A Simulation of Co-identity with Rules in Simple and Nested Graphs

Jean-François Baget

LIRMM
161, rue Ada, 34392 Montpellier - FRANCE
baget@lirmm.fr

Abstract. Equality of markers and co-reference links have always been a convenient way to denote that two concept nodes represent the same entity in conceptual graphs. This is the underlying cause of counterexamples to projection completeness with respect to these graphs FOL semantics. Several algorithms and semantics have been proposed to achieve completeness, but they do not always suit an application specific needs. In this paper, I propose to represent identity by relation nodes, which are first-class objects of the model, and I show that conceptual graphs rules can be used to represent and simulate reasonings defined by various semantics assigned to identity, be it in the case of simple or nested graphs. The interest of this method is that we can refine these rules to manage the identity needed by the application.

Introduction

Identity of concept nodes in conceptual graphs [10] has always been a crucial problem for completeness of projection with respect to these graphs logical semantics. According to the widely accepted FOL semantics Φ, two concept nodes sharing the same individual marker represent the same entity. Moreover, the necessity to express that some generic concept nodes represent the same entity has early led to the adoption of co-reference links which, however useful in simple graphs (SGs), proved to be of uttermost importance for positive nested graphs (NGs). These two representations of identity can be seen as an equivalence relation on concept nodes, which has been called *co-identity* [8].

But Chein and Mugnier [4], as well as Ghosh and Wuvongse [3] gave counterexamples to projection completeness with respect to Φ in the case of SGs, and proposed a normalization procedure to achieve completeness. In the case of NGs, Simonet has extended the normality condition into a k-normality condition [8], which achieves completeness with respect to a natural extension of the semantics Φ. Instead of forcing projection to conform to the semantics Φ, Simonet has proposed the semantics Ψ [7], defined as well for SGs as for NGs, which describes more precisely the graph as well as the projection mechanism. Projection is sound and complete with respect to Ψ, without any condition.

The problem is that the adoption of one semantics or another is a definitive and exclusive choice with respect to the reasonings allowed in the model. In the semantics Ψ, co-identical concept nodes are considered as different points of view on the same entity, and information on one of these nodes is considered to be meaningful only for the particular point of view represented by this node. On the other hand, the semantics Φ considers co-identical concept nodes as partial and complementary representations of an entity: information linked to one of the co-identical nodes is shared by all others. Now suppose we want to represent the sentence: *"A dark-haired man is looking at a photograph representing himself"*. We can represent it in a graph where two co-referent concept nodes represent the man, the man x looking at the photograph, and the man y represented by the photograph. We could want to deduce from this knowledge that y is dark-haired, but we do not want to deduce that he is looking at the photograph. With the semantics Φ, the two deductions are possible, and none of them are using Ψ. Since co-reference links are not first-class objects of the CG model, we cannot implement the desired behavior without rewriting the projection algorithm, with *ad hoc* procedures regarding to *sharable* or *non-sharable* relation types.

Instead, I propose in this paper to represent identity by relation nodes, which are first-class objects of the CG model. The semantics that can be assigned to identity will be simulated with conceptual graph rules [6]. I present the specific sets of rules which describe co-identity in SGs as defined by the semantics Φ and Ψ. In order to extend this work to NGs, I use a generalization of the NGs, that I call *boxed graphs*, where concept nodes in different contexts can be linked by relation nodes. These rules could easily be refined, using types restrictions, to conform to a specific semantics.

1 Co-Identity in Simple Graphs

The formal model for SGs used in this paper is similar to the one presented in [1]. Completeness can be achieved by normalization to conform to the semantics Φ [4], or without restriction with respect to Ψ [7], [9].

1.1 The Model

We consider a *knowledge base* where the basic ontological knowledge is coded in a *support* and the asserted facts are coded in a *simple graph* (SG). The basic operation for reasonings is the graph morphism classically called *projection*.

A support $\mathcal{S} = (T_C, T_R, \nu, \mathcal{I}, \tau)$ is defined by two partially ordered sets, the *concept types* T_C, the *relation types* T_R, a set of *individual markers* $\mathcal{I}$, and two mappings, the *valence*[1] ν and the *conformity relation* τ. T_C has a greatest element, $\top_C$. T_R is partitioned into partially ordered subsets $T_R^i, i \in \{1, \dots, m\}$ of relation types of *valence* i, whose greatest element is $\top_R^i$. $\mathcal{I} \cup \{*\}$ is partially ordered, has a greatest element, the *generic marker* $*$, all others being pairwise incomparable. τ assigns a concept type to each individual marker.

A SG G, on a support $\mathcal{S}$, is a bipartite labelled multigraph (V_C, V_R, E, l), where the nodes in V_C are the *concept nodes*, those in V_R are the *relation nodes*.

[1] It is a simplified version of signature, which does not assign a maximum type to the arguments of a relation

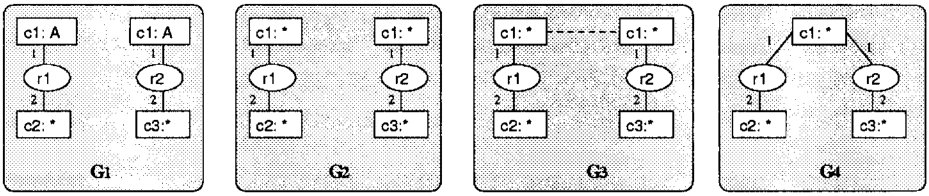

Fig. 1. *Very* simple graphs ...

l labels each concept node by a concept type and an individual or generic marker, called its referent. It is required that the type of an individual node c is exactly $\tau(\text{ref}(c))$. l labels each relation node r of degree i by a relation type of valence i. Edges of E incident on such a node are numbered from 1 to i.

SGs can be extended to simple conceptual graphs with co-reference links. A SG^{ref} is defined as a SG which is added an equivalence relation *co-ref* defined on the set of generic concept nodes. In order to avoid fusionning problems during normalization, only generic concept nodes having the same type can be declared co-referent. The equivalence relation ***co-ident*** is defined on all concept nodes by: $\forall c, c' \in V_C, \text{co-ident}(c, c') \Leftrightarrow \text{co-ref}(c, c')$ or $\text{ref}(c) = \text{ref}(c') \in \mathcal{I}$ The relation *co-ref* is traditionally represented in the drawing of the graph by a dotted line called a co-reference link[2] between concept nodes (see graph G_3 in fig. 1).

Let H and G be two SGs defined on a common support $\mathcal{S}$. A *projection* Π from H to G is given by a mapping from the nodes of H to the nodes of G, preserving edges and their numbering, and respecting the order defined on labels. In the case of SG^{ref}s, two co-referent generic concept nodes can only be projected into the same co-identity class. In fig. 1, the graph G_3 can be projected into the graph G_1, but G_3 cannot be projected into G_2. Instead of the traditional notation $H \geq G$, I note $H \sqsubseteq G$ if there exists a projection from H to G. This points out that all information in H is contained in G.

1.2 The Semantics Φ and SGs Normal Form

SGs and SG^{ref}s are assigned FOL semantics, Φ being the one presented in [10]. The support determines the vocabulary used in formulas, and is associated a formula $\Phi(\mathcal{S})$, translating the partial orderings on T_C and T_R. Each *co-ref* class in a graph G is assigned a unique variable. $\Phi(G)$ is the existential closure of the conjunction of the atoms associated with all nodes of the graph: $\forall c \in V_C$, $C(t)$ is defined by the unary predicate C assigned to its type and the term t, which is the variable associated to its co-reference class if c is generic, the constant assigned to its marker otherwise; $\forall r \in V_R$ such that $\nu(\text{degree}(r)) = i$, $R(t_1, ..., t_i)$ is defined by the i-ary predicate R assigned to its type, and the terms t_k associated with the k^{th} neighbors of r. The graph G_3 in fig. 1 is interpreted by:

$$\Phi(G_3) = \exists xyz\ (\text{c1}(x) \wedge \text{c1}(x) \wedge \text{c2}(y) \wedge \text{c3}(z) \wedge \text{r1}(x, y) \wedge \text{r2}(x, z))$$

[2] A co-reference class of n concept nodes can be represented by $n - 1$ such links.

Projection is sound with respect to Φ [10]. Concerning completeness, Chein and Mugnier[4], and Ghosh and Wuvongse [3], pointed out that two co-identical concept nodes are assigned exactly the same atom by Φ, whereas projection treats them as two distinct nodes. By example, in fig. 1, assuming that all types involved are incomparable, there is no projection from G_4 into any of the other graphs, though $\Phi(G_4)$ can be deduced from $\Phi(G_1)$ or $\Phi(G_3)$.

A SG or SG^{ref} is said in *normal form* if the co-identity classes are restricted to the trivial ones: they only contain a single concept node. A graph G is put into its normal form $\mathcal{N}_{\mathcal{F}}(G)$ by fusionning all its co-identical nodes. In fig. 1, $G_4 = \mathcal{N}_{\mathcal{F}}(G_3)$. This normalization does not issue any problem since all concept nodes in the same co-identical class have the same type[3] and the same marker.

Theorem 1 (Soundness and Completeness [4]). *Let $\mathcal{S}$ be a support, and H and G be two* SG*s (or* SG^{ref}*s) defined on $\mathcal{S}$. Then $H \sqsubseteq \mathcal{N}_{\mathcal{F}}(G)$ iff $\Phi(\mathcal{S}), \Phi(G) \models \Phi(H)$.*

1.3 Adapting the Semantics to Projection: the Ψ Solution

The semantics Ψ was introduced in order to translate the whole information encoded in a graph. In this semantics, two terms are associated to each concept node. As in Φ, the first term represents the co-identity class of the node. The second term is a variable representing the node itself. Let G be a SG or SG^{ref} defined on a support $\mathcal{S}$. $\Psi(\mathcal{S})$ is obtained in the same way as $\Phi(\mathcal{S})$, excepted that the predicates associated with concept types become binary. $\Psi(G)$ is the existential closure of the conjunction of atoms obtained in the following way: If Φ associates an atom $C(t)$ to a concept node, then Ψ associates it an atom $C(t, v)$ where the term v is a new variable representing the node itself. Each relation node of degree i is assigned an atom $R(v_1, ..., v_i)$, where R is the predicate of arity i assigned to its type, and the v_k are the second terms associated with the k^{th} neighbors of the node. The graph G_3 in fig. 1 is interpreted by:

$$\Psi(G_3) = \exists xyzabcd\ (\mathrm{c1}(x, a) \wedge \mathrm{c1}(x, b) \wedge \mathrm{c2}(y, c) \wedge \mathrm{c3}(z, d) \wedge \mathrm{r1}(a, c) \wedge \mathrm{r2}(b, d))$$

Theorem 2 (Soundness and Completeness [7], [9]). *Let $\mathcal{S}$ be a support, and H and G be two* SG*s or* SG^{ref}*s defined on $\mathcal{S}$. Then $H \sqsubseteq G$ iff $\Psi(\mathcal{S}), \Psi(G) \models \Psi(H)$.*

The co-reference as described by the semantics Ψ implements what I call a *"weak co-identity"*. Thanks to the additional constraint defined on projection, concept nodes belonging to the same co-reference class are assured to be projected into the same co-identity class. But the information on a concept node stays the property of this particular node. These nodes represent different and independent points of view on the same entity. On the other hand, the "normalization + projection" process implements a *"strong co-identity"*. Thanks to normalization, the whole information on a specific concept node is shared by every other node of its co-identity class.

[3] An explicit constraint for co-referent generic nodes, a consequence of the conformity relation for individual ones.

2 Implementing Co-Identity with Simple Graph Rules

As shown by the example in introduction, *"strong co-identity"* can be too strong, in the sense that too much information can be deduced from the graph. But *"weak co-identity"*, though conforming exactly to the projection mechanism, is definitively too weak. I propose to represent co-identity by relation nodes in the graph, and to simulate its desired behavior by conceptual graphs rules [6], [5]. I first give a modified (though exactly equivalent) version of rules, then show how they can be used to simulate weak and strong co-identities.

2.1 Conceptual Graphs Rules

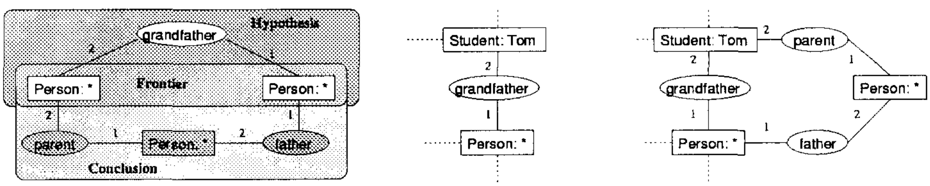

Fig. 2. A rule and its application

A simple graph rule (SGR) is defined as a simple graph which is associated a mapping *color* from $V_C \cup V_R$ into $\{0,1\}$. A node v such that $\text{color}(v) = 0$ is called an *hypothesis* node, otherwise it is called a *conclusion* node. An hypothesis node having a conclusion node as a neighbor is called a *frontier* node. The subgraph of a SGR generated by the set of all hypothesis nodes is called the hypothesis of the rule. It must be a syntactically valid SG (see fig. 2). The subgraph generated by frontier nodes and conclusion nodes is called the conclusion.

Let $\mathcal{S}$ be a support, G a SG defined on $\mathcal{S}$, and R a SGR defined on $\mathcal{S}$. The rule R is *applicable* to G if there exists a projection from the hypothesis of R to G. In this case, the application of R on G following a projection Π is a SG G' constructed by making the disjoint union of G with the conclusion of R, then by fusionning[4] frontier nodes of the conclusion with the corresponding nodes in the projection of the hypothesis. The rule R in fig. 2 can be read *"If a person has a grandfather, then he has a parent whose father is this grandfather"*.

Let us consider a knowledge base $\mathcal{KB} = \{\mathcal{S}, \mathcal{R}\}$, where $\mathcal{S}$ is a support, and $\mathcal{R}$ a set of SGRs defined on $\mathcal{S}$. Let G and G' be two SGs defined on $\mathcal{S}$. We note $G \vdash G'$, if G' is obtained by an application of a rule of $\mathcal{R}$ on G. We say that G *derives* G' and we note $G \Vdash G'$, where $\Vdash$ is the reflexo-transitive closure of $\vdash$. We say that H is *deduced* from G and note $G \models H$ if there exists G' such that $G \Vdash G'$ and $H \sqsubseteq G'$.

[4] This fusion is not exactly similar to the one used in the normalization process, since projection can transform the label of a frontier node into a more specific one. The most specific label is kept for the resulting node.

Rules are given FOL semantics by extending Φ in the following way: each node is assigned the same atom as the one in the SG interpretation. $\Phi(R)$ is the universal closure of the formula $\Phi_H(R) \rightarrow \Phi_C(R)$ where $\Phi_H(R)$ is the conjunction of the atoms associated to the hypothesis nodes, and $\Phi_C(R)$ is obtained by existentially quantifying the variables of the conjunction of the other atoms, when these variables do not appear in $\Phi_H(R)$. The semantics Ψ can be extended to SGRs in the same way. The rule R in fig. 2 can be interpreted by the formulas:

$$\Phi(R) = \forall xy((\mathrm{Pe}(x) \wedge \mathrm{Pe}(y) \wedge \mathrm{gr}(x,y)) \rightarrow (\exists z(\mathrm{Pe}(z) \wedge \mathrm{pa}(z,y) \wedge \mathrm{fa}(x,z))))$$
$$\Psi(R) = \forall xyab((\mathrm{Pe}(x,a) \wedge \mathrm{Pe}(y,b) \wedge \mathrm{gr}(a,b)) \rightarrow (\exists zc(\mathrm{Pe}(z,c) \wedge \mathrm{pa}(c,b) \wedge \mathrm{fa}(a,c))))$$

Deduction is sound with respect to Φ [10]. To achieve completeness, we define a *normalizing derivation*. We note $G \vdash_{\mathcal{F}} G'$, when G' is the normal form of an application of a rule on G. It defines a derivation $\Vdash_{\mathcal{F}}$ and a deduction $\models_{\mathcal{F}}$.

Theorem 3 (Soundness and Completeness [6]). *Let $KB = \{\mathcal{S}, \mathcal{R}\}$ be a knowledge base, and G and H two* SG*s or* SG^{ref}*s defined on $\mathcal{S}$. Then $\mathcal{N}_{\mathcal{F}}(G) \models_{\mathcal{F}} H$ iff $\Phi(\mathcal{S}), \Phi(\mathcal{R}), \Phi(G) \models \Phi(H)$.*

Theorem 4 (Soundness and Completeness). *Let $KB = \{\mathcal{S}, \mathcal{R}\}$ be a knowledge base, and G and H two* SG*s or* SG^{ref}*s defined on $\mathcal{S}$. Then $G \models H$ iff $\Psi(\mathcal{S}), \Psi(\mathcal{R}), \Psi(G) \models \Psi(H)$.*

Proof. Thanks to completeness of projection with respect to the semantics Ψ, the proof is the same as the one given in [6] for the semantics Φ. □

2.2 The *Weak Co-Identity* Relation

Let us now represent co-identity by a new relation typed `co-ident` and associated rules. Given a support $\mathcal{S}$, $\mathcal{S}^{\mathcal{I}}$ is obtained by adding a new binary relation type `co-ident` and a new greatest element to $T^2_{\mathcal{R}}$, which covers both `co-ident` and $\top^2_{\mathcal{R}}$. Let $\mathcal{R}_{\mathcal{W}}$ be the set of rules defined in fig. 3. The first three rules indicate that `co-ident` is an equivalence relation, the last one is a set of rules, one for each individual marker in $\mathcal{S}$. These rules are obtained by replacing `Marker` and `Type` by m and $\tau(m)$, $\forall m \in \mathcal{I}$. Note that, since markers are not first-class objects of the graph, we cannot express that two nodes sharing the same individual marker represent the same entity in a single rule. The modified support and this set of rules define a knowledge base $\mathcal{KB}^{\mathcal{W}}(\mathcal{S}) = \{\mathcal{S}^{\mathcal{I}}, \mathcal{R}_{\mathcal{W}}\}$.

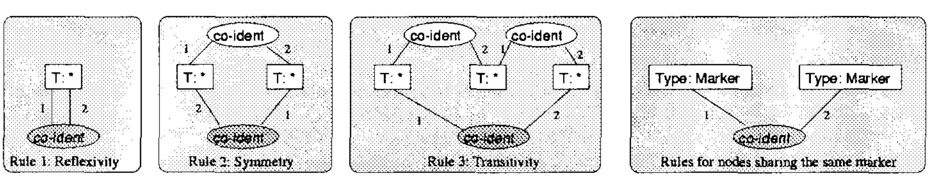

Fig. 3. Rules for weak co-identity

Proposition 1 (Equivalence). *Let $\mathcal{S}$ be a support, and G and H be two* SG^{ref}*s defined on $\mathcal{S}$. Let G' and H' be the* SG*s defined on $\mathcal{KB}^{\mathcal{W}}(\mathcal{S})$, obtained by replacing all co-reference links in G and H by relation nodes typed* `co-ident`. *Then $G' \vDash H'$ iff $\Psi(\mathcal{S}), \Psi(G) \vDash \Psi(H)$.*

Proof. We obtain a SG G'' by doing a complete irredundant expansion of G' using the rules in $\mathcal{R}^{\mathcal{W}}$. This operation is finite, since we create at most $|V_C(G)|^2$ relation nodes. Two concept nodes of G are co-identical iff there exists a relation node typed `co-ident` between these nodes in G''. □

2.3 The *Strong Co-Identity* Relation

Let $\mathcal{S}$ be a support. To simulate normalization, we now consider $\mathcal{KB}^{\mathcal{S}}(\mathcal{S}) = \{\mathcal{S}^{\mathcal{I}}, \mathcal{R}^{\mathcal{W}} \cup \mathcal{R}^{\mathcal{S}}\}$, where $\mathcal{R}^{\mathcal{S}}$ is the set of rules defined in fig. 4. There are in $\mathcal{R}^{\mathcal{S}}$ i rules of the form given in fig. 4 for each relation type of valence i. They are obtained by replacing the node typed `relation` by this relation type. These rules express that if a concept node C is linked by a relation node R to a concept node C', then every node indicated *co-ident* to C must be linked by R to C'.

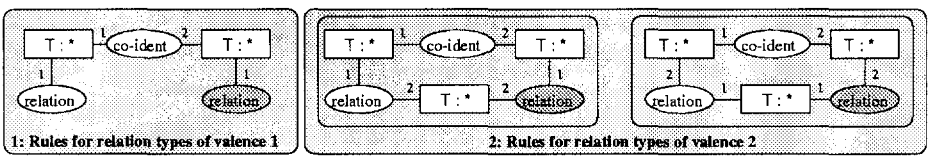

Fig. 4. Rules for strong co-identity

Proposition 2 (Equivalence). *Let $\mathcal{S}$ be a support, and G and H be two* SG^{ref}*s defined on $\mathcal{S}$. Let $G' = \vartheta(G)$ and $H' = \vartheta(H)$ be the* SG*s defined on $\mathcal{KB}^{\mathcal{S}}(\mathcal{S})$ obtained by replacing all co-reference links in G and H by relation nodes typed* `co-ident`. *Then $G' \vDash H'$ iff $\Phi(\mathcal{S}), \Phi(G) \vDash \Phi(H)$.*

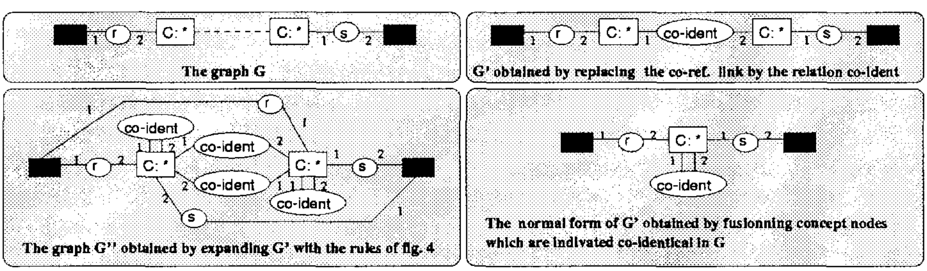

Fig. 5. The expansion of G' is equivalent to its normal form

Proof. The complete irredundant expansion G'' of G' is equivalent to $\vartheta(\mathcal{N}_{\mathcal{F}}(G'))$ (i.e. $G'' \sqsubseteq \vartheta(\mathcal{N}_{\mathcal{F}}(G')) \sqsubseteq G''$). This is proved by showing that co-identical nodes in G'' can be projected into the node C resulting from their fusion in $\mathcal{N}_{\mathcal{F}}(G')$, and that C can be projected into any of the co-identical nodes in G'' (fig. 5). □

3 From Nested Graphs to Boxed Graphs

The nested graphs (NGs) model allows to associate any concept node an internal description in the form of a NG. The formal model I use for NGs or NG^{ref}s is the one presented in [1] or [9]. Projection in this model is sound and complete without any restriction with respect to a natural extension of Ψ, but we have to consider a k-normality condition when using the semantics Φ [1], [9]. In order to extend the previous treatment of identity to NGs, I introduce the *boxed graphs* (BGs) model. BGs are a a generalization of NGs which allow relation nodes to link concept nodes in different descriptions. Moreover, I show that these boxed graphs are a "high-level representation" of a particular class of SGs, which are used to define BG rules.

3.1 Nested Graphs, k-Normality and the semantics Ψ

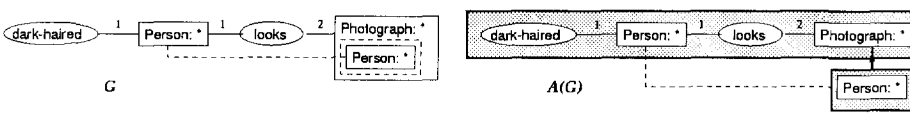

Fig. 6. A nested graph and its associated rooted tree

A NG is defined on a support identical to the one defined for SGs. A basic NG is obtained from a SG by adding to the label of each concept node a third field, called the *description* of the node, which is equal to ** (the empty description). A NG is obtained from a basic NG by replacing some of its descriptions by a NG. A NG^{ref} is a NG which is added an equivalence relation *co-ref* on the set of all its generic concept nodes, this relation is extended to *co-ident* as in SGs. Any NG is associated a rooted tree (fig. 6) whose nodes are the SGs used in its construction and edges (c, G) indicate that the concept node c is described by the NG whose root is G. Projection in the NG model can be defined on this tree: let H and G be two NGs, and $\mathcal{A}(H), \mathcal{A}(G)$ be their associated rooted trees. A projection from H to G is given by the projections (in the sense of simple graphs) of all nodes of $\mathcal{A}(H)$ into nodes of $\mathcal{A}(G)$ such that the root of $\mathcal{A}(H)$ is projected into the root of $\mathcal{A}(G)$, and the root of the description of a concept node can only be projected into the root of the description of its projection. Constraints on co-identical nodes must also be respected in the case of NG^{ref}s.

The semantics Φ and Ψ are extended to NG^{ref}s by associating another term to the atoms interpreting each node of the graph: n-ary predicates become $(n+1)$-ary. Every node of G in a node $\mathcal{X}$ of the rooted tree is associated the same additional term: the constant ρ if $\mathcal{X}$ is the root of $\mathcal{A}(G)$, otherwise the term identifying the concept node C such that $\mathcal{X}$ is the root of the description of C (the only term for Φ, the unique variable associated with the node in Ψ). By example, the graph G in fig. 6 is interpreted by:

$$\Phi(G) = \exists xy\ (\text{Pe}(x,\rho) \wedge \text{Ph}(y,\rho) \wedge \text{Pe}(x,y) \wedge \text{dh}(x,\rho) \wedge \text{lo}(x,y,\rho))$$
$$\Psi(G) = \exists xyabc\ (\text{Pe}(x,a,\rho) \wedge \text{Ph}(y,b,\rho) \wedge \text{Pe}(x,c,b) \wedge \text{dh}(a,\rho) \wedge \text{lo}(a,b,\rho))$$

Theorem 5 (Soundness and Completeness [7], [9]). *Let $\mathcal{S}$ be a support, G and H be two* NG*s or* NG^{ref}*s. Then $H \sqsubseteq G$ iff $\Psi(\mathcal{S}), \Psi(G) \models \Psi(H)$*

In order to achieve completeness with respect to the semantics Φ, Simonet has defined a normal form for NG^{ref}s, which is not always possible to compute, and a k-normal form, that I briefly recall here. A NG^{ref} G is said in k-normal form if every node of $\mathcal{A}(G)$ is a SG^{ref}in normal form, and for every concept node C such that its depth in $\mathcal{A}(G)$ is less than k, if C is co-identical to a concept node C' whose description is not empty, then the root of the description of C' must be exactly equivalent[5] to the root of the description of C. Putting a graph G into its k-normal form $\mathcal{N_F}^k(G)$ (by normalizing every node of $\mathcal{A}(G)$, and copying the roots of the descriptions of co-identical concept node as long as required by the k-normal form) does not change the semantics $\Phi(G)$.

Theorem 6 (Soundness and Completeness [8], [9]). *Let $\mathcal{S}$ be a support, G and H be two* NG*s or* NG^{ref}*s, and $k \geq depth(\mathcal{A}(H))$ be a number. Then $H \sqsubseteq \mathcal{N_F}^k(G)$ iff $\Phi(\mathcal{S}), \Phi(G) \models \Phi(H)$*

3.2 Boxed Graphs

The main difference between NGs and *boxed graphs* (BGs) is that there can be relation nodes linking concept nodes which belong to different descriptions.

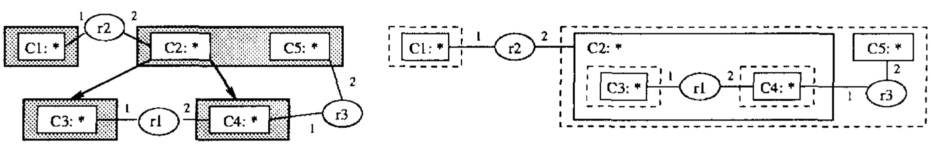

Fig. 7. A boxed graph and usual graphical representation

Definition 1 (Boxed Graphs). *Let $\mathcal{S}$ be a support, as defined for* SG*s. A* boxed graph *G is defined as a simple graph which is added a partition of V_C into* boxes *$\mathcal{B} = \{B_1, \ldots, B_k\}$, and a partial mapping* desc *from $\mathcal{B}$ into V_C, such that the oriented graph $\mathcal{A}(G) = \{\mathcal{B}, E\}$ defined by $(B_i, B_j) \in E$ iff $\exists x \in B_i, x = desc(B_j)$ is a collection of rooted trees.*

Boxes and *desc* are used to translate the nesting relation, and can be represented in the drawing of the graph by dotted rectangles drawn inside the concept node they describe (see fig. 7). Though this representation is similar to the one adopted for NGs, important differences must be noted. First, there can be several boxes describing the same concept node. Next, though there is a unique root in the rooted tree associated to NGs, the boxes for which *desc* is not defined are multiple *root boxes*. This will be of great utility for defining BG rules. Finally, there can be relation nodes linking concept nodes which belong to different boxes. This property will be used to represent the relation *co-ident* by relation nodes.

[5] Not only there is a projection from one to the other, and *vice-versa*, but these projections must map any generic concept node C into a generic concept node co-referent to C.

Projection on BGs is defined as on SGs, but two concept nodes in the same box must be projected into the same box ("a box is projected into a box"), and a box describing a concept node C must be projected into a box describing the image of C. Note the difference with projection as defined on NGs: a root box does not need to be projected into a root box.

A BG is said *nested* if it has only one root box, each concept node is described by at most one box, and there is no relation node linking concept nodes in different boxes. We can associate to each "nested BG" G the NG having the same graphic representation as G, and conversely, to each NG we associate a boxed graph which has the property of being nested. A *rooted projection* on a boxed graph is a projection such that root boxes can only be projected into root boxes. The proof of the next proposition is immediate, and it justifies the assertion that BGs are a generalization of NGs.

Proposition 3. *Let G and H be two* NG*s defined on $\mathcal{S}$, and G' and H' be their associated* BG*s. Then there is a projection from H to G iff there is a rooted projection from H' to G'.*

3.3 Associated Simple Graphs

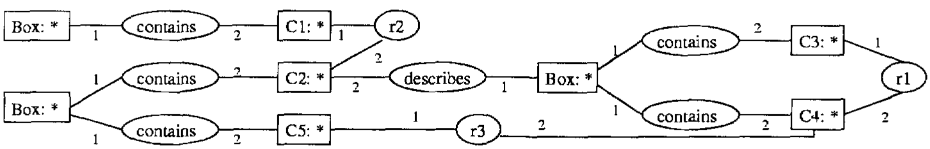

Fig. 8. The simple graph associated to the boxed graph of fig. 7

A problem with BGs is that boxes are not first-class objects of the model, but an assertion on some nodes of the graph which is represented in the drawing of the graph: boxes cannot be manipulated by rules. I define here the associated SG of a BG, where boxes and *desc* are reified into nodes of the SG.

Let S be a support, and G be a BG. The graph $\mathcal{S}_G(G)$ associated to G is a SG defined on a support $\mathcal{S}_G(\mathcal{S})$: T_C is added two types of concepts, $\top\top_C$, and *Box*, such that $\top\top_C$ covers $\top_C$ and *Box*. T_R is added three binary relation types, $\top\top^2_R$, *describes* and *contains*, such that $\top\top^2_R$ covers $\top^2_R$, *describes* and *contains*. $\mathcal{S}_G(G)$ is the SG obtained by adding a generic concept node typed *Box* for every box in G, linking this node by a relation node typed *contains* to every concept node belonging to this box, and if this box describes a concept node C, linking the node typed *Box* to C by a relation node typed *describes*. The BG in fig. 7 is associated the SG in fig. 8.

Proposition 4. *Let $\mathcal{S}$ be a support, and G and H be two boxed graphs defined on $\mathcal{S}$. Let $\mathcal{S}_G(\mathcal{S})$ be the support obtained from $\mathcal{S}$ as indicated above, and $\mathcal{S}_G(G)$ and $\mathcal{S}_G(H)$ the* SG*s defined on $\mathcal{S}_G(\mathcal{S})$, associated with G and H. Then $H \sqsubseteq G$ iff $\mathcal{S}_G(H) \sqsubseteq \mathcal{S}_G(G)$.*

Again, the proof is immediate. I will now consider BGs (and NGs, thanks to prop. 3) as a high-level representation for a particular class of SGs whose support includes the types *Box*, *describes*, and *contains*. I will use indifferently the BG representation or the SG representation for these graphs. In particular, I will use this SG representation to define *boxed graphs rules*. A BG rule is defined as a SG rule such that its hypothesis can be associated to a BG, the rule as a whole can be associated to a BG, and its subgraph generated by nodes typed *Box*, *describes*, and *contains* (the rooted forest representing nesting levels) is such that no conclusion node stands between an hypothesis node and its root. The consequence of this restriction is that BG rules only derive valid BGs. The result of the application of a BG rule R on a BG G is the boxed representation of the SG G' obtained by applying R (in a SGR sense) to the SG representation of G.

The semantics $\Psi_{\mathcal{B}}$ associated to a BG G or a BG rule R can then be defined as the semantics Ψ associated to $\mathcal{S}_{\mathcal{G}}(G)$, or R. Thanks to th. 6 and prop. 4, the proof of the following theorem is immediate.

Theorem 7 (Soundness and Completeness). *Let $KB = \{\mathcal{S}, \mathcal{R}\}$ be a knowledge base, where $\mathcal{R}$ is a set of* BG *rules, and G and H be two* BG*s. Then $G \vDash H$ iff $\Psi(\mathcal{S}), \Psi_{\mathcal{B}}(\mathcal{R}), \Psi_{\mathcal{B}}(G) \vDash \Psi_{\mathcal{B}}(H)$*

4 Rules Simulating Co-Identity in Boxed Graphs

I will now present the rules simulating the semantics Ψ and Φ in the particular class of boxed graphs (with *co-ident* relation nodes) corresponding to NG^{ref}s. Assigning a semantics to co-identity by extending Φ or Ψ to any BG is beyond the scope of this paper.

Let G be a NG^{ref}, defined on S. It is still required that only concept nodes sharing the same label can be declared co-identical. The BG $\mathcal{B}_{\mathcal{G}}(G)$ associated to G is defined on a support $\mathcal{S}^{\mathcal{I}}$ obtained by adding the *co-ident* relation type in $\mathcal{S}$, and $\mathcal{B}_{\mathcal{G}}(G)$ is represented by a graph obtained by replacing every co-reference link in G by a relation node typed *co-ident*. As these relation nodes can link concept nodes in different boxes, these *nested*^{ref} BGs are not nested BGs.

As I will now work only with nested *nested*^{ref} BGs, and in order to present more intuitive BG rules and equivalence results, I will use a slightly modified version of the associated SGs (note that these modifications only concern BGs, and not BG rules). First, in order to simulate a *rooted projection*, concept nodes typed *Box* representing root boxes will be labelled with the individual marker ρ. Next, for every concept node C such that there is no box describing C, we add a concept node typed *Box*, linked to C by a relation node typed *describes*. This feature will be used when defining rules in such a way that applying a rule does not create more than a box describing a single concept node.

4.1 The Semantics Ψ

Rules simulating co-identity in SG^{ref}s are updated to conform to the boxed graphs syntax. The set of rules $\mathcal{R}_{\mathcal{B}}^{\mathcal{W}}$ is obtained from the rules presented in fig. 9 in the same way as in sec. 2.2. Since these rules only add *co-ident* relation nodes, they generate only nested^{ref} BGs.

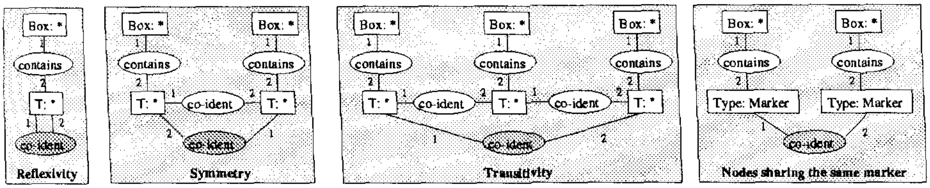

Fig. 9. Rules for weak-co-identity in boxed graphs

Proposition 5 (Equivalence). *Let $\mathcal{S}$ be a support, and G and H two* NG*s or* NG^{ref}*s defined on $\mathcal{S}$. Let G' and H' be their associated* BG*s, defined on the knowledge base $\mathcal{KB}_B^{\mathcal{W}}(\mathcal{S}) = \{\mathcal{S}^{\mathcal{I}}, \mathcal{R}_B^{\mathcal{W}}\}$. Then $G' \vDash H'$ iff $\Psi(\mathcal{S}), \Psi(G) \vDash \Psi(H)$.*

Proof. We must check that the *co-ident* classes can be computed regardless of the boxes containing the nodes. This can be done since the multiple roots in the hypothesis of the rules can be projected into any box of a given graph. □

4.2 The Semantics Φ

To simulate the semantics Φ in NG^{ref}s, I present a set of BG rules which mimic the operations used to put a graph into its k-normal form. The set of rules $\mathcal{R}_B^{\mathcal{W}}$ presented in fig. 9 will be used to generate all *co-ident* relation nodes.

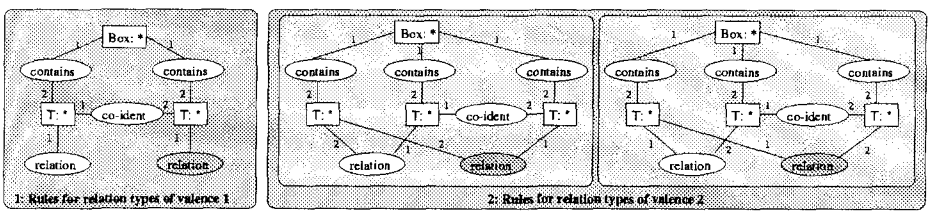

Fig. 10. Rules normalizing SGs in the same boxes

The set of rules $\mathcal{R}_B^{\mathcal{S}1}$ presented in fig. 10 is a slightly updated version of the rules in fig. 4 used for simple graphs. Let G be a nested^{ref} BG, and G' its associated NG^{ref}. The application of these rules on G mimics normalization on every node of $\mathcal{A}(G')$. See that, as these rules only create relation nodes that link concept nodes in the same box, these rules only generate nested^{ref} BGs.

In order to simulate that two co-identical concept nodes must be described by the same box, the method used until now (duplicating the relation nodes) would create graphs which are not boxed. The set of rules $\mathcal{R}_B^{\mathcal{S}2}$ in fig. 11 simulates a recursive copy of the contents of this description. Assuming that there are in $\mathcal{S}$ n_1 concept types and n_2 individual markers, the first rule drawn in fig. 11 represents $n_1 + n_2$ rules, the first n_1 being obtained by replacing Marker and Type by $*$ and t, $\forall t \leq \top_C$, the other n_2 by replacing them by m and $\tau(m), \forall m \in \mathcal{I}$. The second rule drawn also represents a set of rules, one for each relation type in $\mathcal{S}$. These rules simulate the fact that the descriptions of co-identical concept nodes must be exactly equivalent. I will now consider the set of rules $\mathcal{R}_B^{\mathcal{S}} = \mathcal{R}_B^{\mathcal{W}} \cup \mathcal{R}_B^{\mathcal{S}1} \cup \mathcal{R}_B^{\mathcal{S}2}$, necessary to simulate the semantics Φ.

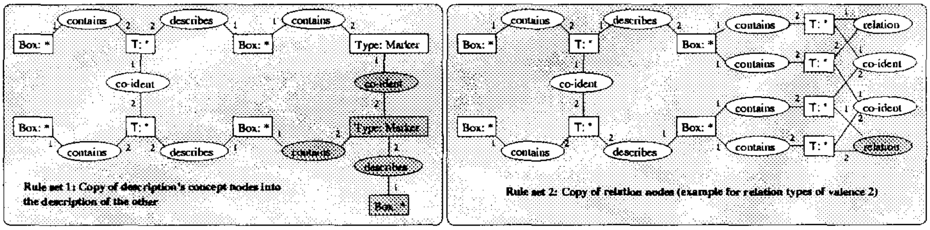

Fig. 11. Rules copying the description of co-identical nodes

Proposition 6 (Equivalence). *Let $\mathcal{S}$ be a support, and G and H two* NG*s or* NG^{ref}*s defined on $\mathcal{S}$. Let G' and H' be their associated* BG*s, defined on the knowledge base $\mathcal{KB}^{\mathcal{S}}_{\mathcal{B}}(\mathcal{S}) = \{\mathcal{S}^{\mathcal{I}}, \mathcal{R}^{\mathcal{S}}_{\mathcal{B}}\}$. Then $G' \models H'$ iff $\Phi(\mathcal{S}), \Phi(G) \models \Phi(H)$.*

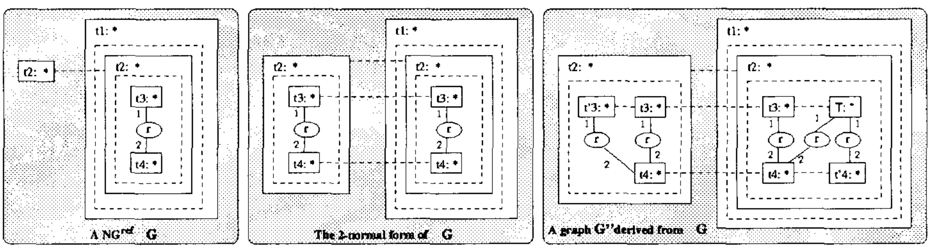

Fig. 12. Simulating the k-normal form with rules

Proof. We cannot, as for prop. 2, reason on the complete expansion of G'. Even with the simple example[6] in fig. 12, where any k-normal form of G is identical to its 2-normal form, we can derive from G a graph whose size is not bounded. But we can prove that, for any natural integer k, there exists a BG G'' such that $G' \Vdash G''$ and the BG associated with the k-normal form of G can be projected into G''. Conversely, we can prove that, for any BG G'' derived from G', there exists a natural integer $k = \text{depth}(\mathcal{A}(G''))$ such that G'' can be projected into the BG (with *co-ident* relation nodes) associated with the k-normal form of G.

A problem with the rules copying concept nodes in the descriptions of co-identical nodes (first rule of fig. 11) is that they do not generate valid nested^{ref} BGs. As shown in the graph G'' of fig. 12, for any concept node C labelled `(t, m)`, these rules can create an infinity of concept nodes co-identical to C, whose label can be a superlabel of `(t, m)`. As its co-identical concept nodes have different types, the graph obtained in such a way cannot be associated to a NG^{ref}. In order to solve this problem, we have to weaken the constraints on the relation *co-ident*: concept nodes sharing the same individual marker are in the same co-identity class, and the set of labels of concept nodes in the same co-identity class

[6] I adopted for these graphs a "nested graph representation", which is somewhat easier to read.

has a smallest element. Co-identical nodes are fusionned into a node having the most specific type and marker during normalizations. Adopting this weakened constraint solves this problem, without changing any of the preceding results.

□

Conclusion

In this paper, I show that a model generalizing $\mathtt{NG}^{ref}$s can be exactly represented with SGs, in such a way that all projections are preserved when translating graphs from one model to the other. In order to simulate the various reasonings induced by co-identity, be it in SGs or NGs, interpreting identity with the semantics Φ or Ψ, I show that we only need the "SG + rules" model. At least, the rules presented can be seen as a "graphical illutration" of the different operations required by the co-identity relation. At most, this model can be seen as a "low-level layer" for the implementation of $\mathtt{NG}^{ref}$s, boxes being only a "man-machine interface" layer. I prefer to see this model as a "protyping tool", a way to rapidly define and test various semantics of identity, using a development platform such as CoGITaNT [2]. But, for an efficiency purpose, several reasonings that can be represented by rules must still be given a "hard-coded", specific algorithmic solution.

References

1. M. Chein, M-L. Mugnier, and G. Simonet. Nested Graphs : A Graph-based Knowledge Representation Model with FOL Semantics. In *Proceedings of the 6th International Conference "Principles of Knowledge Representation and Reasoning" (KR'98).* Morgan Kaufmann Publishers, 1998.
2. D. Genest and E. Salvat. A Platform Allowing Typed Nested Graphs: How CoGITo Became CoGITaNT. In *Proceedings of the 6th International Conference on Conceptual Structures*, Lecture Notes in AI. Springer, 1998.
3. B.C. Ghosh and V. Wuvongse. Computational Situation Theory in the Conceptual Graph Language. In *Proceedings of ICCS'96*, Lecture Notes in AI. Springer, 1996.
4. M-L. Mugnier and M. Chein. Représenter des connaissances et raisonner avec des graphes. *Revue d'Intelligence Artificielle*, 10-1:7–56, 1996.
5. E. Salvat. *Raisonner avec des opérations de graphes : graphes conceptuels et règles d'inférence.* PhD thesis, Université de Montpellier II, 1997.
6. E. Salvat and M-L. Mugnier. Sound and Complete Forward and Backward Chainings of Graph Rules. In *Proceedings of the 4th International Conference on Conceptual Structures*, Lecture Notes in AI. Springer, 1996.
7. G. Simonet. Une autre sémantique logique pour les graphes conceptuels simples ou emboîtés. Research Report, 1996.
8. G. Simonet. Une sémantique logique pour les graphes conceptuels emboîtés. Research Report, 1996.
9. G. Simonet. Two FOL Semantics for Simple and Nested Conceptual Graphs. In *Proceedings of the 6th International Conference on Conceptual Structures*, Lecture Notes in AI. Springer, 1998.
10. J. F. Sowa. *Conceptual structures : Information processing in mind and machine.* Addison-Wesley, 1984.