Reference Hashed

Frank Schilder
Department for Informatics
University of Hamburg
Vogt-Kölln-Str. 30
22527 Hamburg, Germany
schilder@informatik.uni-hamburg.de

Abstract

This paper argues for a novel data structure for the representation of discourse referents. A so-called hashing list is employed to store discourse referents according to their grammatical features. The account proposed combines insights from several theories of discourse comprehension. Segmented Discourse Representation Theory (Asher, 1993) is enriched by the ranking system developed in centering theory (Grosz et al., 1995). In addition, a tree logic is used to represent underspecification within the discourse structure (Schilder, 1998).

1 Introduction

Discourse referents are represented quite differently by current discourse theories. Discourse Representation Theory (DRT), for example, employs a rather unstructured data structure for the domain of discourse referents: a *set* (Kamp and Reyle, 1993). A DRT-implementation by Asher and Wada (1988), however, employs a more complex data type: a *tree* representation. In further work by Asher (1993) referents are grouped together into segments depending on the discourse structure. His Segmented DRT (SDRT) uses a *tree*-like representation for the discourse structure.[1]

Centering Theory (CT) proposes a *list* structure for the entities one preferably refers to in subsequent sentences. In order to cover coreference over discourse segments the centering model was extended by a *stack* mechanism (Grosz and Sidner, 1986). Recently, these data structures have been criticized by Walker (1998), because they seem to be too restrictive. She proposes a *cache* storage for the referents in the focus of attention.

I propose instead a novel data structure for the representation of discourse referents. A *hashing list* is used to distinguish between the different types of referents according to their grammatical features wrt. number or gender. This list structure is furthermore combined with a hierarchical tree structure.

The remaining part of the paper is organised as follows. Section 2 introduces the main claims made by past theories. Focusing on SDRT and CT, I will highlight the (dis-) advantages of these two approaches. Section 3 provides the reader with an introduction to *hashing lists* and how they can be used for linguistic data. Section 4 discusses how the different advantages of former approaches can be combined. First, DRT will be amended by using a *hashing list* for the discourse referents instead of a simple *set*. Second, the centering model will be applied to the representation gained. Finally, the shortcomings of a flat representation are presented and the introduction of discourse segments is discussed. Subsequently, section 5 describes a detailed formalisation of one example sequences by the representation proposed, before section 6 concludes.

2 Background

It has been commonly agreed that a discourse is hierarchically organised. However, this is already the lowest common denominator among current approaches to discourse grammars and text comprehension. There is a wide range of views of what a formal representation of a discourse should look like. The following sections give a short introduction to two strands of research concerned with discourse processing. The first one, DRT and followers, is a linguistically-oriented approach that, generally speaking, captures the hierarchical structure of a discourse by a tree-like representation. The second strand, based on CT, is motivated by psychological experiments and models the structure of a discourse as a list representation of possible discourse referents. Further developments of CT have employed a stack structure or a cache storage.

[1]Similarly, Rhetorical Structure Theory (RST) makes use of a *tree* representation (Mann et al.,1988).

2.1 Hierarchical discourse grammars

SDRT (and RST) assume that so-called discourse (or rhetorical) relations are the links between discourse segments. A discourse relation has to be derived in order to achieve a coherent discourse. More importantly, the choice of this relation has a crucial influence on possible antecedents for anaphoric expressions.[2] Asher (1993) defines in SDRT the terms *subordination* and *openness* that specify where open attachment sites are in a discourse structure. A tree-like representation illustrates the hierarchical structure of the discourse. Basically, the nodes on the so-called 'right frontier' of the discourse structure are assumed to be available for further attachment (Webber, 1991).

Generally speaking, all nodes which dominate the current node of the newly processed sentence are open (i.e. *D-Subordination*). However, a restriction is introduced by the term *D-Freedom* which applies to all nodes that are directly dominated by a topic (i.e. $\alpha \Downarrow \beta$), unless it is the current node (see figure 1). An informal definition for possible attachment sites looks like the following:

1. The last clause represented as a Discourse Representation Structure (DRS) K.
2. Any DRSs that are embedded in K.
3. Any DRSs that dominate the DRSs in 1. and 2. through *Explanation*, *Elaboration* or $\Downarrow$

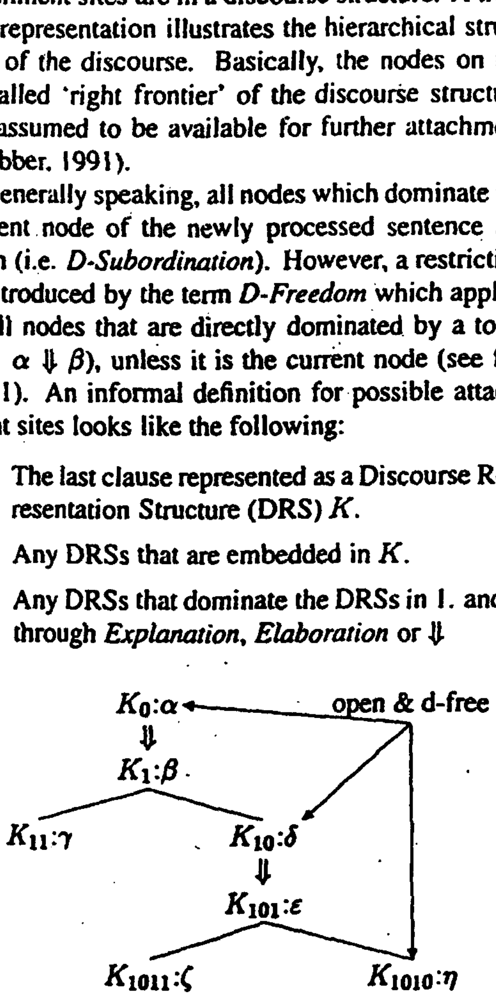

Figure 1: *Openness* and *D-Freedom*

SDRT exploits discourse relations to establish a hierachical ordering of discourse segments. A so-called constituent graph indicates the dependencies between segments, especially highlighting the open attachment points.

SDRT has been sucessful when phenomena are considered that are explainable because of the hierarchical structure of the discourse. This approach is too restrictive when an anaphoric reference is drawn over segment boundaries:

(1) (a) Mary once organised a party. (b) Tom bought the beer. (c) Peter was in charge of the food. (d) Years later Mary still complained that it was too spicy.

The sentence (1d) continues at the top level of the discourse, but the antecedent of *it* (i.e. *food*) is still avaliable even though it is deeply embedded in an *Elaboration* segment (i.e. (1a-c)).

Other shortcomings concern formal features. First, SDRT is not capable of expressing underspecification for ambiguous sequences. Second, the derivation of the discourse structure is not monotonic. Once derived, SDRSs are overwritten by an update.

2.2 Centering

CT proposed by Grosz et al. (1995) offers a text comprehension model that describes the relation between the focus of attention and the choices of referring expressions within a discourse segment. The main idea of this theory is that a sentence possesses a *center* and that normally one continues to write (or talk) about this center. Each utterance U_i gets a list of *forward-looking* centers $C_f(U_i)$ assigned to it. Basically, all the entities mentioned by the sentence are ranked according to their degree of being in the center of the utterance. Each sentence also has a unique *backward-looking* center $C_b(U_i)$. A main claim by the theory is that the most likely $C_b(U_{i+1})$ is the most highly ranked $C_f(U_i)$. Hence, the criteria for ranking the entities on the forward-looking center list are crucial for the predicative power of this theory. Normally, the grammatical relations *subject*, *object* etc. determine the preferred $C_p(U_i)$ (i.e. the first entity on the $C_f(U_i)$ list).

As mentioned earlier in this section, the initial account to centering is only concerned with the choice of referring expressions *within* a discourse segment. Since a more general theory to referring expressions is needed, an extension is presented by Grosz and Sidner (1986). They use a *stack* mechanism for representing the different discourse segments. If one segment is closed off, the information regarding the

[2] I will concentrate on how SDRT deals with this issue in the following. A study that shows how RST can be used to make predictions regarding anaphora resolution in a text can be found in Fox (1987).

forward- and backward looking centers is popped off the stack. The new top element of the stack contains the centering information from the old segment that the subsequent discourse continues with.

This simple stack mechanism has been criticised. In particular Walker (1998) points out that (i) a long intervening discourse segment can make it difficult to return back to earlier mentioned discourse referents and (ii) discourse referents introduced in a subordinated segment can easily be carried over to a higher segment (e.g. (1)). Note that a stack model would discard the information of a closed-off discourse segment. Walker proposes a *cache* storage that keeps often-used discourse referents within a storage. If reference is made to an antecedent mentioned earlier in the discourse the information is restored from long term memory.

Unfortunately, it is not quite clear how this retrieval operation can be formalised. In addition, it should be acknowledged that there are structured constraints of the discourse structure that do not allow the choosing of a recently mentioned referent. Data discussed within DRT, such as the sentence below, have been presented as evidence for a notion of (in-)accesibility.[3] Negation is a standard example that does not allow a reference to a discourse entity in the prevous sentence:

(2) No man walks in the park. #He whistles.

In the given example sequences the pronoun *he* cannot refer back to the discourse referents introduced in the previous sentence. Another example can be found in (3) that involves a conditional:

(3) If a farmer owns a donkey, he beats it. #He hates it.

Again, neither a pronominal reference by *he* nor by *it* is possible. It may be concluded from these data that a cache approach is not restrictive enough.

The discussion so far has shown that the data structures used for discourse processing are either too restrictive or not restrictive enough. The next section presents a novel way of representing discourse referents introduced by a text. The data structure presented is called a *hashing list* and allows for an efficient way to access stored information.

[3]Gordon et al. (1998), for example, blend DRT with CT.

3 Hashing lists

The following section describes how a hashing list works before the subsequent section shows how this data structure can be used for discourse processing.

3.1 Data hashed

One of the main problems for the design of computer systems is the question of how data is stored and efficiently accessed. Hashing lists are often used for this purpose since this data structure is specifically designed for easy retrieval of stored data.

I will first describe the data structure in more detail and then I will give an example of how data can be retrieved from a hashing list.

Hashing lists. The basic data structure for a hashing list is an array `A[min..max]` (i.e. an indexed list `A` that has a preset length of `n` elements). An array with the name `year` could be defined as follows:

```
TYPE hash = ARRAY[0..99] of
integer;
```

The *random access* structure of this data type allows the programmer to assign a single cell of the array directly (e.g. `hash[99] := 9;`). This is an advantage over other data structures such as trees.

Hashing functions. A function has to be designed that tells us how to store data on the hashing list. This function takes the item to be stored and gives back an appropriate key `k`. The item can now be stored at the right place on the list.

Suppose we want the program to store the integer `2000` on the hashing list `year` defined earlier. A hashing function `H(i)` has to be be chosen such that this function gives back an index `k`. With this information the assignment `hash[k]:=2000;` can take place. A hashing function for integers may be the Modulo function. For the given example the key `k` would be `20` (i.e. `2000 mod 99 = 20`).

The hashing function can also give back an index `k` for a new item that has already been taken by another item (e.g. `119` has the same key). For the case of a collision a special treatment is required. The most common one is the administration of an *overflow area*. The single places on the hashing list are lists that would handle colliding items. Figure 2 shows a part of the hash list `hash[19..21]` with two items `2000` and `119` inserted.

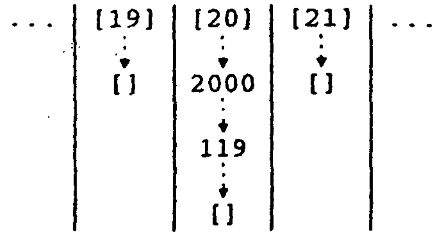

Figure 2: hash[19..21] with collision resolution for two items

3.2 Discourse hashed

I now show how a hashing list can be employed as a data structure for linguistic data. This may not be obvious after using only integers for storing on a hashing list.

Domains of referents. Natural language processing requires a richer data structure than storing integers. However, in the end a hashing function for linguistic data will also consist of an array.

Considering the different types of discourse referents, we can assume at least the following list of referents to be relevant: singular male, singular female, singular neuter, plural and event referents.[4] We now take these conceivable referents and reserve each of them a slot in the domain array:

domain[sgM, sgF, sgN, pl, ev]

Note that this way of writing the hashing list is actually only syntactic sugar for a normal definition such as domain[1..5].

Referent function. A function is needed that can assign a cell on the array domain to a newly introduced discourse referent. The semantic and syntactic information that comes with the a discourse referent gives us the key for this. Take for example a proper name such as *Peter*. The information that comes with it could be encoded as a feature value matrix such as proposed by Dale (1992) (see figure 3). The hashing function takes the information under the agreement feature AGR and checks for the values regarding number and gender.

$$\begin{bmatrix} \text{NUM:} & \textit{sing} \\ \text{GEND:} & \textit{male} \end{bmatrix}$$

$$\begin{bmatrix} \text{INDEX:} & x_1 \\ \text{SYN:} & \begin{bmatrix} \text{CAT} & \textit{pn} \\ \text{AGR:} & \begin{bmatrix} \text{NUM:} & \textit{sing} \\ \text{GEND:} & \textit{male} \end{bmatrix} \end{bmatrix} \\ \text{SEM:} & \begin{bmatrix} \text{NAMED:} & \textit{peter} \end{bmatrix} \end{bmatrix}$$

Figure 3: The representation for *Peter*

The function returns sgM (or 1) as key for the array domain in the example given.

Summing up, a hashing list was proposed to store discourse referents while processing natural language discourse. This kind of list contains several "slots" that await discourse referents described by a discourse. The grammatical features of gender and number distinguish the different referents.

The following section discusses how this data structure is embedded into a discourse grammar.

4 Referents in discourse

The linguistic data presented earlier demonstrates the need for hierarchical constraints on anaphora resolution. But the data also show that previous approaches such as SDRT overemphasise this restriction. A refusal of any discourse structure constraints, on the other hand, also does not seem to be appropriate. A cache storage that stores the frequently used discourse referents does not account for the data that were explainable by (S)DRT.

This section describes how a hashing list can be used for the storage of discourse referents. The list is integrated into an SDRT framework. The information about the discourse segments is kept in order to cover data that is explainable by the hierarchical discourse structure. In addition, the insight that a sentence has a center as proposed by CT is also reflected by the theory proposed. The discourse referents are ordered according to centering preference.

The following sections describe in more detail how the different concepts are integrated in the system proposed by this paper. First, the way discourse referents are stored via a hashing list is explained. Second, the ordering regarding the centering preference is imposed on the slots of the hashing list. And finally, a tree structure is presented that binds all the components together.

[4]This is only a first list of very fundamental referents. But the list can easily be extended by more differentiated plural types, speech acts or types of referents.

4.1 Referents hashed

In the system proposed, a hashing list stores the referents introduced by the discourse. The hashing list contains at least the following slots: `sgM`, `sgF`, `sgN`, `pl`, `ev`. Since the basic formalism is DRT, we need to incorporate the hashing list into the formalism. In DRT, a DRS consists of the domain of discourse referents and the set of conditions imposed on the referents. A sentence such as *Peter sighs* is represented by the box notation as follows:

x_1 e_1
peter(x_1) e_1 : sigh(x_1)

A hashing list substitutes for the set of discourse referents offering different slots for the discourse referents to be stored in:

x_1				e_1
peter(x_1) e_1 : sigh(x_1)				

The representation of a more complex sentence such as *Peter gave John a book* containing several discourse referents is in the following DRS:

x_1 x_2		x_3		e_1
peter(x_1) john(x_2) book(x_3) e_1 : give(x_1, x_2, x_3)				

This Hashed DRS (HDRS) contains a complex domain sub-box. The slot for male and singular discourse referents is filled with the two items x_1 and x_2. The two referents are on a collision list as described earlier. Additionally, the list reflects the ordering for the centering list. The subject NP *Peter* was processed before the object NP *John* and is therefore the first entry on the preferred centering list. Note that only referents that share the same grammatical features are listed in the same slot.

4.2 Referents re-centered

After blending a DRT representation with a hashing list for a structured representation of discourse referents, I will introduce the centering feature into the formalism. The different slots already contain the ordering of the referents regarding the centering preference. An apparent advantage over the centering approach should become clear: the referents are already separated from each other.

A discourse such as (4) without any competing antecedents for the pronoun *she* is formalised by a HDRS as follows:

(4) (a) Peter gave Mary a book. (b) It was about sailboats. (c) She was thrilled.

x_1	x_2	y_1		e_1 s_2 s_3
peter(x_1) mary(x_2) book(y_1) e_1:give(x_1,x_2,y_1) s_2:about(y_1, s_boats) s_3:thrilled(x_2)				

CT would predict for (4b) that *the book* is the preferred forward looking center C_p. The backward looking center of (4c) is *Mary*. This is called a *rough shift* in CT. A continuation of the center ($C_p(U_i) = C_b(U_{i+1})$) is the preferred and most coherent constellation according to this theory. However, contrary to what CT would predict, it is no problem to read (4).

The HDRS format seems to work fine with pronominal references to persons or objects. But we run into problems when the slot regarding the descibed events and states is considered. The following example (5) illustrates that a simple flat list representation as indicated above by e_1, s_2, s_3 is not sufficient for more complex anaphoric expressions such as event anaphora (Allen 1995):

(5) (a) When Jack entered the room, everyone threw balloons at him. (b) In retaliation, he picked up the ladle and started throwing punch at everyone. (c) Just then, the chairman walked into the room. (d) Jack hit him with a ladleful, right in the face. (e) Everyone talked about it for years afterwards.

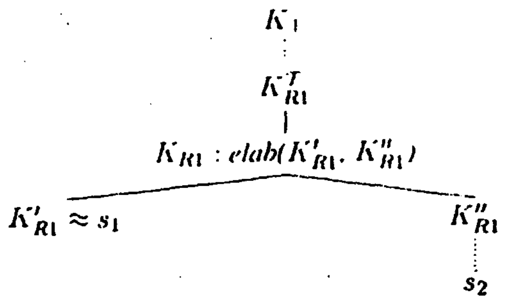

Figure 4: The discourse structure for *Elaboration*

The pronoun *it* in (5e) may refer to the entire situation described by (5a) through (5d). But this is not the only conceivable antecedent for *it*. The situation described by (5d) may be referred to by *it* as well, if we consider an alternation of (5e) as in the following:

(5e') It was a foolish thing to do.

Note that the situation in (5d) is the only situation available from the sequence (5a-d). The list structure for the `event` slot does not reflect the structure of the discourse. A segmented discourse structure is needed here.

4.3 Discourse segments

The derivation of discourse structure used in this account is that proposed by SDRT. This discourse grammar, as well as others, claims that discourse segments originate from the derivation of so-called discourse relations (e.g. *Narration*, *Elaboration* etc.) due to our background or world knowledge.

The account proposed by this paper assumes that HDRSs are grouped together wrt. their discourse segment. Consider now the following sequence (6) with the possible continuation (6e) with a male and female pronoun (depending on whether a male or female protagonist was introduced by the first sentence).

(6) (a) Mary/Mark once organised a party. (b) Tom wrote the invitation cards. (c) Peter bought the booze.

(6e) She/He was glad that everything worked out so nicely.

The first continuation does not cause any problems, although the antecedent for *she* was introduced by the first sentence of the sequence. Since no other competing discourse referents have been mentioned, the resolution process works without problem. However, substituting a male protagonist called Mark for the female protagonist in the first sentence does cause problems for the understanding of (6). In this case, it is unclear who was meant by *he*. Note furthermore that only two antecedents are available, even though three male antecedents have been introduced. Only the one in the last sentence (i.e. *Peter*), or the one introduced by the first sentence (i.e. *Mark*) are conceivable antecedents. A different continuation does not show this ambiguity:

(6e') He decided just to buy beer.

The continuation in (6e') is an elaboration of the last sentence. Hence *Peter*, who was responsible for the booze, is the only possible antecedent.

The following sentence is the last piece of evidence that the discourse segment allows only antecedents that are available on the so-called right frontier. The following sentence shows that it is not possible to refer to *Tom*, who wrote the cards, with the last sentence:

(6e") #He decided to use thick blue paper.

5 Formalisation

This section is an introduction to the formalism used. The formalism consists of the following parts:

DRT The standard DRT theory is used to obtain a semantic representation for the meaning of a clause (Kamp and Reyle, 1993). However, the set of discourse referents is more structured than in the standard approach. It also goes beyond the approach by Asher and Wada (1988) (see below for further details).

Hashing lists The data structure of hashing lists is used to divide the set of discourse referents up into different slots. Each slot contains only referents of the same type, as there are singular male, female, or neuter referents, plural entities and events.

SDRT A hierarchical discourse structure is needed to explain anaphoric expressions that refer back over segments boundaries. In addition, a theory is needed that takes into account world knowledge for the derivation of discourse relations (Asher, 1993).

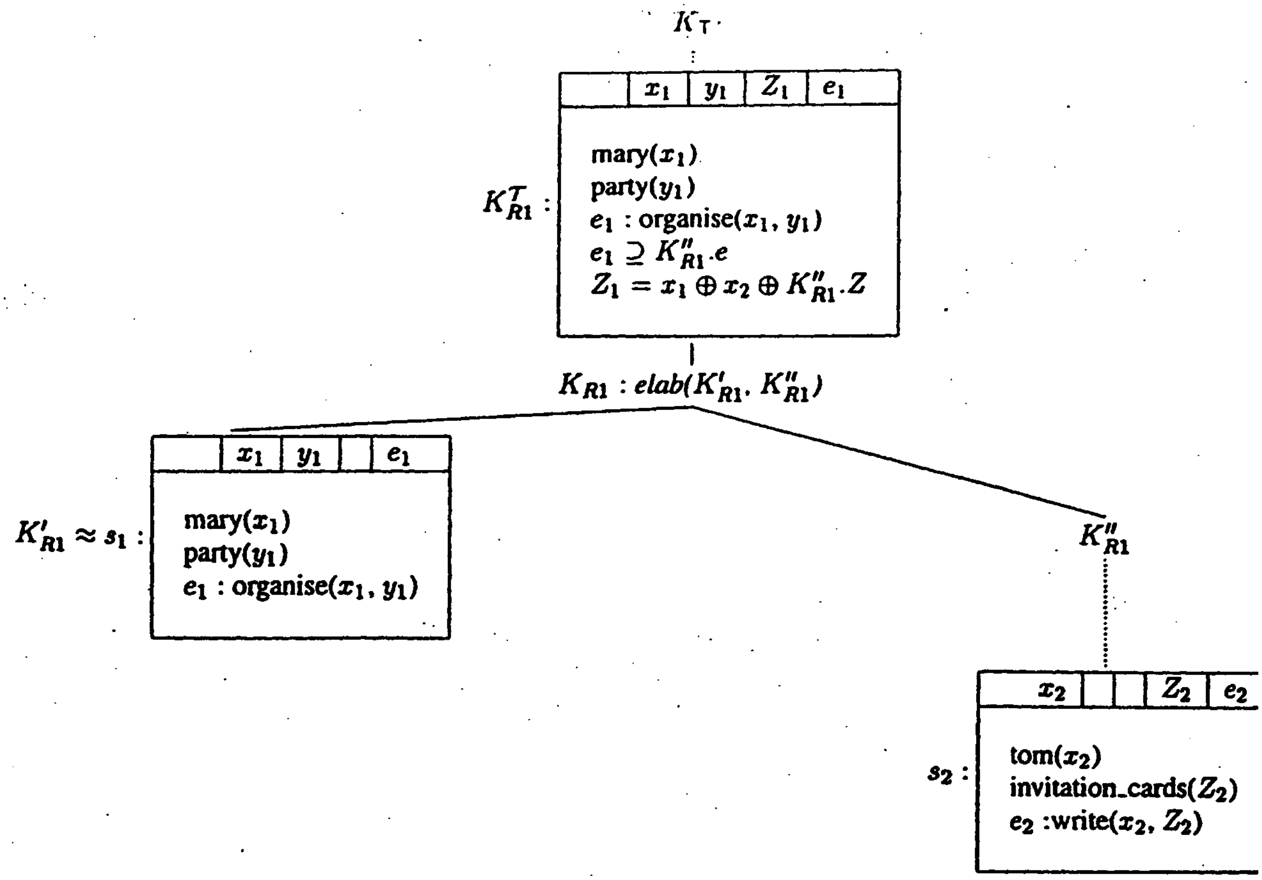

Figure 5: The Segmented HDRS for (6a-b)

Underspecified Tree Structure Data have been discussed that show an ambiguity regarding the discourse structure. In order to express the ambiguity formally an underspecification mechanism is employed (Schilder, 1998).

I will now present the derivation of the sequence in (6).

5.1 Elaboration

First, a HDRS representation is to be derived for the first sentence. The HDRS for (6a) looks like a normal DRS, the only difference is the hashing list that contains the discourse referents in different slots. Second, a HDRS for the second sentence is derived and, in addition, a discourse relation is inferred from our world knowledge. An *elaboration* relation links the two HDRSs in the given case. Within an underspecified version of SDRT this discourse structure is represented as shown in figure 4.

The nodes in the tree are labels for (Segmented) HDRSs. The two labels s_1 and s_2 denote the semantic content of the two first sentences, respectively. The label K_{R1} refers to the derived relation *elaboration* that holds between the two segments K'_{R1} and K''_{R1}. Note that the left daughter node of the K_{R1} is already determined by setting K'_{R1} equal to s_1. The right daughter node, however, is left open. This is indicated by the dotted line between K''_{R1} and s_2. This line expresses graphically the dominance relation between tree nodes ($\lhd^*$) in contrast to the straight line that indicates an *immediate* dominance relation ($\lhd$).[5]

The underspecification of the tree structure allows us to define where possible attachment points are on the right frontier of the discourse structure. The tree structure in figure 4 possesses two attach-

[5] I follow here the description of a tree logic such as that used by Kallmeyer (1996) or Muskens (1995).

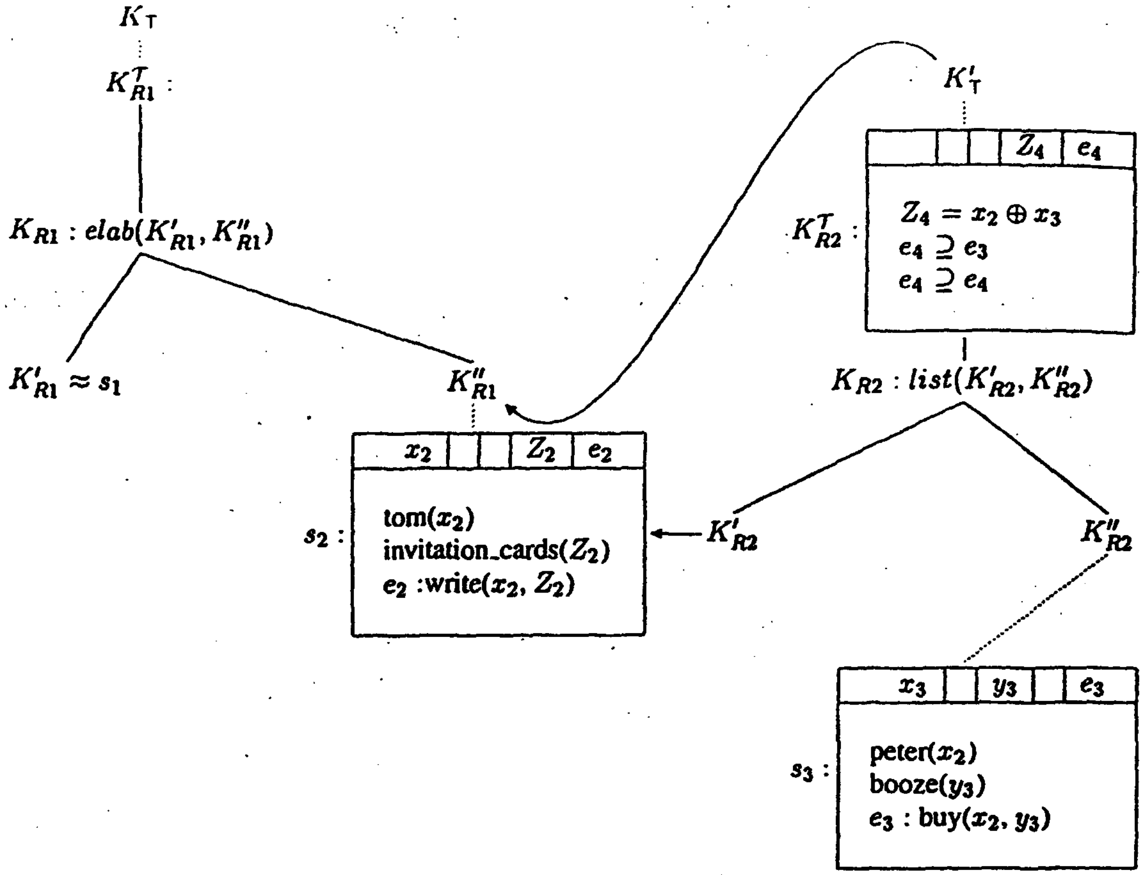

Figure 6: The third sentence (6c) added

ment points: one is between K''_1 and s_2 and the other one is between $K_\top$ and $K^{\top}_{R1}$. This latter node denotes the *topic* of the current discourse segment which is the situation described by s_1 for (6a-b).

Two further remarks are to be made regarding the representation in figure 5 before continuing with the sequence. First, an additional condition is added to the *topic* node. The information about the temporal relation between the situation e_1 and the subordinated situation was added on this level of the discourse tree. Note that it is still open which event e will finally show up in the node referred to by K''_{R1}.[6] Only after closing off this discourse segment will it be clear which event(s) elaborated the situation e_1.

Second, a plural entity Z_1 is stored in $K^{\top}_{R1}$. This entity combines the singular entities into a plural one. A more elaborate mechanism is needed here in order to combine only entities of the same type (e.g. persons). For the time being, all plural entities are stored in this one slot.

[6] $K''_{R1}.Z$ can be described as a pointer to the plural slot of K''_{R1}.

5.2 Continuing the thread

A *list* relation can be derived for the sentences (6b) and (6c) (see figure 6). The semantic content s_3 is added to the discourse tree linked by the discourse relation and furthermore a common topic is added at $K^{\top}_{R2}$. The *topic* information has to be an abstract representation of the two HDRSs s_2 and s_3. In order to achieve that, two *new* discourse referents are introduced: a plural entity Z_4 comprising x_2 and x_3 (i.e. *Tom* and *Peter*) and a complex situation e_4 temporally covering e_3 and e_4.

5.3 Looking back

After the third sentence has been processed, the next sentence contains a pronoun. In case of *she* the pronoun looks for a female singular antecedent. The appropriate discourse referent is found on the right frontier in the appropriate slot. Alternatively, if sequence (6) contains the male protagonist named *Mark* in the first sentence instead of *Mary*, the personal pronoun *he* could have two possible antecedents: *Peter* or *Mark*. How can that be explained by the formalism?

Figure 7 depicts the formalisation of the discourse (6a-e) only showing the hashing lists on the right frontier. The dotted arrows indicate the hashing list as it is distributed over the right frontier of the discourse structure. There is only one entry for a female singular antecedent over the levels of nodes on the right frontier. However, if there were a male protagonist, the hashing list for the referents in node K^{T}_{Ra} would contain a discourse referent in the first slot. The list of possible antecedents for *he* would be x_3 and x_1.

The separation of different reference types also allows us to explain sequences such as (1). The discourse continues on the highest level, but it is possible to refer to discourse referents that got introduced on a lower level of the discourse structure. The link between two situations can be made via a rhetorical relation, and at the same time the slots for the other referents at the right frontier are still accessible.

The hashing list also models a hashed right frontier. Past approaches always collapsed discourse attachment with the restriction regarding possible antecedents for anaphora (cf. the stack mechanism in CT or the tree representation for (S)DRT).

The formalisation can also provide an explanation of why competing antecedents can cause an ambiguity for the pronoun resolution. The accessibility of hashing lists on different levels of the discourse structure explains why, in this example, a female antecedent can be used as an antecedent even over several intervening sentences. It is important to highlight the difference of the account presented here to past approaches: The discourse referents are grouped together according to their agreement features. The DRT account by Asher and Wada, for instance, stores the discourse referents in a tree according to the accessibility conditions imposed by DRT and singles out the appropriate antecedents according to number and gender information as well as other criteria. There the agreement information is used to "weed out" possible antecedents, whereas within a HDRS the discourse referents are already accordingly stored.

It should also be clear that an embedded discourse cannot be extended infinitively, as shown by the cache approach. A restriction has to be imposed on the number of levels where an antecedent can be looked for. Future research, however, has to clarify this issue further.

Note that this formalisation capitalizes on the insight gained from the cache approach. An elaboration cannot be continued for too long, since the working memory of the reader might lose track of the protagonist(s) introduced on the highest level. On the other hand, this formalisation also covers more data than the cache approach. A text comprehension theory that employs a cache storage cannot account for a discourse such as (6) with a non-competing female protagonist. The discourse referent for *Mary* would have been stored in long term memory, because no differentiation is made between the grammatical types of possible antecedents according to the cache approach.

6 Conclusion

I have proposed a new data structure for the processing of coreference in a discourse. A *hashing list* was employed to store referents according to their grammatical features such as number or gender. Because of this, better accessibility to non-competing antecedents can be modelled by the approach presented.

The discourse grammar used combined insights from different approaches to discourse processing: (S)DRT and CT. In addition, a tree logic was used to allow underspecification in the representation of ambiguous sequences.

Future research has to focus on the evaluation of the proposed theory to anaphora resolution. Copora investigation and psychological experiments will provide more evidence. In addition, an implementation of hashed DRT is being programmed.

Acknowledgments

I would like to thank the two annomynous reviewers to their comments and feedback. Special thanks to Christie Manning for providing me with all her help.

References

Allen, James. 1995. *Natural Language Understanding*. The Benjamin/Cummings Publishing Company, Redwood City, California.

Asher, Nicholas. 1993. *Reference to abstract Objects in Discourse*, volume 50 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht.

Asher, Nicholas and Hajime Wada. 1988. A computational account of syntactic, semantic and dis-

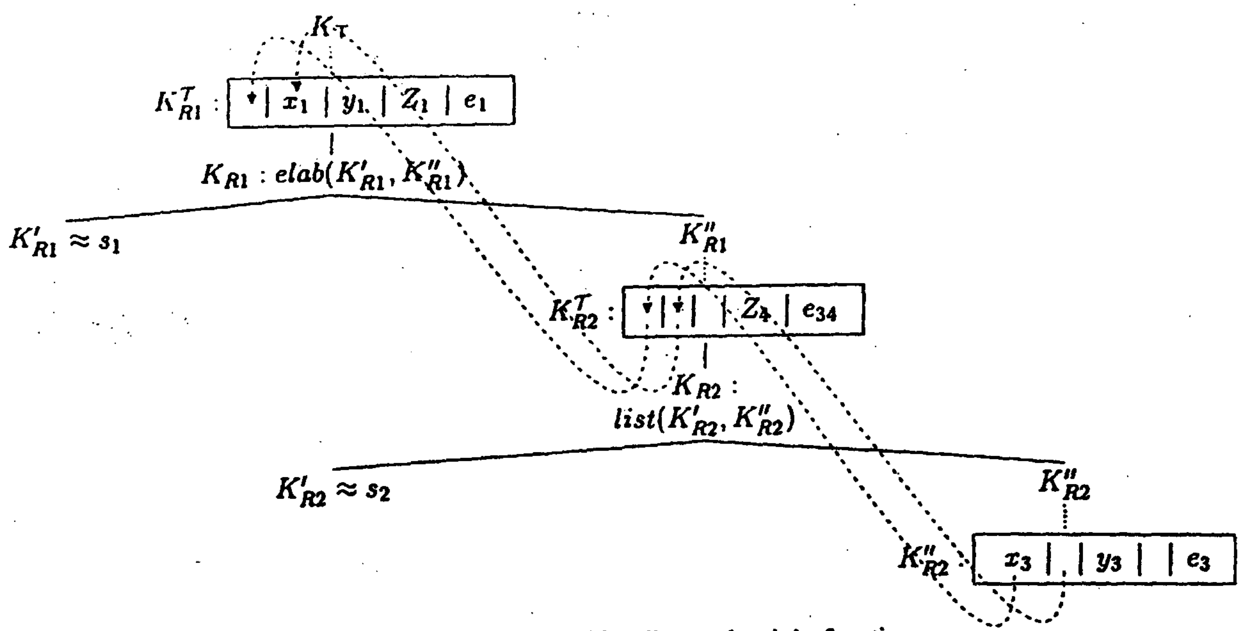

Figure 7: The entries of the hashing list on the right frontier

course principles for anaphora resolution. *Journal of Semantics*, 6:309–344.

Dale, Robert. 1992. *Generating Referring Expressions*. MIT Press, Cambridge, Massachusetts.

Fox, Barbara. 1987. *Discourse structure and anaphora*. Cambridge University Press.

Gordon, Peter and Randall Hendrick. 1998. The representation and processing of coreference in discourse. *Cognitive Science*, 22(4):389–424.

Grosz, Barbara J., Aravind Joshi, and Scott Weinstein. 1995. Centering: A framework for modelling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Grosz, Barbara J. and Candace L. Sidner. 1986. Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Kallmeyer, Laura. 1996. Underspecification in Tree Description Grammars. Arbeitspapiere des Sonderforschungsbereichs 340 81, University of Tübingen, Tübingen, December.

Kamp, Hans and Uwe Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language*, volume 42 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht.

Mann, William and Sandra Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organisationn. *Text*, 8(3):243–281.

Muskens, R. 1995. Order-independence and underspecification. In J. Groenendijk, editor, *Ellipsis, Underspecification, Events and more in Dynamic Semantics*, number R.2.2.C in DYANA Deliverable. ILLC/Department of Philosophy, University of Amsterdam, Amsterdam, The Netherlands, pages 15–34.

Schilder, Frank. 1998. An underspecified segmented discourse representation theory (USDRT). In *Proceedings of the 17 th International Conference on Computational Linguistics (COLING '98) and of the 36 th Annual Meeting of the Association for Computational Linguistics (ACL '98)*, pages 1188–1192, Université de Montréal, Montréal, Québec, Canada.

Walker, Marilyn. 1998. Centering, anaphora resolution, and discourse structure. In Marilyn Walker, Aravind Joshi, and Ellen Prince, editors, *Centering Theory in Discourse*. Clarendon Press, Oxford, pages 401–435.

Webber, B. L. 1991. Structure and ostension in the interpretation of discourse deixis. *Language and Cognitive Processes*, 6(2):107–135.