A Conceptual Graph Approach to the Generation of Referring Expressions

Madalina Croitoru
IAM, Electronics and Computer Science,
University of Southampton, UK

Kees van Deemter
Department of Computing Science,
University of Aberdeen, UK

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 rigorous, semantically rich, approach to GRE. A number of advantages over existing work are discussed. The new framework is also used to revisit existing complexity results in a fully rigorous way, showing that the expressive power of CGs does not increase the theoretical complexity of GRE.

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, as when we say 'the wooden bookcase', 'the cup on the table', and so on, to single out the object in question. A GRE program 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. We do not argue for one particular flavour of GRE, but show how different flavours can be 'implemented' using CGs.

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. 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 address, while maintaining all the other advantages of the approach of Krahmer et al.

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 CG 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 CG 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). CGs have been used in computational linguistics before (e.g. Nicolov et al. 1995) but, to the best of our knowledge, never to GRE.

2 Conceptual Graphs (CGs)

2.1 Syntax

Here we introduce the 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 repre-

sented 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$= [S, G, \lambda]$:

- 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 $v_c \in V_C$;

- $\lambda : V \rightarrow S$ 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 λ preserves the (pair-wise extended) order on T_C (T_R) and considers $\mathcal{I}$ elements mutually incomparable.

Consider the following KB described in Figure 1. The CG scene graph description is given in Figure 2.
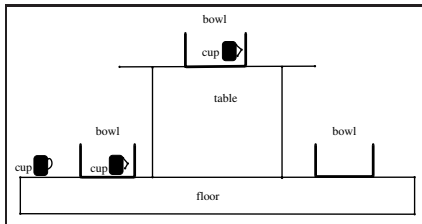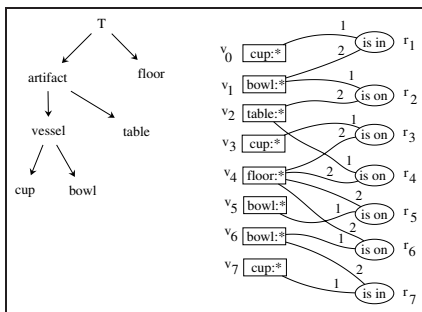

Figure 1: A scene

Figure 2: A CG-style scene graph

In Figure 2 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 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 graphs 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. This is modelled by using two instances r_1 and r_2 of giving, each of which has a labelled arc to its three arguments.

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 an arbitrary one; if it is an individual marker then the entity is a specific one. In Figure 2 all concepts have generic markers and the nodes v_0, v_3 and v_7 designate three (different) 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 $deg(r)$). The objects designated by its concept node neighbours are in the relation designated by the label. In Figure 2 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 2 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

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 $\Phi(CG)$ 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 concept vertex $r \in V_R$, with $\lambda(r) = type_r$ and $deg_G(r) = k$, the formula associated is $\phi(r) = type_r(a_{N_G^1(r)}, \ldots, a_{N_G^k(r)})$. The existential

closure of the conjunction of all formulas associated to the vertices of G is $\Phi(\text{CG})$. That is, if $V_C(*) = \{v_{i_1}, \ldots, v_{i_p}\}$ is the set of all concept vertices having generic markers, then $\Phi(\text{CG}) = \exists v_1 \ldots v_p(\wedge_{v \in V_C \cup V_R}\phi(v))$.

If G is the graph in Figure 2, then (using only one quantifier) $\Phi(G) = \exists x_{v_0}x_{v_1}x_{v_2}x_{v_3}x_{v_4}x_{v_5}x_{v_6}x_{v_7}(cup(x_{v_0}) \wedge bowl(x_{v_1}) \wedge table(x_{v_2}) \wedge cup(x_{v_3}) \wedge floor(x_{v_4}) \wedge bowl(x_{v_5}) \wedge bowl(x_{v_6}) \wedge cup(x_{v_7}) \wedge isin(x_{v_0}, x_{v_1}) \wedge ison(x_{v_1}, x_{v_2}) \wedge ison(x_{v_1}, x_{v_2}) \wedge ison(x_{v_3}, x_{v_4}) \wedge ison(x_{v_2}, x_{v_4}) \wedge ison(x_{v_5}, x_{v_4}) \wedge ison(x_{v_6}, x_{v_4}) \wedge isin(x_{v_7}, x_{v_6})$.

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_C^G is the ith neighbour of relation vertex $r \in V_R^G$ 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. Informally $G \geq H$ means that if H holds then G holds too. $\Pi(G, H)$ denotes the set of all projections from G to H.

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 $N_G(A)$. 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 CG. Given a CG G, a concept node v_0 in G, a CG H and a concept node w in H, we define that the pair (w, H) refers to the pair (v_0, G) if there is a projection π from H to G such that $\pi(w) = v_0$. Furthermore, (w, H) *uniquely refers* to (v_0, G) if (w, H) refers to (v_0, G) and there is no concept node v in G different from v_0 such that (w, H) refers to (v, G). The GRE problem may now be stated as follows: *given a CG G and a concept node v_0 in G, find a pair (w, H) such that (w, H) uniquely refers to (v_0, G).*

But (analogous to Krahmer et al. 2003), the GRE problem means that we are interested in referring graphs 'part of' the scene graph. By the above definition, if (w, H) uniquely refers to (v_0, G), then there is a projection π from H to G such that $\pi(w) = v_0$. 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 such that there is no projection π_1 from $\pi(H)$ to G such that $\pi_1(v_0) \neq v_0$. Therefore it is possible to formulate GRE using only the combinatorial structure CG G and the vertex v_0.

More precisely, any spanned subgraph $G' = [A]_G$ containing v_0 (that is, if $A \neq \emptyset$ then $v_0 \in N_G(A)$, and if $A = \emptyset$ then $G' = (\{v_0\}, \emptyset, \emptyset)$) is called a v_0-*referring subgraph of*

G. A v_0-referring subgraph $[A]_G$ is called v_0-*distinguishing* 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 2. $A = \emptyset$ 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 intensional description of the entity (cup) represented by v_0 in G_1, $\Phi_{G_1, v_0}(x) = \exists x_{v_1}\exists x_{v_2}(bowl(x_{v_1}) \wedge table(x_{v_2}) \wedge isin(x, x_{v_1}) \wedge ison(x_{v_1}, x_{v_2}))$ has the intuitive meaning *the cup in the bowl on the table*, which clearly individualizes this cup in the set of the three cups in the scene).

Note that if $G_1 = [A]_G$ is a v_0-distinguishing subgraph in CG then if we denote by A' the relation node set of the connected component of G_1 containing v_0, then $[A']_G$ is a v_0-distinguishing subgraph in CG too. For brevity, 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 omitted for reasons of space.)* In particular, taking $A' = V_R$, we obtain:

Corollary 1 *There is a v_0 distinguishing subgraph in G iff there is no projection π from G to G such that $\pi(v_0) \neq v_0$.*

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.

3.2 Complexity results

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. In what follows, we shall show that this generalisation does not affect the theoretical complexity of finding Shortest Descriptions.[1]

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

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

---

[1]To the best of our knowledge this is the first formal proof of the NP- completeness of any GRE problem.

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 $N_G(v_0) \neq \emptyset$.

A simple case considering all relation vertices $r \in V_R^G$ unary. This means that 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. Even with these enhancements we will show that the GRE problem complexity results remain similar.

Indeed, 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 $r_i \in N_G(v_0)$ let: $X_i := \{w | w \in Distractors^0(v_0)$ 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 $\pi(v_0) = w$).

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

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

Proof. If there is a v_0-distinguishing subgraph $[A]_G$, then we can suppose that $A \subseteq N_G(v_0) = \{r_1, \ldots, r_p\}$ (because we can suppose that $[A]_G$ is connected). Suppose that there is $w \in Distractors^0(v_0) \backslash \cup_{i=1}^p X_i$. By the definition of the sets X_i it follows that for each $r_i \in A$ there is $r_i' \in N_G(w)$ such that $\lambda(r_i) \geq \lambda(r_i')$. If we take $\pi(v_0) = w$ and $\pi(r_i) = r_i'$ for each $r_i \in A$, then π is a projection of $[A]_G$ to G with $\pi(v_0) = w \neq v_0$, contradicting the hypothesis that $[A]_G$ is a v_0-distinguishing subgraph.

Conversely, if $\cup_{i=1}^p X_i = Distractors^0(v_0)$, then taking $A = N_G(v_0) = \{r_1, \ldots, r_p\}$ $[A]_G$ is a v_0-distinguishing subgraph. Indeed, if there is π a projection of $[A]_G$ to G such that $\pi(v_0) = w \neq v_0$, then for each $r_i \in A$ the edge $v_0 r_i$ is projected in some edge wr_i' such that $\lambda(v_0) \geq \lambda(w)$ and $\lambda(r_i) \geq \lambda(r_i')$. This means that $w \in Distractors^0(v_0) \backslash \cup_{i=1}^p X_i$ is a contradiction.

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 proof 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.

We now prove that the decision problem associated with minimum cover can be polynomially reduced to the problem of finding a concise distinguishing subgraph. Let us consider the following two decision problems:

Minimum Cover
Instance: $O = \{o_1, \ldots, o_n\}$ a finite nonempty set;
$\mathcal{S} = (S_1, \ldots, S_p)$ a finite family of subsets of O;
q a positive integer.
Question: Are there $S_{i_1}, \ldots, S_{i_t} \in \mathcal{S}$ such that
$\cup_{j=1}^t S_{i_j} = O$ and $t \leq q$?

Shortest Description
Instance: $S\text{CG} = (G, S)$ a conceptual graph 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 $|A| \leq s$?

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

Proof. We have shown that checking if a guessed $A \subseteq V_R$ such that $[A]_G$ is v_0-distinguishing can be done in polynomial time (we need to test if $v_0 \in N_G(A)$ and $\cup_{r_i \in A} X_i = Distractors^0(v_0)$; all sets involved can be computed in polynomial time). Hence **Shortest Description** belongs to **NP**.

We outline a polynomial reduction of **Minimum Cover** to **Shortest Description**. Let $O = \{o_1, \ldots, o_n\}$, $\mathcal{S} = (S_1, \ldots, S_p)$ and q an instance of **Minimum Cover**. Let us consider the bipartite graph G with $V_C = \{o_0, o_1, \ldots, o_n\}$. The label of each concept vertex is $(\top, *)$. The relation vertices set of G is initialized as $V_C = \{r_{ij} | i \in \{0, \ldots, n\}, j \in \{0, \ldots, p\}\}$, where each vertex r_{ij} is a neighbor of the vertex o_i (all edges $o_i r_{ij}$ are added to G). The type of the vertex r_{ij} is a_i. The final CG G is obtained by deleting some relation vertices using the following rule: for each i and j, if $o_i \in S_j$ then r_{ij} is deleted from G (together with the edge $o_i r_{ij}$). Clearly the graph G is constructed from the instance of the **Minimum Cover** problem in polynomial time. The desired instance of the **Shortest Description** problem is obtained by taking as input this graph, vertex o_0 and $s = q$. An illustration of this construction for the instance $O = \{o_1, o_2, o_3\}$, $\mathcal{S} = (S_1 = \{o_1, o_2\}, S_2 = \{o_1, o_3\}, S_3 = \{o_2, o_3\})$ and $q = 1$, of **Minimum Cover** is given in Figure 3:
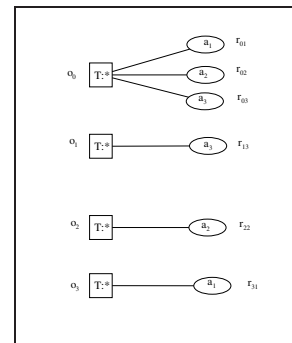

Figure 3: Minimum Cover Illustration

It is easy to see that there is no o_0-distinguishing subgraph with at most $s = 1$ relation vertices and also there is no cover of O with at most $q = 1$ members of $\mathcal{S}$. If in the instance of the **Minimum Cover** problem we have $q = 2$ then the answer is yes since (S_1, S_2), (S_1, S_3) or (S_2, S_3) are all covering of O with 2 members of $\mathcal{S}$. The corresponding instance of the **Shortest Description** problem (the same graph and $s = 2$) also returns "yes"; since $[r_{01}, r_{02}]_G$, $[r_{01}, r_{03}]_G$ or $[r_{02}, r_{03}]_G$ are all o_0-distinguishing subgraphs with 2 relation vertices.

Returning to the proof, we must show that in the constructed conceptual graph there is an o_0-distinguishing subgraph with at most s relation vertices if and only if there is a covering of O with at most $q = s$ members from $\mathcal{S}$. But this is an easy consequence of the observation that the above construction is such that the sets X_i (destroying the distractors by the relation vertices r_{0i}) are exactly the sets $S_i \in \mathcal{S}$.

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 :
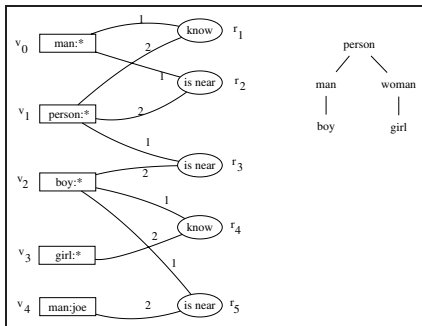

Figure 4: Scene Illustration

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 suggests an algorithm for constructing a v_0-distinguishing subgraph in general. 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 $i \geq k$. This parameter is called the *eccentricity* of v_0

and is denoted $ecc(v_0)$. We can inductively construct distractors of higher order for a vertex v_0, as follows: If we consider $Distractors^0(v_0) = \{w|w \in V_C - \{v_0\}, \lambda(v_0) \geq \lambda(w)\}$, then for each $i = 1, ecc(v_0)$, $Distractors^i(v_0) = \{w|w \in Distractors^{i-1}(v_0), [N^i(v_0)]_G \geq [N^i(w)]_G\}$.

Theorem 3 *Let* $(CG, \{v_0\})$ *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 undistinguishable vertex.*

Proof. The theorem clearly holds due to the definition of the $Distractors$ sets.

This theorem basically defines a breadth-first search algorithm for finding a v_0 distinguishing subgraph.

4 Generalisations and extensions

Apart from the benefits shown throughout the paper, using CGs for representing GRE means that a number of interesting and useful extensions can be developed. This section sketches two kinds of extensions: the addition of inferential rules (to increase reasoning power), and the addition of negation and full quantification (to increase expressive power).

4.1 Adding inferential Rules

An important further generalisation of the GRE problem can be obtained by bringing inferential *rules* to bear, as an integral part of CG (Baget and Mugnier 2002). If S is the support of the CG representing the scene, a rule defined on S is any CG, H, over the support S, and having specified a bipartition $(Hyp, Conc)$ of its set of relation nodes V_R^H. The subgraph of H spanned by Hyp, $[Hyp]_H$ is called the *hypothesis of the rule* H, and the subgraph spanned by $Conc$, $[Conc]_H$, is the *conclusion of the rule* H. *Applying a rule* H *to a* CG G means to find a projection π from $[Hyp]_H$ to G, to add a disjoint copy of $[Conc]_H$ to G, and finally to identify in this graph each concept node $v \in V_C^{[Conc]_H} \cap V_C^{[Hyp]_H}$ to $\pi(v)$, its image by π. The new CG obtained, G', is called an *immediate derivation* of G by the application of rule H following π. Let $\mathcal{R}$ a set of rules defined on S and G a CG over S. Then $G, \mathcal{R}$ derives a CG G' if there exists a sequence of immediate derivations leading to G' by applications of rules in $\mathcal{R}$.

Let us consider again the scene represented in Figure 4. If we want to express that the conceptual relation *near* is symmetric, then we add the rule H described in Figure 5:
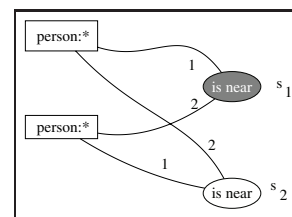

Figure 5: Rule Illustration

Here the Hyp set contains only the "gray" relation node s_1 and the set $Conc$ is the singleton set $\{s_2\}$. Without this rule, the vertex in Figure 4 $v_4 \in Distractors^0(v_0)$ can be

removed by by adding r_2 to a v_0-referring subgraph (despite of the existence of a relation neighbor r_5 labelled *is near*, v_4 is the second neighbor of r_5 while v_0 is the first neighbor of r_2). Considering the rule H, which intuitively asserts that "if a person a *isnear* a person b then b *isnear* a, too", v_4 can not be removed by adding r_2 to a v_0-referring subgraph.

We have therefore an interesting extension of the basic GRE problem, which can be formally stated as follows. Let S, G and $\mathcal{R}$ a scene representation, and v_0 a concept node in G. A *derived v_0-distinguishing subgraph* is any v_0-distinguishing subgraph of some derived CG G' from G and $\mathcal{R}$. The extended GRE problem asks to find such a *derived v_0-distinguishing subgraph* or the conclusion that it does not exists for a specified S, G and $\mathcal{R}$ scene representation.

4.2 Increasing expressive power

Like all other GRE algorithms that we are aware of, the CGs discussed so far are essentially limited to the expression of atomic facts. In the CG framework, however, various extensions are naturally forthcoming.

Negation. Explicit negation can be added to the KB, which is an interesting way to allow GRE to produce descriptions involving the absence of a property (cf. van Deemter and Krahmer (in press) for alternative approaches).

Let $A, F \subseteq V_R$ such that $A \cap F = \emptyset$. The pair $([A]_G, [F]_G)$ is called *a v_0-distinguishing pair of subgraphs* if $v_0 \in N_G(A)$ (that is, $[A]_G$ is a v_0-referring subgraph), $v_0 \notin N_G(F)$ and each projection π from $[A]_G$ to G such that $\pi(v_0) \neq v_0$ can be extended to a projection π' from $[A \cup F]_G$ to G (that is, $\pi'|_{[A]_G} = \pi|_{[A]_G}$) such that we have $\pi'(v_0) \in N_G(\pi'(F))$.

The set $F \subset V_R$ (of *forbidden* relation nodes) is used to kill each concept node from the set $Confusable^A(v_0)$, where $Confusable^A(v_0) = \{w | w \in V_C - \{v_0\}, \exists \pi \in \Pi([A]_G, G) \text{ such that } \pi(v_0) = w\}$.

In the example in Figure 4 $(\{r_1\}, \{r_5\})$ is a v_0-distinguishing pair of subgraphs, with the intuitive meaning *the man knowing a person and which is not near Joe*. Similarly, for the scene represented in figure (2), $(\{r_5\}, \{r_7\})$ is a v_5-distinguishing pair of subgraphs, with the intuitive meaning: *the bowl on the floor, not containing a cup*.

Clearly, if $[A]_G$ is a v_0-distinguishing subgraph, then $([A]_G, [F]_G)$ is a v_0-distinguishing pair of subgraphs, for each $F \subseteq V_R$ such that $A \cap F = \emptyset$. The basic GRE problem can be extended by replacing the output requirement with:

Output: $A, F \subseteq V_R$ such that $([A]_G, [F]_G)$ is a v_0-distinguishing pair of subgraphs in SG, or the answer that *there is no v_0-distinguishing pair of subgraphs in SG*.

Full quantification. Suppose our CG-based KB expressed the (non-atomic) proposition that *every table has exactly one cup on it*; this plus the fact that there is only one wooden table (for example) would be sufficient to justify a description like 'the cup on the wooden table', even if the KB does not say which table this is. Extensions of this kind would enable GRE programs to model human communication far more extensively than has so far been possible.

5 Conclusions

Using CG to formalise GRE means that we can 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. Further extensions of the expressive power of the formalism are possible, as we have seen (section 4).

- 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 CG projection have been studied in Croitoru and Compatangelo (2004). This inferential mechanism can be further enhanced by the addition of 'axiomatic' rules (section 4).

6 References

Baget and Mugnier (2002) Baget, J.-F., and Mugnier, M.-L. Extensions of Simple Conceptual Graphs: the Complexity of Rules and Constraints. Jour. of Artif. Intell. Res. 16:425 465.

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.

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.

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.

Nicolov et al. (1995) Nicolov, N., C.Mellish, and G.Ritchie. Sentence Generation from Conceptual Graphs. In *Conceptual Structures: Applications, Implementation and Theory*. (LNAI 954), p.74-88. Berlin: Springer.

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

van Deemter (2002) 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.