Active Logic and Heim's Rules for Updating Discourse Context

John Gurney
Processing and Display Division
Army Research Laboratory, Adelphi,
MD 20783
gurney@aic.nrl.navy.mil

Don Perlis and **Khemdut Purang**
Dept. of Computer Science
University of Maryland, College Park,
MD 20742
{perlis, kpurang}@cs.umd.edu

Abstract

Discourse unfolds in time, giving rise to a cascade of belief changes in the listener. Yet this temporal evolution of discourse and belief is typically ignored in theoretical treatments of discourse. It has been claimed (see Soames [Soames, 1989]) that Heim's [Heim, 1983] theory of discourse context accounts for non-implicative discourse updating. We will present a new non-implicative discourse that cannot be accounted for with Heim's use of global or local accommodation and which appears to require attention to *evolution* of discourse. We use this example to motivate remaking Heim's update function, aimed toward a unified approach to discourse—one in which Heim's rules for discourse updating can account for more of the problem cases for the theory of discourse context. These rules and the revised update function can then serve as principles that constrain the building of representations for discourse context (such as the Discourse Representation Structures, of Discourse Representation Theory, [Kamp and Reyle, 1993]). We propose *active logic* as a convenient tool for executing the required inferences (as called for by our revised version of Heim's update function) as the discourse evolves through time.

Keywords: presupposition, discourse, context, accommodation, active logic

1 Background on active logic and presupposition.

Active logic [Elgot-Drapkin and Perlis, 1990, Perlis and Miller, 1993] is a family of formalisms developed for the purpose of modeling the reasoning process in a way that respects the passage of time as reasoning proceeds. That is, the reasoner may have beliefs (draw inferences) concerning what time it is (now), and these are updated during reasoning since what time it is changes during reasoning. Moreover, this update is itself reasoned, i.e., governed by the same inferential processes as the rest of the reasoning. These formalisms have been applied to a number of domains, from multi-agent interaction to deadline-coupled planning, from fully-decidable default reasoning to reasoning in the presence of contradictions, from correcting misidentification errors to perceptual reference.

Active-logic formalisms can have time-sensitive inference rules. The most obvious case of such a rule is the "clock-rule": from Now(t) [the time is now t] infer Now(t+1) [the time is now t+1] (and do *not* retain Now(t) as a belief). In such a logic, the update (or truth-maintenance) in which new information is integrated into the belief base, is not a separate process from the drawing of (defeasible) conclusions. Rather than proceeding from one nonmonotonic theory (with one set of axioms) to another nomonotonic theory (with an updated set of axioms) there is one evolving theory in which the evolution is itself determined by the inference rules. In particular the incorporation of new information is adjudicated as an inference process, including the decision as to which "axioms" and "inferences" to retain and which to reject. It is not assumed that newer information is necessarily more trustworthy than previous information; this is determined by reasoning with the full benefit of world-knowledge. Moreover, in active logic the reasoning is informed by its own temporal history: an inference at time t can be made partly on the basis of what has or has not been inferred at particular past time steps.

The above rule for Now-update is the ultimate in nonmonotonicity: it undoes its own antecedent! This rule is also the basis for most of the interesting behaviors of active logic: in active-logic inference, what might be inferred from B at time t need not be inferred from B at times t+1 or t−1. Thus in active logic it is possible to infer, at time t, that Q, given that it is now t ("Now t") and given "Q will be true at time t". Traditional temporal logics do not treat the present (now) as a changing entity during reasoning but rather tacitly treat reasoning as occurring in a timeless present; while this may be appropriate for some purposes, it does not meet the needs of a reasoner embedded in the world, and particularly of a participant in a dialog.

Here we propose to apply active logic to dialog, and in particular to presuppositional pragmatic inference. The motivation for this is in part that in conversation, information comes in the form of utterances over time, and thus there is no fixed axiom set on which to base reasoning. Rather, as new information comes along (in the form of further utterances) the reasoner-listener must revise its beliefs, perhaps retracting some altogether and altering others. Garden path sentences[1] [Crain and Steedman, 1985] are particularly striking examples of this, but we think that even ordinary sentences can have such effects, as in "My house was never built." Before the utterance is fully heard, there is a strong presupposition that my house exists, but this is then denied by the rest of the sentence. Multi-sentence dialog is even more clearly of this sort, as in "The roses are not in the fridge. I did not buy any roses."

Traditional studies of presupposition have focused on characterizing the "right" final conclusion (e.g., that there are no roses) while paying scant attention to the intermediate processes involved in coming to such a conclusion. We think that there is much to be gained from a more fine-grained approach, modeling the underlying step-by-step reasoning a listener may perform during the unfolding of an ongoing conversation. Active logic seems to us to be a potentially powerful tool for modeling such behavior, allowing contradictory beliefs to trigger *reasoned* belief revision.

This paper sketches some of our ideas for applying active logic to presupposition in dialog, with particular attention to Heim's treatment of negation. We are broadly sympathetic to Heim's approach; in trying to formalize it for our purposes we have made a number of observations that have further focused our efforts. One such is that of multiple presuppositions, only one of which is defeated by a later utterance, as in "The roses are *not* in the fridge. There *are* no roses." Here an inference to the effect that there are roses and that there is a refrigerator seems called for by the first sentence; such an inference is presuppositional: it appears necessary in order to understand the sentence, whether or not the sentence is true; a listener "accommodates" by making these assumptions. But then the second sentence requires us to revoke the presupposition that there are roses, while leaving intact the belief that there is a refrigerator.

Straightforward application of Heim's rules to such examples appears either to revoke not only the initial presupposition that there are roses but also that there is a fridge or to revoke neither. An active-logic treatment allows us a more fine-grained control over the underlying time-situated reasoning in which only the appropriate defeat is performed. We have devised a number of active logic inference rules for this purpose, and applied them to the above example and others. We are now in the process of implementing such an active logic theory of presuppositional defeat, which we hope to report on in another paper.

In this paper we will present a treatment of the above "fridge and roses" problem, as a key illustration of our ideas. First however we provide some material to orient the reader to our assumptions and point of view, followed by additional examples of dialogs and then a brief discussion of Heim's rules for handling context change and accommodation.

[1] E.g., "The horse raced past the barn fell".

2 Orientation

Here we briefly note some of our background assumptions and terminology.

1. As a discourse unfolds, the participants' understandings of what is being conveyed grow (and perhaps also shrink). The understanding by a given participant of what is being conveyed is the (changing) context discourse (for that participant).

2. We represent the discourse context as an n-tuple of first order formulae. It represents what one person accepts the discourse to be saying. We do not treat presumed background beliefs (for now).

3. In the rather simple discourses we discuss, rhetorical relations are always signaled explicitly by occurrences of the words: 'the', 'but', 'because', 'if', 'then', 'so'. We will show how an active-logic reasoner can use knowledge of these relations to make both tentative and ultimate decisions about how to update the discourse context.

4. Natural language presupposition and cancellation of presupposition is one place where the avoidance of contradiction plays a role (that is, it plays a role in the theoretical discussion of presupposition). We therefore take as our first material such presuppositional discourses. But (as we will show) our treatment of contradiction differs from the norm. We think the appearance of contradiction is something to be reasoned about—for often there is more than one way to remove a contradiction and there is more than one way to proceed after such removal. Active logic facilitates reasoning about (and in the presence of) contradictions.

5. We have chosen to implement a (modified) version of Heim's rules for updating discourse context (the Context Change Potential (CCP) Rules). Our reasons are:

   (a) These rules explicitly represent context (although as semantic items). And active logic is a system that (typically) operates on explicit representations.

(b) The CCP rules seem to explain some presupposition phenomena correctly in terms of anaphoric structure of the discourse. Thus Heim's system gets some things right that the notable alternatives (deriving from Gazdar) get wrong. And we believe that we can build a system that will ultimately get all of these context updating phenomena right.

6. In our thinking about these problems we and others (see Heim, etc.) have often selected for study a few simple but rather unnatural discourses. The assumption was that pragmatic phenomena like presupposition can be accounted for using a few simple relations between, say, pairs of sentences. Once these relations were found and formalized, perhaps we could generalize to more complex, more natural discourses. In this paper we set that working hypothesis aside:

   (a) we will show that there are slightly more complex discourses that may not be generalizable from the simple ones;

   (b) we propose that these (and therefore the others) must be understood by appeal to somewhat more complex and variable forms of reasoning, involving, for example, local rhetorical relations, along with consistency criteria.

3 More Examples

The presupposition phenomena we will discuss occur in the following discourses.

D_1 = ⟨ There are roses and tulips. But the roses are not yellow. ⟩

D_2 = ⟨ There *is* no king. So the king is *not* in hiding. ⟩

D_3 = ⟨ If John has a son, then his son is not here. ⟩

D_4 = ⟨ If I discover that Bill is in New York there will be trouble. ⟩

D_5 = ⟨ The King is not in hiding, Because there is no king. ⟩

D_6 = ⟨ There are no roses. So the roses are not in the fridge. ⟩

D_7 = ⟨ The roses are not in the fridge. Because there are no roses. ⟩

It is clear that presuppositional behavior is complex—even for relatively simple discourses. Apparently certain syntactic forms (such as the definite description 'The roses') give rise to a presupposition ('There are roses') which may or may not produce a presupposition in the total discourse.[2]

[2] For our purposes here we will only work with examples where presupposition is caused by the use of a definite description. We assume that what we say could also be said for some other sources of presupposition, including factive verbs as in D_4, possessives as in D_3 and cleft sentences.

We call D_5 (where "There is a king" is first added to the discourse context only to be later withdrawn) a "garden path *discourse*". These are often ignored by theorists, who may be assuming that if we could just get the usual or normal sorts of (monotonically increasing) discourse right then perhaps we could go on to tackle those other ones. We believe that this divide-and-conquer approach may be misguided, that garden path problems can already be found in so-called normal discourses.

4 Heim's CCP Rules

A partial function + takes a discourse context c followed by an utterance u into a new context c′. Although we have no rigorous specification of what makes up a discourse context we can say that it should include the propositional content of what is mutually accepted by the discourse participants. For discourse D_1 above, the original context would first be updated by processing u_1, the first utterance.

c_1 + There are roses and tulips = c_2.

Presumably c_2 would entail that there are roses and that there are tulips. Next c_2 would be updated to c_3 after processing u_2, "But the roses are not yellow."

The function + is subject to the restriction that c + u is undefined unless c is a subset of (that is, entails) every presupposition of u. Thus the context for an utterance must entail all presuppositions of the utterance. We can interpret propositions as sets of possible worlds, while the utterances that give rise to and affect these propositions are tokens of natural language.

Much of the presupposition data, including the discourses D_1 through D_3, are predicted by Heim's three rules of context updating, the CCPs to which we have added a basis rule CCPB.

CCPA: c + (u and v) = (c + u) + v. [conjunction]

CCPN: c + (not u) = c \ (c + u).[3] [negation]

CCPC: c + (if u then v) = $c\backslash$ ((c + u) \ ((c +u) + v)). [conditional]

CCPB: c + u = c ∩ [[u]]. [atomic basis]

The first three rules are rewrite rules for complex utterances. Repeated application of these rules will reduce a complex utterance to a formula containing atomic utterances where rule CCPB can be applied.

Rule CCPN is an analog of a rule for logical conjunction [[P ∧¬ Q]] = [[P]] \ [[Q]]. In effect, Heim can be assumed to be proposing that, for discourse updating where the utterance is a negation, not u, we do the following: (i) Start with c and then imagine that the discourse had been u (the positive form) rather than not u— but in the same context. Next (ii) subtract the proposition that this imagined discourse would yield as its updated context from the original, real context. Heim allows contradictions to both appear and *persist*

[3] In general, p\q is the set theoretic intersection of p with the complement of q, i.e., the subtraction of q from p.

in the imagined discourse (that is, the context that will be subtracted). On this point we disagree with Heim. It happens that for some (but not all) discourses this gives the wrong result. We will show that when the CCP rules are modified and used as rules of active logic they can reasonably lead ultimately to updated contexts where contradictions have been removed and all the correct presuppositions are retained.

In the basis rule CCPB, when c + [[u]] is defined the new context is simply c ∩ [[u]],[4] where [[u]] is the proposition expressed by the utterance u. In many cases c ∩ [[u]] gives the empirically correct result. We will argue below that in other cases it gives a wrong result. Then we will propose a more realistic interpretation of +. This will be our chief modification to the theory.

5 Accommodation

Context updating becomes interesting when we consider discourses where c + u is undefined but where things can be made right through accommodation. For example, assume

c = [[Grass is green]][5]

D = ⟨ The roses are yellow. ⟩

According to the applicable rule CCPB, c + u would be infelicitous because c does not entail [[There are roses]] which is the presupposition of D. But it seems that the understander can easily accommodate to get c′ = c ∩ [[There are roses]], thus making c′ + u felicitous. We therefore allow this kind of accommodation when required in the application of the CCP rules.

We begin to encounter difficulties with the simple picture of accommodation in certain discourses involving negative existentials. In D_2, u_1 produces a context c which entails [[There is no king]]. The next sentence is

u_2 = ⟨ The king is not in hiding. ⟩

The applicable rule for this discourse is CCPN:

c + not (x is the king and x is in hiding) = c\((c + x is the king) + x is in hiding).[6]

Here accommodation is required because the first + operation to the right of the back slash is not defined; c does not entail the presupposition [[There is a king]]. If we accommodate as in the previous discourse we would add [[There is a king]] to c on the left hand side. We would then proceed to develop the right hand side where c′ replaces c. However this would yield a contradiction in the final updated context. Heim suggests using local accommodation which operates only on c in the imagined discourse to the right of the back slash. This produces the correct result. However everything to the right of the back slash becomes inconsistent. This does no harm in this case, leaving, of course, c as the final context. But we have been asked to imagine a discourse that begins with and stays with a contradiction.

Notice that D_5 is a variant of D_2 and is equally well treated à la Heim. But D_6 which resembles D_5, encounters a severe difficulty.

c_1 = [[Grass is green]]

D_6 = ⟨ There are no roses. So the roses are not in the fridge ⟩

We think that a speaker of this discourse could reasonably be taken to be asserting that there are no roses while also presupposing that there is a fridge. So there must be some manner of accommodation of the presupposition that there is a fridge (at utterance u_2). But neither global nor local accommodation will produce the correct context updating for this seemingly straightforward and simple discourse. Global accommodation yields a self contradictory interpretation of the discourse. On the other hand, local accommodation yields a consistent interpretation (as desired) but one that is missing the presupposition that there is a fridge. The applicable rule is CCPN. Local accommodation proceeds by first adding [[There are roses]] to the occurrence of c_2 in the imagined discourse to the right of the back slash as shown below.

$c_3 = c_2\backslash(((c_2 \cap$ [[There are roses]] + x are the roses) + y is the fridge) + in(x, y))

The trouble with this is that c_2 already entails [[There are no roses]] from the processing of u_1. Thus the context $c_2 \cap$ [[There are roses]] we get by accommodation is itself contradictory and so all further + operations are "felicitous" — according to Heim's definition of +.[7] Thus the final context in the imagined discourse (everything to the right of the back slash) is simply the null context, a contradiction. This leaves c_2 as the final context for the total discourse. But c_2 is simply the context as it stood after u_1; the presupposition that there is a fridge never makes it into the discourse context.[8]And what is more, there are no signals that there is anything wrong. Our diagnosis will be that something is wrong with the way we have been accommodating presuppositions. In active logic, we can add presuppositions (accomodate them) and later reason about whether they should be withdrawn. Thus where Heim's (and other presupposition theories) seem to deal with nonmonotonicity in discourse, in fact, they only allow a context to change monotonically (once the decision has been made to accomodate)

[4] Assumed by Soames [Soames, 1989] and explicitly stated by Heim [Heim, 1992] as well as by Chierchia and McConnell-Ginet [Chierchia and McConnell-Ginet, 1990]

[5] We choose [[Grass is green]] as a stand in for a context that is irrelevant to the discourse.

[6] We follow Heim by using free variable syntactic forms of sentences

[7] If a context is contradictory, it will, of course entail any presupposition of any utterance. So c + u will always be defined, that is, felicitous.

[8] The variant D_7 is similarly problematic: global accomodation leaves an inconsistent context, while local accomodation retains both the presuppositions that there is a fridge and that there are roses.

6 Our Proposal

Elsewhere (Gurney and Morreau [Gurney and Morreau, 1995]) we propose a solution to the above problem that does without the global/local distinction. It requires that a context be intersected with a maximal consistent subset of the set of potential presuppositions of u. Here we would like to consider a more far reaching and potentially more fruitful solution. This will require two commitments: (a) The CCP rules should be taken as criteria that constrain manipulation of discourse representations. If, for example, we use a version of Discourse Representation Theory [Kamp and Reyle, 1993] then the CCP rules can be made into constraints on how to use the DRS construction rules. All the manipulation is performed here; none on the CCP rules. For example CCPN would call for constructing an imaginary DRS, then perhaps negating that to form a condition in the main DRS. (b) The three CCP rewrite rules remain unchanged. And the form of the basis rule CCPB remains but with a more liberal interpretation of the partial function +.

Until now we have presumed that, when defined, the rule CCPB prescribes (possible world) set intersection, c + u = c ∩ [[u]]. Henceforth we will allow that c + u can be felicitously performed in ways other than this. For a start we will say that c + u should result in the best possible new context. In many cases intersection of c with [[u]] is the best possible new context. But in others including "garden path" discourses like D_5 as well as "rejection" dialogues like

$D_8 = \langle$ Sam: There is no king here, Sally: But I just saw the king on TV $\rangle$

intersection is not the best policy. In D_5 at u_2 the presupposition that there is a king should be withdrawn from the current context. And in D_7 Sally is attempting to at least get [[There is no king]] withdrawn and perhaps also get [[There is a king]] added to the context. These are some of the variations in context updating allowed by our new theory of +. This is still a partial function, however, which still enforces felicity as regards presupposition. We will now illustrate CCPN and CCPB in action— first with a rather simple case. Let c_1 entail that nothing is both red and green. Then

$c_1 + \langle$ Grass is green, Grass is not red $\rangle$

has a felicitous normal (consistent) interpretation. First apply CCPB to process u_1 to get c_2 which entails that grass is green. Next apply CCPN to process u_2:

$c_2 + \langle\text{Grass is not red}\rangle = c_2 \setminus (c_2 + \langle\text{Grass is red}\rangle)$.

On the old interpretation of + we get a contradiction to the right of the back slash, leaving c_2 (which simply entails [[Grass is green]] as well as [[Nothing is both red and green]]) as the final context c_3. This may be the most natural correct final result. But on the new interpretation of + there are other possibilities for $c_2 + \langle\text{Grass is red}\rangle$ in the imagined discourse. Treating this as a possible discourse in itself, one should want to achieve a best result, something other than a contradiction. We can either remove [[Grass is green]] from c_2 or simply reject [[Grass is red]]. The latter would make the total discourse a contradiction, not a good result. The former would in this case yield the same result as the old way. So we see that allowing contradictions to appear on the right of the back slash is not necessary. There is another way that produces the same result but also is potentially more fruitful.

Similar strategies allow us to use the CCP rules to account for the troublesome D_6, as well as the garden path discourse D_5. Perhaps D_4 and others involving conversational implicature can also be accounted for using the revised + function. Several of our examples use 'so', 'because', and other rhetorical clue words. This information should also be applied in the decision to find the best possible update c + u.

If the structures of DRT are used to represent the discourse, then the CCP rules should be taken as principles that govern legal manipulations of the DRSs. We would like to regard the CCP rules in some way analogously to the semantic criterion for a valid rule of inference. The latter states that an inference rule must preserve truth, that it be impossible to generate a false conclusion from true premisses. But our discourse rules are more complex. They do place some restrictions on whatever algorithms or heuristics one uses to interpret a series of utterances in a discourse.

It is our view that dialog-understanding is best seen as on ongoing form of reasoning in which conclusions are reached and (perhaps) later relinquished in light of additional information. This is more than plain-and-simple nonmonotonicity, for we have in mind that there is no static theory (set of axioms) valid for even a single sentence of a discourse. Rather, the reasoning that supports discourse-understanding constantly evolves, constantly re-assesses its own conclusions, that the context is constantly changing (at least with each new word uttered). This is the basis for our hope that active logic may provide a useful basis for the formalization of the above ideas.

Thus, in D_5 above it may first be inferred that there is a king because nothing in the dialog *so far* (i.e., in the current context) rules out that potential presupposition, even though a few seconds later the evolving dialog may provide a new context that does rule this out. It is crucial to this approach that the belief set (set of conclusions) is at every moment finite, consisting of what has been proven (and not rejected) up til *now*, where the notion of *now* is indexed to a clock that advances as reasoning (and dialog) proceeds.

7 Applying Active Logic to Discourse

Active logic has generally been used to represent the *beliefs* of an agent at a given step (normally a given time). It is assumed that a finite set of FOPC like formulae can represent this belief state. A change in the belief state of the agent yields a new step, that is, a new set of formulae. In this paper we will adapt active logic for the purpose of understanding discourse. For this we will be concerned primarily with formulae that represent the *discourse context*, that is, the record of what has been said up to the current step. At step n the information state (context, etc.) might look something like

Step n: ctxt([...], n)

[...] is an ordered list of the agent's model of the discourse context.

Although some of the agent's other beliefs will normally change as the discourse unfolds we will ignore this possibility and only represent beliefs about the discourse in our examples.

Here we introduce the predicates and rules used in our active logic.

7.1 Predicates used.

1. *now(t)* indicates that we are now at the t^{th} step of computation.
2. *ctxt(c, t)* represents that the context at time t consists of the list c of formulae.
3. *ut('X', t)* represents that X has been uttered at time t.
4. *parse(X, t)* is the parse obtained at time t by processing an utterance at the previous step.
5. *dfnt(X)* represents a definite description in the utterance. (This is produced by the parser).
6. *update(X, t)* represents at time t, elements of the discourse that still need to be incorporated into the context according to Heim's rules. X is a list of contexts, atoms from the inputs and the + and \ operators. In the subsequent presentation of rules and active logic steps, + will be denoted as PLUS, and \ as SLASH.
7. *presup(X)* marks X as a presupposition in the context.
8. *exists(x, P(x))* indicates that an object with property P exists in the discourse context.[9]
9. *assert(X)* marks X as having been asserted in an utterance.
10. *contra(X, Y, t)* indicates that there is a contradiction between the formulae X and Y in the context at time $t-1$.
11. *NULL(X)* indicates that formula X is not to be "trusted".
12. *SUSPECT(X)* indicates that formula X has given rise to a contradiction.

[9] Our use of "exists" here is not the usual logical use with narrow scope. Rather, it has wider scope as used in DRT and by Heim.

7.2 Rules of inference used.

The rules will be presented in the form:

i: X
i+1: Y

If X is believed at step i, then Y is added to the beliefs at step i+1. Nothing else is added to the beliefs that is not mentioned by these rules.

1. i: ut('X', i)
   i+1: parse(Y, i+1)

   where Y is a parse of X.
2. i: ctxt(C, i) parse(X, i)
   i+1: update(Z, i+1)

   Z is a list of operators and operands such that successively applying the operators to their operands results in updating the context with the parsed input utterance according to Heim's rules (CCPA, CCPN, CCPC).
3. i: update(X, i)
   i+1: update(Y, i+1)

   where Y is the result of applying the first operator (PLUS or SLASH) in the list Y to its arguments. There are several cases depending on the operator and on the form of the operands. These are not detailed here for brevity.
4. i: update(X, i)
   i+1: ctxt(X, i+1)

   this rule is a subcase of the previous and is applied when all context updating is complete for one particular utterance. Once the update is complete, the new context is put back into the set of beliefs of the system.
5. i: ctxt([..., foo(X), ..., bar(not(Y)), ...], i)
   i+1: ctxt([..., SUSPECT(foo(X), ...,
   SUSPECT(bar(not(Y))), ...,
   contra(foo(X), bar(not(Y)))], i+1)

   This rule detects direct contradictions in the context. Here, X and Y are unifiable and *foo* and *bar* are either *assert* or *presup*. Note that *foo(X)* and *bar(not(X))* are tagged as being "suspect" at *i+1*.
6. i: ctxt([..., SUSPECT(foo(X), ...,
   SUSPECT(bar(not(Y))), ...,
   contra(foo(X), bar(not(Y)))], i+1)
   i+1: ctxt(Z, i+1)

   Z is the context resulting from resolving the contradiction flagged at step i. The contradiction can be resolved by using various additional sources of information including:[10]

[10] See Miller [Miller, 1992] for more on contradiction resolution in active logic

- Other elements in the context, for example rhetorical relations, formulae in the context relevant to the contradictands, the sequence of inferences leading to the derivation of the contradictands.
- General knowledge which may be outside the context (though we do not treat this here).
- The status of the contradictands— whether they are assertions, presuppositions or distrusted.

Resolving the contradiction can result in one or both of the formulae being distrusted, and in further changes in the context. Note that the resolution of a contradiction is itself defeasible—this resolution could later lead to other contradictions which could undo the changes done at this point.

7. i: ctxt(X, i)
   i+1: ctxt(X, i+1)

   We simply inherit the context to the next step if there is no change.

8. i: ctxt(X, i), ctxt(Y, i)
   i+1: ctxt(X ∪ Y, i+1)

   Note that taking the union of the 2 contexts could introduce contradictions in the total context. That will be detected at the next step.

9. i: now(i)
   i+1: now(i+1)

   This is the "clock rule". Time does not stand still while we are reasoning.

8 Steps Galore

We consider only discourses that depend on 'but', 'so', 'because', overtly. We will treat them as intersentential relevance markers.

The first example

We now present our first example.[11]

D_1 = ⟨There are roses and tulips. But the roses are not yellow⟩

Step

0 ctxt([], 0),ut('There are roses and tulips')

Let c_1 = [].[12]

1 c_1, parse(and(exists(x,R(x)),exists(y,T(y))))

This is the result of parsing the utterance and inheriting the previous context.

2 c_1, update([c_1,exists(x,R(x)),PLUS,
exists(y,T(y)),PLUS])

[11]Some details are not shown, for example the argument representing time in the predicates.

[12]We will use c_i for both the list of formulae in the context and for the predicat ctxt(c_i, j). Which is meant will be evident from the context.

update is the result of applying Heim's rules recursively to the parsed utterance. Note that this is in postfix form, however.

3 c_1, update([c_2,exists(y,T(y)),PLUS])

where c_2 = [assert(exists(x,R(x)))]

The first operation is (c_1,exists(x,R(x)),PLUS). We just assert the new atom into the context.

4 c_1,update(c_3)

where $c_3 = c_2 \cup$ assert(exists(y,T(y)))

We assert the second part of the utterance into the context too.

⋮

7 c_3

At the end of processing the first utterance, the context contains the assertions that there are both roses and tulips in the discourse context. We now add the next utterance.

8 c_3, ut('But the roses are not yellow')

9 c_3, parse(and(but,not(and(dfnt(R(z)),Y(z)))))

The new utterance has been parsed and we now need to incorporate it into the context.

10 c_3, update(c_3,but,PLUS,c_3,but,PLUS,
dfnt(R(z)),PLUS,Y(z), PLUS, SLASH])

⋮

12 c_3,update(c_4,c_4,dfnt(R(z)),PLUS,Y(z),
PLUS,SLASH])

where $c_4 = c_3 \cup$ assert(but)

Now we have to add "the roses" to the context. We search in the context for roses that were previously mentioned the closest to the present time—i.e., a mention of roses closest to the tail of the list.

13 c_3,update(c_4,c_5,Y(z),PLUS,SLASH)

where $c_5 = c_4 \cup$ assert(x=z)

We have in fact mentioned roses before, and we make the new mention of roses designate the same roses as the previous mention by asserting x=z.

14 c_3, update(c_4, c_6,SLASH)

where $c_6 = c_5 \cup$ Y(z)

15 c_3, update($c_4 \cup$ not(and(assert(x=z),assert(Y(z))))

Set difference between the two contexts is done by adding to the first context the negation of the elements in the second context but not in the first.

Here we have a choice of what to negate: either that x=z or that z are yellow, or both. It is at this point that we make appeal to rhetorical information in *but* to help us make the best choice. If we make the right choice here, which seems to be not to doubt that the roses mentioned in the second utterance are the same roses mentioned in the first, we get eventually:

ctxt([assert(exists(x,R(x))),assert(exists(y,T(y))),
assert(but),not(and(assert(x=z),assert(Y(z))),
assert(not(Y(z))),assert(x=z).

We could of course have made a bad choice here. Had that happened, it could have led to a contradiction later on in the discourse and the choice we made at this point would then be questioned.

A second example

We now show an example of active logic using the rules and predicates discussed above to a garden path sentence. This is essentially D_7.

D_9 = ⟨John bought flowers, [Are the roses in the fridge?], No, (the roses are not in the fridge), Because there were no roses⟩

Here we have a case where something is first added to the discourse context only to be promptly removed. As presented in [Heim, 1983], the CCP rules cannot deal with this discourse. The last sentence would simply produce a null set of possible worlds, making anything that followed "felicitous."

To save space we will consider only the shortened discourse D_7:

D_7 = ⟨The roses are not in the fridge. Because there are no roses.⟩

Step

0 ctxt([],0) ut('The roses are not in the fridge')

Let c_1 = [].

1 c_1, parse(not(and(dfnt(R(x)),dfnt(F(y)),in(x, y))))

This is the result of parsing the utterance and inheriting the previous context.

2 c_1, update([c_1, c_1,dfnt(R(x)),PLUS,dfnt(F(y)), PLUS, in(x,y), PLUS, SLASH])

We get the update predicate by applying Heim's rules as before.

3 c_1, update([c_2,c_2,dfnt(F(y)),PLUS,in(x,y), PLUS,SLASH])

where c_2 = [presup(exists(x,R(x)))]

The first operation is (c_2, dfnt(F(y)), PLUS). Since we have a definite descriptor, we first search the previous context (c_1) for a previous mention of roses. As there is none, we accomodate (globally) the context with the presupposition that there are roses.

4 c_1, update([c_3, c_3,in(x,y),PLUS,SLASH])

where c_3= [presup(exists(x,R(x))),
presup(exists(y,F(y)))]

Similarly, we accomodate by adding the presupposition that there is a fridge to our context.

5 c_1, update([c_3, c_3 ∪ [assert(in(x,y))],SLASH]

We simply assert that the roses are in the fridge. Note that here, only one context is being updated. We do not add the new assertion globally.

6 c_1, update([c_3 ∪ [not(assert(in(x,y)))]])

⋮

11 c_4, ut('Because there are no roses')

where c_4= c_3 ∪ [not(assert(in(x,y)))]
= [presup(exists(x,R(x))),
presup(exists(y,F(y)))
assert(not(in(x, y)))])

After some processing, we end up with a new context that contains the presuppositions that there are roses and a fridge and that the roses are in the fridge. To this new context, the second utterance is added.

12 c_4,parse(and(because,not(exists(z,R(z)))))

We repeat the processing we did above.

13 c_4, update(c_4,because,PLUS,c_4,because,PLUS, exists(Z,R(z)),PLUS,SLASH)

⋮

17 c_4, update(c_5)

where c_5= c_4 ∪ [assert(because),
not(assert(exists(z,R(z))))]

We have asserted "because" in the context because "because" can serve as a clue to picking the right choice among several alternatives we could encounter in later processing.

⋮

21 ctxt([presup(exists(x,R(x))),presup(y,F(y)),
asssert(not(in(x,y))),assert(because),
assert(not(exists(z,R(z))))

We now have a context which presupposes that there are roses and which asserts that there are none.

22 ctxt([SUSPECT(exists(x,R(x))),presup(y,F(y)),
asssert(not(in(x,y))),assert(because),
SUSPECT(not(exists(z,R(z))))
contra(presup(exists(x,R(x))),
assert(not(exists(z,R(z))))]

That contradiction is detected and flagged. The formulae that caused the contradiction appear at this step flagged as being "suspect".

23 ctxt([NULL(exists(x,R(x))),presup(y,F(y)),
asssert(not(in(x,y))),assert(because),
assert(not(exists(z,R(z))))

Using the fact that one of the contradictands was a presupposition and the other an assertion we conclude that we were mistaken about the presupposition. This is not always the right decision but it is a good heuristic. We make the presupposition that roses exist to be suspect and we reinstate the assertion that roses do not exist (in the discourse context).

24 ctxt([NULL(exists(x,R(x))),presup(y,F(y)),
asssert(NULL(in(x,y))),assert(because),
assert(not(exists(z,R(z))))

Since we doubt the existence of roses, we doubt the truthfulness of the statement that the roses are in the fridge.

At the end of processing D_7, we "know" the following:

- There is a fridge.
- There are no roses.

And we have doubts about the following:

- There are roses.
- The roses are in the fridge.

And this is what we expect to get.

9 Conclusion

In conclusion, we have shown that active logic can be applied to the problem of updating according to the + function in Heim's system of rules for discourse context. Heim's rules account for certain effects of complex structure in utterances. And active logic accounts for the problem of how to alter a given context when processing simple atomic utterances that appear in the rewritten forms that utterances take after application of the three CCP rewrite rules. In this way the resources of active logic can be brought to bear on some of the complexity of natural language discourse processing. Our long-range goal in this work is the design and implementation of a time-situated natural-language discourse-understanding system based on a formal theory of pragmatic reasoning.

Acknowledgements

This research was supported in part by the National Science Foundation and the Army Research Office.

References

[Chierchia and McConnell-Ginet, 1990] G. Chierchia and S. McConnell-Ginet. *Meaning and Grammar.* MIT Press, Cambridge, Ma, 1990.

[Crain and Steedman, 1985] Crain and M. Steedman. On not being led up the garden path. In R. Dowty, D., L. Kartunnen, and A. Zwicky, editors, *Natural Language Parsing.* Cambridge University Press, New York, 1985.

[Elgot-Drapkin and Perlis, 1990] J. Elgot-Drapkin and D. Perlis. Reasoning situated in time I: Basic concepts. *Journal of Experimental and Theoretical Artificial Intelligence*, 2(1):75–98, 1990.

[Gurney and Morreau, 1995] J. Gurney and M. Morreau. Presupposition and the concept of a non-monotonic discourse. Utrecht, 1995. International Workshop on Nonmonotonic Reasoning.

[Heim, 1983] I. Heim. On the projection problem for presuppositions. In S. Davis, editor, *Pragmatics.* Oxford, 1983.

[Heim, 1992] I. Heim. Presupposition projection and the semantics of attitude verbs. *Journal of Semantics*, 9:183–221, 1992.

[Kamp and Reyle, 1993] H. Kamp and U. Reyle. *From Discourse to Logic.* Kluwer, Dordrecht, 1993.

[Miller, 1992] M. Miller. *Reasoning about One's own mistaken beliefs.* PhD thesis, Department of Computer Science, University of Maryland, College Park, Maryland, 1992.

[Perlis and Miller, 1993] D. Perlis and M. Miller. Presentations and this and that: Logic in action. Boulder, Colorado, 1993. Cog-Sci-93.

[Soames, 1989] S. Soames. *Handbook of Philosophical Logic*, volume IV, chapter Presuppositions. Reidel, 1989.