Towards the Generation of Overspecified Multimodal Referring Expressions

Ielka van der Sluis
Computational Linguistics & AI,
Faculty of Arts, Tilburg University
I.F.vdrSluis@uvt.nl

Emiel Krahmer
Communication & Cognition,
Faculty of Arts, Tilburg University
E.J.Krahmer@uvt.nl

1 Introduction

Recently, there has been an increased interest in the generation of multimodal referring expressions (e.g., Salmon-Alt and Romary, 2000; Lester et al., 1999; André and Rist, 1996; Reithinger, 1992; Claassen, 1992). The task involved in generation of multimodal referring expressions is to decide what is the best way to refer to a target via different modalities in the current context. Existing algorithms that generate referring expressions focus on **distinguishing references** (e.g., descriptions that only apply to the referent and not to any other object in the domain) or **minimal references** (e.g., the shortest distinguishing descriptions possible for a given referent). However in human conversation **overspecified references** (e.g., descriptions which include more information than necessary for identification) are relatively more common (Arts, 2004; Beun and Cremers, 1998; Pechmann, 1989). But in fact, what kind overspecification do human speakers produce and why? And secondly, how can this be mimicked in automatic generation? This paper attempts to answer these questions by: (1) addressing overspecification as occurring in human communication with respect to two production experiments used for the evaluation of the graph-based multimodal algorithm proposed by Krahmer and van der Sluis (2003) and (2) proposing a variant of this multimodal algorithm that can generate overspecified descriptions based on strategies observed in human communication. The basic multimodal algorithm, an extension of the graph-based algorithm (Krahmer et al., 2003), differs from other multimodal algorithms in that it can generate various pointing gestures differing in precision as a function of distance, where the type of pointing gesture co-varies with the amount of linguistic information in the accompanying description. The algorithm employs a set of cost functions to select properties, relations and pointing gestures according to preference. In this paper a notion of certainty is proposed that together with the cost functions causes a flexibility with which the algorithm can generate the whole range of possible referring expressions, from minimal ones to the utmost overspecified ones. This paper is organized as follows: In Section 2 overspecification in human communication is discussed. Section 3, presents a variant of the multimodal graph-based algorithm, that is able to generate unimodal and multimodal overspecified referring expressions. Section 4 ends this paper with a discussion.

2 Overspecification in Human Communication

Why do speakers produce overspecified referring expressions? In the literature a number of partially overlapping suggestions can be found. For instance, the experiments performed by Pechmann (1989) explain overspecification with the assumption that language production is incremental in nature, meaning that perceived properties are almost simultaneously verbalized. According to this view, speakers are highly affected by their perception of the domain of conversation. This causes for example salient (i.e. easy perceptible) properties to be mentioned earlier than other object properties (c.f. Mangold and Pobel, 1988). It may be that a first property is made redundant by the inclusion of a later property, which leads to an overspecified description. This view is consistent with current theories of reference proposed by Ariel (2001) and Gundel et al. (1993). These theories explain the degree of overspecification in terms of accessibility or focus of attention, that are influenced by features like the intrinsic properties of the objects in the domain, the discourse history and the focus space. The less accessible or salient an object in the discourse, the more overspecified the referring expression used to indicate the object.

Apart from salience, object or domain related influences and aspects that concern language production itself, factors that relate to the performance of the discourse have also been argued to play a role in the production of referring expressions (c.f. Jordan, 2002; Maes et al., 2004). For instance discourse goals, task importance, the different modes of communication and situational conditions bear upon the speaker's behavior. With respect to these discourse related factors, Maes et al. (2004) state that the production of overspecified referring expressions is affected by the principle of distant responsibility (Clark and Wilkes-Gibbs, 1986), which says that a speaker must be certain that the information provided in an utterance is understandable for the user. In distinguishing a target that is not salient the speaker might be relatively uncertain and use a highly overspecified description. In contrast, when the target is a salient object in the domain the speaker can be rather confident in identifying it and use a less overspecified or minimal description. The notion of task importance relates in the same way to the speaker's certainty. In cases where the speaker wants to be sure that the hearer understands, the degree of overspecification is relatively high. Hence, in the automatic generation of referring expressions discourse related factors can be used as indicators of the probability that the hearer might misunderstand a particular referring expression. Thus, the degree of overspecification can be determined on the basis of this estimation.

But then, what kind of overspecification do speakers actually produce as a result of the factors mentioned above? For data-driven development and testing of multimodal interpretation and generation modules it is important to collect data on how humans produce referring expressions combining speech and gesture (e.g., Kranstedt et al., 2003; Piwek and Beun, 2001). Several studies on human production of referring expressions, both unimodal and multimodal, have been conducted (e.g. Maes et al., 2004; Arts, 2004; Beun and Cremers, 1998; Pechmann, 1989; van der Sluis and Krahmer, 2004a and 2004b). Surprisingly, in the experiments reported by Maes et al. (2004), functional properties are rarely used. In contrast, the kind of properties used to describe the target are all of a perceptual nature. Of the perceptual properties salient properties of an object are likely to be included in a referring expression because these properties are more easily perceived and thereby facilitate identification for both speaker and hearer (c.f Beun and Cremers, 1998; Pechmann, 1989). The perception and production experiments in a block domain conducted by Arts (2004) show that the inclusion of locative expressions is highly beneficial; compared to object descriptions that included only intrinsic properties, like *shape* and *color*, objects referred to by overspecified descriptions that included locative expressions were faster identified.

Two studies, conducted to evaluate the graph-based multimodal algorithm, are reported by van der Sluis and Krahmer(2004a; 2004b). In these studies subjects that participated in an object identification task, were divided into two groups. One group performed the task in the 'near condition': standing close to the domain, subjects could touch a target while producing a 'precise pointing gesture'. The other group performed the task in the 'far condition': subjects were located further away from the target domain and used 'imprecise pointing gestures' to roughly indicate the location of a target. The results of both studies indicate that speakers indeed vary the linguistic part of a multimodal referring expression in relation to their distance to the target, and that the amount of linguistic material co-varies with the kind of pointing gesture. In both studies, when the target is close, speakers reduce the linguistic material to almost zero and only generate a precise pointing gesture, whereas subjects tend to produce overspecified descriptions combined with imprecise pointing gestures when the target is located further away. The production of overspecified descriptions that accompany the imprecise pointing gestures can be due to an inherent uncertainty of imprecise pointing. Speakers may not be sure whether their pointing gesture is sufficiently clear. To guarantee that their reference is distinguishing they include additional information. When looking at the linguistic descriptions accompanying the imprecise pointing gestures, the kind of target also appears to play a role. In case the features of the target are harder to recognize, for example because the object is small, or because it does not have prominent features, redundant locative expressions are produced to single out the target. In contrast when the target is easy to describe overspecification occurs as well, but in these cases, instead of locative expressions, redundant properties are added to the target description.

Arguably, uncertainty on the side of the speaker invokes overspecification due to the distance to and the kind of target. On the one hand, overspecification is caused by the inclusion of redundant linguistic properties (e.g., **unimodal overspecification**). On the other hand, overspecification is due to the production of pointing gestures together with locative expressions (e.g., **multimodal overspecification**), where locative expressions are defined as linguistic pointing following Arts (2004).[1] In the next section a variant of the graph-based multimodal algorithm is proposed that is able to generate overspecified descriptions similar to the ones occurring in human communication.

3 Generation of Overspecified Referring Expressions

The multimodal graph-based algorithm uses a **domain graph** to represent the domain of conversation as a labelled directed graph. The objects in the domain graph are defined as the vertices (or nodes) in the graph. The properties, relations and pointing gestures that can be used to identify the objects in the domain are represented as edges (or arcs). To generate a multimodal referring expression the graph-based algorithm searches for the cheapest subgraph (i.e. a **referring graph** that represents the target in the domain graph. Cost functions that assign weights to the edges are used to determine their order of preference in selection. Correspondingly, the decision to point is based on a trade-off between the costs of pointing and the costs of the linguistic properties. The algorithm differs from other algorithms in that it can generate various kinds of pointing gestures, precise and imprecise ones. The implemented model that incorporates these different types of pointing gestures, may be likened to a flashlight as illustrated in Figure 1. When one holds a flashlight just above a surface, (i.e. **precise pointing**), it covers only a small area (the target). Moving the flashlight away, (i.e. **imprecise pointing**), enlarges the cone of light, shining on the target but probably also on one or more other objects. For the sake of simplicity, the various pointing gestures in Figure 1 have a very precise scope. Of both precise and imprecise pointing gestures it is clear (from the speaker's perspective) which of the objects in the domain are contained in the scope of the gesture. As indicated in Section 2, in reality it is not that simple. The scope of a precise pointing gesture may uniquely indicate one object, but, at least from a hearers point of view, the boundaries of the scope of an imprecise pointing gestures is vague. A direct consequence of this **Flashlight Model for Pointing** is that it predicts that the amount of linguistic properties required to generate a distinguishing multimodal referring expression co-varies with the kind of pointing gesture, as confirmed by the studies briefly described in Section 2. Imprecise pointing requires more additional linguistic properties to single out the intended referent than precise pointing.

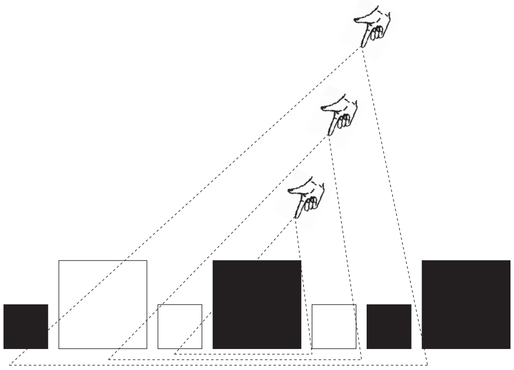

Figure 1: Flashlight model for Pointing

In human communication the linguistic descriptions used together with imprecise pointing gestures are of a different type than the multimodal graph-based algorithm generates, namely they

[1]As such, multimodal overspecification occurs if a pointing gesture and a locative expression are used together in a referring expression to indicate the same object, where one of them would be distinguishing. However, because both the scope of a pointing gesture and the scope of a locative expression might be vague, it is uncertain to what extent they converge. Therefore, it might happen that a pointing gesture rules out other objects than a locative expression and lead to partly overspecified referring expressions. Discovery of the exact scope of pointing gestures and the scope of locative expressions demands a detailed analysis, which cannot be captured from the data of the experiments presented here.

are overspecified. Because the algorithm searches for the cheapest referring graph that describes the intended referent, overspecified referring expressions are never generated. In an attempt to adapt the algorithm to overspecification as appearing human production, in this section a variant of the algorithm is proposed, in which the cheapest subgraph may be expanded to an overspecified graph. To determine the degree and the kind of overspecification of the referring expression to be generated, the algorithm makes use of a Certainty Score. Intuitively, the Certainty Score of a referring graph represents the speaker's estimate of the probability that the resulting expression may be misunderstood by the hearer. This probability may depend on, for instance perceptual salience or vagueness of a property. The context determines what an acceptable chance for misunderstanding is, for a particular task (compare the principle of distant responsibility). This is captured using a Certainty Threshold. The issue which properties to select to extend the graph is addressed by providing the edges in the domain graph with additional Certainty Scores. The Certainty Scores are used as indicators of confidence with respect to the referring expression generated so far. The Certainty Score and the Certainty Threshold are explained in Section 3.1. In Section 3.2 the distribution of the Certainty Scores over the various edges is addressed. In Section 3.3 the workings of the algorithm are illustrated with an example.

3.1 Certainty Score

To be able to decide if the degree of overspecification of a graph that describes the target is satisfying or not, the algorithm uses a **Certainty Score**. The Certainty Score is defined as a numerical value on a scale [0,1], which indicates the speaker's calculation of the chance on an incorrect or correct understanding by the hearer. The value 0 expresses that the hearer is probably not able to interpret the referring expression, while the value 1 reflects absolute certainty that the hearer can resolve the target. In the generation process every generated graph receives a Certainty Score which is the sum of the certainty scores that relate to the properties, relations and pointing gestures contained, i.e. every edge in a graph has a Certainty Score. To determine if a graph is adequate to refer to the target, the Certainty Score of the graph is compared to a **Certainty Threshold**. The Certainty Threshold, is a value in the interval [0, 1], which depends on aspects that relate to contextual factors such as task importance and the principle of distant responsibility. In case the Certainty Score of a graph is below the Certainty Threshold, the algorithm cannot be sure that the hearer can resolve the target of the referring expression that can be realized from that graph. Consequently, the graph needs to be extended with extra edges, thereby increasing the degree of overspecification in order to reach the required confidence level.

3.2 Choice of Edges

The Certainty of Properties In general it can be concluded that the Certainty Score of a graph increases when a property or a relation is appended to the graph. As seen in Section 2, additional linguistic information strengthens the speakers confidence in the hearer's understanding. Moreover, properties that are easily perceived like absolute properties are generally faster produced and interpreted than object properties that are not so easy to discover. Intuitively, the use of absolute properties in comparison to relative properties, has a higher positive influence on the speakers confidence about being understood by the hearer. Thus, the preference of absolute properties over relative properties can be used to determine the Certainty Scores of the edges in the graph. For instance, in a block domain, the Certainty Score of the absolute property *color* is higher than the Certainty Score of the relative property *size*. Based on evidence provided in Section 2 spatial relations (i.e. locative expressions) are determined to have a high Certainty Score as well.

The Certainty of Pointing Like additional linguistic edges, pointing gestures increase confidence. Intuitively, the inclusion of a precise pointing gesture takes away all uncertainty about the identity of the target, which causes the Certainty Score to reach full confidence, i.e. 1 on the scale [0,1]. In contrast, imprecise pointing gestures have a lower Certainty Score, because the scope of such gestures may not only include the target but also other objects. Hence, the Certainty Scores of the pointing edges vary with the precision of the pointing gestures. The higher the precision, the higher the Certainty Score of the edge. Arguably, the Certainty Scores of pointing are determined by two factors: the size of the target (large objects are easier to point to than small objects) and the distance between the target and the pointing device (objects that are near are easier to point to than objects that are further away). Interestingly, Fitts' law (Fitts, 1954), a fundamental law on the human motor system derived from empirical evidence, 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 the target and the distance to the target. Accordingly, the Certainty Scores for the pointing gestures can be computed by taking the inverse of the Index of Difficulty (i.e subtracting the Index of Difficulty from 1 on a scale [0,1]). Where A is the distance from the tip of the pointing finger to the target while performing the pointing gesture, and W is the size of target. The algorithm computes the Certainty Scores of a pointing gesture (g) as follows:

$$\mathbf{CertaintyScore}(g) = 1 - (0.1 \times (\log_2(\frac{A}{W} + 1)))$$

The use of Fitt's law to compute the Certainty Scores can be demonstrated accordingly. When a pointing gesture is very precise (i.e. the pointing finger is touching the target) the distance, A, is 0. Correspondingly, the Certainty Score (1 - (0.1×0)) = 1 (i.e. maximal confidence). In case the distance between the target and the pointing finger gets larger, the Certainty Score of the gesture decreases. Suppose a pointing gesture is applied for which the distance from the tip of the pointing finger to the target is 25 cm, where the target is a block with sides of 1 cm. The Certainty Score of this pointing gesture can then be calculated as (1 - (0.1×4.7)) = 0.53. When the distance from the pointing finger to the same object is increased to 70 cm, the corresponding Certainty Score for such a more imprecise pointing gesture is (1 - (0.1×6.2)) = 0.38.

3.3 Worked Example

In this section the multimodal algorithm is informally illustrated with examples in a block domain. A pseudocode of the algorithm is included in the appendix as Figure 3. The function GenerateReferringExpression constructs a multimodal domain graph for the current target from which the function FindGraph generates the cheapest graph that uniquely refers to the target. If this cheapest graph does not satisfy the Certainty Threshold, the function *FindOverspecifiedGraph* takes as input the result of *FindGraph* and generates the cheapest graph that satisfies the Certainty Threshold, by adding more edges. The kind of edges to be appended is determined by a trade-off between the costs of the edges and the Certainty Score of the graph. The Certainty Score of a graph is computed by summing over the Certainty Scores of the edges in the graph.

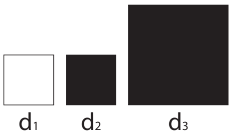

Figure 2: Example Domain

Currently, the algorithm selects properties to describe a target on the basis of cost functions inspired by the notion of **preferred attributes** as proposed by Dale and Reiter (1995). The relevant properties are ordered according to the preference that human speakers and hearers have when discussing objects in a certain domain. Although the exact ordering of properties for a particular domain is an empirical matter, it is generally assumed that speakers have a preference for *absolute* properties such as *color*, over *relative* properties such as *size*. This may be explained by the fact that relative properties are less easily observed and always require inspection of other objects in the domain. Hence, for the objects in the Example Domain presented in Figure 2 the costs of absolute properties are determined to be lower than the costs of relative properties, while spatial relations are even more expensive: *type* = 0, *color* = 1, *shape* = 1.5, *size* = 2 and *spatial* = 3. Based on the observations in Section 2, indicating that locative expressions and perceptual properties such as *color* are helpful, the Certainty Scores are determined as follows: *type* = 0, *color* = 0.20, *shape* = 0.05, *size* = 0.15 and *spatial* = 0.30. Note that *type* and *shape* are not informative in this domain. For the sake of simplicity we include only two pointing gestures in this example: a very precise one, P, with Certainty Score 1 which costs 7 and an imprecise pointing gesture, (VIP, with Certainty

Score 0.30 and costs 2.5.[2] In the following the effects of the Certainty Score on the description of object d_3 in Figure 2 are demonstrated for three cases. In the first case the Certainty Threshold is extremely low, i.e. the objects in the domain are relatively accessible, for instance as a result of the fact that the object domain has been talked about already. The second case shows what happens if the Certainty Threshold has a moderate value. Finally, the third example sketches the outcome of the algorithm if a very high Certainty Threshold is applied, i.e. it is very important that the algorithm makes sure that the hearer can resolve the target, for example because of a high task importance.

In the first case where accessibility is high, suppose that the Certainty Threshold is 0.1. To describe d_3 the function FindGraph generates the cheapest graph that contains the properties *size* and *type*, of which the latter is included for free. The Certainty Score of this graph is 0.15, which means that the Certainty Threshold is met with the minimal graph, and no further computation is necessary. Accordingly, the algorithm produces a graph for costs 2, which can be realized as "the large block".

In the second case, the Certainty Threshold has a moderate value of 0.6. This means that the cheapest graph that FindGraph produces to describe d_3 is not satisfying ($0.15 \leq 0.6$). Consequently, the algorithm invokes the function FindOverspecifiedGraph to increase the Certainty Score of the current graph. The function FindOverspecifiedGraph searches for the cheapest expansion of the graph, that already contains the properties *size* and *type*, in order to increase confidence on the lowest costs. First the property *color* is selected, which takes the Certainty Score up to 0.35. Successively, both the addition of a spatial relation and the addition of a very imprecise pointing gesture, *VIP*, are candidates for the graph with a Certainty Score that levels the threshold. The cheapest option, i.e. *VIP*, is selected. The resulting graph with costs $(2 + 1 + 2.5) = 5.5$ and Certainty Score $(0.15 + 0.20 + 0.30) = 0.65$, can be realized as "the large black block" together with a very imprecise pointing gesture.

In the third case the Certainty Threshold is set very high, 0.9. To generate an adequate description for d_3 the algorithm has to call FindOverpecifiedGraph to expand the cheapest graph resulting from FindGraph. Adding the property *color*, a very imprecise pointing gesture and a spatial relation results in a graph that has an Certainty Score$(0.15 + 0.20 + 0.30 + 0.30) = 0.95$, which fulfils the Certainty Threshold for costs of $(2 + 1 + 2.5 + 3) = 8.5$. However, this graph is more expensive than the costs of a graph with a precise pointing gesture, P and the property *type*, that causes maximal certainty for costs 7. Correspondingly, the algorithm generates this cheaper graph, that can be realized as "this block" with a precise pointing gesture directed to d_3.

4 Discussion

In this paper a graph-based multimodal algorithm is presented which employs a notion of certainty and a set of cost functions to generate multimodal referring expressions that range from minimal to highly overspecified ones. For the cheapest distinguishing graph a Certainty Score is computed to indicate the algorithm's estimation of the probability that the hearer might misunderstand the referring expression that can be realized from that graph. If the Certainty Score meets the Certainty Threshold that is derived from context and discourse related factors, the graph is returned. Otherwise a function FindOverspecifiedGraph is evoked which extends the cheapest graph to an overspecified graph that satisfies the Certainty Threshold on the lowest costs. Compared to the algorithm discussed here the overspecified referring expressions generated by the Incremental Algorithm by Dale and Reiter (1995) seem somewhat accidental. The Incremental Algorithm is able to generate overspecified descriptions due to the interaction between the composition of the list of preferred attributes and the definition of the set of objects from which the target has to be distinguished. For instance, the Incremental Algorithm would describe object d_3 in Figure 2 as "the large black block", where *color* is redundantly included, because it is preferred over *size* and because it rules out object d_1. In contrast the algorithm discussed here determines the

[2]Lack of space prevents the discussion of the cost function for pointing gestures, see Krahmer and van der Sluis (2003) for a precise definition.

degree of overspecification based on an independent method derived from observations in human communication. One limitation of the current approach is that it uses three independently motivated parameters (costs, certainty scores and a certainty threshold). The proper setting of these parameters is an empirical issue, which we hope to address in future work.

References

André, E. and T. Rist (1996). Coping with temporal constraints in multimedia presentation planning. In *Proceedings of the 13th AAAI.*

Ariel, M. (2001). *Text Representation: Linguistic and Psycholinguistic Aspects*, Chapter Accessibility Theory: An overview, pp. 29–87. John Benjamins Publishing Company.

Arts, A. (2004). *Overspecifation in Instructive Texts.* Ph. D. thesis, Tilburg University.

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

Claassen, W. (1992). *Aspects of Automated Natural Language Generation, Lecture Notes in Artificial Intelligence*, Volume 587, Chapter Generating Referring Expressions in a Multimodal Environment, pp. 263–276. Springer Verlag, Berlin.

Clark, H. and D. Wilkes-Gibbs (1986). Referring as a collaborative process. *Cognition 22*, 1–39.

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

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

Gundel, J., N. Hedberg, and R. Zacharski (1993). Cognitive status and the form of referring expressions in discourse. *Language 69*(2), 274–306.

Jordan, P. (2002). *Information Sharing: Reference and Presupposition in Language Generation and Interpretation*, Chapter Contextual Influences on Atrribute Selection for Repeated Descriptions, pp. 295–328. CSLI Publications, Stanford.

Krahmer, E. and I. van der Sluis (2003). A new model for generating multimodal referring expressions. In *Proceedings of the 9th ENLG*, Budapest, Hungary, pp. 47– 54.

Krahmer, E., S. van Erk, and A. Verleg (2003). Graph-based generation of referring expressions. *Computational Linguistics 29*(1), 53–72.

Kranstedt, A., P. Kühnlein, and I. Wachsmuth (2003). Deixis in multimodal human computer interaction: An interdisciplinary approach. In *Proceedings of the 5th GW'03*, Genova, Italy, pp. 112–123.

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.

Maes, A., A. Arts, and L. Noordman (2004). Reference management in instructive discourse. *Discourse Processes 37*, 117–144.

Mangold, R. and R. Pobel (1988). Informativeness and instrumentality in referential communication. *Journal of Language and Social Psychology 7*(3-4), 181–191.

Pechmann, T. (1989). Incremental speech production and referential overspecification. *Linguistics 27*, 98–110.

Piwek, P. and R. Beun (2001). Multimodal referential acts in a dialogue game. from empirical investigations to algorithms. In *Proceedings of the IWIPNMD'01*, Verona, Italy.

Reithinger, N. (1992). *Aspects of Automated Natural Language Generation, Lecture Notes in Artificial Intelligence*, Volume 587, Chapter The Performance of an Incremental Generation Component for Multimodal Dialog Contributions. Springer Verlag, Berlin.

Salmon-Alt, S. and L. Romary (2000). Generating referring expressions in multimodal contexts. In *Proceedings of the INLG'00, Workshop on Coherence in Generated Multimedia*, Mitzpe Ramon, Israel.

van der Sluis, I. and E. Krahmer (2004a). Evaluating multimodal nlg using production experiments. In *Proceedings of the LREC'04*, Lisbon, Portugal.

van der Sluis, I. and E. Krahmer (2004b). The influence of target size and distance on the production of speech and gesture in multimodal referring expressions. In *Proceedings of the 5th ICSLP'04*, Jeju, Korea.

APPENDIX: A Variant of the Multimodal Graph-based Algorithm

GenerateReferringExpression(v, G, T)
  construct(v, F_v, G)
  $M := F_v \cup G$
  $BestGraph := \bot$
  $H := \langle \{v\}, \emptyset \rangle$
  $BestGraph :=$ **FindGraph**$(v, BestGraph, H, M)$
  if CertaintyScore$(BestGraph) < T$ **then**
    $H := BestGraph$
    $BestGraph :=$ **FindOverSpecifiedGraph**$(v, BestGraph, H, M, T)$
  end if
  return $BestGraph$

FindGraph$(v, BestGraph, H, M)$
  if $BestGraph \neq \bot$ **and Cost**$(BestGraph) \leq$ **Cost**(H) **then**
    return $BestGraph$
  end if
  $C := \{n \mid n \in V_M \wedge$ **MatchGraphs**$(v, H, n, M)\}$
  if $C = \{v\}$ **then**
    return H
  end if
  for each adjacent edge e **do**
    $I :=$ **FindGraph** $(v, BestGraph, H + e, M)$
      if $BestGraph = \bot$ **or Cost**$(I) \leq$ **Cost**$(BestGraph)$ **then**
        $BestGraph := I$
      end if
  end foreach
  return $BestGraph$

FindOverspecifiedGraph$(v, BestGraph, H, M, T)$
  if CertaintyScore$(BestGraph) \geq T$ **and**
   Cost$(BestGraph) \leq$ **Cost**(H) **then**
  return $BestGraph$
  end if
  for each adjacent edge e **do**
    $I :=$ **FindOverspecifiedGraph** $(v, BestGraph, H + e, M)$
      if Cost$(I) \leq$ **Cost**$(BestGraph)$ **and**
        CertaintyScore$(I) \geq T$ **then**
        $BestGraph := I$
      end if
  end foreach
  return $BestGraph$

Figure 3: Pseudocode of the algorithm's main function **GenerateReferringExpression** and the subgraph construction functions **FindGraph** and **FindOverspecifiedGraph**, where the global variable T is the Certainty Threshold. An explanation of the graph-based algorithm is given in Krahmer and van der Sluis (2003)