
SODA : Une approche structurelle pour l’alignement d’ontologies OWL-DL

Sami Zghal^{*,} — Sadok Ben Yahia^{**}
Engelbert Mephu Nguifo^{*} — Yahya Slimani^{**}**

** CRIL CNRS FRE 2499, Université d’Artois, IUT de Lens
Rue de l’Université - S.P. 16, 62307 Lens Cedex, France
{zghal, mephu}@cril.univ-artois.fr*

*** Département des Sciences de l’Informatique, Faculté de Sciences de Tunis
Campus Universitaire, 1060 Tunis, Tunisie
sami.zghal@planet.tn, {sadok.benyahia, yahya.slimani}@fst.rnu.tn*

RÉSUMÉ. L’alignement d’ontologies représente un grand intérêt dans le domaine de la gestion des connaissances hétérogènes. La littérature du domaine propose plusieurs méthodes d’alignement d’ontologies. Ces méthodes exploitent différents formats d’ontologies mais très peu s’intéressent au format OWL-DL. L’alignement d’ontologies repose sur le calcul des mesures de similarité. Ce papier décrit une nouvelle méthode d’alignement d’ontologies OWL-DL. Elle propose une approche d’alignement d’ontologies qui définit un modèle global de calcul de similarité. Nous présentons aussi une discussion sur les résultats d’expérimentations réalisées sur des bases test d’ontologie.

ABSTRACT. Ontology Matching is of great interest in knowledge management domain especially when dealing with heterogenous knowledge. Different approaches have been reported for ontology alignment. Those approaches are based on similarity measurements. They also deal with different types of ontology format. This paper describes a novel ontology alignment method for OWL-DL format. The new method used a different approach that consists in computing local and global similarities. A thorough experimentation of this method has been conducted on different standard benchmarks, and the results are presented and discussed.

MOTS-CLÉS : Alignement d’ontologies, similarités locale et globale, similarité structurelle, OWL-DL.

KEYWORDS: Ontology matching, local and global similarities, structural similarity, OWL-DL.

1. Introduction

La première définition de l'ontologie dans le domaine de l'informatique est proposée par Neches et *al.*, (Neches et *al.*, 1991). Ils définissent l'ontologie comme : "*les termes et les relations de base comportant le vocabulaire d'un domaine aussi bien que les règles pour combiner les termes et les relations afin de définir des extensions du vocabulaire*" (Neches et *al.*, 1991). Les ontologies sont étudiées dans le domaine de l'intelligence artificielle. Elles permettent la représentation des connaissances, et elles sont aussi évoquées dans le Web sémantique. Une définition consensuelle, précise et complète des ontologies dans le contexte du Web sémantique n'existe pas encore (Bach, 2006). Cependant, Gruber en 1993 proposait la définition la plus citée. Il définit l'ontologie comme étant "*une spécification explicite d'une conceptualisation*" (Gruber, 1993). La conceptualisation est le résultat d'une analyse du domaine étudié et l'abstraction du monde de ce domaine. Cette conceptualisation est représentée dans une forme concrète où les concepts, les relations ainsi que les contraintes sont explicitement définis dans un format et langage formel.

Partant du fait que plusieurs connaissances peuvent prendre des représentations différentes, nous trouvons de nos jours plusieurs ontologies de domaine pour un même champ d'application. Il est alors nécessaire de disposer d'outils permettant de faire le lien entre les connaissances exprimées dans chacune des ontologies. Ainsi, l'alignement d'ontologies permet aussi de réconcilier, d'un point de vue sémantique, les opinions de plusieurs experts (Bach et *al.*, 2004). Le problème d'alignement d'ontologies (connu aussi comme l'intégration d'ontologies, l'intégration sémantique, correspondance d'ontologies, etc.) joue un rôle central dans le développement des systèmes à base de connaissances. Les nouvelles technologies augmentent l'utilisation des ontologies dans le Web sémantique pour la représentation des connaissances. Cette tendance a conduit au développement de nouvelles ontologies, d'où le nombre élevé des ontologies disponibles sur le Web. Le Web sémantique exploite les ontologies dans la représentation des connaissances (Charlet et *al.*, 2005). Cette exploitation est essentielle dans la réutilisation des ontologies dans des systèmes permettant leur manipulation. Cette manipulation est réalisée par l'alignement des ontologies. Les techniques d'alignement jouent un rôle crucial dans la construction d'un lien sémantique entre les ontologies d'un même domaine. Quelques approches d'alignement (Aleksovski et *al.*, 2006, Stuckenschmidt et *al.*, 2004, van Hage et *al.*, 2005) considèrent que l'utilisation d'une connaissance sur le domaine est une manière assurant la correspondance sémantique entre la dissimilarité syntaxique des ontologies. L'obtention de la bonne connaissance sur le domaine est primordiale. D'autres approches n'exploitent pas une connaissance sur le domaine et ne réalisent pas un modèle sémantique formel pour l'alignement des structures produites. Dans ce cas, la structure obtenue est difficile à exploiter, *e.g.*, pour répondre aux requêtes interrogeant les ontologies (Lopez et *al.*, 2006). En outre, les approches courantes d'alignement d'ontologies (Noy, 2004, Shvaiko et *al.*, 2005, Bach, 2006) sont basées sur les mesures de similarité entre chaînes de caractères et des structures composites. Les ontologies à aligner peuvent être représentées avec différents langage.

Dans la littérature, plusieurs méthodes d'alignement d'ontologies ont été proposées. Ces méthodes exploitent des ontologies décrites dans différents langages (RDF(S), DAML+OIL, OWL, etc.). Les méthodes d'alignement ANCHOR-PROMPT (Noy, 2004), NOM (Ehrig *et al.*, 2004b), QOM (Ehrig *et al.*, 2004a) et ASCO1 (Bach, 2006) exploitent des ontologies décrites avec le langage RDF(S). Les méthodes d'alignement OLA (Euzenat *et al.*, 2004a, Euzenat *et al.*, 2004b) et EDOLA (Zghal *et al.*, 2007a, Zghal *et al.*, 2007b) alignent des ontologies représentées avec le langage OWL-Lite. La méthode ASCO2¹ (Bach, 2006) considère des ontologies OWL-DL (Smith *et al.*, 2004) (Ontology Web Language Description Logic). Étant donné que le langage OWL est un standard pour les ontologies, toute méthode d'alignement n'exploitant pas ce format présente un inconvénient. Plusieurs méthodes d'alignement, telles que OLA, ASCO1 et , ASCO2 exploitent un processus de stabilisation de la similarité à travers un seuil pour aboutir à l'alignement. Ce seuil limite la propagation de la similarité à travers le voisinage

La nouvelle méthode d'alignement, SODA (Structural Ontology OWL-DL Alignment), implémente un nouvel algorithme d'alignement d'ontologies OWL-DL. La nouvelle méthode d'alignement repose sur le calcul des mesures de similarité. Il définit deux modèles de calcul de similarité (locale et globale). La méthode combine les mesures de similarité locale (terminologique et structurelle) pour l'évaluation de la similarité globale. Elle permet de générer un alignement exploitant l'aspect structurel du voisinage des entités à apparier.

L'article est organisé comme suit. La deuxième section présente les langages de représentation d'ontologies. La troisième section définit le problème d'alignement ainsi que les mesures d'évaluation des alignements. La quatrième section décrit la méthode d'alignement d'ontologies ASCO2. Dans la cinquième section, la nouvelle méthode SODA d'alignement d'ontologies OWL-DL est introduite. La sixième section illustre une évaluation expérimentale. La conclusion et les travaux futurs font l'objet de la dernière section.

2. Langages de représentation d'ontologies

Un langage ontologique est un langage formel permettant de représenter les différents éléments constituant une ontologie. Plusieurs langages sont proposés pour la description des ontologies : RDF(S) (RDF et RDF Schema) (Klyne *et al.*, 2004), DAML+OIL (Connolly *et al.*, 2001) et OWL (Smith *et al.*, 2004). Une ontologie permet de décrire des connaissances d'un domaine. Il semble intéressant de représenter l'ontologie dans un langage expressif. Le langage RDF(S) ne permet pas représenter la cardinalité d'une relation. Il n'exprime pas aussi les caractéristiques des relations : la transitivité, la symétrie, la fonctionnalité, etc. De même, il ne permet pas les restrictions pour certaines classes. Ainsi, le W3C a recommandé un langage standardisé pos-

1. La méthode ASCO2 est la seule méthode d'alignement d'ontologies qui considère les ontologies OWL-DL.

sédant un niveau d'expressivité plus élevé. Le langage d'ontologie recommandé par W3C est le langage OWL. Ce langage est inspiré du langage DAML+OIL. Il couvre la majorité des caractéristiques du langage DAML+OIL, en attribuant de nouveaux noms à la plupart de ses primitives. Le langage OWL est spécialement développé pour la représentation des ontologies dans le cadre du Web sémantique. Ce langage permet aussi de créer, partager et échanger des connaissances dans le Web sémantique (Bach, 2006).

Le langage d'ontologie OWL offre trois sous-langages (OWL-Lite, OWL-DL et OWL-Full) avec une puissance d'expressivité ascendante (Bach, 2006). La raison de cette division concerne la complexité, la calculabilité et l'implémentation du langage. Le sous-langage OWL-Lite possède la complexité formelle la plus basse et l'expressivité minimale. Il est suffisant pour la représentation des thésaurus et d'autres taxonomies ou des hiérarchies de classification avec des contraintes simples. Le sous-langage OWL-DL, possède une expressivité maximale tout en maintenant les propriétés de complétude computationnelle (*i.e.*, toutes les conclusions sont garanties d'être calculées) et de décidabilité (*i.e.*, tous les calculs finiront dans le temps fini). Le sous langage OWL-DL correspond à la variante de la logique de description $SHOIN(\mathcal{D})$ (Horrocks *et al.*, 2003). Le sous-langage OWL-DL est approprié pour la représentation des ontologies ayant besoin de la puissance d'expressivité tout en gardant la calculabilité. Le sous-langage OWL-Full est conçu pour les développeurs, les implémenteurs et les utilisateurs qui ont besoin de l'expressivité maximale, de la liberté syntaxique de RDF mais sans se soucier d'une garantie de calculabilité. Actuellement, il n'existe pas encore d'outils ou de logiciels de raisonnement capables de supporter des raisonnements complets pour toutes les caractéristiques de OWL-Full (Bach, 2006).

En ce qui concerne de la compatibilité de ces sous-langages, le sous-langage OWL-Full peut être considéré comme une extension de RDF, tandis que OWL-Lite et OWL-DL peuvent être considérés comme des extensions d'une vue restreinte du langage RDF. OWL-Full est une extension de OWL-DL, et ce dernier est à son tour une extension de OWL-Lite. Une ontologie valide en OWL-Lite est aussi valide en OWL-DL et OWL-Full. Tous les documents en OWL (Lite, DL ou Full) sont des documents valides en RDF. Un document RDF est un document OWL-Full, mais seulement quelques documents en RDF sont des documents valides en OWL-Lite ou OWL-DL (Bach, 2006). La nouvelle méthode d'alignement développée aligne des ontologies exprimées dans le langage OWL-DL puisqu'il offre une expressivité maximale, une complétude computationnelle et une décidabilité. La section suivante définit l'alignement, les différentes mesures de similarité utilisées au cours du processus d'alignement ainsi que les métriques d'évaluation de l'alignement obtenu.

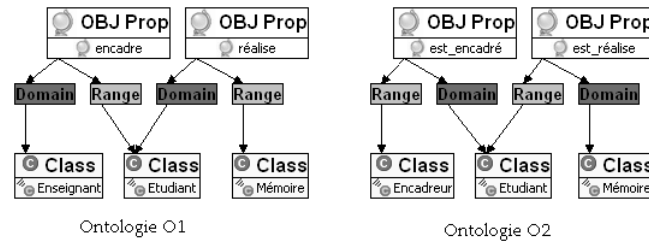


Figure 1. Exemples de connaissances de deux ontologies représentées sous forme de graphes

3. État de l'art sur l'alignement d'ontologies

L'alignement de deux ontologies revient à trouver une correspondance entre leurs entités qui sont sémantiquement similaires (Ehrig *et al.*, 2004b). D'une façon formelle, l'alignement est défini par la fonction *map* comme suit :

$$\text{map} : O \longrightarrow O' \text{ tel que } \text{map}(e_1) = e'_1 \quad \text{si } \text{sim}(e_1, e'_1) > t,$$

où O et O' sont les deux ontologies à aligner, t désigne un seuil minimal de similarité appartenant à l'intervalle $[0,1]$, $e_1 \in O$ et $e'_1 \in O'$. e_1 et e_2 représentent les entités au niveau des deux ontologies. Le seuil t indique le niveau minimum pour que deux entités soient similaires.

La figure 1 présente deux exemples de représentation de connaissances de deux ontologies. La première ontologie, $O1$, indique qu'un enseignant encadre un étudiant qui réalise son mémoire. La seconde ontologie, $O2$, indique qu'un mémoire est réalisé par un étudiant, qui est encadré par un enseignant. L'alignement des deux ontologies $O1$ et $O2$ revient à déterminer la correspondance entre les différentes entités ontologiques par catégorie. Le tableau 1 permet de donner les différentes entités à comparer au niveau des deux ontologies. Toutes les méthodes d'alignement déterminent des correspondances entre les entités ontologiques en utilisant des mesures de similarité (Zghal *et al.*, 2007b).

3.1. Mesures de similarité

Les différentes mesures de similarité utilisées dans le processus d'alignement sont organisées selon la classification suivante (Rahm *et al.*, 2001) :

Entité de l'ontologie O_1	Entité de l'ontologie O_2
Encadre	Est_Encadrer, Est_réaliser
Réalise	Est_Encadrer, Est_réaliser
Enseignant	Encadreur, Étudiant, Mémoire
Étudiant	Encadreur, Étudiant, Mémoire
Mémoire	Encadreur, Étudiant, Mémoire

Tableau 1. Entités à comparer au niveau des deux ontologies à aligner

1) **La méthode terminologique** : compare les labels des entités. Elle est décomposée en approches purement syntaxiques et celles utilisant un lexique. L'approche syntaxique effectue la correspondance à travers les mesures de dissimilarité des chaînes (*e.g.*, EditDistance). Tandis que, l'approche lexicale effectue la correspondance à travers les relations lexicales (*e.g.*, synonymie, hyponymie, etc.) ;

2) **La méthode de comparaison des structures internes** : compare les structures internes des entités (*e.g.*, intervalle de valeur, cardinalité d'attributs, etc.) ;

3) **La méthode de comparaison des structures externes** : compare les relations d'entités avec d'autres. Elle est décomposée en méthodes de comparaison des entités au sein de leurs taxinomies et méthodes de comparaison des structures externes en tenant compte des cycles ;

4) **La méthode de comparaison des instances** : compare les extensions des entités, *i.e.*, elle compare l'ensemble des autres entités qui lui sont attachées (instances des classes) ;

5) **La méthode sémantique** : compare les interprétations (ou plus exactement les modèles) des entités.

3.2. Métriques d'évaluation

Les mesures de *Précision*, *Rappel*, *Fallout* et *Fmesure* (Do *et al.*, 2002) ont été des métriques largement exploitées pour évaluer la qualité des alignements obtenus. Le EON² "Evaluation of Ontology-based Tools" (EON, 2004, EON, 2006, Euzenat *et al.*, 2006) retient ces mesures pour l'évaluation de la qualité de l'alignement. L'objectif principal de ces mesures est l'automatisation du processus de comparaison des méthodes d'alignement ainsi que l'évaluation de la qualité des alignements produits. La première phase dans le processus d'évaluation de la qualité de l'alignement consiste à résoudre le problème manuellement. Le résultat obtenu manuellement est considéré comme l'alignement de référence. La comparaison du résultat de l'alignement de référence avec celui de l'appariement obtenu par la méthode d'alignement produit trois

2. <http://oaei.ontologymatching.org/2004/Contest/> et <http://km.aifb.uni-karlsruhe.de/ws/eon2006/>

ensembles : N_{found} , $N_{expected}$ et $N_{correct}$. L'ensemble N_{found} représente les paires alignées avec la méthode d'alignement. L'ensemble $N_{expected}$ désigne l'ensemble des couples appariés dans l'alignement de référence. L'ensemble $N_{correct}$ est l'intersection des deux ensembles N_{found} et $N_{expected}$. Il représente l'ensemble des paires appartenant à la fois à l'alignement obtenu et l'alignement de référence. La *précision* est le rapport du nombre de paires pertinentes trouvées, *i.e.*, " $N_{correct}$ ", rapporté au nombre total de paires, *i.e.*, " N_{found} ". Il renvoie ainsi, la partie des vraies correspondances parmi celles trouvées. Ainsi, la fonction *précision* est définie par :

$$précision = \frac{|N_{correct}|}{|N_{found}|}.$$

Le *rappel* est le rapport du nombre de paires pertinentes trouvées, " $N_{correct}$ ", rapporté au nombre total de paires pertinentes, " $N_{expected}$ ". Il spécifie ainsi, la part des vraies correspondances trouvées. La fonction *rappel* est définie par :

$$rappel = \frac{|N_{correct}|}{|N_{expected}|}.$$

La mesure *Fallout* permet d'estimer le pourcentage d'erreurs obtenu au cours du processus d'alignement. Elle est définie par le rapport des paires erronées, " $(N_{found} - N_{correct})$ ", rapporté au nombre total des paires trouvées, " N_{found} ", *i.e.*,

$$Fallout = \frac{|N_{found}| - |N_{correct}|}{|N_{found}|}.$$

La mesure *Fmesure* est une moyenne harmonique. Cette mesure combine la *précision* et le *rappel*. La mesure *Fmesure* est définie par (EON, 2004) :

$$Fmesure = \frac{2 \times précision \times rappel}{précision + rappel}.$$

Étant donné que le langage OWL constitue un standard pour la représentation des ontologies et en particulier le langage OWL-DL, la section suivante passe en revue la méthode ASCO2 d'alignement d'ontologies OWL-DL.

4. Méthode d'alignement d'ontologies OWL-DL : ASCO2

L'algorithme ASCO2 propose un modèle de calcul de similarité sur deux étapes (Bach, 2006) : la similarité partielle et la similarité finale. La similarité partielle entre deux entités des deux ontologies est déduite entre les composantes correspondantes aux entités en question. Ces composantes sont des pièces de connaissance contenues

dans les définitions de l'entité en employant des primitives du langage OWL. Les valeurs de similarité partielle sont ensuite agrégées dans un schéma de pondération variable pour obtenir une meilleure valeur de similarité finale de ces deux entités (Bach, 2006).

Une ontologie en OWL-DL est un document RDF valide et les descriptions des entités en OWL-DL sont les ensembles des triplets RDF. Étant donné le triplet (s, p, o) un triplet, une ontologie O en OWL-DL est un ensemble de triplets RDF, noté $\mathcal{O}=\{(s_i, p_i, o_i)\}$ (Bach, 2006).

Une description d'une classe ou d'une relation de l'ontologie en OWL-DL est un ensemble de triplets RDF, notée $E=\{(s, p_i, o_i)\}$, $E \subseteq O$, où s correspond à l'entité en question. Suivant la conceptualisation, l'ensemble de triplets d'une entité peut contenir un seul triplet ou plusieurs triplets. Il peut arriver que dans l'ensemble, il y ait plusieurs triplets ayant un même prédicat. Par exemple, le langage OWL permettant le multi-héritage, une classe peut être définie comme sous-classe de plusieurs classes. Dans ce cas, la description de la classe se compose de plusieurs triplets (Bach, 2006).

Ainsi, la comparaison de similarité entre deux entités e_1 et e_2 correspond à la comparaison de deux ensembles de triplets $E_1=\{(s_1, p_{1i}, o_{1i})\}$ et $E_2=\{(s_2, p_{2j}, o_{2j})\}$ représentant les deux entités à aligner. La comparaison de deux ensembles de triplets dépend des prédicats p_{1i} et p_{2j} et des objets o_{1i} et o_{2j} . Pour chaque prédicat p , nous obtenons deux ensembles $O_1=\{o_{1i}|(s_1, p, o_{1i})\}$ et $O_2=\{o_{2j}|(s_2, p, o_{2j})\}$. La similarité entre deux ensembles O_1 et O_2 est calculée, notée $S_{Ensemble}(O_1, O_2)$. Nous appelons cette similarité la similarité partielle de deux entités sur le prédicat p , notée $S_p(e_1, e_2)$ et définie par $S_p(e_1, e_2)=S_{Ensemble}(O_1, O_2)$ (Bach, 2006).

La similarité de deux ensembles est calculée à partir des calculs des similarités de leurs éléments respectifs. Les éléments dans ces ensembles sont de même type et donc ils sont comparables. Soient O_1 et O_2 deux ensembles d'éléments de même type. La similarité des deux ensembles est une fonction de similarité $S_{Ensemble} : 2^E \times 2^E \rightarrow \mathbb{R}$, telle que :

$$S_{Ensemble} = \frac{\sum_{o_1 \in O_1} MSim(o_1, O_2) + \sum_{o_2 \in O_2} MSim(o_2, O_1)}{|O_1| + |O_2|}$$

où $MSim(o_i, O) = \max_{o_j \in O} (Sim(o_i, o_j))$. Le type de ces éléments sont le co-domaine du prédicat. Les éléments peuvent être des littéraux tels que les chaînes de caractères, les nombres, les classes, les propriétés, les instances et les ontologies (Bach, 2006).

La similarité finale de deux entités e_1 et e_2 , notée $S_{Finale}(e_1, e_2)$, est calculée en agrégeant les valeurs de similarité partielle. L'agrégation est réalisée dans un schéma de pondération variable. La similarité finale S_{Finale} est stockée dans la matrice de similarité $MSim$ (qui contient toutes les valeurs de similarité finales de deux entités de deux ontologies). Les valeurs de similarité dans cette matrice sont réutilisées pour calculer la similarité de deux autres entités. Ainsi, l'algorithme ASCO2 exploite un calcul d'équation à point fixe pour mettre à jour les valeurs de similarité entre deux

entités après chaque itération. La valeur de la similarité finale est obtenue après un certain nombre d'itérations ou quand les valeurs de similarité dans la matrice M_{Sim} deviennent stables, *i.e.*, la différence entre les valeurs de deux itérations consécutives est inférieure à un seuil prédéfini (Bach, 2006).

La matrice de similarité M_{Sim} est initialisée par les valeurs de similarité linguistique de deux entités calculées à partir de trois prédicats en employant des mesures de similarité. Seuls les trois prédicats des triplets ayant des objets du type textuel sont pris en compte pour calculer les similarités partielles. Les valeurs, ainsi agrégées, sont pondérées avec des poids prédéfinis pour obtenir la valeur de similarité linguistique finale (Bach, 2006).

La méthode ASCO2 calcule la similarité finale en exploitant la similarité partielle ainsi qu'un seuil de stabilisation de la similarité finale. Ce seuil peut handicaper la propagation de la similarité à travers le voisinage. La nouvelle méthode d'alignement proposée permet d'exploiter le voisinage (l'aspect structurel) des entités à apparier afin de déterminer les couples d'entités les plus similaires en se basant sur le calcul de similarité locale et globale.

5. Nouvelle méthode d'alignement d'ontologies

La nouvelle méthode d'alignement d'ontologies SODA, que nous introduisons, prend en entrée deux ontologies décrites en format OWL-DL. Les ontologies OWL-DL à apparier sont transformées sous forme d'un graphe DL-GRAPH que nous introduisons aussi.

Le graphe DL-GRAPH représente toutes les informations contenues dans une ontologie OWL-DL (Smith *et al.*, 2004) : les classes, les relations et les instances. Les nœuds du graphe sont les entités de l'ontologie : les classes, les propriétés et les instances. Les arcs du graphe décrivent les relations qui existent entre ces différentes entités. Chaque entité de l'ontologie OWL-DL est décrite par une ressource, représentée par une notion associée dans le formalisme RDF (Klyne *et al.*, 2004), et identifiée par un URI. Les entités dans l'ontologie OWL-DL sont décrites moyennant des constructeurs du langage OWL. Ces descriptions sont représentées sous forme des triplets RDF : sujet, prédicat et objet. Les entités à représenter sont les sujets des triplets. Les prédicats du triplet sont des primitives OWL, tandis que les objets en sont les ressources. La description d'une classe ou d'une relation dans une ontologie OWL-DL est réalisée par un triplet RDF. Le sujet correspond à la classe ou à la relation. Les prédicats sont des primitives OWL, qui représentent les propriétés du langage OWL et celle du langage RDF. Elles sont employées dans les descriptions de l'entité pour la définir. Chaque propriété, utilisée dans un triplet, apporte une connaissance à propos de l'entité à décrire. La combinaison de toutes ces connaissances construit une définition de l'entité.

La nouvelle méthode d'alignement, SODA, est une approche reposant sur un modèle de calcul des similarités locale et globale. Ce modèle exploite la structure du

<ul style="list-style-type: none"> - O_1, O_2 : les deux ontologies à aligner en format OWL-DL - V_{SL} : le vecteur de la similarité linguistique - V_{SS} : le vecteur de la similarité structurelle - V_{VG} : le vecteur de la similarité globale <p>Les caractéristiques de chaque de nœuds sont :</p> <ul style="list-style-type: none"> - Type : la catégorie du nœud - Nom : le nom du nœud
<p>Les vecteurs V_{SL}, V_{SS} et V_{VG} se caractérisent par :</p> <ul style="list-style-type: none"> - Nœud 1 : le nœud de l'ontologie O_1 - Nœud 2 : le nœud de l'ontologie O_2 - Sim : la valeur de similarité

Tableau 2. Notations utilisées dans les algorithmes PHASE1_SIMLING, PHASE2_SIMSTRUC et PHASE3_SIMGLOB

graphe DL-GRAPH pour apparier les nœuds des deux ontologies et calculer les mesures de similarités. Le modèle d'alignement calcule pour chaque catégorie de nœuds, appartenant au graphe DL-GRAPH, une fonction d'agrégation. La fonction d'agrégation prend en considération toutes les mesures de similarité et la structure des nœuds à apparier. Ainsi, cette fonction exploite toute l'information descriptive de ce couple. L'algorithme qui implémente la méthode d'alignement SODA prend en entrée deux ontologies à aligner en format OWL-DL et fournit en sortie un fichier XML. La méthode SODA fonctionne en deux phases successives. La première phase, implémentée par le biais des fonctions PHASE1_SIMLING et PHASE2_SIMSTRUC, permet de calculer la similarité locale (linguistique et structurelle). La deuxième étape, *c.f.* la fonction PHASE3_SIMGLOB, permet de calculer la similarité globale, dite *sémantique*. Les notations utilisées dans les algorithmes développés sont résumées dans le tableau 2.

5.1. Calcul de la similarité locale

Le calcul de la similarité locale s'effectue en deux étapes successives. La première étape, permet de calculer la similarité linguistique pour chaque couple de nœuds appartenant à la même catégorie. La deuxième étape, permet de calculer la similarité structurelle en exploitant la structure du voisinage.

5.1.1. Calcul de la similarité linguistique

La mesure de similarité linguistique des couples d'entités de même type (classe, propriété et instance) est calculée par l'intermédiaire de l'Algorithme 1 (*c.f.*, la fonction PHASE1_SIMLING). Le calcul de la similarité linguistique est effectué entre les noms des entités pour les propriétés et les instances. Pour les classes, le calcul de la similarité linguistique intègre aussi les commentaires et les étiquettes. Le calcul de la similarité linguistique s'effectue une seule fois pour chaque nœud de même catégorie.

La similarité linguistique a été calculée par l'intermédiaire des fonctions de JARO-WINKLER ou MONGE-ELKAN (Monge *et al.*, 1996). La fonction PHASE1_SIMLING permet de calculer les similarités linguistiques des couples de nœuds des deux ontologies. Elle prend en entrée les deux ontologies O_1 et O_2 à aligner, représentées sous la forme de deux graphes DL-GRAPH, ainsi que la fonction de similarité linguistique à utiliser, $Fonct_{SL}$, et donne en retour un vecteur de similarité linguistique, V_{SL} , de chaque couple de nœuds. La fonction CALCULSIMLING (*c.f.*, ligne 8 de l'Algorithme 1) prend en entrée deux nœuds, Nud_1 et Nud_2 , ainsi que la fonction de la similarité linguistique et retourne une valeur de similarité linguistique Sim_L . Cette fonction est assurée par l'une des méthodes de calcul de similarité suivante : la mesure de JARO-WINKLER ou MONGE-ELKAN. La mesure de JARO-WINKLER est adaptée pour les chaînes de caractères courtes, telles que celles représentant des noms et des étiquettes (Monge *et al.*, 1996). Par contre, la mesure MONGE-ELKAN est plus appropriée pour les chaînes longues telles que les commentaires (Monge *et al.*, 1996). La similarité linguistique pour les différents couples d'entités est exploitée par la suite pour le calcul de la similarité structurelle. La section suivante décrit en détail le processus de calcul de la similarité structurelle de SODA.

```

1 Fonction : PHASE1_SIMLING
  Données :
    1)  $O_1$  et  $O_2$  : deux ontologies à aligner
    2)  $Fonct_{SL}$  : fonction de similarité linguistique
  Résultat :  $V_{SL}$  : vecteur de similarité linguistique
xdébut
3  /* parcours des nœuds de l'ontologie  $O_1$  */
4  pour chaque ( $Nud_1 \in O_1$ ) faire
5    /* parcours des nœuds de l'ontologie  $O_2$  */
6    pour chaque ( $Nud_2 \in O_2$ ) faire
7      si  $Nud_1.type == Nud_2.type$  alors
8         $Sim_L = \text{CALCULSIMLING}(Nud_1, Nud_2, Fonct_{SL})$ 
9        /* Ajouter  $Nud_1$ ,  $Nud_2$  et  $Sim_L$  à  $V_{SL}$  */
10       AJOUTER( $(Nud_1, Nud_2, Sim_L), V_{SL}$ )
11  retourner( $V_{SL}$ )
1xfin

```

Algorithme 1. PHASE1_SIMLING

5.1.2. Calcul de la similarité structurelle

La similarité structurelle est calculée en exploitant la similarité linguistique du couple de nœud à appairer ainsi que la structure avoisinante. Les nœuds voisins adjacents de l'entité à aligner sont classés par catégorie, *c.f.*, PHASE2_SIMSTRUC. Par la suite, la fonction PHASE2_SIMSTRUC calcule la mesure de similarité pour chaque deux ensembles de même catégorie. La fonction PHASE2_SIMSTRUC prend en en-

trée les deux ontologies à aligner O_1 et O_2 , le vecteur de similarité linguistique V_{SL} et les poids assignés à chaque catégorie Π_C . Elle produit comme résultat un vecteur de similarité structurelle V_{SS} . La fonction EXTRAITRENŒUDS, (c.f., lignes 9 - 11 de l'Algorithme 2), permet d'extraire pour chaque nœud à aligner les nœuds du voisinage et de les stocker dans les vecteurs V_{Nud_i} , où Nud_i désigne un nœud de O_1 ou O_2 . Les deux vecteurs V_{Nud_1} et V_{Nud_2} ainsi que le poids de la similarité de chaque catégorie, Π_C , sont exploités par la fonction CALCULSIMSTRUCT (c.f., ligne 13 de l'Algorithme 2) pour le calcul de la similarité structurelle, Sim_S . Pour effectuer ce calcul, la mesure de similarité "Match-Based similarity" (Touzani, 2005) est utilisée :

$$MSim(E, E') = \frac{\sum_{(i,i') \in Paires(E,E')} Sim(i, i')}{Max(|E|, |E'|)},$$

où E et E' représentent deux ensembles de nœuds de même catégorie dans les vecteurs V_{Nud_1} et V_{Nud_2} respectivement. Cette fonction requiert que les similarités linguistique des couples (i, i') soient déjà calculées et donne comme résultat les couples de l'ensemble $P = E \times E'$. Les couples (i, i') , intervenant dans le calcul, doivent présenter les meilleurs mesures de similarité. La similarité structurelle, Sim_S , est calculée comme suite :

$$Sim_S = \sum_{(E,E') \in (V_{Nud_1}, V_{Nud_2})} \Pi_{(E,E')} MSim(E, E').$$

La similarité, Sim_S , est normalisée puisque $\sum(\Pi_{(E,E')}) = 1$. Les valeurs de similarité des deux vecteurs de similarité linguistique, i.e. V_{SL} , et structurelle, i.e. V_{SS} , ainsi obtenues sont combinées pour le calcul de la similarité globale. La section suivante décrit en détail le processus de calcul de la similarité globale.

5.2. Calcul de la similarité globale

La similarité globale est une mesure de similarité agrégée des deux similarités linguistique et structurelle (similarité locale). L'Algorithme 3 prend en entrée les deux ontologies à aligner, O_1 et O_2 , les deux vecteurs des similarités linguistique, V_{SL} , et structurelle, V_{SS} , ainsi que les poids attribués pour la similarité linguistique et structurelle (Π_L et Π_S). Il produit en résultat un vecteur de similarité globale, V_{SG} . La fonction EXTRAIRESIM (c.f., lignes 9 - 11 de l'Algorithme 3) permet d'extraire du vecteur de similarité (V_{SL} ou V_{SS}) la valeur de la similarité correspondante aux deux nœuds (Nud_1 et Nud_2). Pour chaque couple d'entités, Nud_1 et Nud_2 , appartenant à une même catégorie des deux ontologies à aligner, O_1 et O_2 , la similarité globale est calculée comme suit (c.f., ligne 13 de l'Algorithme 3) :

$$Sim_G(e_1, e_2) = \Pi_L Sim_L(e_1, e_2) + \Pi_S Sim_S(e_1, e_2).$$

```

1 Fonction : PHASE2_SIMSTRUC
Données :
    1)  $O_1$  et  $O_2$  : deux ontologies à aligner
    2)  $V_{SL}$  : vecteur de similarité linguistique
    3)  $\Pi_C$  : les poids de la similarité de chaque catégorie de nœuds
Résultat :  $V_{SS}$  : vecteur de similarité structurelle
début
3  /* parcours des nœuds de l'ontologie  $O_1$  */
4  pour chaque ( $Nud_1 \in O_1$ ) faire
5      /* parcours des nœuds de l'ontologie  $O_2$  */
6      pour chaque ( $Nud_2 \in O_2$ ) faire
7          si  $Nud_1.type == Nud_2.type$  alors
8              /*Extraire dans le vecteur  $V_{Nud_1}$  les nœuds voisins de  $Nud_1$ */
9               $V_{Nud_1} = \text{EXTRAITRENŒUDS}(Nud_1)$ 
10             /*Extraire dans le vecteur  $V_{Nud_2}$  les nœuds voisins de  $Nud_2$ */
11              $V_{Nud_2} = \text{EXTRAITRENŒUDS}(Nud_2)$ 
12             /*calculer la similarité structurelle*/
13              $Sim_S = \text{CALCULSIMSTRUC}(V_{Nud_1}, V_{Nud_2}, \Pi_C)$ 
14             /* Ajouter  $Nud_1$ ,  $Nud_2$  et  $Sim_S$  à  $V_{SS}$ */
15              $\text{AJOUTER}((Nud_1, Nud_2, Sim_S), V_{SS})$ 
16 retourner( $V_{SS}$ )
17fin

```

Algorithme 2. PHASE2_SIMSTRUC

Dans la section suivante, une évaluation expérimentale de la méthode SODA est présentée.

6. Évaluation expérimentale et discussion

L'évaluation expérimentale de la méthode d'alignement SODA a été menée sur l'aspect, dit "inter-méthodes", qui permet de comparer les résultats qualitatifs obtenus par la méthode SODA à la méthode, ASCO2. Dans le cadre des expérimentations menées, quelques tests fournis dans la base benchmark mise à la disposition de la communauté par la compétition EON (EON, 2004, EON, 2006) sont utilisés. Les principaux tests sont décrits dans le tableau 3 (EON, 2004, EON, 2006). L'ontologie de base est constituée par un ensemble de références bibliographiques. Elle représente une version plus allégée en nombre d'entités ontologiques comparativement avec des ontologies réelles. Chaque cas de test de la base benchmark met en exergue une caractéristique de la deuxième ontologie à aligner avec la base de test. L'objectif de cette base de tests est de prendre en charge tous les aspects qui existent dans une ontologie

```

1 Fonction : PHASE3_SIMGLOB
   Données :
       1)  $O_1$  et  $O_2$  : deux ontologies à aligner
       2)  $V_{SL}$  : vecteur de similarité linguistique
       3)  $V_{SS}$  : vecteur de similarité structurelle
       4)  $\Pi_L$  et  $\Pi_S$  : les poids respectifs des similarités linguistique et structurelle
   Résultat :  $V_{SG}$  : vecteur de similarité globale
    début
3   /* parcours des nœuds de l'ontologie  $O_1$  */
4   pour chaque ( $Nud_1 \in O_1$ ) faire
5       /* parcours des nœuds de l'ontologie  $O_2$  */
6       pour chaque ( $Nud_2 \in O_2$ ) faire
7           si  $Nud_1.type == Nud_2.type$  alors
8               /*Extraire la similarité linguistique de  $Nud_1$  et  $Nud_2$  de  $V_{SL}$ */
9                $Sim_L = \text{EXTRAIRESIM}(V_{SL}, Nud_1, Nud_2)$ 
10              /*Extraire la similarité structurelle de  $Nud_1$  et  $Nud_2$  de  $V_{SS}$ */
11               $Sim_S = \text{EXTRAIRESIM}(V_{SS}, Nud_1, Nud_2)$ 
12              /*calculer la similarité globale*/
13               $Sim_G = \Pi_L Sim_L + \Pi_S Sim_S$ 
14              /* Ajouter  $Nud_1$ ,  $Nud_2$  et  $Sim_S$  à  $V_{SG}$ */
15              AJOUTER( $(Nud_1, Nud_2, Sim_G), V_{SG}$ )
16   retourner( $V_{SG}$ )
17 fin

```

Algorithme 3. PHASE3_SIMGLOB

OWL-DL et qui pourraient avoir un impact considérable sur les métriques d'évaluation du résultat de l'alignement.

Le tableau 4 récapitule les résultats obtenus par les deux méthodes d'alignement SODA et ASCO2 (Bach, 2006). En se basant sur les valeurs des métriques de la qualité de l'alignement (mesures de *précision*, *rappel*, *Fallout* et *Fmesure*), la méthode SODA présente de meilleurs résultats (*c.f.*, tableau 4) dans presque la majorité des tests. Dans ce cadre, il est important de rappeler que la méthode SODA effectue un calcul de la similarité globale en exploitant la similarité locale (linguistique et structurelle). La méthode SODA exploite la notion de structure du voisinage pour le calcul de la similarité structurelle. Elle classe les nœuds adjacents par catégorie et calcule par la suite la similarité maximale par catégorie. L'alignement, produit par la méthode SODA à chaque test, est comparé à l'alignement de référence. Ainsi, les valeurs des mesures de qualité d'alignement (*précision*, *rappel*, *Fallout* et *Fmesure*) sont calculées.

Les meilleurs résultats des valeurs de précisions de SODA sont obtenus lorsque les structures d'ontologies sont semblables ou identiques, *i.e.*, les tests 10x, 22x et 23x. Ainsi, SODA obtient des valeurs de précision pour ces tests qui sont égales à

Test	Caractéristiques de l'ontologie
101	La même ontologie de base
103	Les axiomes non reconnus sont remplacés par leur généralisation
104	Les contraintes non disponibles ont été supprimées
201	Les libellés des entités sont remplacés par des libellés arbitraires
202	Les libellés sont remplacés par d'autres aléatoires. Les commentaires sont supprimés
203	Les libellés sont remplacés par d'autres aléatoires consistants
204	Des conventions d'appellation différentes sont employées. Les commentaires sont supprimés
205	Les noms des entités sont remplacés par leurs synonymes
221	Les assertions de sous-classe liées aux classes nommées sont supprimées
222	La hiérarchie des classes est maintenue mais elle est strictement réduite
223	De nombreuses classes intermédiaires sont ajoutées à la hiérarchie
224	Les individus sont supprimés
225	Les restrictions de classes exprimées par des propriétés sont supprimées
228	Les propriétés et les relations entre les objets sont complètement supprimées
230	Quelques composants de classes sont étendus dans la structure de classe
231	Certaines classes sont subdivisées sur autres classes
301	L'ontologie à comparer est réelle et semblable à l'ontologie de base
302	L'ontologie est très semblable à la précédente
303	L'ontologie est réelle
304	L'ontologie est aussi réelle et semblable à l'ontologie de base

Tableau 3. *Caractéristiques des principaux ensemble de tests d'ontologies de petite et moyenne taille*

1,00 (*c.f.*, au tableau 5). Ceci s'explique par le fait que l'approche SODA explore plus efficacement les structures avoisinantes des entités à aligner à travers la similarité structurelle. D'où, les entités qui ont presque la même structure sont correctement alignés. Les résultats des tests où la valeur de précision est moins bonne s'explique par le fait que l'algorithme SODA calcule les mesures de similarités des entités de même catégorie. Ceci induit que certains couples d'entités ne sont pas pris en considération par le processus d'alignement. D'où, l'ensemble des paires appartenant à la fois à l'alignement obtenu et l'alignement de référence, $N_{Correct}$, est faible. Par conséquent, la valeur de précision est affaiblie. En outre, les couples qui ont été exclus du processus d'alignement peuvent aider à l'augmentation des mesures de similarités des couples d'entités voisines et par conséquent, augmenter le nombre de couples correctement alignés. À partir des données présentées dans le tableau 4, la méthode d'alignement SODA se montre meilleure sur plusieurs tests par rapport à la méthode ASCO2. En effet, la méthode d'alignement, SODA, fournit des valeurs de mesures de qualité plus

Test	<i>Précision</i>		<i>Rappel</i>		<i>Fallout</i>		<i>Fmesure</i>	
	SODA	ASCO2	SODA	ASCO2	SODA	ASCO2	SODA	ASCO2
101	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
103	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
104	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
201	0,33	0,87	0,01	0,87	0,67	0,13	0,02	0,87
202	0,33	0,31	0,01	0,30	0,67	0,69	0,02	0,30
203	1,00	1,00	1,00	1,00	0,000	0,00	1,00	1,00
204	1,00	1,00	0,93	1,00	0,000	0,00	0,96	1,00
205	0,82	1,00	0,33	0,85	0,18	0,00	0,47	0,92
221	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
222	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
223	1,00	0,98	1,00	0,00	0,02	1,00	1,00	0,98
224	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
225	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
228	1,00	0,97	1,00	1,00	0,00	0,03	1,00	0,98
230	0,94	1,00	1,00	1,00	0,06	0,00	0,97	1,00
231	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
232	1,00	1,00	1,00	1,00	0,00	0,03	1,00	0,98
233	1,00	0,97	1,00	1,00	0,00	0,03	1,00	0,98
236	1,00	0,97	1,00	1,00	0,00	0,03	1,00	0,98
237	1,00	1,00	1,00	1,00	0,00	0,00	1,00	1,00
238	1,00	0,98	1,00	0,98	0,00	0,02	1,00	0,98
239	0,88	0,97	1,00	1,00	0,12	0,03	0,94	0,98
240	0,89	0,97	1,00	0,94	0,11	0,03	0,94	0,95
241	1,00	0,97	1,00	1,00	0,00	0,03	1,00	0,98
246	0,88	0,97	1,00	1,00	0,12	0,03	0,94	0,98
247	0,89	0,97	1,00	0,94	0,11	0,03	0,94	0,95
248	0,25	0,30	0,01	0,29	0,75	0,70	0,02	0,29
249	0,33	0,32	0,01	0,31	0,67	0,68	0,02	0,31
252	0,16	0,26	0,01	0,22	0,84	0,74	0,02	0,24
253	0,33	0,42	0,01	0,30	0,67	0,58	0,02	0,35
258	0,60	0,36	0,10	0,37	0,40	0,64	0,17	0,36
259	0,20	0,38	0,01	0,32	0,80	0,62	0,02	0,35
301	0,86	0,92	0,39	0,57	0,150	0,08	0,54	0,70
302	0,96	0,74	0,58	0,35	0,04	0,26	0,72	0,48
303	0,70	0,51	0,81	0,76	0,30	0,49	0,75	0,61
304	0,96	0,90	0,95	0,87	0,04	0,10	0,95	0,88

Tableau 4. Comparaison entre SODA et ASCO2

Test	<i>Précision</i>		<i>Rappel</i>	
	SODA	ASCO2	SODA	ASCO2
10x	1,00	1,00	1,00	1,00
20x	0,70	0,84	0,46	0,80
22x	1,00	0,99	1,00	1,00
23x	0,98	0,98	1,00	1,00
24x	0,71	0,75	0,67	0,75
25x	0,32	0,36	0,03	0,30
30x	0,87	0,77	0,68	0,64
Moyenne	0,80	0,81	0,70	0,78

Tableau 5. Comparaison entre SODA et ASCO2 par catégorie de tests

performantes sur la quasi-totalité des tests réels, tests 30x, *c.f.*, aux tableaux 4 et 5. Ces meilleurs résultats s'expliquent par les deux aspects suivants :

1) Le premier aspect est le fait que SODA exploite le voisinage des nœuds à aligner pour le calcul de la similarité globale, tandis que la méthode ASCO2 exploite la notion de similarité finale. La similarité globale est calculée en se basant sur la similarité locale des entités à aligner. Cette similarité locale comprend les similarités linguistique et structurelle. La similarité linguistique se calcule par l'intermédiaire des fonctions JARO-WINKLER ou MONGE-ELKAN. La similarité structurelle exploite la similarité linguistique ainsi que la similarité du voisinage par catégorie. Par contre, la méthode ASCO2 calcule la similarité partielle des entités à aligner. Cette similarité se limite au calcul de la similarité entre les descriptifs des entités à aligner ;

2) Le deuxième aspect est que la méthode SODA, calcule la similarité globale en agrégeant les éléments de la similarité locale (linguistique et structurelle). La similarité globale représente une mesure intégrée des différentes mesures de similarité des entités à aligner. Par contre, la méthode ASCO2 calcule la similarité finale uniquement en exploitant la similarité partielle. La similarité finale est calculée par un processus itératif (un certain nombre d'itérations) ou par une stabilisation de la similarité finale (Bach, 2006).

7. Conclusion

Dans ce papier, nous avons présenté une nouvelle méthode d'alignement d'ontologies OWL-DL. La nouvelle méthode d'alignement, SODA, développée exploite les graphes DL-GRAPH pour la recherche des meilleurs couples à appairer. Les résultats obtenus par le module d'alignement, SODA, sont satisfaisantes comparées aux résultats obtenus par la méthode d'alignement ASCO2. En outre, la méthode proposée offre des meilleurs résultats sur la majorité des tests réalisés par rapport à la méthode ASCO2. La méthode SODA est performante sur les petites et moyennes ontologies.

La méthode SODA n'est pas encore très bien adaptée pour les ontologies de grande taille. La prise en considération des ontologies réelles permet de rendre la méthode exploitable dans la confrontation des points de vues multiples.

Plusieurs améliorations sont possibles sur la méthode d'alignement, SODA, permettant de la rendre plus pertinente. Ces améliorations incluent : une prise en charge des ontologies réelles, un calcul plus riche et plus complet de la similarité terminologique, un calcul de similarité inter-catégorie. La prise en charge des ontologies réelles de grande taille par la réduction de l'espace de recherche en exploitant des techniques d'élagage des noeuds à apparier. La similarité terminologique est un facteur clé dans le modèle de calcul de la similarité globale. En effet, le gain en similarité terminologique contribue significativement à des améliorations des similarités finales. Ce gain peut être obtenu par l'utilisation des fonctions de similarités terminologiques sur toutes les données informationnelles des entités à savoir les noms, les libellés, etc. Le calcul de similarité inter-catégorie permet de chercher des similarités entre des catégories différents. Ceci augmente aussi bien le nombre de couples correctement alignés que les valeurs de similarités. Ce calcul contribue à une amélioration considérable de la pertinence de l'algorithme d'alignement.

Remerciements

Les auteurs tiennent à remercier Amine FERJANI et Marouen KACHROUDI pour les efforts fournis dans l'implémentation de la méthode SODA ainsi que les relecteurs pour leurs remarques constructives et pertinentes. Le présent travail est partiellement soutenu par le projet d'action intégrée Franco-Tunisien PAI CMCU 05G1412 intitulé : "fouille de données et parallélisme".

8. Bibliographie

- Aleksovski Z., Klein M., Kate W. T., Harmelen F. V., « Matching Unstructured Vocabularies using a Background Ontology », *Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management*, Hong Kong, China, p. 182-197, 2006.
- Bach T. L., Construction d'un web sémantique multi-points de vue, PhD thesis, École des Mines de Nice à Sophia Antipolis, 2006.
- Bach T. L., Dien-Kuntz R., Gandon F., « On Ontology Matching Problems - for Building a Corporate Semantic Web in a Multi-Communities Organization », *Proceedings of ICEIS*, Porto, Portugal, p. 236-243, 2004.
- Charlet J., Bachimont B., Troncy R., *Le Web sémantique*, vol. 4 of *Hors série de la revue Information - Interaction - Intelligence*, Cépaduès, chapter 4 : Ontologies pour le Web sémantique, p. 7-20, 2005.
- Connolly D., Harmelen F. V., Horrocks I., McGuinness D. L., Patel-Schneider P. F., Stein L. A., DAML+OIL : Reference Description, Technical report, W3C : World Wide Web Consortium, <http://www.w3.org/TR/2001/NOTE-daml+oil-reference-20011218>, December, 2001.

- Do H., Melnik S., Rahm E., « Comparison of schema matching evaluations », *Proceedings of the 2nd Int. Workshop on Web Databases*, German Informatics Society, Erfurt, Germany, p. 221-237, 2002.
- Ehrig M., Staab S., « QOM : Quick Ontology Mapping », *Proceedings of The 3rd ISWC, GI Jahrestagung (1)*, Hiroshima, Japon, p. 356-361, November, 2004a.
- Ehrig M., Sure Y., « Ontology Mapping - An Integrated Approach », *Proceedings of the 1st European Semantic Web Symposium*, vol. 3053, Springer Verlag, Hersounious, p. 76-91, 2004b.
- EON W., « EON 2004 : Ontology Alignment Contest », *Proceedings of the 3rd Workshop Evaluation of Ontology-based Tools (EON)*, <http://oaei.ontologymatching.org/2004/Contest/>, 2004.
- EON W., « EON 2006 : Evaluation of Ontologies for the Web », *Proceedings of the 4th International EON Workshop*, <http://km.aifb.uni-karlsruhe.de/ws/eon2006/>, 2006.
- Euzenat J., Loup D., Touzani M., Valtchev P., « Ontology Alignment with OLA », *Proceedings of the 3rd International Workshop : Semantic Web Conference EON*, Hirochima, Japan, p. 341-371, November, 2004a.
- Euzenat J., Mochol M., Shvaiko P., Stuckenschmidt H., Svab O., Svatek V., van Hage W. R., Yatskevich M., « Results of the Ontology Alignment Evaluation Initiative 2006 », *Proceedings of the First ESWC 2006 international workshop on ontology matching*, 2006.
- Euzenat J., Valtchev P., « Similarity-based ontology alignment in OWL-Lite », *Proceedings of the 15th European conference on Artificial Intelligence*, Valence, Espagne, p. 333-337, 2004b.
- Gruber T., « A translation approach to portable ontology specifications », *Knowledge Acquisition*, vol. 5, n° 2, p. 199-220, 1993.
- Horrocks I., Patel-Schneider P. F., van Harmelen F., « From SHIQ and RDF to OWL : The making of a web ontology langage », *Journal of Web Semantics*, vol. 1, n° 1, p. 7-26, 2003.
- Klyne G., Carroll J. J., Resource Description Framework (RDF) : Concepts and Abstract Syntax, Technical report, W3C : World Wide Web Consortium, <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>, February, 2004.
- Lopez V., Motta E., Uren V., « Next Generation Semantic Web Applications », *Proceedings of the First Asian Semantic Web Conference (ASWC)*, p. 24-29, 2006.
- Monge A., Elkan C., « The field-matching problem : algorithm and applications », *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, p. 267-270, 1996.
- Neches R., Fikes R., Finin T., Gruber T., Patil R., Senator T., Swartout W. R., « Enabling Technology for Knowledge Sharing », *Artificial Intelligence Magazine*, 36-56, 1991.
- Noy N., « Semantic Integration : A Survey of Ontology-based Approaches », *SIGMOD*, 2004.
- Rahm E., Bernstein P., « A survey of approaches to automatic schema matching », *VLDB Journal*, vol. 10, n° 4, p. 334-350, 2001.
- Shvaiko P., Euzenat J., « A Survey of Schema-based Matching Approaches », *Journal on Data Semantics*, 2005.
- Smith M. K., Welty C., Mcguinness D. L., OWL : Ontology Web Language Guide, Technical report, W3C : Word Wide Web Consortium, <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>, February, 2004.

- Stuckensschmidt H., Harmelen F. V., Serafini L., Bouquet P., Giunchiglia F., « Using C-OWL for the Alignment and Merging of Medical Ontologies », *Proceedings of the First Int. WS. on Formal Biomedical K. R. (KRMed)*, 2004.
- Touzani M., « *Alignement des ontologies OWL-Lite* », Master's thesis, University of Montreal, 2005.
- van Hage W., Katrenko S., Schreiber G., « A Method to combine Linguistic Ontology-Mapping Techniques », *Proceedings of International Semantic Web Conference*, Sardinia, Italy, p. 732-744, 2005.
- Zghal S., Kamoun K., Yahia S. B., Nguifo E. M., « Une nouvelle méthode d'alignement et de visualisation d'ontologies OWL-Lite », *Proceedings of 7^{ème} Journées francophones Extraction et Gestion des Connaissances EGC'2007*, vol. 1, Namur, Belgique, p. 197-198, 2007a.
- Zghal S., Kamoun K., Yahia S. B., Nguifo E. M., Slimani Y., « EDOLA : Une nouvelle méthode d'alignement d'ontologies OWL-Lite », *Proceedings of 4^{ème} conférence francophone en Recherche d'information et Applications CORIA'2007*, Saint-Étienne, France, p. 351-366, 2007b.