A New Model for Generating Multimodal Referring Expressions

Krahmer, E.; van der Sluis, Ielka

Published in:
Proceedings of the 9th European Workshop on Natural Language Generation (ENLG'03)

A New Model for Generating Multimodal Referring Expressions

Emiel Krahmer
Communication and Cognition
Tilburg University
E.J.Krahmer@uvt.nl

Ielka van der Sluis
Computational Linguistics and AI
Tilburg University
I.F.vdrSluis@uvt.nl

Abstract

We present a new algorithm for the generation of multimodal referring expressions (combining language and deictic gestures).[1] The approach differs from earlier work in that we allow for various gradations of preciseness in pointing, ranging from unambiguous to vague pointing gestures. The model predicts that linguistic properties realized in the generated expression are co-dependent on the kind of pointing gesture included. The decision to point is based on a trade-off between the costs of pointing and the costs of linguistic properties, where both kinds of costs are computed in empirically motivated ways. The model has been implemented using a graph-based generation algorithm.

1 Introduction

The generation of referring expressions is a central task in Natural Language Generation (NLG), and various useful algorithms which automatically produce referring expressions have been developed (recent examples are van Deemter 2002, Gardent 2002 and Krahmer et al. 2003). A typical algorithm takes as input a single object v (**the target object**) and a set of objects (**the distractors**) from which the target object needs to be distinguished (borrowing terminology from Dale and Reiter 1995). The task of the algorithm is to determine which set of properties is needed to single out the target object from the distractors. This is known as the **content determination** problem for referring expressions. On the basis of this set of properties a **distinguishing description** in natural language can be generated; a description which applies to v but not to any of the distractors.

We describe a new algorithm which aims at producing **multimodal** referring expressions: natural language referring expressions which may include deictic pointing gestures. There are at least two motivations for such an extension. First, in various situations a purely linguistic description may simply be too complex, e.g., because the domain contains many highly similar objects. In those cases, including a deictic pointing gesture may be the most efficient way to single out the intended referent. Second, if we look at human communication it soon becomes apparent that referring expressions which include pointing gestures are rather common (Beun and Cremers 1998). Various algorithms for the generation of multimodal referring expressions have been proposed (e.g., Cohen 1984, Claassen 1992, Huls et al. 1995, André and Rist 1996, Lester et al. 1999, van der Sluis and Krahmer 2001).[2] Most of these are based

[1]This paper greatly benefitted from discussions with Mariët Theune and Kees van Deemter. Thanks are also due to Sebastiaan van Erk, Fons Maes, Paul Piwek and André Verleg. Krahmer's work was done within the context of the TUNA project, funded by Engineering and Physical Sciences Research Council (EPSRC) in the UK, under grant reference GR/S13330/01.

[2]These algorithms all operate on domains which are in the direct visual field of both speaker and hearer. Throughout this paper we will make this assumption as well.

on the assumption that a pointing gesture is precise and unambiguous. As soon as a pointing gesture is included, it directly eliminates the distractors and singles out the intended referent. As a consequence, the generated expressions tend to be relatively simple and usually contain no more than a head noun (*this block*) in combination with a pointing gesture. Moreover, most algorithms tend to be based on relatively simple, context-independent criteria for the decision whether a pointing gesture should be included or not. For instance, Claassen 1992 only generates a pointing gesture when referring to an object for which no distinguishing linguistic description can be produced. Lester et al. 1999 generate pointing gestures for all objects which cannot be referred to with a pronoun. Van der Sluis and Krahmer (2001) use pointing if the object is close or when a purely linguistic description is too complex, where both closeness and complexity are measured with respect to a predefined threshold.

The approach described in this paper differs from these earlier proposals in a number of ways. We do not assume that pointing is always precise and unambiguous. Rather we allow for various gradations of preciseness in pointing, ranging from unambiguous to vague pointing gestures. Precise pointing has a high precision. Its scope is restricted to the target object, and this directly rules out the distractors. But, arguably, precise pointing is 'expensive'; the speaker has to make sure she points precisely to the target object in such a way that the hearer will be able to unambiguously interpret the referring expression. Imprecise pointing, on the other hand, has a lower precision —it generally includes some distractors in its scope— but is intuitively less 'expensive'.[3]

The model for pointing we propose may be likened to a **flashlight**.[4] If one holds a flashlight just above a surface, it will cover only a small area (the target object). Moving the flashlight away will enlarge the cone of light (shining on the target object but probably also on one or more distractors). A direct consequence of this "Flashlight model for pointing" is that we predict that the amount of linguistic properties required to generate a distinguishing multimodal referring expression is dependent on the *kind* of pointing gesture. Imprecise pointing will require more additional linguistic properties to single out the intended referent than precise pointing.

In our proposal, the decision to point is based on a trade-off between the costs of pointing and the costs of a linguistic description. The latter are determined by summing over the costs of the individual linguistic properties used in the description. Arguably, the costs of precise pointing are determined by two factors: the size of the target object (a big object is easier to point at than a small objects) and the distance between the target object and the pointing device (objects which are near are easier to point to than objects that are further away). As we shall see, Fitts' law —a fundamental empirical law about the human motor-system due to Fitts (1954)— can be used to model the costs of precise pointing. In addition, we shall argue that Fitts' law allows us to capture the intuition that imprecise pointing is cheaper than precise pointing.

The algorithm we describe in this paper is a variant of the graph-based generation algorithm described in Krahmer et al. (2003). It models scenes as labelled directed graphs, in which objects are represented as vertices (or nodes) and the properties and relations of these objects are represented as edges (or arcs). Cost functions are used to assign weights to edges. The problem of finding a referring expression for an object is treated as finding the *cheapest* subgraph of the scene graph which uniquely characterizes the intended referent. For the generation of multimodal referring expressions, the scene graph is enriched with edges representing the various kinds of pointing gestures. Since the algorithm looks for the cheapest subgraph, pointing edges will only be selected when linguistic edges are relatively expensive or when pointing is relatively cheap.

The rest of this paper is organized as follows. In section 2 we describe the ingredients of the multimodal graph-based approach to the generation

[3] This intuition is in line with the alleged existence of neurological differences between precise and imprecise pointing. The former is argued to be monitored by a slow and conscious feedback control system, while the latter is governed by a faster and non-conscious control system located in the center and lower-back parts of the brain (see e.g., Smyth and Wing 1984, Bizzi and Mussa-Ivaldi 1990).

[4] This analogy was suggested by Mariët Theune (*p.c.*)

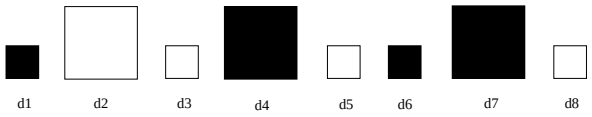

Figure 1: An example scene.

of referring expressions. Section 3 is devoted to determining the costs of linguistic properties and gestures. Section 4 describes the algorithm, and illustrates it with a worked example. In section 5, we summarize and discuss some of the properties and predictions of the model.

2 Generating multimodal referring expressions

2.1 Scene graphs Consider the visual scene depicted in Figure 1, consisting of a set of objects with various properties and relations. In this particular scene $M = \{d_1, \ldots, d_8\}$ is the set of entities, *Prop* $=$ { small, large, black, white, block } is the set of properties of these objects and *Rel* $=$ { left-of, right-of } the set of relations. We represent a scene as a **labelled directed graph**. Let $L = \mathit{Prop} \cup \mathit{Rel}$ be the set of labels with *Prop* and *Rel* disjoint, then $G = \langle V_G, E_G \rangle$ is a labelled directed graph, where $V_G \subseteq M$ is the set of vertices and $E_G \subseteq V_G \times L \times V_G$ is the set of labelled directed edges.[5] Two other notions that we use in this paper are graph union and graph extension. The **union** of graphs $F = \langle V_F, E_F \rangle$ and $G = \langle V_G, E_G \rangle$ is the graph $F \cup G = \langle V_F \cup V_G, E_F \cup E_G \rangle$. If $G = \langle V, E \rangle$ is a graph and $e = (v, l, w)$ is an edge between vertices v and w and with label $l \in L$, then the **extension** of G with e (notated $G + e$) is the graph $\langle V \cup \{v, w\}, E \cup e \rangle$.

Figure 2 contains a graph representation of the scene depicted in Figure 1.[6] Notice that properties are represented as **loops**, while relations are modelled as edges between different vertices.

2.2 Referring graphs Suppose we want to generate a distinguishing description referring to d_4. Then we have to determine which properties

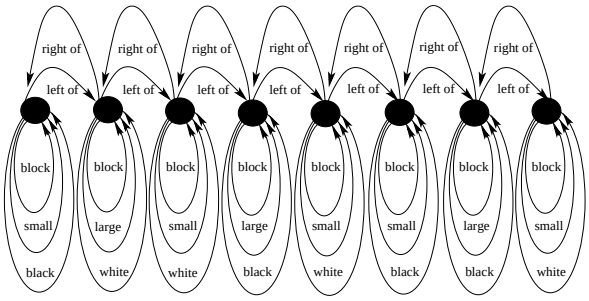

Figure 2: Example scene as a graph.

and/or relations are required to single out d_4 from its distractors This is done by creating *referring graphs*, which at least include a vertex representing the target object. Informally, a vertex v (the target object) in a referring graph H **refers** to a given entity in the scene graph G iff the graph H can be "placed" over the scene graph G in such a way that v can be placed over the vertex of the given entity in G and each edge from H with label l can be "placed over" a corresponding edge in G with the same label. Furthermore, a vertex-graph pair is **distinguishing** iff it refers to exactly one vertex in the scene graph.[7]

Consider Figure 3, containing a number of potential referring graphs for d_4, each time with a circle around the intended referent. The first one, H_1 has all the properties of d_4 and hence can refer to d_4. It is not distinguishing, however: it fails to rule out d_7 (the other large black block). Graph H_2 *is* distinguishing. Here, the circled vertex can only be "placed over" the intended referent d_4 in the scene graph. A straightforward linguistic realization (expressing properties as adjectives and relations as prepositional phrases) would be something like "the large black block to the left of a small white block and to the right of another small

[5] Here and elsewhere subscripts are omitted when this can be done without creating confusion.

[6] We only model the direct spatial relations under the assumption that a distinguishing description would not use a distant object as a relatum when a closer one can be selected.

[7] The informal notion of one graph being "placed over" another corresponds with a well-known mathematical construction on graphs, namely **subgraph isomorphism**. $H = \langle V_H, E_H \rangle$ can be "placed over" $G = \langle V_G, E_G \rangle$ iff there exists a subgraph G' of G such that H is isomorphic to G'. H is isomorphic to G' iff there exists a bijection $\pi : V_H \rightarrow V_{G'}$, such that for all vertices $v, w \in V_H$ and all $l \in L$:

$$(v, l, w) \in E_H \Leftrightarrow (\pi.v, l, \pi.w) \in E_{G'}$$

Given a graph H and a vertex v in H, and a graph G and a vertex w in G, we define that the pair (v, H) **refers** to the pair (w, G) iff H is connected and H is mapped to a subgraph of G by an isomorphism π and $\pi.v = w$.

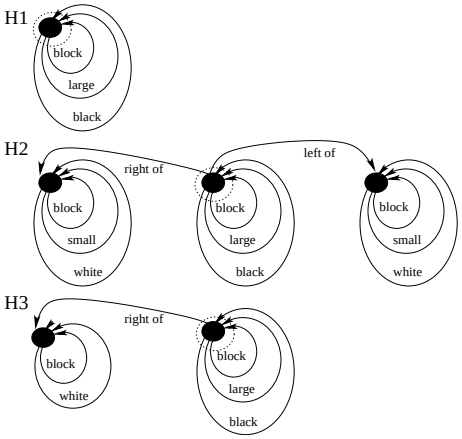

Figure 3: Three potential referring graphs for d_4.

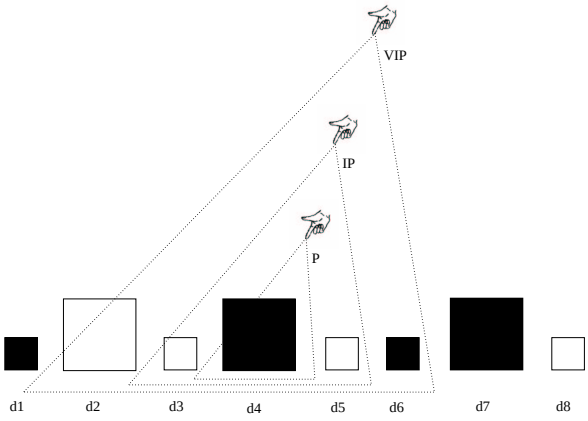

Figure 4: Pointing into the scene

white block".[8] Generally there is more than one distinguishing graph referring to an object. In fact, H_2 is not the *smallest* distinguishing graph referring to d_4. This is H_3. It might be realized as "the large black block to the right of a white block". This is a distinguishing description but not a particular natural one; it is complex and arguably difficult for the hearer to interpret. In such cases, having the possibility to simply point to the intended referent would be very useful.

2.3 Gesture graphs Suppose we want to *point* to d_4. Clearly this can be done from various distances and under various angles. The various hands in Figure 4 illustrate three levels of deictic pointing gestures, all under the same angle but each with different distances to the target object: **precise** pointing (P), **imprecise** pointing (IP) and **very imprecise** pointing (VIP). We shall limit the presentation here to these three levels of precision and a fixed angle, although nothing hinges on this. Naturally, the respective positions of the speaker and the target object co-determine the angle under which the pointing gesture occurs; this in turn fixes the 'scope' of the pointing gesture and thus which objects are ruled out by it.[9] If these respective positions are known, computing the scope of a pointing gesture is straightforward, but the actual mathematics falls outside the scope of this paper.

Just as properties and relations of objects can be expressed in a graph, so can various pointing gestures to these objects. All objects in the scope of a potential pointing gesture (with a certain degree of precision) are associated with an edge labelled with an indexed pointing gesture. Selecting this edge implies that all objects which fall outside the scope of the gesture are ruled out. We represent this information using a **gesture graph**. Let $PG_v = \{\mathsf{P}_v, \mathsf{IP}_v, \mathsf{VIP}_v\}$ be the set of pointing gestures to a target object v. Then, given a scene graph $G = \langle V_G, E_G \rangle$, a gesture graph $D_v = \langle V_G, E_D \rangle$ is a labelled directed graph, where V_G is the set of vertices from the scene graph and $E_D = V_G \times PG_v \times V_G$ the set of pointing edges. Figure 5 displays a graph modelling the various pointing gestures in Figure 4. Notice that there is one gesture edge which is only associated with d_4, the one representing precise pointing to the target object (modelled by edge P_4). No other pointing gesture eliminates all distractors.

2.4 Multimodal graphs Now the generation of multimodal referring graphs is based on the union of the scene graph G (which is relatively fixed) with the deictic gesture graph D (which varies with the target object). Figure 6 shows three distinguishing multimodal referring graphs for our target object d_4. H_1 is the smallest, only consisting of an edge modelling a precise pointing ges-

[8] A somewhat more involved lexicalization module (using aggregation) might realize this graph as "The large black block in between the two small white blocks".

[9] Here, for the sake of simplicity, we assume that an object falls inside the scope of a pointing gesture if the 'cone' shines on part of it. A more fine-grained approach might distinguish between objects in the center (where the light shines brightly) and objects in the periphery (where the light is more blurred).

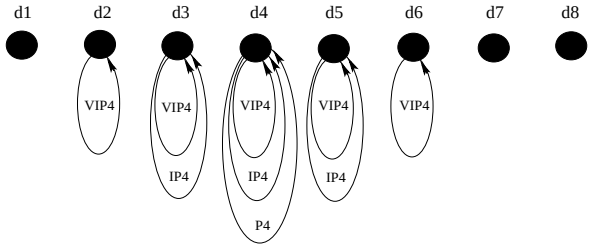

Figure 5: Deictic gesture graph

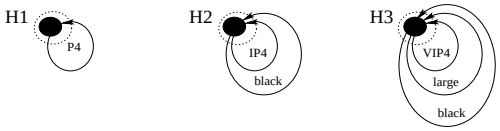

Figure 6: Three distinguishing multimodal referring graphs for d_4.

ture. It might be realized as "this one" combined with a precise pointing gesture. H_2 incorporates an imprecise pointing gesture (of the kind shown in Figure 4). Since this imprecise pointing gesture does not eliminate the distractors d_3 and d_5, a further edge is required, expressing that d_4 is black. This graph could be realized as "this black one" combined with an imprecise pointing gesture. Finally, H_3 is a distinguishing graph which incorporates a very imprecise pointing gesture. Including such an edge only rules out the distractors d_1, d_7 and d_8. At least two additional edges are required for the construction of a distinguishing graph, expressing that d_4 is both large and black. The resulting graph might be realized as "this large black one" in combination with a very imprecise pointing gesture. Arguably, in the scene of interest these multimodal referring expressions seem preferable to the linguistic expression from section 2 (*the large black block to the right of a white one*).

3 Cost functions

We now have many ways to generate a distinguishing referring expression for an object. Cost functions are used to give preference to some solutions over others. Costs are associated with subgraphs H of the scene graph G. We require the cost function to be **monotonic**. This implies that extending a graph H with an edge e can never result in a graph which is cheaper than H.[10] We assume that if H is a subgraph of G, the costs of H (notated *cost*(H)) can be determined by summing over the costs associated with the edges of H.

3.1 The costs of properties The idea that certain linguistic properties are 'cheaper' than others is already implicit in the notion of *preferred attributes* in the incremental algorithm of Dale and Reiter (1995), and is based on psycholinguistic evidence. If someone wants to describe an object, (s)he will first describe the "type" (what *kind* of object it is; a block, an animal or whatever). If that does not suffice, first **absolute** properties like color may be used, followed by **relative** ones such as size. In terms of costs, we assume that type properties (block) are for free. Other properties are more expensive. Absolute properties (colors such as black and white) are cheaper than relative ones (representing size, such as small or large). There is little empirical work on the costs of relations, but it seems safe to assume that for our example scene atomic relations are more expensive than atomic properties. First, relations are comparable to relative properties (they can not be verified on the basis of the intended referent alone). In addition, using a relation implies that a second object (the **relatum**) needs to be described as well and describing two objects generally requires more effort than describing a single object.

3.2 The costs of pointing Arguably, at least two factors co-determine the costs of pointing: (*i*) the size S of the target object (the bigger the object, the easier, and hence cheaper, the reference), and (*ii*) the distance D which the pointing device (in our case the hand) has to travel in the direction of the target object (a short distance is cheaper than a long one).[11] Interestingly, the pioneering work of Fitts (1954) captures these two factors in the *Index of Difficulty*, which states that the difficulty to reach a target is a function of the size of and the distance to a target: $ID = \log_2(\frac{2D}{S})$. Thus with each doubling of distance and with each halving

[10] Formally, $\forall H \subseteq G\ \forall e \in E_G : cost(H) \leq cost(H + e)$.

[11] A third factor which seems to be relevant is the *salience* of the target. For a detailed discussion of this aspect we refer to van der Sluis and Krahmer (2001). See also Section 5.

of size the index of difficulty increases with 1 bit. The addition of the factor 2 in the numerator is unmotivated; Fitts added it to make sure that in his experimental conditions the *ID* was always positive. He performed three experiments (a tapping, a disk transfer and a pin transfer task) and in all three found a high correlation between the time subjects required to perform the task and the index of difficulty. In recent years various alternatives for the original *ID* have been proposed. MacKenzie's (1991) alternative removes the unmotivated 2 from the numerator and starts counting from 1 assuring that the *ID* is always positive.

$$ID = \log_2(\frac{D}{S} + 1)$$

MacKenzie shows that this version of the *ID* fits the experimental data slightly better. Below we derive the costs of pointing from this index of difficulty. As argued, it seems a reasonable assumption that *imprecise* pointing is cheaper than precise pointing; it rules out fewer distractors, but also requires less motoric precision and effort from the speaker. The index of difficulty allows us to capture this intuition. We do not interpret the distance D as the distance from the neutral, current position of the hand to the target object, but rather as the distance from the current position of the hand to the target position of the hand. For the imprecise variants of pointing this distance will be smaller and hence the index of difficulty will be lower.

4 Sketch of the algorithm

In this section we describe an algorithm which outputs the cheapest distinguishing graph for a target object, and illustrate it with an example. Whether this cheapest graph will include pointing edges, and if so, of what level of precision, is determined by a trade-off between the costs of the linguistic edges representing properties and relations of the target object and the costs of pointing. The algorithm is a multimodal extension of the algorithm described in Krahmer et al. (2003), to which paper we refer for more details about complexity, motivation and implementation.

Suppose we want to generate a description for d_4 from the scene graph G in Figure 2. Before we illustrate the workings of this function we need to specify a cost function. Let us assume that d_4 is a cube with sides of 1 inch, and that 31 inches is the distance from the current neutral position of the hand to the target position required for precise pointing, 15 inches for imprecise pointing and 7 inches for very imprecise pointing. Some easy calculations will show that the index of difficulty in the three cases is 5 bits, 4 bits and 3 bits respectively. Thus, precise pointing (P) costs 5.00 points, imprecise pointing (IP) 4.00 and very imprecise pointing (VIP) 3.00. The preferred order for attributes in the current domain is (1) type, (2) color, (3) size and (4) relations. In terms of costs, let us assume for the sake of illustration that type edges (block) are for free, color edges cost 0.75, size edges cost 1.50 and relational edges 2.25.

```
makeReferringExpression(v, G) {
    construct D_v;
    M := D_v ∪ G;
    bestGraph := ⊥;
    H := ⟨{v}, ∅⟩;
    return findGraph(v, bestGraph, H, M); }

findGraph(v, bestGraph, H, M) {
    if [bestGraph ≠ ⊥ and cost(bestGraph) ≤ cost(H)]
    then return bestGraph;
    distr := {n ≠ v | n ∈ V_M ∧ (v, H) refers to (n, M)};
    if distr = ∅ then return H;
    for each adjacent edge e do
        I := findGraph(v, bestGraph, H + e, M );
        if [bestGraph = ⊥ or cost(I) ≤ cost(bestGraph)]
        then bestGraph := I;
    return bestGraph; }
```

Figure 7: Sketch of the algorithm.

We call the function **makeReferringExpression** (d_4, G), outlined in figure 7. First of all the deictic gesture graph D_{d_4}, adding pointing edges of various levels of precision to d_4, is constructed (see Figure 4), and merged with G. This gives us a multi-modal graph M. The variable *bestGraph*, for the cheapest solution found so far, is initialized as the undefined graph $\perp$ (no solution was found yet), and the referring graph under construction H is initialized as the graph only consisting of the vertex d_4. We call the function **findGraph** with as parameters the target object d_4, the best graph so

far ($\perp$), the graph under construction H and the multi-modal graph M. Now the algorithm systematically tries all relevant subgraphs H of M. It starts from the graph which only contains the vertex d_4 and the algorithm recursively tries to extend this graph by adding *adjacent* edges (that is edges which start in d_4 or possibly in any of the other vertices added later on to the H under construction). For each graph H it checks to which objects in M (different from d_4) the vertex-graph pair (d_4, H) may refer; these are the *distr*actors. As soon as this set is empty we have found a distinguishing graph referring to d_4. This graph is stored in the variable *bestGraph* for the cheapest distinguishing graph found so far. In the end the algorithm returns the cheapest distinguishing graph which refers to the target object, if one exists, otherwise it returns the undefined null graph $\perp$. In the current set up the latter possibility will never arise due to the presence of unambiguous pointing gestures (expensive though they may be). Which referring graph is the first to be found depends on the order in which the edges are tried (clearly this is a place where heuristics are helpful, e.g., it will generally be beneficial to try cheap edges before expensive ones). Let us say, for the sake of argument, that the first distinguishing graph which the algorithm finds is H_3 from Figure 3. This graph costs 5.25. At this point, graphs which are as expensive as this graph can be discarded (since due to the monotonicity constraint they will never end up being cheaper than the best solution found so far). In the current situation, the cheapest solution is H_2 from Figure 6, which costs a mere 4.75.[12] The resulting graph could be realized as "this black one" combined with an imprecise pointing gesture.

5 Discussion

We have described a new model for the generation of multimodal referring expressions. The approach is based on only a few, independently motivated assumptions. The starting point is a graph-based algorithm which tries to find the cheapest referring expression for a particular target object (Krahmer et al. 2003). We assume that linguistic properties have certain costs (c.f., the preferred attributes from Dale & Reiter 1995). And, finally, we propose a "flashlight" model of pointing allowing for different gradations of pointing precision, ranging from precise and unambiguous to imprecise and ambiguous. The costs of these various pointing gestures are derived from an empirically motivated adaptation of Fitts' (1954) law.

The model has a number of nice consequences. We have described two in detail: (1) we do not need an *a priori* criterion to decide when to include a pointing gesture in a distinguishing description. Rather the decision to point is based on a trade-off between the costs of pointing and the costs of a linguistic description. And (2) we predict that the amount of linguistic properties required to generate a distinguishing multimodal referring expression is dependent on the *kind* of pointing gesture. One further neat consequence of the model is that an isolated object does not require precise pointing; there will always be a graph containing a less precise (and hence cheaper) pointing edge which has the same objects in its scope as the more precise pointing act. Notice also that the algorithm will never output a graph with multiple pointing edges, since there would always be a cheaper graph which omits the less precise one. In most situations, it will also not happen that a distinguishing graph will include both an imprecise pointing gesture and a relational edge. Under most cost functions it will be more 'cost effective' to include a precise pointing edge than an imprecise pointing edge *plus* a relational edge *plus* the edges associated with the relatum.

The algorithm we have described has been implemented in Java 2 (J2SE, version 1.4). The computation described in section 4 requires 110 ms. on a PC with a 900 mHz AMD Athlon Processor and 128 Mb RAM. Due to the presence of precise pointing edges it will always be possible to single out one object from the others. As a side effect of this we obtain a polynomial upperbound for the theoretical complexity.[13] It has been argued that

[12] Note that if pointing would have been cheaper (because the distance between the current position of the hand and the required position for precise pointing was, say, 3 inches), the algorithm would output "this one" plus a precise pointing edge (i.e., H_1 from Figure 6, for 2.00). If pointing would be more expensive (because even for very imprecise pointing the distance would be substantial), the algorithm would output H_3 from Figure 3, for 5.25.

[13] We know the costs of at least one distinguishing graph

some notion of *focus of attention* could be used to tackle the computational complexity. We may assume that objects which are currently in the focus of attention are more salient than objects which are not in focus. Now the distractor set for a target object need not include *all* objects in the domain, but only those that are at least as salient as the target object. A distinguishing description only needs to rule out those objects. There are two interesting connections between focus of attention and multimodality. First, pointing gestures typically serve to demarcate the focus of attention. Second, the model described in this paper predicts that a distinguishing description for an object which is salient is less likely to contain a pointing gesture. If an object is salient, this generally implies that its distractor set is relatively small (typically, only a few objects are somehow salient). This in turn implies that fewer (or less expensive) edges are required to rule out the distractors, hence there is less need for deictic pointing gestures.

It is interesting to observe that, even though we borrow the idea of preferred attributes from the Incremental Algorithm (arguably the most influential algorithm for the generation of referring expressions), an incremental approach to multimodal descriptions does not seem to be straightforward. One might consider extending the list of preferred attributes with VIP, IP and P (in that preference order, modelling the increase in costs). On this approach, we would first select a number of linguistic edges (independent of the kind of pointing gesture) followed by one or more pointing edges. But that would not work, since the lack of backtracking (which is inherent to incrementality) entails that all selected properties will be realized. This seems to suggest that the model outlined in this paper is inherently non-incremental.

We are currently running an experimental evaluation of the model, particularly addressing the vague pointing gestures and their interaction with linguistic realization. We hope to present the results of this evaluation in a sequel to this paper.

for our target object; the graph consisting of only a vertex for the target object and a precise pointing edge. This means that we do not have to inspect all subgraphs of the merged multimodal graph M, but only those subgraphs which do not cost more than the precise pointing graph. Thus, we only need to inspect graphs with less than K edges (for some K depending on the costs of precise pointing), which requires in the worst case $\mathcal{O}(n^K)$, with n the number of edges in the graph M. It should be added that this worst case complexity is computationally rather unattractive for larger values of K.

References

André, E. and T. Rist (1996), Coping with Temporal Constraints in Multimedia Presentation Planning, *Proceedings of the 13th AAAI*, 142–147.

Beun, R.J. & A. Cremers (1998), Object reference in a shared domain of conversation, *Pragmatics & Cognition* **6**(1/2):121–152.

Bizzi, E. and F. Mussa-Ivaldi (1990), Muscle properties and the control of arm movement, In: *Visual Cognition and Action (vol 2)*, D. Osherson, et al. (eds.), MIT Press.

Cohen, P. (1984), The pragmatics of referring and the modality of communication, *Computational Linguistics* **10**(2):97–125.

Claassen, W. (1992), Generating referring expressions in a multimodal environment, in: *Aspects of Automated Natural Language Generation*, R. Dale et al. (eds.), Springer Verlag, Berlin.

Dale, R. and E. Reiter (1995), Computational interpretations of the Gricean maxims in the generation of referring expressions, *Cognitive Science* **18**:233–263.

van Deemter, K. (2002), Generating referring expressions: Beyond the Incremental Algorithm, *Computational Linguistics* **28**(1):37–52.

Fitts, P. (1954), The information capacity of the human motor system in controlling amplitude of movement, *Journal of Experimental Psychology* **47**:381–391.

Gardent, C. (2002), Generating minimal definite descriptions, *Proceedings of the 40th ACL*, Philadelphia, USA.

Huls, C. E. Bos & W. Claassen (1995), Automatic referent resolution of deictic and anaphoric expressions, *Computational Linguistics* **21**(1):59–79.

Krahmer, E. S. van Erk & A. Verleg (2003), Graph-based Generation of Referring Expressions, *Computational Linguistics*, **29**(1): 53–72.

Lester, J., J. Voerman, S. Towns and C. Callaway (1999), Deictic Believability: Coordinating gesture, locomotion, and speech in lifelike pedagogical agents, *Applied Artificial Intelligence* **13**(4-5):383–414.

Smyth, M. and Wing, A. (1984), *The Psychology of Human Movement*, New York: Academic Press.

MacKenzie, I.S. (1991), *Fitts' law as a performance model in human-computer interaction*, doctoral dissertation, University of Toronto, Canada.

van der Sluis, I. and E. Krahmer (2001), Generating Referring Expressions in a Multimodal Context: An empirically motivated approach. *Selected Papers from the 11th CLIN Meeting*, W. Daelemans et al. (eds.), Rodopi, Amsterdam.