An inferential approach to the Generation of Referring Expressions

Madalina Croitoru and Kees van Deemter

University of Southampton University of Aberdeen

Abstract This paper presents a Conceptual Graph (CG) framework to the Generation of Referring Expressions (GRE). Employing Conceptual Graphs as the underlying formalism allows a new rigorous, semantically rich, approach to GRE: the intended referent is indentified by a combination of facts that can be deduced in its presence but not if it would be absent. Since CGs allow a substantial generalisation of the GRE problem, we show how the resulting formalism can be used by a GRE algorithm that *refers* uniquely to objects in the scene.

1 Introduction

Generation of Referring Expressions (GRE) is a key task in Natural Language Generation (Reiter and Dale 2000). Essentially, GRE models the human ability to verbally identify objects from amongst a set of distractors: given an entity that we want to refer to, how do we determine the content of a referring expression that uniquely identifies that intended referent?

In the classical approach, a GRE generator takes as input (1) a knowledge base (KB) of (usually atomic) facts concerning a set of domain objects, and (2) a designated domain object, called the *target*. The task is to find some combination of facts that singles out the target from amongst all the distractors in the domain. These facts should be true of the target and, if possible, false of all distractors (in which case we speak of a *distinguishing* description). Once expressed into words, the description should ideally be 'natural' (i.e., similar to human-generated descriptions), and effective (i.e., the target should be easy to identify by a hearer). Many of the main problems in GRE are summarized in Dale and Reiter (1995). (See also Dale and Haddock 1991 for GRE involving relations; Van Deemter 2002 and Horacek 2004 for reference to sets and for the use of negation and disjunction). Here, we focus on logical and computational aspects of the problem, leaving empirical questions about naturalness and effectiveness, as well as questions about the choice of words, aside.

Recently, a graph-based framework was proposed (Krahmer et al. 2003), in which GRE was formalised using labelled di-graphs. A two-place relation R between domain objects x and y was represented by an arc labelled R between nodes x and y; a one-place predicate P true of x was represented by an looping arc (labelled P) from x to x itself. By encoding both the description and the KB in this same format (calling the first of these the *description graph* and the second the *scene graph*), these authors described the GRE problem in graph-based terms using subgraph isomorphisms. This provides the ability to make use of different search strategies and weighting mechanisms when adding properties to a description. Their approach is elegant and has the advantage of a visual formalism for which efficient algorithms are available, but it has a number of drawbacks. Most of them stem from the fact that their graphs are not part of an expressively rich overarching semantic framework that allows the KB to tap into existing ontologies, and to perform automatic inference.

It is these shortcomings that we addressed in Croitoru and van Deemter (2006), while maintaining all the other advantages of the approach of Krahmer et al. (2003). The core of our proposal is to address GRE using a Conceptual Graph (GRE) framework. CGs provide a simple approach that adds discriminatory power. This emphasizes the important role the underlying representation plays in the generation of referring expressions: if we want to emulate what people do, then we not only need to design algorithms which mirror their behaviour, but these algorithms have to operate over the same kind of data. Another interesting quality of our approach is that the algorithm devised explicitly tracks the focus of attention. Objects which are "closely related" (in the combinatorial structure provided by the CG) to the most recent target object are taken to be more salient than objects which are not in the current focus space. Conceptual Graphs are a visual, logic-based knowledge representation (KR) formalism. They encode ontological ('T Box') knowledge in a structure called *support*. The support consists of a number of taxonomies of the main concepts and relations used to describe the world. The world is described using a bipartite graph in which the two classes of the partition are the objects, and the relations respectively. The CGs semantics translate information from the support in universally quantified formulae (e.g., 'all cups are vessels'); information from the bipartite graph is translated into the existential closure of the conjunction of formulae associated to the nodes (see section 3.2). A key element of CGs is the logical notion of subsumption (as modelled by the notion of a projection), which will replace the graph-theoretical notion of a subgraph isomorphism used by Krahmer et al. (2003).

The main contribution of the present paper is to highlight that the CG framework allows us to replace the GRE-classical content determination approach by an *inferential* approach: *the target is now individualized by a logical formulae which can be deduced from the information associated to the* CG*-scene, but which can not be deduced from the information associated to the* CG*-scene without the target*. We believe it is important to draw attention to the deep role played by inference in addressing GRE in a CG framework, which provides a simple and effective mechanism for handling a more realistic setting than those used by the existing work in the field.

The aim of this paper is therefore to present a new and effective application of CGs in the area of Natural Languages Processing (NLP). This reveals also, some new interesting questions related to the combinatorial and algorithmic properties of CGs. For example, we found in a natural way, that the "eccentricity" of a concept node can be considered as its salience in the description provided by the CG. This can be used by a CG layout tool in order to enhance the visual quality of the picture, by placing "central concept nodes" in the middle of the picture. Also, we arrived at the notion of "non-ambiguous description" provided by a CG, that is a description in which no two concept nodes could be confused. Recognizing such a property of a CG is obviously important for the CG models of real world applications. This can be viewed as a certain discipline of modelling in an area which is sometimes dominated by rhetorical metaphors.

2 Conceptual Graphs (CGs)

2.1 Syntax

Here we discuss the (simple) conceptual graph (CG) model and explain how it can be used to formalise the information in a domain (or 'scene') such as Figure 1. In section 3 we show how the resulting CG-based representations can be used by a GRE algorithm that *refers* uniquely to objects in the scene.

The CG model (Sowa (1984)) is a logic-based KR formalism. Conceptual Graphs make a distinction between ontological (background) knowledge and factual knowledge. The ontological knowledge is represented in the support, which is encoded in hierarchies. The factual knowledge is represented by a labelled bipartite graph whose nodes are taken from the support. The two classes of partitions consist of concept nodes and relation nodes. Essentially, a CG is composed of a support (the concept / relation hierarchies), an ordered bipartite graph and a labelling on this graph which allows connecting the graph nodes with the support.

We consider here a simplified version of a support $S = (T_C, T_R, \mathcal{I})$, where: $(T_C, \leq)$ is a finite partially ordered set of *concept types*; $(T_R, \leq)$ is a partially ordered set of *relation types*, with a specified *arity*; $\mathcal{I}$ is a set of *individual markers*.
Formally (Chein and Mugnier (1992)), a (simple) CG is a triple CG$= [S, G, \lambda]$, where:

- S is a support;
- $G = (V_C, V_R, E)$ is an ordered bipartite graph ; $V = V_C \cup V_R$ is the node set of G, V_C is a finite nonempty set of *concept nodes*, V_R is a finite set of *relation nodes*; E is the set of edges $\{v_r, v_c\}$ where $v_r \in V_R$ and $v_c \in V_C$; the edges incident to each relation node are ordered and this ordering is represented by a positive integer label attached to the edge; if the edge $\{v_r, v_c\}$ is labelled i in this ordering then v_c is the i-neighbor of v_r and is denoted by $N^i_G(v_r)$;
- $\lambda : V \rightarrow S$ is a labelling function; if $v \in V_C$ then $\lambda(v) = (type_v, ref_v)$ where $type_v \in T_C$ and $ref_v \in \mathcal{I} \cup \{*\}$; if $r \in V_R$ then $\lambda(r) \in T_R$.

For simplicity we denote a conceptual graph CG$= [S, G, \lambda]$ by G, keeping support and labelling implicit. The order on $\lambda(v)$ preserves the (pair-wise extended) order on T_C (T_R), considers $\mathcal{I}$ elements mutually incomparable, and $* \geq i$ for each $i \in \mathcal{I}$. The fact that two concept labels with distinct individual markers is in concordance with the unique name assumption, that is, there is an unique name of naming a specific entity.

Consider the following KB described in Figure 1. The Krahmer et al. (2003) associated scene digraph is illustrated in Figure 2 and the CG scene graph description is given in Figure 3.

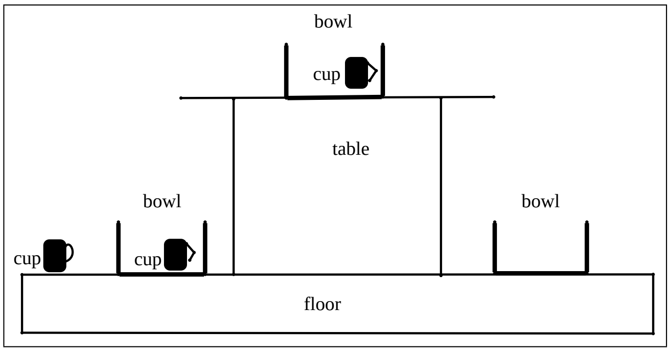

Figure 1. A scene

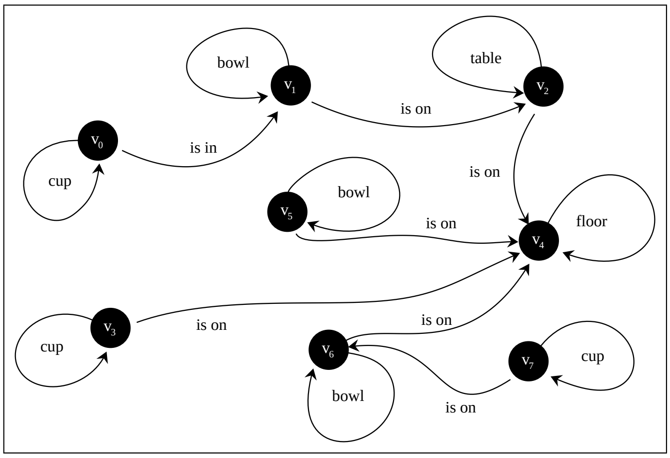

Figure 2. Krahmer et al. scene digraph

In Figure 3 the concept type hierarchy T_C of the support is depicted on the left. The factual information provided by Figure 1 is given by the labelled bipartite graph on the right. There are two kinds of nodes: rectangle nodes

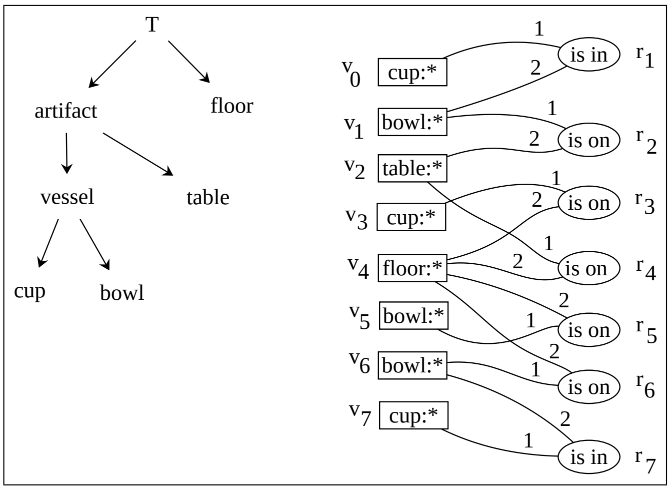

Figure 3. A CG-style scene graph

representing concepts (objects) and oval nodes representing relations between concepts. The former are called concept nodes and the second relation nodes. The labels r_i and v_i outside rectangles and ovals are only used for discussing the structure of the graph, they have no meaning. $\{v_0, \ldots, v_7\}$ are the concept nodes and $\{r_1, \ldots, r_7\}$ are the relation nodes. Each edge of the graph links a relation node to a concept node. The edges incident to a specific relation node are ordered and this ordering is represented by a positive integer label attached to the edge. For example, the two edges incident to the relation node r_1 are $\{r_1, v_0\}$, labelled 1 and $\{r_1, v_1\}$, labelled 2; we also say that v_0 is neighbor 1 of r_1 and v_1 is neighbor 2 of r_1.

In the digraphs of Krahmer et al. (2003), relations with more than two places are difficult to handle, but CGs can represent these naturally, because relation instances are *reified*. Consider that x gives a car y to a person z, and a ring u to v. Using CGs, this is modelled by considering two instances r_1 and r_2 of giving, each of which has a labelled arc to its three arguments. We note also that in the scene digraphs of Krahmer et al. (2003), object's attributes are encoded using labelled loops, and this can conduct to unpleasant complications of for the graphical representation. Using relation nodes of degree 1 is more expressive in the CG representation.

The label of a concept node (inside the rectangle) has two components: a concept type and either an individual marker or $*$, the *generic marker*. The concept node designates an entity of the type indicated by the first component. If the second component is $*$, this entity is arbitrary ; if it is an individual marker then the entity is specific. Intuitively, by using labels, CGs have associated, by definition, a "local" referential mechanism: each concept node refers to an entity belonging to the subset of the universe established by the interpretation of its type. In Figure 3 all concepts have generic markers and the nodes v_0, v_3 and v_7 designate three arbitrary objects of type *cup*, v_4 designates an arbitrary object of type *floor*, etc. For a relation node, the label inside the oval is a relation type from T_R. The arity of this relation type is equal to the number of vertices incident to the relation node r (denoted by $deg(r)$. Intuitively, this means that the objects designated by its concept node neighbours are in the relation designated by the label. In Figure 3 the relation node r_2 asserts that the bowl designated by v_1 is on the table designated by v_2.

Overall the conceptual graph in Figure 3 states that there is a floor on which there are a table, a cup and two bowls; on the table there is a a bowl and in this bowl there is a cup.

2.2 Formal Semantics of CGs

Usually, CGs are provided with a logical semantics via the function Φ, which associates to each CG a FOL formula (Sowa (1984)). If S is a support, a constant is associated to each individual marker, a unary predicate to each concept type and a n-ary predicate to each n-ary relation type. We assume that the name for each constant or predicate is the same as the corresponding element of the support. The partial orders specified in S are translated in a set of formulae $\Phi(S)$ by the following rules: if $t_1, t_2 \in T_C$ such that $t_1 \leq t_2$, then $\forall x(t_2(x) \rightarrow t_1(x))$ is added to $\Phi(S)$; if $t_1, t_2 \in T_R$, have arity k and $t_1 \leq t_2$, then $\forall x_1 \forall x_2 \ldots \forall x_k (t_2(x_1, x_2, \ldots, x_k) \rightarrow t_1(x_1, x_2, \ldots, x_k))$ is added to $\Phi(S)$.

If CG= $[S, G, \lambda]$ is a conceptual graph then a formula Φ(CG) is constructed as follows. To each concept vertex $v \in V_C$ a term a_v and a formula $\phi(v)$ are associated: if $\lambda(v) = (type_v, *)$ then $a_v = x_v$ (a logical variable) and if $\lambda(v) = (type_v, i_v)$, then $a_v = i_v$ (a logical constant); in both cases, $\phi(v) = type_v(a_v)$. To each relation vertex $r \in V_R$, with $\lambda(r) = type_r$ and $deg_G(r) = k$, the formula associated is $\phi(r) = type_r(a_{N^1_G(r)}, \ldots, a_{N^k_G(r)})$.

Φ(CG) is the existential closure of the conjunction of all formulas associated with the vertices of the graph. That is, if $V_C(*) = \{v_{i_1}, \ldots, v_{i_p}\}$ is the set of all concept vertices having generic markers, then Φ(CG)$= \exists v_1 \ldots \exists v_p (\wedge_{v \in V_C \cup V_R} \phi(v))$. If G is the graph in Figure 3, then
$\Phi(G) = \exists x_{v_0} \exists x_{v_1} \exists x_{v_2} \exists x_{v_3} \exists x_{v_4} \exists x_{v_5} \exists x_{v_6} \exists x_{v_7} [cup(x_{v_0}) \wedge bowl(x_{v_1}) \wedge table(x_{v_2}) \wedge cup(x_{v_3}) \wedge floor(x_{v_4}) \wedge bowl(x_{v_5}) \wedge bowl(x_{v_6}) \wedge cup(x_{v_7}) \wedge isin(x_{v_0}, x_{v_1}) \wedge ison(x_{v_1}, x_{v_2}) \wedge ison(x_{v_1}, x_{v_2}) \wedge ison(x_{v_3}, x_{v_4}) \wedge ison(x_{v_2}, x_{v_4}) \wedge ison(x_{v_5}, x_{v_4}) \wedge ison(x_{v_6}, x_{v_4}) \wedge isin(x_{v_7}, x_{v_6})]$.

If (G, λ_G) and (H, λ_H) are two CGs (defined on the same support S) then $G \geq H$ (G subsumes H) if there is a *projection* from G to H. A projection is a mapping π from the vertices set of G to the vertices set of H, which maps concept vertices of G into concept vertices of H, relation vertices of G into relation vertices of H, preserves adjacency (if the concept vertex v in V^G_C is the ith neighbour of relation vertex $r \in V^G_R$ then $\pi(v)$ is the ith neighbour of $\pi(r)$) and furthermore $\lambda_G(x) \geq \lambda_H(\pi(x))$ for each vertex x of G. A projection is a morphism between the corresponding bipartite graphs with the property that labels of images are decreased. $\Pi(G, H)$ denotes the set of all projections from G to H.

Informally $G \geq H$ means that if H holds then G holds too. This is motivated by the fact that the subsumption relation corresponds to deduction for the fragment of first order logic (FOL) associated to CGs. More precisely, if $G \geq H$ then $\Phi(S), \Phi(H) \models \Phi(G)$ (*soundness*) (Sowa (1984)). *Completeness* (if $\Phi(S), \Phi(H) \models \Phi(G)$ then $G \geq H$) only holds if the graph H is in *normal form*, i.e. if each individual marker appears at most once in concept node labels (Chein and Mugnier (1992)). Using only CGs in normal form is a natural condition for our GRE purposes and this will be assumed implicitly in the following.

For the GRE problem the following definitions are needed to rigorously identify a certain type of a subgraph. If $G = (V_C^G, V_R^G, E)$ is an ordered bipartite graph and $A \subseteq V_R^G$, then the *subgraph spanned by A in G* is the graph $[A]_G = (N_G(A), A, E')$ where $N_G(A)$ is the neighbour set of A in G, that is the set of all concept vertices with at least one neighbour in A, and E' is the set of edges of G connecting vertices from A to vertices from $N_G(A)$. It is easy to see that if G is a CG then the subgraph $[A]_G$ and the restriction of λ_G to its vertices is a CG too, the spanned conceptual subgraph of G. Clearly $[A]_G \geq G$ since the identity is a trivial projection from $[A]_G$ to G.

3 CGs for Generation of Referring Expressions

3.1 Stating the problem

Let us see how the GRE problem can be stated in terms of CGs.

Definition 1. *Let G be a CG and v_0 be a concept node in G. We define that a CG H (on the same support S as G)* **uniquely refers** *to v_0 in G if :*
$$H \geq G \text{ and } H \not\geq G - v_0.$$

Since projection is sound and complete with respect to Sowa's semantics Φ for (normal) CGs, it follows that H *uniquely refers* to v_0 in G if and only if $\Phi(S), \Phi(G) \models \Phi(H)$ and $\Phi(S), \Phi(G - v_0) \not\models \Phi(H)$. This intuitively means that H *uniquely refers* to v_0 in G if and only if the facts stated by H can be logically deduced from the facts stated by scene G, but this is no longer the case if the target v_0 is removed from the scene.

It is easy to see that if H uniquely refers to v_0 in G and H' is any subgraph of H such that $H' \not\geq G - v_0$, then H' also uniquely refers to v_0 in G. Clearly, in the GRE problem we will be interested in obtaining only minimal CGs H that uniquely refers to v_0 in G.

On the other hand, let us note that if H uniquely refers to v_0 in G, then there is π a projection from H to G (since $H \geq G$) and a concept node w in H such that $\pi(w) = v_0$ (otherwise, π is a projection from H to $G - v_0$). Hence, if $\pi(H)$ is the image of H, then $\pi(H)$ is a spanned subgraph of G namely, $[\pi(V_R^H)]_G$, containing v_0. Clearly, $\pi(H) \geq G$ (identity is an obvious projection) and, furthermore, $\pi(H) \not\geq G - v_0$ (if there is a projection π_1 from $\pi(H)$ to $G - v_0$

then $\pi_1 \circ \pi$ is a projection from H to $G - v_0$). Therefore, we have obtained that $\pi(H)$ uniquely refers to v_0 in G.

It follows that (analogous to Krahmer et al. 2003) in the GRE problem we can restrict only to referring graphs 'part of' the scene graph. It is possible to formulate GRE using only the combinatorial structure CG G and the vertex v_0.

Definition 2. *Let G be a* CG *and v_0 be a concept node in G.*
*A v_0-**referring subgraph of** G is the subgraph $G' = (\{v_0\}, \emptyset, \emptyset)$ or any spanned subgraph $G' = [A]_G$ containing v_0 (that is, $A \neq \emptyset$ and $v_0 \in N_G(A)$).*
*A v_0-referring subgraph $[A]_G$ is called v_0-**distinguishing** if $[A]_G \not\geq G - v_0$.*

It is not difficult to verify that a v_0-referring subgraph $[A]_G$ is v_0-distinguishing if and only if v_0 is a fixed point of each projection π from $[A]_G$ to G, that is $\pi(v_0) = v_0 \; \forall \pi \in \Pi([A]_G, G)$.

The GRE problem is now:

> **Instance:** CG$= [S, G, \lambda]$ a conceptual graph representation of the scene; v_0 a concept vertex of G.
> **Output:** $A \subseteq V_R$ such that $[A]_G$ is a v_0-distinguishing subgraph in CG, or the answer that *there is no v_0-distinguishing subgraph in cg.*

Example. Consider the scene described in Figure 3. $A = \emptyset$ is not a solution for the GRE instance (CG, $\{v_0\}$) since $G_1 = (\{v_0\}, \emptyset, \emptyset)$ can be projected to $(\{v_7\}, \emptyset, \emptyset)$ or $(\{v_3\}, \emptyset, \emptyset)$. However, $A = \{r_1, r_2\}$ is a valid output since $G_1 = [\{r_1, r_2\}]_G$ is a v_0-distinguishing subgraph. Note that the description of the entity represented by v_0 in G_1 has the intuitive meaning *the cup in the bowl on the table*, which does individuates this cup. In our inferential approach this holds since $\Phi(G_1) = \exists x_{v_0} \exists x_{v_1} \exists x_{v_2} (cup(x_{v_0}) \wedge bowl(x_{v_1}) \wedge table(x_{v_2}) \wedge isin(x_{v_0}, x_{v_1}) \wedge ison(x_{v_1}, x_{v_2}))$ can be deduced from $\Phi(G)$ but not from $\Phi(G - v_0)$.

If $G_1 = [A]_G$ is a v_0-distinguishing subgraph in CG, and if we denote by A' the relation nodes set of the connected component of G_1 containing v_0, then $[A']_G$ is a v_0-distinguishing subgraph in CG too. Hence, by the minimality assumption, we consider only connected v_0-distinguishing subgraphs. On the other hand, intuitively the existence of a v_0-distinguishing subgraph is assured only if the CG description of the scene has no ambiguities.

Theorem 1. *Let* (CG, $\{v_0\}$) *be a* GRE *instance. If $[A]_G$ is v_0-distinguishing then $[A']_G$ is v_0-distinguishing for each $A' \subseteq V_R^G$ such that $A \subseteq A'$.*

Proof. Indeed, since $A \subseteq A'$ and $v_0 \in N_G(A)$ it follows that $v_0 \in N_G(A')$, therefore $[A']_G$ is v_0-referring. If $[A']_G$ is not v_0-distinguishing then there is π a projection from $[A']_G$ to G such that $\pi(v_0) \neq v_0$. But then, π_A, the restriction of π to the subgraph $[A]_G$, has the same property, $\pi_A(v_0) \neq v_0$, contradicting the hypothesis that $[A]_G$ is v_0-distinguishing.
In particular, taking $A' = V_R$, we obtain:

Corollary 1. *There is a v_0-distinguishing subgraph in G iff $G \not\geq G - v_0$.*

Proof. If there is $[A]_G$ a v_0-distinguishing subgraph in G, then (since $A \subseteq V_R$ and $[V_R]_G = G$), by the above theorem, G is v_0-distinguishing and therefore

$G \not\geq G - v_0$. Conversely, if $G \not\geq G - v_0$ then it follows that G is a v_0-distinguishing subgraph.

A concept vertex v_0 which does not have a v_0-distinguishing subgraph is called an *undistinguishable concept vertex* in G. We say that a CG provides an *well-defined scene representation* if it contains no undistinguishable vertices. Testing if a given GRE instance defines such an ambiguous description is, by the above corollary, decidable.

Let $v_0 \in V_C$ be an arbitrary concept vertex. The set of concept vertices of G different from v_0, in which v_0 could be projected, is (by projection definition) contained in the set

$$Distractors^0(v_0) = \{w | w \in V_C - \{v_0\}, \lambda(v_0) \geq \lambda(w)\}.$$

Clearly, if $Distractors^0(v_0) = \emptyset$ then v_0 is implicitly distinguished by its label (type + referent), that is $(\{v_0\}, \emptyset, \emptyset)$ is a v_0-distinguishing subgraph.

Therefore we are interested in the existence of a v_0-distinguishing subgraph for concept vertices v_0 with $Distractors^0(v_0) \neq \emptyset$. In this case, if $N_G(v_0) = \emptyset$, clearly there is no v_0-distinguishing subgraph (the connected component containing the vertex v_0 of any spanned subgraph of G is the isolated vertex v_0). Hence we assume $N_G(v_0) \neq \emptyset$.

3.2 Complexity

Some of the main complexity results in GRE are presented in Dale and Reiter (1995). Among other things, these authors argue that the problem of finding a uniquely referring description that contains the minimum number of properties (henceforth, a Shortest Description) is NP-complete, although other versions of GRE can be solved in polynomial or even linear time. As we have argued, CG allows a substantial generalisation of the GRE problem. We proved in Croitoru and van Deemter (2006) that this generalisation does not affect the theoretical complexity of finding Shortest Descriptions. More precisely, we proved that the decision problem associated with minimum cover (Garey and Johnson (1979)) can be polynomially reduced to the problem of finding a concise distinguishing subgraph. If this later problem is

Shortest Description

Instance: G a CG such that $d_G(r) = 1$, for each relation node $r \in V_R$; a vertex $v_0 \in V_C$; s a positive integer.

Question: Is there a v_0-distinguishing subgraph $[A]_G$ such that $|A| \leq s$?

then we proved (Croitoru and van Deemter (2006)):

Theorem 2. Shortest Description *is* **NP**-*complete.*

Note that in the above problem we considered the simple case when all relation vertices $r \in V_R^G$ unary. In other words, G is a disjoint union of stars centered in each concept vertex. Intuitively, this means that each object designated by a concept vertex in the scene represented by G is characterized by its label (type and reference) and by some other possible attributes (properties) and each $r \in V_R^G$ designates an unary relation. This is the classical framework of the GRE problem, enhanced with the consideration of basic object properties (the types) and the existence of a hierarchy between attributes.

In this particular case, if $N_G(v_0) = \{r_1, \ldots, r_p\}$ ($p \geq 1$) (the properties of the concept designated by v_0) then for each $r_i \in N_G(v_0)$ we can consider:
$X_i := \{w | w \in Distractors^0(v_0)$ such that there is no $r \in N_G(w)$ with $\lambda(r_i) \geq \lambda(r)\}$.

In words, X_i is the set of v_0-distractors which will be removed if r_i would be included as a single relation vertex of a v_0-distinguishing subgraph (since there is no $r \in N_G(w)$ such that $\lambda(r_i) \geq \lambda(r)$ it follows that there is no projection π of the subgraph $[r_i]_G$ to G such that $\pi(v_0) = w$).

The proof of the above theorem is based on the following lemma which we have proved in Croitoru and van Deemter (2006):

Lemma 1. *There is a v_0-distinguishing subgraph in G iff:*

$$\cup_{i=1}^{p} X_i = Distractors^0(v_0).$$

To summarize, if all relation vertices have degree 1, *deciding if a vertex v_0 admits a v_0-distinguishing subgraph can be done in polynomial time.*
However, the above lemma shows that $[A]_G$ is a v_0-distinguishing subgraph if and only if $A \subseteq N_G(v_0)$ and $\cup_{r_i \in A} X_i = Distractors^0(v_0)$. Therefore the problem of finding a v_0-distinguishing subgraph with a **minimum number** of vertices (e.g., Dale and Reiter 1995) is reduced to the problem of finding a minimum cover of the set $Distractors^0(v_0)$ with elements from $X_1, \ldots, X_p$, which is an NP-hard problem.

3.3 A simple GRE algorithm

In the general case, each object in the scene represented by G is characterized by its label (type and reference), by some other possible attributes (properties) and also by its relations with other objects, expressed via relation nodes of arity ≥ 2. In this case, if v_0 an arbitrary concept node, it is possible to have vertices in $Distractors^0(v_0)$ which cannot be distinguished from v_0 using individual relation neighbors but which could be removed by collective relation neighbors. Let us consider the scene described in Figure 4 :

Note that relation labels are assumed to be incomparable. Clearly, $N_G(v_0) = \{r_1, r_2\}$ and $Distractors^0(v_0) = \{v_2, v_4\}$. The vertex v_4 can be removed by r_1 (v_4 has no relation neighbor with a label at least *know*) and by r_2 (despite of the existence of a relation neighbor r_5 labelled *is near*, v_4 is the second neighbor of r_5; v_0 is the first neighbor of r_2). The vertex v_2 cannot be removed by r_1 ($[r_1]_G \geq [r_3]_G$)) and by $r_2([r_2]_G \geq [r_4]_G)$, but $\{r_1, r_2\}$ destroys v_2 (there is no projection of $[\{r_1, r_2\}]_G$ mapping v_0 to v_2 and in the same time mapping v_1 to a common neighbor of r_3 and r_4).

This example shows a way to obtain an algorithm for constructing a v_0-distinguishing subgraph in general.

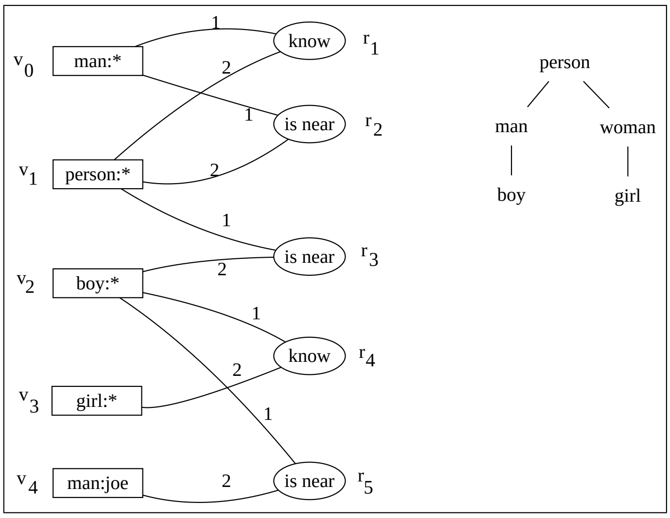

Figure 4. Scene Illustration

For an arbitrary concept vertex v_0, let us denote $N^0(v_0) = \emptyset$, $N^1(v_0) := N_G(v_0)$ and for $i \geq 2$, $N^i(v_0) = N_G(N_G(N^{i-1}(v_0)))$. Clearly, since G is finite, there is $k \geq 1$ such that $N^i(v_0) = N^k(v_0)$ for each $i \geq k$ ($N^k(v_0)$ is the relation nodes set of the connected component of G which contains v_0). This parameter is called the *eccentricity* of v_0 and is denoted $ecc(v_0)$. The Figure 5 illustrates the construction of this sequence of relation nodes.

The basic idea is to test successively if the above constructed relation neighbors sets of v_0 destroy the set $Distractors^0(v_0)$.

We can consider, inductively, distractors of higher order for a vertex v_0. We will use the following notation: if G is a CG containing a vertex v and H is a CG containing a vertex w then $G \geq_{v \to w} H$ means that there is a projection π from G to H such that $\pi(v) = w$. Now, $Distractors^i(v_0)$ are defined by:

$Distractors^0(v_0) = \{w | w \in V_C - \{v_0\}, \lambda(v_0) \geq \lambda(w)\}$, and, for each $i = 1, ecc(v_0)$,
$Distractors^i(v_0) = \{w | w \in Distractors^{i-1}(v_0), [N^i(v_0)]_G \geq_{v_0 \to w} [N^i(w)]_G\}$.

Note that $Distractors^0(v_0) \supseteq Distractors^1(v_0) \supseteq \ldots \supseteq Distractors^{ecc(v_0)}(v_0)$. However, only the set $Distractors^0(v_0)$ can be computed in polynomial time. The set $Distractors^i(v_0)$, $i \geq 1$, contains the vertices w from the previous set, $Distractors^{i-1}(v_0)$, which cannot be destroyed by $N^i(v_0)$ and this means that we need to test if $[N^i(v_0)]_G \geq [N^i(w)]_G$. But the last test is, in general, non-polynomial.

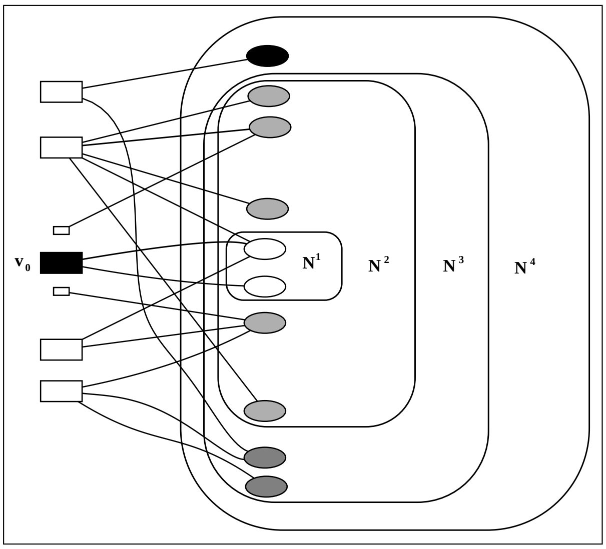

Figure 5. Successive Relation Neighbors Sets

Theorem 3. *Let $(G, \{v_0\})$ be a GRE instance, and let i_0 be the first $i \in \{0, \ldots, ecc(v_0)\}$ such that $Distractors^i(v_0) = \emptyset$. If i_0 exists then $[N^i(v_0)]_G$ is a v_0-distinguishing subgraph, otherwise v_0 is an undistinguishing vertex.*

Proof. We can suppose that G is connected. Therefore $[N^{ecc(v_0)}(v_0)]_G = G$. Also, if $Distractors^0(v_0) = \emptyset$ then the theorem holds trivially. Inductively, we can prove that

(∗) *If $Distractors^i(v_0) \neq \emptyset$, then $[N^i(v_0)]_G$ is not a v_0-distinguishing subgraph.*

Using the theorem 1 we obtain from (∗) that there is no v_0-distinguishing subgraphs in $[N^i(v_0)]_G$. Therefore, if $Distractors^{ecc(v_0)}(v_0) \neq \emptyset$ then there is no v_0-distinguishing subgraphs in $[N^{ecc(v_0)}(v_0)]_G = G$.
It follows also from (∗) that for each $w \in Distractors^i(v_0)$ there is a projection $\pi \in \Pi_{[N^i(v_0)]_G \to G}$ such that $\pi(v) = w$. If i_0 is the first index i such that $Distractors^i(v_0) = \emptyset$, then, clearly, $[N^{i_0}(v_0)]_G$ is a v_0-distinguishing subgraph.

Therefore, the theorem is completely proved if we show that (∗) holds. But this follows easily from the definition of the sets $Distractors^i(v_0)$, using an inductive argument.

The above theorem basically defines a breath first search algorithm for finding a v_0-distinguishing subgraph which can be described as follows.

```
Input: CG = [S, G, λ] a CG representation of the scene; v_0 a concept vertex of G.
Output: A ⊆ V_R such that [A]_G is a v_0-distinguishing subgraph in G,
        or the answer that there is no v_0-distinguishing subgraph in G.
{   D ← ∅
    for each w ∈ V_C^G − {v_0} do
        if λ(v_0) ≥ λ(w) then  D ← D ∪ {w}
    N ← N_G(v_0); finished ← false
    while D ≠ ∅ and not finished do
        {   for  each w ∈ D do
                if not [N]_G ≥_{v_0→w} G then D ← D − {w}
            if N = N_G(N_G(N)) then finished ← true
                      else N ← N_G(N_G(N))
        }
    if D = ∅ then return N
        else return there is no v_0-distinguishing subgraph in G.
}
```

4 Conclusions

This paper presents a new and useful application of CGs in the area of Natural Languages Processing (NLP). Employing Conceptual Graphs as the underlying formalism to the Generation of Referring Expressions (GRE) allows a new, rigorous and semantically rich approach to GRE: the intended referent is indentified by a combination of facts that can be deduced in its presence but not if it would be absent. More precisely, using CG to formalise GRE means that we benefit from:

- The existence of a support. CGs make possible the systematic use of a set of "ontological commitments" for the knowledge base. A support, of course, can be shared between many KBs.
- A properly-defined formal semantics, reflecting the precise meaning of the graphs and their support, and including a general treatment of n-place relations.
- Projection as an inferential mechanism. Projection replaces the purely graph-theoretical notion of a subgraph isomorphism by a proper logical concept (since projection is sound and complete with respect to subsumption). Optimized algorithms (for example Croitoru and Compatangelo (2004)) can be used to improve the new GRE algorithm developed in the present paper.

At the same time, applying conceptual graphs to address the GRE problem raises novel interesting questions related to the combinatorial and algorithmic properties of CGs:

- The "eccentricity" of a concept node which can be used by a CG layout tool in order to enhance the visual quality of the picture.
- "Non-ambiguous descriptions", descriptions in which no two concept nodes could be confused, is obviously important for the CG models of real world applications.

To conclude, the deep role played by inference in addressing GRE in a CG framework provides a simple and effective mechanism to model the way humans refer to objects in a rich inferential setting which has never been used by existing work in the field.

5 References

[Chein and Mugnier (1992)] Chein, M., and Mugnier, M.-L. Conceptual graphs: Fundamental notions. *Revue d'Intelligence Artificielle* **6**(4):365

[Croitoru and Compatangelo (2004)] Croitoru, M. and Compatangelo, E. A combinatorial approach to conceptual graph projection checking. *In Proc. of the 24th International Conf. of the Brit. Comp. Soc. Spec. Group on Artif. Intell. (AI 2004)*, 130 143. Springer.

[Croitoru and van Deemter (2006)] Croitoru M. and Kees van Deemter. A Conceptual Graph approach for the generation of referring expressions. *In Proc. of the 20th International Joint Conference on Artificial Intelligence*, IJCAI 2007. To appear.

[Dale and Haddock (1991)] Robbert Dale and Nickolas Haddock. Generating referring expressions containing relations. *Procs. of the 5th Conference of the European Chapter of the ACL*, EACL-91.

[Dale and Reiter (1995)] Robbert Dale and Ehud Reiter. Computational Interpretations of the Gricean Maximes in the Generation of Referring Expressions. *Cognitive Science* **18**: 233-263.

[Garey and Johnson (1979)] Garey, M. and Johnson D. Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman. [Horacek (2004)] Horacek, H. On referring to Sets of Objects Naturally. *Proc. of Third Int. Conf. on Natural Language Generation* (INLG-2004), p.70-79, Brockenhurst, UK.

[Krahmer et al (2003)]. Krahmer,E. van Erk,S. and Verleg,A. Graph-based Generation of Referring Expressions. To appear in *Computational Linguistics* 29 (1):53-72.

[Sowa (1984)] Sowa, J. F. Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley.

[van Deemter (2002)] van Deemter, K. Generating Referring Expressions: Boolean Extensions of the Incremental Algorithm. *Computational Linguistics* **28** (1), pp.37-52. March 2002.

[van Deemter and Krahmer] (IN PRESS). K. van Deemter and E.Krahmer. Graphs and Booleans: on the generation of referring expressions". To appear in "*Computing Meaning*, Volume 3", "Studies in Linguistics and Philosophy" series, Kluwer, Dordrecht.