Monads: Some Linguistic Applications

Ash Asudeh
University of Oxford & Carleton University

February 15, 2014

*This is joint work with **Gianluca Giorgolo** (University of Oxford). The formal aspects are his work. However, since he's not here, only I am to blame for misrepresentations or errors.*

1 Introduction

- In a number of recent papers and in work in progress, Gianluca Giorgolo and I have been using *monads*, an idea from category theory with wide application in computer science, to analyse various phenomena in natural language semantics and pragmatics.

- This work, and this talk, have used monads as way to augment the meaning language side of meaning constructors in an LFG+Glue setting (Dalrymple 1999, 2001, Asudeh 2005a,b, 2011, 2012). So it *is* related to LFG, but I'm afraid only tangentially so...

- The basic intuition behind a monad is that they provide a way to map from a) a simple space of objects and mappings between those objects to b) a more complex space of objects and mappings between them, while preserving important properties of the input space/mappings.

- They thus provide a useful analytical tool for natural language semantics and pragmatics, as first sketched by Shan (2001), because they allows a more expressive language for lexical meanings while preserving a relatively simple compositional system.

- We have used monads to analyze the following phenomena:

  1. Conventional implicature (Giorgolo and Asudeh 2011, 2012a)
  2. Optional arguments (Giorgolo and Asudeh 2012b)
  3. Perspective and opacity (Giorgolo and Asudeh 2013a,c)
  4. Conjunction fallacies (Giorgolo and Asudeh 2013b)

- In this talk, I will:

  ⊙ Introduce some of the basic intuitions behind monads

  ⊙ Review some of the basic intuitions behind our analyses of CIs and optional transitivity

  ⊙ Focus on presenting the intuitions behind our recent work on perspective semantics and conjunctional fallacies

2 Monads[1]

- Monads are a concept from category theory (Awodey 2010). They have found successful application in the semantics of programming languages (Moggi 1989, Wadler 1992).
- A monad is a special kind of endofunctor:[2] a structure-preserving mapping.
- A monad has the algebraic properties of a monoid: an object that
  1. can be combined with other objects of the same kind to return an object of the same kind; and
  2. has a unit object defined for it, such that combining an object α with the unit object returns α.
     For example: natural numbers with the operation of $+$ and the unit 0 or with the operation of $\times$ and the unit 1.
- We will think of monads as ways to structure-preservingly map a collection of types (named sets of objects) and mappings (functions) between these types to another collection of related types and mappings that we consider more complex.
- Structure-preservation means that things like the result of applying a function to an argument or the composition of two functions should be preserved in the more complex image, modulo the additional structure that comes with the new types.

  Example: The operation of pairing any object/type with another element of a *fixed* type is a monad. The image types are simply the pairings of the original types with the fixed type. The original functions are mapped so that they only operate on the values of the original type.

- We define monads here following the tradition in computer science. A monad is defined by three things:
  1. The functor that specifies
     (a) how we can associate the original types to the new types; and
     (b) what are the functions between these new types that correspond to the old functions
  2. an operation called *unit* that maps the actual values of the original types to their images under the functor
  3. an operation called *bind* that functions similarly to functional application (although it is normally written down in reverse argument, argument first and function second) and that allows us to "extract" the original values from their new more complex state and use them to generate new monadic values
- We write the monad as the triple $\langle M, \eta, \star \rangle$, where M is the functor, η the unit and $\star$ the bind.
- The type of η is $\forall a.a \rightarrow M(a)$.
- The type of $\star$ is $\forall a.\forall b.M(a) \rightarrow (a \rightarrow M(b)) \rightarrow M(b)$.[3]

[1] This section is lightly adapted from notes by Gianluca Giorgolo.

[2] An endofunctor is a functor that maps a category to itself. A category is a kind of algebraic structure consisting of objects linked by arrows, such that arrows can be composed associatively and each object has an arrow mapping from and to itself (an identity arrow).

[3] As you can see, $\star$ is like a flipped around functional application that would have the type $(a \rightarrow b) \rightarrow a \rightarrow b$; we use this flipped version to save on parentheses when we write our terms.

- Among the laws that guarantee a proper monad, we want to highlight these three:[4]

$$\eta(x) \star f = f(x) \tag{1}$$
$$m \star \eta = m \tag{2}$$
$$(m \star f) \star g = m \star (\lambda x. f(x) \star g) \tag{3}$$

  ⊙ (1) and (2) ensure that η doesn't do anything special to our values or attaches any additional information to them. Basically we want η to be like 0 with respect to +, but in this case η brings us to a new type.[5]

  ⊙ (3) requires ⋆ to be associative.[6]

2.1 Example: Parametrizing values to indices

- Let us define the monad that arises from the operation of parametrizing a value over some index of type i:[7] $\langle R, \eta, \star \rangle$.

- The functor, discussed as M above, we have renamed R, for *Reader*.[8]

  1. R maps every type τ to the type $i \to \tau$, i.e. the type of functions from indices to objects of type τ.
  2. R maps every function $f : a \to b$ to a function $R(f) : R(a) \to R(b)$; unwrapping the R on types, we get $R(f) : (i \to a) \to (i \to b)$.

  ⊙ Operationally $R(f)$ is just the composition of its first argument of type $i \to a$ with the input f (i.e., our old f):

$$R(f) = \lambda x. f \circ x \tag{4}$$

- The unit for this monad takes a value and returns a constant function that always returns that value independently of the index passed. In other words, unit makes sure that the original value is *not* actually parametrized (remember that it has to behave as a kind of 0). Here is how we define it:

$$\eta(x) = \lambda i.x \tag{5}$$

- Finally ⋆ basically work as a power version of functional application that also takes care of passing around the index passed as argument:

$$m \star f = \lambda i. f(m(i))(i) \tag{6}$$

[4] There are also other laws that a monad has to satisfy and that are related to its functorial nature, but we leave this aside here, as they would require discussion of the categorical nature of monads.

[5] Although, we also have the trivial identity monad.

[6] Bind doesn't *have* to be commutative, but in some cases it is.

[7] We purposefully choose to be vague by talking about indices, but we could instead consider a type w for worlds in a possible worlds semantics.

[8] This is its usual name in computer science as it models the operation of producing values that are *read* from some environment.

3 Glue with Monads[9]

Unit : Lozenge Introduction **Bind : Lozenge Elimination**

$$\frac{x : a}{\eta(x) : \Diamond a}\ \Diamond I$$

$$\frac{m : \Diamond a \qquad \begin{array}{c}[x : a]_i \\ \vdots \\ n : \Diamond b\end{array}}{m \star \lambda x.n : \Diamond b}\ \Diamond E_i$$

4 Linguistic Applications

4.1 Conventional Implicature

- Conventional implicature (Grice 1975, Potts 2005, 2007) is a challenging phenomenon for natural language semantics and pragmatics, because it displays an interaction between truth-conditional and non-truth-conditional meaning that is systematic but not entirely free.

(7) A: Most goddamn neighbourhood dogs crap on my damn lawn.
B: No, that's not true.
⇒ No, the neighbourhood dogs don't crap on your lawn.
⇏ No, there's nothing wrong with dogs and/or their crapping on your lawn.

(8) A: John Lee Hooker, the bluesman from Tennessee, appeared in *The Blues Brothers*.
B: No, that's not true.
⇒ No, John Lee Hooker did not appear in *The Blues Brothers*.
⇏ No, John Lee Hooker was not from Tennessee.
B: True, but actually John Lee Hooker was born in Mississippi

- The basic intuitions behind the analysis of Potts (2005) are:

  1. There are two separate dimensions to asserted meaning: the at-issue and CI dimensions.
  2. The at-issue dimension feeds the CI dimension, but not vice versa.

- Potts (2005: 52ff.) observed that there do seem to be exceptional cases where the at-issue dimension depends on the CI dimension.

  1. **Presupposition**

     (9) Mary, a good drummer, is a good singer <u>too</u>.

     Here the presupposition that Mary has some additional musical talent besides being a good singer is supported by the information conveyed by the nominal appositive *a good drummer*

  2. **Anaphora**

     (10) Jake$_1$, who almost killed a woman$_2$ with <u>his</u>$_1$ car, visited <u>her</u>$_2$ in the hospital.

     In (10) the pronoun *her* founds its antecedent in the non-restrictive relative clause, in a way leaking information from the CI dimension to the at-issue dimension.

  3. **VP ellipsis**

     (11) Lucy, who doesn't help her sister, told Jane <u>to</u>.

[9] This is based on Benton et al. (1998).

Similarly here the elided VP is first introduced in the relative clause, i.e. in the CI dimension.

4. **Nominal ellipsis/anaphora**

(12) Melinda, who won three games of tennis, lost because Betty won six.

Also here the nominal ellipsis (or the pronominal use of *six*) seems to break the Pottsian rule of information flowing only from the at-issue to the CI dimension.

- AnderBois et al. (2010, 2013) argue against the Pottsian division, based on examples like this.

- Giorgolo and Asudeh (2011, 2012a) propose a different solution based on the observation that the interaction between at-issue and side-issue content is limited to a certain class of discourse related phenomena and follows the same patterns we observe when dealing with proper discourse fragments.[10]

  ⊙ We started from the intuition that at the level of logical form generation we have two distinct semantic dimensions.

  ⊙ The interaction between these two dimensions follow Potts's principle of limited interaction: at-issue meaning resources can be re-used in the CI dimension but side-issue content never leaks into the at-issue dimension.

  ⊙ Once the logical form is complete (and possibly contains things like free variables and presuppositions to be satisfied) the boundaries between dimensions are lifted and the resolution of discourse-related uncertainties can take place, under the condition that we have two propositions that function as two distinct discourse segments.

  ⊙ The non-interaction between conventional implicatures and logical operators is explained, following Potts, in terms of multidimensionality. For instance if we apply our analysis to example (13)

  (13) Luke Skywalker is so gullible that he believes that Jabba the Hutt, a notorious scammer, is a trustworthy business partner.

  we obtain two distinct propositions, one expressing the fact that Skywalker is so gullible that he believes that Jabba is a good business partner and the other expressing the fact that Jabba is a notorious scammer.

  ⊙ The limited interactions we observe in examples (9–12) are instead explained as a discourse-level phenomenon.

  ⊙ This also allows us to predict the acceptability of (14) and the non-acceptability of (15).

  (14) All Cairo taxi drivers$_1$, who by the way painted their$_1$ taxis red in protest, are on strike.

  (15) *Every Cairo taxi driver$_1$, who by the way would threaten me with his$_1$ gun, is on strike.

[10]Another approach to modeling the composition of at-issue and side-issue meaning is the one of Arnold and Sadler (2010, 2011). Arnold and Sadler's analysis is cast in the framework of LFG and follows a suggestion by Potts (2005: 85ff.) in capturing multidimensionality directly in the logic for composition.

4.1.1 How Monads Help

- Giorgolo and Asudeh (2012a) propose that expressions contributing to the CI dimension can be seen as computations that, besides yielding a (possibly empty) value usable in the at-issue dimension, *log* some additional information to a special place, the CI dimension.
- The operation of logging information is well known in computer science and it is normally modelled in terms of a monad known as the *Writer* monad.
- Our basic intuition was that the information necessary to compute the satisfaction of the presupposition becomes accessible only at the end of the compositional process, since the log produced by the Writer monad cannot be examined before the monadic computation terminates.

4.2 Optional Arguments

- It is well-known that certain 2-argument verbs, like *eat* and *drink*, do not necessarily correspond to syntactic transitives:

(16) Alice ate yesterday afternoon. ⇔ Alice ate something yesterday afternoon / Alice ate something edible yesterday afternoon.

(17) Bob drank last night. ⇔ Bob drank something last night / Bob drank something alcoholic last night.

- The second argument of these verbs is thus syntactically optional, and when it is unexpressed there are often restrictions on its interpretation.
- In contrast, certain other 2-argument verbs, like *devour* and *quaff* require that their second arguments be syntactically expressed:

(18) a. Alice devoured the cake yesterday afternoon.
b. *Alice devoured yesterday afternoon.

(19) a. Alice quaffed the ale with gusto.
b. *Alice quaffed with gusto.

- We therefore need to capture these differences in terms of lexical differences.

4.2.1 How Monads Help

- Giorgolo and Asudeh (2012b) adapt a proposal by Blom et al. (2012) (in Abstract Categorial Grammar) to handle such cases in terms of an `option` operator in the semantics.
- We proposed using the *Maybe* (aka *Option*) monad, which handles error types, through the use of η and $\star$. The crucial bit is:

$$m \star k = \begin{cases} * & m = * \\ k(m) & \text{otherwise} \end{cases} \tag{20}$$

We assume a freely available `error` object, such that only lexical entries that can have optional arguments can handle this object.

- The lexical entry for, e.g., *eat*, has a packed meaning in which the `option` operator expresses the fact that the second argument can be existentially bound, while simultaneously potentially adding extra information about the nature of the object.

(21) eat V $(\uparrow \text{PRED}) = \text{‘}eat'$
$\lambda o \lambda s(\texttt{option}(o, \lambda u(\mathbf{eat}(s,u)), \exists x(\mathbf{eat}(s,x) \wedge \mathbf{food}.\mathbf{for}(x,s))))$
$\Diamond(\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap \uparrow_\sigma$

- Here is an example analysis:

$$\dfrac{\dfrac{}{[\![\text{John}]\!] : j}\text{John} \qquad \dfrac{\dfrac{}{[\![\text{ate}]\!] : \Diamond n \multimap j \multimap e}\text{ate} \qquad \dfrac{}{* : \Diamond n}\texttt{error}}{\lambda s \exists x(\mathbf{eat}(s,x) \wedge \mathbf{food}.\mathbf{for}(x,s)) : j \multimap e} \multimap E}{\exists x(\mathbf{eat}(\mathbf{john},x) \wedge \mathbf{food}.\mathbf{for}(x,\mathbf{john})) : e} \multimap E$$

- In contrast, the lexical entry for *devour* does not contain the `option` operator or corresponding lozenge modality in the Glue logic, and therefore cannot handle the `error` object. Therefore, *devour* must actually find a resource corresponding to its object argument, which can only be present if there is a syntactic object to contribute it.

4.3 Perspective and Opacity

- Opaque contexts have been an active area of research in natural language semantics since Frege's original discussion of the puzzle (Frege 1892).

- A sentence like (22) has a non-contradictory interpretation despite the fact that the two referring expressions *Hesperus* and *Phosphorus* refer to the same entity, the planet we know as Venus.

(22) Reza doesn't believe Hesperus is Phosphorus.

- The fact that a sentence like (22) includes the modal *believe* has influenced much of the analyses proposed in the literature, and has linked the phenomenon with the notion of modality.

- In Giorgolo and Asudeh (2013a), we challenge this view and try to position data like (22) inside a larger framework that also includes other types of expressions.

- We decompose examples like (22) along two dimensions: the presence or absence of a modal expression, and the way in which we multiply refer to the same individual.

- In the case of (22), we have a modal and we use two different co-referring expressions. Examples (23)-(25) complete the landscape of possible combinations:

(23) Dr. Octopus punched Spider-Man but he didn't punch Spider-Man.

(24) Mary Jane loves Peter Parker but she doesn't love Spider-Man.

(25) Reza doesn't believe Jesus is Jesus.

- (23) is clearly a contradictory statement, as we predicate of Dr. Octupus that he has the property of having punched Spider-Man and its negation.

- Notice that in this example there is no modal and the exact same expression is used twice to refer to the object individual.

- In the case of (24) we still have no modal and we use two different but co-referring expressions to refer to the same individual. However in this case the statement has a non-contradictory reading.

- Similarly (25) has a non-contradictory reading, which states that, according to the speaker, Reza doesn't believe that the entity he (Reza) calls Jesus is the entity that the speaker calls Jesus (e.g., is not the same individual or does not have the same properties). This case is symmetrical to (24), as here we have a modal expression but the same expression is used twice to refer to the same individual.
- If the relevant reading of (25) is difficult to get, consider an alternative in which Kim suffers from Capgras Syndrome[11] and thinks that Sandy is an impostor. The speaker says:

  (26) Kim doesn't believe Sandy is Sandy

- We propose an analysis of the non-contradictory cases based on the intuition that the apparently co-referential expressions are in fact interpreted using different interpretation functions, which correspond to different perspectives that are pitted against each other in the sentences.
- Furthermore, we propose that modal expressions are not the only ones capable of introducing a shift of perspective, but also that verbs that involve some kind of mental attitude of the subject towards the object have the same effect.
- Notice that this last observation distinguishes our approach from one where a sentence like (24) is interpreted as simply saying that Mary Jane loves only one *guise* of the entity that corresponds to Peter Parker but not another one.
- The problem with this interpretation is that if it is indeed the case that different co-referring expressions simply pick out different guises of the same individual, then a sentence like (27) should have a non-contradictory reading, while this seems not to be the case.

  (27) Dr. Octopus killed Spider-Man but he didn't kill Peter Parker.

- While a guise-based interpretation is compatible with our analysis,[12] it is also necessary to correctly model the different behaviour of verbs like *love* with respect to others like *punch* or *kill*.
- In fact, we need to model the difference between, for example, *kill* and *murder*, since *murder* does involve a mental attitude of intention and the corresponding sentence to (27) is not necessarily contradictory:

  (28) Dr. Octopus murdered Spider-Man but he didn't murder Peter Parker.

4.3.1 How Monads Help

- We use the *Reader* monad to analyze these phenomena.
- We represent linguistic expressions that can potentially be assigned different interpretations as functions from interpretation indices to values.
- Effectively we construct a different type of lexicon that does not represent only the linguistic knowledge of a single speaker but also her (possibly partial) knowledge of the language of other speakers.

[11]From Wikipedia: "[Capgras syndrome] is a disorder in which a person holds a delusion that a friend, spouse, parent, or other close family member has been replaced by an identical-looking impostor."

[12]Indeed, one way to understand guises is as different ways in which we interpret a referring term Heim (1998).

- So, for example, we claim that (25) can be assigned a non-contradictory reading because the speaker's lexicon also includes the information regarding Reza's interpretation of the name *Jesus* and therefore makes it possible for the speaker to use the same expression, in combination with a verb such as *believe*, to actually refer to two different entities.

- We argue that in one case the name *Jesus* is interpreted using the speaker's interpretation while in the other case it is Reza's interpretation that is used.

- Notice that we can apply our analysis to any natural language expression that may have different interpretations. This means that, for example, we can extend our analysis, which is limited to referring expressions here for space reasons, to other cases, such as the standard examples involving ideally synonymous predicates like *groundhog* and *woodchuck* (see, e.g., Fox and Lappin (2005)).

- The full readings derived in our system are shown in the figures that follow on the next page, based on the lexicon in table 1 for the speaker (only semantic details shown).

- Figure 1 reports the four non-equivalent readings that we derive in our system for example (22).[13]

  ⊙ Reading (38) assigns to both Hesperus and Phosphorus the subject interpretation and results, after contextualising the sentence by applying it to the standard σ interpretation index, in the truth conditions in (29), i.e. that Reza does not believe that the evening star is the morning star. This reading would not be contradictory in an epistemic model (such as Reza's model) where the evening star and the morning star are not the same entity.

  $$\neg\mathbf{B}(\mathbf{r})(\mathbf{es}_r = \mathbf{ms}_r) \tag{29}$$

  ⊙ In the case of (39) and (40) we get a similar effect although here we mix the epistemic models, and one of the referring expressions is interpreted under the speaker perspective while the other is again interpreted under Reza's perspective. For these two readings we obtain respectively the truth conditions in (30) and (31).

  $$\neg\mathbf{B}(\mathbf{r})(\mathbf{v}_\sigma = \mathbf{ms}_r) \tag{30}$$
  $$\neg\mathbf{B}(\mathbf{r})(\mathbf{v}_\sigma = \mathbf{es}_r) \tag{31}$$

  ⊙ Finally for (41) we get the contradictory reading that Reza does not believe that Venus is Venus, as both referring expressions are evaluated using the speaker's interpretation index.

  $$\neg\mathbf{B}(\mathbf{r})(\mathbf{v}_\sigma = \mathbf{v}_\sigma) \tag{32}$$

- If we consider a case like sentence (23), we ought to get only a contradictory reading as the statement is intuitively contradictory. Our analysis produces a single reading that indeed corresponds to a contradictory interpretation:

$$\begin{aligned}&[\![\text{Spider-Man}]\!] \star \lambda x. [\![\text{Spider-Man}]\!] \star \\ &\lambda y.\eta([\![\text{but}]\!]\,([\![\text{punch}]\!]\,([\![\text{Dr. Octopus}]\!])(x)) \\ &([\![\text{not}]\!]\,([\![\text{punch}]\!]\,([\![\text{Dr. Octopus}]\!])(y))))\end{aligned} \tag{33}$$

[13] The logic generates six different readings but the monad we are using here has a commutative behaviour, so four of these readings are pairwise equivalent.

- The verb *punch* is not a verb that can change the interpretation perspective and therefore the potentially controversial name Spider-Man is interpreted in both instances using the speaker's interpretation index. The result is unsatisfiable truth conditions, as expected:

$$\mathbf{punch}(\mathbf{o}_\sigma)(\mathbf{pp}_\sigma) \wedge \neg\mathbf{punch}(\mathbf{o}_\sigma)(\mathbf{pp}_\sigma) \quad (34)$$

- In contrast a verb like *love* is defined in our lexicon as possibly changing the interpretation perspective of its object to that of its subject. Therefore in the case of a sentence like (24), we expect one reading where the potentially contentious name Spider-Man is interpreted according to the subject of *love*, Mary Jane. This is in fact the result we obtain.

- Figure 2 reports the two readings that our framework generates for (24).

  ⊙ Reading (42), corresponds to the non-contradictory interpretation of sentence (24), where Spider-Man is interpreted according to Mary Jane's perspective and therefore is assigned an entity different from Peter Parker:

$$\mathbf{love}(\mathbf{mj}_\sigma)(\mathbf{pp}_\sigma) \wedge \neg\mathbf{love}(\mathbf{mj}_\sigma)(\mathbf{sm}_{mj}) \quad (35)$$

  ⊙ Reading (43) instead generates unsatisfiable truth conditions, as Spider-Man is identified with Peter Parker according to the speaker's interpretation:

$$\mathbf{love}(\mathbf{mj}_\sigma)(\mathbf{pp}_\sigma) \wedge \neg\mathbf{love}(\mathbf{mj}_\sigma)(\mathbf{pp}_\sigma) \quad (36)$$

- Our last example, (25) (or its Capgras alternative (26)) is particularly interesting as we are not aware of previous work that discusses this type of sentence.

- The non-contradictory reading that this sentence has seems to be connected specifically to two different interpretations of the same name, *Jesus*, both under the syntactic scope of the modal *believe*.

- Our system generates three non-equivalent readings, reported here in figure 3.[14]

  ⊙ Readings (44) and (45) correspond to two contradictory readings of the sentence: in the first case both instances of the name Jesus are interpreted from the subject perspective and therefore attribute to Reza the non-belief in a tautology, similarly in the second case, even though in this case the two names are interpreted from the perspective of the speaker.

  ⊙ In contrast the reading in (46) corresponds to the interpretation that assigns two different referents to the two instances of the name Jesus, producing the truth conditions in (37) which are satisfiable in a suitable model.

$$\neg\mathbf{B}(\mathbf{r})(\mathbf{j}_\sigma = \mathbf{j}_r) \quad (37)$$

[14] Again, there are six readings that correspond to different proofs, but given the commutative behaviour of the Reader monad, the fact that equality is commutative, and the fact that we have in this case two identical lexical items, only three of them are non-equivalent readings.

$$[\![\text{believe}]\!]\,([\![\text{Hesperus}]\!] \star \lambda x.\,[\![\text{Phosphorus}]\!] \star \lambda y.\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{38}$$

$$[\![\text{Hesperus}]\!] \star \lambda x.\,[\![\text{believe}]\!]\,([\![\text{Phosphorus}]\!] \star \lambda y.\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{39}$$

$$[\![\text{Phosphorus}]\!] \star \lambda x.\,[\![\text{believe}]\!]\,([\![\text{Hesperus}]\!] \star \lambda y.\eta([\![\text{is}]\!]\,(y)(x)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{40}$$

$$[\![\text{Hesperus}]\!] \star \lambda x.\,[\![\text{Phosphorus}]\!] \star \lambda y.\,[\![\text{believe}]\!]\,(\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{41}$$

Figure 1: Non-equivalent readings for *Reza doesn't believe Hesperus is Phosphorus.*

$$[\![\text{love}]\!]\,(\eta([\![\text{Peter Parker}]\!]))([\![\text{Mary Jane}]\!]) \star \lambda p.\,[\![\text{love}]\!]\,([\![\text{Spider-Man}]\!])([\![\text{Mary Jane}]\!]) \star \lambda q.\eta([\![\text{but}]\!]\,(p)([\![\text{not}]\!]\,(q))) \tag{42}$$

$$[\![\text{love}]\!]\,(\eta([\![\text{Peter Parker}]\!]))([\![\text{Mary Jane}]\!]) \star \lambda p.\,[\![\text{Spider-Man}]\!] \star \lambda x.\,[\![\text{love}]\!]\,(\eta(x))([\![\text{Mary Jane}]\!]) \star \lambda q.\eta([\![\text{but}]\!]\,(p)([\![\text{not}]\!]\,(q))) \tag{43}$$

Figure 2: Non-equivalent readings for *Mary Jane loves Peter Parker but she doesn't love Spider-Man.*

$$[\![\text{believe}]\!]\,([\![\text{Jesus}]\!] \star \lambda x.\,[\![\text{Jesus}]\!] \star \lambda y.\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{44}$$

$$[\![\text{Jesus}]\!] \star \lambda x.\,[\![\text{Jesus}]\!] \star \lambda y.\,[\![\text{believe}]\!]\,(\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{45}$$

$$[\![\text{Jesus}]\!] \star \lambda x.\,[\![\text{believe}]\!]\,([\![\text{Jesus}]\!] \star \lambda y.\eta([\![\text{is}]\!]\,(x)(y)))([\![\text{Reza}]\!]) \star \lambda z.\eta([\![\text{not}]\!]\,(z)) \tag{46}$$

Figure 3: Non-equivalent readings for *Reza doesn't believe Jesus is Jesus.*

WORD	DENOTATION	TYPE
Reza	$\mathbf{r}_\sigma$	e
Hesperus	$\lambda i.\begin{cases}\mathbf{es}_r & \text{if } i = \mathbf{r},\\ \mathbf{v}_\sigma & \text{if } i = \sigma\end{cases}$	$\Diamond e$
Phosphorus	$\lambda i.\begin{cases}\mathbf{ms}_r & \text{if } i = \mathbf{r},\\ \mathbf{v}_\sigma & \text{if } i = \sigma\end{cases}$	$\Diamond e$
Dr. Octopus	$\mathbf{o}_\sigma$	e
Mary Jane	$\mathbf{mj}_\sigma$	e
Peter Parker	$\mathbf{pp}_\sigma$	e
Spider-Man	$\lambda i.\begin{cases}\mathbf{sm}_i & \text{if } i = \mathbf{o} \text{ or } i = \mathbf{mj},\\ \mathbf{pp}_\sigma & \text{if } i = \sigma\end{cases}$	$\Diamond e$
Jesus	$\lambda i.\begin{cases}\mathbf{j}_r & \text{if } i = \mathbf{r},\\ \mathbf{j}_\sigma & \text{if } i = \sigma\end{cases}$	$\Diamond e$
not	$\lambda p.\neg p$	$t \to t$
but	$\lambda p.\lambda q.p \wedge q$	$t \to t \to t$
is	$\lambda x.\lambda y.x = y$	$e \to e \to t$
believe	$\lambda c.\lambda s.\lambda i.\mathbf{B}(s)(c(\kappa(s)))$	$\Diamond t \to e \to \Diamond t$
punch	$\lambda o.\lambda s.\mathbf{punch}(s)(o)$	$e \to e \to t$
love	$\lambda o.\lambda s.\lambda i.\mathbf{love}(s)(o(\kappa(s)))$	$\Diamond e \to e \to \Diamond t$

Table 1: Lexicon

4.4 Conjunction Fallacies

- Conjunction fallacies have been an active area of research in cognitive science for more than three decades now.

- The phenomenon was first discussed by Tversky and Kahneman (1983).

- They noticed that in a task asking for ratings of the relative likelihoods of different events, the majority of the subjects consistently rated the likelihood of the conjunction of two events as higher than the likelihood of one of the conjoined events.

- One of their examples is the well-known “Linda paradox”. Subjects were given the following statement and, as part of the experimental task, where asked to rank the probability that various statements were true of Linda; the resulting ranking for the relevant cases is given below the context.

  (47) Linda is 31 years old, single, outspoken and very bright. She majored in philosophy. As a student, she was deeply concerned with issues of discrimination and social justice, and also participated in anti-nuclear demonstrations.

  Linda is active in the feminist movement. [F(eminist)]
  Linda is a bank teller and is active in the feminist movement. [T&F]
  Linda is a bank teller. [T(eller)]

  The context is obviously designed to bias towards the label *Feminist* and it is unproblematic and unsurprising that the relevant proposition is ranked most likely, but the result that the joint probability T&F is ranked higher than T is interesting, and constitutes an instance of *conjunction fallacy*: a conjunction of two propositions is reported by subjects to be more probably than the probability of one of the two propositions on its own.

- This result disagrees with the rules of probability, as the probability of the conjunction of two events, being the intersection of the two events, cannot be higher than the likelihood of any of the two events, formally for any two events A and B:

$$P(A \text{ and } B) \leq P(A), P(B) \tag{48}$$

- These results have been replicated by many studies that have investigated different ways in which this apparently fallacious response can be elicited (Yates and Carlson 1986, Tentori et al. 2004).

- The original explanation of these results by Tversky and Kahneman (1983) was in terms of representativeness.

- The authors claimed that the observed responses are due to the fact that subjects do not operate in terms of probabilistic reasoning, but rather use a *representativeness heuristic*.

- According to Tversky and Kahneman (1983), subjects check the degree of correspondence between the events and a certain model of reality and select those event that are closer to what the model predicts as being the more likely events.

- Representativeness tends to covary with frequency but not necessarily. A crucial point of Tversky and Kahneman (1983)’s analysis is that this heuristic operates on the conjunction as whole, or as they put it:

> the judged probability (or representativeness) of a conjunction cannot be computed as a function (e.g., product, sum, minimum, weighted average) of the scale values of its constituents. (Tversky and Kahneman 1983: p. 305)

- This last property is rather problematic if we intend to integrate their observations with current linguistic theories of meaning composition.
- It goes against the *principle of compositionality* (Janssen 2011: among many others), which is ultimately based on the philosophy of language of Frege (1952a).
- This principle states that the meaning of a composite expression is determined by the meaning of its constituents. This is in clear contradiction with the analysis of Tversky and Kahneman.
- However compositionality has proven a very important tool in linguistic research, in theoretical semantics and in the philosophy of language, representing the semantic counterpart of linguistic productivity.
- Moreover compositionality seems to be a pervasive, and indeed quite successful, strategy in human cognition. Therefore we should be careful before discarding it.

4.4.1 How Monads Help

- In the model of Giorgolo and Asudeh (2013b), the observation made in the literature (Hertwig and Gigerenzer 1999) that conjunction fallacies arise only under specific conditions and can be cancelled if other conditions are imposed is explained in terms of cognitive/computational economy.
- In fact the same computational structure, the monad, can be used together with a number of different underlying semirings, one of them being the probability semiring.
- We predict that, in general, if the subjects are presented with a task where there are "no stakes" they will base their judgement on the basis of a reasoning modeled using a representation corresponding to a semiring defined over a relatively simple set with generally simple operations.
- Using this strategy will generally lead subjects to make overconfident estimations, which tend not to correspond to the reality of things.
- In terms of psychological heuristics, our proposal is that conjunction fallacies involve the heuristic of *satisficing* Simon (1956), either in addition to or instead of the heuristic of *representativeness* Tversky and Kahneman (1983).
- If, on the other hand, subjects are forced to evaluate the consequences of their judgements, such as in the context of a gaming scenario, or if explicitly primed to think in frequentist terms, then we will observe a switch to a more complex representation, with properties that better approximate those of probability theory.
- Crucially, in our model, logical operators such as *and* or *or* maintain their core logical meaning, while the probabilistic behaviour is determined by the context in which they operate.
- In this sense, with respect to Hertwig et al.'s analysis, we switch the ambiguity to the context rather than assuming that a word like *and* has multiple meanings.
- This is in fact a problem with Hertwig et al.'s analysis, which effectively treats *and* as ambiguous, i.e. truly polysemous.

- We instead follow the standard Gricean approach (Grice 1975): we treat *and* unambiguously like logical conjunction but allow it to derive additional "implicatures" depending on the *context* of use.

- In this sense we recuperate Grice's fundamental intuition that despite the fact that the same words are *used* with different meanings, speakers are not necessarily confused about the semantic meaning, as linguistic expressions are always evaluated with respect to a context (Grice 1975).

- Moreover the different meanings are often explainable as contextual modification of a core meaning.

- We first need to introduce the concept of a *semiring* from algebra. A semiring is a set A with two distinguished elements 0 and 1 and equipped with two binary operations $+$ and $\cdot$, satisfying the following axioms, for all x, y and $z \in A$

$$(x+y)+z = x+(y+z) \tag{49}$$
$$x+y = y+x \tag{50}$$
$$x+0 = 0+x = x \tag{51}$$
$$(x \cdot y) \cdot z = x \cdot (y \cdot z) \tag{52}$$
$$x \cdot 1 = 1 \cdot x = x \tag{53}$$
$$x \cdot 0 = 0 \cdot x = 0 \tag{54}$$
$$x \cdot (y+z) = (x \cdot y) + (x \cdot z) \tag{55}$$
$$(x+y) \cdot z = (x \cdot z) + (y \cdot z) \tag{56}$$

- In the case of the probability semiring, A is the real interval $[0,1]$, with 0 and 1 representing the two units and $+$ and $\cdot$ defined in the usual way.

- The fact that probability distributions form a monad in the category of measurable spaces was an early discovery in category theory Lawvere (1962), Giry (1982).

- We take the space of simple objects to be represented by the collection of different semantic types of natural language expressions, e.g. expressions that refer to an individual, such as proper names, expressions that denote some truth about reality, such as propositions, and expressions that denote a collection of individuals, such as common nouns.

- The mappings between these objects are expressed by similar expressions that "bring" us from one type of expression to another: for example we take a predicate expressed by a verb as a way to map the individual denoted by the subject expression to a truth value (when uttering *John sleeps* the verb *sleeps* maps the individual John to truth if John is indeed asleep and to false otherwise).

- The probability monad lifts these simple objects to probability distributions over inhabitants of the types, and so the mappings are transformed so that they relate different probability distributions (in our previous example the predicate would map from the probability distribution corresponding to the denotation of the subject, one that possibly assigns the entire probability mass to the individual John, to the probability distribution over the truth values, effectively giving us an estimate of the likelihood of the event that John is asleep).

- Formally our monad is defined by the triple $\langle P, \eta, \star \rangle$. P is a functor that operates in the way described above: it maps our semantic types to probability distributions over the inhabitants of the type and lifts the mappings between the types so that they operate between probability distributions.

- η is an operation that creates a special type of probability distribution that corresponds to a categorical judgement, i.e. assigning the entire probability mass to a single element of a semantic type. Basically this is a way to integrate certainty into our model of uncertainty.

- Formally we define it as follows (where we represent probability distributions as functions from types to the interval $[0,1]$, or in general to any base set of a semiring):

$$\eta(x) := y \mapsto \begin{cases} 1 \text{ if } y = x \\ 0 \text{ otherwise} \end{cases} \quad (57)$$

  where x represents an element of any semantic type (e.g. an individual, a truth value, a collection of individuals).

- The second operation, $\star$, is the way in which we combine uncertainty. Its definition is based on the definition of joint probability.

- It is important to notice that the joint probability we are discussing here is not the one we are trying to model in the case of a conjunction.

- The one discussed here is much more primitive and is at the core of the process that constructs the probability of an event from the linguistic elements that describe it.

- The joint probability that we will discuss in what follows arises as the byproduct of a large number of interactions described by the bind operation.

- There is also another difference between the way the bind operation is used and what we normally mean by joint probability.

- While the joint probability of two events gives us the likelihood of a third event (the occurrence of both atomic events), bind returns the probability distribution of what we can consider another atomic event. Bind is defined as in (58).

$$m \star k := y \mapsto \sum_{x \in A} m(x) \cdot k(x)(y) \quad (58)$$

  where A is the set of elements measured by the probability distribution m.

- Bind takes two arguments, a probability distribution (m) and a function (k) from elements of the probability distribution m to probability distributions over elements of (possibly) another set.

- The resulting probability distribution is obtained by collecting all possible ways in which we can obtain the various results from this second set, and by scaling them ($\cdot$) using the likelihood that they emerge from the first distribution. The results are collected together using addition.

- It is quite clear that despite the fact that we have been talking about a probability monad, all operations involved in its definition are those we have described for semiring.

- This means that we can use the same general structure we have discussed here and replace the meanings of 0, 1, $\cdot$, and $+$ with constants and operations defined for other semirings.

- We can reproduce the results reported in Yates and Carlson (1986) using our monadic infrastructure. We use their data as it is the only example in the literature where the relative likelihood of the atomic events in conjunctions has been (at least partially) controlled for. This gives us the possibility of deriving the overall likelihood of the conjoined event in a compositional fashion starting from the atomic events.

Likelihood of event A	Likelihood of event B	Observed rating
U	U	$P(A \text{ and } B) \leq P(A), P(B)$
U	L	$P(A) \leq P(A \text{ and } B) \leq P(B)$
L	L	$P(A), P(B) \leq P(A \text{ and } B)$

Table 2: Results reported by Yates and Carlson (1986)

- The first step is to define a suitable base for our semiring. Yates and Carlson (1986) employ a discrete scale based on the general prediction made by their summation model.
- Their model does not take into consideration the extreme cases, i.e. impossible and certain events.
- We want to include them in our model as they are necessary in order to model what we know about the logical entailment behaviour of the word *and*. Tentori et al. (2004) showed that subjects that commit conjunction errors correctly apply the rules of logic.
- Therefore we will use a simple discrete set as the base for our semiring: {I(mpossible), U(unlikely), P(ossible), L(ikely), C(ertain)}. I and C correspond to 0 and 1 respectively. The only additional condition we need to impose so that I and C behave as boolean values is that for all x in our set $x + C = C$.
- There are sixteen possible well behaved semirings that we can define for this set.
- The next step is to select those semirings that reproduce the results reported in Yates and Carlson (1986) and here summarised in table 2.
- To do so we have to explain how we expect the uncertainty attached to the two atomic events to propagate to the conjunction. This is described formally by the function in (59).[15]

$$\langle p, q \rangle \mapsto p \star (x \mapsto q \star (y \mapsto \eta(x \wedge y))) \tag{59}$$

- This function takes a pair of probability distributions over truth values (describing how likely an event is realised in the real world) and constructs a new probability distribution over the results of conjoining the truth value of the first event with the one of the second.
- In this particular case there are three ways in which the final result can be false, and only one in which it turns out true (when both events are realized).
- Using this schema we can evaluate how the sixteen semirings match the simplified results in table 2.
- It turns out that only one of the sixteen semirings satisfies the conditions induced by the data.

[15]This function is actually generated in a completely compositional fashion by the grammatical system, as shown here.

+	I	U	P	L	C
I	I	U	P	L	C
U	U	U	P	L	C
P	P	P	P	L	C
L	L	L	L	L	C
C	C	C	C	C	C

·	I	U	P	L	C
I	I	I	I	I	I
U	I	U	U	U	U
P	I	U	P	P	P
L	I	U	P	L	L
C	I	U	P	L	C

Table 3: The one semiring.

−	I	U	P	L	C
	C	L	P	U	I

Table 4: Complement operation for the one semiring.

- We show the one semiring in table 3. Notice that this semiring is not homomorphic to the probability semiring, meaning that we cannot reproduce its behaviour using probability theory.

- Recall that we assume that conjunction fallacies are true errors, in the sense that they lead to overestimation of the likelihood of events, but at the same time they are correct applications of a different strategy for computing the likelihood of composed events.

- Our explanation is given in terms of cognitive/computational economy. If there are no real stakes on the table, and therefore there is no incentive in using a safer but costlier computational system, subjects will employ a form of shortcut, represented in our model by a simpler semiring. If on the other hand, subjects are pushed to think about the consequences of their judgements, the more expensive solution is selected.

- The semiring described in table 3 is undoubtedly simpler than the standard probability semiring. First of all it is based on a simpler base set.

- But it also has another important property: it is in fact possible to reconstruct the entire semiring on the basis of only one of the two operations and a simpler complement operation, $-$. If we define the complement as in table 4 (which seems to be most intuitive way to define it, as we just reverse the order of the relative likelihoods), then we can observe that for all x and y in our base set we have that $x \cdot y = -(-x + -y)$ or alternatively $x + y = -(-x \cdot -y)$. This means that this particular encoding of uncertainty has much lower representational costs than other competing possibilities. Its symmetry makes it a particular simple and efficient computational object.

- From a processing perspective this means that we can posit a single compositional process that computes the likelihood of two conjoined events in a way that is completely blind to the specific details of how uncertainty is encoded.

- We just require that the encoding satisfies the axioms of a semiring. Moreover we observe the same general process, here formalized in terms of monads, at play in other areas of natural language meaning, as noted throughout this talk (see also Unger 2011).
- The two encodings we have discussed are instead selected by the context under which the description of the conjoined event is evaluated.
- If there are no real risks and judgements have no real consequences, then we predict that subject will select a computationally and representationally cheap encoding (the one semiring), otherwise they will apply the rules of probability, which require a much higher degree of computation and are representationally more costly. In terms of psychological heuristics, our proposal is that conjunction fallacies involve the heuristic of *satisficing* Simon (1956), either in addition to or instead of the heuristic of *representativeness* Tversky and Kahneman (1983).

5 Conclusion

- I have reviewed previous work and work in progress that shows how monads can analyze the following phenomena:
    1. Conventional implicature: the *Writer* monad
    2. Optional arguments: the *Maybe* monad
    3. Perspective and opacity: the *Reader* monad
    4. Conjunction fallacies: the *Probability* monad
- Monads are a powerful tool for natural language semantics and pragmatics that hold much potential.
- One line of future work is to look at interactions between phenomena analyzed with monads, since monads can be composed in many circumstances.

References

AnderBois, Scott, Adrian Brasoveanu, and Robert Henderson. 2010. Crossing the Appositive / At-Issue Meaning Boundary. In Nan Li and David Lutz, eds., *Semantics and Linguistic Theory (SALT) 20*, 328–346. Ithaca, NY: CLC Publications.

—. 2013. At-issue Proposals and Appositive Impositions in Discourse. *Journal of Semantics* .

Arnold, Doug, and Louisa Sadler. 2010. Pottsian LFG. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of LFG10*, 43–63. Stanford, CA: CSLI Publications.

—. 2011. Resource Splitting and Reintegration with Supplementals. In Butt and King 2011, 26–46.

Asudeh, Ash. 2005a. Control and Semantic Resource Sensitivity. *Journal of Linguistics* 41(3): 465–511.

—. 2005b. Relational Nouns, Pronouns, and Resumption. *Linguistics and Philosophy* 28(4): 375–446.

—. 2011. Towards a Unified Theory of Resumption. In Alain Rouveret, ed., *Resumptive Pronouns at the Interfaces*, 121–187. Amsterdam: John Benjamins.

—. 2012. *The Logic of Pronominal Resumption*. Oxford: Oxford University Press.

Awodey, Steve. 2010. *Category Theory*. Oxford University Press, 2nd edn.

Benton, Nick, G. M. Bierman, and Valeria de Paiva. 1998. Computational Types from a Logical Perspective. *Journal of Functional Programming* 8(2): 177–193.

Blom, Chris, Philippe de Groote, Yoad Winter, and Joost Zwarts. 2012. Implicit Arguments: Event Modification or Option Type Categories? In Maria Aloni, Vadim Kimmelman, Floris Roelofsen, Galit Sassoon, Katrin Schulz, and Matthijs Westera, eds., *Logic, Language and Meaning, 18th Amsterdam Colloquium, Amsterdam, The Netherlands, December 19-21*, vol. 7218 of *Lecture Notes in Computer Science*, 240–250. Berlin/Heidelberg: Springer.

Butt, Miriam, and Tracy Holloway King, eds. 2011. *Proceedings of the LFG11 Conference*. Stanford, CA: CSLI Publications.

Dalrymple, Mary, ed. 1999. *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*. Cambridge, MA: MIT Press.

Dalrymple, Mary. 2001. *Lexical Functional Grammar*. San Diego, CA: Academic Press.

Fox, Chris, and Shalom Lappin. 2005. *Foundations of Intensional Semantics*. Oxford: Blackwell.

Frege, Gottlob. 1891. Funktion und Begriff. In Geach and Black 1952. Translated as Frege (1952a).

—. 1892. Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik* 100: 25–50. Translated as Frege (1952b).

—. 1952a. Function and Concept. In Geach and Black 1952, 22–41. Translation of Frege (1891).

—. 1952b. On Sense and Reference. In Geach and Black 1952, 56–78. Translation of Frege (1892).

Geach, Peter T., and Max Black, eds. 1952. *Translations from the Philosophical Writings of Gottlob Frege*. Oxford: Blackwell.

Giorgolo, Gianluca, and Ash Asudeh. 2011. Multidimensional Semantics with Unidimensional Glue Logic. In Butt and King 2011, 236–256.

—. 2012a. $\langle M, \eta, \star \rangle$ Monads for Conventional Implicatures. In Ana Aguilar Guevara, Anna Chernilovskaya, and Rick Nouwen, eds., *Proceedings of Sinn und Bedeutung 16*. MIT Working Papers in Linguistics.

—. 2012b. Missing Resources in a Resource-Sensitive Semantics. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG12 Conference*, 219–239. Stanford, CA: CSLI Publications.

—. 2013a. Monads as a Solution for Generalized Opacity. Ms., University of Oxford.

—. 2013b. One Semiring to Rule Them All. Ms., University of Oxford.

—. 2013c. Perspective Semantics. Ms., University of Oxford.

Giry, Michele. 1982. A categorical approach to probability theory. In *Categorical aspects of topology and analysis, Proc. int. Conf., Ottawa 1981, Lect. Notes Math. 915, 68-85 (1982).*.

Grice, H. Paul. 1975. Logic and Conversation. In Peter Cole and Jerry Morgan, eds., *Speech Acts*, vol. 3 of *Syntax and Semantics*, 43–58. New York: Academic Press.

Heim, Irene. 1998. Anaphora and semantic interpretation: A reinterpretation of Reinhart's approach. In Uli Sauerland and Orin Percus, eds., *The Interpretive Tract*, vol. 25 of *MIT Working Papers in Linguistics*, 205–246. Cambridge, MA: MITWPL.

Hertwig, Ralph, Björn Benz, and Stefan Krauss. 2008. The conjunction fallacy and the many meanings of *and*. *Cognition* 108: 740–753.

Hertwig, Ralph, and Gerd Gigerenzer. 1999. The "Conjunction Fallacy" Revisited: How Intelligent Inferences Look Like Reasoning Errors. *Journal of Behavioral Decision Making* 12: 275–305.

Janssen, Theo M. V. 2011. Compositionality. In Johan van Benthem and Alice ter Meulen, eds., *Handbook of Logic and Language*, 495–553. London: Elsevier, 2nd edn.

Lawvere, Francis William. 1962. The Category of Probabilistic Mappings. Seminar handout.

Moggi, Eugenio. 1989. Computational Lambda-Calculus and Monads. In *LICS*, 14–23. IEEE Computer Society.

Potts, Christopher. 2005. *The Logic of Conventional Implicatures*. Oxford: Oxford University Press.

—. 2007. The Expressive Dimension. *Theoretical Linguistics* 33(2): 165–197.

Shan, Chung-chieh. 2001. Monads for Natural Language Semantics. In Kristina Striegnitz, ed., *Proceedings of the ESSLLI-2001 Student Session*, 285–298. 13th European Summer School in Logic, Language and Information.

Simon, Herbert A. 1956. Rational Choice and the Structure of the Environment. *Psychological Review* 63(2): 129–138.

Tentori, Katya, Nicolao Bonini, and Daniel Osherson. 2004. The conjunction fallacy: a misunderstanding about conjunction? *Cognitive Science: A Multidisciplinary Journal* 28(3): 467–477.

Tversky, Amos, and Daniel Kahneman. 1983. Extensional Versus Intuitive Reasoning: The Conjunction Fallacy in Probability Judgment. *Psychological Review* 90(4): 293–315.

Unger, Christina. 2011. Dynamic Semantics as Monadic Computation. In Manabu Okumura, Daisuke Bekki, and Ken Satoh, eds., *New Frontiers in Artificial Intelligence - JSAI-isAI 2011*, 68–81.

Wadler, Philip. 1992. Comprehending Monads. In *Mathematical Structures in Computer Science*, 61–78.

Yates, J. Frank, and Bruce W. Carlson. 1986. Conjunction Errors: Evidence for Multiple Judgment Procedures, Including "Signed Summation". *Organizational Behavior and Human Decision Processes* 37: 230–253.