FUNCTIONAL PARASITIC GAPS

Benjamin Russell, Brown University

Functional readings in parasitic gap (PG) constructions are limited as follows: the argument of the functional PG must be bound by the same quantifier as the matrix gap. To illustrate, in the PG construction (1-a), the president must have offended every little boy's mother, not his own, in contrast to the across-the-board extraction in (1-b), where two binders are permitted.

(1) a. Which relative did every little boy hug _ after the president offended _ ?
for which f : every little boy x hugged $f(x)$ after the president y offended $f(x)$, *$f(y)$

b. Which relative did every little boy hug _ and the president kiss _ ?
for which f : every little boy x hugged $f(x)$ and the president y kissed $f(x)$, $f(y)$

Munn (2001) argues that the no-two-binders generalization for (1-a) is a consequence of PGs' semantic type: he proposes that they must be individual-denoting, or variables of type e. Under this hypothesis, the functioanl PG data are explained by the same principle that causes non-NP PGs to be ungrammatical (Engdahl 1983, Postal 1993), and both generalizations are connected to the Weak Island theory of Szabolcsi and Zwarts (1997). But Munn does not develop a compositional account of this hypothesis. In this paper, I argue that Munn's proposal cannot be implemented in a compositional system that relies on variables, movement-induced predicate abstraction, and predicate modification to compose PG constructions. I show that Munn's idea may, however, be naturally implemented using a variable-free semantics (VFS) account of binding (Jacobson 1999) combined with Steedman's (1987) combinatory categorial grammar (CCG) theory of PGs. This paper makes two contributions: it provides an empirical analysis of functional parasitic gaps, and it provides evidence for the VFS theory of binding over the variable-based approach.

The only attempt at a compositional analysis of PG constructions that makes use of a standard semantics with variables is developed in Nissenbaum (2000), who uses a Minimalist syntax and Heim and Kratzer's (1998) semantics. The main challenge in composing PG constructions like (1-a) is to combine the VP "hug" with the adjunct "after the president offended". Nissenbaum composes these with Predicate Modification because this is apparently the only way to ensure coreference of the two variables, which are not guaranteed to be coindexed because the traces are not in a chain together (following Chomsky (1986)). In Nissenbaum's system, movement of the object of "hug" out of the VP induces lambda abstraction over that VP, making it a predicate of type $\langle et \rangle$; the movement of the null operator object of "offend" within the parasitic adjunct makes *that* into a predicate of type $\langle et \rangle$; these two predicates combine through predicate modification, identifying the semantics of the objects of the verbs.

Nissenbaum does not provide an analysis of functional PGs. Because Munn uses Chierchia's (1993) version of Engdahl's (1986) theory of functional extraction, I implement this theory in Nissenbaum's system. Under the Engdahl/Chierchia system, functional traces have two indices: the index of the function and the index of its argument. In simple functional questions, the variable representing the argument of the function is bound by a quantifier, and movement causes the function variable itself to be abstracted over, forming the question. Crucially, functional traces in this system are the same semantic type as regular traces: type e. In PG constructions, the matrix gap must be a doubly-indexed functional gap, but there is some uncertainty as to the semantics of the PG. There are two possibilities:

(2) a. The PG trace, like the VP trace, can be a doubly-indexed functional variable. Then movement induces lambda abstraction over the functional variable, making the adjunct predicate type $\langle ee, t \rangle$. This predicate can combine with the VP via Predicate Modification. But the argument variable is then available to be bound by an Adjunct-internal quantifier, and nothing prevents the two-binders reading.

b. Otherwise, the PG trace must be a singly-indexed variable. Then, when this moves, the abstracted parasitic adjunct is a predicate over individuals. But that makes the Adjunct type $\langle et \rangle$, whereas the VP is type $\langle ee, t\rangle$. These cannot combine by any of the rules in Nissenbaum's theory. Therefore, the semantics needs a new rule for combining terms of type $\langle ee, t\rangle$ with those of type $\langle et \rangle$.

Both possibilities lead to major difficulties; one is empirical, the other theoretical. Furthermore, the restriction in (2-b) would apparently have to be expressed syntactically because there is no difference in semantic type between a function of type $\langle ee \rangle$ applied to an individual and an individual. So Munn's explanation of the functional PG restriction would therefore be a syntactic solution, not a semantic one as he claims.

My account of functional PGs relies on a categorial grammar with two particular combinators: the **S** combinator adopted by Steedman for PGs and the **z** combinator developed by Jacobson for binding. A definition of each in terms of the ⟨syntax, semantics⟩ of expressions is given in (3).

(3) $\mathbf{S} : \langle (\alpha/\beta)/\gamma,\ f_{\langle e,\langle \tau, \upsilon\rangle\rangle}\rangle \mapsto \langle (\alpha/\gamma)/(\beta/\gamma),\ \lambda g \lambda x_e [f(x)(g(x))]\rangle$
$\mathbf{z} : \langle (\alpha/\beta)/\gamma,\ f\rangle \mapsto \langle (\alpha/\beta)/\gamma^{\beta},\ \lambda g \lambda h [f(g(h))(h)]\rangle$

Notice that the **S** combinator does not apply generally to expressions of arbitrary type $\langle \sigma, \langle \tau, \upsilon\rangle\rangle$; it applies only to functions whose first argument is type e. This restriction is a relatively direct implementation of Munn's hypothesis that PGs are type e, but it differs from Munn's proposal in that PGs are here not represented with variables in the semantics. Examples of the application of **z** and **S** follow.

(4) a.
$$\dfrac{\begin{array}{c}\text{hug}\\ \mathrm{VP/NP}\\ [\![\textbf{hug}]\!]\end{array}}{\begin{array}{c}\mathrm{VP/NP^{NP}}\\ \lambda f \lambda x[[\![\textbf{hug}]\!](f(x))(x)]\end{array}}\ \mathbf{z}$$

b.
$$\dfrac{\begin{array}{c}\text{after the president offended}\\ \mathrm{(VP\backslash VP)/NP}\\ \lambda x \lambda P \lambda y[P(y) \succ [\![\textbf{offend}]\!](x)(b)]\end{array}}{\begin{array}{c}\mathrm{(VP/NP)\backslash(VP/NP)}\\ \lambda Q \lambda x \lambda y[Q(x)(y) \succ [\![\textbf{offend}]\!](x)(b)]\end{array}}\ \mathbf{S}$$

S constructs PGs by allowing expressions of type $\langle e, \langle et, et\rangle\rangle$, functions from individuals to predicate modifiers, to shift to type $\langle\langle e, et\rangle, \langle e, et\rangle\rangle$. So incomplete adjuncts can "modify" incomplete predicates. **z** accounts for binding by allowing a verb to take a functional rather than an individual object and identifying the semantics of the argument of that function with the subject of the verb.

Given these two definitions, the no-two-binders generalization follows automatically. To bind each gap with a different subject, the **z** combinator would have to apply to both verbs: the one in the VP and the one in the adjunct. This would then change the types of the VP and the adjunct to $\langle ee, et\rangle$ and $\langle ee, \langle et, et\rangle\rangle$, respectively. And because of the type e restriction on the **S** combinator, these cannot combine. But the system can construct the desired single-binder functional PGs by applying **S** before applying **z**. **S** combines the Adjunct with the VP, yielding a complex transitive verb phrase "hug after the president offended", which is type $\langle e, et\rangle$. In this expression, the **S** combinator has identified the semantics of the object of hug with the object of offend; when this expression undergoes **z**, it therefore binds both gaps simultaneously.

References

Chierchia, G.: 1993, Questions with quantifiers, *Natural Language Semantics* **1**(2), 181–234.
Chomsky, N.: 1986, *Barriers*, The MIT Press, Cambridge, Massachusetts.
Engdahl, E.: 1983, Parasitic gaps, *Linguistics and Philosophy* **6**(1), 5–34.
Engdahl, E.: 1986, *Constituent Questions*, D. Reidel Publishing Co., Dordrecht.
Heim, I. and Kratzer, A.: 1998, *Semantics in Generative Grammar*, Blackwell Publishers, Oxford.
Jacobson, P.: 1999, Towards a variable-free semantics, *Linguistics and Philosophy* **22**(2), 117–185.
Munn, A.: 2001, Explaining parasitic gap constructions, *in* P. W. Culicover and P. M. Postal (eds), *Parasitic Gaps*, MIT Press, Cambridge, Massachusetts.
Nissenbaum, J.: 2000, *Investigations of Covert Phrase Movement*, PhD thesis, Massachusetts Institute of Technology.
Postal, P. M.: 1993, Parasitic gaps and across-the-board phenomena, *Linguistic Inquiry* **24**, 735–754.
Steedman, M.: 1987, Combinatory grammars and parasitic gaps, *Natural Language and Linguistic Theory* **5**, 403–439.
Szabolcsi, A. and Zwarts, F.: 1997, Weak islands and an algebraic semantics for scope taking, *in* A. Szabolcsi (ed.), *Ways of Scope Taking*, Kluwer Academic Publishers, Dordrecht, pp. 217–262.