

1 Outline

- Extant CCG treatments of dynamic anaphora: [de Groote 2006](#), [Barker & Shan 2008](#). [Motivation for pursuing another approach. BS clearly no good ([Charlow 2010](#)). dG works well enough, and can be combined with BS regime, but...?]
- Continuized CCGs offer a grammar-wide generalization of scope-taking (“ubiquitous scopal pied piping”) using three operations: **lift**, **triv**, and **scope**. Any constituent can be a scope-taker.
- Proposal: replace **lift** and **triv** with options that countenance side effects ([Shan 2005](#)). Any side effects regime can be grafted onto a continuized CCG, by replacing **lift** and **triv** with monadic functors ([Moggi 1989](#), [Wadler 1992, 1994, 1995](#), [Shan 2002](#)).
- We provide a general technique for integrating a monadic approach to side effects with continuations-based approaches to scope in CCG. We relate our approach to the ContT monad transformer ([Liang et al. 1995](#)). Offers a type-theoretic way to track effects, integrate them into a well-developed CCG framework for scope-taking.
- Dynamic semantics is ([Shan 2001](#)):¹
 - State: ability to manipulate the discourse context, i.e. create discourse referents.
 - Nondeterminism: analogizes indefinites to referential expressions. Treats indefinites as referring expressions, though ones which refer indeterminately.
- Corollary: there is no need to settle on a single (“the”) grammar. Different and quite varied side effects regimes can be modularly grafted onto a simple applicative (“pure”) core. Lexical entries that would seem incongruous in a flat-footed standard perspective integrate seamlessly in a single grammar.
- Monads as a natural way to extend a continuations-based grammar with tools for dynamic binding and exceptional scope. In the end: you have functional application, plus the functors from whichever monads are implicated in a given language.
- The standard continuations-based perspective of [Barker 2002](#), [Shan & Barker 2006](#), [Barker & Shan 2014](#) is an instantiation of a more general perspective.

¹ NB: does not characterize all varieties of dynamic semantics. Dynamic treatments following [Groenendijk & Stokhof 1990](#) (e.g. [Zimmermann 1991](#), [Dekker 1993](#), [Szabolcsi 2003](#), [de Groote 2006](#)) provide a way for indefinites to extend their binding domain but do not treat indefinites as nondeterministic analogs of proper names.

- Standard dynamic techniques (DPL, DMG) not reducible to monads.
- Broader question: how this relates to the idea that continuations can simulate any monad (?). I don’t understand this result well enough to say anything.

2 Adding side effects to k

- Standard continuized grammar:
 - **lift**: $\lambda k. k x$
 - **triv**: $\lambda x. x$
 - **scope**: $\lambda k. m (\lambda f. n (\lambda x. k (f x)))$
- Type-theoretic details here
- Adding side effects ([Wadler 1994, 1995](#), [Shan 2002](#)): monads
- Monad laws / punting
- Relating monads to continuized grammars:
 - Replace **lift** with \star
 - Replace **triv** with η
 - **scope** stays the same
- Two type constructors:
 - Bipartite Cont:
 - Unary Monadic:

3 Finding the dynamic monad

- The meat of PLA ([Dekker 1994](#)): sentences are relations on stacks. Non-empty relations correspond to truth. Non-functional pairs in the relation correspond to nondeterminism introduced by indefinites (and perhaps disjunction).

$$\llbracket \text{a linguist} \rrbracket = \lambda k s. \bigcup_{x \in \text{ling}} k x \widehat{s} x$$

- A different perspective on this: treating nondeterminism and state modification as side effects, within a functional programming setting for side effects.
- Monad for nondeterminism:

Definition 1 (The Set monad).

$$\begin{aligned} M a &::= a \rightarrow t \\ \eta x &::= \{x\} \\ m \star k &::= \bigcup_{x \in m} k x \end{aligned}$$

- Monad for state (generalization of monad for environment-sensitivity):

Definition 2 (The State monad).

$$\begin{aligned} Ma &::= s \rightarrow a \times s \\ \eta x &:= \lambda s. \langle x, s \rangle \\ m \star k &:= \lambda s. k (ms)_0 (ms)_1 \end{aligned}$$

- Use StateT to stitch the two together²

Definition 3 (The StateT monad transformer).

$$\begin{aligned} Ma &::= s \rightarrow L(a \times s) \\ \eta x &:= \lambda s. \eta_L \langle x, s \rangle \\ m \star k &:= \lambda s. ms \star_L \lambda \pi. k \pi_0 \pi_1 \end{aligned}$$

Definition 4 (The State_Set monad).

$$\begin{aligned} Ma &::= s \rightarrow (a \times s) \rightarrow t \\ \eta x &:= \lambda s. \{ \langle x, s \rangle \} \\ m \star k &:= \lambda s. \bigcup_{\pi \in ms} k \pi_0 \pi_1 \end{aligned}$$

- Static lexicon, dynamic lexicon
- Modular treatment of binding.

Previous : **bind** $m := \lambda k. m (\lambda x. k x x)$
 Proposal : **bind** $m := \lambda k. m (\lambda x s. k x \widehat{s x})$

- Summing up: three combinators for “order-insensitive” (i.e. continuized combination). **unit**, **run**, **bind**

	lift m	M triv	bind M
Previous	$\lambda k. k m$	$M (\lambda x. x)$	$\lambda k. m (\lambda x. k x x)$
Proposal	$\lambda k. m \star k$	$M \eta$	$\lambda k. m (\lambda x s. k x \widehat{s x})$

4 Examples

- Some upshots: no dynamic conjunction, completely standard model theory (cf. de Groote 2006). “Contexts of evaluation” are constructed on the fly.

de Groote 2001 Charlow 2014 Bumford to appear

References

- Barker, Chris. 2002. Continuations and the Nature of Quantification. *Natural Language Semantics* 10(3). 211–242. <http://dx.doi.org/10.1023/A:1022183511876>.
- Barker, Chris & Chung-chieh Shan. 2008. Donkey Anaphora is In-Scope Binding. *Semantics & Pragmatics* 1(1). 1–46. <http://dx.doi.org/10.3765/sp.1.1>.
- Barker, Chris & Chung-chieh Shan. 2014. *Continuations and Natural Language*. Oxford: Oxford University Press.

- Bumford, Dylan. to appear. Incremental quantification and the dynamics of pair-list phenomena. *Semantics & Pragmatics* 8(9).
- Charlow, Simon. 2010. Two kinds of binding out of DP. Unpublished ms.
- Charlow, Simon. 2014. *On the semantics of exceptional scope*: New York University Ph.D. thesis.
- Dekker, Paul. 1993. *Transsentential meditations: ups and downs in dynamic semantics*: University of Amsterdam Ph.D. thesis.
- Dekker, Paul. 1994. Predicate Logic with Anaphora. In Mandy Harvey & Lynn Santelmann (eds.), *Proceedings of Semantics and Linguistic Theory 4*, 79–95. Ithaca, NY: Cornell University.
- Groenendijk, Jeroen & Martin Stokhof. 1990. Dynamic Montague Grammar. In Laszlo Kalman & Laszlo Polos (eds.), *Proceedings of the Second Symposium on Logic and Language*, 3–48. Budapest: Eötvös Loránd University Press.
- de Groote, Philippe. 2001. Type raising, continuations, and classical logic. In Robert van Rooij & Martin Stokhof (eds.), *Proceedings of the Thirteenth Amsterdam Colloquium*, 97–101. University of Amsterdam.
- de Groote, Philippe. 2006. Towards a Montagovian account of dynamics. In Masayuki Gibson & Jonathan Howell (eds.), *Proceedings of Semantics and Linguistic Theory 16*, 1–16. Ithaca, NY: Cornell University.
- Liang, Sheng, Paul Hudak & Mark Jones. 1995. Monad Transformers and Modular Interpreters. In *22nd ACM Symposium on Principles of Programming Languages (POPL '95)*, 333–343. ACM Press.
- Moggi, Eugenio. 1989. Computational lambda-calculus and monads. In *Proceedings of the Fourth Annual Symposium on Logic in computer science*, 14–23. Piscataway, NJ, USA: IEEE Press.
- Shan, Chung-chieh. 2001. A Variable-Free Dynamic Semantics. In Robert van Rooij & Martin Stokhof (eds.), *Proceedings of the Thirteenth Amsterdam Colloquium*, University of Amsterdam.
- Shan, Chung-chieh. 2002. Monads for natural language semantics. In Kristina Striegnitz (ed.), *Proceedings of the ESSLLI 2001 Student Session*, 285–298.
- Shan, Chung-chieh. 2005. *Linguistic Side Effects*: Harvard University Ph.D. thesis.
- Shan, Chung-chieh & Chris Barker. 2006. Explaining Crossover and Superiority as Left-to-right Evaluation. *Linguistics and Philosophy* 29(1). 91–134. <http://dx.doi.org/10.1007/s10988-005-6580-7>.
- Szabolcsi, Anna. 2003. Binding on the Fly: Cross-Sentential Anaphora in Variable-Free Semantics. In Geert-Jan M. Kruijff & Richard T. Oehrle (eds.), *Resource-Sensitivity, Binding and Anaphora*, 215–227. Dordrecht: Kluwer Academic Publishers.
- Wadler, Philip. 1992. Comprehending monads. In *Mathematical Structures in Computer Science*, vol. 2 (special issue of selected papers from 6th Conference on Lisp and Functional Programming), 461–493.
- Wadler, Philip. 1994. Monads and composable continuations. *Lisp and Symbolic Computation* 7(1). 39–56. <http://dx.doi.org/10.1007/BF01019944>.

² Fn. about SetT

- Wadler, Philip. 1995. Monads for functional programming. In Johan Jeuring & Erik Meijer (eds.), *Advanced Functional Programming*, vol. 925 Lecture Notes in Computer Science, 24–52. Springer Berlin Heidelberg. http://dx.doi.org/10.1007/3-540-59451-5_2.
- Zimmermann, Thomas Ede. 1991. Dynamic logic and case quantification. In Martin Stokhof, Jeroen Groenendijk & David Beaver (eds.), *Quantification and Anaphora I* (DYANA Deliverable R2.2.A), 191–195.