

Aligner des ontologies lourdes : une méthode basée sur les axiomes

Frédéric Fürst, Francky Trichet

► **To cite this version:**

Frédéric Fürst, Francky Trichet. Aligner des ontologies lourdes : une méthode basée sur les axiomes. IC - 16èmes Journées francophones d'Ingénierie des Connaissances, May 2005, Nice, France. Presses universitaires de Grenoble, pp.121-132, 2005. <hal-01023903>

HAL Id: hal-01023903

<https://hal.inria.fr/hal-01023903>

Submitted on 15 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Aligner des ontologies lourdes : une méthode basée sur les axiomes

Frédéric Fürst, Francky Trichet

Laboratoire d'Informatique de Nantes Atlantique (CNRS-FRE 2729)
2 rue de la Houssinière - BP 92208 - 44322 Nantes CEDEX 03
{furst, trichet}@univ-nantes.fr

Résumé : L'alignement d'ontologies de domaine est une question centrale dans de nombreuses applications, en particulier le Web sémantique, nécessitant la prise en compte d'une interopérabilité sémantique entre ontologies existantes et couvrant des domaines connexes. Dans cet article, nous introduisons une nouvelle approche de l'alignement d'ontologies qui est essentiellement basée sur l'utilisation des descriptions intentionnelles des concepts et des relations, descriptions appelées *axiomes*. De ce fait, cette approche s'avère principalement dédiée à l'alignement d'ontologies dites *lourdes*. Cependant, elle peut également être appliquée aux ontologies dites *légères* et est ainsi complémentaire des approches plus courantes fondées sur l'analyse des expressions en langage naturel, des extensions des concepts et/ou des structures taxinomiques. Cette nouvelle approche est définie dans le cadre du modèle des Graphes Conceptuels.

Mots-clés : Alignement d'Ontologies, Graphes Conceptuels

1 Introduction

L'alignement d'ontologies de domaine est au cœur de la problématique de la gestion conjointe de plusieurs ontologies, devenue récemment nécessaire dans de nombreuses applications requérant une interopérabilité sémantique, en particulier le Web sémantique. Les stratégies utilisées pour aligner deux ontologies sont diverses. A titre d'exemple, citons la classification hiérarchique (Doan *et al.*, 2004), l'analyse formelle de concepts - Formal Concept Analysis - (Stumme & Maedche, 2001)), l'analyse terminologique des noms de concepts et/ou des définitions en langage naturel (Noy & Musen, 2003) ou l'analyse structurelle des taxinomies de concepts (Bach *et al.*, 2004). Néanmoins, la plupart des travaux actuels porte exclusivement sur des ontologies qualifiées de légères (« *lightweight ontologies* »), *i.e.* des ontologies composées uniquement de taxinomies de concepts et de relations (Gomez-Perez *et al.*, 2003). Aucun outil existant ne propose de fonctionnalités basées sur d'autres composants, en particulier les axiomes qui sont

pourtant essentiels à la spécification de la sémantique des concepts et relations d'une ontologie (Staab & Maedche, 2000).

Le travail présenté dans cet article vise à définir une nouvelle approche de l'alignement basée sur l'usage de tous les composants des ontologies lourdes, *i.e.* (1) structure hiérarchique des concepts, (2) structure hiérarchique des relations et (3) axiomes. Cette approche suppose que les axiomes soient explicitement représentés au niveau conceptuel, et non au niveau opérationnel comme cela est généralement le cas dans les travaux d'ingénierie ontologique. Par exemple, dans Protégé (Noy & Musen, 2003), les axiomes sont directement représentés sous une forme opérationnelle (*i.e.* une règle ou une contrainte dotée d'une sémantique opérationnelle prédéfinie) à l'aide du langage PAL. Du fait que les axiomes sous forme opérationnelle ne peuvent facilement être comparés, les méthodes d'alignement existantes utilisent uniquement les hiérarchies de concepts (parfois les hiérarchies de relations), et ne peuvent exploiter la richesse sémantique apportée par les axiomes, éléments caractéristiques des ontologies lourdes. Notre travail a pour ambition de contribuer à améliorer les techniques actuelles en proposant une approche complémentaire fondée sur (1) la représentation des axiomes au niveau conceptuel et (2) l'utilisation de ces axiomes pour la découverte d'appariements sémantiques entre concepts et relations.

Pour représenter les ontologies lourdes au niveau conceptuel, nous utilisons OCGL (*Ontology Conceptual Graphs Language*) (Fürst *et al.*, 2004), un langage de modélisation fondé sur le formalisme des Graphes Conceptuels (CGs) (Sowa, 1984), et permettant la représentation d'ontologies lourdes. La représentation des axiomes dans un modèle doté de mécanismes de raisonnement basés sur l'homomorphisme de graphes facilite leur comparaison topologique. Cet aspect est au cœur de la méthode que nous proposons : comparaison structurelle d'ontologies par représentation et raisonnement à l'aide de graphes.

La section 2 présente succinctement le langage OCGL. La section 4 introduit les éléments de base de notre méthode d'alignement. La section 4 expose les principes de l'algorithme qui l'implémente. La section 5 commente les résultats d'une expérimentation menée dans le contexte d'une ontologie des relations familiales. La section 6 compare notre approche avec les travaux existants.

2 Contexte du travail

2.1 Le langage de modélisation OCGL

Représenter une ontologie en OCGL (*Ontology Conceptual Graphs Language*) consiste principalement à (1) spécifier le vocabulaire conceptuel du domaine via des concepts, des relations et des instances ontologiques¹ et (2) spécifier la sémantique de ce vocabulaire à travers des schémas d'axiome et des axiomes de domaine (Fürst *et al.*, 2004).

1. Par exemple, en mathématiques, π est une instance ontologique du concept *Nombre*, car l'expression d'axiomes du domaine requiert cette instance. Mais 3.54, par exemple, n'est pas une instance ontologique.

Les **Schémas d’Axiome** proposés par OCGL sont : (1) les *liens ISA* entre deux concepts ou deux relations (propriété de subsomption), utilisés pour structurer les taxinomies de concepts et de relations (en arbres ou treillis), (2) l’*abstraction* d’un concept (appelée *Exhaustive-Decomposition* dans (Gomez-Perez *et al.*, 2003)), (3) la *disjonction* entre deux concepts, (4) la *signature* d’une relation, (5) les *propriétés algébriques* d’une relation (symétrie, réflexivité, transitivité, irréflexivité, antisymétrie), (6) l’*exclusivité* et l’*incompatibilité* entre deux relations (l’incompatibilité entre R_1 et R_2 est formalisée par $\neg(R_1 \wedge R_2)$, l’exclusivité par $\neg R_1 \Rightarrow R_2$.) et (7) les *cardinalités* maximale et minimale d’une relation.

OCGL a été implémenté dans un outil, appelé TooCoM (*a Tool to Operationalize an Ontology with the Conceptual Graph Model*), dédié à l’édition et l’opérationnalisation d’ontologies de domaine (Fürst *et al.*, 2004) (TooCoM est disponible sous licence GNU GPL à <http://sourceforge.net/projects/toocom/>). La figure 1 présente un extrait de la hiérarchie des relations d’une ontologie des relations familiales.

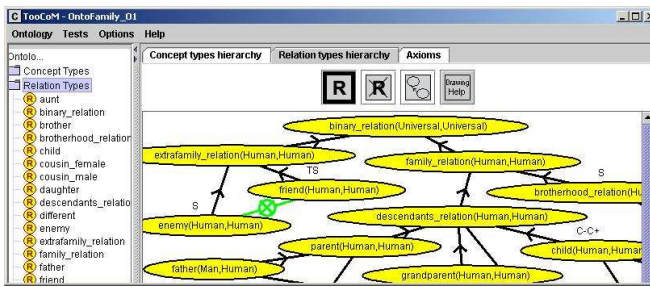


FIG. 1 – Représentation d’une hiérarchie de relations dans TooCoM. Une flèche symbolise un lien de subsomption, un cercle barré une exclusivité, une lettre une propriété algébrique (S pour la symétrie, T pour la transitivité, etc.).

Les **Axiomes de Domaine** diffèrent des schémas d’axiome dans le sens où ils sont totalement spécifiques au domaine considéré alors que les schémas d’axiome caractérisent des propriétés classiques des concepts et relations. La syntaxe graphique d’OCGL utilisée pour exprimer de tels axiomes est basée sur le modèle des Graphes Conceptuels. Ainsi, un axiome est composé d’une partie Antécédent et d’une partie Conséquent, la sémantique formelle d’une telle construction pouvant s’exprimer intuitivement comme suit : *si la partie Antécédent est vraie, alors la partie Conséquent est vraie*. La figure 2 présente le graphe représentant l’axiome « les ennemis de mes amis sont mes ennemis ».

2.2 Le domaine d’expérimentation OntoFamily

Afin d’illustrer nos idées, nous considérons dans cet article le domaine simple (mais intuitif) des relations familiales. Ce domaine volontairement limité inclut les notions suivantes : *père, mère, grand-père, grand-mère, fils, fille, cousin, cousine, neveu, nièce, oncle, tante, sœur, frère, épouse, mari, ami, ennemi*. En plus

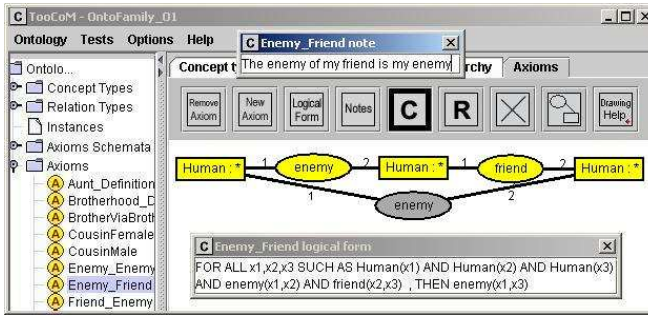


FIG. 2 – Représentation d’un axiome dans TooCoM. Les nœuds clairs composent la partie antécédent, les nœuds foncés la partie conséquent. Chaque partie contient des nœuds concepts (rectangles) et des nœuds relations (ellipses).

d’être facilement compréhensible, ce domaine est particulièrement intéressant du fait qu’il requiert la définition d’axiomes de domaine (par exemple, « *les ennemis de mes ennemis sont mes amis* »), non représentables par des schémas d’axiome. Dans le cadre d’un cours d’Ingénierie Ontologique, deux groupes d’étudiants de niveau Master ont travaillé séparément à la définition d’une ontologie de ce domaine. Cette expérience a conduit à la construction de deux ontologies : OntoFamily O_1 et OntoFamily O_2 ². O_1 est composée de 3 types de concepts qui définissent une partition (*Human*, un concept abstrait, et ses 2 sous-concepts *Man* et *Woman* qui sont disjoints), 31 relations binaires structurées en un treillis de profondeur 3, 11 schémas d’axiomes (1 abstraction, 1 disjonction, 1 exclusivité, 3 symétries, 4 cardinalités et 1 transitivité), et 18 axiomes. O_2 est composée de 3 types de concepts (*Human*, qui n’est pas abstrait, et ses 2 sous-concepts *Male* et *Female* qui sont disjoints), 23 relations binaires structurées en un arbre de profondeur 2, 10 schémas d’axiomes et 27 axiomes.

3 Alignement basé sur les axiomes : les principes

L’objectif du processus d’alignement d’ontologies est de découvrir et d’évaluer des liens d’identité entre primitives conceptuelles (concepts et relations) de deux ontologies données, supposées être construites à partir de domaines connexes. La méthode que nous proposons repose sur l’utilisation du niveau axiomatique des ontologies pour découvrir des analogies sémantiques entre primitives, de façon à révéler des identités entre elles et à calculer leur coefficient de similarité, *i.e.* le coefficient qui indique la proximité sémantique de deux concepts ou relations. Deux principes régissent notre approche : (1) l’utilisation de la *stabilité de modélisation* et de la *rareté* d’une propriété conceptuelle pour fixer le poids de

2. Ces ontologies sont disponibles en CGXML sur <http://sourceforge.net/projects/toocom/>. CGXML est le format XML de stockage d’OCGL.

cette dernière dans l'évaluation des appariements et (2) l'utilisation des *métra-représentations* des axiomes de domaine pour comparer leurs structures.

3.1 Postulats concernant la stabilité et la rareté

Les propriétés conceptuelles d'un domaine, exprimées en OCGL par des schémas d'axiome et des axiomes de domaine, peuvent être modélisées de différentes façons. Par exemple, deux hiérarchies de concepts peuvent être différentes si des concepts additionnels sont ajoutés ou non pour les structurer (par exemple le concept *Human* aurait pu être omis dans OntoFamily O_1). De façon similaire, la signature d'une relation peut être différente d'une modélisation à l'autre (par exemple, la notion de *cousin* peut être représentée par seulement une relation $cousin(Human, Human)$, ou par deux relations en introduisant le genre $cousinF(Woman, Human)$ et $cousinM(Man, Human)$). La stabilité de modélisation d'une propriété (schéma d'axiome ou axiome de domaine) correspond à son degré de stabilité d'une ontologie à l'autre. La recherche de correspondances entre ontologies doit favoriser les propriétés possédant les meilleures stabilités de modélisation, les analogies entre ces propriétés étant les plus pertinentes.

De plus, au sein d'une même ontologie, une propriété peut être très commune ou, au contraire, extrêmement rare. Par exemple, la propriété de symétrie est très commune. Mais un axiome de domaine est, par définition, très particulier et seulement quelques axiomes de même sémantique formelle peuvent exister dans des ontologies portant sur un domaine donné. La rareté d'une propriété dans les ontologies considérées accroît le poids des appariements découverts grâce à cette propriété. Donc, au début du processus d'alignement, ces raretés doivent être évaluées de façon à adapter le poids de chaque propriété aux caractéristiques des ontologies à aligner.

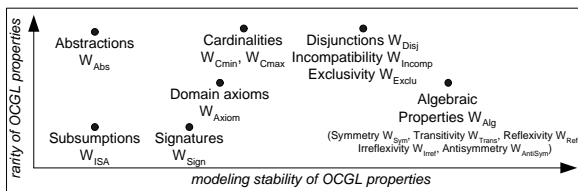


FIG. 3 – *Stabilités de modélisation et raretés des propriétés d'OCGL. Les subsumptions et les signatures sont très communes et peu stables. Les axiomes de domaine et les propriétés algébriques sont moins répandues, et la stabilité des axiomes de domaine est plus faible que celle des propriétés algébriques du fait que les représentations des propriétés algébriques sont fixées. Les cardinalités sont moins stables que les propriétés qui lient deux concepts ou relations (disjonction, incompatibilité et exclusivité), car à la fois leurs valeurs et les relations sur lesquelles elles portent peuvent varier.*

La figure 3 présente les valeurs relatives de stabilité et de rareté des propriétés

d’OGCL. Ces évaluations des stabilités et raretés ne sont que des postulats permettant de fixer les poids par défaut pour chaque propriété. Ces poids peuvent être modifiés par l’utilisateur dans l’outil pour améliorer les résultats du processus d’alignement. Par défaut, les valeurs des poids ont été ordonnées comme suit, à partir de considérations expérimentales : $W_{Alg}(W_{Sym}, W_{Trans}, W_{Refl}, W_{Irref}, W_{AntiSym}) > W_{Disj} = W_{Incomp} = W_{Exclu} > W_{Cmin} = W_{Cmax} > W_{Axiom} > W_{Sign} > W_{Abst} > W_{ISA}$.

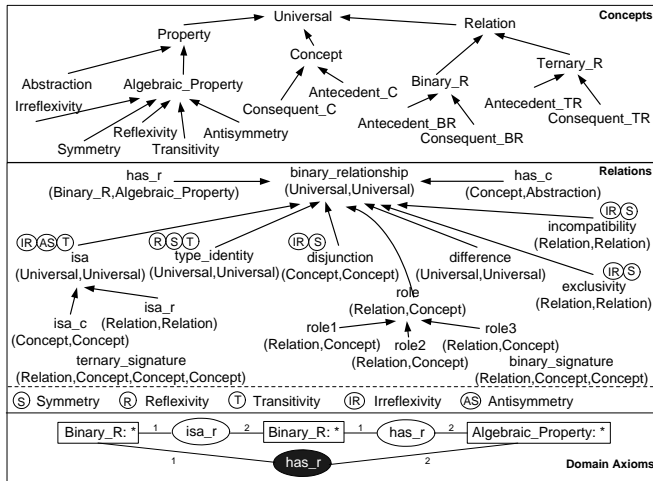


FIG. 4 – Les hiérarchies de concepts et relations de MetaOCGL. L’axiome présenté (en bas de la figure) exprime l’héritage d’une propriété algébrique.

3.2 MetaOCGL : une ontologie de représentation

Afin de détecter des analogies entre axiomes représentés par des graphes, et ainsi de détecter des analogies entre les primitives correspondant aux nœuds des graphes, les axiomes de domaine sont transcrits sous une forme plus abstraite, qui préserve les structures topologiques des graphes initiaux. Ces représentations abstraites sont basées sur une ontologie du langage OCGL, exprimée en OCGL, et appelée **MetaOCGL**. Comme indiqué en figure 4, MetaOCGL inclut tous les concepts d’OCGL et ses relations (relation *isa*, exclusivité/incompatibilité entre relations, disjonction de concepts, liens entre relations et concepts dans un graphe qui exprime un axiome). La figure 5 présente les graphes OCGL dédiés à la représentation des deux axiomes « *les ennemis de mes ennemis sont mes amis* » et « *les ennemis de mes amis sont mes ennemis* », ainsi que les méta-graphes correspondant en MetaOCGL.

Les comparaisons entre axiomes représentés en MetaOCGL sont réalisées grâce à l’opérateur de *projection* du modèle des Graphes Conceptuels, opération correspondant à un homomorphisme de graphes : étant donné deux graphes G_1 et

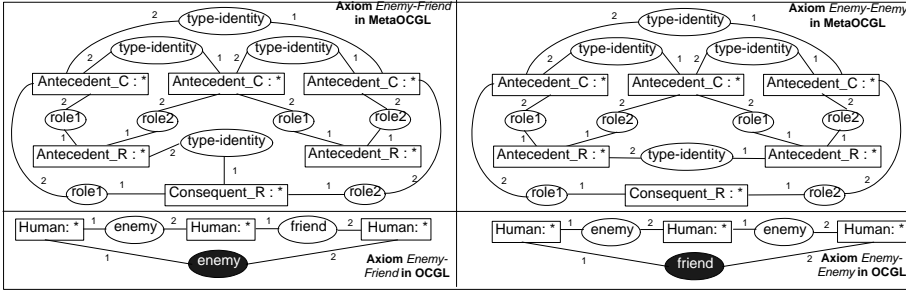


FIG. 5 – Deux axiomes d’OntoFamily O_1 représentés en MetaOCGL. Le lien `type_identity` indique que les nœuds ont le même type dans l’axiome.

G_2 , qui représentent en MetaOCGL deux axiomes A_1 et A_2 , si deux projections existent de G_1 dans G_2 et de G_2 dans G_1 , alors A_1 et A_2 ont la même structure ; dans ce cas, les axiomes A_1 et A_2 expriment le même type de propriété, et l’analogie entre les deux axiomes peut être étendue aux primitives qui apparaissent dans les axiomes.

4 Alignement basé sur les axiomes : l’algorithme

Notre algorithme prend en entrée deux ontologies O_1 et O_2 , représentées en OCGL, et produit en sortie des similarités potentielles entre 2 concepts ou 2 relations. Le résultat est un ensemble d’appariements (p_i, p'_j, c) , où p_i et p'_j sont respectivement des primitives conceptuelles (concepts ou relations) de O_1 et O_2 , et c le coefficient de similarité entre p_i et p'_j . Bien entendu, étant donné une primitive p_i de O_1 , plusieurs appariements peuvent exister avec des primitives de O_2 (ou aucun), et vice versa. Les schémas d’axiome et les axiomes de domaine sont utilisés pour évaluer ou découvrir des appariements de primitives. Le poids de chaque propriété d’OCGL est utilisé pour moduler son influence sur l’évaluation des appariements.

4.1 Utiliser les schémas d’axiome pour évaluer les appariements

Les schémas d’axiome qui impliquent seulement une primitive (*i.e.* propriétés algébriques et abstractions) sont comparées de O_1 à O_2 , afin de découvrir des appariements de primitives. Si les primitives de l’appariement (p_1, p_2, c) portent toutes deux un même schéma d’axiome, le coefficient c est augmenté du poids du schéma ; si l’appariement n’existe pas, il est créé avec ce poids ; si seule une des primitives porte le schéma, c est diminué du poids du schéma (ou l’appariement est créé avec le poids négatif s’il n’existe pas). Les schémas d’axiome qui impliquent deux primitives (*i.e.* disjonctions, incompatibilités et exclusivités) sont utilisés de même, mais concernent les 4 appariements possibles des 2 primitives.

Min Card	Relation r_1 in Ontology 1	Relation r_2 in Ontology 2	Action
	0 (resp. $c_{min} \geq 1$)	$c_{min} \geq 1$ (resp. 0)	$-2 * W_{cmin}$
	$c_{min} \neq 0$	$c_{min} \neq 0$	$+2 * W_{cmin}$
	$c1_{min} \neq 0$	$c2_{min} \neq 0$ and $\neq c1_{min}$	$-W_{cmin}$
Max Card	Relation r_1 in Ontology 1	Relation r_2 in Ontology 2	Action
	∞ (resp. $c_{max} \geq 1$)	$c_{max} \geq 1$ (resp. ∞)	$-W_{cmax}$
	$c_{max} \neq \infty$	$c_{max} \neq \infty$	$+2 * W_{cmax}$
	$c1_{max} \neq \infty$	$c2_{max} \neq \infty$ and $\neq c1_{max}$	$-2 * W_{cmax}$

TAB. 1 – Modifications du coefficient de l'appariement (r_1, r_2, c) en fonction des cardinalités des relations. c_{min} et c_{max} sont les valeurs des cardinalités.

Le tableau 1 présente les différentes opérations induites par la prise en compte des analogies ou différences de cardinalités. Si l'appariement entre les relations considérées n'existe pas alors qu'une analogie entre cardinalités est découverte, l'appariement est créé avec le coefficient correspondant.

4.2 Utiliser les axiomes de domaine pour évaluer les appariements

Pour chaque couple d'axiomes (a_1, a_2), où $a_1 \in O_1$ et $a_2 \in O_2$, les représentations de a_1 et a_2 en MetaOCGL, $meta(a_1)$ et $meta(a_2)$, sont construites. Ces représentations peuvent être enrichies par l'ajout de relations d'identité de type entre les nœuds (cf. figure 5). Deux types d'équivalence topologique sont considérés : l'**équivalence**, qui est attestée lorsque deux projections existent de $meta(a_1)$ dans $meta(a_2)$ et de $meta(a_2)$ dans $meta(a_1)$, sans considérer les relations *type_identity*, et l'**équivalence typée** attestée quand deux projections existent en tenant compte des relations *type_identity*. Bien entendu, le poids d'une équivalence typée est plus élevé que celui d'une équivalence. Une équivalence typée (resp. équivalence) entre deux axiomes augmente le coefficient des nœuds liés par projection, du poids de l'équivalence typée (resp. équivalence). Si aucune projection n'existe (ou seulement une), aucune modification n'intervient. Par exemple, les deux axiomes de la figure 5 sont équivalents car 2 projections existent entre leurs méta-graphes sans considérer les relations *type-identity*. En considérant les relations *type-identity*, il n'existe pas de projection, les axiomes ne sont donc pas équivalents typés.

4.3 Résoudre les conflits d'appariements

Du fait qu'elles lient une relation et un ensemble de concepts, les signatures sont uniquement utilisées pour augmenter ou diminuer les coefficients des appariements, et non pour en créer de nouveaux. En outre, deux usages possibles sont distingués : (1) la modification des coefficients d'appariements de relations à partir des appariements entre concepts de leurs signatures et (2) la modification des coefficients d'appariements de concepts à partir des appariements entre relations possédant ces concepts dans leurs signatures. Par exemple, si les relations $mother1(Woman1, Human1)$ de O_1 et $parent2(Human2, Human2)$ de O_2 sont appariées, les coefficients des appariements de concepts ($Woman1, Human2$)

et $(Human1, Human2)$, s'ils existent, sont augmentés de W_{Sign} . Mais si les appariements $(Woman1, Human2)$ et $(Human1, Human2)$ existent, le coefficient de l'appariement de relations $(mother1, parent2)$ peut aussi être augmenté de W_{Sign} . Le choix entre ces deux possibilités est basé sur l'utilisation d'un **Matching Ratio** appelé MR_c (resp. MR_r) et défini entre le nombre d'appariements de concepts (resp. relations) et le nombre moyen de concepts (resp. relations) dans O_1 et O_2 : si MR_c est supérieur à MR_r (i.e. la proportion d'appariements de concepts découverts est supérieure à celle des appariements de relations), les signatures sont utilisées pour réévaluer les coefficients des appariements de relations existant à l'aide des appariements de concepts existant, et vice versa. Ainsi, avant d'utiliser les signatures, les appariements de concepts (ou de relations) doivent être résolus, et ensuite les signatures sont utilisées pour affiner les coefficients des appariements de relations (ou de concepts).

La résolution des appariements de concepts ou de relations repose sur les valeurs de leurs coefficients : quand plusieurs appariements existent pour une primitive donnée, l'appariement avec le coefficient le plus élevé est retenu. Bien entendu, des contradictions peuvent exister entre 2 ou plusieurs appariements, que les valeurs des coefficients ne permettent pas toujours de résoudre. Dans ce cas, l'intervention de l'utilisateur est nécessaire pour trancher.

5 Résultats expérimentaux

L'application de notre algorithme sur OntoFamily O_1 et OntoFamily O_2 (avant la phase de résolution des appariements) produit 201 appariements, dont 9 couples de concepts (parmi 9 possibles), et 192 couples de relations (parmi 713 possibles)³. Beaucoup d'appariements n'apparaissent qu'à l'occasion de moins de 4 comparaisons d'axiomes, et jamais lors de la comparaison de schémas d'axiomes. Ils sont donc rejetés du fait de leur faible pertinence et seuls 9 couples de concepts et 69 couples de relations sont ainsi considérés pour la phase de résolution. Les valeurs des rapports MR_c et MR_r sont respectivement de 3 et ≈ 2.55 , les couples de concepts sont donc résolus avant les couples de relations. Le couple $(Human, Human)$ possède le coefficient le plus élevé des appariements liant les concepts *Human*. Ensuite, le couple de plus fort coefficient est $(Woman, Female)$ et le troisième couple de plus fort coefficient est $(Man, Male)$. Tous les liens entre concepts des 2 ontologies sont donc retrouvés par l'algorithme.

Les signatures sont ensuite utilisées pour affiner les valeurs des coefficients des couples de relations. La résolution des couples de relations ainsi constitués conduit à 9 liens entre relations de O_1 et O_2 (parmi 17 liens réellement possibles) : $(aunt, aunt)$, $(daughter, daughter)$, $(enemy, enemy)$, $(father, father)$, $(friend, friend)$, $(husband, husband)$, (son, son) , $(uncle, uncle)$ et $(wife, wife)$. Ces résultats, bien que perfectibles, s'avèrent encourageants étant donné le faible niveau de complexité des 2 ontologies considérées. En effet, notre approche pourrait

3. Un rapport détaillé de cette expérimentation est disponible sur <http://sourceforge.net/projects/toocom/>.

s'avérer encore plus pertinente dans un contexte d'ontologies plus hétérogènes et plus riches à la fois en termes de concepts, relations et axiomes. Cette expérience a également montré que l'usage des sousomptions peut améliorer l'efficacité de l'algorithme. Par exemple, les couples de relations (*parent,mother*) et (*parent,father*) ont de forts coefficients, ce qui pourrait permettre de déduire l'appariement (*parent,parent*).

6 Discussion

Actuellement, de nombreux outils sont proposés pour découvrir des correspondances entre ontologies (Kalfoglou & Schorlemmer, 2003). La première façon de classer ces outils est de considérer l'objectif visé : (1) fusionner deux ontologies pour en créer une nouvelle (*e.g.* iPROMPT (Noy & Musen, 2003)), (2) définir une fonction de transformation d'une ontologie à l'autre (*e.g.* OntoMorph (Chalupsky, 2000)) ou (3) déterminer des correspondances entre concepts (*e.g.* ANCHORPROMPT (Noy & Musen, 2003), GLUE (Doan *et al.*, 2004), S-MATCH (Giunchiglia *et al.*, 2004), FCA-Merge (Stumme & Maedche, 2001) ou ASCO (Bach *et al.*, 2004)). Notre travail s'inscrit dans le cadre de ce dernier objectif tout en y ajoutant la détermination de correspondances entre relations. Une autre façon de caractériser les outils est de considérer le type de représentations utilisées : (1) les noms de classes et les définitions en langage naturel, (2) les hiérarchies de classes et leurs propriétés (ANCHORPROMPT, GLUE), (3) les instances de classes (FCA-Merge) ou (4) les descriptions de classes (S-MATCH). Certains outils s'appuient sur plusieurs représentations, par exemple l'algorithme ASCO utilise à la fois (1), (2) et (4) (Bach *et al.*, 2004). Notre approche repose à la fois sur (2) et (4) et ajoute un nouveau type d'entrée : les axiomes de domaine.

Ensuite, dans (Ehrig & Sure, 2004), les différentes mesures utilisées pour l'alignement d'ontologies sont organisées au sein d'une échelle de similarité à 5 niveaux, intégrant des représentations de plus en plus élaborées dans les mesures : *Entities*, *Semantic Nets*, *Description Logics*, *Restrictions* et *Rules*. Au premier niveau, deux concepts sont égaux s'ils ont même(s) label(s) ; au second niveau, deux concepts sont égaux s'ils ont mêmes attributs ; au troisième niveau, deux concepts sont égaux s'ils ont les mêmes concepts parents. Cependant, pour les niveaux *Restrictions* et *Rules*, aucune mesure n'est proposée. Notre travail doit être considéré comme une extension de cette classification dans le sens où il offre une mesure de similarité basée sur les axiomes de domaine, qui englobent à la fois les niveaux *Restrictions* et *Rules*, puisque nous estimons qu'il n'est pas possible de considérer les règles et les contraintes au niveau ontologique, et que, de notre point de vue, les deux niveaux *Restrictions* et *Rules* doivent être réunis dans un unique niveau baptisé *Axioms* (Fürst *et al.*, 2004).

Enfin, afin de comparer les différentes méthodes de calcul de distances entre entités, (Euzenat & Valtchev, 2004) proposent la taxinomie suivante :

- *Terminological (T)*: comparaison des labels des entités, décomposée en

(TS), approche purement syntaxique et (TL) utilisant en plus un lexique de synonymes et d'hyponymes ;

- *Internal structure comparison (I)* : comparaison des structures internes des entités (par exemple la valeur d'une cardinalité) ;
- *External structure comparison (S)* : comparaison des relations entre entités, décomposée en (ST), comparaison des entités au sein de leurs taxinomies et (SC) comparaison des structures externes en tenant compte des cycles ;
- *Extensional comparison (E)* : comparaison des extensions des entités ;
- *Semantic comparison (M)* : comparaison des interprétations (plus exactement des modèles) des entités.

L'originalité de notre travail est de s'attaquer à la *Semantic Comparison (M)*, via l'utilisation des axiomes de domaine. Comme rappelé dans (Euzenat & Valtchev, 2004), la seule approche travaillant sur les interprétations des entités est l'algorithme de (Giunchiglia *et al.*, 2004), qui utilise un moteur de raisonnement pour déterminer des subsomptions ou équivalences de classes à partir des équivalences initiales de quelques classes et de l'analyse des relations taxonomiques. Notre travail se différencie de cette approche par la prise en compte non seulement de la relation *sorte-de*, mais également de toutes les représentations sémantiques des entités incluant les schémas d'axiome et les axiomes de domaine. Il est à noter que nous considérons également I et ST, mais pas E. Ce travail étend certains travaux visant à rapprocher deux ontologies exprimées dans le modèle des GCs, mais ne considérant que les hiérarchies de concepts et de relations (et les instances de concepts), et basés sur la comparaison des labels des primitives et leur contexte (place dans les hiérarchies) (Dieng & Hug, 1998).

7 Conclusion

Dans cet article, nous avons introduit une nouvelle approche de l'alignement d'ontologies, principalement basée sur l'utilisation des axiomes. Cette approche a été définie dans un contexte particulier où les composants des ontologies sont représentés à l'aide de graphes et manipulés via les mécanismes de raisonnement du modèle des Graphes Conceptuels. Contrairement aux méthodes d'alignement proposées jusqu'alors, qui pour la plupart considèrent uniquement un sous-ensemble des composants des ontologies légères, notre méthode a pour avantage de prendre en compte tous les composants des ontologies lourdes, vers lesquelles tendent de nombreux travaux, en particulier dans le cadre du Web sémantique.

Ce travail préliminaire se poursuit actuellement sur l'étude de la contribution possible des liens de subsomption et des instances à la méthode d'alignement proposée. Nous étudions également l'apport que pourrait constituer la prise en compte du processus inverse de celui présenté dans ce papier, c'est-à-dire non plus utiliser les axiomes pour découvrir et évaluer des identités entre

concepts/rerelations, mais utiliser les identités précédemment découvertes entre concepts/rerelations pour découvrir et évaluer des identités entre axiomes.

Références

- BACH T., DIENG-KUNTZ R. & GANDON F. (2004). On Ontology Matching Problems - for Building a Corporate Semantic Web in a Multi-Communities Organization. In *Proceedings of the 6th International Conference on Enterprise Information Systems (ICEIS'2004)*, volume 4, p. 236–243.
- CHALUPSKY H. (2000). OntoMorph: A Translation System for Symbolic Knowledge. In *Principles of Knowledge Representation and Reasoning*, p. 471–482.
- DIENG R. & HUG S. (1998). Comparison of « Personal Ontologies » Represented through Conceptual Graphs. In H. PRADE, Ed., *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI'1998)*, p. 341–345: John Wiley and Sons.
- DOAN A., MADHAVAN J., DOMINGOS P. & HALEVY A. (2004). Ontology Matching: A Machine Learning Approach. In *Handbook on Ontologies in Information Systems*, p. 397–416.
- EHRIG M. & SURE Y. (2004). Ontology Mapping - an integrated approach. In *Proceedings of the First European Semantic Web Symposium*, p. 76–91: Springer-Verlag (LNCS 3053).
- EUZENAT J. & VALTCHEV P. (2004). Similarity-based ontology alignment in OWL-Lite. In R. L. DE MANTARAS & L. SAITTA, Eds., *European Conference on Artificial Intelligence (ECAI'2004)*, p. 333–337: IOS Press.
- FÜRST F., LECLÈRE M. & TRICHET F. (2004). Operationalizing domain ontologies: a method and a tool. In R. L. DE MANTARAS & L. SAITTA, Eds., *European Conference on Artificial Intelligence (ECAI'2004)*, p. 318–322: IOS Press.
- GIUNCHIGLIA F., SHVAIKO P. & YATSKEVICH M. (2004). S-Match: an Algorithm and an Implementation of Semantic Matching. In *Proceedings of the First European Semantic Web Symposium*, p. 61–65: Springer-Verlag (LNCS 3053).
- GOMEZ-PEREZ A., FERNANDEZ-LOPEZ M. & CORCHO O. (2003). *Ontological Engineering*. Springer, Advanced Information and Knowledge Processing.
- KALFOGLOU Y. & SCHORLEMMER M. (2003). Ontology mapping: the state of the art. *The Knowledge Engineering Review*, **18**(1), 1–31.
- NOY N. F. & MUSEN M. (2003). The PROMPT suite: Interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies*, **59**(6), 983–1024.
- SOWA J. (1984). *Conceptual Structures : information processing in mind and machine*. Addison-Wesley.
- STAAB S. & MAEDCHE A. (2000). *Axioms are objects too: Ontology Engineering beyond the modeling of concepts and relations*. Research report 399, Institute AIFB, Karlsruhe.
- STUMME G. & MAEDCHE A. (2001). FCA-MERGE: Bottom-up merging of ontologies. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI'2001)*, p. 225–234.