Académie de Montpellier

Université Montpellier II

- sciences et techniques du languedoc -

Diplôme d'études approfondies

- INFORMATIQUE -

Propositions pour l'extension du modèle des Graphes Conceptuels

Emboîtement, co-référence et règles de graphes

Jean-François Baget

Date de soutenance : 30 Juin 1998

Tutrices de stage : Marie-Laure Mugnier, Geneviève Simonet

Remerciements

Ce mémoire de DEA est l'aboutissement d'un stage de quatre mois au sein de l'équipe Graphes Conceptuels du LIRMM.

Je tiens à adresser mes premiers remerciements à mes deux tutrices de stage, Marie-Laure Mugnier et Geneviève Simonet. L'attention, la minutie et la précision dont elles ont fait preuve en corrigeant les premières versions partielles de ce rapport m'ont été d'une aide précieuse pendant la dernière phase de rédaction.

Ces remerciements sont aussi adressés à toute l'équipe de Graphes Conceptuels, Michel Chein en tête, mais aussi David Genest et Eric Salvat, qui m'ont si vite considéré comme un membre à part entière de cette équipe.

Je ne veux pas oublier ici les enseignants-chercheurs du LIRMM, à savoir Michel Habib, qui m'a fait suffisamment confiance pour me permettre de faire, dès cette année, mes premières armes en tant qu'enseignant, Roland Ducournau, à qui je dois mes premiers pas en représentation des connaissances, et tous ceux qui, de ma licence à cette année de DEA, m'ont fait découvrir et apprécier l'informatique fondamentale.

La bonne humeur de mes camarades de DEA a été d'un grand réconfort pendant ces dernières nuits passées au LIRMM. Nos discussions sur nos sujets respectifs m'ont souvent permis de classifier et de clarifier mes idées. J'adresse donc ici à Yannic, Florent et Jean-Marie les remerciements qu'ils méritent.

Enfin, je remercie Christine de la patience et du soutien qu'elle m'a accordés pendant certaines périodes critiques (examens, rapports) de ces trois dernières années. C'est un peu grâce à elle, et pour elle, que je l'ai fait.

Table des matières

Introduction

Ce mémoire de DEA concerne diverses extensions envisageables d' un modèle formel de représentation des connaissances, du type réseau sémantique, connu sous le nom de graphes conceptuels. Dans son livre de 1984 [Sow84], John F. Sowa avait défini un modèle de base pour des graphes conceptuels que l'on peut appeler simples, et esquissé un certain nombre d'extensions à ce modèle.

Ce modèle distingue deux niveaux de connaissances :

- Un support, la connaissance ontologique de base, qui est une relation d'ordre partiel sur un ensemble de types de concepts et un ensemble de types de relations. On pourrait mettre ce support en relation avec la hiérarchie des classes dans les langages à objets.
- Le graphe conceptuel lui-même, qui est un multi-graphe étiqueté biparti, défini sur le support, et peut être considéré comme l'ensemble des assertions sur le monde. Ce graphe comprend deux types de sommets, les concepts, qui représentent des entités ou des instances de leur type, et les relations qui représentent les relations entre ces entités.

Ce graphe conceptuel peut s'interpréter comme une formule logique en logique positive conjonctive existentielle du premier ordre. Quant au raisonnement sur de tels graphes, on peut dégager deux approches très différentes :

- Un graphe conceptuel est une représentation graphique d'une formule logique, et la déduction logique s'illustre graphiquement (au sens d'un graphique, d'un dessin) par des manipulations du graphe, qui peuvent ne pas correspondre à des algorithmes de graphes identifiés.
- Un graphe conceptuel est un graphe (au sens de la théorie des graphes) [CM95], et le raisonnement sur ces graphes se fait par des opérations de graphes particulières, basées sur un morphisme de graphe traditionnellement appelé ¡¡ projection¿¿.

C'est cette deuxième approche qui a été favorisée au LIRMM depuis la première étude de Michel Chein et Marie-Laure Mugnier sur les graphes conceptuels [CM92]. Il semble en effet désormais acquis que l'algorithmique de graphes étiquetés est un domaine de recherche à part entière en représentation des connaissances.

La représentation de connaissances par des graphes étiquetés du type graphes conceptuels offre des avantages non négligeables :

- Communication homme/machine : ces graphes étiquetés semblent plus facilement assimilables que la logique du premier ordre par un utilisateur ¡¡ final ¿¿ qui souhaite formaliser des connaissances complexes. Ce formalisme permet également de suivre graphiquement les étapes du raisonnement. C'est le pari fait par Bernard Botella et Patrick Taillibert à Dassault Électronique, en considérant les graphes conceptuels comme un outil intuitif et puissant de modélisation de comportements [BBV97] [Bos98].
- Efficacité algorithmique : le raisonnement sur les graphes conceptuels peut bénéficier de la grande quantité de résultats en algorithmique de graphes. De plus, le raisonnement vu comme projection de graphes peut s'appuyer sur la forte structuration de la représentation du monde. Eric Salvat et Stéphane

Coulondre ont commencé au LIRMM une étude comparative sur l'efficacité du chaînage arrière dans les graphes conceptuels et celle de l'unificateur PROLOG sur les formules logiques équivalentes [Sal98] [CS98].

Le modèle des graphes conceptuels simples établi, ses fondations ont pu être étayées par la preuve de l'adéquation et la complétude de la projection par rapport à la sémantique logique introduite par John F. Sowa [CM92] [MC96]. Il est à noter que cette équivalence ne veut pas dire que les graphes conceptuels sont une illustration de la logique du premier ordre. Donner aux graphes une sémantique logique consistante et complète doit plutôt être vu comme un moyen de s'assurer la cohérence de nos opérations de graphes. D'autres sémantiques peuvent être utilisées, comme la sémantique ensembliste présentée dans [MC96], qui est plus proche d'une vision "théorie des modèles".

Ces bases fermement établies, plusieurs extensions sur le modèle des graphes conceptuels ont été proposées au LIRMM, et sont en cours d'implémentation sur la plate-forme CoGITo / CoGITaNT [Hae95] [GS98].

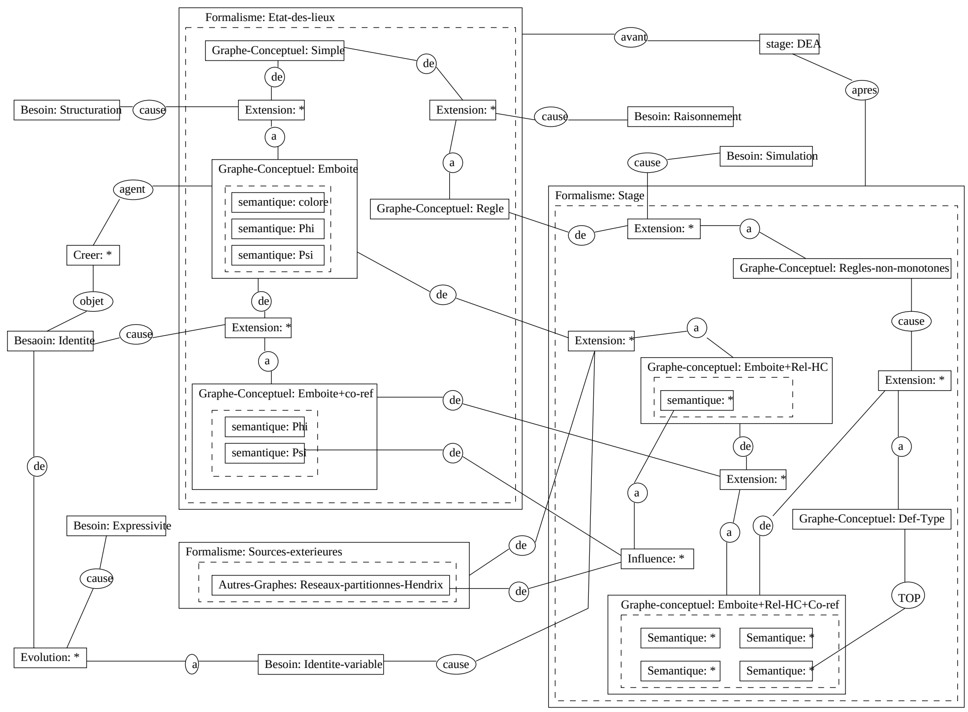
- Graphes Conceptuels Emboîtés : cette extension, dûe à un besoin de représenter le contexte dans lequel se situe un graphe conceptuel, répond autant à un critère de simplification de la communication homme-machine qu'à celui de l'efficacité algorithmique, en permettant une meilleure structuration du graphe [MC96] [PMC95] [Sim96b] [Sim96a].
- Liens de co-référence : cette extension concerne le traitement de l'identité entre sommets concepts, rendu nécessaire par l'emboîtement. En effet, il peut être utile de spécifier que deux concepts dans des contextes différents représentent la même entité.
- Règles de graphes conceptuels : cette extension donne la possibilité de définir un ensemble de règles sur un graphe conceptuel, qui représentent les règles d'inférence sur la base de faits. Eric Salvat [Sal97] a formalisé le raisonnement en marche avant et en chaînage arrière sur des graphes conceptuels simples munis de règles monotones.

Le sujet de ce stage de DEA porte sur les règles de graphes emboîtés typés avec liens de co-référence. Ce sujet étant trop vaste pour être traité exhaustivement dans le cadre de ce stage, j'ai du privilégier l'aspect formalisation au dépend de l'aspect algorithmique. Les choix de formalisation devront prendre en compte les critères suivants :

- Expressivité : Les extensions du modèle des graphes conceptuels doivent non seulement permettre une représentation plus fine des connaissances, mais surtout prendre en compte un raisonnement plus précis sur ces connaissances, que ce soit par les requêtes ou l'emploi de règles de graphes. Un bon test pour cette expressivité me semble être la possibilité de représenter et de raisonner sur des situations complexes du Monde Réel™.
- Généralité : Les graphes conceptuels étant utilisés dans un grand nombre de domaines, simulation comportementale [BBV97] [Bos98], recherche documentaire [Gen96], tous les choix portant sur une extension du modèle des graphes conceptuels devront se faire en faveur du modèle le plus général possible.
- Simplicité : Les choix de formalisation devront prendre en compte la lisibilité du graphe, et les opérations de graphes tenant lieu de raisonnement devront être aussi claires et intuitives que possible. Idéalement, les idées de l'utilisateur devraient se traduire de façon quasi-immédiate en un graphe conceptuel …
- Efficacité : Tout formalisme permettant une plus grande structuration d'un graphe conceptuel peut être exploité au niveau algorithmique. Il faudra faire tout particulièrement attention aux changements de classes de complexité entrainés par une extension du modèle [1].

[1] Même si le travail présent néglige – faute de temps – les considérations algorithmiques. Mea

FIG. 1 – Influences réciproques de diverses extensions pour les graphes conceptuels



La méthodologie utilisée tout au long de mon stage de DEA peut s'exprimer en terme de "feedback". Tout travail de formalisation sur une extension du modèle des graphes conceptuels ne peut en effet se faire qu'en fonction de ce qu'on veut pouvoir exprimer avec l'extension suivante. Cette démarche est illustrée par la figure 1 qui, dans un légitime besoin de clarté (?), est exprimée sous la forme d'un graphe conceptuel. Il sera difficile de rendre compte des tortueuses dépendances suggérées par ce graphe à travers la structure pauvrement linéaire qu'impose un mémoire universitaire. La structuration de ce mémoire [2] se fera alors de la façon suivante :

- Emboîtement : Après avoir rappelé les différents modèles ayant été proposés au LIRMM, une brève étude prospective sur la signification de l'identité démontrera le besoin d'une généralisation de ce modèle. Je montrerai que ce modèle est absolument équivalent à celui des graphes simples muni d'une sémantique adéquate, qui sera donc tout naturellement adéquate et complète pour la projection.
- Co-référence : Deux approches semblent possibles pour indiquer la co-référence. Soit on utilise un traitement particulier, soit on utilise une relation du support à laquelle on peut associer un ensemble de règles. Je discuterai de ces différentes approches, plus en termes d'expressivité (ce que l'on peut déduire du graphe), qu'en termes d'efficacité algorithmique. Le traitement de l'identité m'amènera à considérer un problème proche de celui de la définition de types.

Tout en parcourant ce mémoire, le lecteur devra accepter d'examiner des propositions à la lumière d'extensions du modèle qui ne seront, à cet instant précis, définies que de manière intuitive. Les fortes dépendances entre les différents problèmes étudiés contraignant à cet exercice, qu'il veuille bien, par avance, m'en excuser.

---

culpa ...

[2] Que l'on peut voir comme un arbre recouvrant du graphe précédent ...

Chapitre 1

Emboîtement et notion de contexte

Je présenterai dans un premier temps le modèle des graphes simples tel qu'il est défini dans [CM92] [MC96]. Puis, après avoir présenté différentes sémantiques pour les graphes conceptuels emboîtés [Sim96b] [Sim96a], je justifierai le besoin d'étendre ce modèle et présenterai une généralisation de ce modèle.

1.1 Graphes conceptuels simples

Un graphe conceptuel simple est un multigraphe étiqueté (c'est à dire qu'il peut y avoir plusieurs arêtes entre deux sommets) biparti que l'on définit sur un support. Ce support détermine la connaissance ontologique de base, le vocabulaire, et donc détermine ce que l'on est en droit d'écrire dans le graphe. Le graphe conceptuel muni de son support a une sémantique en logique du premier ordre, et l'opération de projection est adéquate [Sow84] et complète (sous une certaine condition de normalité [CM92]) par rapport à cette sémantique.

1.1.1 Support et graphes conceptuels

Définition 1 (Support d'un graphe simple) *Le support d'un graphe conceptuel simple est défini par la donnée de :*

(i) Un ensemble partiellement ordonné par une relation $\leq_C$ [1] $\mathcal{T}_C$ *de types de concepts, muni d'un élément maximal,* $\top_C$*, et d'un élément minimal,* $\bot_C$*.*

(ii) Un ensemble $\mathcal{T}_R$ *de types de relations, qui est partitionné en* k *sous-ensembles* $\{\mathcal{T}_R^i\}_{i \in \{1 \ldots k\}}$ *de types de relations d'arité* i*, chacun de ces sous-ensembles étant partiellement ordonné par une relation* $\leq_R^i$*, et admettant un élément maximal* $\top_R^i$ *et un élément minimal* $\bot_R^i$*.*

(iii) Une application σ *appelée signature qui associe à tout type de relation de* $\mathcal{T}_R^k$ *un tuple composé de* k *types de concepts.* $\sigma_k(r)$ *est le type maximal du* $k^{ième}$ *argument de la relation* r*. La signature respecte la contrainte : si* r_1 *et* r_2 *sont deux types de relations de* $\mathcal{T}_R^k$ *et* $r_1 \leq_R r_2$*, alors* $\forall i \in \{1 \ldots k\} \sigma_i(r_1) \leq_C \sigma_i(r_2)$*.*

(iv) Un ensemble $\mathcal{I}$ *de marqueurs individuels et un marqueur générique* $*$*, munis d'une relation d'ordre partiel telle que* $*$ *est plus grand que tous les marqueurs de* $\mathcal{I}$ *et tous les marqueurs de* $\mathcal{I}$ *sont incomparables.*

(v) Une application τ *appelée relation de conformité qui à tout marqueur individuel associe un type de concept différent de* $\bot_C$*.*

[1] Ce n'est pas nécessairement un treillis

Les éléments minimaux $\bot_C$ et $\bot_R^i$ n'étant ni instantiables dans le graphe, ni utiles aux démonstrations, il sera préférable de s'en passer.

Définition 2 (Graphes conceptuels simples) *Un graphe conceptuel simple* $G = (C, R, U, \epsilon)$*, défini sur un support* $\mathcal{S}$*, est un multigraphe étiqueté biparti satisfaisant les contraintes suivantes :*

(i) C *est un ensemble non vide de sommets concepts.*

(ii) R *est un ensemble de sommets relations.*

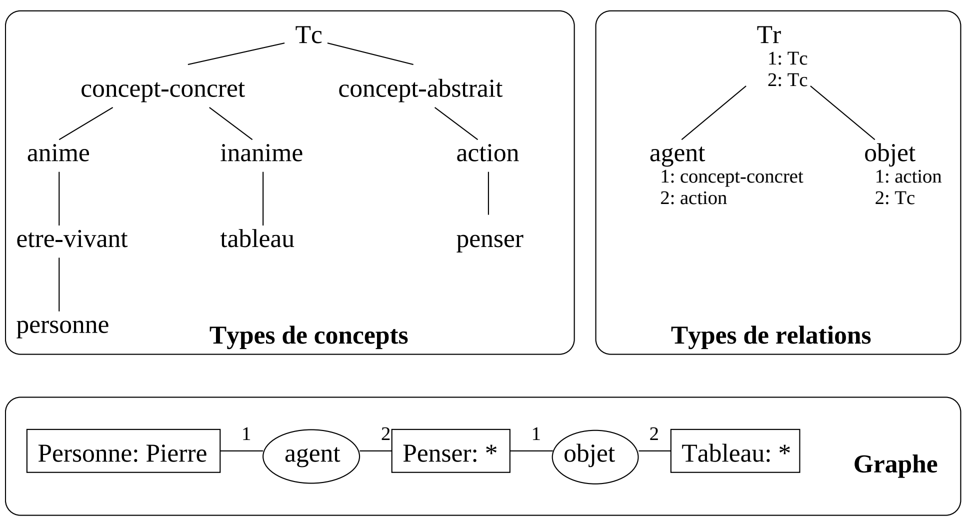
(iii) U *est l'ensemble des arêtes. L'ensemble des arêtes adjacentes à un sommet relation* S_R *est totalement ordonné par une numérotation des arêtes de 1 à degré*(S_R).

(iv) ϵ *est une application qui associe à tout sommet relation de degré k une étiquette qui est un type de relation de* $\mathcal{T}_R^k$*, et qui associe à tout sommet concept une étiquette de* $((\mathcal{T}_C \setminus \{\bot_C\}) \times (\mathcal{I} \cup *))$*. Un sommet concept dont le marqueur est un élément de* $\mathcal{I}$ *est appelé sommet concept individuel, les autres sont des sommets concepts génériques.*

(v) ϵ *obéit aux contraintes définies par la signature et la relation de conformité : si* S_R *est un sommet relation de* G*, alors type*$(G_i(S_R)) \leq_C \sigma_i(type(S_R))$*, et si* S_C *est un sommet concept individuel, alors type*$(S_C) = \tau(marqueur(S_C))$.

Vu que je n'ai plus $\bot_C$, l'étiquette des sommets concepts *(iv)* devient un élément de $(\mathcal{T}_C \times (\mathcal{I} \cup *))$.

FIG. 1.1 – Un exemple de graphe conceptuel simple, muni de son support : "Pierre pense à un tableau"



La figure 1.1 donne un exemple de graphe conceptuel simple défini sur un support "minimaliste". Les graphes conceptuels suivants seront donnés sans leur support.

1.1.2 Interprétations logiques

Je parlerai tout au long de ce mémoire de deux sémantiques logiques pouvant être associées aux graphes conceptuels : la sémantique Φ [Sow84] [CM92], et la sémantique Ψ [Sim96a], moins intuitive, mais qui décrit plus précisément le graphe en fonction de ses sommets.

La sémantique Φ

On associe à un graphe conceptuel muni de son support une sémantique logique définie de la façon suivante :

Interprétation de $\mathcal{T}_C$ À tout couple (C_1, C_2) de types de concepts tel que C_1 couvre C_2 pour la relation d'ordre $\leq_C$ [2], on associe la formule $\forall x (C_2(x) \rightarrow C_1(x))$ [3]. On interprète le type universel et le type absurde par $\forall x \top_C(x)$ et $\forall x \neg(\bot_C(x))$.

L'interprétation de $\mathcal{T}_C$ est la conjonction de l'ensemble de ces formules.

Interprétation de $\mathcal{T}_R$ Pour chaque $\mathcal{T}_R^k$, à tout couple (R_1, R_2) de types de relations d'arité k de $\mathcal{T}_R^k \times \mathcal{T}_R^k$ tel que R_1 couvre R_2 pour la relation d'ordre $\leq_R^k$, on associe la formule $\forall x_1 \ldots \forall x_k (R_2(x_1, \ldots, x_k) \rightarrow R_1(x_1, \ldots, x_k))$.

L'interprétation de $\mathcal{T}_R$ est la conjonction de l'ensemble de ces formules.

Interprétation de la signature A chaque type de relation R de $\mathcal{T}_R^k$, de signature $\sigma(R) = (C_1, \ldots, C_k)$ on associe la formule $\forall x_1 \ldots \forall x_k (R(x_1, \ldots, x_k) \rightarrow (C_1(x_1) \wedge \ldots \wedge C_k(x_k)))$.

L'interprétation de la signature est la conjonction de l'ensemble de ces formules.

Interprétation de la relation de conformité A tout marqueur individuel m tel que $\tau(m) = C$, on associe la formule $C(m)$ [4].

L'interprétation de la relation de conformité est la conjonction de l'ensemble de ces formules.

L'interprétation du support est la conjonction de ces interprétations. Quelques remarques toutefois à propos de cette interprétation :

- Comme on n'a jamais de sommet de type $\bot_C$ dans le graphe, il n'est pas nécessaire de l'interpréter. De même, toutes les instanciations nécessaires de la formule $\forall x \top_C(x)$ se déduisent de l'interprétation de la relation d'ordre $\leq_C$, où $\top_C$ est l'élément maximal. Cette interprétation peut donc être considérée comme redondante.
- L'interprétation de la signature est redondante, car cette information découle des contraintes imposées par la signature sur la construction du graphe et de l'interprétation de la relation d'ordre sur les types de concepts. J'ignorerai donc également cette interprétation. [5]
- L'interprétation de la relation de conformité sera elle aussi ignorée, car elle poserait des problèmes pour la preuve de la complétude [6].

À titre d'exemple, voici l'interprétation logique du support présenté dans la figure 1.1.

$$\begin{array}{rcl}\Phi(\mathcal{T}_C) & = & (\forall x(\text{personne}(x) \rightarrow \text{etre-vivant}(x))) \\ & \wedge & (\forall x(\text{etre-vivant}(x) \rightarrow \text{anime}(x))) \\ & \wedge & (\forall x(\text{anime}(x) \rightarrow \text{concept-concret}(x)))\end{array}$$

[2]i.e. C_1 est un plus petit majorant de C_2 pour la relation d'ordre $\leq_C$.

[3]Par un abus d'écriture bien compréhensible, on écrit de la même manière le type de concept et le prédicat qui lui est associé.

[4]Où encore une fois, par un abus d'écriture, on identifie le marqueur défini dans le support à une constante de la formule logique.

[5]Pourtant, on verra à la fin de ce mémoire que ces formules peuvent avoir une certaine importance.

[6]On pourrait déduire de la seule interprétation du support des atomes représentatnt des sommets qui ne se projettent pas dans le graphe.

$$
\begin{array}{rcl}
 & \wedge & (\forall x(\text{concept-concret}(x) \rightarrow \top_C(x))) \\
 & \wedge & (\forall x(\text{tableau}(x) \rightarrow \text{inanime}(x))) \\
 & \wedge & (\forall x(\text{inanime}(x) \rightarrow \text{concept-concret}(x))) \\
 & \wedge & (\forall x(\text{penser}(x) \rightarrow \text{action}(x))) \\
 & \wedge & (\forall x(\text{action}(x) \rightarrow \text{concept-abstrait}(x))) \\
 & \wedge & (\forall x(\text{concept-abstrait}(x) \rightarrow \top_C(x))) \\
\Phi(\mathcal{T}_R) & = & (\forall x \forall y(\text{agent}(x,y) \rightarrow \top_R^2(x,y))) \\
 & \wedge & (\forall x \forall y(\text{objet}(x,y) \rightarrow \top_R^2(x,y))) \\
\Phi(\mathcal{S}) & = & \Phi(\mathcal{T}_C) \wedge \Phi(\mathcal{T}_R)
\end{array}
$$

Reste à donner l'interprétation logique du graphe lui-même, ce qui se fait par l'interprétation des sommets concepts, puis celle des sommets relations.

Interprétation des sommets concepts A tout sommet concept individuel du graphe, de type C et de marqueur m, on associe la formule logique $C(m)$, où m est la constante associée au marqueur m. A tout sommet concept générique de type C, on associe la formule $C(x)$, où x est une nouvelle variable.

Interprétation des sommets relations A tout sommet relation de type R et de degré k, dont les sommets concepts voisins sont implicitement ordonnés $C_1, \ldots, C_k$ par l'ordre des arêtes, on associe la formule logique $R(m_1, \ldots, m_k)$ où m_i est la constante associée au marqueur de C_i si celui-ci est un sommet concept individuel, et la variable associée à l'interprétation de C_i sinon.

L'interprétation du graphe est la fermeture existentielle de la conjonction de ces interprétations.

Sur le même exemple, voici l'interprétation logique du graphe conceptuel de la figure 1.1.

$$
\begin{array}{rcl}
\Phi(G) & = & \exists x_1 \exists x_2(\text{Personne}(\text{Pierre}) \\
 & \wedge & \text{Penser}(x_1) \\
 & \wedge & \text{Tableau}(x_2) \\
 & \wedge & \text{agent}(\text{Pierre}, x_1) \\
 & \wedge & \text{objet}(x_1, x_2))
\end{array}
$$

La sémantique Ψ

Avec cette sémantique, un sommet concept n'est plus seulement identifié par la constante ou la variable associée à son marqueur, mais aussi par une variable associée au sommet lui-même, en tant qu'élément d'un graphe. Avec cette sémantique, deux termes sont associés à un sommet concept :

- Une nouvelle variable associée au sommet lui-même.
- Une variable ou une constante associée au marqueur, de la même façon que pour la sémantique Φ.

L'interprétation du support est modifiée : les formules atomiques associées aux sommets concepts ne sont plus des formules unaires, mais binaires.

Interprétation de $\mathcal{T}_C$ À tout couple (C_1, C_2) de types de concepts tel que C_1 couvre C_2 pour la relation d'ordre $\leq_C$, on associe la formule $\forall x \forall y (C_2(x,y) \rightarrow C_1(x,y))$.

L'interprétation de $\mathcal{T}_C$ est la conjonction de l'ensemble de ces formules.

Interprétation de $\mathcal{T}_R$ L'interprétation par Ψ de $\mathcal{T}_R$ est identique à celle obtenue par Φ.

Comme dans le cas de la sémantique Φ,l 'interprétation du support est la conjonction de ces interprétations. Voici comment s'interprète, par la sémantique Ψ, le support présenté dans la figure 1.1.

$$
\begin{array}{rcl}
\Psi(\mathcal{T}_C) & = & (\forall x \forall y (\text{personne}(x,y) \rightarrow \text{etre-vivant}(x,y))) \\
& \wedge & (\forall x \forall y (\text{etre-vivant}(x,y) \rightarrow \text{anime}(x,y))) \\
& \wedge & (\forall x \forall y (\text{anime}(x,y) \rightarrow \text{concept-concret}(x,y))) \\
& \wedge & (\forall x \forall y (\text{concept-concret}(x,y) \rightarrow \top_C(x,y))) \\
& \wedge & (\forall x \forall y (\text{tableau}(x,y) \rightarrow \text{inanime}(x,y))) \\
& \wedge & (\forall x \forall y (\text{inanime}(x,y) \rightarrow \text{concept-concret}(x,y))) \\
& \wedge & (\forall x \forall y (\text{penser}(x,y) \rightarrow \text{action}(x,y))) \\
& \wedge & (\forall x \forall y (\text{action}(x,y) \rightarrow \text{concept-abstrait}(x,y))) \\
& \wedge & (\forall x \forall y (\text{concept-abstrait}(x,y) \rightarrow \top_C(x,y))) \\
\Psi(\mathcal{T}_R) & = & \Phi(\mathcal{T}_R) \\
\Psi(\mathcal{S}) & = & \Psi(\mathcal{T}_C) \wedge \Psi(\mathcal{T}_R)
\end{array}
$$

Interprétation des sommets concepts A tout sommet concept individuel du graphe, de type C et de marqueur m, on associe la formule logique $C(x, M)$, où m est la constante associée au marqueur m et x est une nouvelle variable. A tout sommet concept générique de type C, on associe la formule $C(x, y)$, où x et y sont de nouvelles variables.

Interprétation des sommets relation A tout sommet relation de type R et de degré k, dont les sommets concepts voisins sont implicitement ordonnés $C_1, \ldots, C_k$ par l'ordre des arêtes, on associe la formule logique $R(x_1, \ldots, x_k)$ où x_i est la variable associée au ¡¡ sommet physique ¿¿ C_i, i.e. le premier terme de l'atome $\Psi(C_i)$.

L'interprétation du graphe est la fermeture existentielle de la conjonction de ces interprétations.

À titre d'exemple, voici l'interprétation logique du graphe conceptuel de la figure 1.1.

$$
\begin{array}{rcl}
\Psi(G) & = & \exists x_1 \exists x_2 \exists x_3 \exists y_1 \exists y_2 (\text{Personne}(x_1, \text{Pierre}) \\
& \wedge & \text{Penser}(x_2, y_1) \\
& \wedge & \text{Tableau}(x_3, y_2) \\
& \wedge & \text{agent}(x_1, x_2) \\
& \wedge & \text{objet}(x_2, x_3))
\end{array}
$$

1.1.3 Projection, adéquation et complétude

Définition 3 (Projection de graphes conceptuels simples) *Une projection d' un graphe conceptuel simple $H = (C_H, R_H, U_H, \epsilon_H)$ dans un graphe conceptuel simple $G = (C_G, R_G, U_G, \epsilon_G)$, définis sur un même support $\mathcal{S}$, est un morphisme de graphe biparti, i.e. un couple d'applications $\Pi = (\pi_C : C_H \to C_G, \pi_R : R_H \to R_G)$ respectant les contraintes imposées par le support :*

(i) Conservation des arêtes et de leur numérotation : si C est un sommet concept de H et R un sommet relation de H reliés par une arête de U_H étiquetée par un entier n, alors $\pi_C(C)$ et $\pi_R(R)$ sont reliés par une arête de U_G étiquetée par n.

(ii) Restriction des étiquettes des sommets relation : si R est un sommet relation de H, alors $type(\pi_R(R)) \leq_R type(R)$.

(iii) Restriction des étiquettes des sommets concept : si C est un sommet concept de H, alors $type(\pi_C(C)) \leq_C type(C)$ et $marqueur(\pi_C(C)) \leq_C marqueur(C)$.

Cette opération de projection peut se décomposer en un ensemble d'opérations élémentaires de spécialisation [Sow84] [CM92].

Définition 4 (Forme normale d'un graphe simple) *Soit S un support et G un graphe conceptuel simple défini sur S. On dit que G est sous forme normale si tous ses sommets concepts individuels ont un marqueur différent.*

On peut mettre n'importe quel graphe conceptuel simple sous forme normale en fusionnant les sommets concepts ayant même marqueur individuel. Les formules logiques associées par la sémantique Φ à un graphe conceptuel simple et à sa forme normale sont trivialement équivalentes.

Théorème 1 (Adéquation et complétude par rapport à Φ) *Soient G et H deux graphes conceptuels définis sur un support $\mathcal{S}$, G étant sous forme normale. Alors il existe une projection de H dans G si et seulement si $\Phi(H)$ est une conséquence logique de $\Phi(G)$ et $\Phi(\mathcal{S})$.*

C'est le théorème central, qui justifie la formule "raisonner avec des graphes". La preuve de l'adéquation est déjà dans [Sow84] [CM92]. Pour la preuve de la complétude, voir [MC96]. Quand à la sémantique Ψ, on a un résultat équivalent, mais sans aucune restriction sur le graphe [Sim96a].

Théorème 2 (Adéquation et complétude par rapport à Ψ) *Soient G et H deux graphes conceptuels définis sur un support $\mathcal{S}$, sans condition de normalité. Alors il existe une projection de H dans G si et seulement si $\Psi(H)$ est une conséquence logique de $\Psi(G)$ et $\Psi(\mathcal{S})$.*

Le graphe H que l'on cherche à projeter pouvant être vu comme une requête sur la base de données représentée par un graphe G, non nécessairement connexe [CMS98], je parlerai souvent de "graphe requête" pour le graphe que l'on cherche à projeter.

Au point de vue algorithmique, la projection d'un graphe conceptuel dans un autre est un problème NP-complet. Dans le cas où le graphe requête est un arbre, le problème est polynomial. Comme on peut raisonnablement espérer que des requêtes formulées par un être humain auront une structure simple, un filtrage du traitement par projection d'un arbre recouvrant de la requête devrait donner de bons résultats [Sal93] [Mug95].

1.2 Graphes conceptuels emboîtés

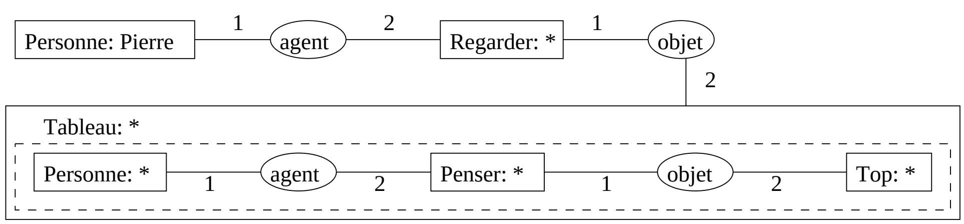
L'extension des graphes conceptuels simples aux graphes conceptuels emboîtés répond à deux besoins précis :

– Communication homme/machine : les grands graphes conceptuels deviennent vite illisibles. Il est alors nécessaire de structurer ces graphes en parties bien distinctes, *relativement* indépendantes les unes des autres. Par ailleurs, cette structuration faite par l'utilisateur peut être exploitée algorithmiquement.
– Expressivité : l'emboîtement est un moyen naturel de formaliser la notion de contexte, nécessaire, par exemple, dans le traitement de la langue naturelle. Mais une définition claire et opérationnelle de ce que l'on entend par "contexte" me semble encore manquer.

1.2.1 Qu'est-ce-qu'un contexte ?

John F. Sowa [Sow97] propose trois aspects de l'utilisation du contexte, inspirés des trois catégories de Charles Peirce :
– Aspect syntaxique : la fonction syntaxique du contexte est de regrouper, de délimiter une partie de l'information.
– Aspect sémantique : l'information présentée dans un contexte peut se référer à une situation "réelle ou hypothétique". Le contexte sert donc à donner son sens à l'information.
– Aspect pratique : "Contextualiser" une certaine partie de l'information lui donne un intérêt particulier.

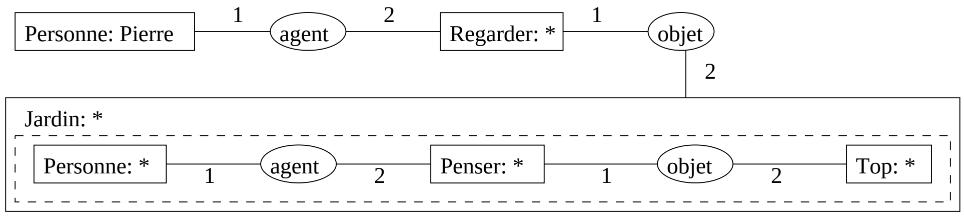
FIG. 1.2 – Un exemple de graphe conceptuel emboîté : "Pierre regarde un tableau, celui-ci représente une personne qui pense à quelque-chose."



La représentation que nous avons adoptée, comme le montre la figure 1.2, illustre ces principes.
– Syntaxiquement, l'emboîtement regroupe dans un certain graphe une partie de l'information : celle qui est représentée par le tableau. Cette description est englobée dans une boîte, représentée par des pointillés, contenue dans le sommet concept *Tableau*. Cette boîte, qui semble à priori redondante, donne la possibilité d'associer plusieurs descriptions, plusieurs aspects à une entité, chacun étant représenté par une boîte distincte.
– Au niveau sémantique, l'information que l'on retrouve dans le contexte du tableau ne prend son sens que dans ce contexte : si quelqu'un pense à quelque-chose en dehors de ce contexte, ce n'est que par hasard.
– Au niveau pratique, l'emboîtement n'est pas toujours aussi nécessaire. Dans la figure 1.3, il y a bien une personne qui pense, *que l'on s'intéresse au jardin ou pas.* C'est celui qui conçoit le graphe qui décide que l'information trouve son intérêt dans le contexte du graphe.

Ces trois catégories se retrouvent dans la formalisation des graphes conceptuels emboîtés : syntaxiquement, nous aurons besoin de définir ce qu'est un graphe emboîté, ce qui est l'objet de la définition suivante. Sémantiquement, il nous faudra savoir ce que devient la projection dans de tels graphes, et quelle est la sémantique logique qui lui est associée. Je présenterai dans un premier temps trois sémantiques logiques proposées au LIRMM. Enfin, pour répondre à un besoin pratique, il faudra garder à l'esprit que l'emboîtement répond à l'*intérêt* que peut avoir le concepteur

FIG. 1.3 – Un autre exemple de graphe conceptuel emboîté : “Pierre regarde un jardin, à l’interieur, une personne pense à quelque-chose.”



du graphe pour un certain sous-ensemble de l’information, intérêt qu’il manifeste en regroupant cette information dans une “boîte”. Il ne sera donc pas question d’ imposer des règles sur le “bon emboîtement”, le “mauvais emboîtement”, ou “l’emboîtement de confort”.

La définition suivante présente les graphes conceptuels emboîtés tels qu’ils sont définis dans [MC96].

Définition 5 (Graphes conceptuels emboîtés) *Soit la donnée de G un graphe conceptuel simple. On construit un graphe conceptuel emboîté G' à partir de G en rajoutant à chaque sommet concept de G un troisième champ, que l’on peut considérer comme la description de ce sommet concept, et qui est initialisé à $**$. On a ainsi construit un graphe de base. Dans un graphe conceptuel emboîté, on peut remplacer une description égale à $**$ par un graphe conceptuel emboîté.*

On définit inductivement la projection de graphes conceptuels emboîtés de la façon suivante :

Définition 6 (Projection de graphes conceptuels emboîtés) *Une projection Π d’un graphe conceptuel emboîté H dans un graphe conceptuel emboîté G est un couple d’applications tel que :*
- *La restriction de Π au graphe de base ¡¡ racine ¿¿ de H est une projection de graphe simple dans le graphe de base ¡¡ racine ¿¿ de G.*
- *Inductivement, Si S est un sommet de H contenant une description non vide H_k, alors la restriction de Π à H_k est une projection de graphe simple dans le graphe de base $\Pi(S)$.*

1.2.2 Différentes sémantiques

Les sémantiques Φ et Ψ associées à un graphe emboîté sont des extensions de celles définies sur les graphes simples, obtenues en rajoutant aux atomes un nouvel argument appelé ¡¡ argument de contexte ¿¿.

Extension de la sémantique Φ

Dans l’interprétation du support, les prédicats unaires associés aux types de concepts deviennent des prédicats binaires, et les prédicats n-aires associés aux types de relations d’arité n deviennent des prédicats $(n+1)$-aires.

Un graphe conceptuel emboîté s’interprète de la manière suivante :
- Les sommets concepts appartenant au graphe de base ¡¡ racine ¿¿ s’interprètent par un prédicat binaire $C(a_0, m)$ où C est le type du sommet concept, m une variable ou constante associée au marqueur, et a_0 la constante (c’est toujours la même) associée à la racine du graphe.
- Inductivement, chaque sommet concept d’un graphe de base constituant la description d’un sommet concept S s’interprète par un prédicat $C(x, m)$ où x

est la variable ou la constante associée au marqueur de S et m la variable ou la constante associée au marqueur du sommet interprété.

– Les sommets relations s'interprètent de même dans un contexte, i.e. un sommet relation de type T et de degré n s'interprète par le prédicat $T(x, y_1, \ldots, y_n)$

Par exemple, le graphe emboîté de la figure 1.2 s'interprète de la façon suivante :

$$\Phi(G) = \exists x_1 \exists x_2 \exists x_3 \exists x_4 \exists x_5 (\text{Personne}(a_0, \text{Pierre}) \tag{1.1}$$
$$\wedge \ \text{Regarder}(a_0, x_1) \tag{1.2}$$
$$\wedge \ \text{Tableau}(a_0, x_2) \tag{1.3}$$
$$\wedge \ \text{agent}(a_0, Pierre, x_1) \tag{1.4}$$
$$\wedge \ \text{objet}(a_0, x_1, x_2) \tag{1.5}$$
$$\wedge \ \text{Personne}(x_2, x_3) \tag{1.6}$$
$$\wedge \ \text{Penser}(x_2, x_4) \tag{1.7}$$
$$\wedge \ \top(x_2, x_5) \tag{1.8}$$
$$\wedge \ \text{agent}(x_2, x_3, x_4) \tag{1.9}$$
$$\wedge \ \text{objet}(x_2, x_4, x_5)) \tag{1.10}$$

La formule 1.3 peut se lire *¡¡ Il y a un tableau (x_2) dans le contexte racine. ¿¿*, la formule 1.4 *¡¡ Pierre est l'agent de l'action x_1 dans le contexte racine. ¿¿*, et la formule 1.6 *¡¡ Il y a une personne (x_3) dans le contexte de x_2 (le tableau) ¿¿*.

La condition de normalité devenant : ¡¡ chaque graphe de base intervenant dans la construction de G est un graphe simple sous forme normale ¿¿, on peut étendre le résultat d'adéquation et complétude [Sim96a].

Théorème 3 (Adéquation et complétude) *Soient G et H deux graphes conceptuels emboîtés définis sur un support $\mathcal{S}$, G étant sous forme normale. Alors il existe une projection de H dans G si et seulement si $\Phi(H)$ est une conséquence logique de $\Phi(G)$ et $\Phi(\mathcal{S})$.*

Extension de la sémantique Ψ

L'extension de la sémantique Ψ aux graphes conceptuels emboîtés est similaire à celle de la sémantique Φ, mis à part que la liaison de l'argument de contexte se fait sur la variable associée au ¡¡ sommet physique ¿¿ du concept, et pas sur la variable ou la constante associée au marqueur.

L'interprétation du graphe emboîté de la figure 1.2 devient :

$$\Psi(G) = \exists x_1 \exists x_2 \exists x_3 \exists x_4 \exists x_5 \exists y_1 \exists y_2 \exists y_3 \exists y_4 \exists y_5 \exists y_6 (\text{Personne}(a_0, y_1, \text{Pierre}) \tag{1.11}$$
$$\wedge \ \text{Regarder}(a_0, y_2, x_1) \tag{1.12}$$
$$\wedge \ \text{Tableau}(a_0, y_3, x_2) \tag{1.13}$$
$$\wedge \ \text{agent}(a_0, y_1, y_2) \tag{1.14}$$
$$\wedge \ \text{objet}(a_0, y_2, y_3) \tag{1.15}$$
$$\wedge \ \text{Personne}(y_3, y_4, x_3) \tag{1.16}$$
$$\wedge \ \text{Penser}(y_3, y_5, x_4) \tag{1.17}$$
$$\wedge \ \top(y_3, y_6, x_5) \tag{1.18}$$
$$\wedge \ \text{agent}(y_3, y_4, y_5) \tag{1.19}$$
$$\wedge \ \text{objet}(y_3, y_5, y_6)) \tag{1.20}$$

La formule 1.11 peut se lire : *¡¡ Il y a dans le contexte racine un sommet, auquel j'associe la variable y_1, qui représente la personne : Pierre ¿¿.*

Geneviève Simonet a prouvé l'adéquation et la complétude de la projection par rapport à la sémantique Ψ, sans aucune restriction sur le graphe [Sim96a].

Théorème 4 (Adéquation et complétude) *Soient G et H deux graphes conceptuels emboîtés définis sur un support $\mathcal{S}$, sans condition de normalité. Alors il existe une projection de H dans G si et seulement si $\Psi(H)$ est une conséquence logique de $\Psi(G)$ et $\Psi(\mathcal{S})$.*

1.2.3 Nécessité de généralisation

Deux considérations m'ont amené à revoir le modèle des graphes conceptuels emboîtés :

FIG. 1.4 – Graphe conceptuel emboîté : "Une personne pense à quelque-chose."

Trop grande rigidité de la projection à la racine Le graphe de la figure 1.4 ne se projette ni dans le graphe de la figure 1.2 ni dans celui de la figure 1.3. Ceci est raisonnable dans le premier cas : il n'y a pas de personne qui pense dans le graphe, mais seulement la représentation d'une personne qui pense. Mais dans le second cas, il me semble plus hasardeux de répondre non à la requête : il y a bien une personne qui pense dans le monde représenté par le graphe.

La meilleure solution me semble être celle qui, alors qu'elle était proposée dans [MC96] [Sim96b], n'a été retenue dans aucune des sémantiques précédentes : il faudrait dans les deux cas répondre "oui, dans le contexte de [. . .]".

Une projection Π d'un graphe H dans un graphe G ne serait plus alors simplement la donnée des deux applications Π_C et Π_R, mais aussi celle du contexte de G dans laquelle elle s'applique.

Affiner le raisonnement (la projection) sur les graphes Il semble souhaitable de pouvoir représenter une même entité, une même instance d'un concept dans des contextes différents. On peut ainsi considérer les contextes comme différents "points de vue" sur ce concept. Encore faut-il pouvoir représenter l'"identité" entre ces concepts, et raisonner de façon correcte sur cette connaissance. C'est l'objectif du lien de co-référence, que je définirai de façon plus formelle au chapitre 2. Je me contenterai pour l'instant de dire que le lien de co-référence est une relation particulière entre deux sommets concepts de même type, symbolisée par un trait en pointillés, dont le but est d'indiquer que ces sommets sont "identiques". Notons tout de même que lorsque ces sommets sont des sommets individuels, le marqueur individuel suffit à indiquer qu'ils représentent la même entité.

Observons les deux graphes conceptuels emboîtés avec lien de co-référence des figures 1.5 et 1.6. Ces deux graphes ont exactement la même structure, et pourtant les déductions que l'on souhaiterait faire à partir de ces deux graphes sont -structurellement- très différentes. Dans le premier cas, le lien de co-référence ne permet de déduire rien d'autre que si ce lien n'y était pas : le bateau que pierre regarde n'est pas nécessairement rouge (on a pu le repeindre), et le bateau représenté dans le tableau n'est pas forcément gros (c'est une question d'échelle). On pourrait même imaginer que ces deux objets n'ont pas le même nom (seule une forme caractéristique permettant d'identifier le bateau représenté). Dans le deuxième cas, par contre, les deux bateaux devraient avoir même nom, mêmes attributs (couleur et taille), et être tous deux regardés par Pierre...

FIG. 1.5 – Exemple de graphe conceptuel emboîté avec lien de co-référence : Pierre posséde un tableau représentant la mer Méditerranée, sur laquelle navigue un bateau rouge. Pierre regarde un gros bateau, le même que celui représenté dans le tableau.

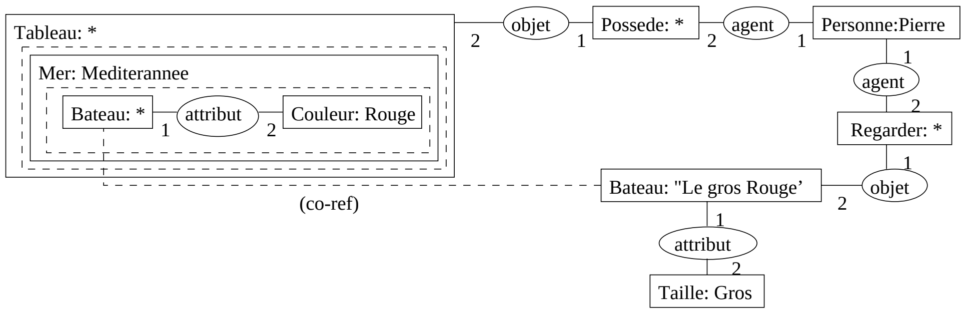
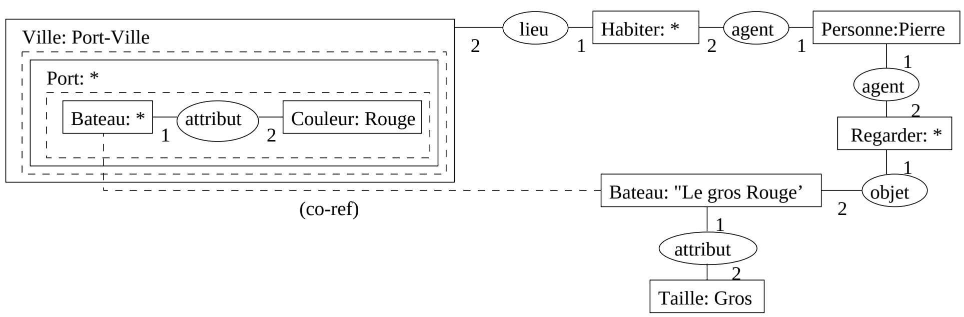


FIG. 1.6 – Un autre exemple de graphe conceptuel emboîté avec lien de co-référence : Pierre habite à Port-Ville, dont le port abrite un bateau rouge. Pierre regarde ce gros bateau.



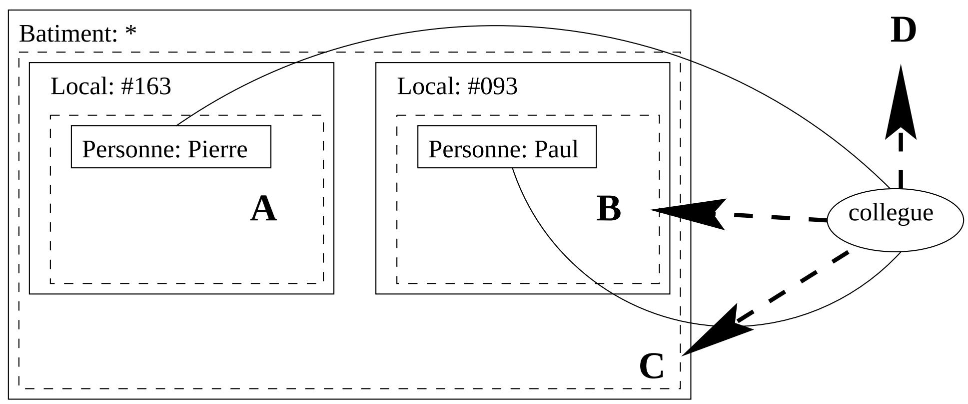
Sans nous attacher pour l'instant à formaliser le raisonnement sur *ces* identités, il semble désormais évident que ce sont deux liens de co-référence différents, qu'il faudra représenter différemment. Et ces deux co-références sont-elles suffisantes ? On pourrait imaginer toute une gamme de nuances allant de la représentation abstraite à l'identité forte, en passant par la représentation exacte, l'identité à un moment précis, ...

Une solution permettant d'éviter un traitement particulier pour chacune de ces versions de la co-référence semble être de considérer ces différents liens comme des relations "normales" du graphe (i.e. qui correspondent à un type de relation de $\mathcal{T}_R$), et de gérer leur fonctionnement spécifique par des règles de graphes conceptuels telles qu'elles sont traitées dans [Sal97].

Pour cela, il sera nécessaire de manipuler des graphes conceptuels emboîtés permettant de mettre en relation deux concepts n'appartenant pas au même contexte. Ceci n'est pas sans rappeler les libertés autorisées pour les "espaces" des réseaux partitionnés de Hendrix [Hen79].

Une question peut se poser : si on met en relation deux concepts n'étant pas dans le même contexte, alors quel est le contexte de la relation ? La figure 1.7 illustre la difficulté liée au plongement de la relation dans un contexte. Mettre le sommet relation "dans" un des locaux (A ou B) poserait un problème de symétrie : pourquoi privilégier un choix plutôt que l'autre ? "Dans" le batiment, la relation suggérerait que Pierre et Paul ne sont collègues que dans le contexte de ce batiment, ce qui n'a rien à voir. Enfin, plonger la relation dans le "contexte racine" revient à

FIG. 1.7 – Problème lié à la généralisation des relations dans un graphe emboîté : quel est le contexte d'une relation ?



ignorer le contexte de cette relation : c'est la solution que j'ai retenue. On pourra voir indifféremment les relations comme étant "hors-contexte" ou le contexte d'une relation comme le tuple formé des contextes de ses arguments.

1.3 Formalisation : les graphes de boîtes

Je propose donc une généralisation des graphes conceptuels emboîtés permettant d'avoir des relations entre des concepts situés dans des contextes différents : j'appellerai de tels graphes ¡¡ graphes de boîtes ¿¿ pour les différencier des graphes emboîtés tels qu'ils ont déjà été définis.

Après avoir défini ces graphes et leur projection, je leur donnerai une sémantique logique par rapport à laquelle la projection est adéquate et complète.

1.3.1 Graphes de boîtes et projection

Définition 7 (Boîtes) *La composante principale d'un graphe de boîtes est la boîte. Une boîte peut se définir inductivement de la façon suivante :*

(i) Une boîte est un ensemble de sommets concepts.

(ii) Tout sommet concept d'une boîte peut contenir un ensemble [7] de boîtes.

La figure 1.8 montre un exemple de boîte. Celle-ci comporte deux sommets concepts, dont l'un est étiqueté par Concept1, $*$) et l'autre par Concept2, A). Le premier contient lui-même deux boîtes, le second n'en contient aucune.

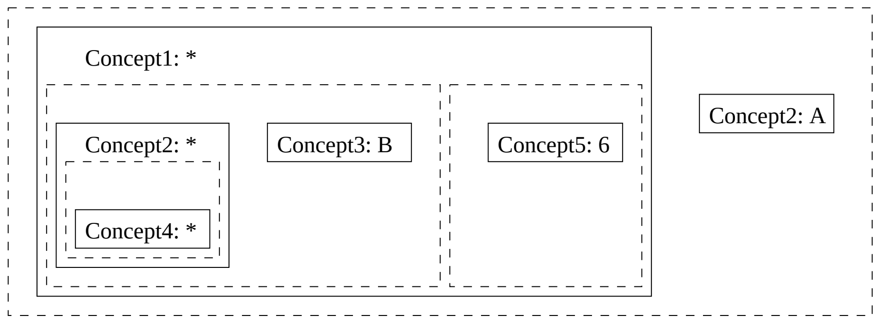
Définition 8 (Graphe de boîtes) *On peut définir un graphe de boîtes sur un support $\mathcal{S}$ de la manière suivante :*

(i) Un graphe de boîtes est un ensemble de boîtes.

(ii) Entre k sommets concepts quelconques d'un graphe de boîtes, on peut ajouter un sommet relation de degré k à condition de respecter la signature de cette relation.

[7] Un ensemble de boîtes – i.e. plusieurs contextes décrivant un même concept – ne semble pas d'une grande utilité. On pourrait donc contraindre davantage ces graphes de boîtes en imposant à un sommet concept de ne contenir au plus qu'une seule boîte. Cette définition prendra cependant tout son sens avec le typage des emboîtements.

FIG. 1.8 – Une boîte



Nous pouvons aussi voir un graphe de boîtes sous une autre forme, que j'appelle forme structurelle d'un graphe de boîtes. Cette forme structurelle rend, je pense, la définition de la projection plus claire et la démonstration de l'adéquation et de la complétude plus aisée.

Définition 9 (Forme structurelle d'un graphe de boîtes) *On peut représenter de façon équivalente un graphe de boîtes G sous une forme dite structurelle, qui est un graphe à trois types de sommets : les sommets concepts et les sommets relations de G reliés de la même façon que dans G par des arètes étiquetées, et un troisième type de sommets, appelés sommets contextes, qui symbolisent les boîtes.*

- *(i) A chaque boîte on associe un sommet contexte distinct.*
- *(ii) Si une boîte est immédiatement contenue dans un sommet concept C, alors on rajoute un arc qui va de C au sommet contexte représentant la boîte.*
- *(iii) Si un sommet concept C est immédiatement contenu dans une boîte, alors on rajoute un arc qui va du sommet contexte représentant la boîte au sommet C.*

Le sous-graphe engendré par les sommets concepts et les sommets contextes de la représentation structurelle est appelé squelette de cette représentation. Par un abus d'écriture, j'écrirai souvent squelette du graphe. De même, le sous-graphe engendré par les sommets concepts et les sommets relations sera appelé le sous-graphe relationnel.

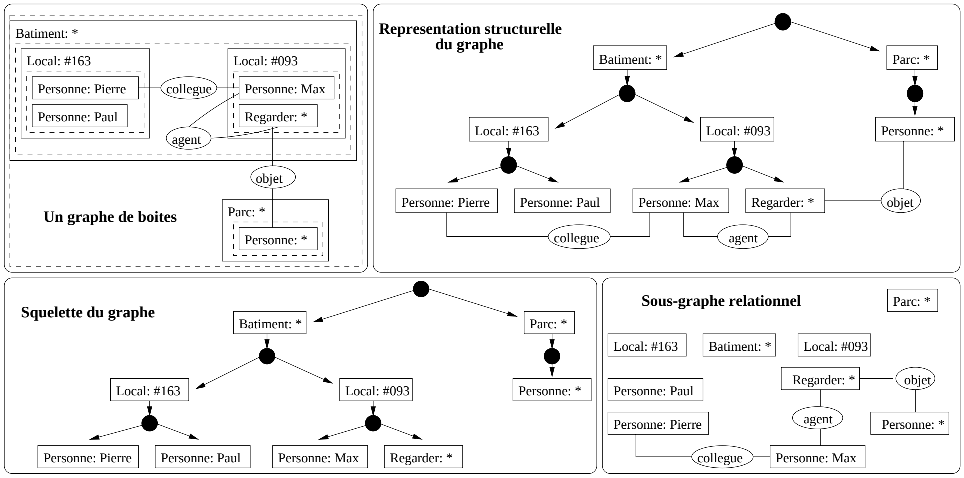
Le sous-graphe relationnel est un graphe simple. Le squelette est un ensemble d'arborescences biparties telles que toutes les racines et toutes les feuilles sont des sommets contextes. On peut se demander pourquoi un tel ensemble d'arborescences, alors qu'avec les graphes emboîtés, on se satisfait pleinement d'une racine unique... Ceci trouvera sa justification au chapitre 2. Jusque là, on pourra ne considérer que des graphes de boîtes standards, c'est à dire qui ne sont constitués que d'une seule boîte. Dans ce cas, on négligera souvent de représenter cette boîte, qui est implicite, dans le graphe.

La figure 1.9 nous montre un tel graphe de boîtes et sa représentation structurelle équivalente. On peut considérer un graphe de boîtes standard comme un graphe emboîté auquel on peut rajouter des relations portant sur des sommets qui ne sont pas dans le même contexte.

Il faut maintenant définir la projection sur ces graphes conceptuels.

Définition 10 (Projection de graphes de boîtes) *Soient G et H deux graphes de boîtes définis sur un support S, que l'on considère sous leur forme structurelle. Une projection Π de H dans G est un couple d'applications Π_S qui, à un sommet du squelette de H associe un sommet du squelette de G, et Π_R qui, à un sommet relation de H associe un sommet relation de G tel que :*

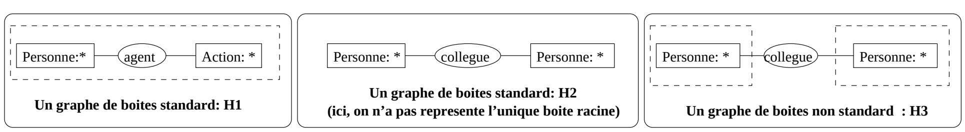
FIG. 1.9 – Représentation structurelle d'un graphe de boîtes, squelette et sous-graphe des relations.



- Π_S *associe à une racine de* H *un sommet contexte quelconque de* G
- *Si* (S, S') *est un arc du squelette de* H*, alors* $(\Pi_S(S), \Pi_S(S'))$ *est un arc du squelette de* G*.*
- *La donnée de* Π_R *et de la restriction de* Π_S *aux sommets concepts de* H *est une projection de graphes simples du sous-graphe relationnel de* H *dans le sous-graphe relationnel de* G

Il est important de voir que, contrairement à la projection dans les graphes emboîtés, la projection dans les graphes de boîtes standards ne se fait pas de la racine de H vers la racine de G, mais de la racine de H vers un contexte quelconque de G. Dans le cas général, la projection se fait de chaque racine de H dans un contexte quelconque de G.

FIG. 1.10 – Graphes de boîtes utilisés comme requêtes sur celui de la figure 1.9



Soient les graphes de la figure 1.10, que l'on essaie de projeter sur le graphe de la figure 1.9. H_1 se projette dans G, dans le contexte du local #093. H_2 ne se projette pas dans G, car les deux collègues sont dans des contextes différents de G. Par contre, H_3 se projette dans G, la première boîte se projetant dans le contexte du local #163, la deuxième dans le contexte du local #093.

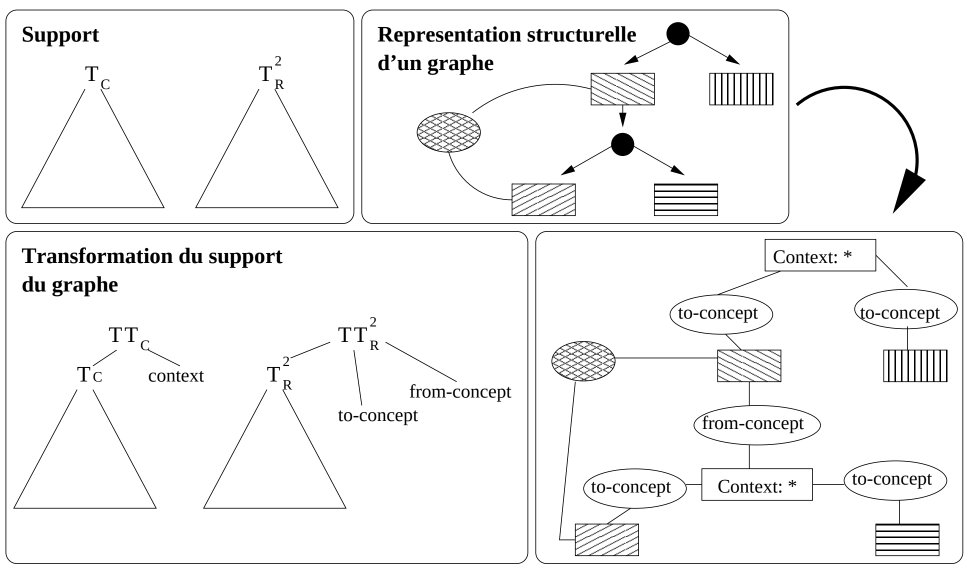
1.3.2 Sémantique logique, adéquation et complétude

Si G est un graphe de boîtes défini sur un support $\mathcal{S}$ et $\sigma(G)$ sa représentation structurelle, alors soit Θ l'application qui transforme G en un graphe conceptuel simple $\Theta(G)$ défini sur un support $\Theta(\mathcal{S})$. Cette application est définie de la façon suivante :

- Soit $\mathcal{T}_C$ l'ensemble partiellement ordonné des types de concepts de $\mathcal{S}$. On construit $\Theta(\mathcal{T}_C)$ en rajoutant à $\mathcal{T}_C$ un nouvel élément maximal, $\top\top_C$, et un nouveau type, CONTEXT, tel que $\top\top_C$ couvre CONTEXT qui est incomparable à tous les autres types.

- Soit $\mathcal{T}_R$ l'ensemble partiellement ordonné des types de relationss de $\mathcal{S}$. On construit $\Theta(\mathcal{T}_R)$ en rajoutant à $\mathcal{T}_R^2$ un nouvel élément maximal pour les relations d'arité 2, $\top\top_R^2$, et deux nouveaux types, FROM-CONCEPT et TO-CONCEPT, tels que $\top\top_R^2$ couvre FROM-CONCEPT et TO-CONCEPT qui sont incomparables à tous les autres types. Ces deux relations ont pour signature respective $(\top_C, \text{CONTEXT})$ et $(\text{CONTEXT}, \top_C)$.
- Les graphes G et H, que l'on considère sous leur forme structurelle, sont modifiés de la façon suivante : si u est un arc allant d'un sommet concept à un sommet contexte, alors on remplace cet arc par une relation FROM-CONCEPT entre ces deux sommets. Si v est un arc allant d'un sommet contexte à un sommet concept, alors on le remplace par une relation TO-CONCEPT entre ces deux sommets. Lorsqu'il n'y a plus de tels arcs dans le graphe, on remplace tous les sommets contextes par des sommets concepts de type CONTEXT et de marqueur $*$.

FIG. 1.11 – La transformation Θ d'un graphe de boîtes en graphe conceptuel simple



La figure 1.11 montre une telle transformation d'un graphe de boîtes en un graphe simple.

Proposition 1 (Equivalence pour la projection) *Il existe une projection de H dans G (que j'ai définie comme une projection de $\sigma(H)$ dans $\sigma(G)$), si et seulement si il existe une projection de graphes simples de $\Theta(H)$ dans $\Theta(G)$.*

Sémantique Φ associée à un graphe de boîtes L'interprétation du support se fait de la même façon que pour les graphes conceptuels simples. L'interprétation par Φ d'un graphe de boîtes G est définie comme l'interprétation par Φ du graphe simple $\Theta(G)$.

Par exemple, le graphe de boîtes de la figure 1.7 a pour interprétation :

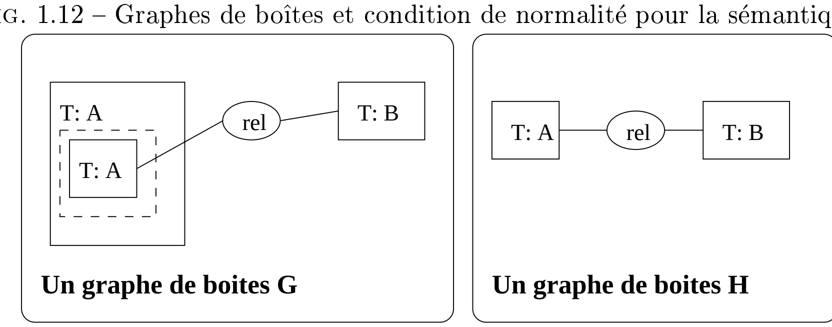
$$\begin{array}{rcl} \Phi(G) & = & \exists x_1 \exists x_2 \exists x_3 \exists x_4 \exists x_5 (\text{Contexte}(x_1) \\ & \wedge & \text{from-context}(x_1, x_2) \end{array}$$

$$\begin{array}{ll}
\wedge & \text{Batiment}(x_2) \\
\wedge & \text{to-context}(x_2, x_3) \\
\wedge & \text{Contexte}(x_3) \\
\wedge & \text{from-context}(x_3, \#163) \\
\wedge & \text{from-context}(x_3, \#093) \\
\wedge & \text{Local}(\#163) \\
\wedge & \text{Local}(\#093) \\
\wedge & \text{to-context}(\#163, x_4) \\
\wedge & \text{to-context}(\#093, x_5) \\
\wedge & \text{Contexte}(x_4) \\
\wedge & \text{Contexte}(x_5) \\
\wedge & \text{from-context}(x_4, Pierre) \\
\wedge & \text{from-context}(x_5, Paul) \\
\wedge & \text{Personne}(Pierre) \\
\wedge & \text{Personne}(Paul) \\
\wedge & \text{collegue}(Pierre, Paul))
\end{array}$$

Un graphe de boîtes G est sous forme normale si $\Theta(G)$ est un graphe simple sous forme normale.

Théorème 5 (Adéquation et complétude par rapport à Phi**)** *Soient G et H deux graphes de boîtes définis sur un support $\mathcal{S}$, G étant sous forme normale. Alors il existe une projection de H dans G si et seulement si $\Phi(H)$ est une conséquence logique de $\Phi(G)$ et $\Phi(\mathcal{S})$.*

FIG. 1.12 – Graphes de boîtes et condition de normalité pour la sémantique Φ



La condition de normalité sur le graphe simple associé par la transformation Θ suffit à assurer la complétude. Mais peut-on relâcher cette contrainte, par exemple en n'imposant la normalité que dans chaque boîte ? La réponse est malheureusement négative : la condition de normalité ¡¡ par niveau d'emboîtement ¿¿ ne suffit pas à assurer la complétude.

La figure 1.12 montre deux graphes de boîtes tels qu'il n'y a pas de projection de l'un dans l'autre. Pourtant, l'interprétation de H se déduit de l'interprétation de G :

$$\Phi(G) = \exists x_1 \exists x_2 (\text{Context}(x_1) \wedge \text{from-context}(x_1, A) \wedge \text{from-context}(x_1, B) \wedge \top(A)$$

$$\wedge \top(B) \wedge \text{to-context}(A, x_2) \wedge \text{Context}(x_2) \wedge \text{from-context}(x_2, A) \wedge \top(A) \wedge \text{rel}(A, B))$$

$$\Phi(H) = \exists y_1(\text{Context}(y_1) \wedge \text{from-context}(y_1, A) \wedge \text{from-context}(y_1, B) \wedge \text{rel}(A, B))$$

Sémantique Ψ associée à un graphe de boîtes L'interprétation du support se fait de la même façon que pour les graphes conceptuels simples. L'interprétation par Ψ d'un graphe de boîtes G est définie comme l'interprétation par Ψ du graphe simple $\Theta(G)$.

Par exemple, le graphe de boîtes de la figure 1.7 a pour interprétation :

$$\begin{array}{rcl}
\Psi(G) & = & \exists x_1 \exists x_2 \exists x_3 \exists x_4 \exists x_5 \exists y_1 \exists y_2 \exists y_3 \exists y_4 \exists y_5 \exists y_6 \exists y_7 \exists y_8 \exists y_9 (\text{Contexte}(y_1, x_1) \\
& \wedge & \text{from-context}(y_1, y_2) \\
& \wedge & \text{Batiment}(y_2, x_2) \\
& \wedge & \text{to-context}(y_2, y_3) \\
& \wedge & \text{Contexte}(y_3, x_3) \\
& \wedge & \text{from-context}(y_3, y_4) \\
& \wedge & \text{from-context}(y_3, y_5) \\
& \wedge & \text{Local}(y_4, \#163) \\
& \wedge & \text{Local}(y_5, \#093) \\
& \wedge & \text{to-context}(y_4, y_6) \\
& \wedge & \text{to-context}(y_5, y_7) \\
& \wedge & \text{Contexte}(y_6, x_4) \\
& \wedge & \text{Contexte}(y_7, x_5) \\
& \wedge & \text{from-context}(y_6, y_8) \\
& \wedge & \text{from-context}(y_7, y_9) \\
& \wedge & \text{Personne}(y_8, Pierre) \\
& \wedge & \text{Personne}(y_9, Paul) \\
& \wedge & \text{collegue}(y_8, y_9))
\end{array}$$

Théorème 6 (Adéquation et complétude par rapport à Psi) *Soient G et H deux graphes de boîtes définis sur un support $\mathcal{S}$, sans condition de normalité. Alors il existe une projection de H dans G si et seulement si $\Psi(H)$ est une conséquence logique de $\Psi(G)$ et $\Psi(\mathcal{S})$.*

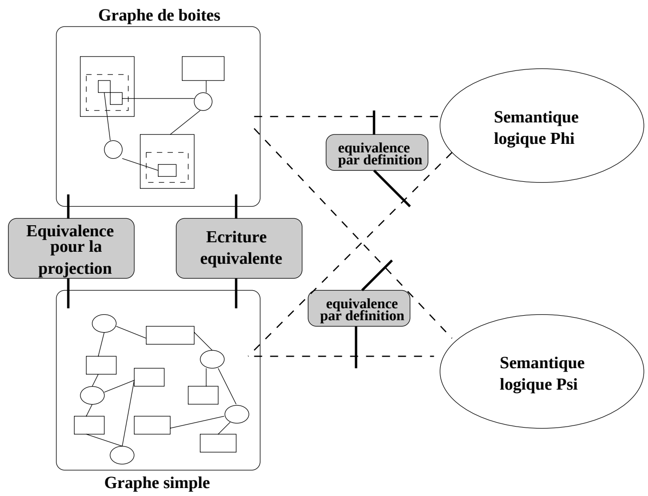
Les théorèmes 5 et 6 se démontrent de façon identique : il y a une projection de H dans G si et seulement si il y a une projection de $\Theta(H)$ dans $\Theta(G)$, i.e. si on peut déduire l'interprétation de H (que ce soit par Φ ou par Ψ) de l'interprétation du support et de l'interprétation de G. C'est cette équivalence de problèmes qui est représentée à la figure 1.13.

1.3.3 Extension aux graphes emboîtés typés

Comme le montre la figure 1.14, le typage des emboîtements permet une représentation plus fine des connaissances exprimées par le graphe. Cette "valeur ajoutée" devra bien sûr être exploitée par les règles de graphes conceptuels.

Le travail précédent facilite grandement l'extension des graphes de boîtes aux graphes de boîtes typées. Un bon résumé de la prose qui s'ensuit serait : on n'a plus qu'à typer les sommets contextes, et à autoriser plusieurs contextes dans un même sommet.

FIG. 1.13 – Graphes de boîtes : adéquation et complétude par rapport à la sémantique Φ et Ψ



Définition 11 (Support d'un graphes de boîtes typées) *On rajoute au support tel qu'il est défini un ensemble partiellement ordonné des types d'emboîtement, $\mathcal{T}_X$, qui admet un élément maximal $\top_X$.*

Définition 12 (Graphes de boîtes typées) *Un GBT G est un graphes de boîtes dont tous les sommets contextes (sous forme structurelle, sinon lire ¡¡ toutes les boîtes ¿¿)sont étiquetés par un type d'emboîtement. Le sommet contexte racine est étiqueté par $\top_X$.*

Définition 13 (Projection de GBT) *Soient G et H deux GBT définis sur un support $\mathcal{S}$. Une projection de H dans G est une projection de graphes de boîtes $\Pi = (\Pi_S, \Pi_R)$ de H dans G telle que pour tout sommet contexte S de H, type$(\Pi_S(S)) \leq_X$ type(S).*

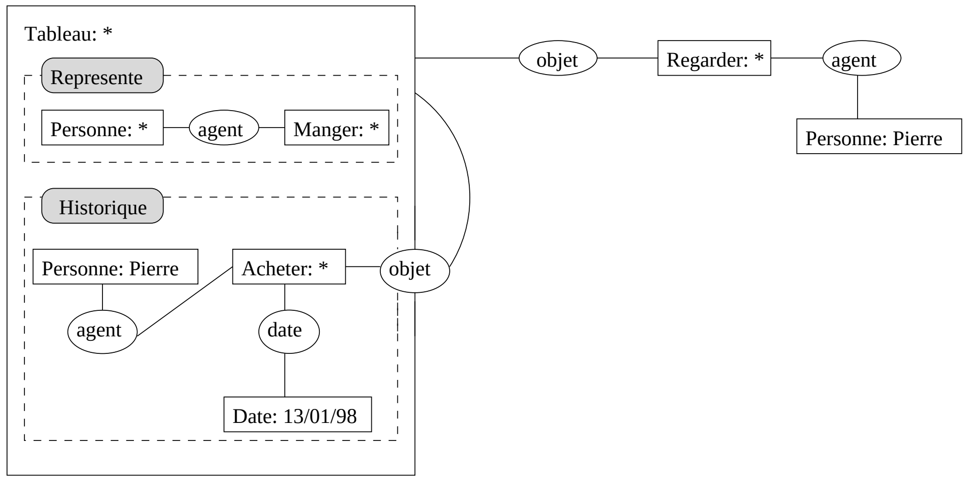
J'étends l'application Θ à la transformation d'un GBT et de son support en graphe simple de la manière suivante :
– Le nouvel ensemble des types de concepts est construit sur $\mathcal{T}_C$ et $\mathcal{T}_X$ en rajoutant un nouvel élément maximal $\top\top_C$ qui couvre $\top_C$ et $\top_X$.
– La signature des relations FROM-CONCEPT et TO-CONCEPT est respectivement $(\top_C, \top_X)$ et $(\top_X, \top_C)$.
– Après la transformation des arcs du squelette, la transformation des sommets contextes se fait en les remplaçant par un sommet concept dont le type est le nouveau type de concept associé au type du sommet contexte, et dont le marqueur est $*$.

La figure 1.15 illustre cette transformation.

La sémantique Φ ou Ψ associée à un GBT est celle de son graphe simple associé. Nous avons de la même manière que pour les graphes de boîtes l'adéquation et la complétude de la projection par rapport à la sémantique Φ (avec la même condition de normalité) et à la sémantique Ψ (sans aucune restriction)...

J'ai proposé dans ce chapitre une version ¡¡ étendue ¿¿ des graphes emboîtés, à laquelle correspond une sémantique logique équivalente – à un changement de support près – à celle d'un graphe conceptuel simple associé. Il y a deux intérêts à une telle formalisation :

FIG. 1.14 – Typage des emboîtements : Pierre regarde un tableau représentant une personne en train de manger. Il a acheté ce tableau le 13 janvier 1998



- Théorique : tous les résultats théoriques sur les graphes simples s'étendent sans difficulté aux graphes de boîtes, et par là aux graphes de boîtes typées.
- Pratique : tous les algorithmes utilisés dans CoGiTo pour la projection de graphes simples restent valides pour les graphes de boîtes et les graphes de boîtes typées, il n'y a qu'à écrire la transformation en graphe simple...

À propos de ce deuxième point, un petit bémol toutefois. L'emboîtement entraine une structuration supplémentaire, et il peut être intéressant d'un point de vue algorithmique de profiter de cette structuration. Une idée naturelle est de préfiltrer la recherche d'une projection de H dans G par un calcul des projections du squelette de H dans le squelette de G. Ce préfiltrage peut donner d'excellents résultats si le nombre de boîtes n'est pas trop petit devant le nombre de sommets concepts. Trop peu d'utilisations ¡¡ de grande taille ¿¿ des graphes conceptuels ont été faites pour savoir si cette hypothèse est réaliste ou non.

On peut voir les graphes conceptuels emboîtés comme une généralisation non stricte des graphes simples :

- Enrichissement de la représentation : les graphes simples sont un cas particulier des graphes emboîtés, telle que la restriction de la projection de graphes emboîtés aux graphes simples est équivalente à la projection de graphes simples.
- Pas d'enrichissement du modèle : les graphes emboîtés peuvent se représenter comme des graphes simples, de façon à ce que toute projection de graphes emboîtés soit équivalente à la projection des graphes simples associés.

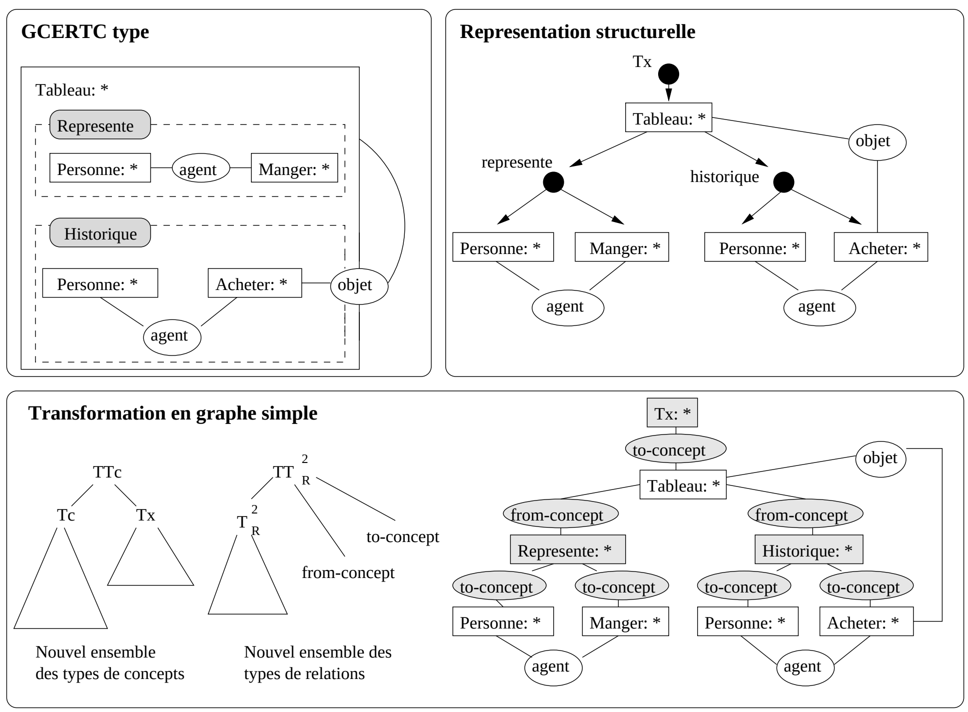
De même, on peut voir les graphes de boîtes comme une généralisation non stricte des graphes emboîtés.

Les modifications apportées par le modèle des graphes de boîtes

- relations entre concepts situés dans des contextes différents
- pas de racine unique pour les contextes
- projection d'une racine dans un contexte quelconque

trouveront leur justification et leur intérêt au chapitre suivant.

FIG. 1.15 – Un GBT, sa représentation structurelle et sa transformation par Θ en graphe simple



Chapitre 2

Règles de graphes conceptuels et traitement de la co-référence

Eric Salvat [SM96] [Sal97] a muni les graphes conceptuels simples d'un ensemble de règles, de types règles d'inférence, permettant de transformer ces graphes. Le problème de l'existence d'une projection d'un graphe H dans un graphe G (qui est NP-complet dans le cas des graphes conceptuels simples [CM92]), devient donc : existe-t-il une séquence d'applications des règles au graphe construisant un graphe G' telle que H se projette dans G'.

Après avoir défini ces règles, je montrerai comment on peut s'en servir pour représenter et raisonner sur l'identité de sommets concepts, cette identité étant représentée par un sommet relation. Je montrerai comment de telles règles peuvent simuler la co-référence telle que définie dans [Sim96a], et peuvent même servir de base pour des déductions qui semblent ¡¡ naturelles ¿¿, mais que la co-référence ne permettait pas.

Cette gestion des conséquences de l'identité m'amènera à considérer des types de concepts construits, qui s'interprètent comme la conjonction des types de concepts primitifs définis dans le support. Ceci me permettra de modifier dynamiquement, lors de l'application des règles, les types des sommets du graphe.

2.1 Règles de graphes conceptuels

2.1.1 Application d'une règle à un graphe

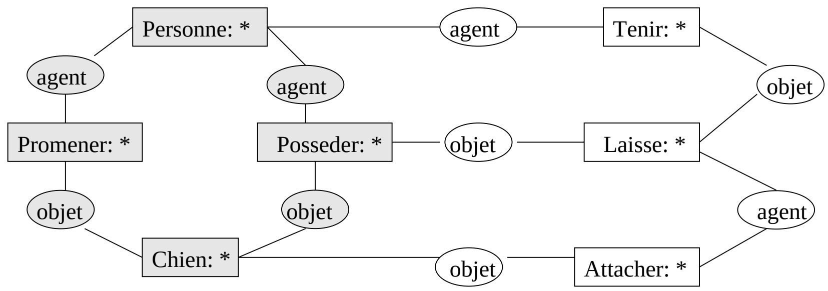
Les règles de graphes conceptuels servent à formaliser des connaissances du type ¡¡ si A alors B ¿¿, et sont donc capitales pour pouvoir factoriser les connaissances.

Je commence par donner la définition d'une règle de graphes conceptuels simples, équivalente à celle utilisée dans [SM96] [Sal97], mais que je trouve plus ¡¡ pratique ¿¿ à utiliser, car basée uniquement sur des opérations de graphes, sans avoir à manipuler les ¡¡ lambda - abstractions ¿¿ définies par E. Salvat.

Définition 14 (Règle de graphes conceptuels simples) *On peut définir une règle de graphe conceptuel simple R comme un graphe conceptuel simple bicolore (i.e. on a attribué deux couleurs aux sommets de R), défini sur un support $\mathcal{S}$, tel que le sous-graphe P_R de R engendré par les sommets de la première couleur (aussi appelé précondition de R, et noté précond(R)) est un graphe conceptuel simple.*

Le sous-graphe C_R engendré par les sommets de la deuxième couleur est appelé conclusion de R, et est noté concl(R).

FIG. 2.1 – Un exemple de règle de graphe conceptuel simple : ¡¡ Si une personne promène son chien, alors elle le tient avec une laisse qu'elle possède ¿¿.



La figure 2.1 donne un exemple d'une telle règle de graphe conceptuel simple, il reste à définir comment on l'utilise, c'est-à-dire comment on applique une telle règle sur un graphe.

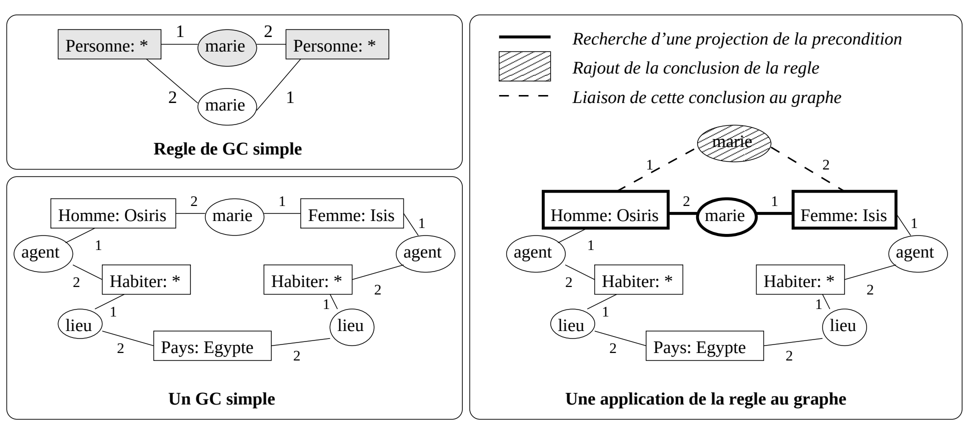
Définition 15 (Application d'une règle à un graphe) *On dit qu'une règle R est applicable à un graphe G si et seulement si il existe une projection de précond(R) dans G.*

Une projection Π de précond(R) dans G définit un graphe conceptuel simple $G' = \mathcal{A}_R(\Pi, G)$ qui est le résultat de l'application de R à G (selon la projection Π), et se définit de la façon suivante :

(i) On fait d'abord l'union disjointe de G et concl(R).

(ii) Pour toute arête d'étiquette i de R entre un sommet S_P de précond(R) et un sommet S_C de concl(R), on rajoute une arête d'étiquette i dans G' entre $\Pi(S_P)$ et S_C.

FIG. 2.2 – Un exemple d'application de la règle ¡¡ Si A est marié à B, alors B est marié à A ¿¿ sur un graphe conceptuel



La figure 2.2 montre comment se construit le graphe obtenu par l'application d'une règle [1] suivant une certaine projection. Cette règle suggère d'ailleurs l'idée

[1] Il aurait été plus élégant de rajouter une ¡¡ relation-symétrique ¿¿ majorant ¡¡ marié ¿¿ dans le support ...

d'une séquence d'applications de celle-ci ...

2.1.2 Sémantique logique

Une règle de graphe conceptuel simple a une sémantique logique en logique du premier ordre, c'est celle de l'implication. La formule logique associée à une règle de graphes conceptuels est obtenue de la manière suivante :

- Soient $\{P_i\}$ les formules atomiques associées aux sommets $\{S_i\}$ de la règle, considérée comme un graphe conceptuel simple.
- Une formule P_i est une formule de précondition si elle est obtenue à partir d'un sommet S_i de la précondition de la règle, sinon, c'est une formule de conclusion.
- L'interprétation de la règle est la formule :

$$(\forall x_1 \ldots \forall x_2 (P_{i_1} \wedge P_{i_r})) \rightarrow (\exists y_1 \ldots \exists y_n (P_{j_1} \wedge P_{j_q}))$$

  où les P_{i_k} sont les formules de précondition de la règle, les P_{j_k} sont les formules de conclusion de la règle, x_i les variables apparaissant dans les P_{i_k} et y_i les variables apparaissant dans les P_{j_k} mais pas dans les P_{i_k}.

L'interprétation des sommets peut se faire avec la sémantique Φ ou la sémantique Ψ. Je considère jusqu'à mention du contraire qu'on ne travaille que sur des graphes simples, ce qui est suffisant (voir chapitre 1).

Ainsi, la règle de la figure 2.1 s'interprète , avec la sémantique Φ, par la formule :

$$\begin{array}{ll}
(\forall x_1 \forall x_2 \forall x_3 \forall x_4 & (\ \mathrm{Personne}(x_1) \\
& \wedge\ \mathrm{Promener}(x_2) \\
& \wedge\ \mathrm{Posseder}(x_3) \\
& \wedge\ \mathrm{Chien}(x_4) \\
& \wedge\ \mathrm{agent}(x_1, x_2) \\
& \wedge\ \mathrm{agent}(x_1, x_3) \\
& \wedge\ \mathrm{objet}(x_2, x_4) \\
& \wedge\ \mathrm{objet}(x_3, x_4))) \\
\rightarrow\ (\exists x'_1 \exists x'_2 \exists x'_3 (\ \mathrm{Tenir}(x'_1) & \\
& \wedge\ \mathrm{Laisse}(x'_2) \\
& \wedge\ \mathrm{Attacher}(x'_3) \\
& \wedge\ \mathrm{agent}(x_1, x'_1) \\
& \wedge\ \mathrm{objet}(x_3, x'_2) \\
& \wedge\ \mathrm{objet}(x'_3, x_4) \\
& \wedge\ \mathrm{objet}(x'_1, x'_2) \\
& \wedge\ \mathrm{objet}(x'_2, x'_3)))
\end{array}$$

Et avec la sémantique Ψ, par la formule :

$$\begin{array}{ll}
(\forall x_1 \forall x_2 \forall x_3 \forall x_4 \forall y_1 \forall y_2 \forall y_3 \forall y_4 & (\ \mathrm{Personne}(x_1, y_1) \\
& \wedge\ \mathrm{Promener}(x_2, y_2) \\
& \wedge\ \mathrm{Posseder}(x_3, y_3) \\
& \wedge\ \mathrm{Chien}(x_4, y_4)
\end{array}$$

$$
\begin{array}{lll}
 & & \wedge\ \mathrm{agent}(x_1, x_2) \\
 & & \wedge\ \mathrm{agent}(x_1, x_3) \\
 & & \wedge\ \mathrm{objet}(x_2, x_4) \\
 & & \wedge\ \mathrm{objet}(x_3, x_4))) \\
\rightarrow & (\exists x'_1 \exists x'_2 \exists x'_3 \exists y'_1 \exists y'_2 \exists y'_3 & (\ \mathrm{Tenir}(x'_1, y'_1) \\
 & & \wedge\ \mathrm{Laisse}(x'_2, y'_2) \\
 & & \wedge\ \mathrm{Attacher}(x'_3, y'_3) \\
 & & \wedge\ \mathrm{agent}(x_1, x'_1) \\
 & & \wedge\ \mathrm{objet}(x_3, x'_2) \\
 & & \wedge\ \mathrm{objet}(x'_3, x_4) \\
 & & \wedge\ \mathrm{objet}(x'_1, x'_2) \\
 & & \wedge\ \mathrm{objet}(x'_2, x'_3)))
\end{array}
$$

2.1.3 $\mathcal{R}$-projection

Définition 16 ($\mathcal{R}$-projection) *Soit G un graphe conceptuel simple, muni d'un ensemble de règles $\mathcal{R} = \{R_1, \ldots, R_k\}$. On appelle construction immédiate maximale sur G selon $\mathcal{R}$ et on note $\aleph_{\mathcal{R}}(G)$ le graphe obtenu à partir de G par l'application sur G de chacune des R_i suivant toutes les projections $\Pi_{i,j}$ de précond(R_i) sur G.*

On appelle graphe potentiel de G au rang n selon $\mathcal{R}$ le graphe défini inductivement par :

- $\aleph^0_{\mathcal{R}}(G) = G$
- $\aleph^{n+1}_{\mathcal{R}}(G) = \aleph_{\mathcal{R}}(\aleph^n_{\mathcal{R}}(G))$

On dit qu'il y a une $\mathcal{R}$-projection de H dans G si et seulement si il existe n tel que H se projette dans $\aleph^n_{\mathcal{R}}(G)$.

Voir que, pour la construction de $\aleph^k_{\mathcal{R}}(G)$, on n'a pas besoin d'effectuer les applications de la règle sur les mêmes projections que celles utilisées pour la construction de $\aleph^{k-1}_{\mathcal{R}}(G)$, car on ne construirait ainsi que de l'information redondante.

Cette définition n'étant en fait qu'une reformulation du chaînage avant défini par Eric Salvat [Sal97], le problème de la projection dans les graphes simples devient :

PROBLÈME : $\mathcal{R}$-PROJECTION DE GC DONNÉES : Un graphe conceptuel simple G, un ensemble de règles de graphes conceptuels $\mathcal{R}$, un graphe conceptuel simple H. QUESTION : Existe-t-il une $\mathcal{R}$-projection de H dans G ?

Ce problème peut bien entendu être considéré de façon équivalente en termes de déduction logique :

Théorème 7 (Adéquation et complétude) *Soient G et H deux graphes conceptuels définis sur un support $\mathcal{S}$, $\mathcal{R} = \{R_1, \ldots R_k\}$ un ensemble de règles. Alors il existe une $\mathcal{R}$-projection de H dans G si et seulement si $\Phi(\mathcal{S}), \Phi(G), \Phi(R_1), \ldots, \Phi(R_k) \vDash \Phi(H)$.*

Ce théorème découle de l'adéquation et de la complétude du chaînage avant [Sal97].

Par analogie avec les résultats obtenus sur des sous-ensembles similaires de la logique du premier ordre, ce problème est conjecturé comme semi-décidable.

2.2 La co-référence comme représentation de l'identité

Il peut être utile dans un graphe conceptuel de représenter par deux sommets distincts un même individu, une même instance d'un concept.

Mais au delà de cette assertion ¡¡ deux sommets distincts du graphe représentent le même individu ¿¿, il faut savoir quelles sont les déductions possibles à partir de cette assertion, i.e. quel doit être le traitement de la projection sur des sommets co-référents ?

Je commencerai par présenter la co-référence telle qu'elle a été définie au LIRMM [Sim96b] [Sim96a], puis après avoir proposé une typologie des conséquences possibles de la co-référence, je présenterai une méthode pour son traitement basée sur les règles de graphes conceptuels.

2.2.1 Liens de co-référence pour les graphes simples

Un graphe conceptuel (simple ou emboîté) avec liens de co-référence est un graphe conceptuel auquel on associe une relation d'équivalence *co-ref* sur des sommets concepts génériques ayant même type. L'interprétation intuitive de cette relation de co-référence est ¡¡ ces sommets représentent la même entité, la même instance ¿¿.

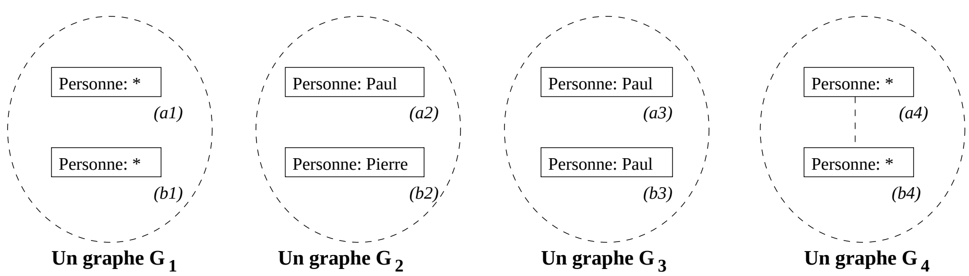
Cette relation est traditionnellement représentée dans le dessin d'un graphe par un trait en pointillés entre des sommets co-référents. Comme l'identité de marqueur est aussi une façon de représenter le fait que deux sommets représentent la même entité, Geneviève Simonet [Sim96a] définit une relation plus générale, dite de co-identité, par :

$$\text{co-ident}(C, C') \text{ ssi } (\text{co-ref}(C, C') \text{ ou } m = m')$$

où C et C' sont deux sommets concepts de marqueurs individuels respectifs m et m'.

Une projection d'un graphe conceptuel avec liens de co-référence H dans G est une projection de graphe conceptuel qui respecte la co-identité, i.e. si C et C' sont deux sommets co-référents de H, alors $\Pi(C)$ et $\Pi(C')$ sont deux sommets co-identiques de G. Autrement dit, deux sommets co-référents ont même image, ou des images co-référentes, ou des images ayant même marqueur individuel.

FIG. 2.3 – Exemples de graphes conceptuels simples avec liens de co-référence



Prenons comme exemples les graphes de la figure 2.3.

- G_4 ne se projette dans G_1 que par la projection de $a4$ et b_4 dans a_1 ou la projection de a_4 et b_4 dans b_1, car les deux personnes de G_1 ne sont, ni *co-ref*, ni *co-ident*

– Il y a deux projections possibles de G_4 dans G_2 : les deux personnes de G_4 se projettent sur Pierre, ou les deux personnes se projettent sur Paul
– Il y a 4 projections possibles de G_4 dans G_3 : les deux personnes de G_4 se projetant sur le premier sommet Paul, ou sur le deuxième sommet Paul, ou a_4 sur a_3 et b_4 sur b_3, ou a_4 sur b_3 et b_4 sur a_3, ces projections étant possibles puisque, a_3 et b_3 ayant même marqueur, ils sont *co-ident*.

L'interprétation d'un graphe conceptuel (simple ou emboîté) se fait par la sémantique Φ ou Ψ, modifiées de la façon suivante : soient C et C' sont deux sommets concepts co-référents de type t

– Sémantique Φ : ils sont interprétés par la même formule atomique $t(y)$, où y est la variable associée à la classe d'équivalence de C et C'.
– Sémantique Ψ : ils sont interprétés par les formules $t(x_1, y)$ et $t(x_2, y)$ où x_1 et x_2 sont les variables associées à ces deux sommets physiques et y est la variable associée à la classe d'équivalence de C et C'.

On peut ainsi retrouver les résultats d'adéquation et de complétude de la projection avec co-référence par rapport à ces deux sémantiques [Sim96a].

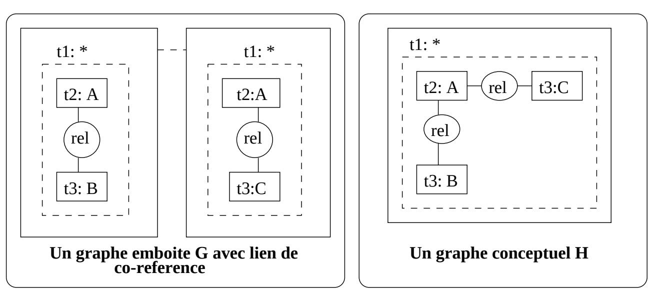
La forme normale pour les graphes simples avec liens de co-référence est obtenue en fusionnant tous les sommets concepts appartenant à la même classe de co-identité.

Théorème 8 (Adéquation et complétude pour les graphes simples) *Soient G et H deux graphes conceptuels simples avec liens de co-référence définis sur un support $\mathcal{S}$.*
– *G étant un graphe sous forme normale, il existe une projection de H dans G si et seulement si on peut déduire $\Phi(H)$ de $\Phi(G)$ et $\Phi(\mathcal{S})$.*
– *Sans condition de normalité, il existe une projection de H dans G si et seulement si on peut déduire $\Psi(H)$ de $\Psi(G)$ et $\Psi(\mathcal{S})$.*

2.2.2 Liens de co-référence pour les graphes emboîtés

La relation de co-référence s'étend naturellement aux graphes emboîtés : on peut représenter de la même façon que deux sommets concepts génériques, ayant même type, sont co-référents.

FIG. 2.4 – Exemples de graphes conceptuels emboîtés avec liens de co-référence



La figure 2.4 montre deux graphes conceptuels emboîtés avec liens de co-référence. On n'a pas de projection de H dans G, et pourtant $\Phi(H)$ se déduit de $\Phi(G)$.

Pour conserver les résultats d'adéquation et de complétude de la projection par rapport à la sémantique Φ, Geneviève Simonet a défini une condition de k-normalité [Sim96a]. Je n'entrerai pas dans les détails de sa démonstration et présenterai juste les principaux résultats.

Définition 17 *Soit G un graphe conceptuel emboîté avec liens de co-référence. On dit que G est sous forme k-normale si :*

(i) Tous les graphes simples intervenant dans la construction de G sont sous forme normale.

(ii) Si C et C' sont deux sommets concepts co-identiques *situés dans des contextes différents, tels que la description de C n'est pas vide, alors si C' est dans un contexte de profondeur k par rapport au contexte racine :*
- *La description de C' n'est pas vide*
- *Le graphe de base constituant la description de C est exactement équivalent au graphe de base constituant la description de C'*

Avec cette condition de normalité, on peut déduire les résultats d'adéquation et de complétude suivants :

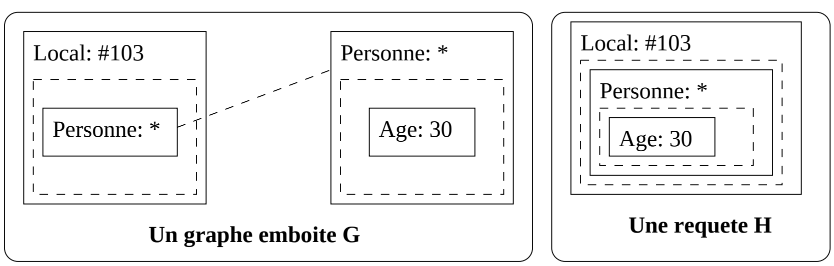
Théorème 9 (Adéquation et complétude pour les graphes emboîtés) *Soient G et H deux graphes conceptuels emboîtés avec liens de co-référence définis sur un support $\mathcal{S}$.*
- *G étant un graphe sous forme k-normale, où $k = p \div 2$, p étant la hauteur du squelette de G, il existe une projection de H dans G si et seulement si on peut déduire $\Phi(H)$ de $\Phi(G)$ et $\Phi(\mathcal{S})$.*
- *Sans condition de normalité, il existe une projection de H dans G si et seulement si on peut déduire $\Psi(H)$ de $\Psi(G)$ et $\Psi(\mathcal{S})$.*

2.2.3 La co-référence suffit-elle ?

On peut souhaiter pouvoir déduire certaines informations de l'identité entre concepts. Le lien de co-référence assure seulement la projection de sommets concepts co-identiques dans une même classe d'équivalence de concepts co-identiques. Pourtant, intuitivement, affirmer que deux individus, deux instances d'un concept, sont identiques devrait apporter plus d'information que ceci.

J'ai essayé d'établir une typologie des différentes déductions que l'on peut vouloir faire à partir de l'identité entre deux concepts.

FIG. 2.5 – Peut-on déduire la description d'un sommet concept de celle de son co-référent ?

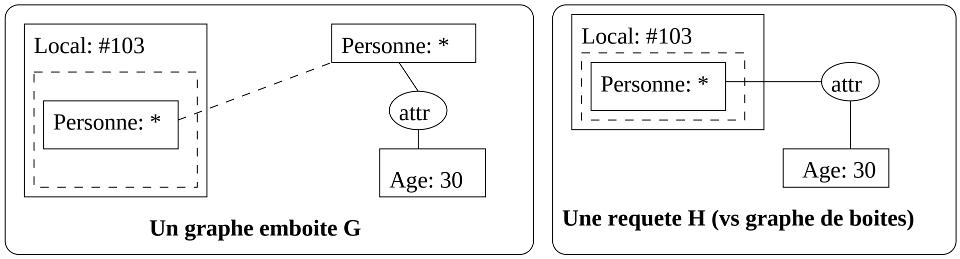


Partage de la description Si deux sommets concepts sont déclarés identiques, alors chacun devrait pouvoir ¡¡ hériter ¿¿ de la description de l'autre. Ainsi, dans la figure 2.5, je voudrais pouvoir projeter le graphe H dans le graphe G.

Pourtant, le graphe H de la figure 2.5 ne se projette pas dans le graphe G. En mettant le graphe G sous sa forme k-normale G' pour la sémantique Φ, on a une projection de H dans G'.

Nous avons donc une sémantique Φ pour laquelle le partage de la description se fait, à une modification du graphe près, et une sémantique Ψ pour laquelle ce partage de description n'a pas lieu.

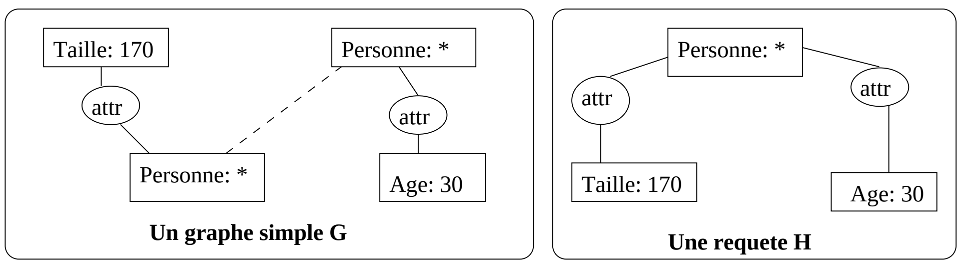
FIG. 2.6 – Peut-on déduire les relations sur un sommet concept de celles de son co-référent ?



Partage des relations De même, si deux sommets concepts sont identiques, on peut vouloir en déduire que tout ce qui est en relation avec l'un est en relation avec l'autre.

Pourtant, le graphe H de la figure 2.6 ne se projette pas dans le graphe G, et ceci même avec la sémantique Φ, le graphe G étant déjà sous forme k-normale.

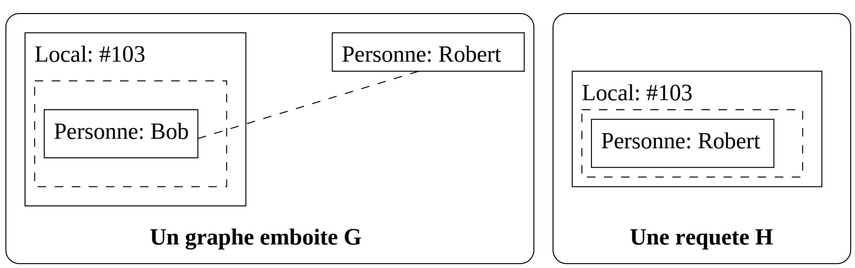
FIG. 2.7 – Peut-on déduire les relations sur un sommet concept de celles de son co-référent ? – Version graphes simples



De même, le graphe H de la figure 2.7 ne se projette pas dans le graphe G. Pourtant, H se projette dans la forme normale de G.

Le partage des relations par les sommets concepts co-référents s'effectue donc parfois (i.e. dans le même niveau d'emboîtement) avec la sémantique Φ, et jamais avec la sémantique Ψ.

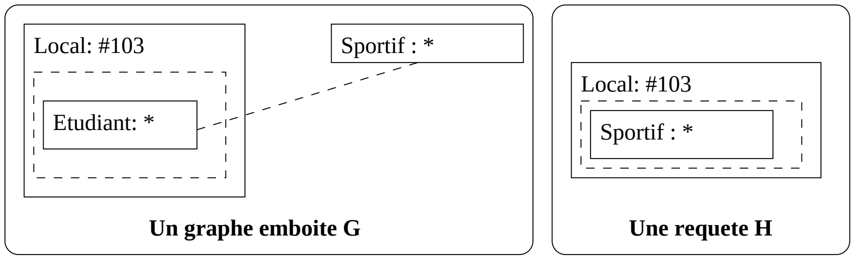
FIG. 2.8 – Quel est le rôle du marqueur dans l'identité entre concepts ?



Rôle du marqueur On pourrait considérer que deux sommets concepts co-identiques peuvent avoir des marqueurs différents. Cela reviendrait à gérer des classes d'équivalences de marqueurs individuels, la co-identité étant assurée par l'appartenance des marqueurs à la même classe d'équivalence.

Cet ¡¡ enrichissement ¿¿ du modèle ne poserait aucun problème d'un point de vue théorique, reste à voir si il offre un quelconque intérêt. Dans l'état actuel des choses, le graphe G de la figure 2.8 n'est pas un graphe conceptuel valide. Aussi, je n'en parlerai plus au cours de ce mémoire.

FIG. 2.9 – Quel est le rôle du type dans l'identité entre concepts ?



Déductions sur les types Le graphe G de la figure 2.9 n'est pas non plus un graphe conceptuel valide. Mais si on relâche la contrainte d'identité de type pour des sommets concepts co-référents, on voit bien le problème qui se pose : peut-on déduire un sous-type commun (i.e. la conjonction des types) de la co-référence portant sur des sommets concepts ayant des types différents ? Ce problème sera examiné plus attentivement en section 2.4.

2.3 Simulation de la co-référence par des règles

Je montre tout d'abord comment on peut simuler la co-référence telle qu'elle a été définie dans [Sim96b] [Sim96a] [CMS98] avec des règles de graphes conceptuels. Ensuite, je montrerai comment on peut compléter de telles règles afin de pouvoir établir les déductions présentées précedemment.

On voit ici pourquoi j'ai eu besoin des graphes de boîtes : les liens de co-référence reliant deux sommets du dessin du graphe vont être remplacés par une relation, qui est un objet du graphe, et qui devra, comme le lien de co-référence, mettre en relation des sommets concepts qui ne sont pas dans le même contexte.

Cette simulation par des règles offre deux intérêts :

- La relation de co-référence devient ainsi un objet du modèle. Il me semble souhaitable d'éviter, dans la mesure du possible, d'apporter un traitement *ad-hoc* pour chaque ¡¡ enrichissement ¿¿ souhaitable du modèle. Il est préférable de n'avoir que des ¡¡ objets de première classe ¿¿ dans le modèle formel, quitte à effectuer des traitements particuliers pour des raisons d'efficacité.
- En tant qu'objet du modèle, cette relation pourra être modifiée, affinée suivant les besoins de l'utilisateur, simplement en modifiant les règles de graphes qui s'appliquent sur cette relation. C'est ce besoin que j'avais déjà présenté au chapitre 1.

2.3.1 La co-référence dans les graphes simples

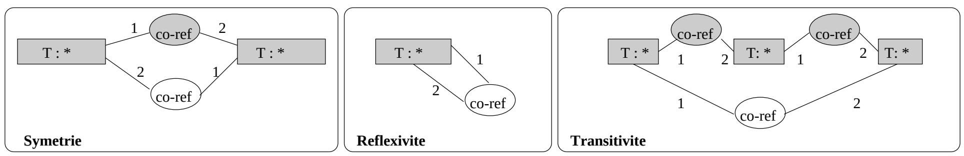
Pour simuler la co-référence avec des règles, j'ai besoin d'un type de relation binaire particulier, *co-ref*. La co-référence étant définie pour tous les types de concepts, sa signature sera $(\top_C, \top_C)$. Quelle sera la place de ce type de relation dans l'ordre défini pour l'ensemble des types de concepts ?

- Soit cette relation est une relation ¡¡ normale ¿¿, c'est à dire inférieure à l'élément maximal $\top_R^2$.
- Soit cette relation a un statut particulier, et est incomparable à tous les autres types de relations, y compris à $\top_R^2$.

Le choix d'une solution est avant tout philosophique : soit on considère que la co-référence est une relation du graphe comme toutes les autres, et on choisit la première solution, soit on considère que la co-référence est un jugement sur les sommets concepts, qui se situe à un niveau d'abstraction superieur, mais que l'on souhaite tout de même représenter dans le même modèle formel. Dans ce cas, il faudra effectuer une modification du support similaire à celle de l'application Θ (voir chapitre 1) : rajouter un nouveau majorant pour les types de relations binaires, $\top\top_R^2$, qui couvre $\top_R^2$ et la relation *co-ref*.

La relation *co-ref* est une relation d'équivalence, c'est à dire une relation réflexive, symétrique et transitive, ce qui peut s'exprimer par les trois règles de la figure 2.10.

FIG. 2.10 – Reflexivité, symétrie et transitivité de la relation d'identité exprimée sous forme de règles de graphes conceptuels.



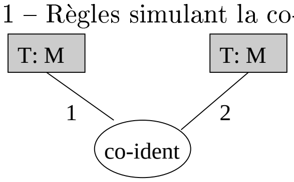
Dans un premier temps (puisque je suis en train de simuler le lien de co-référence), j'imposerai à la relation *co-ref* de ne mettre en relation que des sommets concepts génériques ayant même type. Le lecteur pourra s'assurer que, si un graphe G ne contient que des relations *co-ref* reliant des sommets concepts génériques de même type, alors les trois règles ne peuvent rajouter la relation *co-ref* qu'entre des sommets concepts respectant la même contrainte.

Ces trois règles assurent que des sommets co-référents (c'est à dire reliés par la relation binaire *co-ref*) ne puissent se projeter que sur des sommets co-référents (par projection du sommet relation de type *co-ref* dans un sommet relation de type *co-ref*), ou sur un même sommet (qui est *co-ref* à lui-même par application de la règle de réflexivité).

Et pour que ces ¡¡ règles de co-référence ¿¿ fonctionnent de la même manière que le lien de co-référence, il faut rajouter une relation et un ensemble de règles qui simulent la co-identité :

- Le type de relation binaire *co-ident*, dont la signature est $(\top_C, \top_C)$, est couvert par le type de relation *co-ref*.
- Pour chaque marqueur individuel M apparaissant dans le graphe, il y a une règle qui rajoute la relation *co-ident* entre deux sommets concepts ayant M comme marqueur (figure 2.11).

FIG. 2.11 – Règles simulant la co-identité



Ainsi, si deux sommets ont même marqueur (et d'après les contraintes qu'on a sur les graphes conceptuels, ils ont alors nécessairement même type), on en déduit un sommet relation de type *co-ident* entre ces deux sommets. Deux sommets co-référents d'un graphe H pourront alors se projeter dans deux sommets co-identiques d'un graphe G, *co-ref* étant un type de relation plus général que *co-ident*.

Ainsi, ces règles et ces relations simulent parfaitement le fonctionnement du lien de co-référence, deux sommets co-référents ne pouvant se projeter que dans des sommets co-identiques. En d'autres termes, et de façon plus formelle :

Proposition 2 (Equivalence entre liens et règles de co-référence) *Soient G et H deux graphes conceptuels simples avec liens de co-référence, définis sur un support $\mathcal{S}$. Soient G' et H' les graphes définis sur le support S auquel on a rajouté* [2] *les types de relations* co-ref *et* co-ident.

G' et H' sont obtenus à partir de G et H en remplaçant tous les liens de co-référence représentés sur le dessin du graphe par une relation co-ref *entre les sommets concepts concernés du graphe.*

Soit $\mathcal{R}$ un ensemble de règles comportant les trois règles de co-référence et une règle de co-identité par marqueur individuel distinct apparaissant dans G.

Alors il existe une projection de H dans G si et seulement si il existe une $\mathcal{R}$-projection de H' dans G'.

La co-référence étant modélisée par un ensemble de règles de graphes conceptuels, on peut intégrer le lien de co-référence dans les règles si on s'assure que le graphe obtenu par l'application d'une telle règle conserve encore les contraintes d'identité de type et de généricité du marqueur pour les sommets concepts reliés par une relation de co-référence. Ceci est garanti par les contraintes suivantes :

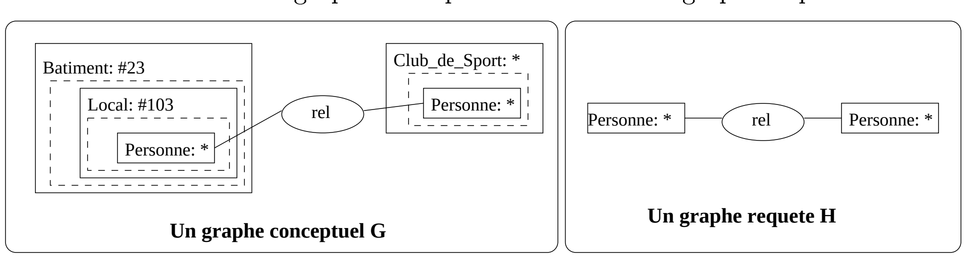
Soit R une règle de graphes conceptuels telle qu'un sommet relation de type *co-ref* apparait dans la conclusion de la règle. Alors, soit il met en relation deux sommets concepts génériques de même type apparaissant eux aussi dans la conclusion de la règle, soit il met en relation deux sommets concepts co-référents de la précondition.

Je proposerai par la suite une méthode permettant de relâcher cette contrainte, en rétablissant la cohérence du graphe après chaque application d'une règle.

2.3.2 Extension aux graphes de boîtes

Je viens de montrer comment le lien de co-référence peut être simulé par des règles dans le cas des graphes simples. Ceci ne peut en aucun cas s'appliquer aux graphes emboîtés, car ce modèle ne permet pas d'établir des relations entre des sommets concepts qui sont dans des contextes différents. Je vais donc étendre la relation de co-référence aux graphes de boîtes définis dans le chapitre 1 (c'est d'ailleurs bien dans cette perspective qu'ils avaient été définis).

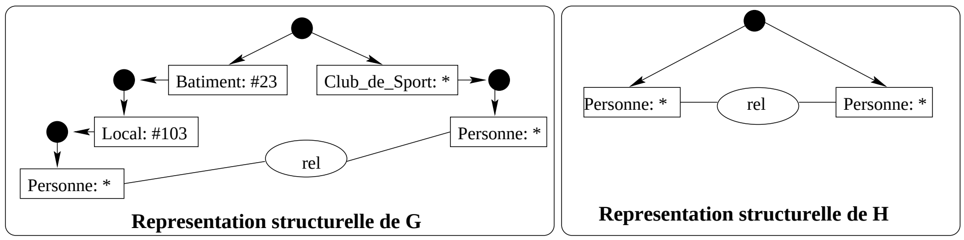
FIG. 2.12 – Un graphe conceptuel emboîté et un graphe requête



[2]Soit comme des relations incomparables à toutes les autres, soit comme des relations ¡¡ normales ¿¿ du support

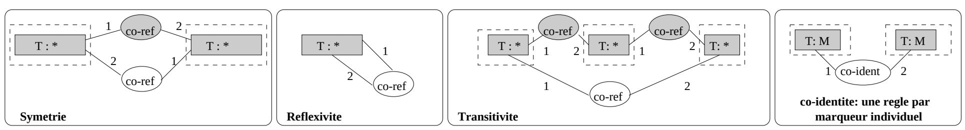
Le problème est qu'aucune des règles définies pour les graphes simples ne s'applique ! En effet, si deux sommets concepts reliés par une relation de co-référence sont dans des contextes différents, aucune projection des règles définies pour les graphes simples ne peut avoir lieu. Le graphe H de la figure 2.12 ne se projette pas dans le graphe G de cette même figure.

FIG. 2.13 – Représentation structurelle : il n'y a pas de projection de H dans G



Ceci se voit de façon plus claire en donnant la représentation structurelle de ces mêmes graphes (voir figure 2.13).

FIG. 2.14 – Règles de co-référence et de co-identité pour les graphes de boîtes : utilisation de graphes de boîtes non standards



C'est pour cette raison que j'avais défini les graphes de boîtes comme un *ensemble de boîtes*. On peut ne pas avoir un contexte racine unique, c'est ce que j'avais appelé un graphe de boîtes non standard. La projection d'un graphe de boîtes H dans un graphe de boîtes G se faisant par la projection de chaque contexte racine de H dans un contexte racine quelconque de G, on peut redéfinir les règles de co-référence et de co-identité représentées dans la figure 2.14, en utilisant des règles dont la précondition est un graphe de boîtes non standard.

En utilisant ces règles, on simule de la même manière que dans le cas des graphes simples le fonctionnement de la règle de co-référence.

Proposition 3 (Equivalence entre liens et règles de co-référence) *Soient G et H deux graphes de boîtes avec liens de co-référence, définis sur un support $\mathcal{S}$. Soient G' et H' les graphes définis sur le support S auquel on a rajouté les types de relations* co-ref *et* co-ident*.*

G' et H' sont obtenus à partir de G et H en remplaçant tous les liens de co-référence représentés sur le dessin du graphe par une relation co-ref *entre les sommets concepts concernés du graphe.*

Soit $\mathcal{R}$ un ensemble de règles comportant les trois règles de co-référence et une règle de co-identité par marqueur individuel distinct apparaissant dans G.

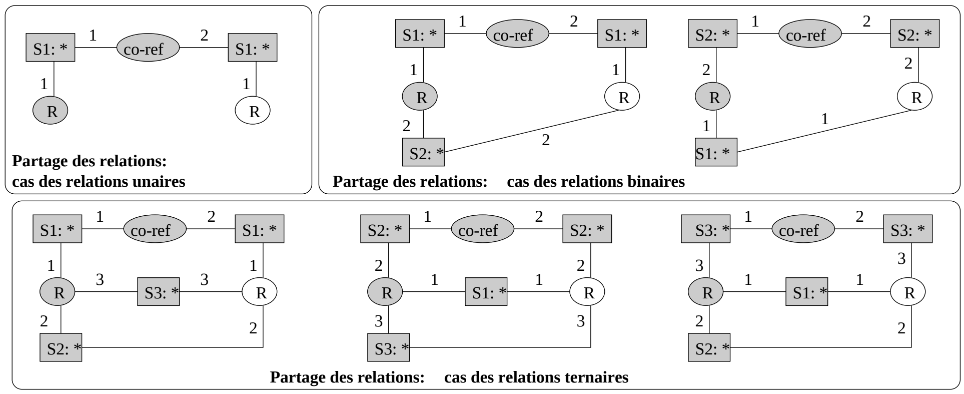
Alors il existe une projection de H dans G si et seulement si il existe une $\mathcal{R}$-projection de H' dans G'.

Les remarques portant sur l'utilisation de la relation de co-référence dans les règles, qui avaient été faites dans le cas des graphes simples, restent toujours valides.

Il faut aussi voir que, pour la sémantique Φ, la condition de normalité impose à tous les sommets concepts individuels du graphe de boîtes d'avoir des marqueurs distincts. La classe d'équivalence des sommets co-identiques est donc réduite à la classe triviale (chaque sommet n'est co-identique qu'à lui-même), ce qui réduit grandement l'intérêt de la co-référence.

2.3.3 Extension de la relation de co-référence

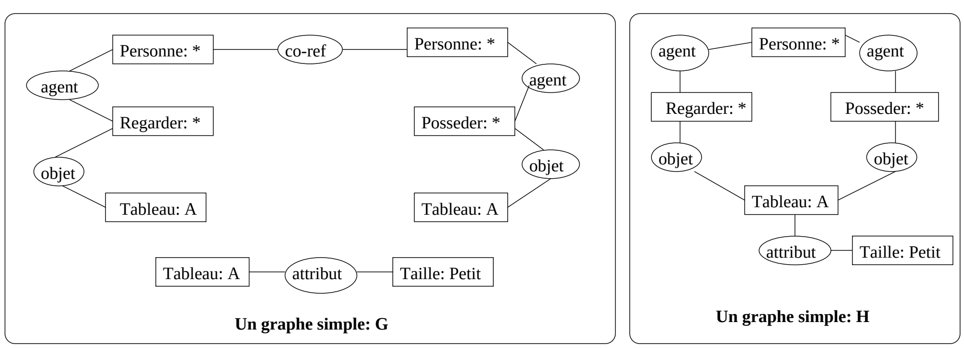
FIG. 2.15 – Règles simulant le partage des relations



Je souhaite représenter par des règles de graphes les raisonnements, déjà présentés, que l'on peut souhaiter faire à partir de l'identité entre instances. Nous avons vu que le partage de description comme le partage des relations se faisait – parfois – d'office avec la sémantique Φ. Pour avoir une gestion plus fine et personnalisable de ce genre de raisonnement, je n'utiliserai plus désormais que la sémantique Ψ, ce qui nous permettra de s'affranchir de la condition de normalité, trop contraignante dans le cas des graphes de boîtes.

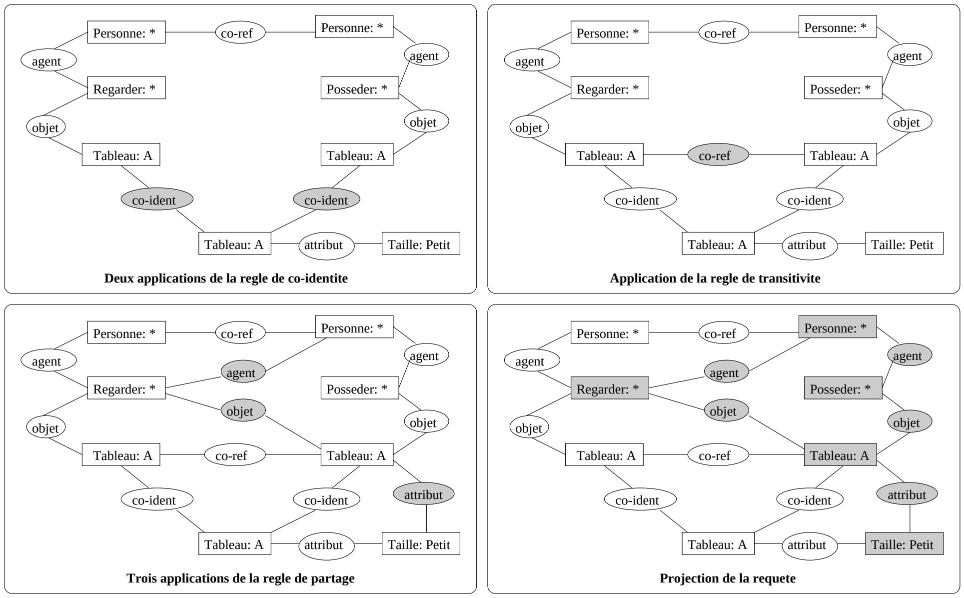
Partage des relations : La figure 2.15 montre comment un ensemble de règles utilisant le type de relation *co-ref* donne la possibilité de déduire les mêmes informations sur deux sommets co-référents distincts. Ces règles s'appliquent dans le cas des graphes simples, et les sommets concepts reliés par une arête d'étiquette i à la relation R ont pour type $\sigma_i(R)$. Nous aurons besoin d'une telle règle pour chaque relation unaire et, de façon plus générale, de n règles pour chaque relation n-aire.

FIG. 2.16 – Deux graphes sonceptuels simples, utilisant la relation de co-référence



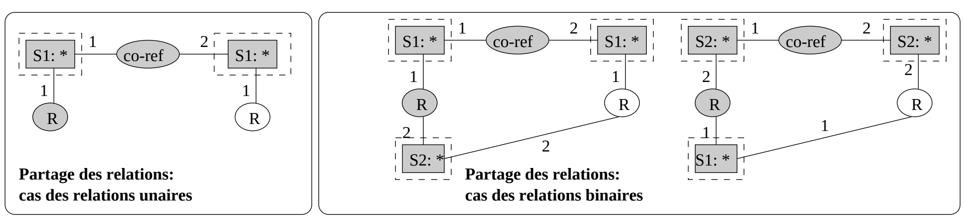
A titre d'exemple, soit la figure 2.16, qui montre deux graphes conceptuels simples, G et H. Je vais montrer que, si $\mathcal{R}$ est l'ensemble des règles pour la co-référence que je viens de définir, il existe une $\mathcal{R}$-projection de H dans G.

FIG. 2.17 – $\mathcal{R}$-projection de H dans G : une séquence possible d'application de règles



La figure 2.17 montre une séquence possible d'application des règles permettant la projection de H. Notons au passage qu'à la deuxième étape, il apparait une relation de co-référence entre deux sommets concepts qui ne sont pas génériques. Ceci n'est cependant pas un problème, car on peut montrer que, avec les contraintes admises sur la relation de co-référence, si l'application d'une règle crée une relation de co-référence entre deux sommets concepts, alors soit ce sont deux sommets concepts génériques de même type, soit ce sont deux sommets concepts individuels ayant même marqueur. On crée ainsi, dans le graphe, des classes d'équivalence distinctes pour les sommets génériques et individuels.

FIG. 2.18 – Règles simulant le partage des relations : version graphes de boîtes

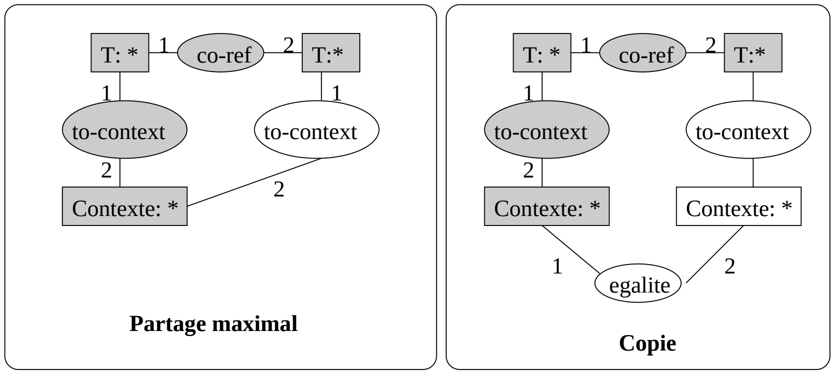


Ces règles s'étendent naturellement aux graphes de boîtes en utilisant des règles de boîtes non standard, comme le montre la figure 2.18.

Partage du contexte : On peut maintenant écrire de façon similaire des règles assurant le partage du contexte. Je présenterai ces règles comme des règles s'appliquant au graphe simple associé, i.e. le graphe équivalent obtenu par l'application de Θ.

La version ¡¡ partage maximal ¿¿ de la figure 2.19, qui reprend la même idée que pour le partage de relations, est la plus efficace car elle ne nécessite pas de récursion. Son inconvénient est qu'elle ne crée pas un graphe de boîtes. En effet, le contexte

FIG. 2.19 – Deux façons de simuler le partage de description – version graphe simple associé



partagé devient la description de deux concepts distincts (en termes de sommets) et on n'a plus une arborescence de contextes.

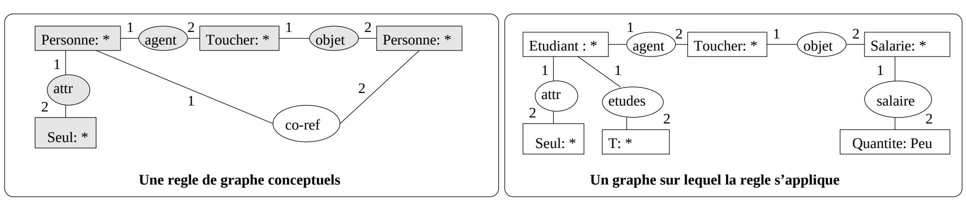
La version ¡¡ copie ¿¿ doit donc être choisie, bien qu'elle soit moins efficace d'un point de vue algorithmique. Voir qu'elle n'entraine aucune violation sur les contraintes que nous avions imposées à la co-référence : on crée un lien de co-référence entre deux sommets concepts de même type (contexte) et ayant un marqueur générique.

Si on souhaite étendre ces règles aux graphes de boîtes typées, il nous faudra une telle règle par type de contexte.

2.4 Co-référence et conjonction de types

J'ai jusqu'à présent toujours appliqué les règles de façon à respecter les contraintes établies pour la co-référence : il n'y a dans le graphe des relations de co-référence qu'entre sommets de même type, ayant un marqueur générique. Ainsi, ne peuvent être co-référents, même après l'application des règles que j'ai définies (par recherche de la classe d'équivalence) que des sommets de même type, et les règles de partage des relations n'entrainent pas de violation de la signature des relations.

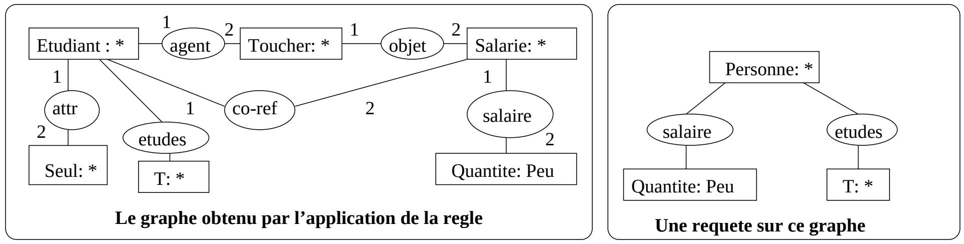
FIG. 2.20 – Une règle de graphe simple : ¡¡ Si une personne seule est en contact physique avec une autre, alors ce sont les mêmes ¿¿ et un graphe simple sur lequel la règle s'applique.



Pourtant, pouvoir rajouter la relation de co-référence dans la conclusion d'une règle mène facilement à la violation de cette contrainte, comme le montre la figure 2.20.

Alors que la règle ne semble pas violer la contrainte d'identité de types, puisque la relation de co-référence est entre deux sommets de même type, son application au graphe crée une relation de co-référence entre des sommets de types différents.

FIG. 2.21 – Une requête sur le graphe obtenu par l'application de la règle précédente



La figure 2.21 montre le graphe obtenu par l'application de la règle au graphe, et une pseudo-requête [3] sur celui-ci. On peut s'attendre à ce que la règle de partage des relations s'applique, et qu'on puisse ainsi obtenir une projection valide. Or cette règle ne s'applique pas, car comme chacun le sait, *Etudiant* ne respecte pas la signature de *salaire* et *Salarié* ne respecte pas la signature de *études*.

2.4.1 La requête vue comme un graphe faiblement contraint

Le premier problème est que la requête de la figure 2.21 n'est pas un graphe conceptuel valide, car elle ne respecte pas la signature des relations. Pourtant, il me semble important de pouvoir déduire ¡¡ Il y a une personne qui reçoit un salaire ¿¿ du fait ¡¡ Un salarié reçoit un salaire ¿¿, même si le type de concept ¡¡ Personne ¿¿ est trop général pour la signature de la relation. On pourrait pourtant déduire cette information (la projection se fait sans aucun problème), mais *on n'a pas le droit d'écrire ce graphe.*

La seconde difficulté est que les règles assurant le partage des relations ne s'appliquent pas. Par exemple, la règle qui rajouterait la relation *salaire* entre *Quantité* et *Etudiant* ne peut pas s'appliquer, car le type du sommet de la précondition que l'on cherche à projeter sur *Etudiant* est *Salarié*, et ces deux types de concepts sont incomparables.

Définition 18 (Graphes à signature faible) *Un graphe simple (resp. emboîté, de boîtes) à signature faible, défini sur un support $\mathcal{S}$, est défini de la même façon qu'un graphe conceptuel simple (resp. emboîté, de boîtes), à la seule différence que la contrainte sur la signature peut ne pas être respectée. C'est-à-dire qu'il peut exister un sommet concept C et un sommet relation R, reliés par une arête d'étiquette i, tels que $\neg(type(C) \leq_C \sigma_i(type(R)))$.*

Un graphe de faits sera donc un graphe conceptuel au sens habituel du terme. En effet, notre base de connaissances se doit de respecter le support. Un graphe requête sera un graphe à signature faible.

La projection d'un graphe requête dans un graphe de faits est définie de la même manière que la projection ¡¡ habituelle ¿¿. Les résultats d'adéquation et de complétude de la projection par rapport aux différentes sémantiques étudiées sont conservés, car on ne fait que projeter des graphes qui se projetaient déjà, dont la sémantique logique pouvait déjà se déduire, mais que l'on n'avait pas le droit d'écrire.

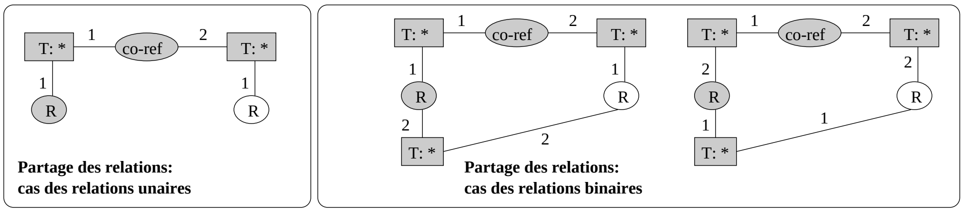
Ceci pourra s'adapter aux règles de graphes conceptuels :

Définition 19 (Règles faiblement contraintes) *Une règle de graphe conceptuels (simple, emboîté ou de boîtes) faiblement contrainte est un graphe faiblement contraint bicolore dont la précondition est un graphe faiblement contraint.*

[3] Ce n'est pas un graphe conceptuel valide, la signature des relations n'étant pas respectée.

De telles règles nous permettraient d'écrire des règles de partage des relations qui nous donneraient le résultat voulu, comme le montre la figure 2.22.

FIG. 2.22 – Une version assurant le partage des relations quel que soit le type de concept



Le problème est de s'assurer que l'application d'une telle règle au graphe de faits (qui est un graphe conceptuel au sens ¡¡ classique ¿¿ du terme) retourne bien un graphe de faits.

2.4.2 Utilisation d'une liste de types de concepts

En fait, pour pouvoir résoudre ce problème, il faut se donner la possibilité de créer un type *Etudiant, Salarié*, c'est à dire de manipuler un ¡¡ type construit ¿¿, qui est la conjonction de plusieurs types.

Définition 20 (Graphe conceptuel multi-concepts) *Un graphe conceptuel multi-concepts G, défini sur un support $\mathcal{S}$, est un graphe conceptuel modifié de la façon suivante :*

(i) Les sommets concepts sont étiquetés par un ensemble non vide de types de concepts, qui est le type du sommet, et un marqueur générique ou individuel.

(ii) Si R est un sommet relation, alors pour chaque sommet concept C_i de R (où C_i est le i^evoisin de R), $\exists t \in type(C_i)\ t \leq_R \sigma_i(R)$.

Ceci définit les graphes de faits que je manipulerai. Les graphes requêtes sont des graphes définis de façon similaire, mais qui ne respectent pas la contrainte *(ii)*.

Il reste à définir la projection dans les graphes multi-concepts, c'est à dire redéfinir la relation d'ordre pour les types de concepts construits.

Définition 21 (Relation d'ordre pour les concepts construits) *Soient L_1 et L_2 deux types de concepts construits, i.e. deux listes de types de concepts primitifs. La relation d'ordre $\leq_{CC}$ sur les types de concepts construits est définie à partir de la relation d'ordre $\leq_C$ des types de concepts primitifs, de la façon suivante :*

$$L_1 \leq_{CC} L_2 \Leftrightarrow \forall t_2 \in L_2\ \exists t_1 \in L_1 \quad t_1 \leq_C t_2$$

Cette définition traduit uniquement le fait qu'un type de concept construit $L = t_1, \ldots, t_k$ s'interprète comme la conjonction [4] des types primitifs $t_1 \sqcap \ldots \sqcap t_k$.

On peut ainsi redéfinir la projection de graphes multi-concepts, en remplaçant la contrainte $\text{type}(\Pi(C)) \leq_C \text{type}(C)$ par la contrainte $\text{type}(\Pi(C)) \leq_{CC} \text{type}(C)$.

On peut étendre les sémantiques logiques Φ et Ψ aux graphes multi-concepts en interprétant un type construit comme la conjonction des types, de la façon suivante :

- L'interprétation du support est inchangée.
- L'interprétation des sommets relations est inchangée.

[4] Notons au passage que les termes utilisés (concepts primitifs, concepts construits) comme les symboles ($\sqcap$) ne sont pas sans rappeler ceux utilisés dans les logiques de description... Ceci n'est pas un hasard...

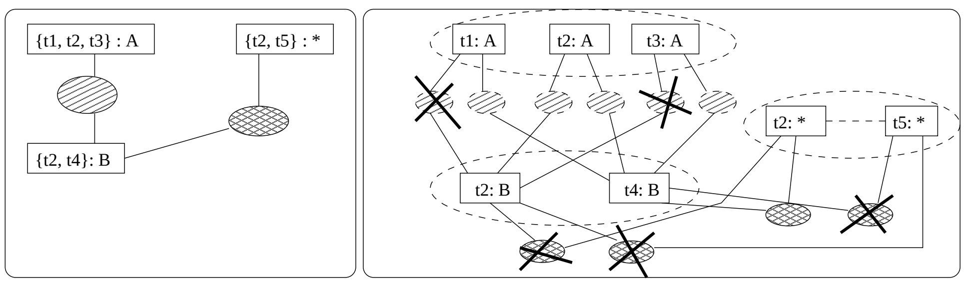
- Si C est un sommet concept de type $T = \{t_1, \ldots, t_k\}$, alors l'interprétation de ce sommet est :
  - $t_1(m) \wedge \ldots \wedge t_k(m)$ pour la sémantique Φ associée aux graphes simples.
  - $t_1(c,m) \wedge \ldots \wedge t_k(c,m)$ pour la sémantique Φ associée aux graphes emboîtés.
  - $t_1(x,m) \wedge \ldots \wedge t_k(x,m)$ pour la sémantique Ψ associée aux graphes simples.
  - $t_1(c,x,m) \wedge \ldots \wedge t_k(c,x,m)$ pour la sémantique Ψ associée aux graphes emboîtés.

  où m est la variable ou la constante associée au marqueur, c la variable associée au contexte, et x la variable associée au sommet.

Les résultats d'adéquation et de complétude de la projection par rapport aux sémantiques logiques Φ et Ψ se retrouvent pour les graphes multi-concepts.

Théorème 10 (Adéquation et complétude) *Soient G et H deux graphes multi-concepts (simples, emboîtés ou graphes de boîtes), définis sur un support $\mathcal{S}$. Alors, si G est sous la forme normale imposée pour ce type de graphe avec liens de co-référence, et pour la sémantique logique concernée, H se projette dans G si et seulement si l'interprétation de H peut se déduire de celle de G et de celle de $\mathcal{S}$.*

FIG. 2.23 – Transformation d'un graphe multi-concepts en un graphe avec liens de co-référence



En effet, on peut transformer un graphe multi-concepts en un graphe conceptuel ¡¡ standard ¿¿ avec lien de co-référence en remplaçant chaque sommet multi-concepts par une classe co-identique de sommets dont le type est primitif (voir figure 2.23), et en remplaçant chaque relation entre sommets ayant un type construit par une relation portant sur tous les sommets de la classe d'équivalence associée. Ensuite, il n'y a plus qu'à enlever les sommets relations tels que leurs voisins ne respectent pas la signature (il y en a toujours au moins un qui reste entre chaque classe de co-identité qui était en relation dans le graphe multi-concepts).

On voit que cette transformation Γ conserve la même interprétation logique, et qu'il y a une projection de H dans G si et seulement si il y a une projection de $\Gamma(H)$ dans $\Gamma(G)$.

Ce modèle de graphes conceptuels se servant de types construits étant bien défini, il ne reste plus qu'à voir comment s'en servir pour maintenir la cohérence du graphe de faits par rapport à la signature pendant l'application des règles.

L'application d'une règle doit donc être suivie d'une modification du type des sommets qui violent la contrainte de la signature.

Si R est un sommet relation rajouté par l'application d'une règle, et C un sommet concept i-voisin de R de type $T = \{t_1, \ldots, t_k\}$ tel que C viole la contrainte de la signature de R, alors il faut modifier le type de C de la façon suivante :

$$T \leftarrow T \cup \sigma_i(\text{type}(R))$$

Cette modification du moteur d'application des règles ne change en fait pas la sémantique logique associée à nos graphes. Il suffit de rajouter l'interprétation de la signature, qui était jusqu'à présent inutile, pour pouvoir interpréter cette opération.

Soit r un type de relation tel que $\sigma_i(r) = t$. L'interprétation de la signature se fait de la façon suivante :

– Sémantique Φ associée aux graphes simples :

$$\forall x_1 \dots \forall x_p (r(x_1, \dots, x_i, \dots, x_p) \rightarrow t(x_i))$$

– Sémantique Ψ associée aux graphes simples :

$$\forall x_1 \dots \forall x_p \forall m ((r(x_1, \dots, x_i, \dots, x_p) \wedge \top(x_i, m)) \rightarrow t(x_i, m))$$

– Sémantique Φ associée aux graphes emboîtés :

$$\forall x_1 \dots \forall x_p \forall c (r(c, x_1, \dots, x_i, \dots, x_p) \rightarrow t(c, x_i))$$

– Sémantique Ψ associée aux graphes emboîtés :

$$\forall x_1 \dots \forall x_p \forall m \forall c ((r(c, x_1, \dots, x_i, \dots, x_p) \wedge \top(c, x_i, m)) \rightarrow t(c, x_i, m))$$

J'ai proposé dans ce chapitre un ensemble de règles de graphes qui simulent la co-référence pour les graphes conceptuels simples et les graphes de boîtes. D'autres règles peuvent être données afin de pouvoir inférer des informations sur les individus à partir de cette co-référence. L'intérêt d'une telle démarche est que, la relation de co-référence étant un objet du modèle, on peut modifier ces règles de façon à obtenir une relation d'égalité ¡¡ à la carte ¿¿, i.e. dont le fonctionnement peut s'adapter à une application particulière.

A condition de modifier le moteur d'application de règles au graphe, ces règles peuvent être étendues de façon à pouvoir inférer l'identité de types de l'identité d'individus. La contrainte d'identité de type pour la co-référence n'est donc plus nécessaire. Ceci est encore une généralisation non stricte du modèle : on s'est juste donné le droit d'écrire des graphes qui pouvaient déjà se projeter dans les graphes conceptuels, et dont on pouvait déjà déduire l'interprétation logique.

Conclusion

J'ai proposé dans ce mémoire certaines extensions au modèle des graphes conceptuels :

- Graphes de boîtes : cette généralisation non stricte du modèle des graphes conceptuels emboîtés permet de mettre en relation des sommets concepts qui sont dans des contextes différents, et, avec une vision ¡¡ bases de données ¿¿, d'écrire des requêtes sur un graphe sans avoir une connaissance à priori de la structure d'emboîtement de ce graphe, même si de telles requêtes sont algorithmiquement moins efficaces. Je pense qu'on garde ainsi les avantages de représentation et de structuration qu'apportaient les graphes emboîtés, sans être pénalisés par cet aspect ¡¡ boîte noire ¿¿ qu'avaient les contextes.
- Relation de co-référence : j'ai montré que le lien de co-référence tel qu'il avait été défini pouvait être exprimé par des relations du graphe, associées à des règles. Ceci a l'avantage de recentrer cette extension sur un modèle particulier, celui des graphes conceptuels simples munis de règles. Comme la co-référence est désormais un ¡¡ objet de première classe ¿¿ de ce modèle (c'est un sommet du graphe), je montre que l'on peut désormais raisonner sur cet objet.
- Conjonction de types : je propose un modèle de graphes conceptuels dont les sommets concepts sont étiquetés par une conjonction de types. Quitte à redéfinir la méthode d'application des règles à un graphe, je montre que ces ¡¡ types construits ¿¿ permettent de spécialiser dynamiquement les types de concepts au cours d'une séquence d'application de règles, et ceci sans modifier la sémantique logique des graphes conceptuels.

L'ensemble de ces propositions forme une généralisation non stricte du modèle des graphes conceptuels simples avec les règles étudiées dans [Sal97]. L'étude algorithmique de ce modèle doit donc être centrée sur deux points précis :

- La projection, qui est dans le cas général un problème NP-complet.
- La recherche d'une $\mathcal{R}$-projection, qui semble être un problème semi-décidable.

Certaines pistes de recherche que j'ai explorées n'ont pas trouvé leur place dans ce rapport :

- $\mathcal{R}$-projection : conjecturant que le problème de la R-projection est décidable, j'ai – longuement – travaillé à essayer de borner la longueur d'une séquence d'applications de règles suffisant à obtenir la projection d'un graphe et, à tout le moins, à caractériser des sous-ensembles de règles pour lesquelles ce problème est décidable. N'étant parvenu, pour l'instant, à aucun résultat probant, le travail que j'ai fait pour formaliser et utiliser le graphe représentant les séquences d'applications de règles, n'a pas été intégré dans ce rapport. Je pense toutefois que ce travail mérite une suite...
- Définitions de types : il parait naturel de pouvoir nommer un type construit, c'est à dire de définir un type de concept par une conjonction de types, telle qu'elle a été formalisée au chapitre 2. Cette notion peut s'étendre facilement à la définition d'un type par un graphe conceptuel. L'équivalence que traduit la définition de type peut être modélisée par une règle de graphe ayant comme précondition le type défini, à qui un certain sous-ensemble de sommets

concepts de la conclusion sont co-référents. Cette règle devra pouvoir s'expanser complètement dans un graphe requête (i.e. il n'y aura qu'une séquence finie d'applications possibles de la règle au graphe suivant des projections différentes), ce qui sera assuré par l'absence de cycles et de boucles dans le graphe de dépendance des types de concepts définis.
Les logiques de description [DLNN95] [Duc98] peuvent être une bonne source d'inspiration pour étendre la définition de types de concepts à la disjonction ou à la négation de types. C'est ce travail qui resterait à faire.

– Règles de graphes non monotones : Des applications utilisant un raisonnement à base de projection telles que la modélisation de comportements [BBV97] [Bos98] ou la planification en synthèse organique [Tog98] utilisent une forme de raisonnement non monotone. Il serait manifestement utile d'étendre les règles de graphes conceptuels afin de permettre un tel raisonnement.
On peut étendre les graphes bicolores que sont nos règles de graphes à des graphes tricolores, une couleur représentant la partie stable de la projection, une deuxième ce que l'on doit pouvoir projeter sur des sommets qui seront supprimés, une troisième symbolisant les sommets à rajouter. Le graphe représente alors un ensemble de faits qui évolue par l'application d'une telle règle. L'existence d'une projection dans le graphe obtenu par une séquence d'applications de règles n'est plus la seule information importante. On peut se demander si cette information est stable, certaine, où même vouloir calculer sa probabilité en utilisant des règles probabilisées.
Une étude algorithmique de ces règles ne pourra donc pas se passer d'une réflexion approfondie sur la sémantique de celles-ci.

[]

Bibliographie

[BBV97] Corinne Bos, Bernard Botella, and Philippe Vanheegue. Modeling and simulating human behaviors with conceptual graphs. In *Conceptual Structures : Fulfilling Peirce's Dream, ICCS'97 Proc.*, LNAI 1257, pages 275–289. Springer Verlag, 1997.

[Bos98] Corinne Bos. *Modéliser pour Simuler pour Modéliser : Contribution à l'Acquisition de Connaissances Comportementales.* PhD thesis, Université des Sciences et Technologies de Lille, 1998.

[CM92] Michel Chein and Marie-Laure Mugnier. Conceptual graphs : fundamental notions. *Revue d'Intelligence Artificielle*, 6(4), 1992.

[CM95] Michel Chein and Marie-Laure Mugnier. Conceptual graphs are also graphs. Technical report, LIRMM, 1995.

[CM97] Michel Chein and Marie-Laure Mugnier. Positive nested conceptual graphs. In *Conceptual Structures : Fulfilling Peirce's Dream, ICCS'97 Proc.*, LNAI 1257, pages 95–109. Springer Verlag, 1997.

[CMS98] Michel Chein, Marie-Laure Mugnier, and Geneviève Simonet. Nested graphs : A graph-based knowledge representation model with fol semantics. To appear in KR'98, 1998.

[CS98] Stéphane Coulondre and Eric Salvat. Piece resolution : Towards larger perspectives. To appear in ICCS'98, 1998.

[CsBc95] John Mc Carthy and Sa sa Buvač. Formalizing context (expanded notes). http ://www-formal.stanford.edu/jmc/index.html, 1995.

[DLNN95] Francesco M. Donini, Morizio Lenzerini, Daniele Nardi, and Werner Nutt. The complexity of concept languages. Technical report, Deutsches Forschungszentrum für Künstliche Intelligenz, Saarbrücken, 1995.

[Duc98] Roland Ducournau. Des langages à objets aux logiques terminologiques : les systèmes classificatoires. Technical report, LIRMM, Montpellier, 1998.

[Gen96] David Genest. Une utilisation des graphes conceptuels pour la recherche documentaire. Master's thesis, Université de Montpellier II, 1996.

[GS98] David Genest and Eric Salvat. A platform allowing typed nested graphs : How cogito became cogitant. To appear in ICCS'98, 1998.

[Hae95] Olivier Haemmerlé. *CoGITo : une plate-forme de développement de logiciels sur les graphes conceptuels.* PhD thesis, Université de Montpellier II, 1995.

[Hen79] Gary C. Hendrix. Encoding knowledge in partitioned networks. In N. Finder, editor, *Associative Networks.* Academic Press, 1979.

[MC96] Marie-Laure Mugnier and Michel Chein. Représenter des connaissances et raisonner avec des graphes. *Revue d'Intelligence Artificielle*, 10(1), 1996.

[Mug95] Marie-Laure Mugnier. On specialization / generalization for conceptual graphs. *Journal of Experimental and Theoretical Artificial Intelligence*, 7(3), 1995.

[PMC95] Anne Preller, Marie-Laure Mugnier, and Michel Chein. A logic for nested graphs. Research Report 95-038, LIRMM, 1995. To appear in Computational Intelligence.

[Sal93] Eric Salvat. Algorithmes de projection pour les graphes conceptuels. Master's thesis, Université de Montpellier II, 1993.

[Sal97] Eric Salvat. *Raisonner avec des opérations de graphes : graphes conceptuels et règles d'inférence.* PhD thesis, Université de Montpellier II, 1997.

[Sal98] Eric Salvat. Theorem proving using graphs operations in the conceptual graph formalism. To appear in ECAI'98, 1998.

[Sim96a] Geneviève Simonet. Une autre sémantique logique pour les graphes emboités. Research Report 96-048, LIRMM, 1996.

[Sim96b] Geneviève Simonet. Une sémantique logique pour les graphes emboités. Research Report 96-047, LIRMM, 1996.

[SM96] Eric Salvat and Marie-Laure Mugnier. Sound and complete forward and backward chaining of graphs rules. In *Conceptual Structures : Knowledge Representation as Interlingua, ICCS'96 Proc.*, pages 248–262. Springer Verlag, 1996.

[Sow84] John F. Sowa. *Conceptual Structures - Information Processing in Mind and Machine.* Addisson-Wesley, 1984.

[Sow97] John F. Sowa. Peircean foundations for a theory of context. In *Conceptual Structures : Fulfilling Peirce's Dream, ICCS'97 Proc.*, LNAI 1257. Springer Verlag, 1997.

[Tog98] Yannic Tognetti. Classification approchée dans une hiérarchie de graphes : Application à la planification en synthèse organique. Master's thesis, Université de Montpellier II, 1998.

[Woo91] W. A. Woods. Understanding subsumption and taxonomy : A framework for progress. In John F. Sowa, editor, *Principles of Semantic Networks : Explorations in the Representation of Knowledge.* Morgan Kaufmann, 1991.