Information Sharing

Kees van Deemter and Rodger Kibble (eds.)

November 29, 2004

**CENTER FOR THE STUDY
OF LANGUAGE
AND INFORMATION**

Contents

1

Efficient Context-Sensitive Generation of Referring Expressions

Emiel Krahmer and Mariët Theune

1.1 Introduction

In their interesting 1995 paper, Dale & Reiter present various algorithms they developed alone or in tandem to determine the content of a *distinguishing description*. That is: a definite description which is an accurate characterization of the object being referred to, but not of any other object in the current 'context set'. They argue that their *Incremental Algorithm*, discussed in more detail below, is the best one from a computational point of view (it has a low complexity and is fast) as well as from a psychological point of view (humans appear to do it in a similar way).

Even though Dale & Reiter (1995) primarily aimed at investigating the computational implications of obeying the Gricean maxims,[1] the Incremental Algorithm has become more or less accepted as the state of the art for generating descriptions. However, due to their original motivation, various other aspects of the generation of definites remained somewhat underdeveloped. In this chapter, we flesh out a number of these aspects, *without losing sight of the attractive properties of the original algorithm* (speed, complexity, psychological plausibility). In particular, we focus on the role of context-sensitivity for referring expression generation.

The basic idea we want to pursue is that a definite description refers

[1] For a discussion of this aspect of their work, see Oberlander 1998.

Information Sharing.
Kees van Deemter and Rodger Kibble (eds.).

to the *most salient* element satisfying the descriptive content (Lewis 1979: 348-350, see Krahmer 1998 for a formalization in dynamic semantics). Lewis mentions the following example (due to McCawley):

(1) The dog got in a fight with another dog.

Lewis notes that this statement can only be true in a domain which contains at least two dogs, which entails that *the dog* cannot be a distinguishing description. According to Lewis, (1) means that the most salient dog got in a fight with some less salient dog. Lewis does not mention descriptions which refer to 'unique' objects, but it is readily seen that they can also be understood in terms of salience: if there is only one object with the relevant properties, it *has* to be the most salient one. Arguably, a notion of salience is implicit in Dale & Reiter's usage of context sets, and also extensions such as Horacek (1997:210) and Stone & Webber (1998:183) explicitly remark that some form of salience is important. None, however, specify how salience should be determined nor which repercussions the inclusion of a notion of salience has for the generation algorithm.

In this chapter we show that it is possible to integrate an explicit notion of salience into Dale & Reiter's Incremental Algorithm and that this paves the way for efficient context-sensitive generation of referring expressions.[2] This is done in a number of stepwise refinements of Dale & Reiter's original Incremental Algorithm. In the first step we describe a generalized version of the Incremental Algorithm which incorporates a notion of salience. This provides us with a way of modelling salience differences between objects: some objects can be more prominent in the linguistic context than others. We argue that the modified algorithm only needs to distinguish the intended referent from those objects which are equally or more salient. As we shall see, this means that in those cases the modified algorithm typically outputs shorter descriptions, which are not 'distinguishing' in the sense of Dale and Reiter, but which are anaphoric

[2]Horacek (1997) and Stone & Webber (1998) are closely related in spirit to the current work. Horacek (1997) makes various interesting observations about the limitations of the Incremental Algorithm and its ilk and proposes a new algorithm with 'flexible interfaces' to other modules. Many of the issues discussed by Horacek are also addressed in this chapter, and some of his suggestions are taken over, in particular the integration of linguistic constraints during generation. This integration is also a central ingredient of Stone & Webber (1998), who go one step further and argue for the simultaneous inclusion of semantic and pragmatic information as well. While both approaches look promising, it is difficult to make a precise judgement of their respective performance and predictions since central components (e.g., selection of properties, salience determination) are left unspecified. As a consequence, the computational properties of their algorithms are not clear (see Horacek 1997:212 and Stone & Webber 1998:186).

to previous references to the intended referent and thus are only distinguishing in context. If we make the (unrealistic) assumption that all objects in the domain are equally salient, then the generalized algorithm produces the same output as the Incremental Algorithm, albeit in a slightly different way. We also propose a number of extensions required for embedding the algorithm in a general data-to-speech system (i.e., a system which converts structured, non-linguistic data into spoken language). For this purpose, we follow Horacek (1997) and extend the algorithm with a check on the expressibility of properties and a procedure to build a natural language expression within the algorithm. In addition, it is directly checked whether properties are contrastive or not, which is relevant for prosody computation in speech generation.

Of course, it only makes sense to integrate a notion of salience in the Incremental Algorithm if there is a method to assign salience weights in a principled way. Fortunately, a number of such methods exist. We discuss and compare two of those, selected more or less at random; one is based on the hierarchical focusing constraints of Hajičová (1993), the other on the constraints of Centering Theory (Grosz *et al.* 1995). The advantages and disadvantages of both methods are discussed and a synthesis of the two is proposed. While this synthesis is certainly open for further refinements, it provides us with a workable salience function which can be used in tandem with the modified Incremental Algorithm.

The proposed modified algorithm and method for salience weight assignment have been implemented and integrated in a data-to-speech system called D2S. In addition, we carried out an experiment verifying some of the basic assumptions underlying the modified algorithm. In this experiment it was investigated whether subjects prefer the reduced and generalized descriptions that the modified algorithm generates over the full distinguishing descriptions that Dale & Reiter's Incremental Algorithm gives rise to. This was indeed the case, although there was, as we shall see, one somewhat surprising finding: subjects only prefer more general descriptions if these have the same head noun as their antecedent. Thus, once an object has been introduced as a mastiff, people prefer subsequent references to have the same head noun and not a more general one, such as 'dog'. The modified algorithm will be extended to take this finding into account.

A number of further extensions of the modified Incremental Algorithm can be made. First, a conservative pronominalization decision can be taken within the algorithm; a pronoun can be used to refer to the single most salient object in the domain. Second, and more substantially, it is shown how a slight modification of the modified algorithm, enables the generation of relational descriptions, that is, descriptions which describe an object in terms of its relation to another object (called the relatum). This has various useful consequences. In particular, the method of determining the best values for certain attributes used in the Incremental Algorithm carries over in an interesting way to the relational algorithm in that now also the best values of *relations* between objects can be determined.[3] Finally, it is argued that an algorithm which can generate pronouns and relational descriptions, can in principle also generate bridging descriptions. From the current perspective, a bridging description is treated as a relational description with a highly salient relatum.

This chapter is structured as follows. In section 1.2 we briefly describe Dale & Reiter's (1995) original Incremental Algorithm. Then, in section 1.3 we describe the modified version of the Incremental Algorithm using salience weights. In section 1.4, the two aforementioned ways of assigning salience weights to objects are discussed and compared, and a synthesis is offered. Section 1.5 describes the experiment that was carried out to verify some of the basic assumptions underlying the modified Incremental Algorithm. In section 1.6 a number of extensions of the modified Incremental Algorithm are sketched, which can generate pronouns (section 1.6.1), relational descriptions (section 1.6.2) and, finally, bridging descriptions (1.6.3). In the concluding section we briefly discuss the implementation of the modified algorithm and its integration in a data-to-speech system, and mention some lines for future research.

1.2 The Incremental Algorithm (Dale & Reiter 1995)

The Incremental Algorithm (and the extensions that are discussed in the rest of this chapter) can be used in any domain which meets the following criteria (Dale & Reiter 1995:254): (*i*) each object in the domain is characterized by a list of attribute value pairs, or *properties*, (*ii*) each object has at least an attribute *type* and (*iii*) there may be a subsumption hierarchy on the values of certain attributes. Some of the values in such a hierarchy are the *basic level values* (e.g., Rosch 1978) for a certain attribute. Basic levels are the first levels which children learn to understand and use. Additionally, and most relevant for current purposes, basic levels are normally used in neutral contexts. Lakoff (1987:46): "*For example*, There's a dog on the porch *can be used in a neutral context, whereas special contexts are needed for* There's a mammal on the porch *or* There's a wire-haired terrier on the porch. *(See Cruse 1977)*." From the perspective of generation, the basic level values might not only be viewed as context-dependent (cf. Rosch 1978:42) but also as

[3] A version of the relational extension of the modified algorithm has recently been implemented, but is not integrated yet in D2S.

user-dependent (Dale, p.c.). For instance, it seems likely that for a professional dog-breeder the 'basic level' is below 'dog' in the subsumption hierarchy. Here, the basic levels are simply treated as given.

Dale & Reiter (1995) use a dogs and cats domain to illustrate the workings of the Incremental Algorithm. Although this domain may not be an obvious choice for language generation applications, it is suitable for illustrative purposes, as it fits the above criteria very well: cats and dogs are familiar physical entities with easily perceivable properties, and at least one of their attributes ('type') can be organised in a subsumption hierarchy. We therefore follow Dale & Reiter and stick to the cats and dogs domain throughout this chapter.

Suppose, for the sake of illustration, that we have a domain D_1 consisting of the following four objects $d_1 - d_4$.

d_1	⟨ type, chihuahua ⟩, ⟨ size, small ⟩, ⟨ colour, black ⟩
d_2	⟨ type, chihuahua ⟩, ⟨ size, large ⟩, ⟨ colour, white ⟩
d_3	⟨ type, siamese cat ⟩, ⟨ size, small ⟩, ⟨ colour, black ⟩
d_4	⟨ type, poodle ⟩, ⟨ size, small⟩, ⟨ colour, white ⟩

Assume that there is a subsumption hierarchy for the attribute 'type': 'dog' subsumes 'chihuahua' and 'poodle', 'cat' subsumes 'siamese cat', and 'animal' in its turn subsumes 'dog' and 'cat'. Following Dale & Reiter, the basic level values for 'type' are taken to be 'dog' and 'cat' respectively.

The input for the Incremental Algorithm is an object r, a *context set* C consisting of alternative objects from which r has to be distinguished (the *distractors*), and, crucially, a list of preferred attributes. This list contains, in order of preference, the attributes that human speakers and hearers prefer for a particular domain. For instance, it seems likely that a human speaker would first try to describe an animal by its 'type' (is it a dog? is it a cat?), and if that does not help attributes like 'colour' and 'size' may be used. It is reasonable to assume that speakers have a general preference for *absolute* properties such as 'colour', which are easily observed without taking the other objects into account, over *relative* properties such as 'size', which are less easily observed and always require inspection of the distractors.[4] Thus let us assume that the list of preferred attributes for the example domain is ⟨ type, colour, size ⟩. Essentially, the Incremental Algorithm goes through the list of preferred attributes, and for each attribute it encounters it looks for the *best value* of this property. The best value of a property is the value closest to

[4] The literature on perception contains a wealth of empirical evidence for such general orderings, see, for instance, Pechmann (1989), Levelt (1989) and, more recently, Beun & Cremers (1998).

the basic level value such that there is no *more specific* value that rules out *more* distractors. If adding this best value to the already selected properties has the effect of ruling out any of the remaining distractors, it is included in the list of properties to be used in the generation of the distinguishing description. The algorithm stops when the end of the list of preferred attributes is reached (failure), or when the set of distractors is empty (success). In the latter case, it is checked whether the 'type' property was included, and if not, its basic value is added to the selected list of properties.

Dale & Reiter (1995:247) argue that this algorithm has a polynomial complexity and that the theoretical run time can be characterized as $n_d n_l$: the run time depends solely on the number of distractors n_d and the number of iterations (i.e., selected properties) n_l. This means that the Incremental Algorithm is the fastest algorithm discussed in Dale & Reiter (1995). One of the central features of the Incremental Algorithm is that there is no backtracking (hence the term 'incremental'): once a property p has been selected, it will be realised in the final description, even if a property which is added later would render the inclusion of p redundant with hindsight. This aspect is partly responsible for the efficiency of the algorithm, but Dale & Reiter additionally claim that this property is 'psychologically realistic' since human speakers also often include redundant modifiers in their referring expressions (see e.g., Pechmann 1989).[5]

1.3 A Modification of the Algorithm Based on Salience

1.3.1 Motivation: Determining the Context Set

The contents of the descriptions generated by the Incremental Algorithm are to a large extent determined by the context set. Nevertheless, Dale

[5] Dale & Reiter (1995:248): "*For example, in a typical experiment a participant is shown a picture containing a white bird, a black cup, and a white cup and is asked to identify the white bird; in such cases, participants generally produce the referring expression* the white bird, *even though the simpler form* the bird *would have been sufficient.*" Yet, it seems to us that the Incremental Algorithm would produce the description *the bird* in this situation: if we make the natural assumption that 'type' is the most preferred attribute, the property 'bird' will be the first one selected and immediately rules out the black and the white cup. In general, it should be noted that Pechmann's notion of incrementality refers to *speech production*; the fact that speakers, when describing an object, start uttering properties of that object without making sure whether these are actually distinctive or not. Dale & Reiter's incrementality refers to the lack of backtracking in property selection, but the order in which properties are selected is not related to the order in which properties are realized in speech. It is worth noting that full incrementality in the latter sense (each property is uttered as soon as it is selected), cannot be obtained without taking a certain amount of lookahead into account (Levelt 1989).

and Reiter do not address the question how such context sets are constructed nor how their contents can be updated during the generation process. They only write:

> We define the context set to be the set of entities that the hearer is currently assumed to be attending to; this is similar to the set of entities in the focus spaces of the discourse focus stack in Grosz and Sidner's (1986) theory of discourse structure. (Dale & Reiter 1995:236)

Dale (1992:192-193) is somewhat more explicit and briefly describes a full and a partial order on focus spaces. However, he concludes that:

> [i]n the present domain [recipes, K&T], since the number of entities we are dealing with is relatively small, it is adequate to take the global working set to be the context.

(The 'working set' refers to the set of "identifiable distinct entities in the domain at any point in time," Dale 1992:56.) Our aim is to be explicit about the continuously changing contents of context sets and the repercussions this has for the Incremental Algorithm. It is instructive at this point to look at some of the options we have at our disposal. We assume that there always has to be a domain of discourse D: the total set of objects which can be referred to (in a data-to-speech system this set is given as part of the data). Some people have suggested that a context set may be a proper subset of D, containing those objects of D which have been referred to before (or which are prominent for some other reason). This should make it possible to generate reduced anaphoric descriptions, because the intended referent only has to be distinguished from the members of the context set, and not from all domain objects. However, there appear to be some problems with this approach. We illustrate this using the following example.

Suppose the domain of discourse consists of all dogs that are present at a dog show, and thus contains hundreds of dogs of all kinds, sizes, and colours. We assume that initially, the context set equals the domain of discourse, since at the beginning of the discourse all objects in the domain are equally prominent (or, in terms of Grosz & Sidner 1986, the focus space is still empty).[6] If, in this situation, we wish to generate a distinguishing description for one of these dogs, say d_{45}, this description will necessarily include many of its properties, e.g., *the large black long-haired sausage dog.* After d_{45} has been referred to, it seems an obvious

[6] This is a simplification, which has no influence on the arguments presented here. In reality, it will often be the case that some objects in the domain are initially more salient than others, for instance because they are (physically) closer to the conversational situation.

move to restrict the context set so that it only includes d_{45}, as the hearer's attention has now been directed to this object (in other words, d_{45} is now added to the focus space). If we now wish to generate a second reference to d_{45}, the reduced context set allows us to generate the anaphoric description *the dog*. After all, in this situation d_{45} has no distractors. If we would not restrict the context set, but would re-refer to d_{45} with respect to the entire domain, this would cause a repetition of the initial description *the large black long-haired sausage dog*, which is clearly undesirable. On the other hand, if instead of generating a second reference to d_{45} we wish to refer to another object from the domain, say d_{53} (which happens to be a small grey pygmy poodle with a perm wave), it becomes clear that the context set cannot be restricted to contain only d_{45}, because then describing d_{53} as *the grey dog* would be sufficient to distinguish d_{53} from d_{45}, its only distractor. Obviously, the introduction of d_{53} to the discourse requires a description which is distinguishing relative to the entire domain, and not only relative to d_{45}. This means that the context set cannot be restricted to d_{45}, even though d_{45} is the only dog that has been mentioned so far.

In sum, restricting the context set to a proper subset of the domain, and then generating all distinguishing descriptions relative to this set does not always produce the desired results. Apparently, different context sets should be used depending on the object to be described. For instance, an object that is being newly introduced to the discourse must be distinguished from all other objects in the domain, whereas an object that has been previously mentioned can have a reduced description. Our proposed solution to this problem is to structure the domain by marking certain objects as more prominent than others. As our metric of prominence we shall use *salience weights*.

1.3.2 Preliminaries

The underlying idea of our modifications is the following:

> A definite description 'the $\overline{\mathrm{N}}$' is a suitable description of an object d in a state s iff d is the most salient object with the property expressed by $\overline{\mathrm{N}}$ in state s

Since the denotation of the $\overline{\mathrm{N}}$ is an important factor, we use the notion of a *value set*. Let L be the list of properties expressed by some $\overline{\mathrm{N}}$. The value set of L in some domain D (notation: $\mathsf{Val}_D(L)$) is the set of objects $d \in D$ which have the properties expressed by L.[7] More formally:

[7] The use of value sets marks a minor deviation from Dale & Reiter. Whereas they use a function $\mathsf{RulesOut}$ which determines the objects which do *not* have a certain property p, we check which objects *do* have this property. This difference is akin to

Definition 1 *(Value sets)*

1. $\mathsf{Val}_D(\langle A, V\rangle) = \{d \in D \mid d \textit{ has the property expressed by } \langle A, V\rangle\}$,
2. $\mathsf{Val}_D(\{p_1, \ldots, p_n\}) = \mathsf{Val}_D(p_1) \cap \ldots \cap \mathsf{Val}_D(p_n)$,
   where $p_i = \langle A_i, V_i\rangle (1 \leq i \leq n)$.

Thus, as an example: $\mathsf{Val}_{D_1}(\{\langle \text{colour}, \text{white}\rangle, \langle \text{type}, \text{chihuahua}\rangle\})$ amounts to $\mathsf{Val}_{D_1}(\langle \text{colour}, \text{white}\rangle) \cap \mathsf{Val}_{D_1}(\langle \text{type}, \text{chihuahua}\rangle) = \{d_2, d_4\} \cap \{d_1, d_2\} = \{d_2\}$. The domain subscript and the attributes are omitted when this does not lead to confusion; e.g., we write Val(small, chihuahua). By definition, the value set of the empty list of properties is the entire domain ($\mathsf{Val}_D(\{\}) = D$). Following common practice, we use $|S|$ to denote the cardinality of a set S.

How can we model the salience of an object? For that purpose we use a function variable sw (salience weight) which per state represents a function mapping elements in the domain to a natural number.[8] For the sake of simplicity, we shall assume that in the initial state (s_0), say the beginning of the generation process, all objects are minimally salient (represented as a zero salience level). Formally, $\forall d \in D : sw(s_0, d) = 0$. Below, in section 1.4, we discuss and compare two methods for salience weight assignment and offer a useful synthesis of the two. For the time being we simply assume that the salience weights are given. So let sw be the function assigning salience weights, then we can define that an object r is the most salient object having certain properties L in a state s (notation: $\mathsf{MostSalient}(r, L, s)$) if, and only if, every object in $\mathsf{Val}(L)$ different from r has a lower salience weight in s than r itself.[9]

Definition 2 *(Salience condition)*
$\mathsf{MostSalient}(r, L, s) \Leftrightarrow \forall d \in \mathsf{Val}(L)(d \neq r \rightarrow sw(s, d) < sw(s, r))$

1.3.3 Outline of the Modified Algorithm

Figures 1 and 2 contain our proposal for a modified algorithm in pseudo-code. We have stuck as closely as possible to the algorithm from Dale & Reiter (1995:257) to ease comparison. Below, we illustrate it with a

the difference between a cup which is half full and one which is half empty. We find the use of value sets somewhat more intuitive from a semantic point of view.

[8] It is also possible to assign salience weights to *groups* of objects (such as focus spaces). Notice that the point-wise assignment can be mapped onto a group-wise assignment, but not vice versa. The additional information that point-wise assignments have is potentially useful for the sake of pronominalization. We do not believe, however, that there is a knock-down argument for either of the alternatives. For that the two are too similar.

[9] A computationally more efficient version would intersect the value set of L with the set of objects which are equally or more salient than the referent r (that is $\{d \in D \mid sw(s, r) \leq sw(s, d)\}$). See Theune (2000).

$\boxed{\mathsf{MakeReferringExpression}\ (r, P, s)}$

$L \leftarrow \{\}$, $tree \leftarrow \mathsf{nil}$, $contrast \leftarrow$ **false**
for each member A_i of list P **do**
  $V \leftarrow \mathsf{FindBestValue}\ (r, A_i, \mathsf{BasicLevelValue}(r, A_i), s)$
  $contrast \leftarrow \mathsf{Contrastive}(r, A_i, V)$
  $tree' \leftarrow \mathsf{UpdateTree}\ (tree,\ V,\ contrast))$
  if $(|\mathsf{Val}(L \cup \{\langle A_i, V\rangle\})| < |\mathsf{Val}(L)|\ \vee A_i = \text{type}) \wedge\ tree' \neq \mathsf{nil}$
  then $L \leftarrow L \cup \{\langle A_i, V\rangle\}$
    $tree \leftarrow tree'$
  endif
  if $\mathsf{MostSalient}\ (r, L, s) =$ **true**
  then $tree \leftarrow \mathsf{AddDefDet}\ (tree)$
    return *tree*
  endif
return failure

FIGURE 1 Sketch of the main function of the modified algorithm.

number of examples. First, we give a general overview.

The algorithm is called by MakeReferringExpression (r, P, s) (figure 1); that is, we try to generate a definite description for a referent r given some pre-defined list P of *preferred attributes* in a state s. L is the list of properties which have been selected for inclusion in the expression generated and is initialized as the empty list. The variable *tree* contains the syntactic tree for the NP under construction which corresponds with the current list of properties L. Finally, *contrast* is a boolean variable which indicates whether the property under consideration is contrastive or not. As in the original version of the algorithm, the main loop iterates through the list P of preferred attributes. For each attribute A on this list, the best value V is sought (essentially in the same way as done in the Incremental Algorithm, using the function FindBestValue, see figure 2). Once the best value V is found, the algorithm immediately checks whether the property is contrastive or not (using the function Contrastive, figure 2), and attempts to incorporate V in the NP under construction (using the function UpdateTree). The property $\langle A, V\rangle$ is actually included in the description if it 'shrinks' the value set (and thus rules out one or more distractors) or if the attribute A is 'type',[10] pro-

[10]This latter option marks a minor deviation from Dale & Reiter. They always include the 'type', but only *check* whether it was included after a distinguishing

$\boxed{\mathsf{FindBestValue}(r, A, \mathit{initial\text{-}value}, s)}$

if $\mathsf{UserKnows}(r, \langle A, \mathit{initial\text{-}value}\rangle) = \mathbf{true}$
then $\mathit{value} \leftarrow \mathit{initial\text{-}value}$
else $\mathit{value} \leftarrow \mathsf{novalue}$
endif
if $\mathsf{MostSalient}(r, \{\langle A, \mathit{value}\rangle\}, s) = \mathbf{false} \wedge$
$(\mathit{msv} \leftarrow \mathsf{MoreSpecificValue}(r, A, \mathit{value})) \neq \mathsf{novalue} \wedge$
$(\mathit{new\text{-}value} \leftarrow \mathsf{FindBestValue}\ (r, A,\ \mathit{msv},\ s)) \neq \mathsf{novalue}\ \wedge$
$|\mathsf{Val}(\{\langle A, \mathit{new\text{-}value}\rangle\})| < |\mathsf{Val}(\{\langle A, \mathit{value}\rangle\})|$
then $\mathit{value} \leftarrow \mathit{new\text{-}value}$
endif
return value

$\boxed{\mathsf{MostSalient}(r, L, s)}$

if $\forall d \in \mathsf{Val}(L)(d \neq r \Rightarrow sw(s,d) < sw(s,r))$
then return true
else return false
endif

$\boxed{\mathsf{Contrastive}(r, A, V)}$

$LC \leftarrow \{d \in \mathsf{DR}(\mathsf{PrevS} \cup \mathsf{CurrS}) \mid d \neq r \wedge$
$\mathsf{Parent}(\mathsf{BasicLevelValue}(d, \mathrm{type})) =$
$\mathsf{Parent}(\mathsf{BasicLevelValue}(r, \mathrm{type}))\}$
if $\exists d \in LC : \mathsf{Value}(d, A) \neq V$
then return true
else return false
endif

FIGURE 2 Sketch of auxiliary functions of the modified algorithm.

vided that the function UpdateTree was successful. If this is not the case (i.e., the lexical or syntactic restrictions of the generation module make it impossible to express the property), V is rejected. After that it is checked whether the intended referent r is the most salient object in the current state of the discourse which satisfies L (using the function MostSalient (r, L, s), see figure 2). If so, the function AddDefDet inserts the determiner *the* to produce a full definite description and the algorithm succeeds.

The function Contrastive (r, A, V) checks whether the property under consideration is contrastive, i.e., if it serves to distinguish the object r in some linguistic context LC. LC is defined as the set of objects/discourse referents (DRs) which are referred to in the previous sentence (PrevS) or in the sentence currently being generated (CurrS), and of which the basic level value of the attribute 'type' has the same parent as that of r. A property $\langle A, V\rangle$ of r is considered to be contrastive if there is an element $c \in LC$ which has a different value for the current attribute. Thus, loosely speaking, the adjective *large* in the NP *the large dog* is marked as contrastive in the context of *a small cat* but not in the context of *a small car*.[11]

In contrast with the rest of the algorithm, the function UpdateTree is largely domain and language dependent. Since this function is not the focus of our work, we do not provide a detailed specification of it here, but restrict ourselves to a broad sketch based on a few rather simplified assumptions. Roughly, UpdateTree works as follows: starting from a prototypical NP structure, the function attempts to integrate each new value V in the syntactic tree constructed so far. We assume that the value of the 'type' attribute (and no other value) is always realised as the head noun, and therefore can always be included in the tree under construction.[12] Unary properties are added as prenominal AP modifiers to designated slots in the tree.[13] Relations (discussed in

description has been generated. The result will be the same in all cases where 'type' is first on the list of preferred attributes.

[11]This treatment of contrast is closely related to the proposal of Prevost (1996), who presents an algorithm for deciding which properties should receive contrastive accent in a manner which, incidentally, is somewhat similar to the Incremental Algorithm. See Theune (1997, 2000) for some further discussion on Prevost's approach to contrast.

[12]This is a simplification, as has been pointed out by Horacek (1995) among others, which we make following Dale & Reiter (1995). In fact, we believe that the decision to include type information depends on the kind of domain and on the list of preferred attributes. For example, in a domain where all objects are basically of the same type, type information is unnecessary and should thus not be included at all; see van der Sluis & Krahmer (2001).

[13]For a discussion of adjective orderings, see Dale 1992:127-130, and the references

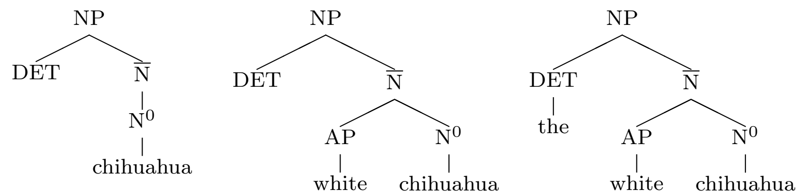
section 1.6.2) may be realized as postnominal PPs or relative clauses. Following Horacek (1997), we allow each available slot in the prototypical NP structure to be filled only once. If this implies that the current property cannot be added to the tree under construction, UpdateTree returns nil. Below we assume that UpdateTree always succeeds unless noted otherwise. If *contrast* is **true**, the expression of the value V is marked by a [+c] feature, which can be taken into account during the computation of prosody in a spoken language generation system. No lexical selection takes place: we assume that each attribute-value pair is associated with a fixed lexicalized tree; e.g., $[_{AP}$ *small*$]$ for $\langle$ size, small $\rangle$.

1.3.4 Examples

First example: non-anaphoric description Reconsider our example domain D_1, and suppose that we start generating a monologue in the initial state s_0. Thus, by assumption, all elements of the domain are equally salient. Now we want to generate an expression for d_2. MakeReferringExpression (d_2, P, s_0) is called, assuming that P is $\langle$ type, colour, size $\rangle$. The list of properties L is initialized as $\{\}$, *tree* as nil and *contrast* as **false**. We consider the first property of d_2, $\langle$ type, chihuahua $\rangle$. The best value for this attribute is 'chihuahua', since $|\mathsf{Val}(\text{chihuahua})| = 2 < |\mathsf{Val}(\text{dog})| = 3$. The property 'type' is always included, but note that it is also informative in that it rules out two distractors: $|\mathsf{Val}(\text{chihuahua})| = 2 < |\mathsf{Val}(\{\})| = 4$. After one iteration, the description under construction is tree (I) from figure 3. The value of L is now $\{\langle$ type, chihuahua $\rangle\}$. $\mathsf{MostSalient}(d_2, \text{chihuahua}, s_0)$ fails because d_1 is also a chihuahua, and d_1 and d_2 are both minimally salient by assumption. So we proceed by taking the second property of d_2, $\langle$ colour, white $\rangle$. Now $|\mathsf{Val}(\text{white, chihuahua})| < |\mathsf{Val}(\text{chihuahua})|$; this property is discriminating and thus included. After adding the property 'white' to the current NP tree, the tree looks as in (II, figure 3). Now, the condition $\mathsf{MostSalient}(d_2, \{\text{white, chihuahua}\}, s_0)$ *is* true: d_2 is the only white chihuahua in the domain, so it is by definition the most salient one. The definite article is added, and the resulting tree (III, figure 3) is returned. When the description is conveyed, d_2 increases in salience, becoming more salient than the other objects in the domain.

Notice that when we assume that all objects in the domain are always equally salient (thus *sw* represents the constant function mapping each $d \in D$ to some n), the modified algorithm selects exactly the same properties as the original Incremental Algorithm. In other words, our version truly generalizes the original, which brings us to the next example.

cited therein. For an alternative, data-oriented approach, see Malouf (2000).



(I) (II) (III)

FIGURE 3 The three stages of generating the first example description.

Second example: anaphoric description An anaphoric description generally contains less information than its antecedent. This may be reflected by the omission of properties, by the use of a more general head noun, or by both, as this example shows. Suppose we want to refer to d_2 in a situation s' where d_2 is more salient than the other animals in the domain (e.g., because d_2 was referred to in the previous sentence). The BasicLevelValue for d_2 with attribute 'type' is 'dog', and since d_2 is currently the most salient dog in the domain, the modified Incremental Algorithm immediately succeeds and returns a tree for *the dog*.[14]

Third example: contrast Now suppose that the preceding sentence referred to both d_1 and d_2 (e.g., '*the black chihuahua and the white chihuahua ...*') and that we call MakeReferringExpression (d_2, P, s). Then the first property added to the description under construction will be 'chihuahua' (as the reader may check for herself). The second property, 'white' is also added to the description, since it distinguishes d_2 from d_1. Moreover, this property is *contrastive*, since d_1 and d_2 have the same basic level value for the 'type' attribute and different values for the 'colour' attribute. The result is '*the white*$^{[+c]}$ *chihuahua*' (where [+c] is a feature indicating a contrastive relation).

1.3.5 Discussion

We have described a generalization of the Incremental Algorithm which extends the original in a number of ways. First and foremost, it explicitly takes the discourse context into account by treating context sets as a combination of a discourse domain with a salience function. This generalization entails that creating a referring expression for one object (here d_2 of domain D_1) can result in e.g., *the white chihuahua* (in the initial

[14] As we will see, the experiment in section 1.5 shows that it is not always a good idea to use more general head nouns. Below the algorithm will be improved with respect to this issue.

state), *the dog* (in a context like *the white chihuahua* ...), *the dog*$^{[+c]}$ (sample context: *the white chihuahua and the cat*) and *the white*$^{[+c]}$ *chihuahua* (context: *the black chihuahua and the white chihuahua* ...). This does not make the algorithm more complex: it still has a polynomial complexity, and a similar theoretical run time as the original algorithm $(n_d n_l)$.[15]

1.4 Determining Salience Weights

So far we did not discuss how salience weights are computed. The literature contains various methods to do so, such as Alshawi (1987), Hajičová (1993), Lappin & Leass (1994), and Grosz *et al*. (1995). The basic idea underlying these methods is the same: objects that have (recently) been mentioned are more salient than objects that have not been mentioned. This basic idea is worked out differently for each method, for instance in the way that factors like syntactic position or discourse function are taken into account. As a consequence, the choice for a specific method of salience weight assignment may influence the outcome of the algorithm. As an illustrative example, we discuss and compare the hierarchical focusing constraints of Hajičová (1993) and the Centering approach of Grosz *et al*. (1995). Throughout this section we let *sw* be a total function mapping the objects in a domain D to the set $\{0, \ldots, 10\}$, with the intuition that 0 represents complete non-salience and 10 maximal salience.[16] We continue assuming that in the initial situation s_0 all objects in the domain are equally (not) salient: $sw(s_0, d) = 0$ for all $d \in D$.

1.4.1 Hajičová: Hierarchical Focus Constraints

Salience weight assignment in the Praguian approach is based on the notions *topic* and *focus* (see e.g., Sgall *et al.* 1986 for precise definitions of these notions). Informally speaking, the topic of utterance U is what U is about (generally this corresponds to the grammatical subject), while the focus[17] of U is what the sentence says about its topic (the rest of the sentence). Informally, the Praguian approach says that objects which have been referred to in the focus part of a sentence have maximal salience; they constitute new information that is assumed to

[15] To see this, observe that the modified algorithm still requires as much iterations as properties realized in the description (n_l) and in each of the iterations has to inspect the set of distractors (n_d).

[16] We are not committed to an 11-point scale of salience *per se*. Any finite subset of the integers will do. In general, it may well be that more than 11 objects are referred to in a particular sentence, but this poses no problems for an 11-point scale because not every object needs to be assigned a unique salience weight.

[17] Notice that the notion of focus used here is very different from that used in e.g., Grosz & Sidner (1986) and Grosz *et al.* (1995).

be in the centre of attention. Objects which have been referred to in the topic have a lower salience weight than those referred to in the focus. Their exact weight is determined by the form of the referring expression that has been used: if a non-pronominal NP has been used, the highest-but-one salience weight is assigned, and if a pronoun has been used, the object's salience weight does not change. If an object has not been referred to in a sentence (i.e., it is part of neither focus nor topic) then its salience weight decreases. The salience weight of objects that have been previously referred to in the topic part of a sentence decreases at a slower rate than that of objects that have been previously referred to in the focus. Formally:

Definition 3 *(Salience weight assignment based on Hajičová (1993))* *Let* U_i *be a sentence uttered in state* s_i*, and let* $\mathsf{topic}(U_i) \subseteq D$ *and* $\mathsf{focus}(U_i) \subseteq D$ *be the sets of objects which are referred to in the topic and the focus part of* U_i *respectively. Then the salience weight of objects in* s_{i+1} *is determined as follows:*

$$sw(s_{i+1}, d) = \begin{cases} 10 & \textit{if } d \in \mathsf{focus}(U_i) \\ 9 & \textit{if } d \in \mathsf{topic}(U_i), \\ & \quad \textit{and } d \textit{ is referred to by a definite NP} \\ sw(s_i, d) & \textit{if } d \in \mathsf{topic}(U_i), \\ & \quad \textit{and } d \textit{ is referred to by a pronoun} \\ \mathsf{max}(0, sw(s_i, d) - 1) & \textit{if } d \notin \mathsf{topic}(U_i) \cup \mathsf{focus}(U_i), \\ & \quad \textit{and } d \in \mathsf{topic}(U_j), j < i \\ \mathsf{max}(0, sw(s_i, d) - 2) & \textit{if } d \notin \mathsf{topic}(U_i) \cup \mathsf{focus}(U_i), \\ & \quad \textit{and } d \in \mathsf{focus}(U_j), j < i \end{cases}$$

The salience function given in definition 3 above closely corresponds to the rules given by Hajičová (1993), except that in our version objects are mapped to a *higher* number as they become more salient, with a maximum of 10, whereas in Hajičová (1993) objects are assigned a *lower* number as they become more salient, maximal salience being indicated by 0. As a consequence, the Praguian approach allows for an infinite decrease in salience weight, keeping track of objects which have faded from the discourse long ago. We do not think this is psychologically realistic, so we assume there is not only a maximal but also a minimal salience level: in our version, the salience weight of an object cannot decrease below zero.

1.4.2 Grosz *et al.*: Centering

An alternative definition of salience weight assignment can be based on the ranking of the so-called *forward looking centers* (C_f) of an utterance. According to Centering Theory (Grosz *et al.* 1995), the set of forward looking centers of an utterance contains the objects referred to in that utterance. This set is partially ordered to reflect the relative prominence of the referring expressions within the utterance. Grammatical roles are a major factor here, so that *subject* > *object* > *other.*

Definition 4 *(Salience weight assignment based on Centering Theory) Let U_i be a sentence uttered in state s_i, in which reference is made to $\{d_1, \ldots, d_n\} \subseteq D$. Let $C_f(U_i)$ (the forward looking centers of U_i) be a partial order defined over $\{d_1, \ldots, d_n\} \subseteq D$. Then the salience weight of objects in s_{i+1} is determined as follows:*

$$sw(s_{i+1}, d) = \begin{cases} 0 & \textit{if } d \notin C_f(U_i) \\ n & \textit{otherwise, where } n = \mathsf{level}(d, C_f(U_i)) \end{cases}$$

Here, $\mathsf{level}(d, C_f(U_i))$ refers to the level of the occurrence of d in the ordering $C_f(U_i)$, defined in such a way that the highest element(s) on the ordering are mapped to 10, the element(s) immediately below are mapped to 9, etc. Thus, $\mathsf{level}(d_i, \langle\, d_1, \ldots, d_n\rangle) = \mathsf{max}(0, 11 - i)$, where $\langle d_1, \ldots, d_n\rangle$ is the partially ordered set of forward looking centers of the relevant utterance.

1.4.3 Examples, Predictions and Comparison

Here we discuss two illustrative examples in some detail, emphasizing the repercussions of choosing either way of assigning salience weights for the generation of referring expressions. We use sw^h to indicate the salience weights as assigned by the Hajičová-based approach (definition 3), while sw^c gives the salience weights as based on the Centering approach (definition 4). Indices on words refer to objects in the domain. In the example sentences, the Praguian topic always coincides with the syntactic subject. Suppose the domain of discourse is D_1, and consider the following example.

(2) a. The$_2$ white chihuahua was angry.
$sw^h(s_1, d_2) = 9$
$sw^c(s_1, d_2) = 10$

b. It$_2$ viciously attacked the$_1$ black chihuahua.
$sw^h(s_2, d_1) = 10$ $\quad$ $sw^h(s_2, d_2) = 9$

$sw^c(s_2, d_1) = 9$ $\quad sw^c(s_2, d_2) = 10$

c. { The$_1$ dog (H)/The$_1$ black dog (C) } barked loudly.
$sw^h(s_3, d_1) = 9$ $\quad sw^h(s_3, d_2) = 8$
$sw^c(s_3, d_1) = 10$ $\quad sw^c(s_3, d_2) = 0$

First we take the Praguian point of view, using sw^h as defined above. In (2)a the modified algorithm produces the description *the white chihuahua* to refer to d_2. Thereby, d_2 becomes the most salient object, with the highest-but-one degree of salience, as it is introduced in the topic of the sentence. In (2)b a new object, d_1, is introduced to the discourse using the description *the black chihuahua*. Since this is done in the focus of the sentence, d_1 rises to maximal salience. Object d_2 is referred to by a pronoun occurring in the topic of the sentence, so its salience weight remains the same.[18] In (2)c, d_1 is referred to for the second time, and as it is currently the most salient dog, the modified algorithm generates the description *the dog*. This result, though marginally acceptable, is not what we want. A simple solution to this problem is to ignore small differences in activation degree between competitors (see also Kruijff-Korbayová & Hajičová 1997:41).

Let us now discuss example (2) from the Centering perspective. After the first sentence has been uttered, we find that C_f ((2)a) is the singleton set containing d_2, and as a result this object is now the most salient object. In (2)b we refer to both d_2 and d_1, and C_f ((2)b) therefore contains both d_1 and d_2. Since d_2 is referred to in subject position, it is ranked higher than d_1. Consequently, to refer to d_1 in (2)c the modified Incremental Algorithm correctly produces *the black dog*.

Now consider the following discourse, based on the dog show example from section 1.3.1. Suppose Joe went to the dog show and bought the dogs d_{45} and d_{53}.

(3) a. Joe bought the$_{45}$ large black long-haired sausage dog and the$_{53}$ small grey pygmy poodle with the perm wave.
$sw^h(s_1, joe) = 9$ $\quad sw^h(s_1, d_{45}) = 10$ $\quad sw^h(s_1, d_{53}) = 10$
$sw^c(s_1, joe) = 10$ $\quad sw^c(s_1, d_{45}) = 9$ $\quad sw^c(s_1, d_{53}) = 9$

b. The$_{45}$ sausage dog was a bargain.
$sw^h(s_2, joe) = 8$ $\quad sw^h(s_2, d_{45}) = 9$ $\quad sw^h(s_2, d_{53}) = 8$
$sw^c(s_2, joe) = 0$ $\quad sw^c(s_2, d_{45}) = 10$ $\quad sw^c(s_2, d_{53}) = 0$

c. { The$_{53}$ poodle (H) / The$_{53}$ small grey pygmy poodle with the

[18] We address the problem of pronoun generation in section 1.6.1 below.

perm wave (C) } was very expensive though.
$sw^h(s_3, joe) = 7$ $sw^h(s_3, d_{45}) = 8$ $sw^h(s_3, d_{53}) = 9$
$sw^c(s_3, joe) = 0$ $sw^c(s_3, d_{45}) = 0$ $sw^c(s_3, d_{53}) = 10$

Comparing the two different salience weight assignments in this example presents the following picture. After generation of the first sentence, both approaches assign high salience weights to *joe* and the dogs d_{45} and d_{53}. The second sentence only contains a reference to d_{45} (*the sausage dog*). Using the Centering approach, this reduces the salience weight of *joe* and d_{53} to zero as they are not mentioned in (3)b, whereas using the Hajičová approach entails that the salience weights of *joe* and d_{53} are reduced much less. The difference in salience weight reduction for d_{53} greatly influences its description in (3)c. Seen from a Praguian perspective, d_{53} is still the most salient poodle at this stage, and the generated description is *the poodle*. However, from the Centering perspective, d_{53} is not salient at all and the modified Incremental Algorithm again produces the description *the small grey pygmy poodle with the perm wave*, just as it did for the first-mention of this dog in (3)a. In our opinion, this shows that the Centering assumption that only objects mentioned in the previous sentence can have a non-zero salience weight, is too strong from the perspective of definite descriptions. Again, there is an obvious way to remedy this shortcoming: we can take the structure of the discourse into account (see e.g., Walker 1998 for a proposal to this effect).

1.4.4 Discussion and Synthesis

There are several principled ways to determine salience weights that can be used in the modified algorithm. Here, we have discussed and compared two of them: one is based on the hierarchical focusing constraints of Hajičová (1993) and the other on the Centering approach of Grosz *et al.* (1995). Probably the most important distinctions between the two are (i) the Praguian assumption that information introduced in the focus of an utterance has a somewhat higher salience weight than information introduced in the topic, as opposed to the Centering assumption that information referred to in subject position is more salient than information referred to in object position,[19] and (ii) the Centering assumption that only objects mentioned in the previous sentence can have a non-zero salience weight, while in the Praguian approach the determination of salience weights is not restricted to the previous sentence.

[19] In most cases, but not always, the subject of a sentence refers to topical information. In those cases, the approaches we have discussed make different predictions.

We have discussed two examples illustrating the differences between the two approaches. In the first case, the Centering approach yields better results, while in the second case, it is the Hajičová way of determining salience weights which pays off. These examples suggest that the Centering ordering of salience is preferable over the Praguian topic/focus ordering, but that a gradual decrease in salience of non-mentioned objects gives better results than an abrupt loss of salience. An obvious step is therefore to combine the Centering definition of salience with a mechanism for gradual decrease. Here, we opt for the most simple form of decrease, where the salience of all discourse objects that have not been mentioned in the current sentence decreases by one. (This simplification of the Praguian approach is sufficient for current purposes.) Thus, we arrive at definition 5:

Definition 5 *(Revised salience weight assignment)*
Let U_i be a sentence uttered in state s_i, in which reference is made to $\{d_1, \ldots, d_n\} \subseteq D$. Let $C_f(U_i)$ (the forward looking centers of U_i) be a partial order defined over $\{d_1, \ldots, d_n\} \subseteq D$. Then the salience weight of objects in s_{i+1} is determined as follows:

$$sw(s_{i+1}, d) = \begin{cases} \mathsf{level}(d, C_f(U_i)) & \textit{if } d \in C_f(U_i) \\ \mathsf{max}(0, sw(s_i, d) - 1) & \textit{if } d \notin C_f(U_i) \textit{ and } d \in C_f(U_j), j < i \end{cases}$$

Using the above definition, the modified algorithm produces the intuitively correct results for the examples discussed so far (as the reader may check). We certainly do not consider definition 5 as the ultimate way of salience weight assignment. It probably needs to be refined to account for salience fluctuations in more complicated examples, e.g., containing complex grammatical constructions. In this respect it would be highly interesting to fine-tune definition 5 using results from more recent, corpus-based approaches to salience (Lappin & Leass 1994, Popescu-Belis *et al.* 1998, Poesio & Viera 1998, and McCoy & Strube 1999). For the time being, definition 5 provides us with a workable salience function to be used in the modified algorithm and at least shows that it is possible to assign salience weights in a principled manner.

For the generation of reduced anaphoric descriptions, it is very important to have some form of salience weight assignment, ensuring that objects which have been recently mentioned are regarded as being (much) more salient than the other objects in the domain. Without such a mechanism, the MakeReferringExpression algorithm would generate a full description of an object at every mention of this object, using the same description throughout the output text. For the generation of reduced definite descriptions, in most cases it seems sufficient to only keep track of large differences in salience weight (e.g., between mentioned versus unmentioned entities); small differences (e.g., between entities that have been mentioned in different positions in the same sentence) appear to be mainly relevant for pronominalization (discussed in section 1.6.1). So far, we have set our 'salience threshold' to one: a referent r counts as being more salient than other objects in the domain if its salience weight is at least one point higher than that of those other objects (in other words, only those objects that are at least equally salient as r count as its distractors). However, to stay on the safe side we might set the threshold higher. A consequence of such a measure would be the generation of fewer reduced descriptions, which reduces the risk of ambiguity but also reduces coherence of the generated texts. In general, the choice between fewer or more reduced anaphoric descriptions involves a trade-off between incoherence and ambiguity. Less ambiguity (due to fewer reduced descriptions) entails more incoherence, and vice versa. The desired ratio between the two might depend (among other things) on text genre (Reiter p.c.). For instance, in legal texts coherence is far less important than unambiguous reference, as observed by Maes (1991).

Finally, a word on the computational consequences of adding a salience function to the algorithm. In general, the determination of salience adds little computational overhead. To compute a new salience function only the weights of the objects mentioned in the current clause and the objects with a non-zero salience weight have to be updated. Even for huge domains, the latter set is highly restricted, containing only the objects mentioned in the last few sentences.

1.5 Experimental Evaluation

Some of the hypotheses underlying the modified Incremental Algorithm have been experimentally tested using a forced choice experiment. In this experiment, the subjects had to indicate for a number of texts which of two versions of these texts they found most natural. The two versions differed only with respect to the description of one item. In this section, the experiment is described and its results are discussed.

1.5.1 Hypotheses and Assumptions

The following hypotheses were tested in the experiment:

> **Hypothesis I**: People prefer anaphoric descriptions that contain fewer properties than their antecedents.

Hypothesis II: People prefer anaphoric descriptions that express attribute values which are closer to the basic level value than those expressed in their antecedents.

Hypothesis III: If an anaphoric expression can refer to only one object, people prefer the anaphor to be pronominalized.

Hypothesis IV: Hypotheses I and II apply also if there is an intervening sentence between the anaphor and its closest antecedent. (This hypothesis is divided into two parts: IVa relates to hypothesis I, and IVb relates to hypothesis II.)

By testing the first three hypotheses, we intended to find out if people really prefer the reduced, generalized or pronominalized descriptions (see section 1.6.1) generated by the modified algorithm over the full descriptions that would be generated if the effects of salience were not taken into account. The fourth hypothesis was added to test if a discourse object still remains salient after it has not been mentioned for one sentence (the gradual decrease in salience discussed in section 1.4).

1.5.2 Method

The modified algorithm is meant to generate referring expressions which will be preferred by human hearers, rather than to reproduce the variety of referring expressions which human speakers produce in various contexts. Therefore a forced-choice experiment was used to test the hypotheses presented above, rather than a production experiment. In the experiment, the subjects had to indicate their preference for one of two alternative referring expressions.

The experiment was performed by 51 naive subjects. All subjects were native speakers of Dutch, except one, whose mother tongue was English. The subjects (of different ages and backgrounds) were presented with 32 texts in Dutch, displayed in two versions on a computer screen. For each text, they had to indicate which of its two versions they found most natural. The texts were presented in a random order. The experiment was self-paced.

1.5.3 Materials

The texts used in the experiment were constructed to test the hypotheses presented above. There were eight texts per hypothesis. (In the case of hypothesis IV, four of these texts were associated with IVa and four texts were associated with IVb.) The texts were written in Dutch and

Hypothesis I: omission of properties

De poedel en de grote terrier maakten ruzie over een bot.
{ *De terrier / De grote terrier* } *ging er met het bot vandoor.*

The poodle and the large terrier had a fight over a bone.
{ The terrier / The large terrier } ran off with the bone.

Hypothesis II: more basic values

De poedel en de kat woonden op de boerderij.
{ *De hond / De poedel* } *waakte over de kippen.*

The poodle and the cat were living at the farm.
{ The dog / The poodle } guarded the chickens.

Hypothesis III: pronominalization

De zwarte siamese kat had honger.
{ *Hij / De kat* } *ging op zoek naar een visje in de keuken.*

The black siamese cat was hungry.
{ It / The cat } went to the kitchen to look for a fish.

Hypothesis IVa: intervening sentences (I)

John kocht een grote witte chihuahua en een poedel in het asiel.
De poedel was een koopje.
{ *De chihuahua / De grote witte chihuahua* } *was iets duurder.*

John bought a large white chihuahua and a poodle at the asylum.
The poodle was a bargain.
{ The chihuahua / The large white chihuahua } was a bit more expensive.

Hypothesis IVb: intervening sentences (II)

De bruine rottweiler en de zwarte rottweiler lagen in de tuin te slapen.
Plots begonnen twee grijze katten te vechten in de tuin.
{ *De bruine hond / De bruine rottweiler* } *werd wakker door het lawaai.*

The brown rottweiler and the black rottweiler were sleeping in the garden.
Suddenly, two grey cats started fighting in the garden.
{ The brown dog / The brown rottweiler } was awakened by the noise.

FIGURE 4 Examples of texts associated with hypotheses I to IV.

	Significant preference for A	Significant preference for B	No significant preference
Hypothesis I	8^a	0	0
Hypothesis II	2^b	5^c	1^d
Hypothesis III	8^a	0	0
Hypothesis IV	4^a	3^a	1^e

a: $p < 0,001$
b: for one text, $p < 0,005$ and for the other text $p < 0,001$
c: for one text, $p < 0,01$ and for the other four $p < 0,001$
d: non-significant preference for A $(0,1 < p < 0,25)$
e: non-significant preference for B $(0,25 < p < 0,5)$

TABLE 1 The number of texts for each hypothesis, for which (*i*) a significant number of subjects preferred version A; (*ii*) a significant number of subjects preferred version B; (*iii*) there was no significant preference for either version.

consisted of two or three sentences. Each text had two versions, A and B, that were the same except for the description of one discourse item (the grammatical subject in the final sentence). The description in the A version was in line with the hypothesis associated with the modified algorithm, whereas the description in the B version was overspecified according to the same hypothesis. Both descriptions were distinguishing. Figure 4 shows an example text for each of the hypotheses. To save space, we do not show both versions of each text. Instead, we show only one text with the pair of differing descriptions in the last line. The first description of each pair is the description that is in line with the corresponding hypothesis, and the second one is the overspecified description.

1.5.4 Results

Table 1 shows the results of the experiment. For each hypothesis, it shows the number of texts for which (*i*) a significant number of subjects preferred version A; (*ii*) a significant number of subjects preferred version B; (*iii*) there was no significant preference for either version. Significance was computed using the χ^2 test.

The results of the experiment can be summed up as follows. For all texts associated with hypotheses I and III, a highly significant number of subjects ($p <$ 0,001) preferred version A, containing a description in line with the associated hypothesis. However, for only two of the texts associated with hypothesis II, the subjects had a significant preference for version A. For five of the texts associated with this hypothesis, there was a significant preference for version B. Hypothesis IV also showed mixed results: for half of the texts, version A was significantly preferred, and for three other texts, B was significantly preferred. This difference in preference corresponds closely to the division of hypothesis IV into parts IVa and IVb. As shown in Table 2, for the texts associated with hypothesis IVa the A version is significantly preferred in three of the four cases, whereas for the texts associated with hypothesis IVb it is the B version which is significantly preferred in three of the four cases.

	Significant preference for A	Significant preference for B	No significant preference
Hypothesis IVa	3^a	0	1^b
Hypothesis IVb	1^a	3^a	0

a: $p < 0,001$
b: non-significant preference for B ($0,25 < p < 0,5$)

TABLE 2 The results for hypothesis IV, divided into two groups of text pairs corresponding to hypothesis I and hypothesis II respectively.

1.5.5 Discussion

The experimental results confirm hypotheses I and III: the subjects in the experiment showed a significant preference for anaphoric descriptions that contained fewer properties than the antecedent, or that were pronominalized. A plausible explanation for this preference is that the reduced descriptions increased the coherence of the example texts, without giving rise to ambiguity. The findings for hypothesis III are in line with the experimental results of Gordon *et al.* (1993), who found that utterances are more difficult to read if they contain a definite description or a proper name where a pronoun could have been used.

Hypothesis II was not confirmed: for only two of the eight texts, a significant number of subjects preferred the version containing a description with a more basic head noun than the antecedent. For five texts, a significant number of subjects preferred the version that did *not* contain a description with a more basic head noun. If we look at the three[20] texts for which a more basic head noun was preferred (in line with hypothesis II), we see that in all three cases, the head noun in the anaphor is a substring of the more specific head noun of the antecedent, e.g., *siamese cat* → *cat*. In the other five texts, this is not the case; here the more basic head noun is completely different from the more specific one. An example is *poodle* → *dog*. These observations suggest that the preference for repeating the same head noun, observed in five of the eight texts associated with hypothesis II, should be seen as a preference for

[20]This includes one text for which a non-significant majority of the subjects preferred version A; see Table 1.

using the same *wording* in both anaphor and antecedent. This is similar to the 'priming' effect that has been found in dialogues: speakers tend to use the same literal expressions as their interlocutors (see e.g., Levelt & Kelter 1982 and Clark & Wilkes-Gibbs 1986). Presumably, holding on to the same wording (both within and among speakers) should be seen as another method for maintaining the coherence of a discourse. When the more basic head noun is lexicalized as a substring of the more specific one (*cat* versus *siamese cat*), then there is no "switch" to a different wording. In such cases, the use of the more basic head noun is preferred.[21]

Like hypothesis II, hypothesis IVb was not supported by the experimental results. This hypothesis was constructed to test if hypothesis II holds if there is an intervening sentence between anaphor and antecedent. For three of the four texts associated with IVb, the B version was preferred. This may be explained through the fact that the A versions of these three texts contained a head noun with a different wording than the antecedent. As discussed above, hypothesis II does not hold for such head nouns. For the fourth text, the A version was preferred. In this text, the more basic head noun used in the anaphor had the same wording as the antecedent. Thus, for hypothesis IVb we see exactly the same picture as for hypothesis II. On the other hand, for three of the four texts associated with hypothesis IVa (constructed to test if hypothesis I holds if there is an intervening sentence), version A was preferred. This is in line with the results for hypothesis I.

The conclusions we can draw from the experiment are the following. People prefer anaphoric descriptions that contain fewer properties than the antecedent (provided that the referent is the most salient object with the included properties); this holds even if there is an intervening sentence between anaphor and antecedent. So, the strategy of not including non-discriminating properties is correct, as is the gradual decrease of salience argued for in section 1.4. In addition, people prefer pronominalized anaphoric descriptions over non-pronominalized ones (provided that the referent is the single most salient discourse object). This justifies the extension of the modified algorithm to generate pronouns, discussed in section 1.6.1.

Finally, people prefer to use a more basic head noun in the anaphor only if the wording is similar to that of the antecedent; otherwise, they prefer to repeat the same head noun even if this leads to over-specification. Apparently, using the same wording in both antecedent and anaphor

[21] Possibly, the subjects perceived these cases as the omission of a property (hypothesis I) rather than the use of a different head noun (hypothesis II).

$\boxed{\mathsf{FindBestValue}(r, A, \textit{initial-value}, s)}$

if $\mathsf{UserKnows}(r, \langle A, \textit{initial-value}\rangle) = \textbf{true}$
then $\textit{value} \leftarrow \textit{initial-value}$
else $\textit{value} \leftarrow \mathsf{novalue}$
endif
if $(\textit{msv} \leftarrow \mathsf{MoreSpecificValue}(r, A, \textit{value})) \neq \mathsf{novalue} \wedge$
  $(\textit{new-value} \leftarrow \mathsf{FindBestValue}\ (r, A,\ \textit{msv},\ s)) \neq \mathsf{novalue}\ \wedge$
    $[(\mathsf{MostSalient}(r, \{\langle A, \textit{value}\rangle\}, s) = \textbf{false} \wedge$
    $|\mathsf{Val}(\{\langle A, \textit{new-value}\rangle\})| < |\mathsf{Val}(\{\langle A, \textit{value}\rangle\})|) \vee$
    $\mathsf{BetterMatch}\ (r, \langle\ A,\ \textit{new-value}\ \rangle, \langle\ A,\ \textit{value}\ \rangle) = \mathsf{true}\ ]$
then $\textit{value} \leftarrow \textit{new-value}$
endif
return *value*

$\boxed{\mathsf{BetterMatch}(r, \langle\ A,\ \textit{new-value}\ \rangle, \langle\ A,\ \textit{value}\ \rangle)}$

if $\exists c : \mathsf{Antecedent}(c, r) \wedge$
  $\mathsf{SameWording}\ (\langle A,\ \textit{value}\ \rangle, c) = \mathsf{false}\ \wedge$
  $\mathsf{SameWording}\ (\langle A,\ \textit{new-value}\ \rangle, c) = \mathsf{true}$
then return true
else return false
endif

FIGURE 5 FindBestValue extended with a check on wording.

leads to a higher degree of perceived coherence than reducing overspecification through the use of a more general head noun. This finding suggests that an additional check should be made in the FindBestValue function of the modified algorithm (see Figure 5). This check should only apply to anaphoric descriptions (referring to the most salient object with the relevant property). In that case, a more specific value (*new-value*) should be chosen only if its words are 'better matching' than those expressing the more general value. Otherwise, the value is chosen that has the smallest value set. The BetterMatch function returns true if the intended referent r has an antecedent c, and *value* has a different wording than the corresponding value expressed in c while *new-value* does not. To check this, a function SameWording is used which takes as input the antecedent c and an attribute value pair $\langle A, V\rangle$ and returns false if the linguistic realization of $\langle A, V\rangle$ is not a substring of the corresponding expression in c. In the experiment, hypothesis II was only tested for the 'type' attribute, as this attribute is generally expressed as the head noun in a description. However, we assume that the effect of wording also holds for other attributes such as 'colour', so that for instance a cat that has been previously described as *the chestnut cat* will not be anaphorically referred to as *the brown cat*.

We end with a brief example to illustrate the effect of the proposed extension. Reconsider domain D_1, and suppose that d_2 is currently the most salient object in the domain (context: *The white chihuahua ...*). This corresponds with the second example from section 1.3.4. Now we want to generate a second reference to the chihuahua. When determining the best value for its 'type' attribute, we consider the values 'dog' (*value*) and 'chihuahua' (*new-value*). Since d_2 is currently the most salient dog, the first disjunct of the condition is false. However, the *new-value* 'chihuahua' provides a better match with the antecedent and is therefore selected. The resulting description is *the chihuahua*, and not *the dog* as our first proposal would have it. As a second example, consider the situation in which we have a persian cat, a siamese cat and a poodle, where the siamese cat is the most salient one (having been described as *the siamese cat*). When generating a second description for this object we consider the values 'cat' (*value*) and 'siamese cat' (*new-value*). In this case, the MostSalientCondition is **true** for 'cat' (after all, the siamese is the most salient cat in this example domain), hence the first disjunct is false. The second disjunct is also false, since *cat* is a substring of *siamese cat* and SameWording returns true for both comparisons. As a consequence, FindBestValue returns 'cat' (the value of *value*).

1.6 Three Sketches of Further Extensions

So far, we have shown in some detail how the modified Incremental Algorithm can generate reduced definite descriptions which are only distinguishing in context. In addition, we think that the modified algorithm can serve very well as a basis for the generation of other kinds of referring expressions, such as pronouns, and relational and bridging descriptions. In this section we outline three further extensions of the modified Incremental Algorithm. In section 1.6.1 a simple form of pronominalization within the modified algorithm is discussed, in section 1.6.2 it is shown how relational descriptions can be generated by slightly modifying the algorithm and in section 1.6.3 it is outlined how the combination of the two preceding extensions paves the way for the generation of bridging descriptions.

1.6.1 Pronominalization

Reiter & Dale (1997:81) point out that a simple but surprisingly effective strategy for pronominalization is to "use a pronoun to refer to an object if the entity was mentioned in the previous clause, and there is no other entity in the previous clause that the pronoun could possibly refer to". This is a fairly conservative strategy, which has the advantage that it will not often produce incorrect pronominalizations. On the other hand, it has been claimed that a pronoun should be used whenever this is possible.[22] The presence of salience weights in the modified algorithm makes it possible to employ a somewhat less conservative strategy, taking the pronominalization decision *within* the algorithm. The basic idea is as follows: if r is the single most salient object in the domain (thus: r is the most salient object with respect to the empty list of properties) and there is an antecedent for this object in the direct linguistic context, then r can be referred to using a pronoun.

$\boxed{\mathsf{MakeReferringExpression}\ (r, P, s)}$

$L \leftarrow \{\}$, *tree* $\leftarrow$ nil, *contrast* $\leftarrow$ **false**
if MostSalient $(r, \{\}, s) =$ **true** $\wedge\ \exists c : \mathsf{Antecedent}(c, r)$
then *tree* $\leftarrow$ Pronominalize (*tree*, r) $\wedge$
    return *tree*
else for each member A_i of list P **do**
$\vdots$

FIGURE 6 Pronominalization within MakeReferringExpression. The remainder of the algorithm is as given in figures 1 and 2.

We certainly do not offer this as the final answer to the problem of generating pronouns. One obvious limitation is that it does not take the role of semantics and common sense into account (see e.g., Kameyama 1996, Passonneau 1996). The current approach is only concerned with the default approach to pronoun generation, which is purely syntactically motivated, and does not address how semantic or pragmatic information can override the default. The success of this strategy depends fully on the adequacy of the underlying model of salience weight determination, but we contend that the approaches discussed in the previous section could serve as a starting point. Suppose, for example, that we determine salience weights as in definition 4 (based on Centering Theory). If sentence (4)a has been uttered at the onset of a discourse, object d_2 is more

[22] This is a specific instance of the DOAP principle from Williams 1997: "Don't Overlook Anaphoric Possibilities."

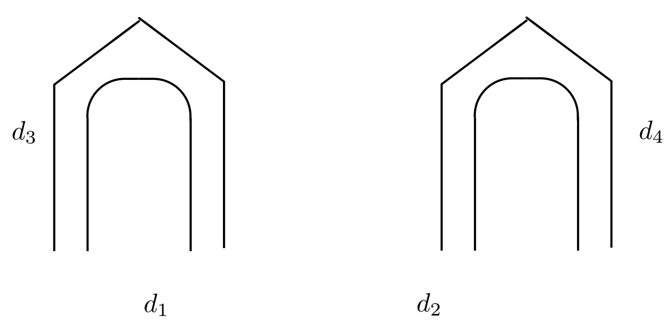
salient than object d_3. If the modified algorithm subsequently generates a referring expression for d_2, it will produce *it*, while a subsequent reference to d_3 will yield a full, anaphoric description. This is intuitively right, in the sense that most people would interpret *it* as referring to the white chihuahua.[23]

(4) a. The$_2$ white chihuahua was chasing the$_3$ cat.
  b. {It$_2$/The$_3$ cat} ran fast.

In pronominalization, the notion of a 'salience threshold' and the trade-off between incoherence and ambiguity are particularly relevant, because pronouns are even more likely to give rise to ambiguities than reduced definite descriptions. For instance, the pronoun *it* in the above example is ambiguous in the sense that it is most likely to refer to d_2, but it could also be interpreted as referring to d_3. This ambiguity can be avoided by raising the salience threshold so that d_2 no longer counts as the single most salient object in the domain. Given the increased salience threshold, d_3 now counts as a distractor from which d_2 has to be distinguished, and this results in the non-ambiguous description of d_2 as *the chihuahua*.

1.6.2 Relational Descriptions

Dale & Haddock (1991) offer an algorithm for the generation of relational descriptions, which is couched in terms of the "Greedy Heuristics algorithm" (Dale 1992). The basic claim we want to make here is that it is possible to generate relational descriptions using a slightly adapted version of the modified Incremental Algorithm and that this has some interesting consequences. Sticking to our continuing cats and dogs theme, let us consider the following situation:



[23]This intuition has been experimentally confirmed by Gordon *et al.* (1993), Walker *et al.* (1994) (for Japanese), Hudson-D'Zmura & Tanenhaus (1998), and others.

In addition to the usual properties (type, colour, size), this new domain, which we denote as D_2, now also includes spatial relations. In particular, d_1 can be found in d_3 and d_2 to the left of d_4. We denote these relations as $\langle$ spatial, in (d_1, d_3) $\rangle$ and $\langle$ spatial, left of (d_2, d_4) $\rangle$ respectively.[24] When an object a is the intended referent and stands in a relation R to some object b, then —following Levelt (1989)— b is referred to as the *relatum* of a.

In addition, we assume that there is a subsumption hierarchy on certain relations. Figure 7 shows a relevant portion of such a subsumption hierarchy for spatial relations. We assume that 'in', 'next to' and 'on top of' are basic level values; the subsuming σ can be thought of as the underspecified spatial relation ("there is some spatial relation between these objects"). The value stored for relations in the domain is the most specific one, just as for the animal types. The advantages of having subsumption hierarchies for relations are discussed below.

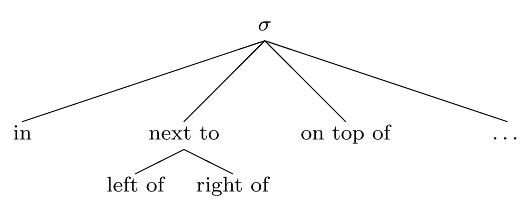


FIGURE 7 Part of the subsumption hierarchy on spatial relations.

It is interesting to note that most of the work on categorization in cognitive science has been concerned with physical objects, and not with relations. A notable exception is Case study 2 in Lakoff (1987). Here Lakoff studies the interrelations between various senses of the preposition *over*, which leads – on a lower level – to a similar structure as shown in Figure 7. However, to the best of our knowledge the notion of basic level values for certain classes of relations has not been studied, and we certainly do not intend to claim psychological reality for our assumptions here.

[24]Of course, d_1 is also located to the left of d_2. However, to keep matters relatively simple, we assume that only spatial relations between objects that are physically close to each other are represented (cf. Horacek 1997:208). The problem of deciding which objects are 'close' may be likened to the problem of deciding which objects are 'large'. In a realistic database, both size and spatial distances would be given in absolute measures (cm/in/pt) and a reasoning component would have to determine which objects are indeed large or nearby. It may be that this kind of (spatial) reasoning in its general form is rather complex and computationally expensive (see e.g., Lemon & Pratt 1997).

Before we show how the MakeReferringExpression algorithm can be extended to include relations in the description of objects, we need to reflect on the ordering of relations and properties. It seems an acceptable assumption that people prefer to describe an object in terms of simple properties, and only shift to relations when properties do not suffice (i.e., properties stand to relations as absolute properties stand to relative properties). This follows from the omnipresent *principle of least effort* (e.g., Zipf 1949, Clark & Wilkes-Gibbs 1986): it takes less effort to consider and describe only one object. How the different kinds of relations (spatial, possessive, etc.) should be ordered is somewhat less obvious; however, it seems likely that people have a preference for relations that are easily perceivable, as is the case with properties. From this it follows that spatial and part-of relations will be preferred over relations that are (usually) less easy to perceive, such as possessive relations. Reasonable as such assumptions seem, we are not aware of any psycholinguistic research into these issues. In the current section, we simply assume that properties are preferred over relations and that spatial relations are preferred over other kinds of relations. Thus, for our example domain D_2, we simply assume that that $P = \langle$ type, colour, size, spatial $\rangle$.

A final preliminary modification that needs to be made is the definition of Value sets. A minor complication is that we now have to keep track of the object whose value set we want to determine (either the first or the second argument of a relation); therefore the intended referent is included as a subscript.

Definition 6 *(Value sets (modified))*
$\mathsf{Val}_{a,D}(\langle A, V\rangle) = \{d \in D \mid d \textit{ has the property expressed by } \langle A, V\rangle\}$
$\mathsf{Val}_{a,D}(\langle A, R(a,b)\rangle) = \{d \in D \mid \exists d' \in D : \langle d, d'\rangle \textit{ stand in relation } R\}$
$\mathsf{Val}_{a,D}(\langle A, R(b,a)\rangle) = \{d \in D \mid \exists d' \in D : \langle d', d\rangle \textit{ stand in relation } R\}$
$\mathsf{Val}_{a,D}(\{p_1, \ldots, p_n\}) = \mathsf{Val}_{a,D}(p_1) \cap \ldots \cap \mathsf{Val}_{a,D}(p_n)$.

For example: $\mathsf{Val}_{d_3,D_2}(\{\langle \text{ spatial, in } (d_1, d_3) \rangle\}) = \{d \in D_2 \mid \exists d' \in D_2 : d'$ is contained in $d\}$ (the set of objects which contain something) $= \{d_3\}$.[25]

Figure 8 shows a version of the modified Incremental Algorithm which is suited for the generation of relational descriptions.[26] The chief

[25]In certain situations the selected properties of the relatum (L_b) are also of interest. So a more general definition of the value set of a relation would be $\mathsf{Val}_{a,D}(\langle A, R(a,b)\rangle) = \{d \in D \mid \exists d' \in \mathsf{Val}_{b,D}(L_b) : \langle d, d'\rangle$ stand in relation $R\}$. For expository reasons, we stick to the simpler definition in the main text.

[26]It has to be kept in mind that the functions FindBestValue, BasicLevelValue and Contrastive are polymorphic here, in that the A_i argument can be either a property or a relation. See Theune (2000) for a version of this algorithm which makes the

$\boxed{\mathsf{MakeReferringExpression}\ (r, P, L, s)}$

tree ← nil, *contrast* ← **false**
for each member A_i of list P **do**
  V ← FindBestValue $(r, A_i, \mathsf{BasicLevelValue}(r, A_i), s)$
  contrast ← Contrastive$(r, A_i, V) \wedge$
  tree' ← UpdateTree (*tree*, V, *contrast*))
  if $(|\mathsf{Val}(L \cup \{\langle A_i, V\rangle\})| < |\mathsf{Val}(L)| \ \vee A_i = \text{type}) \wedge$ *tree'* ≠ nil
  then $L \leftarrow L \cup \{\langle A_i, V\rangle\}$
    tree ← *tree'*
    if V expresses a relation between r and r' *
    then *t2* ← MakeReferringExpression $(r', P, \{\langle A_i, V\rangle\}, s)$ *
      tree ← AddTree (*tree*, *t2*) *
    endif *
  endif
  if MostSalient (r, L, s) = **true**
  then *tree* ← AddDefDet (*tree*)
    return *tree*
  endif
return failure

FIGURE 8 Extension of the modified Incremental Algorithm which incorporates relational descriptions. Other functions as in figure 2. The main differences with the previous version of the MakeReferringExpression algorithm are marked by a *.

novelty is that the algorithm now allows for recursion: as soon as a relation R is included, the MakeReferringExpression function is called again with as parameters the relatum, the list of preferred attributes, the relation (which already provides some information about the relatum!), and the current state. This recursive call yields a description for the relatum which is later included in the main description currently being generated. To enable this recursive call of MakeReferringExpression, the variable L has been promoted to the level of parameters.

Let us discuss an example: suppose we want to generate a description for object d_1 of example domain D_2 in the initial situation s_0 (all objects are equally non-salient). We call the function MakeReferringExpression $(d_1, P, \{\}, s_0)$, where $P = \langle$ type, colour, size, spatial $\rangle$ and $\{\}$

difference between properties and relations explicit in this respect.

is the empty set (of properties of d_1 which have so far been included). As before, we iterate through P. The first property we encounter is $\langle$ type, chihuahua $\rangle$. The best value is 'dog', and including this property rules out the two doghouses. This property is realized as the N^0 in the NP tree under construction. The MostSalient condition is not true: d_2 is a dog as well. The second and third attributes ('colour' and 'size') fail to distinguish d_1 from d_2. Then the algorithm encounters the fourth element of P (spatial), and consequently considers the relation $\langle$ spatial, in $(d_1, d_3)\rangle$. This *does* rule out d_2, which is not inside something. This item is included in the tree under construction as the head of a PP. The resulting tree (I) is given in figure 9. Now we enter the recursion: the function MakeReferringExpression is called with as parameters d_3 (the relatum), the list of preferred attributes, the one property of d_3 already included (namely that it contains d_1) and the state s_0. The first element on the list of preferred attributes is 'type'. The type of d_3 is 'doghouse'. This property is automatically included. Now, MostSalient $(d_3, \{\langle$ type, doghouse $\rangle$, $\langle$ spatial, in $(d_1, d_3)\rangle\}, s_0)$ is true: d_3 is the most salient non-empty doghouse in this situation. The function AddDefDet inserts a definite determiner into the NP and this finalizes the generation of a description referring to the relatum (*the doghouse*). The resulting tree is returned and the initial call of MakeReferringExpression continues. At this point, the description generated for the relatum is added to the description currently being generated for d_1, which results in (II) in figure 9. At this point, the selected properties of d_1 are $L_{d_1} = \{\langle$ type, dog $\rangle$, $\langle$ spatial, in (d_1, d_3) $\rangle$ $\}$. MostSalient (d_1, L_{d_1}, s_0) is true. To wrap things up AddDefDet inserts a definite article into the main NP and the final tree is returned. Thus, the algorithm outputs *the dog in the doghouse*. The interesting thing about this description is that it is distinguishing, while neither *the dog* nor *the doghouse* in isolation are.

In order to show why it is useful to determine the best value for a relation, let us briefly discuss another example: the generation of a description for object d_2 in domain D_2, again in state s_0. For the first part, the algorithm proceeds as in the previous example. The property $\langle$ type, dog $\rangle$ is selected and an expression for it is incorporated in the tree for d_2. Again, colour and size do not help to distinguish d_2 from d_1, so the algorithm goes on to check the relations in which d_2 is involved and finds $\langle$ spatial, left of (d_2, d_4) $\rangle$. When trying to find the best value, it turns out that this is the basic level value *next to*. The more specific value 'left of' is not the best value since it fails to rule out more distractors than 'next to': both have the same value set. Therefore, the more general value of the two is selected, which is 'next to'. The rest of the generation procedure is similar to the previous example, resulting in the description

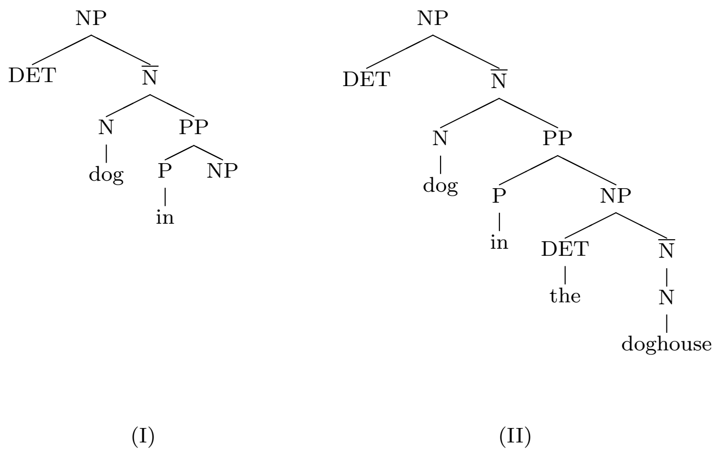


FIGURE 9 Two crucial stages in the generation of *the dog in the doghouse*.

of d_2 as *the dog next to the doghouse*. Simply expressing the 'type' value stored in the database, instead of searching for the best value, would have resulted in the description *the dog left of the doghouse*. This description is overly specific, since the information that d_2 is located to the *left* of doghouse d_4 is irrelevant in the current domain. However, as the reader may check, the addition to the domain of a dog which is located to the right of a doghouse *would* lead to the description of d_2 as *the dog left of the doghouse*.

Let us take stock. We have shown that some simple modifications to the modified Incremental Algorithm suffice for the generation of descriptions involving relations. In contrast with e.g., Horacek (1997) and Stone & Webber (1998), this algorithm is fully explicit about which properties should be tried in which order. The use of subsumption hierarchies on relations seems to offer an attractive and plausible means of obtaining some of the required flexibility. Moreover, this exercise shows how insights of Dale & Haddock (1991) can be incorporated in Dale & Reiter's Incremental Algorithm, which is more efficient than the Greedy Heuristics strategy used by Dale & Haddock. Unfortunately, by including recursion in the way we have done here the algorithm is no longer polynomial. The problem is that, as the algorithm stands, it may make the wrong choice of relatum. Interestingly, Beun and Cremers (1998) show that humans only select *salient* objects as relata. This can be incorporated in the algorithm by always choosing the most salient member of the set of potential relata (see van der Sluis & Krahmer 2001). This increases the chance of finding a solution in polynomial time.

A somewhat related problem of the extension proposed here is that it predicts that generating relations is incremental as well. Suppose that the first relation the algorithm selects fails to rule out all the remaining distractors. Then the algorithm will select a further relation and continue recursively from there. The incrementality assumption implies that the first relation will always be realised, even if adding further relations would render it redundant with hindsight. It would seem rather far-fetched to claim psychological reality for this kind of incrementality. It is unlikely that someone would describe an object as *the dog next to the tree in front of the garage* in a situation where *the dog in front of the garage* would suffice. In Krahmer *et al.* (2001), this problem of 'forced incrementality' is addressed in more detail and a general solution is offered.

1.6.3 Bridging Descriptions

Finally, we would like to point out that the two extensions of the modified Incremental Algorithm described above and displayed in figures 6 and 8 respectively can be combined in a straightforward way, and that this combination paves the way for the generation of bridging descriptions. Bridging descriptions are very complex from an interpretation perspective, because the 'bridge' which an interpreter needs to construct between antecedent and anaphor is left implicit. Things are somewhat easier from a generation perspective, provided that the 'bridge' is part of the input data. In our approach, a bridging description is just a relational description with a highly salient relatum. To illustrate this, consider a domain of discourse which contains three objects d_1, d_2 and d_3: d_1 is a man, while d_2 and d_3 are chihuahua's of the same size and colour, the former being in the possession of d_1, the latter being a stray dog. Suppose that the man has just been mentioned ('A man is walking in the park') and thus is maximally salient. Now we attempt to generate a description referring to d_2. To begin with, the type ('dog') is included. The attributes 'size' and 'colour' are not included in the description since they fail to rule out the other chihuahua. Finally, the possessive relation is encountered: the fact that d_2 is in the possession of d_1 is included as this does rule out the stray chihuahua. At this point, the algorithm enters the recursion. Since d_1 is the single most salient object in the domain, we can pronominalize the reference to d_1 and a suitable pronoun is inserted in the current tree. Normally, this would result in (a tree for) *dog of him*, but following common practice (see e.g., Geurts 1995, Krahmer & van Deemter 1998) the function UpdateTree rewrites such descriptions using a possessive pronoun as determiner, with *his dog* as the net-result. As a rule of thumb, we assume that this happens only if the relatum is animate. Thus, if a particular car c is highly salient and we want to refer to the motor of c, the resulting NP will not be *its motor* but *the motor*. It should be noted that this distinction is highly language dependent. In French, for example, it is common to refer to someone's hand as *la main* instead of *sa main*.

1.7 Concluding remarks

We have discussed a generalization of Dale & Reiter's Incremental Algorithm which extends the original algorithm in a number of respects. To begin with, we have made the notion of context sets more precise by adding salience weights. This makes it possible to generate descriptions in a fully context sensitive manner, without jeopardizing the attractive properties of the original algorithm. Additionally, the algorithm now immediately attempts to incorporate selected properties in the NP tree under construction and marks contrastive properties as such. An experimental evaluation confirmed most of the hypotheses underlying these modifications. In particular, humans prefer reduced and pronominalized anaphoric references. They do not, in general, prefer a more general phrasing if this leads to a different lexical phrasing from the antecedent. A slight modification of the proposed algorithm captures this finding.

Three related extensions of the modified algorithm have been outlined. First, a simple pronominalization decision within the algorithm was discussed. Second, we have shown that some modifications of the modified Incremental Algorithm allow for the generation of relational descriptions. Finally, the combination of these two extensions enables the generation of bridging descriptions.

The modified Incremental Algorithm has been fully implemented and integrated in a data-to-speech system called D2S (see e.g., Van Deemter & Odijk 1997, Theune 2000 and Theune *et al.* 2001). One of the side-effects of this integration is that the generated descriptions are automatically converted to enriched text, that is: they are dressed up with prosodic annotations indicating where prosodic boundaries should be placed and which words should receive a pitch accent (due to newness or contrastiveness, the latter being marked by the modified Incremental Algorithm using the [+c] feature). For example, suppose that D2S generated the following mini-discourse (where the modified Incremental Algorithm is responsible for the definite descriptions).

(5) The white poodle won the hair contest.
The gray poodle came first in the obedience contest.

In the context of the first sentence, the accentuation of the second sentence is determined as follows: the modified Incremental Algorithm assigns contrast [+c] features to *gray* and *obedience*, and consequently these words are assigned a pitch accent. Newness (and its counterpart, givenness) are not determined by the MakeReferringExpression algorithm, but these are accounted for by D2S: the phrases *poodle*, *contest* and *came first* are deaccented since they express concepts that were mentioned in the previous sentence. Determiners and prepositions are only accented when they are contrastive, so the resulting accentuation pattern is as follows (where small capital letters mark pitch accents):

(6) The GRAY poodle came first in the OBEDIENCE contest.

For more details on the implementation of the modified algorithm and the embedding in D2S we refer to Theune (2000).

In this chapter we have primarily been concerned with the context-sensitive generation of descriptions, and we have stuck as close as possible to Dale & Reiter's original Incremental Algorithm. This implies that the algorithms discussed here inherit the limitations of the Incremental Algorithm which are not related to context-issues. For example, as noted in Horacek (1995), the Incremental Algorithm does not take prominence of *properties* into account, and neither does our modified version. This issue is dealt with in Horacek (1995, 1997) by always including a prominent property of the intended referent, even if it does not rule out any distractors (an example might be including the colour when describing a pink elephant in a group of flamingoes). The work of Beun & Cremers (1998), however, indicates that humans do not always systematically include inherently salient properties. Nevertheless, it would be interesting to see whether it is possible to combine Horacek's notion of (property-)salience with the notion of (object-)salience studied here. In fact, van der Sluis and Krahmer (2001) treat salience as a three-dimensional notion, including linguistic salience, but also inherent property-salience and focus-of-attention salience. Another limitation of the Incremental Algorithm which we inherit is the treatment of relative attribute values. Dale & Reiter treat all attribute values as if they are absolute, assuming that relative attribute values such as 'big' are simply given in the domain database. Van Deemter (2000) has offered a novel treatment of such relative properties. The combination of this treatment with the proposals made here should be straightforward. Similarly, neither the Incremental Algorithm nor our extension of it has anything to say about plurals. Stone (2000) presents a treatment which allows for the generation of set descriptions. The combination and integration of such extensions with the proposals made in this chapter is facilitated by the meta-algorithmic approach of Krahmer *et al.* (2001).

One other important open question which we hope to address in future work concerns the coverage of the algorithms proposed in this chapter. It seems safe to conclude that this coverage is considerably larger than that of the Incremental Algorithm, which only generates distinguishing descriptions without taking context into account. Still, it is an open question what the coverage would be with respect to a collection of corpora. One way to answer this question might be to remove all definite NPs (probably with the exclusion of proper names) from a given corpus and ask the modified algorithm to fill the gaps. However, this could not work unless we have access to the underlying domain information: how could the algorithm otherwise decide which properties to include? In general, it is certain that data-oriented approaches to generation will become increasingly important for natural language generation. However, for many natural language generation tasks —and certainly for the generation of referring expressions— both rule-based and statistical techniques will be necessary. One of the main tasks for the future will be finding meaningful combinations of the two.

Acknowledgments

The work described in this paper was carried out at IPO, the Center for User-System Interaction at the Eindhoven University of Technology. The authors are mentioned in alphabetical order. Theune was supported by the Priority Programme Language and Speech Technology, sponsored by the Netherlands Organization for Scientific Research (NWO). The modified algorithm was implemented and integrated in D2S by John-Pierre Verhagen, who also helped carrying out the experiment described in section 1.5. A version of the relational extension of the modified algorithm was implemented by Sebastiaan van Erk and André Verleg. Many thanks to all three of them. Earlier versions of this work have been presented at ICSLP'98 (Sydney), the Semantics Colloquium (Nijmegen, May '99), the ESSLLI workshop on the Generation of Nominals (Utrecht, August '99) and the ITRI seminar (Brighton, December '99). Thanks are due to the audiences at these occasions as well as to Kees van Deemter, Janet Hitzeman, Jan Landsbergen, Jon Oberlander, Massimo Poesio, Ehud Reiter, Petr Sgall, Ielka van der Sluis, Matthew Stone, Marc Swerts, Jacques Terken and three anonymous referees for useful comments. We have greatly benefitted from Robert Dale's constructive and detailed comments on a previous version of this chapter.

References

Alshawi, H. (1987), *Memory and Context for Language Interpretation.* Cambridge University Press, Cambridge, UK.

Beun, R.-J. and A. Cremers, (1998) Object Reference in a Shared Domain of Conversation, *Pragmatics & Cognition* **6**(1/2): 121-152.

Clark, H. & D. Wilkes-Gibbs (1986), Referring as a Collaborative Process, *Cognition* **22**: 1-39.

Cruse, D. (1977), The Pragmatics of Lexical Specificity, *Journal of Linguistics* **13**: 153-164.

Dale, R. (1992), *Generating Referring Expressions: Constructing Descriptions in a Domain of Objects and Processes*, MIT Press, Cambridge, MA.

Dale, R. & N. Haddock (1991), Generating Referring Expressions Involving Relations, *Proceedings of the 5th EACL*, Berlin, 161-166.

Dale, R. & E. Reiter (1995), Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions, *Cognitive Science* **18**: 233-263.

van Deemter, K. (2000), Generating Vague Descriptions, *Proceedings of the First International Natural Language Generation Conference*, Mitzpe Ramon, Israel.

van Deemter, K. & J. Odijk (1997), Context Modelling and the Generation of Spoken Discourse, *Speech Communication* **21**(1/2): 101-121.

Geurts, B. (1995), *Presupposing*, Ph.D. dissertation, University of Osnabrück.

Gordon, P., B. Grosz & L. Gilliom (1993), Pronouns, Names, and the Centering of Attention in Discourse, *Cognitive Science* **17**: 311-347.

Grosz, B., A. Joshi & S. Weinstein (1995), Centering: A Framework for Modeling the Local Coherence of Discourse, *Computational Linguistics* **21**(2): 203-225.

Grosz, B. & C. Sidner (1986), Attention, Intentions and the Structure of Discourse, *Computational Linguistics* **12**: 175-204.

Hajičová, E. (1993), Issues of Sentence Structure and Discourse Patterns. *Theoretical and Computational Linguistics*, Vol 2, Charles University, Prague.

Horacek, H. (1995), More on Generating Referring Expressions, *Proceedings of the Fifth European Workshop on Natural Language Generation*, Leiden, The Netherlands, 43-85.

Horacek, H. (1997), An Algorithm for Generating Referential Descriptions with Flexible Interfaces, *Proceedings of the 35th ACL/EACL*, Madrid, 206-

213.

Hudson-D'Zmura, S. & M. Tanenhaus, (1998), Assigning Antecedent to Ambiguous Pronouns: The Role of Center Attention as the Default Assignment, in: *Centering in Discourse*, M. Walker, A. Joshi, and E. Prince (eds.), Oxford University Press, New York, 199-226.

Kameyama, M. (1996), Indefeasible Semantics and Defeasible Pragmatics, in: *Quantifiers, Deduction and Context*, M. Kanazawa, C. Piñon, and H. de Swart (eds.), CSLI Publications, Stanford, 111-138.

Klabbers, E., E. Krahmer, E., & M. Theune, (1998), A Generic Algorithm for Generating Spoken Monologues, *Proceedings of ICSLP'98*, Sydney, Australia, 2759-2862.

Krahmer, E. (1998), *Presupposition and Anaphora*, CSLI Publications, Stanford.

Krahmer, E. & K. van Deemter (1998), On the Interpretation of Anaphoric Noun Phrases, *Journal of Semantics* **15**(3/4): 355-392.

Krahmer, E., S. van Erk & A. Verleg (2001), A Meta-Algorithm for the Generation of Referring Expressions, *Proceedings of the 9th European Workshop for Natural Language Generation*, Toulouse.

Kruijff-Korbayová, I. & Hajičová, E. (1997), Topics and Centers: A Comparison of the Salience-based Approach and the Centering Theory, *Prague Bulletin of Mathematical Linguistics* **67**: 25-50.

Lakoff, G. (1987), *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*, The University of Chicago Press, Chicago.

Lappin, S. & H. Leass (1994), An Algorithm for Pronominal Anaphora Resolution, *Computational Linguistics* **20**(4): 535-561.

Lemon, O. & I. Pratt (1997), Spatial Logic and the Complexity of Diagrammatic Reasoning, *Machine Graphics & Vision*, **6**(1): 89-108.

Levelt, P. & S. Kelter (1982), Surface From and Memory in Question-Answering, *Cognitive Psychology* **14**: 78-106.

Levelt, W. (1989), *Speaking, from Intention to Articulation*, MIT Press, Cambridge, MA.

Lewis, D. (1979), Scorekeeping in a Language Game, *Journal of Philosophical Logic* **8**: 339-359.

Maes, A. (1991), *Nominal Anaphors and the Coherence of Discourse*, Ph.D. dissertation, Tilburg University.

Malouf, R. (2000), The Order of Prenominal Adjectives in Natural Language Generation, *Proceedings of the 38th ACL*, Hong Kong, 85-92.

McCoy, K. & M. Strube (1999), Generating Anaphoric Expressions: Pronoun or Definite Description?, *Proceedings of the Workshop on the Relation of Discourse/Dialogue Structure and Reference, held in conjunction with the 37th ACL*, Maryland, 63-71.

Oberlander, J. (1998), Do the Right Thing, *Computational Linguistics*, **24**(3): 501-507.

Passonneau, R. (1996), Using Centering to Relax Gricean Informational Constraints on Discourse Anaphoric Noun Phrases, *Language and Speech* **39**(2/3): 229-264.

Pechmann, T. (1989), Incremental Speech Production and Referential Overspecification, *Linguistics* **27**: 98-110.

Poesio, M. & R. Vieira (1998), A Corpus-Based Investigation of Definite Description Use, *Computational Linguistics* **24**(2): 183-216.

Popescu-Belis, A., I. Robba & G. Sabah (1998), Reference Resolution Beyond Coreference: a Conceptual Frame and its Application. *Proceedings of COLING-ACL'98*, Montreal, 1046-1052.

Prevost, S. (1996), An Information Structural Approach to Spoken Language Generation, *Proceedings of the 34th ACL*, Santa Cruz, 294-301.

Reiter, E. & R. Dale (1997), Building Applied Natural Language Generation Systems, *Natural Language Engineering* **3**(1): 57-87.

Rosch, E. (1978), Principles of Categorization, *Cognition and Categorization*, E. Rosch and B. Lloyd (eds.), Erlbaum, Hillsdale NJ, 27-48.

Sgall, P., E. Hajičová & J. Panevová (1986), *The Meaning of the Sentence in its Pragmatic Aspects.* Reidel, Dordrecht.

van der Sluis, I. & E. Krahmer (2001), Generating Referring Expressions in a Multimodal Context: An Empirical Approach, *Proceedings of CLIN XI*, W. Daelemans *et al.* (eds.), Rodopi, Amsterdam/Atlanta.

Stone, M. & B. Webber (1998), Textual Economy Through Close Coupling of Syntax and Semantics, *Proceedings of the Ninth International Workshop on Natural Language Generation*, Montreal, 178-187.

Theune, M. (1997), Contrastive Accent in a Data-to-Speech System, *Proceedings of the 35th ACL/EACL*, Madrid, 519-521.

Theune, M. (2000), *From Data to Speech: Language Generation in Context*, Ph.D. dissertation, Eindhoven University of Technology.

Theune, M., E. Klabbers, J.R. de Pijper, E. Krahmer & J. Odijk (2001), From Data to Speech: A General Approach, *Natural Language Engineering* **7**(1): 47-86.

Walker, M. (1998), Centering, Anaphora Resolution and Discourse Structure, in: *Centering in Discourse*, M. Walker, A. Joshi and E. Prince (eds.), Oxford University Press, New York, 401-435.

Walker, M., M. Iida & S. Cote (1994), Japanese Discourse and the Process of Centering, *Computational Linguistics* **20**(2): 193-232.

Williams, E. (1997), Blocking and Anaphora, *Linguistic Inquiry* **28**: 577-628.

Zipf, G.K. (1949), *Human Behavior and the Principle of Least Effort*, Addison-Wesley, Cambridge, MA.