Handling Speech Repairs and Other Disruptions Through Parser Metarules*

Mark G. Core and Lenhart K. Schubert
Department of Computer Science
University of Rochester
Rochester, NY 14627
mcore,schubert@cs.rochester.edu

Abstract

Mixed-initiative dialogs often contain interruptions in phrase structure such as repairs and backchannel responses. Phrase structure as traditionally defined does not accommodate such phenomena, so it is not surprising that phrase structure parsers are ill-equipped to handle them. This paper presents metarules that specify how phrase structure rules may be restarted or interrupted (including overlapping speech). In the case of overlapping speech or a backchannel response, the metarules allow a constituent to overlap or be embedded inside another constituent that it is unconnected to. In the case of repairs, the metarules operate on the reparandum (what is being repaired) and alteration (the correction) to build parallel phrase structure trees: one with the reparandum and one with the alteration. Consider the utterance, *take the ban- um the oranges*. The repair metarule would build two VPs, one being *take the ban-* and the other being *take the oranges*. The introduction of metarules simplifies the notion of an *utterance* since a sentence interrupted by an acknowledgment such as *okay* can still be one utterance formed around the interrupting acknowledgment. Together metarules and phrase structure rules specify the structures that should be accommodated by a parser for mixed initiative dialogs. A dialog parser should also maintain a dialog chart that stores the results of syntactic and semantic analysis of all of the dialog seen so far. This dialog chart will be a shared resource eliminating the need for maintenance of a separate representation of dialog structure by a dialog manager. In addition, the dialog parser can alert the dialog manager to utterances introducing obligations as well as recognizing acknowledgments and responses based on syntactic information.

Motivation

Repairs and other interruptions of phrase structure (such as acknowledgments) are common in mixed-initiative dialogs; (Core & Schubert 1996) discusses a preliminary dialog parser that accommodates these interruptions through metarules, however these metarules operate by creating additional grammar rules. This paper presents metarules that license certain extra-grammatical, interleaved structures. They either allow the parser to skip over certain material (parsing this material separately) or in the case of repairs, construct alternative representations (one for the erroneous material and one for the corrected material). Treating repairs and other disruptions of phrase structure as extra-grammatical simplifies the definition of "utterance" since interruptions do not necessarily terminate an utterance. We will use the term "utterance" to refer informally to a sentence or a single phrase uttered by a speaker, allowing for repairs and acknowledgments.

Backchannel responses are disruptions (in the middle of an utterance by another speaker) such as *okay* or *right* that act to signal understanding. Parser metarules will handle backchannel responses and repairs by other speakers, and mark them so the dialog manager understands that they are temporary interruptions that do not seize the initiative. Acceptances are syntactically similar to backchannel responses but generally occur at the end of utterances and serve to ground the previous statements. Grounding is the achievement of mutual understanding or mutual acceptance of a plan (Clark & Schaefer 1989). The dialog parser can recognize these acceptances as well as backchannel responses aiding the dialog manager in its work. The parser can also roughly follow the initiative of a dialog since it knows that after an acknowledgment, acceptance, or answer the initiative may be seized or the first speaker may continue with more questions, suggestions, or requests. A dialog manager actively participating in a dialog needs this information in order to decide when to speak. (The term *dialog manager* is used to describe a higher-level reasoning module that follows the grounding and flow of initiative.)

Most dialog systems do actively participate with a user in a dialog and use their dialog manager to orchestrate the understanding of the dialog as well as producing appropriate responses. However, dialogs may also be listened to by a third party, and such a listener

*The official version of this paper has been published by the American Association for Artificial Intelligence (http://www.aaai.org)

should understand the same phenomena that a dialog participant deals with (grounding and turn management). The dialog parser described in this paper is meant to serve either goal; the text of a second speaker may be a transcription or output from a speech recognizer or may be generated by the dialog system itself.

Previous Work

Little work has been done on including speech repairs in a parser's output. Removing repairs before they reach the parser or having the parser skip over such material are reasonable as rough-and-ready expedients, but neglect the important role such segments can play in the dialog structure. For example, repairs can contain referents that are needed to interpret subsequent text (*e.g., Take the oranges to Elmira, uh, I mean, take* them *to Corning*). The problem of detecting and isolating repairs from fluent speech is difficult and one that must be faced no matter how the repair is accommodated. Some of the previous work in this area assumes that a repair can be detected through prosodic cues or ungrammaticality and concentrates on isolating the repair through similarities in what is repaired and its correction. Other research focuses on repair detection since prosodic cues and grammatical cues by themselves are inadequate detectors of speech repairs.

In discussing repairs, terms from (Heeman 1996) will be used to describe the various parts of a repair. The *interruption point* (ip) occurs just before any editing terms (*um* in the example below) and right after the repaired information or *reparandum* (in this case, *at*) ends. The *alteration* is the correction (in this case, *ready at*) and ends at the last word matching a word in the reparandum or if no words match then the alteration is the first word following the editing terms of the repair.[1]

```
s: we'd have the tanker of orange juice
    at         um ready at   Elmira
    |       ip et |------|
reparandum          alteration
```

Isolating the Repair

(Hindle 1983) uses pattern matching and grammatical information to isolate repairs once they are detected. The pattern matching is a preprocessing step to isolate repeated segments of text (separated by an editing term) as repairs. Grammatical information is used during parsing to classify two constituents of the same syntactic category separated by an editing term as a reparandum and alteration.

(Kikui & Morimoto 1994) also concentrates on finding a reparandum and alteration assuming that the presence of an interruption point has already been determined. They use an algorithm originally designed for analyzing coordinate structures to find a reparandum and alteration with the most matching elements. In addition, they test the well-formedness of the utterances remaining after various potential reparanda are removed. These tests use a language model called an adjacency matrix that specifies what categories can follow one another.

One of the first comprehensive studies of speech repairs (Levelt 1983) noted that speech repairs seem to be well-formed if valid corresponding coordinate structures can be constructed. In particular, Levelt suggested that if a repair's reparandum was suitably completed and followed by the conjunction *and* then the utterance should be grammatical. In the example below, the test would involve completing the reparandum (e.g. *picks up the tankers*), inserting *and* and leaving the rest of the utterance *takes the two boxcars at Elmira to Corning*. Thus, the example below is a legal speech repair by Levelt's rule.

```
engine E two picks um takes the two boxcars
 at Elmira to Corning
```

Levelt was an early advocate of using pattern matching to identify the reparandum and alteration, and showed that 48% of his 959 speech repair examples could be handled by two simple pattern matching rules (50% of the time these rules would not apply and 2% of the time they gave incorrect answers.).

Detecting Repairs

As pointed out in (Heeman 1996), neither editing terms, word fragments, nor prosodic clues by themselves are a reliable signal for speech repairs. Bear et al. (Bear, Dowding, & Shriberg 1992) discuss how their parser detects and removes reparanda without assuming a premarked interruption point and instead uses pattern matching to identify likely repairs. To avoid false positives (initially 38% of the posited reparanda were false positives), the parser was run before any pattern matching was attempted. This parser applies both syntactic and semantic constraints to mark sentences as ungrammatical and hence as likely to contain speech repairs. The parser searches ungrammatical sentences for reparanda and if the sentence without the reparandum is grammatical, success is declared. Because 29%-50% of the time the parser is not able to parse a sentence with or without a posited reparandum, Bear et al. ran the parser only on the VP, NP, or PP that possibly contained a reparandum. This approach performed well on the 37 test cases but it remains to be seen how it would do on more data.

To identify repairs, Nakatani and Hirschberg (Nakatani & Hirschberg 1993) used a variety of cues such as the existence of a lull or word fragment, word repetition, and changes in speech signal energy and fundamental frequency. They used a decision tree to

[1](Heeman 1996) adapts this notation from (Levelt 1983) making slight changes in the definitions similar to those made by (Shriberg 1994) and (Nakatani & Hirschberg 1993).

combine evidence from these sources to achieve 83.4% recall and 93.9% precision.

(Heeman 1996) describes a statistical model that assigns part of speech tags and identifies speech repairs based on context (previous editing terms, boundary tones, pauses, and the previous words and their categories). To determine the extent of the repair, word matching plays a role as do the boundary tones, words and word categories before possible referandum beginnings. Using the two approaches gives a recall of 74% and precision of 78%. These results appear low when compared to (Nakatani & Hirschberg 1993); however Nakatani and Hirschberg used test utterances having at least one repair while Heeman did not remove sentences with no speech repairs from the test set.

These various studies show that multiple sources of information will be necessary in processing text where repairs are not already marked. The success of (Nakatani & Hirschberg 1993) and (Heeman 1996) in locating repairs suggests that the speech recognizer should be modified to search for acoustic signals of repairs and that the dialog parser should consider boundary tones and pauses. In addition to acoustic information, a parser-level approach similar to (Hindle 1983) will be needed because of the extra information it can access, namely the patterns of phrase categories present in the input and how these phrases may differ structurally.

The alternative to a parser-level approach might be to remove the reparandum in a pre-processing step, but that would neglect not only the grammatical cues to the structure of the repair but also the role the reparandum can play in dialog structure, such as providing referents needed to interpret subsequent text. The reparandum is also a legitimate part of the semantic content of an utterance; the other speaker hears the reparandum and may refer to it later, *you were right the first time ....*

Metarules

To handle the disfluencies in mixed-initiative dialog caused by repairs, hesitations, and acknowledgments, the dialog parser will use metarules allowing the interleaving of certain phrase structure trees. The *hesitation metarule* allows hesitations such as the one seen below where *s* hesitates in the middle of an utterance and inserts a *lull* (in this case, *um*). For the purposes of the preliminary dialog parser, lulls are defined as pauses, commas, and editing terms (*uh, um, hmm, let's see,* and *I mean*).

```
utt: s: we'll have to get an engine to Dansville
        um to pick up the boxcars
```

In general, if the grammar allows XP to be formed by a series of constituents Y_1 - Y_N ($N \geq 1$), the hesitation metarule allows XP to be formed even though its subconstituents are interrupted in 1 to N-1 places

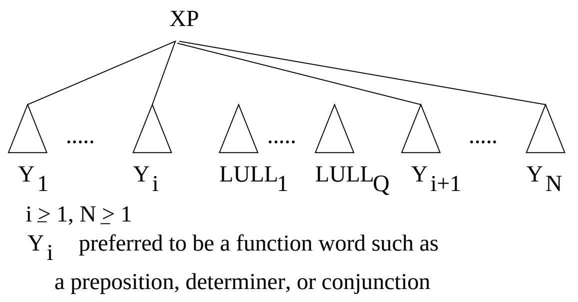


Figure 1: Phrase structure of an interruption allowed by the hesitation metarule

by lulls. The acceptable form of each interruption is shown in Figure 1. Note that this metarule may have already been applied to one or more of the subconstituents, Y_1-Y_N. The hesitation metarule may also have been applied to one or more of the lulls, $LULL_1$-$LULL_Q$; for example, *I mean* may be interrupted by another lull such as *um*: *I um mean.* In all these cases, XP excludes these lulls as well.

The problem-solving dialogs (Heeman & Allen 1995) used in this study, the TRAINS dialogs, involve railway transportation. Most of these dialogs contain overlapping speech, much of it in the form of backchannel responses such as acknowledgments and *continuation prompts.* We define continuation prompts as short utterances such as *hm-mm* or *okay* that prompt a speaker to continue. Both continuation prompts and acknowledgments may be formed from words such as *right* and *okay* while acknowledgments may also be formed from repetitions of what was previously said.

One way for a parser to deal with these responses would be to treat them as part of the main word stream, as in the case of hesitations (but of course marked as second-speaker contributions). However, coercing the contributions of both speakers into a single word stream is in general ill-advised since the contributions often overlap even at the word level. We also note that for backchannel responses there can be some discrepancy between their actual placement and their "intended" placement; i.e., the speaker may have been slow in uttering a response. Thus we opt instead for separate *tracks* for the two speakers. Determining exactly what prior material a backchannel response refers to will be a task for higher-level dialog analysis. Of course, the relative location of the response will provide an essential clue. For the most part, the parser will process the phrase structure of both tracks separately, so backchannel responses will not disrupt the phrase structure of the utterance during which they occur.

Sometimes though, one speaker will finish another's sentence, so the *continuation metarule* allows a constituent to be formed across tracks. The continuation metarule allows the extension of an incomplete phrase

A = s: with the other engine uh we're going to Corning to Dansville
to pick up the other boxcars
B = u: and then back to Corning
C = s: then back to Corning
D = u: to get the oranges
E = s: then we're loading
F = u: and then to Bath
G = s: we're loading the oranges and then we're going
there and that will take seven hours
H = u: okay

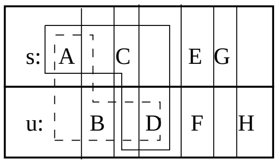


Figure 2: An example of collaborative speech

Y by speaker A with an immediately following (and possibly incomplete) phrase Z by speaker B. Empirical data will be needed to determine how to define "immediately following" as Z may start a little before Y ends, or there may be an intervening hesitation. Notice the metarule is defined so that a constituent may be formed by several rounds of cooperation although that is rare. This metarule would have a low probability to discourage its frequent use.

The example in Figure 2 shows a chart with two tracks for the two speakers and how the continuation metarule links them together. Assume both A and C and A and B form syntactic constituents. The continuation metarule allows A and B to form a constituent P, even though A and C also form a constituent Q. The parser can extend P with D, and by the continuation metarule, Q can be extended by D. Since both speakers are talking at the same time for E and F, it is unlikely that either fully understood the other, so the parser should probably not extend the constituents containing D with E or F. However, both E and F need to be parsed as G refers to the NP, *Bath* uttered in F.

The metarules are concerned solely with syntactic structure so if a speaker goes off on a tangent this does not concern the metarules. However, if speaker A interjects a comment, correction, or question while speaker B is talking then 1) speaker B may acknowledge or answer the interruption, 2) A may make an acknowledgment or continuation prompt, and 3) B may resume his interrupted utterance. In the TRAINS dialogs, interruptions occurred mostly with an interjected repair by another speaker as shown in Figure 7. However, we do not want to rule out dialogs such as the constructed dialog below where a second speaker makes a brief interruption (*oh e1*) and is answered in some way (*right*), followed by the original speaker resuming his utterance without a repair occurring.

```
u: then e1 will have
s: oh e1
u: right
   two boxcars of oranges
```

So the *interruption metarule* is used to deal with interjected corrections, questions, and comments separately from any repair that may follow. The desired structure for the interruption example above is shown in Figure 3. Since the interruption is by *s*, it is on a different track than *u*'s utterance and no special action is needed to allow the parser to form a VP around it. The metarule's action is to allow the VP to skip *u*'s acknowledgment, *right*.

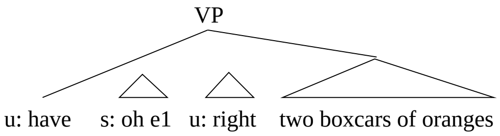
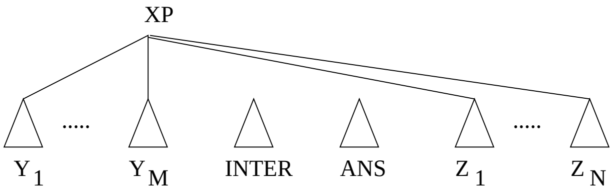


Figure 3: Structure of an interruption



$M \geq 1$, $N \geq 1$, prefer Y_M to have a prosodically marked boundary

INTER is usually a question or a phrase or declarative sentence said in a questioning manner by another speaker

ANS is an answer , acknowledgment, or acceptance of INTER

Figure 4: Definition of interruption metarule

The hesitation metarule could handle this example but gives the wrong interpretation to the utterance. If the dialog manager saw that the hesitation metarule was used here, it would not see that *s*'s interruption had more content than a backchannel response such as *mm-hm*. In addition, as the general form of the interruption metarule (Figure 4) shows, *u* may utter something that is not a valid lull such as *two hours* in answer to a question by *s*.

There are few hard constraints on this rule but for the most part, ANS[2] will be an acknowledgment or acceptance such as *okay*. When INTER is a question, ANS may also be *yes*, *no*, or an NP or PP. So the ambiguity caused by this metarule should not be overwhelming.

A more complex situation occurs when a repair is interjected by a second speaker and the first speaker accepts the repair, replacing earlier spoken material with a correction. First, consider the simpler case where speakers correct themselves as in the constructed example of Figure 5 where the speaker corrects the arguments of the verb phrase. The *repair metarule* creates two VPs from this material; one containing the reparandum (VP) and one containing the alteration (VP').

[2] ANS is just a label used in the figure and is not a syntactic category.

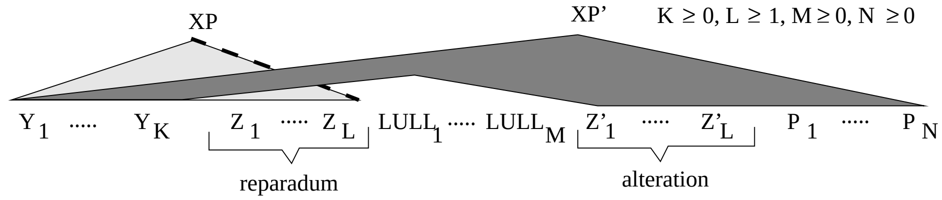


Figure 6: Definition of repair metarule

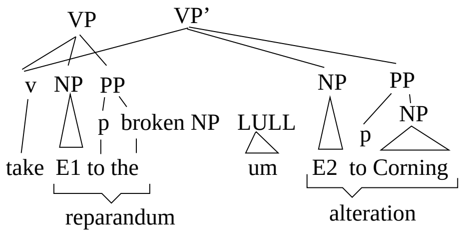


Figure 5: Sample Repair

More generally the repair metarule can be depicted as shown in Figure 6. Y_1-Y_K, Z_1-Z_L, Z'_1-Z'_L, and P_1-P_N are the words involved in the repair. XP is allowed to be *incomplete*; XP is incomplete if it is missing one or more (but not all) constituents from the right side of its phrase structure rule, or the last constituent that is present is incomplete. *Put the chairs ...* is an example of an incomplete phrase since it is missing one of its complements. The example seen in Figure 7, *move the engine at Avon engine E to* , is an incomplete VP since it ends in an incomplete PP, *to*. The vector of words Y covers the initial part of an utterance that is uncorrected and shared by the two interpretations produced by the metarule. Lulls may occur between the reparandum and alteration helping to mark the repair; however M can be 0, i.e., no lulls may mark a repair. The vector of words P contains any additional elements added to XP' that are not direct corrections of the reparandum.

Figure 6 provides no details about the inner structure of XP and XP' but with high probability the reparandum and alteration consist of a series of phrases (Q and Q') that are immediate subconstituents of XP and XP'. There is also a strong tendency for Q_i and Q'_i to be of the same type as well as for Levelt's rule to hold: if Q is suitably completed then *Q and Q'* should form an allowable coordinate structure.

The reason the definition of the repair metarule is somewhat vague is that there are exceptions to the above tendencies. Consider this example from TRAINS dialog d93-8.2: *how does the long does that take.* Here, XP is a wh-question where part of the initial wh-phrase is repaired along with all of the auxiliary verb, the subject, and the VP.

Let us look more closely at Figure 7.[3] In this utterance: 1) speaker *s* interrupts *u* to make a correction, 2) the correction is accepted by repetition, 3) *s* makes a continuation prompt (CPT), and 4) *u* starts to make a correction and then restarts it. The interruption and *u*'s acceptance are handled by the interruption metarule. The repair metarule then forms two nouns from the word fragment *en-* and the noun *engine*. The initial reparandum that *s* was addressing is handled by a second application of the repair metarule forming a VP including the reparandum and two VPs matching the two alterations produced by the first application of the repair metarule. The continuation prompt is not on *u*'s track so it does not have to be handled through a metarule.

The metarules must be applied in some order. Here, we applied the interruption metarule first, then repaired *engine* and finally repaired *engine E to.* Repairing *engine* early makes sense as its alteration is grammatical even before the other two disruptions are addressed. The order of application of the other two metarules does not matter but the unnecessary ambiguity of having both orderings on the chart should be avoided.

Since mid-utterance repairs by other speakers are interruptions, the parser should recognize that the metarule it applies first will not fully license the repaired phrase as either the repair or the interruption will still be unaccounted for. So the definitions above need to include the proviso that a combination of metarules may be needed to account for the disruptions in an utterance.

Dialog Management

In the TRAINS 93 dialog system as described in (Traum 1993), the parser sent the logical form for each utterance to a dialog manager that maintained its own representation of the discourse. It makes more sense though for the parser to add the syntax and semantics of utterances to a *dialog chart* that it shares with the dialog manager. In addition, the parser can help the dialog manager by classifying utterances into surface

[3] In an alternative analysis VP' and VP" would not include *the engine at Avon* but the parser will prefer not to form such a structure in the case of appositives because if the speaker does not correct the first half of the appositive then s/he probably meant to convey that information.

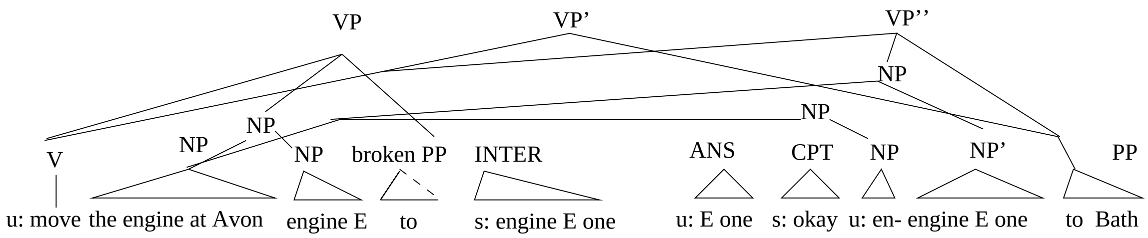


Figure 7: Structure of an exchange from d91-6.1

speech acts (locutionary acts) and sometimes illocutionary acts, and by offering hypotheses about who has the initiative.

In the case of the disfluencies and interruptions whose structure we described above, the parser can classify them into repairs, acknowledgments and continuation prompts based on their syntactic structure and other superficial characteristics. (We regard these as illocutionary categories; e.g., *right* is superficially a declarative utterance but may function as an acknowledgment, acceptance, or answer to a question.) Acknowledgments and continuation prompts can often be distinguished based on their point of occurrence (in mid-clause or at the end of a clause) and on their conventional expression (*okay*, *right*). In the case of a repair by a second speaker, the parser will be able to recognize the short correction subdialog and the resumption of the utterance by the original speaker. In all these cases, the parser can also recognize that there is no change in initiative, despite the change in turn.

The parser can recognize interrogative, imperative, and declarative utterances and with the help of additional syntactic cues, can often classify them as questions, suggestions, requests, and informs. For example, a second-person interrogative utterance containing the adverb *please* is usually a request, and a *let's*-imperative is usually a suggestion. These classifications also help in keeping track of the initiative in the dialog. Questions usually yield the turn but typically do not yield the initiative in a dialog, i.e., often the questioner will continue to hold the initiative after the question has been answered.

In the case of informs, suggestions, and requests, a speaker can contribute several utterances of this type in one turn, but eventually the speaker expects an acknowledgment or acceptance of some sort and may even end a suggestion or request with *right?* indicating the desire for feedback. As in the case of a question, the original speaker will often continue with another suggestion or request after receiving an acknowledgment or acceptance.

For the most part, a request, suggestion, or inform can be used to seize the initiative although as the example of Figure 2 shows that is not always the case. *s* keeps initiative by immediately acknowledging *u*'s suggestion. *u* starts another suggestion (*to get the oranges*) and *s* maintains the initiative by starting to speak again. Both speakers end up talking at the same time and *s* wins out and repeats what he said previously as well as finishing the current request. In this instance the dialog parser would have to make some tricky judgments based on utterance length and who ended up finishing this accepted request. But for the most part the parser should not have a problem following the initiative of the dialog. After a question is answered or a request accepted, the initiative is up for grabs. Correction subdialogs may occur after a request, inform, suggest or even a question. Thus, the dialog parser can signal the dialog manager that a suggestion, request, etc. needs to be judged or that there is a lull in the conversation and maybe a suggestion should be made.

Required Parser Modifications

(Core & Schubert 1996) discusses modifications made to a standard chart parser to allow it to handle simple hesitations, acknowledgments and repairs. However, these modifications were ultimately inadequate because they required corrections to replace immediate subconstituents of the broken phrase (unless the repair was to the rightmost subconstituent). So the dialog excerpt, *how does the long does that take*, could not be handled since only part of the wh-phrase is repaired while all of the inverted sentence is replaced. (Core & Schubert 1996) did not accommodate overlapping speech and assumed a strict ordering on the words spoken by the two speakers. The rest of this section describes future work to convert this old dialog parser to the new framework.

As a chart parser processes a sentence, it attempts to extend arcs on its chart with items from the input. Normally, the parser requires that arcs be extended by an item adjacent to the last element in the arc. When a lull constituent is formed, the parser will have to update all its arcs adjacent to that constituent so it appears as though they end after the lull. Some of this work is already done as outlined in (Core & Schubert 1996).

Chart parsers are defined as having one-track charts and are usually given one utterance at a time. Parsing is an iterative process with each word prompting a series of additions to the chart. To accommodate overlapping speech, the dialog parser can switch back and forth between its two tracks, updating them incrementally.

Change of turns signal that a second speaker may be interrupting or continuing the first speaker's sentence. If the newly started utterance is a question or short phrase (a possible correction if no question was previously asked) then the probability of an interruption increases. In such a case, an answer or acceptance of some kind by the first speaker would have to follow. Then the parser could find arcs ending at the second speaker's interruption and restart them after the first speaker's answer or acceptance.

If an interruption is a correction then the parser will have to apply the repair metarule as well. Cases of self repair are not as easy to detect as shown in the various studies summarized in the previous work section. (Heeman 1996) reports decent results in recognizing the reparandum and alteration of a repair as a preprocessing step.[4] In addition to this preprocessing step, one could imagine searching for patterns of phrases indicating a repair such as "NP PP NP PP" when the parser fails to find an interpretation for an utterance. Once the reparandum (Z) and alteration (Z') are isolated then Y (see Figure 5) can be set to be the preceding words of the utterance. The parser resets all the arcs ending just after Y_M so they start just before Z'_1.

Conclusions

Instead of trying to hide repairs and other interruptions from the parser, the proposed framework allows the parser to accommodate them and not leave out important aspects of the dialog structure when presenting the dialog manager with a representation of the dialog. In addition, this framework enables a tighter coupling between the dialog manager and parser through a shared dialog chart and by having the parser provide syntactic cues to who has the initiative and what has been grounded.

Another reason for having the parser accommodate repairs, hesitations, and interruptions is that the parser has information about the syntactic structure of the utterance and what are allowable structures. These are sources of information that preprocessing routines do not have, and the dialog parser can still use acoustic cues, pattern matching, and other sources of information that preprocessing techniques use.

In addition to the promise of a parser specially tailored to processing dialogs, this work contributes a formal description of the structure of mixed initiative-dialog - with all its interruptions, interleaving, and repairs. This description finesses the problem of what an "utterance" is by using traditional phrase structure as our theory of grammatical structure and using metarules to model the disruption of this structure in dialogs.

[4] A precision of 78% is reported for simple modification repairs and 50.8% for complete restarts.

Acknowledgments

This work was supported in part by National Science Foundation grant IRI-9503312. Peter Heeman's suggestions and help with the TRAINS dialogs are gratefully acknowledged. Thanks also to James Allen for his helpful comments.

References

Bear, J.; Dowding, J.; and Shriberg, E. 1992. Integrating multiple knowledge sources for detection and correction of repairs in human-computer dialog. In *Proc. of the 30th annual meeting of the Association for Computational Linguistics (ACL-92)*, 56–63.

Clark, H. H., and Schaefer, E. F. 1989. Contributing to discourse. *Cognitive Science* 13:259–294.

Core, M. G., and Schubert, L. K. 1996. dialog parsing in the TRAINS system. Technical Report 612, Department of Computer Science, University of Rochester, Rochester, NY 14627-0226.

Heeman, P., and Allen, J. 1995. the TRAINS 93 dialogues. TRAINS Technical Note 94-2, Department of Computer Science, University of Rochester, Rochester, NY 14627-0226.

Heeman, P. A. 1996. *Speech Repairs, Intonational Boundaries and Discourse Markers: Modeling Speakers' Utterances in Spoken Dialog*. Ph.D. Dissertation, Department of Computer Science, University of Rochester.

Hindle, D. 1983. Deterministic parsing of syntactic non-fluencies. In *Proc. of the 21st annual meeting of the Association for Computational Linguistics (ACL-83)*, 123–128.

Kikui, G., and Morimoto, T. 1994. Similarity-based identification of repairs in Japanese spoken language. In *Proceedings of the 3rd International Conference on Spoken Language Processing (ICSLP-94)*, 915–918.

Levelt, W. J. M. 1983. Monitoring and self-repair in speech. *Cognitive Science* 14:41–104.

Nakatani, C., and Hirschberg, J. 1993. A speech-first model for repair detection and correction. In *Proc. of the 31st annual meeting of the Association for Computational Linguistics (ACL-93)*, 46–53.

Shriberg, E. E. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. Dissertation, University of California at Berkeley.

Traum, D. R. 1993. Mental state in the trains-92 dialogue manager. In *AAAI Spring Symposium on Reasoning about Mental States: Formal Theories and Applications*, 143–149.