Reference and the Facilitation of Search in Spatial Domains

Ivandré Paraboni[1] and Kees van Deemter[2]

[1] School of Arts, Sciences and Humanities - University of São Paulo
São Paulo, Brazil (ivandre@usp.br) +55 11 3091-8837
[2] Computing Science Department, University of Aberdeen
Aberdeen, UK (k.vdeemter@abdn.ac.uk) +44 1224 272298

Abstract. *Earlier work has suggested that, in hierarchically-ordered domains (e.g., a document divided into sections and subsections), referring expressions that are judiciously over-specified to a higher extent than is achieved by existing generation algorithms can make it considerably easier for a hearer to find the referent of the referring expression. The present paper investigates over-specification in spatial domains, which play an important role in daily life. We report an experiment whose aim is (1) to find out whether over-specification plays a similar role in spatial domains as in hierarchically ordered domains, (2) to obtain a better understanding of the reasons why over-specification can be helpful to hearers, and (3) to propose an algorithmic model of reference production that takes these findings into account. The results suggest that judicious over-specification can facilitate search in a precisely defined class of problematic conditions (but less so in other cases) even if the hearer has previous knowledge about the domain. The implications of these findings are discussed and an algorithm for the generation of referring expressions is proposed that reflects them as closely as possible.*

Key words: Natural Language Generation, Referring Expressions, Search Effort, Over-specification

1 Introduction

Existing Referring Expressions Generation (REG) algorithms (Krahmer & van Deemter, 2012, for a survey) tend to produce brief descriptions. In the simple domains that have been the subject of most empirical studies in this area, this is a sensible idea that conforms with human language production (Gupta & Stent, 2005; Jordan, 1999; Belz et. al., 2007; Gatt et. al., 2008,2009; Arnold, 2008; Van Deemter et al. 2012). For instance, in a context in which there is only one door, it's better to say 'the door' than 'the red door'; the longer description might even risk the implicature that the context conceals another, possibly overlooked, door (cf., Grice, 1975). Descriptions that contain a logically superfluous part – over-specified[3] descriptions as we shall call them – are sometimes generated, but usually only as a side effect of a generation strategy that is essentially designed to achieve brevity.

The well-known Incremental Algorithm of (Dale & Reiter, 1995) is a case in point: this algorithm builds a referring expression by first ordering all properties that may be used for referring into a sequence, known as the Preference Order, then going through the properties in the list, one by one, adding a property to the incomplete description if it improves it (i.e., if the property includes the referent while excluding some of the objects that are covered by the description). The algorithm continues adding properties to the description until the referent is the only object included in it. This procedure tends to produce short descriptions, but it can occasionally produce lengthy ones, because it does not involve any kind of backtracking. (It may happen by accident, for example, that the property last added to a sizable description *on its own* might have described the referent uniquely.) The algorithm thus involves a certain amount of over-specification, but only as a side effect.[4]

Most algorithms, in other words, do not use over-specification "strategically". Exceptions exist, where over-specification is used for communicative effect (e.g., Jordan, 2000), but even in those cases, over-specification is not performed in order to facilitate comprehension.

In earlier work, we have argued that in domains more complex than the ones usually studied in the evaluation of REG algorithms (for example, the experiment in Gatt et. al. (2007), which formed the basis for the Shared Task Evaluation Campaigns of Belz and Gatt 2007, involved domains with just 6 distractor objects and 1 or 2 target referents), strategies for judicious over-specification are essential to the reference task (Paraboni, 2003; Paraboni et. al., 2007). Brief

[3] The term "over-specification" could be seen as a misnomer because descriptions may need logically superfluous parts to be situationally appropriate. We shall stick with dominant usage in the computational literature on reference, understanding a description to be over-specified if it contains one or more properties that can be removed from the description without making the description referentially ambiguous.

[4] To be precise, the Incremental Algorithm does not only cause over-specification through its use of a fixed preference ordering, but also through its handling of nouns. The latter type of over-specification is also best seen as a side effect, however.

descriptions such as 'the red book' are not of much help, for example, if the intended referent is hidden inside a box. In such cases, an over-specified description that judiciously adds location information can facilitate search (e.g., as in 'the red book inside the box') even if the added information is not logically necessary to single out the target of the referring expression (e.g., the domain contains only one red book). Standard algorithms may cause hearers difficulties in such situations, because the information in question makes no contribution to the "discriminatory power" of the description and would not normally be included in the description.

In Paraboni et al. (2007), we discussed two kinds of problematic situation – associated with the terms Dead End and Lack of Orientation – showing that a production strategy that adds location information in problematic situations is favoured by speakers and benefits hearers.

Let us briefly explain the concepts of Dead End and Lack of Orientation. In some situations a referent can be identified using a simple description. For example, if there is only one picture labelled as '7' in the entire domain, the following description is logically sufficient:

(a) Picture 7

It may be, however, that the referent is not easily accessible, hence searching for it may be time-consuming, whereas additional information (e.g., 'in section 2') might have made things easier. In such situations, minimal references like (a) lead to *Lack of Orientation* (LO) (Paraboni et. al., 2007).

In what may be called *hierarchically*-ordered domains, certain objects may not be easily identifiable without relational (i.e., hierarchical) properties (cf., Dale & Haddock, 1991; Krahmer & Theune, 2002). For instance, if the document contains several pictures with number 7, one may use location information to identify the referent, as in

(b) Picture 7, in section 2

Descriptions of this form, however, create *Dead End* (DE) situations (Paraboni et. al., 2007) if the domain contains several sections numbered 2, while the intended picture is not in the most salient one of these (e.g., not in the second section of the current document part, but in the second section of some other part). Under these assumptions, (b) is perfectly unambiguous, yet difficult to resolve.

The problem in situations like (b) above is avoided by adding even more information, as in

(c) Picture 7, in section 2 of Part B.

In Paraboni et. al., (2007), we showed that traditional approaches to REG frequently fail to produce easily identifiable descriptions in these problematic (DE/LO) situations, necessitating over-specified descriptions (e.g., by adding location information) in order to facilitate search. Moreover, our results suggested that adding more information than what is required to prevent DE/LO

has relatively little impact on ease of search. These claims came from an experiment measuring the number of navigation steps required to identify target objects on web pages given various descriptions. In the research underlying the present paper we decided to investigate whether the same principles were also applicable to other types of domain. If this was the case, a picture would start to emerge in which existing algorithms for the generation of referring expressions describe a strategy that is communicatively flawed in all except the very simplest situations. We decided to focus on spatial domains where some objects are obscured from view. We believe this happens frequently in daily life, for example when we are finding our way in a city where we haven't been before. Instead of testing subjects' behaviour in the real world however, we chose to focus on the interactive virtual environments provided by the GIVE project (e.g., Striegnitz et. al., 2011). Let us briefly motivate this choice.

A GIVE world consists of a 3D virtual space containing doors, tables, chairs, and so on. Virtual environments, though more complex than the domains studied before, still lack some of the complexities of the real world. Yet, for our purposes, this disadvantage was outweighed by other considerations. GIVE did not only allow us to control the details of the environment precisely; it also allowed us to measure search effort in a number of different ways, by logging the time taken and the distance travelled by a subject who is searching for the referent; additionally, it was possible to get at least a rough idea of what objects are within a subject's focus of attention. Measuring a difficult psychological concept like search effort in several ways, instead of just one, appeared to us to be an important advantage.[5] Virtual environments enjoy considerable interest (Byron et. al., 2009; Koller et. al., 2010; Striegnitz et. al., 2011), and this added to our motivation for using them.

The concepts of LO and DE apply straightforwardly to spatial domains such as the GIVE world. Let us assume that the above context includes a number of buttons that are not immediately accessible from the point of view of the hearer because they are partially hidden behind the larger landmark objects (e.g., a plant). In this situation, it would be natural to refer to the accessible landmark, as in (d), thereby facilitating the hearer's search task[6]:

(d) the blue button, behind a plant

If the reference to the landmark is not logically necessary (because there is only one contextually salient blue button in the domain), then a short description as 'the blue button' may lead to Lack of Orientation.

As for DE, suppose the above domain contains at least two blue buttons, so the reference to the landmark (the plant) is necessary for disambiguation. DE

[5] An intriguing alternative would have been to measure effort through a dual task paradigm, e.g., Campana et. al., 2011.

[6] Following Dale & Haddock (1991), it seems plausible that a description like "the bowl on the table" has the same meaning as "the bowl on a table". Although in our experiments we have focused on the latter (i.e., using an indefinite article with the landmark object), we believe that similar results would have been obtained had we used the definite article.

Fig. 1. A spatial domain in GIVE (Striegnitz et. al., 2011)

leads to problems if the hearer comes across another plant before reaching the intended one. Search would be facilitated by the use of additional (i.e., logically redundant) information, for example:

(e) the blue button behind a plant, on the right

Spatial domains are different from hierarchically ordered domains in a number of ways. In spatial domains, domain objects may stand in a wide range of relations as in 'the book inside the box on top of the wardrobe'. Moreover, hierarchical domains may support search strategies that are not feasible in a spatial domain: It is possible to skip large sections of an (electronic or other) document simply by turning several pages at once. In a spatial domain, such shortcuts are typically much more difficult to achieve. These differences make it exciting to see whether the findings of Paraboni et. al., (2007) carry over.

The experiments in Paraboni et. al., (2007) offer only limited insight into the precise reasons why brief descriptions can be problematic. Essentially, these experiments allowed two possible types of explanation. The first is purely epistemic, relating to the information state of the hearer; this explanation asserts that an expression like "the blue button behind a plant" may be defective because the user does not know which plants have buttons behind them. The second explanation is essentially linguistic, relating to the felicity of the expression "the blue button behind a plant" in certain situations; this explanation predicts that the expression in question is bad even if the hearer has excellent knowledge of the domain.

In addition to its main aims, which were mentioned above, the new experiment will attempt to address this confound.

2 Related Work

In recent years a number of psycholinguistic studies have focused on the impact of over-specification on comprehension. Engelhardt et. al. (2006), for example, presented a series of experiments in which speakers over-describe almost one-third of the time, while listeners do not judge over-descriptions to be any worse than minimal descriptions. As the authors observe, these results appear to contradict the Gricean maxim of quantity (Grice, 1975). On the other hand, they also report on an eye tracking study showing that over-specification causes comprehension to take longer in some types of cases. For instance, it was found that instructions as in 'put the apple *on the towel* in the box', in situations where the reference to the towel is not necessary for disambiguation, attract unnecessary attention to the towel, and hence take longer to be interpreted. Similar findings were reported in Engelhardt et. al. (2011), who present an event-repeated potentials (ERPs) study in simple visual domains conveying objects such as squares, circles etc. Once again, over-specified descriptions (as in 'look at the red star' in a context in which there is only one star) took longer to be interpreted than minimal descriptions (as in 'look at the star'). On the other hand, Arts et. al. (2011), who use a setting in which speakers refer to buttons on the display of a piece of electronic equipment, found that when the type attribute alone would suffice for disambiguation (as in 'the button'),

lower recognition times result if speakers use spatial over-specification (as in 'the button *on the top*'), presumably because this allows them to find the referent more quickly.

One might hypothesise that over-specification tends to have a detrimental effect on interpretation *unless* the over-specification is there for a valid purpose. Although this summary is at risk of circularity ("over-specification is bad except when it's good"), it puts the spotlight on the question of what are valid reasons for over-specification. Viewed in this way, our research question is whether the facilitation of spatial search is a valid reason for over-specification and, connected with this, how spatial search works.

Evaluations of computational algorithms for the generation of referring expressions have overwhelmingly emphasised the study of human language production, focussing on what has been called the *human-likeness* of the expressions generated. Paraboni et al. (2007), however, combined a focus on human-likeness, in one experiment, with an assessment, in a second experiment, of how different referring expressions affect the search effort of a hearer. By showing that speakers prefer the same over-specified expressions that hearers benefit from, the experiments of Paraboni et. al (2007) suggested that, in the situations at hand, human speakers take the hearer's perspective seriously, designing their descriptions in such a way that obvious problems for their audience are avoided (cf. Clark and Murphy 1983 and many later publications in this area).

The present article extends this hearer-based perspective on language production, then fleshes out its implications for algorithmic models of reference production. First (1) we propose a simple model of the process whereby humans comprehend a referring expressions, which we call Nearest-First Search (NFS). NFS extends our earlier model of reference by taking spatial domains into account. Next (2), we put this comprehension model to the test, by testing some key predictions that follow from it. Finally (3), we show the implications of our finding for NLG, by presenting a generation algorithm that is directly informed by these findings. This algorithm is designed to minimise the amount of search that is required by an interpreter.

3 Experiment

We decided to put our ideas about reference in spatial domains to the test in a controlled experiment that makes use of the GIVE virtual environment (Striegnitz et. al., 2011). The experiment was primarily designed to test the following hypotheses.

h1: When no previous knowledge about the domain is available, over-specification reduces search distance and time, but more so in *Dead End* (DE) situations than in non-problematic (OK) situations.

h2: When no previous knowledge about the domain is available, over-specification reduces search distance and time, but more so in *Lack of Orientation* (LO) situations than in non-problematic (OK) situations.

h3: When a short reference strategy has been used, having previous knowledge about the domain reduces search distance and time, but more so in *Dead End* (DE) situations than in non-problematic (OK) situations.

h4: When a short reference strategy has been used, having previous knowledge about the domain reduces search distance and time, but more so in *Lack of Orientation* (LO) situations than in non-problematic (OK) situations.

To test these hypotheses we shall focus on problematic situations in spatial domains. However, whether a given situation is problematic (e.g., whether it constitutes a DE or LO) depends on the search strategy of the reader. For this reason, it will be useful to discuss search strategies.

3.1 Nearest-First Search

Given a referring expression, the search strategy followed by its recipient determines to a large extent how long it takes him or her to find the referent. Based on the Ancestral Search model of Paraboni et. al., (2007) (and the Topological Abstraction of Zender et al. (2009), a modified version of Ancestral Search) we

propose the following adaptation to spatial domains, which we call Nearest-First Search (NFS).

In the study of hierarchical domains in Paraboni et. al. (2007), search effort was determined by the hierarchical organisation of domain objects (e.g., the division of a book into chapters, sections, etc.) For a sufficiently large domain, searching the wrong part of the hierarchy was hypothesised to be costly (e.g., searching for a picture within a book chapter to discover that the intended picture is in another chapter.) Search within the intended part, however, was assumed to have negligible cost. In spatial search, where physical distance is essential, we've had to abandon this assumption, addressing both physical distance and the division of space into separate *visual contexts*, which we take to be closed-off areas that limits visibility outside it. In a typical GIVE world, a visual context is delimited by walls that create rooms or corridors. In a domain of objects in boxes, one might think of the space inside a box as different from the space outside it.

We call a *visual context* the set of all objects found within a closed-off area that limits visibility outside it. In a typical GIVE world, the objects in a visual context are delimited by walls that create rooms or corridors. In a domain of objects hidden inside a box, one might consider the existence of one visual context outside the box, and a second visual context inside it, which can only be accessed by first opening the box.

For simplicity, each visual context will be modelled so as to contain a distinct set of domain objects, and the hearer will only attend to one single visual context at a time. For instance, the domain in Figure 2 is assumed to be divided into 5 distinct visual contexts: the central area containing the landmark objects (chair $c1$ and $c2$, and plants $p1$ and $p2$) and 4 corridors containing one button each ($b1 - b4$). Each object will belong to a single visual context, and that contexts do not change with perspective (e.g., all landmarks always belong to the central area, regardless of whether they are seen from the start position or from inside a corridor).

Nearest-First Search (NFS) NFS kicks in when a referring expression attempts to identify a target object located within the physical space to a recipient who is herself located inside the same physical space.

1 The recipient will exhaustively search for the target object within the current visual context (e.g., the current room, corridor etc.) before considering any other visual context.
2 Any decision by the recipient to enter another visual context should follow the instructions contained in the referring expression (e.g., the recipient will not search behind a chair when the referring expression explicitly mentions an object behind a lamp or plant; the recipient will not search on the right when the referring expression says "on the left"). If no instruction is provided, the hearer may choose to perform search in any related context.
3 The nearest object that matches the description is taken to be the intended target. If later this turns out to be mistaken, the hearer will resume search as in NFS1. If two or more objects are equally distant from the hearer, NFS leaves the order in which they are inspected unspecified.

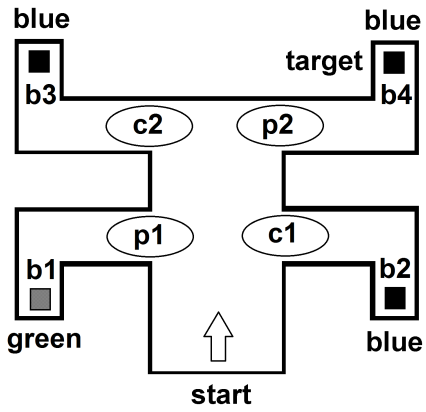

Fig. 2. Schematic view of a spatial domain containing four buttons (b1..b4) hidden behind landmark objects (plants p1,p2 and chairs c1,c2).

4 No object is inspected twice, and search stops when the goal has been reached.

Let us consider the search for 'the blue button behind a plant' in the domain in Figure 2. The local context (i.e., the central room where the hearer start position is located, and from which only the four landmark objects are visible) is inspected first. Within this area

the hearer is expected to search exhaustively (NFS1) for the referent of 'the plant', and then use this information (NFS2) to search for 'the blue button' behind the plant.

Let us assume that the hearer start position is slightly closer to the chair $c1$ than to the plant $p1$, and that $p2$ is the nearest plant to $c1$. In this case, search will end up in immediate success by following the path $c1-p2-b4$. On the other hand, if we assume that the start position is closer to $p1$ than to $c1$, and that $c2$ is the nearest chair to $p1$, then search will involve a detour to inspect the area behind $p1$ first (which is an instance of Dead End). The complete route in this case would be $p1-b1-DE-c2-p2-b4$.

Non-determinism in NFS2 and NFS3 implies that NFS does not define one search path, but a set of possible search paths from start to target. For the above example, depending on the precise distances involved, some of the non-problematic paths (i.e., paths that do not include a DE or LO situation) that lead to the target $b4$ are:

$(c1, p2, b4)$ and $(c1, c2, p2, b4)$.

The following problematic paths (in this example leading to DE in the $p1-b1$ area) are also consistent with NFS:

$(p1, b1, DE, p2, b4)$, $(p1, b1, DE, c1, p2, b4)$, $(p1, b1, DE, c2, p2, b4)$,

$(p1, b1, DE, c1, c2, p2, b4)$, $(p1, b1, DE, c2, c1, p2, b4)$, $(c1, p1, b1, DE, p2, b4)$, $(c1, p1, b1, DE, c2, p2, b4)$, $(c1, c2, p1, b1, DE, p2, b4)$

Clearly, problematic paths are longer than their non-problematic alternatives.

3.2 Experiment Design

It is time to describe the experiment that was conducted to test the hypotheses of section 3.1. We envisaged an experiment setting implemented as a GIVE (Generating Instructions in Virtual Environments) world as illustrated in Fig. 1. GIVE was originally intended as a framework for the evaluation of NLG systems, and has been the subject of three recent NLG Challenges (Byron et. al., 2009; Koller et. at. 2010; Striegnitz et. al., 2011) in which competing instruction generating systems were evaluated by a large number of on-line gamers and tested in terms of objective metrics such as task completion rates and task completion times.

GIVE consists of a 3D interactive virtual environment in the form of a treasure hunt game. In its original application, GIVE players navigate a 3D world and press buttons to manipulate doors and perform other actions required to achieve the goals of the game. The system controls all aspects of the graphical interface manipulation, and provides a plan consisting of a series of steps to be performed by the player (e.g., turn left, move forward, press the button etc.) in order to achieve the goal. This leaves NLG developers free to focus on the actual NLG task, that is, on how to convert the plan steps into natural language instructions.

Rather than using GIVE to develop a instructions generator system, we only take advantage of the existing infrastructure to evaluate REG algorithms. To this end, a small but crucial modification to the standard GIVE implementation was performed: in our experiment setting, we disregard all steps of the plan produced by the system, except those conveying references to objects that the user is required to manipulate. In our case these actions are always requests to press a particular button, that is, they must be realised as natural language instruction of the type 'Press X' in which X is a uniquely identifying description of a button object that requires a REG algorithm to be produced. In this way we reduce the otherwise game-like instruction generator to a system that simply tells the user to press a button, and then the next, with no indication on how to proceed between the two tasks. This setting will allow us to find out how readers react to different kinds of reference strategies, for instance by measuring search distance and time.

Under the assumption that participants will adopt a search strategy consistent with Nearest-First Search, our experiment setting consists of a series of rooms representing problematic (situation=DE or situation=LO) and non-problematic (situation=OK) situations of reference. Each test room consists of four possible referents (buttons) hidden inside four short corridors identified by landmark objects (chair, flower or lamp). All four landmarks are visually accessible, but two of them are closer to the initial player position (see Fig. 2

in the previous section). Buttons are not visible unless the player enters the corresponding corridor. The use of corridors was necessary to provide the required level of salience, and also to allow us to observe significant variations in search time and distance.[7] Subtler variations may only be captured using more sophisticated techniques (e.g., eye tracking).

At the entrance of each room the player is requested to press a particular button as in 'Please press the X', in which X is a referring expression conveying a variable amount of information. Each situation of reference (Situation = DE/LO/OK) will be examined according to two reference strategies. In the first case we will provide a short minimal description (Strategy = short) , and in the second case we will provide a long description (Strategy = long) that adds redundant information to facilitate search. Assuming that NFS holds, the short description strategy used in problematic situations is expected to lead to the wrong visual context, whereas the long strategy will lead to the intended target without any unnecessary detours. In non-problematic situations, both short and long strategies should lead to the correct target.

Both short and long strategies are generated by the algorithms discussed in Section 5, which are the non-hierarchical counterparts to the MD (Minimally Distinguishing) and SL (Scope-Limited) algorithms proposed in Paraboni et. al., (2007)[8].

Finally, to address the question what role the hearer's knowledge plays (cf. the end of section 1), all situations/strategies were tested under two epistemically different conditions. In the first case, the hearer does not have any previous exposure to the context of reference (Knowledge = no); in the second case, the hearer is first exposed to the entire context (i.e., seeing the location of the target and all distractor objects) and only then the hearer will be presented with the referring expression (Knowledge = yes). The cases involving previous domain knowledge are implemented as a familiarisation phase prior to the task itself, as we will explain later.

These experiment conditions (domain knowledge x situation x reference strategy) give rise to the following 12 tests, in which #8, #10 and #12 are fillers relative to our hypotheses.

Figures 3, 4 and 5 illustrate the experiment settings for the Dead End (DE), non-problematic (OK) and Lack of Orientation (LO) situations, in which the player's start position and the target button are as marked. Problematic (DE or LO) situations are expected when (under the NFS assumption) a short description leads the participant to the wrong visual context (i.e., which does not contain the intended referent). The colours of the buttons (green/blue), the land-

[7] A more natural implementation of salience (e.g., placing buttons directly behind landmarks) was not possible in GIVE since the graphical interface does not allow us to completely obscure objects from view in this way, and it does not allow the user to correctly manipulate them.

[8] Briefly, the MD algorithm avoids the inclusion of information not strictly necessary for disambiguation, whereas SL may produce an overspecified description if required to prevent DE or LO situations (Paraboni et. al., 2007).

Table 1. Experiment Conditions

#	Knowledge	Situation	Strategy
1	no	DE	short
2	no	DE	long
3	no	OK	short
4	no	OK	long
5	no	LO	short
6	no	LO	long
7	yes	DE	short
8	yes	DE	long
9	yes	OK	short
10	yes	OK	long
11	yes	LO	short
12	yes	LO	long

mark objects (plants, chairs and lamps), and the left/right direction of reference used in each room are randomly selected.

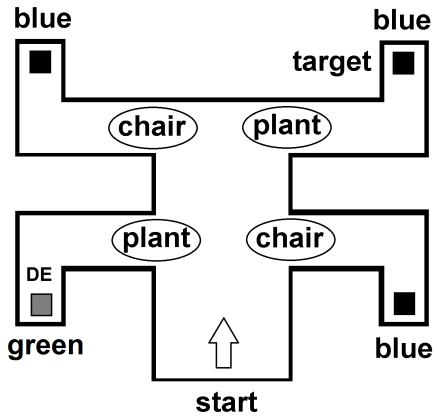

Fig. 3. Schematic view of a potential Dead End (DE) situation for tests #1 and #7 (short: 'the blue button behind a plant') and (possibly unproblematic) tests #2 and #8 (long: 'the blue button behind a plant, on the right')

3.3 Subjects

We recruited 50 students from 38 undergraduate and graduate courses with normal or corrected vision, who responded to an advert published on a university notice board. Participants were from 18 to 43 years-old (22.3 years on average)

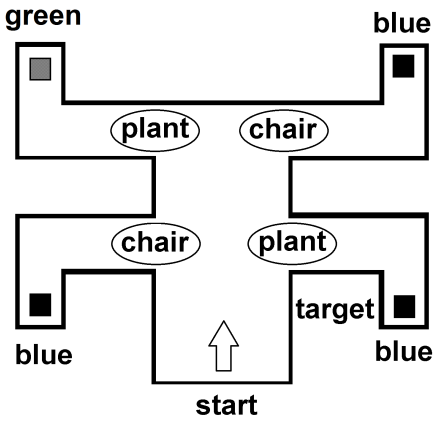

Fig. 4. Schematic view of a non-problematic (OK) situation for tests #3, #9 (short: 'the blue button behind a plant') and tests #4, #10 (long: 'the blue button behind a plant, on the right')

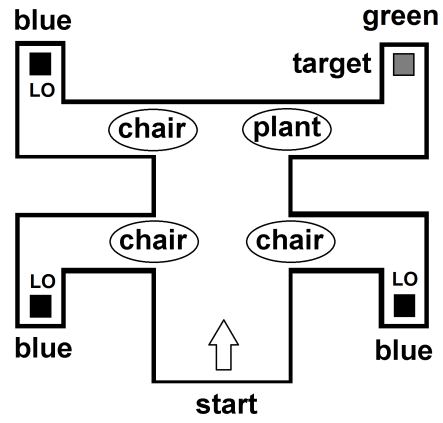

Fig. 5. Schematic view of a potential Lack of Orientation (LO) situation for tests #5, #11 (short: 'the green button') and (possibly unproblematic) tests #6, #12 (long: 'the green button behind a plant')

being 28 (56%) female and 22 (44%) male. All participants had at least average computing skills, and 22 (44%) reported usually playing video or computer games for at least one hour a week. All participants reported having at least good English skills, being 28 (56%) native speakers and further 8 (16%) near-native. Each participant received a small payment for completing the task.

3.4 Procedure

participants took part in the experiment individually, upon pre-arranged appointments over a period of eight days. Participants received instructions regarding the task to be accomplished (which basically consisted of following the instructions on screen to press the button that opens the door to the next room etc.) and were allowed to practice their navigation skills on a GIVE world similar to the actual experiment setting.

During practice, participants were requested to complete as many identification tasks (i.e., rooms) as necessary to achieve an acceptable level of navigation skills (e.g., learning how to get past doors, how to avoid obstacles, how to press buttons, etc.) This level of skill was necessary to prevent basic usability issues from obscuring the results of the identification task, particularly in the case of time measures. Participants were shown a map similar to Figure 3 and were shown that the target of each task could only be one of the four possible buttons hidden inside the corridors. Most participants required around 10 minutes for instructions and practice, and were found to meet the needs of the task after completing one or two practice rooms. A few people, however, took as long as 40 minutes of practice.

The experiment proper consisted of a GIVE game displaying the 12 test rooms in Table 1 in random order. In each room, subjects were requested to identify the referenced button in a particular Situation (DE, LO or OK) given a certain reference Strategy (short or long) and previous Domain Knowledge status (yes or no.) In LO situations, short descriptions conveyed only atomic properties, as in 'the blue/green button'; in other situations, short descriptions were of the form 'the blue/green button behind a plant/chair/lamp'. All long descriptions were of the form 'the blue/green button behind a plant/chair/lamp on the left/right'.

In conditions that involved previous domain knowledge (i.e., in conditions #7 to #12) then, before being presented with the actual referring expression, subjects were asked to examine the entire room to count the number of buttons which they contained, and record the answers by writing them on a form. This familiarisation phase was intended to model a typical situation in which the hearer is somewhat familiar with the context (e.g., visiting a room that he/she has seen before), which should possibly help him/her remember some facts about the context. However, the familiarisation phase did not explicitly require the participants to memorise any object position, and it did not give any indication of the actual referring expression to follow. After the familiarisation phase, the subject was requested to return to the room entrance and press a special-purpose (yellow) button to continue the experiment.

In order to move from one room to the next, participants were requested to press a special-purpose (red) button that opens the door ahead and simultaneously closes the door behind him/her. This helped to minimise the risk of disorientation, and guaranteed that the context set for each room was well-defined. In addition to that, the position (left/right) of the yellow and red buttons, when applicable, was selected at random. By hitting the correct button, the participant had access to the next room. By hitting a wrong button, he/she received an error message and was requested to try again. This however should only occur if a participant did not pay attention to the instructions, as the referring expressions are never ambiguous.

We measured search effort by computing the overall time and distance[9] taken to identify each target, corresponding to the two dependent variables for our hypotheses h1-h4. In order to identify unacceptable search behaviour - and hence control the quality of the results - we also kept track of the number of times the participant asked for help (the H or help key in the GIVE system), the number of clicks performed on distractor objects, and any search activity in unexpected areas of the domain (e.g., searching in a corridor behind a plant when the referring expression explicitly mentioned a chair, or searching on the right hand side when the description says left, etc.) Under normal circumstances all these variables should equal zero, and any higher value may indicate that the participant did not take the experiment seriously. We also assessed whether the participants correctly followed the instructions in each familiarisation task (for conditions #7 to #12) by checking how many objects had been inspected. Ideally, this number should always equal four (i.e., the target plus three distractor objects) as any lower number indicates non-compliance to the experiment instructions. The counts performed by the participants will further assess to what extent the familiarisation task was carried out properly.

Hypotheses h1 and h2 were tested by measuring the "gain" from using long reference strategies over short reference strategies in conditions #1 to #6, that is, when no previous knowledge about the domain is involved; gain was defined in terms of savings in search distance and/or search time. We compared the gain in non-problematic (OK) situations (i.e., #4 over #3) with the gain in problematic (DE) situations (i.e., #2 over #1) for h1, and also with the gain in problematic (LO) situations (i.e., #6 over #5) for h2. In all cases, we expected the gain in problematic situations to be larger.

Similarly, hypotheses h3 and h4 were tested by measuring the gain from having previous knowledge about the domain over not having this knowledge in conditions #1, #3, #5, #7, #9 and #11, that is, when short reference strategies are used. We compared the gain in non-problematic (OK) situations (i.e., #9 over #3) with the gain in problematic (DE) situations (i.e., #7 over #1) for h3, and also with the gain in problematic (LO) situations (i.e., #11 over #5) for h4. In all cases, we expected the gain in problematic situations to be larger.

[9] Our distance is the distance unit used in GIVE in a 2D space. Height is disregarded because both buttons and landmarks are perceived at the same level at a glance (i.e., the hearer never performs any "vertical" search).

3.5 Materials

We used a GIVE world specification as described above for each subject, conveying the 12 test rooms in random order, and an accompanying form to be filled in during each of the familiarisation tasks (i.e., preceding each of the 6 conditions in which previous domain knowledge is assumed). The experiment setting is accompanied by software designed to generate the required short/long referring expressions for each condition, to record users actions and to provide the appropriate feedback to guide the participants through the experiment.

4 Results

The complete task took 13–29 minutes per participant (19 minutes on average.) Four participants (8%) disregarded the location information provided by the descriptions (e.g., 'the blue button behind a chair') when no previous domain knowledge was available (i.e., Knowledge = no) and performed search in unnecessary areas of the domain (e.g., behind a lamp, or even exhaustively.) Since this behaviour violates the conditions of the experiment, these four participants were considered outliers, and the corresponding data were not included in the analysis that follows. Two other participants ignored the information provided by a long description (i.e., visiting unnecessary areas of the domain) in situations in which previous domain knowledge was available (i.e., Knowledge = yes). However, since these mistakes could in principle be an effect of the familiarisation task, these two participants were considered valid.

Our study was therefore based on the data provided by 46 participants. Responses regarding the counting task performed in the conditions involving familiarisation were largely correct, with only 10 (3.6%) wrong answers out of the ($6 * 46 = 276$) questions in total. All these mistakes were made by only four (8.7%) subjects. Three subjects (6.5%, different from the ones who visited unnecessary areas) failed to inspect the entire domain in the familiarisation task, missing out one of the four possible corridors. None of the subjects clicked on any object other than the intended target.

In all hypotheses (h1..h4) we used a factorial ANOVA design with repetition (Situation x Strategy for h1/h2, and Situation x Domain Knowledge for h3/h4.) Tables 2 to 5 show descriptive statistics for the search distance and time spent to complete each task in problematic (DE/LO) and non-problematic (OK) situations, for a given reference strategy (short/long) as required for h1/h2, or for a given domain knowledge status (no/yes) as required for h3/h4. Results are summarised in Table 6.

4.1 Hypotheses h1/h2: Impact of Reference Strategy on Search Distance/Time

The outcomes of the experiment are summarised in Table 6. For hypothesis h1 (Dead End, comparing the gain in test #2 over #1 with the gain in test #4

Table 2. Search distance/time for h1: Situation (DE/OK) x Strategy (short/long)

dist.	**short**		**long**			**time**	**short**		**long**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	7.44	1.61	7.20	0.91	**7.32**		12.23	6.96	11.02	4.59	**11.63**
de	15.36	3.08	8.87	0.80	**12.11**		24.32	11.52	12.69	5.34	**18.51**
	11.40		**8.03**				**18.28**		**11.86**		

Table 3. Search distance/time for h2: Situation (LO/OK) x Strategy (short/long)

dist.	**short**		**long**			**time**	**short**		**long**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	7.44	1.61	7.20	0.91	**7.32**		12.23	6.96	11.02	4.59	**11.63**
lo	24.78	6.25	9.17	0.95	**16.97**		33.94	14.02	11.29	3.83	**22.61**
	16.11		**8.18**				**23.08**		**11.16**		

Table 4. Search distance/time for h3: Situation (DE/OK) x Domain Knowledge (no/yes)

dist.	**no**		**yes**			**time**	**no**		**yes**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	7.44	1.61	9.32	4.59	**8.38**		12.23	6.96	14.24	7.26	**13.23**
de	15.36	3.08	11.83	3.95	**13.59**		24.32	11.52	16.84	8.70	**20.58**
	11.40		**10.57**				**18.28**		**15.54**		

Table 5. Search distance/time for h4: Situation (LO/OK) x Domain Knowledge (no/yes)

dist.	**no**		**yes**			**time**	**no**		**yes**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	7.44	1.61	9.32	4.59	**8.38**		12.23	6.96	14.24	7.26	**13.23**
lo	24.78	6.25	11.96	6.35	**18.37**		33.94	14.02	16.28	8.45	**25.11**
	16.11		**10.64**				**23.08**		**15.26**		

Table 6. Summary of the results for h1..h4

		distance			time		
Hypothesis	Source	F(1,45)	MSE	p	F(1,45)	MSE	p
h1 (DE)	Situation	294.35	3.59	<.0001	77.22	28.18	<.0001
h1 (DE)	Strategy	208.05	2.50	<.0001	66.64	28.46	<.0001
h1 (DE)	Situation x Strategy	116.22	3.85	<.0001	28.36	44.08	<.0001
h2 (LO)	Situation	418.63	10.24	<.0001	115.76	47.94	<.0001
h2 (LO)	Strategy	262.50	11.01	<.0001	119.41	54.82	<.0001
h2 (LO)	Situation x Strategy	243.71	11.13	<.0001	96.44	54.83	<.0001
h3 (DE)	Situation	120.46	10.37	<.0001	43.81	56.68	<.0001
h3 (DE)	Dom.Knowledge	2.42	13.04	.1268	7.45	46.26	.0090
h3 (DE)	Situation x Dom.Knowledge	29.77	11.27	<.0001	19.65	52.68	<.0001
h4 (LO)	Situation	183.78	24.98	<.0001	111.41	58.22	<.0001
h4 (LO)	Dom.Knowledge	60.05	22.93	<.0001	40.33	69.84	<.0001
h4 (LO)	Situation x Dom.Knowledge	97.14	25.53	<.0001	42.78	103.92	<.0001

over #3), we found significant effects of Situation on both search distance and time, and significant effects of Strategy on search distance and time. There was also a significant interaction effect between Situation (DE/OK) and Strategy (short/long) on both search distance and time. The use of long descriptions significantly reduces search distance and time in situations of Dead End, but not so in non-problematic (OK) situations. This confirms hypothesis h1.

For hypothesis h2 (Lack of Orientation, comparing the gain in test #6 over #5 with the gain in test #4 over #3), we found significant effects of Situation on both search distance and time, and also significant effects of Strategy on search distance and time. There was also a significant interaction effect between Situation (LO/OK) and Strategy (short/long) on both search distance and time. The use of long descriptions significantly reduces search distance and time in situations of Lack of Orientation, but not so in non-problematic (OK) situations. This confirms hypothesis h2.

4.2 Hypotheses h3/h4: The impact of Domain Knowledge

For hypothesis h3 (Dead End, comparing the gain in test #7 over #1 with the gain in test #9 over #3), we found significant effects of Situation on both search distance and time. There was no effect of Domain Knowledge on search distance, but the effect of Domain Knowledge on search time was significant. There was a significant interaction effect between Situation (DE/OK) and Domain Knowledge (no/yes) on both search distance and time. Having previous knowledge about the domain significantly reduces search distance and time in situations of Dead End, but not so in non-problematic (OK) situations (in which case the effect is opposite.) This confirms hypothesis h3.

For hypothesis h4 (Lack of Orientation, comparing the gain in test #11 over #5 with the gain in test #9 over #3) we found significant effects of Situa-

tion on both search distance and time, and also significant effects of Domain Knowledge on search distance and time. There was also a significant interaction between Situation (LO/OK) and Domain Knowledge (no/yes) on both search distance and time. Having previous knowledge about the domain significantly reduces search distance and time in situations of Lack of Orientation, but not in non-problematic (OK) situations (once again showing the opposite effect.) This confirms hypothesis h4.

4.3 The impact of over-specification when domains are familiar

The analysis of the role of over-specification in h1-2 has been restricted to the cases in which the hearer had no previous domain knowledge (Knowledge = no). We would like to gain an understanding of the effects of over-specification in situations where the hearer did have such knowledge (Knowledge = yes), that is, in tests #7 to #12. Table 7 and 8 show descriptive statistics for the distance and time taken to complete each task in problematic (DE/LO) and non-problematic (OK) situations when previous domain knowledge was available, for a given reference strategy (short/long).

Table 7. Search distance/time for Situation (DE/OK) x Strategy (short/long) with previous Domain Knowledge

dist.	**short**		**long**			**time**	**short**		**long**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	9.32	4.59	7.20	1.59	**8.26**		14.24	7.26	11.85	6.33	**13.04**
de	11.83	3.95	9.16	3.12	**10.49**		16.84	8.70	14.37	6.94	**15.61**
	10.57		**8.18**				**15.54**		**13.11**		

Table 8. Search distance/time for Situation (LO/OK) x Strategy (short/long) with previous Domain Knowledge

dist.	**short**		**long**			**time**	**short**		**long**		
	mean	sdev	mean	sdev			mean	sdev	mean	sdev	
ok	9.32	4.59	7.20	1.59	**8.26**		14.24	7.26	11.85	6.33	**13.04**
lo	11.96	6.35	9.22	1.37	**10.59**		16.28	8.45	13.09	5.52	**14.69**
	10.64		**8.21**				**15.26**		**12.47**		

For Dead Ends, we found significant effects of Situation on both search distance ($F(1,45)=33.53$, $MSE=6.87$, $p<.0001$) and time ($F(1,45)=8.99$, $MSE=33.66$, $p=.00441$), and also significant effects of Strategy on search distance ($F(1,45)=21.9$, $MSE=12$, $p<.0001$) and time ($F(1,45)=6.77$, $MSE=40.15$, $p=.012505$). There

was no interaction effect on distance ($F(1,45)=0.27$, MSE= 12.86) or time ($F(1,45)=0$, MSE=41.23). In other words, when the subject is familiar with the domain then search distance and time are reduced both if the situation is non-problematic and also if a long reference strategy is used. Crucially, however, over-specification has a beneficial effect in both problematic (DE) and non-problematic (DE) situations.

For Lack of Orientation, we found significant effects of Situation on search distance ($F(1,45)=14.29$, MSE=17.54, p=.000459) but not on search time ($F(1,45)=2.28$, MSE=54.43). We also found significant effects of Strategy on both search distance ($F(1,45)=16.38$, MSE=16.6, p=.000201) and time ($F(1,45)=11.47$, MSE =31.24, p=.001478). There was no interaction effect on distance ($F(1,45)=0.31$, MSE=14.45) or time ($F(1,45)=0.21$, MSE=34.67), that is, when the domain is familiar to the subject, search distance is reduced both if the situation is non-problematic and if a long reference strategy is used. Once again, however, over-specification has a beneficial effect in both types of situation.

4.4 Discussion

Broadly speaking, our hypotheses were confirmed. Both over-specification and domain knowledge facilitate search, but more so in problematic (DE/LO) situations than in non-problematic (OK) situations. Moreover, the two different ways in which we measured search effort have tended to give us similar results,

offering further support to the ideas underlying our hypotheses[10]. These results confirm the findings of Paraboni et. al. (2007), to the effect that complex domains require systematic over-specification over and above what standard models of reference can offer.

Our results also shed light on the question whether the benefits of judicious over-specification remain present when domain knowledge is provided. It turned out that they do: when subjects know the domain, search distance and time are still reduced by over-specification, to broadly the same extent in both problematic and OK situations. However, these benefits are now much smaller than when the domain was unknown, suggesting that the problems caused by DE and LO are of a purely epistemic nature. In other words, had we provided the hearer with complete knowledge about the domain (e.g., by providing a detailed map of the area) the benefits of over-specification would most likely be minimal.

In section 5 we discuss how the insights that were gathered from our experiment can inform a computational model of reference production. However, since our findings suggest that domain knowledge makes the benefits of over-specification less clear, and less clearly associated with DE and LO situations, only situations in which no previous domain knowledge is available will be considered.

[10] Some differences between time and distance measures were however to be expected as the hearer may stop moving around to consider how to proceed. This in some cases makes search times proportionally greater than search distance.

5 A Computational Model

We have seen that problems for readers are likely to occur when minimally distinguishing descriptions are used, as in Figures 3 and 5 in Section 1. Such descriptions are generated by most existing REG approaches (Krahmer and van Deemter 2012), including the classic algorithms discussed in (Dale & Reiter, 1995).

Our motivation behind the experiment reported in this paper was, however, not limited to disproving the effectiveness of standard REG algorithms in spatial domains: we wanted to inform a new computational model that takes the findings of the experiment into account, attempting to optimise the referring expressions generated by the algorithm, in terms of the amount of time and distance required by human recipients to find the referent of the referring expressions. Focussing on the semantic content of the expressions involved (e.g., disregarding lexical choice, see e.g. Krahmer and Van Deemter 2012, section 3.5), a computational model of this kind, which has to make referential decisions in all situations that can face it, inevitably goes beyond what's known about human language comprehension to decide what referring expression should be produced in each situation if the reader's search is to be minimised.

The present section sketches an algorithm that addresses this challenge. The main idea behind the algorithm, which is nicknamed JOVE, is to be largely agnostic about the approach to generating a distinguishing description, but to monitor whether the description that is generated is at risk of causing what we have dubbed a *problematic* situation (i.e., a situation where the reader's search can run into what we have called an obstacle); if it is at risk of this happening, the algorithm adds further information concerning the physical location of the referent (e.g., "behind the plant", "on the right", etc.).

5.1 Problematic Situations Revisited

Before outlining the algorithm itself, we should offer definitions of Lack of Orientation (LO) and Dead End (DE) that are precise enough that they could be implemented on a computer. A referring expression such as (d) in section 1 ("the blue button behind the plant") can be represented as a sequence of two references, one of which ("plant") picks out the plant and the other ("blue button") the button. Calling the button x_1 and the plant x_2, this can be represented as a sequence of two descriptions, each of which consists of a referent and a set of properties describing it. The first description is $(x_2, \{plant\})$ and the second is $(x_1, \{button, blue, behind\ x_2\})$.

So far so good, but our algorithm aims to be more general, so it has to allow descriptions that are stacked arbitrarily deeply. Therefore, sequences may contain any number of descriptions, as in $L =< (x_1, P_1), (x_2, P_2), ..., (x_n, P_n) >$ which are interpreted from right to left, from x_n to x_1. More precisely, for each $1 \leq j \leq n$, P_j is a set of properties intended to uniquely identify the entity x_j within the context in which x_j is found. x_1 is the target referent. If $n > 1$ then, for all $j < n$, the context for finding x_j is provided by x_{j+1}, and so on,

in which case P_j includes at least one relation to x_{j+1}. In our example, this is the relation "behind", which is included in the property "behind x_2". Resolution means searching for x_2 (i.e., a plant), then searching for x_1 (i.e., a blue button) within the context created by the reference to x_2.

Given a description $L =< (x_1, P_1), (x_2, P_2)...(x_n, P_n) >$, the difficult part of the problem, on which we focus here, is for the hearer to find x_n. We therefore define a *search path* to be a list of objects inspected by the hearer (typically starting with an object close to the hearer), and ending with x_n itself[11].

Given a description $L =< (x_1, P_1), (x_2, P_2)...(x_n, P_n) >$ and a search path O, we assume search to be problematic if O contains an *obstacle* y prior to x_n. An obstacle y is an object which is of the same type as x_n but which is not x_n itself, and in which either

1 (DE:) y matches the description P_n or
2 (LO:) y does not match the description P_n, and the intended referent x_n is not in the same (i.e., current) visual context as y.

Situations in which y does not match the description and x_n is present are considered non-problematic.[12] Following Paraboni et. al. (2007), we call (1) a *Dead End* (DE), and (2) a *Lack of Orientation* (LO). DE is illustrated by (d) in section 1 if the hearer comes across an object of type plant (e.g., the nearest plant) that is not the intended one. LO is illustrated by the description 'the blue button' if the hearer comes across any button in an area that does not contain the target button.

5.2 Generating Referring Expressions that Facilitate Search

The resulting algorithm is the spatial equivalent to the Minimally Distinguishing (MD) algorithm discussed in Paraboni et. al. (2007). Our experiments have shown that such approaches produce a referring expression that causes readers significant unnecessary search effort in certain situations. They suggests that search effort may be sharply reduced by using over-specified descriptions in potentially problematic situations, particularly when the domain is not known in advance. This hearer-oriented approach may be implemented as in the following Judicious Over-specification algorithm (JOVE).

The algorithm is organised as a top-level call (JOVE(r,Paths)), the core function *OS.Identify* and the test *PotentialProblem*, which calls the function *Problematic*, applying that function to each of the paths in a set of search paths. *Problematic* itself is a straightforward implementation of the notion of Dead End and Lack of Orientation for spatial domains. We define $denotes(P, x)$ to be true if all properties in P are true of the object x, and we define the *context* of an

[11] Similar problems can arise in the search for x_j, with $j < n$, but these are disregarded here for simplicity.

[12] Situations in which no suitable x_n (indeed no suitable x_j, for any $1 \leq j \leq n$) exists are bound to make search difficult. These situations are not addressed in the present work, but see the discussion at the end of this section.

object x to be the set of all domain objects that are visible within the same visual context (e.g., the same room) as x.

To make JOVE an algorithm in the strictest sense would force us to nail down every single choice, including details that have little to do with spatial location (and on which our experiment has shed little light), and this did not seem very useful. In *OS.Identify* (below), for example, we do not specify fully how an object is to be individuated within a given context (i.e., we do not fully specify what p is). In the simple domains on which our experiment has focussed, the choice of p is always easy to make, but this is not always the case[13].

In the pseudo-code below, r is the target referent, which is assumed to be located in the same domain as the reader. (Both the reader and the domain are left implicit.) *Paths* is the set of all search paths that may plausibly be chosen by the hearer, going from some point near the reader all the way up to x_n. If we had complete knowledge of the reader's search behaviour, *Paths* would contain just one path, but we allow *Paths* to be a set, in recognition of the fact that the reader's search path is not fully known. Given NFS, for example – which embodies a set of assumptions implicitly tested by our experiment – it is possible to determine which paths are permitted, and hence, which paths are in the set *Paths*. A situation is Problematic (line 10) if it involves an obstacle; more precisely, $Problematic(O, x_n, P_n)$ holds if, given the path O, the description P_n of x_n leads to DE or LO.

1 **JOVE(r,Paths):**
2 $L := P$, where the properties in P identify r within the context of r
3 OS.Identify(Paths,x_n,P_n), where (x_n,P_n) is the last element of L

4 **OS.Identify(Paths,x_n,P_n):**
5 IF NOT PotentialProblem(Paths,x_n,P_n)
6 THEN STOP ELSE
7 $P_n = P_n + p$, where p is a suitable property true of x_n
8 OS.Identify(Paths,x_n,P_n)

9 **PotentialProblem(Paths, x_n, P_n):**
10 For one or more paths O in the set of Paths, Problematic(O,x_n,P_n) holds

The algorithm starts (line 2) by producing a short (yet uniquely distinguishing) description $L =< (x_1, P_1), (x_2, P_2)...(x_n, P_n) >$ of the intended referent $r = x_1$. This may be implemented, for instance, by making use of any REG algorithm that is capable of handling relations between objects (e.g., Dale and Haddock 1991, Krahmer et. al. 2003).

Next, *OS.Identify* examines the first reference to be interpreted by the hearer (i.e., (x_n, P_n)) by calling the *PotentialProblem* function, which tests whether (x_n, P_n) may lead to DE/LO. This is done by testing whether any of the search paths in the set *Paths* contains an *obstacle* as defined in section 5.1.

[13] See for instance Kelleher & Genabith (2004) on attribute selection that takes salience into account.

If none of these search paths contain an obstacle, the referring expression is complete and JOVE stops. If an obstacle does exist then an additional property p is included in P_n (line 7) and *OS.Identify* continues recursively (line 8) until the risk of DE/LO has been averted.

Example. Consider a context involving a potential DE situation as in Figure 3 in Section 3.2 and the target object as indicated. Initially, an identifying description is produced (line 1), for example 'the blue button behind a plant'. (The description contains the plant because the colour alone does not single out this button.) The first reference that needs to be interpreted by the hearer ('a plant') is submitted to *OS.Identify* and the situation is tested for potential DE/LO (line 3). If we assume NFS, the hearer will examine the nearest plant first, the hearer may come across another object of the type 'plant' that matches the description, but which is not the intended plant (i.e., the set of *Paths* contains one or more paths that encounter the wrong plant before they encounter the intended plant).

In other words, the short description 'the blue button behind a plant' causes a *PotentialProblem* (line 9, defined on line 10) involving a DE situation. The algorithm therefore decides to amplify the description by adding direction information (p = 'on the right', line 7). This time there is no *PotentialProblem*, so the algorithm ends, producing the non-minimal description 'the blue button behind a plant, *on the right*' that facilitates search by preventing a potential DE situation.

Our experiments have focussed on cases where search is impeded by what we called an obstacle. It is, however, very likely that search can be difficult for other reasons as well, principally because the number of search paths – more precisely, the sum of all the steps in all the search paths in the set *Paths* – is too large. JOVE is easily modified to take this into account, namely by redefining *PotentialProblem* as follows:

9 **PotentialProblem(Paths, x_n, P_n):**
10 For one or more paths O in the set of Paths, Problematic(O,x_n,P_n) holds, or the total number of steps, summed over all the paths in Paths, is too large.

How to determine when the number of steps is "too large" would be for other experiments to ascertain. It seems plausible that a number of steps that is too large in one context of use, and for one reader, may be acceptable in another context and for another reader. The factors examined in the experiment of this paper, which focussed on concrete obstacles for search, appear to have strong validity regardless of context and regardless of who the reader is.

6 Final Remarks

The last decade has seen a substantial amount of work devoted to experiments with human subjects that aim to test, or otherwise inform, algorithmic models of human language production (REG algorithms, in short), particularly focussing on the production of referring expressions. As was noted in the Introduction to

the present article, most of this work has focussed on trying to mimic the referring expressions produced by human speakers (e.g., Passonneau, 1995; Jordan and Walker, 2005; Gupta and Stent, 2005; Viethen and Dale, 2006; Belz and Gatt, 2008; Krahmer et. al., 2008; Goudbeek and Krahmer, 2010; van Deemter et al., 2012), though a number of exceptions exist in which the focus is on effects on listeners (Belz and Gatt, 2008; Campana et al. 2011).

Except Paraboni et al. (2007), all these studies have focussed on domains that were small and transparent enough that human subjects could always be assumed to be aware of the existence, and the main qualities, of all the domain objects. In daily life, the natural *habitat* of natural language, this is often not the case. When we point someone the way, for instance, the set of buildings and streets that we could potentially refer to is large and the relations obtaining between them are complex; moreover we cannot assume that the listener has perfect knowledge of all these objects at the time of listening (or even during her search).

The present paper has followed up on the initial work in Paraboni et al. (2007) to investigate how these complications affect reference. In particular, we have performed an elaborate experiment in the setting of an electronic game to investigate whether a strategy of judicious over-specification might be able to make referring expressions more easily interpretable. The over-specification strategy investigated was motivated by an extension of the comprehension model proposed by Paraboni et al. (2007). This model, which is called Nearest-First Search (NFS), adapts the Ancestral Search model of Paraboni et. al. (2007) to a spatial domain. The JOVE algorithm proposed in section 5.2 is not tied to NFS, however, since it uses the set of search paths that (the speaker believes) the hearer may choose, as a *parameter* to the algorithm. This may be compared to the way in which the Incremental Algorithm uses a Preference Order of properties as a parameter (Dale & Reiter, 1995; Van Deemter et al 2012 for an exploration of the importance of the Preference Order). NFS limits this set in one particular way, but the algorithm works as well if NFS is replaced by some other set of assumptions. Having said that, the outcomes of our experiment are consistent with the assumptions embodied in NFS, therefore in practical applications that involve reference to locations, it would make sense to define the set of Paths in the algorithm in accordance with NFS.

Our studies on over-specification in hierarchical and spatial domains offer a clear illustration of the risks that are involved in generalising from extremely simple domains to the slightly more complex and naturalistic domains that we have investigated: algorithms that work well in the former do not work well in the latter. This raises the difficult question of whether the lessons that we have learned – as expressed most clearly in our generation algorithm – generalise to the even more complex and naturalistic domains of everyday life. Reference in genuinely complex situations is known to be a much more collaborative affair (see e.g. the psycholinguistic work reported in works like Brennan et al. 1996, and the computational work in Heeman and Hirst 1995), with reference tasks being broken down into smaller parts (e.g., transposed to our domain, directing

someone to a plant before telling them about the button behind it) and with contributions from both dialogue partners. Recent studies (e.g., Baker et al. 2008, Guhe 2012) show that human-human direction giving is no exception. Although we hypothesize that the basic principles underlying the need for over-specification that we have discussed in the present article will carry over to those more complex and more collaborative situations, these are matters that can only be settled conclusively by further experimentation.

References

Arnold, Jennifer E. 2008. Reference production: Production-internal and addressee-oriented processes. *Language and Cognitive Processes*, vol.23(4), pages 495-527.

Arts, A., Maes, A., Noordman, L., Jansen, C. 2011. Overspecification facilitates object identification. Journal of Pragmatics 43 (1), pages 361-374.

Baker, R., A. Gill, and J. Cassell 2008. Reactive Redundancy and Listener Comprehension in Direction-Giving. In Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue, pp. 37-45. Columbus, Ohio.

Bard, E. G., Anderson, A. H., Chen, Y., Nicholson H. B. M., Havard, C., Dalzel-Job, S. 2007. Lets you do that: Sharing the cognitive burdens of dialogue. Journal of Memory and Language 57 (2007) pages 616-641.

Belz, Anja and Gatt, A. 2007. The attribute selection for GRE challenge: Overview and evaluation results. In Proc. UCNLG+MT: Language Generation and Machine Translation.

Belz, Anja and Gatt, A. 2008. Intrinsic vs. extrinsic evaluation measures for referring expression generation. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL), Columbus, OH.

Brennan, Susan and Herbert H. Clark. 1996. Conceptual pacts and lexical choice in conversation. Journal of Experimental Psychology, 22(6):14821493. Byron, D., Koller, A., Striegnitz, K., Cassell, J., Dale, R., Moore, J., Oberlander, J. 2009. Report on the First NLG Challenge on Generating Instructions in Virtual Environments (GIVE). In: Proceedings of the 12th European Workshop on Natural Language Generation (ENLG), Athens.

Campana, E., Tanenhaus, M. K. , Allen, J. F., Remington, R. 2011. Natural discourse reference generation reduces cognitive load in spoken systems. Natural Language Engineering, 17 , pages 311-329.

Dale, Robert. 1989. Cooking up referring expressions. In: Proceedings of 27th Annual meeting of the Association for Computational Linguistics (ACL-1989), pages 68-75.

Dale, Robert and Haddock, N. 1991. Generating Referring Expressions involving Relations. EACL-1991, Berlin, pages 161-166.

Dale, Robert and Reiter, E. 1995. Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science* vol. 18., pages 233-263.

Deutsch, W. and Pechmann, T. 1982. Social interaction and the development of definite descriptions. Cognition 11, pages 159184.

Engelhardt, P. E., Bailey, K.G.D. , Ferreira, F. 2006. Do speakers and listeners observe the Gricean maxim of quantity? Journal of Memory and Language 54, pages 554-573.

Engelhardt, P. E., Demiral, S.B., Ferreira, F. 2010. Over-specified referential expressions impair comprehension: an ERP study. Brain and Cognition 77, pages 304-314.

Gatt, Albert and Belz, A. 2010. Introducing Shared Tasks to NLG: The TUNA Shared Task Evaluation Challenges. In Krahmer, E., Theune, M. (eds.) Empirical Methods in Natural Language Generation, Vol. 5980 of Lecture Notes in Computer Science, Springer, pp. 264-293.

Gatt, Albert, Belz, A., Kow, E. 2008. The TUNA Challenge 2008: Overview and Evaluation Results. In Proceedings of the Fifth International Natural Language Generation Conference (INLG-2008), pages 198-206.

Gatt, Albert, Belz, A., Kow, E. 2009. The TUNA-REG Challenge 2009: Overview and Evaluation Results. Proceedings of the 12th European Workshop on Natural Language Generation, pages 174-182.

Gatt, Albert, van der Sluis, I., van Deemter, K. 2007. Evaluating algorithms for the generation of referring expressions using a balanced corpus. In Proc. of the 11th European Workshop on Natural Language Generation (ENLG-2007), pages 49-56.

Goudbeek, Martijn and Krahmer, E. 2010. Preferences versus adaptation during referring expression generation. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL), pages 55-59, Uppsala, Sweden.

Grice, H. P. 1975. Logic and Conversation. In P. Cole and J. Morgan (Eds.), "Syntax and Semantics: Vol 3, Speech Acts" pages 43-58, Academic Press.

Guhe, M. 2012. Utility-Based Generation of Referring Expressions. Topics in Cognitive Science 4 (2), pp 306-329.

Gupta, Surabhi and Stent, A. 2005. Generation of Referring Expressions in Dialog using Corpora. In proceedings of Corpus Linguistics, UCNLG, UK.

Heeman, Peter A. and Graeme Hirst (1995) Collaborating on referring expressions. Computational Linguistics, 21(3):351-382.

Jordan, Pamela. 1999. An Empirical Study of the Communicative Goals Impacting Nominal Expressions. In the Proceedings of the ESSLLI workshop on The Generation of Nominal Expressions, Utrecht.

Jordan, Pamela. 2000. Can Nominal Expressions Achieve Multiple Goals?: An Empirical Study. In the Proceedings of ACL2000, Hong Kong.

Jordan, Pamela W. and Walker, M. (2005) Learning content selection rules for generating object descriptions in dialogue. Journal of Artificial Intelligence Research, 24:157-194.

Kelleher, John and Josef van Genabith (2004) Exploiting Visual Salience for the Generation of Referring Expressions. 17th International Florida Artificial Intelligence Research Society Conference (FLAIRS04), 17 - 19 May, Miami Beach, Florida. Koller, A., Striegnitz, K., Gargett, A., Byron, D., Cassell, J., Dale, R., Moore, J., Oberlander, J. 2010. Report on the Second NLG Challenge on Generating Instructions in Virtual Environments (GIVE-2). In: Proceedings of the 6th International Natural Language Generation Conference (INLG)

Krahmer, Emiel and Theune, Mariët. 2002. Efficient context-sensitive generation of referring expressions. In van Deemter and Kibble (eds.) *Information Sharing.* CSLI Publ., Stanford, pages 223-264.

Krahmer, Emiel, van Erk, S., Verleg, A. 2003. Graph-Based Generation of Referring Expressions. In: Computational Linguistics vol. 29(1), pages 53-72.

Krahmer, Emiel, Theune, M., Viethen, J., Hendrickx, I. 2008. Graph: The costs of redundancy in referring expressions. In Proceedings of the International Conference on Natural Language Generation (INLG), pages 227-229, Salt Fork, Ohio.

Krahmer, E. and van Deemter, K. 2012. Computational Generation of Referring Expressions: A Survey. *Computational Linguistics* vol. 38(1), pages 173-218.

Paraboni, I. Generating references in hierarchical domains: the case of Document Deixis. University of Brighton, PhD thesis (2003).

Paraboni, I., van Deemter, K., Masthoff, J. 2007. Generating Referring Expressions: Making Referents Easy to Identify. In: Computational Linguistics vol. 33(2), pages 229-254.

Passonneau, R. J. 1995. Integrating Gricean and attentional constraints. In Proc. of 14th Int. Joint Conference on Artificial Intelligence (IJCAI-95).

Striegnitz, K., Denis, A., Gargett, A., Garoufi, K., Koller, A., Theune, M. 2011. Report on the Second Second Challenge on Generating Instructions in Virtual Environments (GIVE-2.5). In: Proceedings of the 13th European Workshop on Natural Language Generation (ENLG), pages 270-279.

van Deemter, K., Gatt, A., van der Sluis, I., Power, R. 2012. Generation of Referring Expressions: Assessing the Incremental Algorithm. Cognitive Science **36** (5).

Viethen, J. and Dale, R. 2006. Algorithms for generating referring expressions: Do they do what people do? In Proc. of 4th Int. Conference on Natural Language Generation, INLG-06.

Zender, H., Kruijff, G. M., Kruijff-Korbayova, I. 2009. Situated Resolution and Generation of Spatial Referring Expressions for Robotic Assistants. In: Proceedings of the 21st international jont conference on Artifical intelligence (IJCAI)