus B, B/A \in CType$ if $B \in CType$, $A \in Type$

The type polymorphism comes with a generalized interpretation. We write $\sqcap^X$ (infix notation) for a coordinator of (semantic) type $X \rightarrow X \rightarrow X$.

- $ P \sqcap^t Q := P \land Q$ coordination in type $t$ amounts to boolean conjunction
- $ P\sqcap^{A\rightarrow B} Q := \lambda x^A. (P x)\sqcap^B (Q x)$ distributing the $x^A$ parameter over the conjuncts

---

3Partee and Rooth formulate this in terms of the semantic types obtained from the syntax-semantics homomorphism $h$, with $h(s) = t$ (the type of truth values), $h(np) = e$ (individuals) and $h(A\setminus B) = h(B/A) = h(A) \rightarrow h(B)$. 
The generalized interpretation scheme, then, associates a type transition such as (9) with the Curry-Howard program that would be associated with a derivation involving the copying step. In Section §3 we will obtain the same effect using the Frobenius algebras over our vector-based interpretations.

**Parasitic gaps in adjuncts** Consider the type lexicon for the data in $1a–d4$.

papers, window :: $n$
that :: $(n\backslash n)/(s/\Diamond \Box np)$
Bob :: $np$
rejected :: $(np\backslash s)/np$
reading, closing :: $gp/np$
immediately, carefully :: $iv/iv$
without :: $\Diamond (X\backslash Y)/Z$ (schematic)
without$^{b,c}$ :: $\Diamond (iv/iv)/gp$
without$^{d}$ :: $\Diamond ((iv/\Diamond \Box np)\backslash (iv/np))/(gp/\Diamond \Box np)$

The gap-less example (1b) provides the motivation for the basic type assignment to without as a functor combining with a non-finite gerund clause $gp$ to produce a verb-phrase modifier $iv\backslash iv$. To impose island constraints, we use a pair of modalities $\Diamond$, $\Box$. In order to block the ungrammatical (1c), we follow [11] and lock the $iv\backslash iv$ result type with $\Box$; the matching $\Diamond$ needed to unlock it has the effect of demarcating the modifier phrase without closing the window as an island, represented in the diagram below by means of a dotted line.

An attempt to derive the ungrammatical window that Bob left without closing $\Box$ fails. The derivation proceeds like the one above, but with the gap hypothesis $\Diamond \Box np$ in the place of the window. At that point the $\Diamond$ island demarcation of without closing $\Diamond \Box np$ makes $iv$ abbreviates $np\backslash s$; $gp$ stands for gerund clause, headed by the -ing form of the verb.
it impossible to bring out the hypothesis to the position where it can be withdrawn. This becomes apparent diagrammatically as the gap hypothesis cannot cross the dotted line:

Let us turn then to the adjunct parasitic gapping of (1d). To account for the double use of the gap we replace syntactic copying via controlled Contraction by lexical polymorphism, treating without as a polymorphic item on a par with coordinators and, but. That means we assign to without the following type schema

\[
\text{without} :: (X \backslash Y) / Z
\]

with basic instantiation \(X = Y = iv, Z = gp\). From this basic instantiation, a derived instantiation with \(X = Y = iv / \Diamond \Box np\) and \(Z = gp / \Diamond \Box np\) is obtained for the parasitic gapping example (1d) by uniformly dividing the subtypes \(iv\) and \(gp\) by \(\Diamond \Box np\) using the forward slash.

In Section §3 we will see how the vector-based interpretation of the derived type is obtained in a systematic fashion from the interpretation of the basic type instantiation. For this, it is helpful to factorize the construction of the derived type as the combination of an expansion step and a distribution step. Ignoring the appropriate \(\Box\) decoration to mark off the adjunct as an island, the expansion step here is an instance of the Geach transformation.
A/B \rightarrow (A/C)/(B/C), with A = iv\(\backslash iv\), B = gp, C = \(\lozenge \Box np\).

(basic type) \hspace{1cm} \Box(iv\(\backslash iv\))/gp

\hspace{1cm} \downarrow \hspace{1cm} \text{expand}

\hspace{1cm} (\Box(iv\(\backslash iv\))/\(\lozenge \Box np\))/(gp/\(\lozenge \Box np\))

\hspace{1cm} \downarrow \hspace{1cm} \text{distribute}

(derived type) \hspace{1cm} \Box((iv/\(\lozenge \Box np\))(iv/\(\lozenge \Box np\)))/(gp/\(\lozenge \Box np\))

Setting now A = iv, B = iv, C = \(\lozenge \Box np\), the distribution step is a directional instance of the S combinator \((A\backslash B)/C \rightarrow (A/C)/(B/C)\).

To arrive at the version of the derived type for without as we have it in our lexicon (10), a final calibration is required. We replace the result type \(iv/\Box np\) by \(iv/np\), dropping the modal marking required for controlled associativity/commutativity. The final type \(\Box((iv/\Box np)(iv/np))/(gp/\Box np)\) allows for the derivation of the parasitic gapping example (1c) displayed in Figure 4 but also for cases of Right Node Raising such as

Bob (rejected without reading)\(iv/np\) all papers about linguistics

where all papers about linguistics is a plain np rather than \(\lozenge \Box np\).

Parasitic gaps: co-arguments

(1e) security breach that a report about \(\lhd np\) in the NYT made \(\lhd np\) public

For the co-argument type of parasitic gapping (1e), repeated here for convenience, the relative clause body does not contain a conjunction-like element that would be a suitable candidate to lexically encapsulate the ostensible copying. But we can turn to the relative pronoun itself, and use the mechanisms we relied on for the lexical polymorphism of without to move from the relative pronoun’s basic type assignment for single-gap dependencies to a derived assignment for the double-gap dependency of (1e).

\[ \text{that}^{a,c} :: (n\backslash n)/(s/\Box np) \]
\[ \text{that}^{e} :: (n\backslash n)/(np/\Box np) \otimes ((np\backslash s)/\Box np) \] (11)

Again, we see that these types are derivable from the initial type for that by a combina-
The image contains a diagram illustrating information flow for the double parasitic gap. The text accompanying the diagram explains the expansion and distribution steps in the context of logical formulas and rules.

The expansion step replaces $s$ in antitone position by $np \otimes np \setminus s$, justified by leftward Application $\text{ev} \downarrow : np \otimes np \setminus s \rightarrow s$ and Monotonicity. Here, with $A = np \otimes np \setminus s$, $B = s$, $C = \Diamond \Box np$ and $D = n \setminus n$, we have

$$
\dfrac{A \rightarrow B}{A/C \rightarrow B/C} \quad \text{Appl} \\
\dfrac{A/C \rightarrow B/C}{D / (B/C) \rightarrow D / (A/C)} \quad \text{Mon}^\dagger
$$

Likewise, the distribution step relies on $\text{Mon}^\dagger$ to replace $(A \otimes B) / C$ by $A / C \otimes B / C$ in antitone position. Here, with $A = np$, $B = np \setminus s$, $C = \Diamond \Box np$, $D = n \setminus n$, we have
\[
\begin{array}{c}
\vdash \\
(A/C \otimes C) \otimes (B/C \otimes C) \rightarrow A \otimes B \\
(A/C \otimes B/C) \otimes C \rightarrow A \otimes B \\
A/C \otimes B/C \rightarrow (A \otimes B)/C \\
D/((A \otimes B)/C) \rightarrow D/(A/C \otimes B/C)
\end{array}
\]

Distr Res Mon↑

Figure 5 has the derivation for example (1).

## 3 Frobenius Semantics

The proposed vector-based semantics has two ingredients: first, the *derivational* semantics specifies a compositional mapping that interprets types and proofs of the NL♦ syntax as morphisms of a Compact Closed Category, concretely the category of FVect and linear maps. Second, the *lexical* semantics specifies the word-internal interpretation of individual lexical items; here, we make use of the Frobenius Algebras over FVect to model the copying of semantic content associated with the interpretation of relative pronouns such as *that*, and modifier heads such as *without*.

### 3.1 Diagrams for Compact Closed Categories and Frobenius Algebras

Recall that a Compact Closed Category is a symmetric monoidal category \((C, \otimes, I)\) with duals \(A^*\) for every object \(A\), and contraction and expansion maps for every object. In the case of vector spaces over fixed bases (our concrete semantics) we don’t distinguish between objects and their duals, hence the contraction and expansion maps have signature \(\epsilon : V \otimes V \rightarrow I\) and \(\eta : I \rightarrow V \otimes V\), respectively.

For compact closed categories, there is a complete diagrammatic language available, that uses *cups* and *caps* to represent contraction and expansion, see [18]. These are drawn as connecting two objects either as a cup in the case of \(\epsilon\) or as a cap in the case of \(\eta\). The standard contraction and expansion maps of a CCC form the basis for interpreting derivations of NL♦.

Crucial to our polymorphic approach is the inclusion of Frobenius Algebras in the lexicon. A Frobenius algebra in a symmetric monoidal category \((C, \otimes, I)\) is a tuple \((X, \Delta, \iota, \mu, \zeta)\) where, for \(X\) an object of \(C\), the first triple below is an internal comonoid and the second one is an internal monoid.

\[(X, \Delta, \iota) \quad (X, \mu, \zeta)\]
privacy breach
that
a report
about
in NYT
made
public

Figure 5: Co-argument parasitic gapping.
This means that we have a coassociative map $\Delta$ and and its counit $\iota$:  

$$ \Delta : X \to X \otimes X \quad \iota : X \to I $$

and an associative map $\mu$ and its unit $\zeta$:  

$$ \mu : X \otimes X \to X \quad \zeta : I \to X $$

as morphisms of our category $C$. The $\Delta$ and $\mu$ morphisms satisfy the *Frobenius condition* given below

$$ (\mu \otimes 1_X) \circ (1_X \otimes \Delta) = \Delta \circ \mu = (1_X \otimes \mu) \circ (\Delta \otimes 1_X) $$

Informally, the comultiplication $\Delta$ decomposes the information contained in one object into two objects; the multiplication $\mu$ combines the information of two objects into one. In diagrammatic terms, to visualise the Frobenius operations one adds a white triangle to the diagrammatic language for CCCs that represents the (un)merging of information through the four different Frobenius maps. The resulting graphical language is summarised in Figure 6. 
3.2 Derivational Semantics

For the derivational semantics, we need to define a homomorphism \([\cdot]\) that sends syntactic types and derivations to the corresponding components of the Compact Closed Category of \(\mathbf{FVect}\) and linear maps. This homomorphism has been worked out by Moortgat and Wijnholds [9]. We present the key ingredients below and refer the reader to that paper for full details.

**Types** The target signature has atomic semantic spaces \(N\) and \(S\), an involutive \((\cdot)^*\) for dual spaces and a symmetric monoidal product \(\otimes\). We set

\[
\begin{align*}
[s] & = S, \\
[np] & = [n] = N, \\
[ap] & = [gp] = N^* \otimes S, \\
[\Diamond A] & = [\Box A] = [A], \\
[A/B] & = [A] \otimes [B]^*, \\
[A\setminus B] & = [A]^* \otimes [B]
\end{align*}
\]

Notice that \(ap\) and \(gp\) are mapped to \(N^* \otimes S\). Their understood subject is provided by the context: the main clause subject, in the case of *Bob fell asleep while watching TV*, the direct object in the case of *they made the report public*.

**Derivations** The instances of the Evaluation axioms correspond to generalised contraction operations on vector spaces, the instances of the Co-Evaluation axioms dually are mapped to generalised expansion maps. The structural control postulates stipulate a syntactically limited associativity and commutativity; since the control modalities leave no trace on the semantic interpretation, the structural postulates \(\alpha_\Diamond\) and \(\sigma_\Box\) are interpreted using the standard associativity and symmetry maps of \(\mathbf{FVect}\).

The derivational semantics is represented graphically in Figure 7, where the diagrams of Figure 2 are interpreted in the complete diagrammatic language of compact closed categories of Figure 6.

Under the given interpretation, the diagrammatic derivation of Figure 4 for (1)

\[
\begin{align*}
\text{papers} & \quad \text{that} & \quad \text{Bob} & \quad \text{rejected} & \quad \text{without} & \quad \text{reading} & \quad n \\
n & \quad (n\setminus n)/(s/\Diamond np) & \quad np & \quad (np\setminus s)/np & \quad (\Box (X\setminus Y))/Z & \quad gp np & \rightarrow n
\end{align*}
\]

is sent to the contractions in the interpreting CCC in Figure 8 (red: \([\cdot]\), blue: \([\text{without}]\)).
3.3 Lexical Semantics

For the lexical interpretation of the relative pronoun that and the conjunctive without, we follow previous work [16, 17] and use Frobenius algebras that characterise vector space bases [1]. First, the basic form of the diagram for that is as developed in [16]. The basic diagram for without uses a double instance of a Frobenius Algebra to coordinate the gerundive phrase with the intransitive verb phrase consumed to its left. Recall that the interpretation homomorphism sends np\,s and gp to the same semantic space, $N^* \otimes S$. In Figure 9 we display graphically these basic types as well as how their derived instantiations look.

For the case of parasitic gaps in adjunct positions we use the basic type for that and the
derived type for \textit{without}. For \textit{that}, its basic Frobenius instantiation has the concrete effect of projecting down the verb phrase into a vector which is consecutively multiplied element-wise with the head noun of the main clause. The diagram for \textit{without} then makes sure to distribute the missing hypothesis of the relative clause over the two gaps in the clause body. Given the identification $[iv] = [gp]$, this is essentially the treatment of coordination of [5].

For the co-argument case, we need make use of the derived type for \textit{that}; its function is now to both specify the need for a clause body missing a hypothetical noun phrase, as well as coordinating this noun phrase through two gaps. Hence, the derived instantiation figures an iterative use of the Frobenius $\mu$ to merge three elements together.

With both the derivational semantics of Figure 8 and the lexical specifications of the constituents of Figure 9 we can put everything together to get the (unnormalised) diagram in Figure 10.

This diagram can be normalised under the equations of the diagrammatic language, leading to the normal form of Figure 11.

The above diagrams are morphisms of a symmetric compact closed category with Frobenius algebras and can be written down in that language as done e.g. in [16, 9]. Here, we provide the closed linear algebraic form of the normal form in Figure 11. For Rejected and Not-Reading the rank 3 tensors interpreting rejected and (without) reading, and $\iota$ the unit of the Frobenius coalgebra, this is

$$\text{Papers} \odot (\iota_S \otimes \text{id}_N)(\text{Bob}^T \times (\text{Rejected} \odot \text{Not-Reading})))$$

The closed linear algebraic form says that we take the elementwise multiplication of both cubes, and contract them with the subject \textit{Bob}; then, we collapse the resulting matrix into a vector and compute the elementwise multiplication of this vector with the vector interpreting the head noun \textit{Papers}.

For the co-argument case of parasitic gapping, we insert the derived Frobenius diagram for \textit{that}, to obtain the initial diagram of Figure 12 which normalises to the diagram in Figure 13.
Figure 9: Deriving the lexical semantics for *without* and *that*.
Figure 10: Semantic information flow for the double parasitic gap (initial form).

Figure 11: Semantic information flow for the double parasitic gap (normal form).

4 Discussion

The concrete modelling presented above produces an interpretation of relative clauses that is analogous to the formal semantics account: seeing elementwise multiplication as an intersective operation (cf. set intersection), the interpretation of \textit{papers that Bob rejected without reading} identifies those papers that were both rejected and not reviewed, by Bob.

In the formal semantics account, the head noun and the relative clause body are both interpreted as functions from individuals to truth values, i.e. characteristic functions of sets of individuals, which allows them to be combined by set intersection. In our vector-based modelling, however, the head noun and the relative clause body are initially sent to different semantic spaces, viz. \(N\) for the head noun versus \(N \otimes S\) for the relative clause body. This means we need to appeal to the \(\iota\) operation to effectuate the rank reduction from \(N \otimes S\) to \(N\) that reduces the interpretation of the relative clause body to a vector that can then
privacy breach
that
a
report
about
in
NYT
made
public

Figure 12: Semantic information flow for the co-argument parasitic gap (initial form).
be conjoined with the meaning of the head noun. The rank reduction performed by the $t$ transformation is not a lossless transformation, and it is debatable whether it correctly captures the semantic action we want to associate with the relative pronoun.

As a first step towards a more general model, we abstract away from the specific modelling of the relative pronoun by means of the $t$ map.

As shown in Figure 14, our type translation for the relative pronoun effectively interprets it as a map from a verb phrase $(N \otimes S)$ meaning into an adjectival meaning modifying a (common) noun $(N \otimes N)$.

With this generalization, we are not bound anymore to a specific implementation of the relative pronoun meaning, although the proposed account for now gives a workable solution for experimentation.

We suggest here, that a data-driven approach may lend itself for modelling the relative pronoun, as it essentially binds a verb phrase to its adjectival form. For example, a verb phrase can occur in adjectival form, e.g. “papers that were rejected” vs “rejected papers”. In such cases, we would expect to get the same meaning representation, which crucially relies on being able to project either an adjective onto a verb phrase or vice versa. Formulating this as a machine learning problem, is work in progress.

5 Conclusion/Future Work

We presented a typological ditributional account of parasitic gapping, one of the many linguistic phenomena in which some semantic elements are not present in the sentence (or
more generally discourse) and therefore their corresponding information needs to be provided from some other syntactic element. Rather than relying on some form of copying and/or movement on the syntax side to provide this information, we have solved the problem by using polymorphic typing for function words that play a key role in parasitic gapping (here, *that* and *without*).

The polymorphism carries over to the semantics, where we have used Frobenius algebras to interpret them. This enabled us to handle the coordination of multiple gaps, and where the relative pronoun *that* handles the coordination of the head noun with the body of the relative clause and the pronoun *without* coordinates the second gap that exists in the body and which refers to the same head noun.

We discussed a more general normal form in which the behaviour of the relative pronoun is kept abstract. Investigating alternatives to the current modelling with the $\iota$ map, and looking into data-driven modelling of the relative pronoun, constitutes work in progress.

**References**

[1] Coecke, B., Pavlovic, D., Vicary, J.: A new description of orthogonal bases. Mathematical Structures in Computer Science 1, 269–272 (2008)

[2] Culicover, P., Postal, P. (eds.): Parasitic Gaps. MIT Press (2001)
[3] Došen, K.: A brief survey of frames for the Lambek calculus. Mathematical Logic Quarterly 38(1), 179–187 (1992)

[4] Kanovich, M.I., Kuznetsov, S., Scedrov, A.: Undecidability of a newly proposed calculus for CatLog3. In: Bernardi, R., Kobele, G., Pogodalla, S. (eds.) Formal Grammar - 24th International Conference, FG 2019, Riga, Latvia, August 11, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11668, pp. 67–83. Springer (2019)

[5] Kartsaklis, D.: Coordination in categorical compositional distributional semantics. In: Proceedings of the 2016 Workshop on Semantic Spaces at the Intersection of NLP, Physics and Cognitive Science, SLP@QPL 2016, Glasgow, Scotland, 11 th June 2016. pp. 29–38 (2016)

[6] Lambek, J.: The mathematics of sentence structure. American Mathematical Monthly 65, 154–170 (1958)

[7] Lambek, J.: On the calculus of syntactic types. In: Jakobson, R. (ed.) Structure of Language and its Mathematical Aspects, Proceedings of Symposia in Applied Mathematics, vol. XII, pp. 166–178. American Mathematical Society (1961)

[8] Lambek, J.: Type grammar revisited. In: Lecomte, A., Lamarche, F., Perrier, G. (eds.) Logical Aspects of Computational Linguistics, Second International Conference, LACL ’97, Nancy, France, September 22-24, 1997, Selected Papers. Lecture Notes in Computer Science, vol. 1582, pp. 1–27. Springer (1997)

[9] Moortgat, M., Wijnholds, G.: Lexical and derivational meaning in vector-based models of relativisation. In: Proceedings of the 21st Amsterdam Colloquium. p. 55. ILLC, University of Amsterdam (2017), https://arxiv.org/abs/1711.11513

[10] Moortgat, M.: Multimodal linguistic inference. Journal of Logic, Language and Information 5(3-4), 349–385 (1996)

[11] Morrill, G.: Type Logical Grammar: Categorial Logic of Signs. Kluwer Academic Publishers, Dordrecht (1994)

[12] Morrill, G.: Parsing/theorem-proving for logical grammar CatLog3. Journal of Logic, Language and Information 28(2), 183–216 (2019)

[13] Morrill, G., Valentín, O.: Computational coverage of TLG: Nonlinearity. In: Proceedings of Third Workshop on Natural Language and Computer Science. vol. 32, pp. 51–63. EasyChair Publications (2015)

[14] Morrill, G., Valentín, O.: On the logic of expansion in natural language. In: Logical Aspects of Computational Linguistics. pp. 228–246. Springer (2016)

[15] Partee, B., Root, M.: Generalized conjunction and type ambiguity. In: Bäuerle, R., Schwarze, C., von Stechow, A. (eds.) Meaning, Use, and Interpretation of Language, pp. 361–383. Walter de Gruyter, Berlin (1983)

[16] Sadrzadeh, M., Clark, S., Coecke, B.: The Frobenius anatomy of word meanings I: subject and object relative pronouns. Journal of Logic and Computation 23(6), 1293–1317 (2013)

[17] Sadrzadeh, M., Clark, S., Coecke, B.: Frobenius anatomy of word meanings 2: possessive relative pronouns. Journal of Logic and Computation 26, 785–815 (2014)

[18] Selinger, P.: A survey of graphical languages for monoidal categories. In: Coecke, B. (ed.) New Structures for Physics, Lecture Notes in Physics, vol. 813, pp. 289–355. Springer, Berlin,
Heidelberg (2011)

[19] Steedman, M.: Combinatory grammars and parasitic gaps. Natural Language and Linguistic Theory 5, 403–439 (1987)

[20] Wijnholds, G.J.: Coherent diagrammatic reasoning in compositional distributional semantics. In: International Workshop on Logic, Language, Information, and Computation. pp. 371–386. Springer (2017)