Generating Definite Descriptions
Non-Incrementality, Inference, and Data

Claire Gardent, Hélène Manuélian,
Kristina Striegnitz, Marilisa Amoia

1. Introduction

The generation of referring expressions is a central task for systems that automatically generate natural language texts. Indeed, the correct use of natural language referential devices is crucial for generating successful utterances, i.e., utterances that are easily and correctly understood by the hearer, because referring expressions play an important role in linking an utterance to the previous discourse, the non-linguistic situation the utterance is produced in, and the knowledge of speaker and hearer.

One algorithm that has been particularly influential in this field is the incremental algorithm for generating referring expressions presented in (Dale and Reiter 1995).

In this paper, we both extend this basic algorithm to deal with more complex, inference based, definite descriptions and propose an alternative, non-incremental algorithm which circumvents the shortcomings arising from incrementality. Moreover, we present the results of several corpus studies on definite descriptions in French which suggest some research directions that could help both widening the range of expressions that can be generated and improving the form and content of generated definite descriptions.

We start (Section 2) with a brief presentation of Dale and Reiter's (1995) incremental algorithm. Section 3 identifies a number of problems that arise from incrementality and presents an alternative, non-incremental approach. In Section 4, we turn to the generation of bridging descriptions. Bridging descriptions are definite descriptions that refer to entities which are new to the discourse, but can be linked through world knowledge to an entity that has been mentioned before. We look at the contextual reasoning necessary for generating such definite descriptions and then integrate it into a variant of Dale and Reiter's basic algorithm. Section 5 presents results of a corpus study which examines aspects of definite descriptions related to the tasks discussed in the previous sections. The results offer ways of refining the algorithms and point out directions for further extensions of the basic algorithm.

2. Dale and Reiter's incremental algorithm

The algorithm described in (Dale and Reiter 1995) provides the basis for many of the later approaches to the generation of referring expressions (Horacek 1997; Krahmer and Theune 2001; van Deemter 2002). It can be summarized as follows. The input to this algorithm are

- the *context*: a set C of positive literals associating entities with relations of arbitrary arity such as shown in Figure 1,
- the *target entity*: the entity which needs to be referred to.

$\{rabbit(r_1),\ rabbit(r_2),\ rabbit(r_3),\ hat(h_1),\ hat(h_2),\ bathtub(b_1),\ white(r_1),\ black(r_2),\ white(r_3),\ in(r_1, h_1),\ in(r_2, h_2),\ in(r_3,b_1)\}$

Figure 1. The representation of the context for Dale and Reiter's incremental algorithm.

The task of the algorithm is to find a *distinguishing description* for the target entity, i.e., a subset L of C which uniquely identifies the target. In other words, given the context, L should give so much information about the target entity that it cannot be confused with any other entity mentioned in C. For example, if, given the context of Figure 1, r_2 is the target entity, $\{rabbit(r_2), black(r_2)\}$ would be a solution, as there are no other entities which are rabbits and black in the context. However, if r_1 is the target entity, $L=\{rabbit(r_1), white(r_1)\}$ would not be sufficient, because there is one *distractor,* i.e., one other entity which also fits the description given by L, namley entity r_3. $L=\{rabbit(r_1), white(r_1), in(r_1,h_1), hat(h_1)\}$ would be a solution in this case, because r_1 is the only entity in C, which is a white rabbit and is in a hat. Formally this can be captured as follows: L is uniquely identifying the target entity t, if $D(t,L) = \{t\}$ with $D(t,L)$ the set of objects satisfying the description or more formally:

$$D(t,L) = \{o \mid \exists \text{ substitution } s \text{ such that } s(t)=o \text{ and } s(L) \subseteq C\}.$$

The pseudo-code of the algorithm is given in Figure 2. The algorithm starts with an empty descriptions ($L=\varnothing$). It then adds literals to L until all distractors are ruled out. If a literal that involves other entities than the target entity is added to the description, then the final L has to uniquely indentify these entities as well. An ordered list is used to keep track of these targets.

```
initialize
    targets ← <t>
    L ← ∅
main loop: until no more entities are left on the target list
    while targets ≠ <> do
        o ← targets[1]
if the first entity on the target list is uniquely identified
        if D(o,L) = {o}
then take it from the list
        then
            targets ← targets[2,length(targets)]
else try to extend the description
        else
            Ps = applicable_literals(L,C)
            if Ps = ∅ then return fail.
            p = select_one(Ps)
            update targets and L with p
        endif
    endwhile
    return L
```

Figure 2. Dale and Reiter's incremental algorithm.

The main loop of the algorithm is controlled by this target list. It stops when the list is empty. Otherwise it takes the first element of the list and checks whether the current L uniquely identifies it. If so, that target is removed from the list. Otherwise, L has to be extended. If that is not possible because there are no applicable literals left in C, the algorithm fails. Applicable literals are not yet part of L, they mention the target, and there is at least one distractor to which this literal doesn't apply. If there are several applicable literals, the algorithm has to choose one. The incremental algorithm presented in (Dale and Reiter 1995) assumes a domain dependent ordering of properties which determines this choice. Other variants of the algorithm might use slightly different mechanisms.

The following table shows how the algorithm that we just described would build a description for entity r_1 given the context in Figure 1.

targets	L	distractors	notes
$\langle r_1 \rangle$	$\varnothing$	all entities	extend L
$\langle r_1 \rangle$	$\{rabbit(r_1)\}$	$\{r_1, r_2,, r_3\}$	extend L
$\langle r_1 \rangle$	$\{rabbit(r_1), white(r_1)\}$	$\{r_1, r_3\}$	extend L
$\langle h_1, r1 \rangle$	$\{rabbit(r_1), white(r_1), in(r_1,h_1)\}$	$\{h_1, b_1\}$	extend L
$\langle h_1, r1 \rangle$	$\{rabbit(r_1), white(r_1), in(r_1,h_1), hat(h_1)\}$	$\{h_1\}$	h_1 unique
$\langle r_1 \rangle$	$\{rabbit(r_1), white(r_1), in(r_1,h_1), hat(h_1)\}$	$\{r_1\}$	r_1 unique
$\langle \rangle$	$\{rabbit(r_1), white(r_1), in(r_1,h_1), hat(h_1)\}$		return L

3. A non-incremental algorithm for generating definite descriptions

Generating a definite description involves both computing a *distinguishing description* (i.e., a conjunction of possibly complex properties whose denotation equals the target set) and constructing a *definite description* verbalising this distinguishing description.

On both these levels, known algorithms are usually *incremental* in that the structure built is constructed one step at a time and without backtracking. In particular, both Dale and Reiter's algorithm for computing distinguishing descriptions and its extension by van Deemter (2002) to boolean properties and non singleton sets of individuals extend a set of properties until this set uniquely identifies the target set, i.e., the set of objects to be described. Similarly, the algorithms described in (Horacek 1997; Stone 1998) to build a definite description interleave the incremental computation of a distinguishing description with an incremental construction of the syntactic tree associated by the grammar with this description. Each property selected to better identify the target set is used to retrieve a lexical entry whose semantics is this property and this lexical entry is immediately integrated into the current description.

Thus in these doubly incremental approaches, both the content and the form of a definite description are determined greedily, i.e., locally and without backtracking.

In this part of the paper, we explore the feasibility and usefulness of developing an alternative *non-incremental* algorithm. We start (Section 3.1) by presenting a non-incremental, constraint-based algorithm for generating distinguishing descriptions and showing how it can be integrated in a surface realization algorithm.

We then show that such a non-incremental algorithm is advantageous in at least two ways: it provides linguistically and cognitively better distinguishing descriptions than the incremental algorithm (Section 3.2); and it provides top-down guidance for the surface realization algorithm which better supports the realization of definite descriptions (Section 3.3).

3.1. A constraint-based non-incremental algorithm for generating definite descriptions

As Dale and Reiter (1995) show, the problem of finding minimal distinguishing descriptions can be formulated as a minimal set cover problem and is therefore known to be NP hard (Garey and Johnson, 1979).

The alternative algorithm we propose is therefore based on the use of constraint programming (CP), a paradigm aimed at efficiently solving NP hard combinatoric problems. Instead of following a *generate-and-test* strategy which might result in an intractable search space, CP minimizes the search space by following a *propagate-and-distribute* strategy where propagation draws inferences on the basis of efficient, deterministic inference rules and distribution, i.e., case distinctions for a variable values, is performed only when necessary because no further propagation steps are possible.

The basic algorithm. Consider the definition of a distinguishing description given in (Dale and Reiter, 1995).

> Let `r` be the intended referent, then a set `L` of attribute-value pairs is a distinguishing description for r if the following two conditions hold:
> **C1:** Every attribute-value pair in L applies to `r`: that is, every element of `L` specifies an attribute value that `r` possesses.
> **C2:** For every other entitiy `c` of the context, there is at least one element `l` of `L` that does not apply to `c`: that is, there is an `l` in `L` that specifies an attribute-value that `c` does not possess. `l` is said to rule out `c`

This definition can easily be lifted to the case where the target is not a single entity but a set S of entities. In this case, every attribute-value pair in L has to apply to all $r \in S$. Figure 3 gives a translation of this definition into a constraint satisfaction problem (CSP). A description for the target set S is represented by a pair of set variables $\langle P^+, P^- \rangle$ constrained to be a subset of the set of positive properties of S P^+_S (i.e., properties that are true of all elements in S) and of the set of negative properties of S P^-_S (i.e., properties that are true of none of the elements in S), respectively. The third constraint ensures that the conjunction of properties thus built eliminates all distractors, i.e., each element of the universe which is not in S. More specifically, it states that for each distractor c there is at least one property P such that either P is true of (all elements in) S but not of c or P is false of (all elements in) S and true of c.

I: the universe;
P^+_x: the set of properties some entity x has;
$P^-_x = P \setminus P^+_x$ the set of properties some entity x does not have;
$P^+_S = \cap_{x \in S} P^+_x$: the set of properties true of all elements of S;
$P^-_S = P \setminus \cup_{x \in S} P^+_x$ the set of properties false of all elements of S;
$L = \langle P^+, P^- \rangle$ is a *basic distinguishing description for* S iff:

1. $P^+ \subseteq P^+_S$
2. $P^- \subseteq P^-_S$ and
3. $\forall c \in I \setminus S, \left| (P^+ \setminus P^+_c) \cup (P^- \cap P^+_c) \right| > 0$

Figure 3: A constraint-based approach.

The CSP thus specifies what it is to be a distinguishing description for a given target set. It can be solved using a constraint programming language such as Mozart/Oz (Programming Systems Lab, 1998) which supports set variables. Every assignment of values to variables which satisfies the constraints in Figure 3 is then a possible solution, i.e., a distinguishing description for the given target set in the given context.

Additionally, a *distribution strategy* needs to be made precise which specifies how to search for solutions. We want to ensure that smaller solutions are preferred and therefore distribute (i.e. make case distinctions) over the cardinality of the output description $|P^+ \cup P^-|$ starting with the lowest possible value. That is, first the algorithm will try to find a description $\langle P^+, P^- \rangle$ with cardinality one, then with cardinality two etc. The algorithm stops as soon as it finds a solution. In this way, the description output by the algorithm is guaranteed to always be the shortest possible description.

Extending the algorithm with disjunctive properties. To take into account disjunctive properties, the constraints used can be modified as indicated in Figure 4.

$L_S = L_{S1} \vee \ldots \vee L_{SM}$ is a distinguishing description for a set of individuals iff:

- $1 \leq M \leq |S|$
- $S = S_1 \cup \ldots \cup S_M$
- For $1 \leq i \leq M$, L_{Si} is a basic distinguishing description for S_i

Figure 4: With disjunctive properties.

That is, the algorithm looks for a tuple of sets such that the union $S_1 \cup \ldots \cup S_n$ of the tuple's elements is the target set S and such that for each set S_i in that tuple there is a basic distinguishing description L_{Si}. The resulting description is the disjunctive description $L_{S1} \vee \ldots \vee L_{Sn}$ where each L_{Si} is a conjunctive description of S_i.

As before solutions are searched for in increasing order of size (i.e., of literals occurring in the description) by distributing over the cardinality of the resulting description.

Integration with surface realization. To permit the generation of definite descriptions, the constraint-based algorithm for generating distinguishing descriptions presented above needs to be integrated with surface realization. Assuming, as is usual, that the generation process is driven by communicative goals and, in particular, by informing and describing goals, this can be done by simply updating the current goal semantics with distinguishing descriptions.

Whenever an entity must be described which is discourse old, a distinguishing description will first be computed for that entity using the above constraint solver and then added to the current goal semantics thereby driving further generation. Given some overall goal semantics, the generator then seeks to realize this goal semantics by building a phrase structure tree that (i) realizes that goal semantics, (ii) is syntactically complete and (iii) is pragmatically appropriate.

Implementation. The constraint solver and the surface realization algorithm sketched above have been implemented within the generator

INDIGEN using the concurrent constraint programming language Mozart/Oz (Programming Systems Lab, 1998) which supports set variables ranging over finite sets of integers and provides an efficient implementation of the associated constraint theory. The proof-of-concept implementation includes the constraint solver described above and its integration in a chart-based generator integrating surface realization and inference. For the examples discussed in this paper, the constraint solver returns the minimal solution in milliseconds. The integration of the constraint solver within the generator permits realizing definite NPs including negative information (*the cat that is not white*) and simple conjunctions (*The cat and the dog*).

3.2. Inadequate distinguishing descriptions

As argued in (Gardent 2002), the incremental algorithm, especially when generalized to boolean properties and sets of individuals, might yield cognitively and linguistically inadequate distinguishing descriptions.

Consider for instance a context such as pictured in Table 1 and suppose the target set is $\{x_1, x_2, x_3, x_4\}$

Table 1. The dog, the rabbit and the cat.

	White	Small	Medium	Big	Dog	Rabbit	Cat	Horse	Sheep
x_1	•	•			•				
x_2	•	•				•			
x_3				•			•		
x_4	•			•			•		
x_5	•			•					•
x_6	•		•						•
x_7			•						•
x_8			•					•	
x_9			•					•	
x_{10}			•					•	

Recall that to build a distinguishing description for a given target set, the incremental algorithm goes through the list of available properties in a given order and selects from it those properties which at each step (i) have the target set in their extension (all objects in the target set must have the selected property) and (ii) eliminate some distractor (the extension of the selected property may not be contained in the current distractor set).

The ordering of the properties is fixed in two ways. First, the disjunctive length is considered: the algorithm starts with disjunctive properties of length one, then goes on to disjunctive properties of length two etc. (cf. [vD01]). Second, properties are ordered using sortal information (cf. [DR95]). For instance, the search through the available properties for the above example could be fixed to follow the order:

$$\text{type} < \text{size} < \text{color}$$

Given these assumptions, the steps followed by the incremental algorithm to build a distinguishing description for the target set $\{x_1, x_2, x_3, x_4\}$ in the context sketched in Table 1 might be as sketched in Table 2.

First, a non-disjunctive type property is looked for whose extension contains the target set and is not contained in the current distractor set namely $\{x_5, x_6, x_7, x_8, x_9, x_{10}\}$. Two properties satisfy these criteria: $\neg$*Horse* and $\neg$*Sheep*. Suppose the property $\neg$*Horse* is selected. It is then added to the distinguishing description (which is initially empty) and the distractor set is updated to the intersection of the current distractor set with the extension of the selected property namely, $\{x_5, x_6, x_7\}$.

Table 2. Generating descriptions of sets with the incremental algorithm.

	selected property	distractor set
disjunct of length 1		
type	$\neg$ *Horse*	$\{x_5, x_6, x_7\}$
size	$\neg$ *medium*	$\{x_5\}$
color	No applicable property.	$\{x_5\}$
disjunct of length 2		
type	No applicable property.	$\{x_5\}$
size	No applicable property.	$\{x_5\}$
color	No applicable property.	$\{x_5\}$
disjunct of length 3		
type	*Dog* $\vee$ *Rabbit* $\vee$ *Cat*	$\{x_5\}$

Next the "size properties" are considered and the property $\neg$*medium* is selected reducing the distractor set to $\{x_5\}$. No color property satisfies the selection criteria hence disjunctive properties of length two are considered none of them satisfies the selection criteria. When considering disjunctive properties of length three, the disjunctive type property *Dog* $\vee$ *Rabbit* $\vee$ *Cat* is selected thus yielding an empty distractor set. At this stage generation halts yielding the distinguishing description:

$$(\neg Horse) \wedge (\neg Medium) \wedge (Dog \vee Rabbit \vee Cat)$$

That is, the incremental algorithm will in this case yield a distinguishing description which can be paraphrased as

The dogs, rabbits and cats that are not horses and that are not medium size.

when a much shorter and more natural distinguishing description would in this case be the one paraphrased as

The dogs, the rabbits and the cats.

More generally, this example illustrates three types of problems for the incremental approach:

- **Context redundant descriptions:** the description produced might be context redundant in that a property present in the description might be entailed *in the given context* by some other information present elsewhere in the description. For instance the ($\neg$ *Medium*) property is context redundant in the above description as in the given context, dogs, rabbits and cats are all either big or small hence not medium size.

- **Epistemically redundant descriptions:** the description produced might be epistemically redundant in that a property it contains follows from some other information present elsewhere in the generated description and from our general knowledge about the world. For instance the ($\neg$ *Horse*) property is epistemically redundant in the above description since we know that dogs, rabbits and cats cannot be horses.

- **Logically complex descriptions:** the description produced might be unnecessarily complex due to a high number of logical connectives. For instance, the description generated for the above example by the incremental algorithm contains two negations, two disjunctions and two conjunctions whilst a much simpler distinguishing description exists which contains only two disjunctions.

In contrast, because it produces minimal descriptions, the non-incremental algorithm for computing distinguishing descriptions presented in the previous section is not affected by any of these problems.

3.3. Inadequate definite descriptions

In case of success, the output of the incremental algorithm is a distinguishing description, which is a conjunction of possibly complex properties whose denotation equals the target set.

To further realize this distinguishing description into a definite description, a kind of "double incrementality" has sometimes been proposed (Horacek, 1997, Stone, 1998) which consists in interleaving the incremental algorithm with surface realization, i.e., with the construction of the syntactic tree associated by the grammar with the input semantic representation. Each selected property is used to retrieve a lexical entry whose semantics is this property and this lexical entry is immediately integrated into the tree which has been generated so far. In this way, it is ensured that the distinguishing description can be realized as a definite description (when the current tree cannot be updated with the selected lexical entry, another property is selected).

Thus in these doubly incremental approaches, it is not only the content but also the form which is determined greedily, i.e., locally and without backtracking. In other words, no global information is available which could help in planning the definite NP. As we shall now see, this lack of global information may result in very unnatural definite NPs.

Consider again the above example. As noted, the distinguishing description produced by the incremental algorithm could in this case be

$$(\neg\ Horse) \wedge (\neg\ Medium) \wedge (Dog \vee Rabbit \vee Cat)$$

If, as suggested by the doubly incremental algorithms, properties are realized in order of selection, the successive realization phases will roughly be as follows:

The non horses.
The non medium sized non horses.
The non medium sized non horses that are either dogs, rabbits or cats.

where, as noted above, a much more natural realization of the input distinguishing description would be:

> *The dogs, rabbits and cats that are not horses and that are not medium size*

The problem is that the order in which the incremental algorithm selects properties and the order in which properties can best be realized are governed by completely orthogonal constraints. More generally, the lack of global information concerning the semantics of the NP to be generated means that the overall structure of the NP cannot be optimised. So for instance the generator is in this case unable to recognise that the "best" definite description realizing the computed distinguishing description is a three disjuncts NP with two conjoined modifiers (rather than a one disjunct NP with three disjoined modifiers).

In contrast, a surface realizer that is guided by a goal semantics (e.g., a distinguishing description) can use this semantics to plan and optimise the structure of the generated constitutent (e.g., a definite description).

To start with, since the semantic information to be realized is given globally, realization can be dictated by the grammar on the basis of the selected lexical entries and of their syntactic combinatorics rather than on the order in which properties are selected.

Further, various optimization strategies can be devised based on the structure of the semantic input. Thus for instance, the surface realizer can detect from the form of the input delivered by the constraint solver described in the previous section (in essence a disjunctive normal form, i.e., a disjunction of conjunctions of literals) the number of conjuncts contained in the generated definite description – and recursively the number of disjuncts each conjunct should contain. Similarly, because negative and positive properties are kept separate, realization strategies can be devised to optimise the structure of the NP (if the distinguishing description contains several negative properties, for instance, antonyms can be searched for to mininimise the number of negations). Finally, aggregation techniques could be applied to the input so as to avoid the repetition of semantic material appearing several times in the input distinguishing description.

A further advantage of the non-incremental algorithm presented in Section 3.1 is that it provides a richer input for surface realization. Recall that the information output by the incremental algorithm focuses on property names rather than on property denotations. As a result, appropriate decisions regarding the singular/plural distinction cannot be made: since the cardinality of the extension of the property to be realized is unknown, it is not possible to decide whether the realized constituent should be plural (cardinality greater than one) or singular (cardinality equal to one).

In contrast, the input delivered by the non-incremental algorithm to the surface realization component is a sequence of descriptions L^{1-n} and of correspondings sets S^{1-n} each description L_i in L^{1-n} is a distinguishing description for the corresponding set S_i in S^{1-n} and the sequence of descriptions S^{1-n} is interpreted as a disjunction. This pairing of descriptions with sets, means that the correct number information can be determined: if a set has cardinality one, the corresponding NP will be singular; else the NP should be plural.

In sum, although the constraint based approach, not being incremental, does not allow for a tight interleaving of content planning and surface realization and thus does not guarantee that the planned distinguishing description can actually be realized, it provides top-down guidance for the realizer which better supports the generation of syntactically optimal definite descriptions. Note further that the reverse problem holds for doubly incremental algorithms: although they guarantee that a definite description is built whenever a distinguishing description exists, they fail to guarantee that a referring expression is generated for those cases where no distinguishing description exists. Indeed in such cases, the doubly incremental algorithm will fail whereas the non-incremental algorithm will both detect the lack of distinguishing description and propose an alternative referring expression (e.g., an indefinite NP).

4. Definite descriptions and inference

The algorithms that we have looked at so far have in common that they represent context as a set of atomic facts. The definite descriptions they generate all refer to entities explicitly mentioned in this context and only contain information explicitly given in this context. In this section, we want to look at an extension of the base algorithm that uses a richer notion of context and can therefore generate definite descriptions requiring some reasoning on the part of the hearer. We will first present the data that we want to capture and then describe the extended algorithm.

4.1. Inference based definite descriptions

The prototypical use of definite descriptions is as in the following example.

(1) *A woman came in. **The woman** was wearing a beautiful hat.*

In this example, the definite description is referring to an entity that has explicitly been mentioned before and is only using properties that have explicit been attributed to that entity. This kind of definite description is also known as *directly co-referring definite descriptions*. However, corpus studies have shown that this protoypical use of definite descriptions only accounts for about 30% of all definite descriptions that are found in natural text (Poesio and Vieira 1998). Poesio and Vieira found that additional 24% were what we call *inference based definite descriptions*. We distinguish two types: *indirectly co-referring definite descriptions* and *bridging descriptions*. The following is an example of an indirectly co-referring use.

(2) *An actress entered the stage.* ***The woman*** *was wearing a big hat.*

The definite NP *the woman* is referring to an entity that has explicitly been mentioned before but uses information that has not explicitly been attributed to that entity before. In (2) this information is a *generalization* of what is already known about the referent, and hence, could be inferred from the context if some background information of the form '*actresses are women*' was known. (3) shows examples where this is not the case.

(3) a. *John has bought a new car.* ***The Volvo*** *delights him.*
b. *I met a man yesterday.* ***The bastard*** *stole my money.*

In (3a), the descriptive content of the definite description is a *specification* of what's already known about the referent (Volvos are kinds of cars). In (3b), the definite description is giving a *re-description* of its referent; it adds information which is not in any relation with what's already known about the referent. In the following, we will concentrate in indirectly referring definite descriptions of the type shown in (2).

The second type of inference based definite descriptions, bridging descriptions, are definite descriptions that refer to an entity that has not explicitly been mentioned before. However, it is related to an entity that has been mentioned before. Here is an example of such a case.

(4) *John entered the room.* ***The ceiling*** *was very high.*

The ceiling mentioned in the second sentence is a part of the room mentioned in the first sentence. Moreover, our general knowledge about rooms tells us that every room has a ceiling. So, when hearing that there is a room, we already know that there also must be a ceiling. (5a) shows a bridging description that refers to an entity of which we don't know before that it exists. Not all rooms have windows. Therefore, the existence of the window doesn't follow from the context. However, it is quite probably in the given context, as most rooms have windows. This is not the case in (5b). The fact that there is a chandelier neither follows from the context nor is it probable in the given context. But even as it is not common for rooms nowadays to have chandeliers, it is plausible to link the chandelier to the room as rooms typically have lamps and a chandelier is a type of lamp; moreover, it is a type of lamp that usually is found in rooms.

(5) a. *John entered the room.* ***The windows*** *looked out to the bay.*
 b. *John entered the room.* ***The chandelier*** *was sparkling brightly.*

Before we go on, we have to introduce some more terminology. In the rest of this section, we will follow Dale and Reiter and call the referent of the definite description that we are going to generate the *target.* The entity in the discourse context that the target is linked to is called *anchor.* This link is established via the identity relation in the case of co-reference and via some bridging relation in the case of bridging descriptions. Clark (1977), who introduced the term *bridging*, identified various different types of bridging relation, such as the part-of relation, semantic roles of verbs, reasons, consequences. For the moment, we will use a fairly general part-of relation as our only bridging relation. But see Section 5 for results of a corpus study aiming at a better understanding of what kinds of relation can act as bridging relations.

4.2. An algorithm for generating inference based definite descriptions

We will now see how the basic algorithm of Section 2 can be extended with the necessary knowledge and reasoning to generate inference based definite descriptions. First, we will determine what kind of information sources are involved in the generation of these definite descriptions and re-specify the representation of the context accordingly. Then, we will characterize the conditions under which a definite description can be thought to be uniquely identifying when taking into account this additional knowledge and reasoning. Finally, we integrate these conditions into the incremental algorithm and illustrates the workings of the extended algorithm by going through an example.

4.2.1. The discourse context

In the incremental algorithm, the context is a list of positive literals recording information that is known to both the hearer and the speaker. It specifies entities and lists their properties To deal with bridging definites and indirectly anaphoric definites, this notion of context has to be extended in two ways. First, we need a model of the relevant world and lexical knowledge containing information of the type: *restaurants have cooks, poodles are pets.* Second, we have to be able to access knowledge that the speaker might have about the situation and which is not shared by the hearer. The definite description in (5a), for instance, refers to an entity which the hearer does not know about before hearing the sentence. For generation we have to model this fact. Our context will therefore consist of three parts now:

Discourse Model (DM): A list of positive literals modelling the previous discourse. This is essentially the context used in the incremental algorithm. Entities which are mentioned in the discourse model *discourse old* entities.
Ex.: *DM = {restaurant(r), Italian(r), …}*

World Knowledge (WKL): Background knowledge shared by the speaker and the hearer. WKL together with the DM is the *shared knowledge,*i.e. information that both the speaker and the hearer know about.
Ex.: $WKL = \{\forall x\ (restaurant(x) \rightarrow \exists y\ (cook(y) \wedge part\text{-}of(y,x))), \ldots\}$

Speaker Model (SM): Additional knowledge of the speaker.
Ex.: *DM = {cook(c), part-of(c,r), …}*

4.2.2. Intended and potential anchors

Now, that we have the enriched representation of the context, we have to exmine how it interacts with the requirements under which a definite description can be used. In other words, what does it mean now for an entity to be uniquely identified by a definite description? To answer this question, we employ the terms *intended anchors* and *potential anchors.* Intended anchors (IA) are those discourse old entities which the speaker intends to act as the anchor of the target, and potential anchors (PA) are those entities which from the hearer's point of view (i.e., taking into account only shared knowledge) could act as anchors of the target.

In the basic algorithm, the only intended anchor is always the target itself. With bridging descriptions this definition obviously has to be loosened. (It is, in fact, one of the defining characteristics of bridging descriptions that the anchor is different from the target.) We will say that *intended anchors* are all those discourse old entities o, such that either o is equal to the target, or it follows from the speaker's knowledge that o is related to the target via a bridging relation. For instance: Let c be the target. If c is mentioned in DM, then c is an intended anchor. Furthermore, if $SM \models bridge(o,c)$, then o is an intended anchor of c as well.

The *potential anchors of a target t given the description L* are given by $D(t,L)$ in the basic algorithm. Also this definition needs to be loosened now. The potential anchors for the definite description *the cook,* for example, should include not only all the cooks in the discourse model but also all restaurants. Potential anchors will therefore be all discourse old entities o that, given the shared knowledge, fulfill one of the following conditions:

1. o fits the description L of t. For instance: $L = \{cook(c)\}$, $t = c$, and $DM + WKL \models cook(o)$. This is the case if $cook(o) \in DM$.
2. o is related (via a bridging relation) to some entity which fits the description. For instance: $L = \{cook(c)\}$, $t = c$, and $DM + WKL \models \exists x\ (bridge(x,o) \wedge cook(x))$. This is the case if $restaurant(o) \in DM$ and $\forall x\ (restaurant(x) \rightarrow \exists y\ (cook(y) \wedge part\text{-}of(y,x))) \in WKL$.
3. o is related (via a bridging relation) to some entity which fits a generalization of the description. For instance: $L = \{chandelier(c)\}$, $t = c$, and there is a predicate G such that $DM + WKL \models \exists x\ (bridge(x,o) \wedge G(x))$ and $WKL \models chandelier(c) \rightarrow G(c)$. This is the case if $room(c) \in DM$, $t = c$, and $\forall x\ (room(x) \rightarrow \exists y\ (furniture(y) \wedge part\text{-}of(y,x))) \in WKL$, $\forall x\ (chandelier(x) \rightarrow.\ lamp(x)) \in WKL$, and $\forall x\ (lamp(x) \rightarrow furniture(x)) \in WKL$.

Now, that we have defined intended and potential anchors (for a more detailed and more formal definition of intended and potential anchors see (Gardent and Striegnitz 2002)), we can look at the role they play in determining whether the use of a definite description is contextually appropriate. As we have just seen, intended anchors are those entities in the discourse model that the speaker wants to link the referent of the definite description to, and potential anchors are those entities that the definite description could possibly be linked to when considering only the hearer's knowledge. A successful definite description therefore has to avoid mismatches between the set of intended anchors and the set of potential anchors. The hearer has to recognize the anchors intended by the speaker as potential anchors, i.e., $IA(t) \subseteq PA(t,L)$, and he has to be able to rule out all those entities as anchors which are not intended to be anchors by the speaker, i.e., $PA(t,L) \subseteq IA(t)$. The first condition, which we call *familiarity,* ensures that the information which the speaker uses in the description lets the hearer establish the link between target and anchor. The second condition, which we call *uniqueness of the anchor,* makes sure that the definite description contains enough information to rule out all distracting anchors.

These two conditions explain why the definite description *the cook* is not appropriate in (6a) and (6b): it violates the familiarity condition (zoos don't usually have cooks) and the uniqueness condition (which of the two restaurants is the anchor?), respectively.

(6) a. *John took Jim to the zoo. ???* ***The cook*** *was wearing a white hat.*
b. *There is an Italian restaurant and a Chinese one at the end of the street. ???* ***The cook*** *is very good.*

The two conditions do, however, not explain why *the page* in (7) is inappropriate as well. For cases of this type, we need one more condition saying that the target must be *unique wrt. the anchor*. I.e., it must be coherent with the context to assume that there is only one entity which fits the description and is related to the anchor via a bridging relation.

(7) *John took the book back to the shop.* ***The page*** *was missing.*

4.2.3. Extending the incremental algorithm

Now, we will modify the basic algorithm presented in Section 2, so that it works with the extended representation of the context and uses the notions of intended and potential anchors to decide when to terminate successfully and when to fail. The pseudo-code is given in Figure 4.

Like the basic algorithm, the extended algorithm starts with $L = \emptyset$. It then adds literals to L until the uniqueness condition is satisfied, while making sure that the familiarity condition is not violated. There are two ways in which this algorithm can fail to build a uniquely identifying description. In one case, there are no more applicable literals which could be added to L but the uniqueness condition is not yet satisfied. This is similar to the situation in which the basic algorithm fails. The other case is new: the algorithm fails if no description preserving the familiarity condition can be built. Applicable literals are computed as in the basic algorithm, but we furthermore give preference to literals that preserve the familiarty condition.

```
initialize
      targets ← <t>
      L ← ∅
main loop: until no more entities are left on the target list
      while targets ≠ <> do
      o ← targets[1]
if the description rules out intended anchors of o
          if IA(o) – PA(o,L) = ∅
then return and classify L as unfamiliar
          then
             return <unfamiliar, L>
          endif
if o is uniquely identified
          if PA(o,L) ⊆ IA(o) and o is inique wrt. IA(o)
then take it from the list
          then
             targets ← targets[2,length(targets)]
else try to extend the description
          else
             Ps = applicable_literals(L,C)
             if Ps = ∅ then return <non-uniquely-identified, L>
             p = select_one(Ps)
             update targets and L with p
          endif
      endwhile
      return <uniquely-identified, L>
```

Figure 4. The extended algorithm.

We will now go through an example to illustrate how the extended algorithm works. Assume that the task is to generate an expression referring to entity *c* given the following context:

WKL: $\forall x\ (restaurant(x) \rightarrow \exists y\ (cook(y) \wedge part\text{-}of(y,x)))$
DM: $restaurant(r_1), italien(r_1), restaurant(r_2), chinese(r_2)$
SM: $cook(c), part\text{-}of(c,r_1)$

That is, there are two restaurants in the context, an Italian one and a Chinese one, and the speaker wants to refer to the cook of the Italian one, which has not been mentioned before. The following table shows how the description and the status of the familiarity and the uniqueness condition evolve.

targets	*description*	PA	IA	
$\langle c \rangle$	$\varnothing$	all entities	$\{r_1\}$	Fam.: ✓ Uniq.: ✗
$\langle c \rangle$	$\{cook(c)\}$	$\{r_1, r_2\}$	$\{r_1\}$	Fam.: ✓ Uniq.: ✗
$\langle r_1, c \rangle$	$\{cook(c), part\_of(c, r_1)\}$	$\{r_1, r_2\}$	$\{r_1\}$	Fam.: ✓ Uniq.: ✗
$\langle r_1, c \rangle$	$\{cook(c), part\_of(c, r_1), restaurant(r_1)\}$	$\{r_1, r_2\}$	$\{r_1\}$	Fam.: ✓ Uniq.: ✗
$\langle r_1, c \rangle$	$\{cook(c), part\_of(c, r_1), restaurant(r_1), italian(r_1)\}$	$\{r_1\}$	$\{r_1\}$	Fam.: ✓ Uniq.: ✓
$\langle c \rangle$	$\{cook(c), part\_of(c, r_1), restaurant(r_1), italian(r_1)\}$	$\{r_1\}$	$\{r_1\}$	Fam.: ✓ Uniq.: ✓
$\langle\ \rangle$	$\{cook(c), part\_of(c, r_1), restaurant(r_1), italian(r_1)\}$	$\Rightarrow$ description is uniquely identifying		

5. Corpus study

To better identify ways in which generation algorithms for definite descriptions could be improved, we conducted a corpus study on French data whose aim was twofold.

The first aim was to pin down and quantify the nature and the source of the relations involved in bridging descriptions. This will lay the ground for a larger scale generation of bridging definite descritions. Once the knowledge required to generate bridging definite descriptions has been identified, ontologies can be designed and reasoners optimized on the particular types of inference queries raised by the generation of bridging descriptions so as to better support their generation.

The second aim was to investigate the distribution and modes of realization of given and new information in definite descriptions. As a preliminary investigation of the data has shown, definite descriptions are not necessarily restricted to expressing given (or inferrable) information, they sometimes also communicate new information. A better understanding of the distribution and realization of given/new information in definite descriptions will permit the extension of existing generation algorithms to such cases and thus allow a better coverage of definite description uses.

In this section, we describe our corpus study and discuss the results obtained. We start (Section 5.1) with a description of our corpus analysis of definite descriptions briefly comparing it with a similar corpus study done for English (ours is for French) by Poesio and Viera. We then (Section 5.2) present the results of our study on bridging relations. Section 5.3 concludes with an analysis of given/new information in definite descriptions.

5.1. A first categorization of definite descriptions

The first step of the corpus study aims at identifying definite descriptions and distinguishing between bridging and non bridging definite descriptions. To this end, we carried out a first annotation pass based loosely on the annotation scheme proposed for English by (Poesio and Vieira 1998) and which includes the following categories:

- **Direct coreference:** Antecedent and anaphor are coreferential and the head nouns are the same in the antecedent and in the anaphor.
- **Indirect coreference:** Antecedent and anaphor are coreferential and the head noun of the anaphoric phrase is different from the head noun of the antecedent. Indirect coreference can be realized by using lexical relations such as hyponymy, hypernymy and synonymy.
- **Bridging:**. Antecedent and anaphor do not corefer, but the anaphor is interpreted as a part of the antecedent or as an object linked to the antecedent by world knowledge (Clark, 1977).
- **Attributes and appositions:** We classified in a separated category definite noun phrases which corefer to another noun phrase in apposition, or attributes coreferring with the subject of the verb *to be*, considering that it was a particular case of coreference, expressed explicitly, and using different mechanism than classical coreference.

We applied Poesio and Viera's annotation scheme on a 65 000 words subcorpus extracted from the french PAROLE corpus[1] (Lecomte, 1997). This corpus consists of articles taken from the newspaper *Le Monde* and covers a wide range of topics (sports, culture, politics, economics and leisure). It is annotated at the morpho-syntactic level in accordance with the annotation scheme Multitag/Multext of the GRACE project (Beaumont et al, 1998). In particular, each determiner is marked as either definite, indefinite, contracted (i.e., contraction of a preposition and a determiner), partitive, demonstrative, possessive, relative, exclamative or interrogative.

Starting from this corpus, we used various tools and scripts to automate the identification and annotation of definite descriptions (for more details, see (Gardent et al., 2003)). The results of this first annotation are given in Table 3 below:

Table 3. First annotation results.

first mention	6893	78,53%
bridging	417	4,75%
direct coreference	607	6,92%
indirect coreference	819	9,33%
attributes /appositions	41	0,47%
total	8777	100%

As can be seen, the proportion of *first mention* is very high (almost 80%). In comparison (Poesio and Vieira, 1998) records a rate of 48%. There are several factors which might be responsible for this difference. First, many country names, institution acronyms and idioms in French involve a definite article (e.g., *la France*, *la Côte d'Ivoire* for country names; *le CNRS*, *la CNCL* for acronyms and *avoir la main* for idioms). Second, we classified repeated use of similar definite descriptions separated from each other by a long distance as *first mention*. Third, we found a high number of generic uses. Fourth, the corpus contains a very high percentage (19.63%) of containing inferrables (i.e., definite descriptions such as *the heat of the sun* which are in fact familiar through their explicit relation to a known entity). Note also that in her analysis of definite descriptions in Swedish, (Fraurud, 1990) found that 60.9% of the definite descriptions are first mention. Again by comparison with (Poesio and Vieira, 1998), the percentage of bridging cases is relatively low (4.7% here against 11% in (Poesio and Vieira, 1998)). This might be explained by the fact that contrary to (Poesio and Vieira, 1998), we require a strictly nominal or verbal antecedent for bridging descriptions thus excluding event or discourse deictic anaphora.

With respect to the generation task, these results indicates that only a very small portion of the data can be handled by the Dale and Reiter algorithm which in essence deals only with direct coreference cases that is, 6.92% of the cases found in our corpus. Extending the algorithm as is proposed in Section 4 to bridging descriptions increases the coverage by 4.75%. Once inference is included, however, indirect references can also be generated thus covering another 9.33% of the cases. Nonetheless, there remain 78.53% of first mention definites which cannot be generated. To properly treat these the exiting algorithms have to be extended (i) to deal with new information and (ii) to generate containing inferrables.

5.2 Bridging relations

As we saw in Section 4, it is possible to extend the Dale and Reiter algorithm to bridging descriptions provided the set of bridging relations is known. An important goal of our corpus study is therefore to identify the set of bridging relations used in real data. For the development of generation systems, it is furthermore necessary to know where the implicit bridging relation "comes from": is it a lexical relation (e.g., meronymy, hyponymy, synonymy) whose encoding is part of resources such as WordNet? Is it given by world knowledge? Or is it given by a lexicographic definition?

In this section, we present a corpus study which aims at answering these questions. Starting with a presentation of the typology of bridging relations we adopted for the annotation, we then present the results of the corpus annotation and discuss their implications for the generation of bridging descriptions.

5.2.4 A typology of bridging relations

The annotation scheme we propose classifies bridging descriptions into five types: *set membership, thematic, definitional, co-participant, non-lexical.* Table 4[2] summarises the annotation scheme. It relates each class to a semantic relation, a source for the bridging relation and, additionally, some constraints on the basic ontological types involved (E stands for the set of eventualities i.e. states and events, I for the set of individuals, Loc and T for disjoint subsets of I denoting locations and time intervals respectively). We now discuss each class in more detail showing in particular, how a specific relation is identified and how its semantics is established.

Table 4. A typology for bridging descriptions.

Class	**Sem. Reln**	**Types of Target and Anchor**	**Source**
Set membership	$\subset, \in$	$\langle I, P(I) \rangle$	Hyponymy
Thematic	Thematic roles agent, patient etc.	$\langle I, E \rangle$ or $\langle E, I \rangle$	Event Thematic Grid
Definitional			
Indiv./Attribute	As given by defn.	$\langle I, Features \rangle$	Lexicographic defn
Associate/Indiv.	As given by defn.	$\langle I, I \rangle$, $\langle I, I \rangle$	Lexicographic defn
Meronymic relns	Part of	$\langle I, I \rangle$, $\langle E, E \rangle$ $\langle I, P(I) \rangle$	Meronymy
Co-participants	As given by defns of target and anchor	$\langle I, I \rangle$, $\langle I, E \rangle$ $\langle E, E \rangle$	Lex. Defn
Non lexical			
Circumstantial	spatial or temporal	$\langle I, Loc \rangle$, $\langle I, T \rangle$	Discourse structure
WKL	As given by WKL	Anything	World knowledge

Set membership. This class covers cases such as the following where the target is either a member or a subset of a set (a group of similar individuals).

seminars/the last seminar

the CGT and the FO/the FO

The semantics of this bridging relation is set membership or subset. The anchor must be a set of individuals and the target an individual or a set of individuals.

Thematic. As illustrated by the pair *murder/murderer*, the target can be related to the anchor via a thematic relation (a murderer is the *agent* of a murder). More generally, a thematic bridge links an event to an individual via a thematic relation defined by the thematic grid of the event. As a result, the property denoted by the noun caracterising the individual must be subsumed by the conjunction of the property denoted by the verb or noun caracterising the event and the thematic relation holding between this event and the individual.

Definitional. In this case, the implicit bridging relation holding between anchor and target is given by the dictionary definition of either the target or the anchor. For instance, in the pair *convalescence/operation* a "convalescence" (the target) can be defined to be the period *following* an "operation" (the anchor) or a disease so that in this case, the bridging relation between anchor and target is one of temporal succession.

In a definitional bridge, the definition usually imposes a selectional restriction which must be satisfied by the related object (anchor or target). The property declared (in the text) to hold of the related object must thus be subsumed by the property requested to hold of the related object by the definition.

The definitional category covers several of the cases identified in the literature each of them can be differentiated from the others by various ontological and semantic criteria.

Thus the ***meronymic relations*** are relations which can be expressed using construction with *part of* has *has*. For two objects X and Y to be in a meronymic relation, it must be possible to say that X usually has Y *and* that Y usually is a part of X. The meronymic relation implies (spatial, temporal or abstract) inclusion and can only hold between entities of the same ontological types (individuals, events etc.). Following (Winston et al., 1987), we assume various types of meronymic relations, such as whole/part, whole/piece, individual/stuff, collection/member, place/area, event/subevent; for a more precise definition of each of these relations, we refer the reader to (Winston et al., 1987). Additionally, we assume an individual/functional meronymic relation (e.g., *a club/the president*) which involves a definitional bridge holding between individuals with one of the related individuals being described by his profession or function wrt. the other.

Contrary to the meronymic relations, the two other types of definitional bridging relations do not imply inclusion but a simple implication relation (a teacher implies some audience, a surface implies an object etc.). More specifically, individual/attribute pairs (e.g., *a person/the age*) involve a definitional bridge holding between individuals with one of the related individuals being a feature (i.e., something that takes a value within a finite domain) while the individual/associate (e.g., *question/answer*) pairs may involve two individual, one individual and an event or between two events.

Co-participants. There are cases where the relation holding between target and anchor is given by the definitions of both the anchor and the target. For instance the pair *trip/seat* is related by the relation "in vehicle used for" which can be reconstructed from the definition of the target ("a seat is a place reserved for sitting in a vehicle or a room") and of the anchor ("a trip is a deplacement of persons by some means of transport"). In such cases, the definitions of the target and the anchor involve two properties P_a

and P_t which stand in a subsumption relation (here, *vehicle* is subsumed by *means of transport*).

Non lexical. Finally, there are cases such as in (8) where no amount of lexical knowledge will help and where the relation holding between target and anchor is given either by discourse structure (**circumstantial**) or by our knowledge of the world and of how things work (**WKL**).

(8) a. *Grenoble/the region*

b. *fight/the dead*

5.1.1. Results and discussion

In a second annotation pass, we classified the 359 bridging definite descriptions[3] found in the PAROLE corpus according to the typology presented in section 5.1. The results are given in Table 5 (the total adds up to slightly over 100% due to arbitrary approximations when rounding up decimals).

Table 5. Bridging relations.

Class	No. of occurences	Proportion
set membership	**21**	**5.8%**
thematic	**19**	**5.3%**
definitional	**283**	**80.8%**
individual/attribute	**32**	**9.0%**
individual/associate	**64**	**17.8%**
meronymic relations	**187**	**52.0%**
whole/part	89	24.7%
whole/piece	0	0%
individual/stuff	0	0%
collection/member	22	6.1%
place/area	26	7.2%
event/subevent	16	4.4%
individual/function	34	9.4%
co-participants	**8**	**2.2%**
non lexical	**28**	**7.8%**
circumstantial	17	4.7%
WKL	11	3.0%

These first results on the nature of the bridging relation suggest the following preliminary conclusions.

First, the importance of the **meronymic** relation is confirmed[4] since 52% of the bridging descriptions involve this relation. Since, moreover, the meronymy relation is encoded in WordNet (henceforth, WN), this suggests that many cases of bridging definite descriptions could be processed using WordNet (Fellbaum, 1998). We thus did a first manual search through WordNet, checking for each bridging description involving a meronymic relation encoded in WN (i.e., whole/part, collection/member and individual/stuff) to check whether it was related by a direct or indirect (i.e., inherited through a hyponym) meronymic link to its anchor. Unfortunately, we found that only 38 of our 187 meronymic cases were present in WordNet. However, a closer look at the data shows that the subtypes (town parts, country parts, enterprise parts etc.) of meronymic relations involved in corpora are actually restricted to a relatively small number which again suggest that it should not be very difficult to extend Wordnet with the meronymic information necessary to process most of the bridging definite descriptions involving this relation; or alternatively, to develop the appropriate meronymic knowledge given a specific domain and sublanguage.

Second, and again this is important for processing purposes, the number of cases involving **non-lexical knowledge** is relatively small with 4.7% of the definite descriptions involving a circumstantial relation (i.e., non knowledge based spatial or temporal inclusion e.g., *laguna/the inhabitants*) and 3% involving world knowledge (no lexical relation can be found between anchor and target e.g., *war/survivors*, *fight/dead*). In such cases, the relation between target and anchor can be found either (in the first case) through discourse structure (the structure of discourse determines in some way the relation between predicates, arguments and modifiers) or (in the second type of cases) through some complex reasoning (a fight can result in a person being hurt; one form of being hurt is to be dead etc.).

Third, an important class of bridging that does not appear in the literature but that turned out to be quantitatively non negligeable is the class of **Associate/Indiv** (17.8%). This class covers cases where the lexicographic definition of the target implies the existence of a target related entity whose sort subsumes the sort of the anchor. The bridging relation in such cases is the relation given by the lexicographic definition (cf. Examples as *operation/convalescence athletism/national federation question/answer investigation/witness report*). For computational processing, the individual/associate class is problematic because it presupposes the availability of lexicographic definitions usable computationally.

Finally, the **thematic** class which represents roughly 6% of the found bridging descriptions, could be processed using a tool such as FrameNet (Baker et al., 1998) in which words are associated with a frame (or script) specifying the frame elements (aka thematic roles) likely to participate in the scenario evoked by that frame. A preliminary manual search shows that this is indeed the case – for 14 of the 19 thematic cases, we found a frame containing target and anchor as frame element.

In summary, it seems that for the data found in the PAROLE corpus, roughly 65% of bridging definite descriptions could be processed using either FrameNet, WordNet or some limited form of lexical reasoning. The remaining 35% requires either lexicographic definitions (17.8%), essential attribute information (9%), discourse structure information (4.7%) or deep knowledge based reasoning (3%).

5.2. Given/New information in definite descriptions

Just like it assumes that the target object is given in the current context (inferred targets are not catered for), the Dale and Reiter algorithm also assumes that the properties to be expressed by the definite descriptions are given. As the data shows however, this is not necessarily the case and a definite description may express both new and given information. Furthermore the source of the given information can vary greatly: it can be associated with the antecedent of the definite description, more generally with the linguistic context (the co-text), or it can be inferred from these and world or lexical knowledge.

In this latter part of our corpus study, we concentrate on identifying and quantifying those various possibilities. The results yield some interesting guidelines on how best to realize given and new information within definite descriptions.

Specifically, the annotation scheme distinguishes two main categories namely, Information Repeating Definite Descriptions (IRDD) and Information Adding Definite Descriptions (IADD). Each of these classes is then detailed to identify on the one hand, the source of the given information (for IRDD) and on the other hand, the means by which new information is communicated (for IADD). We now present these classes in more detail and present the results of the corpus study.

5.2.1. Information Repeating Anaphors

As already mentioned, given information in a definite description can come from various sources. In what follows, we list the sources found in our corpus and illustrate each case with an example. We kept the examples in French, translating only the antecedent anaphor pairs and other necessary parts if the text in square brackets at the end of the example.

AO: The given information contained by the definite description is given by the antecedent only.

(9) ***M. Ahmed Allam-Mi, ambassadeur du Tchad en France***, *explique que son pays n'a fait qu'exercer un droit de poursuite contre les colonnes libyennes qui venaient d'attaquer Ounianga-Kebir* (…). ***L' ambassadeur Allam-Mi*** *a declaré à l'agence* (…) [M. Ahmed Allam-Mi, ambassador of Tchad in France – The ambassador Allam-Mi]

In this example, all the information contained in the anaphor is explicitly given in the antecedent (here and in what follows, the information shared by the two NPs is in bold face).

A+C: The given information contained by the definite description is given by the antecedent and the linguistic context.

(10) *[… les assaillants …] Peu après, un officier de police arrive sur les lieux et se voit remettre par le commando* ***un ultimatum expirant à 9 heures*** .(…). ***L'ultimatum des assaillants*** *exige la libération d' »un certain nombre de prisonniers palestiniens » détenus en Israël* (…) [a policeman is given **an ultimatum** finishing at 9 o'clock **by the commando** – the ultimatum of **the attacker**]

In this example, the information contained in the anaphor head (*ultimatum*) is explicitly given by the antecedent while the modifier information (*the attackers*) is mentioned in a previous part of the text.

LR: The given information contained by the definite description is inferred from a lexical relation with the antecedent.

(11) *Ce témoin affirme en effet qu'il aurait assisté, ce soir-la, à Nivelles, à* ***l' assassinat d'un couple venu en voiture faire le plein d'essence****. Et il aurait reconnu, parmi les auteurs* ***du crime****, Jean Bultot (…)* [the murder of a couple which stopped to buy petrol – the crime]

In this example, no new information about the murder is given by the anaphoric NP. There is an hyperonymy relation between the antecedent and the anaphor (*murder/crime).*

LR+C: The given information contained by the definite description is inferred from a lexical relation with the antecedent and from the linguistic context.

(12) *Car toute facture mensuelle reglée avec retard donne lieu au calcul d'****agios*** *. (…). En moyenne*, ***la pénalité pour retard*** *est appliquée à partir du vingtième jour après l'établissement du relevé* (…) [**a late payment** evokes **agios** – the penalty for being late]

In this example, we can observe an hyperonymy between *agios* and *penalty* (*agios* are a kind of *penalty*), and we know from the context that these agios are applied when people are late.

WKL: The given information contained by the definite description is inferred by world knowledge from the antecedent and / or the context.

(13) *Une brusque montée de l'hygrometrie (…) et une absence totale de vent ont provoqué à* ***Barcelone****, dans la nuit du 4 au 5 septembre, une série d'intoxication, dont deux mortelles. Asphyxiées par les émanations de la circulation automobile et les fumées d'usines soudain bloquées au-dessus de* ***la ville****, quelques soixante personnes ont du être transportées d'urgence a l'hopital.* [Barcelona – the town]

Here, world knowledge allows to link the proper noun Barcelona with the noun town. The context does not help here, but can be used in some cases when the town is less well-known than Barcelona.

5.2.2. Information Adding Definite Descriptions

An information adding anaphor can introduce new information by various means. The classification we use for these various means is listed below together with illustrating examples.

SLR: The new information is introduced by a specifying lexical relation.

(14) *La nuit précédente,* ***des pluies torrentielles*** *s'étaient abattues sur la capitale, dont les rues etaient transformées en torrents boueux.* (…)

***La mousson**, enfin …* [violent rain – monsoon]

Here, the hyponymy relation gives new information about the violent rains mentioned in the first sentence: they are the monsoon, which is a particular kind of violent rain.

SLR+Mod: The new information comes from a specifying lexical relation and from modifiers.

(15) *Pour ces ouvrières du bassin houillier dont quelques-unes ont déjà trois heures de transport par jour, la nouvelle – pour l'instant simple rumeur – a relégué au second plan* ***les manoeuvres boursières dont leur entreprise fait l'objet depuis deux mois****. (…)* ***le raid entrepris par les Chargeurs sur le plus grand groupe textile français et ses 18 000 salariés*** *n'a declenché aucune panique.* [the stock market maneuver which their company is the object since two months – the raid began by Les Chargeurs over the biggest textile group and its 18 000 employees.]

Here, the type of *maneuver* is specified by an hyponym, *raid*, which is a kind of maneuver. Moreover, new information is given about the agent of the raid (Les Chargeurs) and about the victim of the raid (it is the biggest textile group and it employs 18 000 persons.)

Mod: The new information comes from modifiers.

(16) *Pour parachever cet itinéraire breton, on aurait pu imaginer* ***le musée de Quimper*** *nous sortant (pourquoi pas ?) Emile Bernard. (…)****Le lieu, une aile de la mairie ayant depuis 1976 fait l'objet d'amenagements sérieux****, est agréable*; [The Quimper museum – the place, a part of the mairie which has been renovated since 1976]

In this example, the head of the anaphoric noun phrase is an hypernym, (a museum is a kind of place), so does not give new information about the referent. However, the modifiers give new information.

No LR: The new information is in the whole phrase and no lexical relation is used.

(17) *La police espagnole a reussi un joli coup de filet en arrêtant, le samedi 5 septembre a Barcelone, trois membres presumés du*

> *commando le plus recherché de* ***l'ETA militaire*** *depuis la plus sanglante tuerie jamais commise par* ***les séparatistes basques*** *(…)*
> [The military ETA – the Basque separatist]

In this example, only world knowledge allows to solve the anaphor. If the fact that ETA is a separatist organization is not known, nor lexical relation, neither the context can help.

5.2.3. Annotation results

To determine the importance of the above mentioned phenomena and better understand the linguistic and extralinguistic means used to produce definite descriptions, we carried out a second corpus study on 1312 coreferential definite descriptions using the classification scheme described above. The following tables show the results of this annotation pass. We removed from these results the 114 NPs with non-nominal antecedents.

Table 6. Results for IRA according to the first annotation categories.

IRA	AO	A+C	LR	LR + C	WKL	**TOTAL**
TOTAL	**511**	**73**	**208**	**53**	**183**	**1028**
PROPORTION	**49,71**	**7,10**	**20,23**	**5,16**	**17,80**	**100,00**

Table 7. Results for IAA according to first annotation categories.

IAA	SLR	SLR mod	modifiers	No LR	**TOTAL**
TOTAL	**13**	**21**	**81**	**169**	**284**
PROPORTION	**4,58**	**7,39**	**28,52**	**59,51**	**100,00**

These results tell us a number of things about the generation of definite descriptions.

A first important result is that in the PAROLE corpus, 25% of the definite descriptions introduce new information and thus cannot be handled by the Dale and Reiter algorithm.

Second, it appears that given information is inferred using either lexical or world knowledge in 43.2% of the cases. This underlines the importance of modeling reasoning when generating definite descriptions – inference is needed not only to verify as for bridging descriptions that the bridging relation is known and thus can be ommitted but also to infer additional properties of the target from the current context and general lexical or world knowledge in the case of information repeating definite descriptions.

Third, new information in IAA definite descriptions is mostly realized by "reclassifying" nouns that is with nouns that do not bear a lexical relation with the head noun of the antecedent. This is reminiscent of demonstrative NPs, which are known (Corblin 1987) to reclassify their antecedent, and raises the question of how to include such cases in a generation algorithm for referring expressions.

6. Conclusions

We have mapped out several new directions in which research on generation algorithms for referring expressions could go and given a first indication on how these could be pursued. In conclusion, we now summarise them and give pointers for further research.

First, the incremental algorithm can be interleaved with reasoning to support the generation of both indirectly co-referring definite descriptions and bridging descriptions. We presented an algorithm based on this idea which supports the generation of these "inference based" definite descriptions.

Further work on that topic includes a large scale implementation of the knowledge involved in processing bridging and indirect definite descriptions and an evaluation of the resulting performance of the generation algorithm. An interesting outcome of such an implementation would be an evaluation of existing reasoners on this particular linguistic problem.

Second, non-incremental versions of the Dale and Reiter's algorithm should be explored both to compute distinguishing descriptions and to construct the associated definite descriptions. We proposed such an algorithm based on the use of set constraints and constraint programming and where a best solution is a minimal solution. Obviously this latter point is an approximation and best solutions are not always minimal solutions. It would be interesting to investigate and model the factors influencing the "quality" of a definite description.

Third, the distribution of given/new information in definite description needs to be taken into account. When and how is new information contained in definite descriptions? Current algorithms for generating referring expressions have little to say about this question. The corpus study presented in Section 5 gives some indications on the distribution and modes of realization of given/new information in definite descriptions but more work is needed to really understand the factors underlying the generation of natural sounding, contextually appropriate definite descriptions.

Notes

1. The PAROLE corpus was created by the CNRS research unit ATILF (Analyse et Traitement Informatique de la Langue Française) and was made available to us in the context of a collaboration between ATILF and the LORIA research unit.
2. For a detailed motivation of this annotation scheme see (Gardent et al., 2003).
3. The missing cases between this number and the 417 cases of bridging annotation is linked to errors in the first annotation, the distinction between containing inferrable and bridging descriptions being sometimes difficult to make.
4. In the literature, this relation is often taken as the canonical example of bridging relation.

References

C. F. Baker, C. J. Fillmore, and J. B. Lowe.
1998 The berkeley framenet project. In *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*.

Beaumont Catherine, Lecomte Josette and Hatout Nabil,
1998 Etiquetage morpho-syntaxique du corpus "Le Monde" pour les besoins du projet PAROLE, Technical Report, INALF, Nancy.

Clark, Herbert H.
1977 Bridging. In *Thinking: Readings in Cognitive Science.* P. N. Johnson-Laird and P. C. Wason (eds.), Cambridge University Press.

Corblin, Francis
1987 *Indéfini, Défini et Démonstratif,* Genève – Paris, Librairie DROZ.

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

Dale, Robert and Reiter, Ehud
1995 Computational Interpretation of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science* 18: 233-263.

Deemter, Kees van
2002 Generation of Referring Expressions: Boolean Extensions of the Incremental Algorithm. *Computational Linguistics* 28(1): 37-52.

C. Fellbaum, editor.
1998 *WordNet. An electronic lexical database*. MIT Press, Cambridge/Mass.

K. Fraurud.
1990 Definiteness and the processing of nps in natural dicourse. *Journal of Semantics*.

Gardent, Claire.
2002 Generating minimal definite descriptions. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, USA.

Gardent, Claire and Kristina Striegnitz

2002 Generating Bridging Definite Descriptions. Submitted. Available at http://www.coli.uni-sb.de/~kris/papers/iwcs4-book.ps.gz.

Gardent Claire, Manuélian Hélène and Kow Eric
2003 Which Bridges for Bridging Descriptions, to appear in *EACL Workshop on Linguistically Interpreted Corpora proceedings*.

W. Garey and D. Johnson.
1979 *Computers and Intractability: a Guide to the Theory of NP-Completeness*. San Francisco, W.H.Freeman.

Horacek, Helmut
1997 An Algorithm for Generating Referential Descriptions With Flexible Interfaces. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*: 206-213.

Krahmer, Emiel and Mariët Theune
2001 Efficient context-sensitive generation of referring expressions. In *Information Sharing: Givenness and Newness in Language Processing,* Kees van Deemter and Roger Kibble (eds.), 223-264. CSLI Publications.

Lecomte Josette
1997 Codage Multext - GRACE pour l'action GRACE / Multitag Technical Report, INALF, Nancy.

G.A. Miller
1995 Wordnet: a lexical database for english. *Communications of the ACM*, 38(11).

Poesio, Massimo and Renata Vieira
1998 A Corpus-based Investigation of Definite Descriptions Use. *Computational Linguistics* 24(2): 183-216.

Programming Systems Lab Saarbrüucken
1998 Oz Webpage: http: //www.ps.uni-sb.de/oz/

Stone Matthew.
1998 *Modality in Dialogue: Planning, Pragmatics and Computation*. PhD thesis, Department of Computer & Information Science, University of Pennsylvania.

M. Winston, R. Chaffin, and D. Herrmann.
1987 A taxonomy of part-whole relations. *Cognitive Science*.