

Optimisation d'Alignement d'une Ontologie Multi-Points de Vue et une Ontologie Classique

Lynda Djakhdjakha¹, Mounir Hemam²,
¹Centre Universitaire de Souk Ahrass, Algérie
²Centre Universitaire de Khenchela, Algérie
¹ldjakhdjakha@yahoo.fr
²Mounir.hemam@gmail.com

Résumé. Dans ce papier nous proposons d'utiliser les métaheuristiques pour optimiser le résultat d'un alignement entre une ontologie multi-points de vue et une ontologie classique. L'idée est de représenter les ontologies à comparer par des graphes et de chercher du meilleur appariement parmi les appariements multivoques des nœuds des graphes, où chaque nœud du graphe de l'ontologie classique peut être apparié à zéro, à un ou à plusieurs nœuds du graphe de l'ontologie multi-points de vue dans laquelle chaque nœud appartient à un point de vue différent.

Mots clés : Optimisation, alignement, ontologie, appariement graphes, colonies de fourmis, points de vue.

1 Introduction

A l'origine, la plupart des métaheuristiques ont été développées pour résoudre les problèmes d'optimisation combinatoire mais de nos jours, beaucoup de recherches ont pris comme objet l'adaptation de ces méthodes aux autres types de problèmes telle que le problème d'alignement d'ontologies.

L'*alignement des ontologies* ayant pour objectif de permettre une utilisation conjointe de plusieurs ontologies. Le résultat de cette tâche assure et facilite l'échange, le partage, la fusion des données et des informations entre systèmes ou des communautés dans le Web sémantique. Il s'agit généralement de construire des *appariements* entre les éléments décrits dans différentes ontologies. Dans la littérature, plusieurs méthodes d'alignement d'ontologies ont été proposées. Elles tirent parti des différents aspects des ontologies. Et elles s'intéressent à l'alignement des ontologies décrites dans différents langages ontologiques. En 2007, et dans le même contexte, une méthode d'alignement EDOLA a été proposée dans [1]. Cette méthode utilise l'algorithme de recherche tabou pour aboutir à un alignement optimal. Par conséquent, la majorité de des méthodes d'alignement permettent de détecter seulement des relations entre des ontologies classiques qui ne prennent pas en compte la notion de multiples points de vue.

Dans ce travail, nous nous intéressons au problème de développement d'ontologies dans une organisation hétérogène en prenant en compte différents points de vue, différentes terminologies des personnes, des groupes voire des communautés diverses au sein de cette organisation. Une telle ontologie, appelée *ontologie multi-points de vue*, permet de faire cohabiter à la fois l'hétérogénéité et le consensus dans une

organisation hétérogène. A la différence d'une ontologie classique, une ontologie multi-points de vue confère à un même univers de discours plusieurs représentations différentes telles que chacune est relative à un point de vue particulier [2]. Ce besoin de prise en compte de connaissances multi-points de vue, au sein d'une même ontologie, provient essentiellement d'un environnement multidisciplinaire où plusieurs groupes de personnes diversifiés coexistent et collaborent entre eux. Chaque groupe a ses intérêts particuliers et perçoit différemment les propriétés et les relations particulières des entités conceptuelles du même univers de connaissances à représenter.

L'objectif de cet article est double, nous proposons, dans un premier temps, une approche d'alignement entre une ontologie multi-points de vue et une ontologie classique. Ensuite, nous essayons d'optimiser le résultat d'alignement. De ce fait, nous proposons de prendre en entrée deux ontologies décrites en logiques de description étendues par le mécanisme d'estampillage et de les transformer en des structures de graphes, en particulier, en deux *graphes étiquetés et attribués*. Les graphes sont considérés comme d'excellents outils permettant la représentation de données structurées. Ainsi, l'idée est d'appliquer un modèle de calcul de similarité entre les deux ontologies qui se réduit à la comparaison des deux graphes. D'un point de vue mathématique, le calcul de similarité entre deux graphes est réalisé par une recherche de morphisme de graphes. Ce genre de problème s'appelle *appariement de graphes* et est un problème NP-complet.

Dans le processus d'alignement, nous nous intéressons à la recherche du *meilleur appariement* parmi les *appariements multivoques* des nœuds des graphes, où chaque nœud du graphe de l'ontologie classique peut être apparié à zéro, à un ou à plusieurs nœuds du graphe de l'ontologie multi-points de vue dans laquelle chaque nœud appartient à un point de vue différent. Cette recherche peut se traduire en un problème de *sélection de sous-ensemble (SS-problème)*, dans le but de trouver un sous-ensemble qui *satisfait certaines propriétés*. Pour résoudre ce problème, nous utilisons une méthode d'optimisation par métaheuristique. Les métaheuristiques sont des méthodes approchées qui traitent les problèmes d'optimisation difficile. Le but d'un problème d'optimisation est de trouver une solution maximisant ou minimisant une fonction objectif donnée. Ainsi, les métaheuristiques constituent une classe de méthodes approchées adaptables à un grand nombre de problèmes d'optimisation combinatoire.

Nous proposons d'utiliser les algorithmes incomplets *d'optimisation par les colonies de fourmis* qui est une métaheuristique récente qui s'inspire de l'intelligence collective des fourmis. Dans notre travail l'algorithme d'optimisation par colonies de fourmis est paramétré par un ensemble de caractéristiques pour être capable de retourner le meilleur sous-ensemble de l'ontologie multi-points de vue qui est apparié à l'ontologie classique.

L'article est organisé comme suit. Dans la section 2, nous décrivons l'algorithme d'optimisation par colonies de fourmis. Dans la section 3, nous clarifions le problème de sélection de sous-ensemble. Dans la section 4, nous détaillons la notion d'une ontologie décrite en logique de description étendue par le mécanisme d'estampillage. Dans la section 5, nous présentons le processus d'alignement d'une ontologie multi-points de vue et une ontologie classique et incluons l'étape d'optimisation d'alignement. Dans la section 6, nous concluons et donnons quelques perspectives pour améliorer notre travail.

2 L'Algorithme d'Optimisation par Colonies de Fourmis

L'algorithme d'optimisation par colonies de fourmis est inspiré du comportement des fourmis à la recherche de nourriture, et a été mis au point par Dorigo en 1992 [3]. Son principe repose sur le comportement particuliers des fourmis qui, elles sont capable de déterminer le chemin le plus court entre leur nid et une source de nourriture grâce à la phéromone qui est une substance que les fourmis déposent sur le sol lorsqu'elles se déplacent. Lorsqu'une fourmi doit choisir entre deux directions, elle choisit avec une plus grande probabilité celle comportant une plus forte concentration de phéromone. Le principe de l'algorithme consiste à reformuler le problème à résoudre en un problème de recherche d'un meilleur chemin dans un graphe appelé graphe de construction et à utiliser des fourmis artificielles pour trouver les bons chemins de ce graphe. A chaque cycle de l'algorithme, chaque fourmi de la colonie construit aléatoirement un chemin du graphe et la quantité de la phéromone est déposée sur les meilleurs chemins découverts lors de ce cycle. Lors des cycles suivants, les fourmis construisent de nouveaux chemins avec une probabilité dépendant de la phéromone déposée lors des cycles précédents et d'une heuristique propre au problème considéré. La colonie de fourmis converge alors peu à peu vers les meilleures solutions.

3 Problème de Sélection de Sous-Ensemble

Les problèmes de sélection de sous-ensembles (SS-problème) ont pour but de trouver un sous-ensemble consistant et optimal d'objets. Plus formellement, un SS-problème est défini par un triplet $(S, S_{consistant}, f)$ tel que :

- S est l'ensemble d'objet ;
- $S_{consistant} \subseteq P(S)$ est l'ensemble de tous les sous-ensembles de S qui sont consistants, afin de pouvoir construire chaque sous-ensemble de $S_{consistant}$ de façon incrémentale (en partant de l'ensemble vide et en ajoutant à chaque itération un objet choisi parmi l'ensemble des objets consistants avec l'ensemble en cours de construction), on impose la contrainte suivante : pour chaque sous-ensemble consistant non vide $S' \in S_{consistant}$, il doit exister au moins un objet $o_i \in S'$ tel que $S' - \{o_i\}$ est aussi consistant.
- $f : S_{consistant} \rightarrow IR$ est la fonction objectif qui associe un coût $f(S')$ à chaque sous-ensemble d'objets consistant $S' \in S_{consistant}$.

Le but d'un SS-problème $(S, S_{consistant}, f)$ est de trouver $S^* \in S_{consistant}$ tel que $f(S^*)$ soit maximal.

4 Ontologie Multi-Points de Vue en Logique de Description Etendue par le Mécanisme d'Estampillage

Le mécanisme d'estampillage consiste à utiliser un index de point de vue PV_i comme préfixe aux expressions du langage local du point de vue PV_i pour délimiter les éléments de connaissances (i.e. concepts, rôles et individus) définis au sein de ce point de vue. Par ailleurs, une ontologie, décrite selon le modèle multi-points de vue proposé dans [2], [4] et [5], est constituée de trois principales parties : **1**) la partie

terminologie (TBox) qui regroupe l'ensemble des concepts globaux, les concepts locaux et les rôles locaux des différents points de vue. Un concept global est un concept vu à partir de deux ou plusieurs points de vue avec certaines caractéristiques communes (i.e. attributs et/ou relations). Un concept local est un concept qui est vu et décrit localement selon un point de vue donné. Un rôle local permet de représenter le rapport entre deux concepts d'un même point de vue **2**) la partie assertions (ABox) qui contient un ensemble d'assertions et de faits sur les individus. Un individu est une instance d'un concept défini selon un point de vue donné et pouvant avoir des rôles (liens) avec d'autres individus du même point de vue ou bien avec des individus d'autres points de vue **3**) la partie liens intermédiaires (LI) qui regroupe l'ensemble des rôles globaux et des passerelles. Un rôle global permet de relier deux concepts de deux points de vue différents et permet ainsi d'exprimer un fait général à propos des membres des concepts qui participent à ce rôle. Par ailleurs, une passerelle décrit une règle entre les concepts de deux ou plusieurs points de vue différents.

5 L'Approche Proposée

L'originalité de notre approche réside dans le fait d'optimiser le résultat d'alignement d'une ontologie multi-points de vue et une ontologie classique en utilisant un algorithme d'optimisation par colonies de fourmis. L'objectif donc est de mettre en correspondance les deux ontologies représentées par des graphes et d'optimiser par la suite ces correspondances afin d'obtenir un alignement optimal et plus similaire. Elle se compose de trois phases successives : le pré-alignement, le processus d'alignement et la phase de post-alignement et d'optimisation (voir la Figure 1).

5.1 Pré-Alignement

Dans la phase de pré-alignement, nous identifions d'abord les informations d'entrée (Input) qui constituent essentiellement les structures destinées à être alignées. Puis nous formalisons ces structures sous forme de graphe.

Format en entrée. Les deux ontologies en entrée sont décrites en logiques de description étendues par le mécanisme d'estampillage décrit précédemment.

Intervention humaine. Du fait qu'il est impossible d'automatiser complètement le processus d'alignement, une intervention de l'expert du domaine est absolument nécessaire pour effectuer les opérations suivantes :

Adaptation de l'ontologie classique. Un expert du domaine est sollicité pour identifier le nouveau point de vue correspondant à l'ontologie classique, les concepts globaux et les concepts locaux. Le préfixe PV_i est rajouté à chaque axiome de sa partie terminologique (TBox) et à chaque assertion de sa partie assertionnelle (ABox), pour avoir une ontologie *mono-point de vue* (i.e., un seul point de vue à considérer). La partie LI de l'ontologie mono-points de vue est vide, puisqu'il n'y a pas de relations globales ni de passerelles dans une ontologie mono-point de vue.

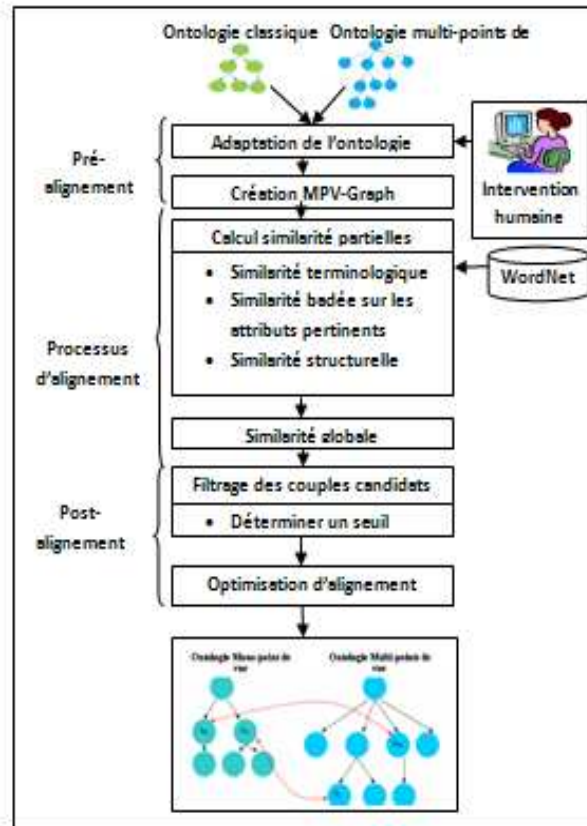


Fig. 1. Processus d'Alignement d'une Ontologie Multi-Points de Vue et une Ontologie Classique

Attribution et pondération des concepts locaux. Cette étape consiste à sélectionner pour chaque concept local les attributs les plus pertinents par rapport au points de vue visé, cette sélection peut se faire en évaluant l'importance des attributs un par un dans ce point de vue, puis trouver un vecteur réel de poids $W=(w_1, \dots, w_n)$. Ce vecteur est utilisé par la suite dans le module de calcul de similarité. Le but de cette étape est d'améliorer les correspondances entre les concepts locaux.

Création du MPV-Graph. Cette étape consiste à prendre les deux ontologies (mono et multi-points de vue) en entrée et les convertir en de deux graphes étiquetés et attribués appelés respectivement MPV-Graph₁, MPV-Graph₂. Ces deux graphes permettent de représenter toutes les informations contenues dans les deux ontologies

(i.e., points de vue, concepts, rôles locaux et individus) et aussi les autres informations concernant l'ontologie multi-points de vue (i.e., rôles globaux et passerelles).

Les nœuds de chaque graphe représentent les trois types suivants : les concepts locaux, les concepts globaux et les individus. Les arcs qui existent dans $MPV\text{-Graph}_1$ représentent « les rôles locaux », et « les relations d'instanciation ». Par ailleurs, les arcs qui existent dans $MPV\text{-Graph}_2$ représentent en plus « les rôles globaux » et « les passerelles ». La création du $MPV\text{-Graph}$ est réalisée en trois étapes :

La transformation de TBox. Chaque axiome est décomposé en un ensemble de concepts, de rôles et de points de vue apparaissant dans cet axiome, chaque concept est représenté par un nœud, il existe un arc de type (rôle local) entre deux nœuds si leurs concepts sont dans le même axiome.

La transformation d'ABox. Chaque assertion est décomposée en un ensemble d'individus, de relations, et de points de vue apparaissant dans cette assertion, chaque individu est représenté par un nœud, il existe un arc de type « rôle local » entre deux nœuds si leurs individus sont dans la même assertion.

Par ailleurs, un arc de type « instanciation » est créé entre deux nœuds si l'un des deux nœuds est de type « concept » et l'autre est de type « individu ».

La transformation de LI¹. Pour chaque entrée de l'ensemble LI, on détermine l'ensemble de concepts, l'ensemble d'individus et l'ensemble de points de vue. en suite, un ou plusieurs arcs de types « rôles globaux ou passerelles » sont créés entre deux ou plusieurs nœuds si leurs concepts (individus) sont dans la même entrée.

Étiquetage du graphe. Dans cette étape nous nous inspirons de l'idée proposée dans [6] pour étiqueter les nœuds et les arcs des deux graphes. Chaque nœud du $MPV\text{-Graph}$ est étiqueté par un préfixe PV_i qui indique le nom du point de vue de ce nœud plus le nom d'un type (concept local, concept global ou individu), en ajoutant aux nœuds de type concept les étiquettes suivantes selon le cas :

- Un concept primitif est étiqueté par son nom.
- Un concept défini par une conjonction, une disjonction ou bien une négation est étiqueté par son nom et par l'opérateur logique utilisé (and, or, ou not).
- Un concept défini par un rôle est représenté par un arc qui porte le nom du rôle et pointe vers un autre concept.

Chaque arc, qui relie deux nœuds, est étiqueté par le type de la relation (rôle local, rôle global ou une passerelle) et le nom de cette relation.

Dans le cas d'un concept défini, l'arc est étiqueté soit par le quantificateur universel (\forall) soit par le quantificateur existentiel (\exists) ou bien par les symboles ' \leq , \geq ' et le nombre n et le rôle R dans le cas de la restriction.

¹ La partie LI de l'ontologie mono-point de vue est vide.

5.2 Processus d'Alignement

Le processus d'alignement se déroulera en quatre étapes, il consiste à : calculer la similarité entre tous les éléments de deux graphes, pour chaque couple de nœuds, on calculera d'abord la similarité terminologique, ensuite la similarité basée sur les attributs pertinents, puis calculer la similarité structurelle entre eux et utiliser les valeurs de ces trois similarités pour calculer la valeur de similarité globale entre ces couples de nœuds.

Similarité terminologique. Cette étape prend en entrée les deux graphes et calcule la similarité entre les noms de différents nœuds de MPV-Graph_s (i.e., concepts, individus et attributs).

Cette similarité est calculée soit en utilisant une fonction de similarité syntaxique comme la distance de Levenshtein qui est une distance normalisée, qui a prouvé sa robustesse face aux fautes d'orthographe, soit en utilisant une fonction de similarité lexicale en exploitant l'API de WordNet [7]. L'idée d'utiliser cette API est de résoudre les problèmes de conflit de terme (e.g., les termes humain et personne).

Le calcul de similarité entre les noms d'attributs nécessite une normalisation manuelle. Le but est de maximiser le nombre d'attributs communs entre les différents concepts. De ce fait, nous essayons de simplifier les chaînes de caractères représentant les attributs pour les ramener à des formats équivalents par un ensemble d'opérations (i.e., retirer les diacritiques, normaliser le nombre d'espaces, supprimer les noms de liaisons, extension des abréviations, etc.).

Le résultat de cette étape est une matrice représentant la similarité terminologique entre les nœuds des MPV-Graph_s qui est une somme pondérée des deux types de similarités précédente, dont les lignes représentent les nœuds du MPV-Graph₁ et les colonnes représentent les nœuds du MPV-Graph₂. Cette matrice est utilisée plus tard dans les autres modules de mise en correspondance.

Mesure basée sur les attributs pertinents. Dans le cas de multi-points de vue, la pondération d'attributs des concepts locaux est une tâche cruciale pour que la mesure de similarité soit pertinente et la plus précise possible. De ce fait, nous proposons d'utiliser une distance entre les attributs pondérés, pour les couples de nœuds de type (concept local, concept local). La mesure est calculée en utilisant une matrice d'attributs des deux concepts en question avec leurs poids et leurs valeurs de similarité terminologique déjà calculée. Dans cette matrice nous définissons un seuil pour classifier l'ensemble d'attributs communs entre les deux concepts. Nous nous inspirons pour cela de la mesure proposée dans [8] pour calculer la similarité entre deux concepts :

$$\text{Sim}_{\text{att}}(C1, C2) = \frac{\sum_{i,j}^n W_i W'_j * \text{Sim}_{\text{ss}}(\text{Att}_i, \text{Att}_j)}{\sum_{i,j}^m W_i W'_j} . \quad (1)$$

Où : n est le nombre d'attributs commun selon le seuil proposé, m=i*j est le nombre total des couples dans la matrice, Sim_{ss}(Att_i, Att_j) est la similarité supérieure au seuil défini, $\sum_{i,j}^n W_i W'_j$: la somme des poids des attributs commun,

$\sum_{i,j}^m W_i W'_j$: la somme des poids de tous les attributs (utilisée pour normaliser la mesure).

Similarité structurelle. Elle est calculée en exploitant la matrice de similarité terminologique ainsi que le voisinage et la hiérarchie des concepts en question dans les MPV-Graph_s. Dans notre modèle de représentation, nous trouvons trois types de voisinage qui dépendent de la position et de type des nœuds dans les MPV-Graph_s :

Racine globale. d'un nœud de type concept local ;

Racine locale. d'un nœud de type individus ;

Pour ces deux types, la similarité est calculée en extrayant le chemin racine possédant le nœud en question et la racine globale (locale), puis nous calculons la distance entre les deux nœuds en utilisant la mesure de Wu et Palmer [9] ;

Voisinage par point de vue. Dans cette section nous exploitons la structure de voisinage par point de vue et nous calculons la mesure de similarité sémantique entre deux ensembles de nœuds voisins, les ensembles sont organisés par point de vue. Pour calculer cette mesure, nous utilisons la mesure proposée dans [10] qui calcule la similarité entre deux ensembles. ENS_{pvi} , ENS_{pvj} représentent deux ensembles de nœuds. Les entités de chaque ensemble sont étiquetées par le même point de vue. La similarité sémantique entre deux nœuds appartenant à deux points de vue différents se calcule alors par la fonction suivante :

$$Sim_{vois}(N1, N2) = \frac{\sum_{(i,i') \in \text{paires}(ENS_{pvi}, ENS_{pvj})} SIM_{term}(i,i')}{\text{Max}(|ENS_{pvi}|, |ENS_{pvj}|)} . \quad (2)$$

Où (i, i') un couple de $ENS_{pvi} * ENS_{pvj}$, $SIM_{term}(i,i')$: la similarité terminologique déjà calculé.

La similarité structurelle Sim_{struc} est calculée en employant la technique de la somme pondérée des trois types précédents suivant le cas de type des nœuds à comparer.

Similarité globale. La valeur de similarité globale est calculée en employant la méthode de la somme pondérée des trois valeurs de similarité partielles (terminologique, basée sur les attributs pondérés et la similarité structurelle) :

$$Sim_{glob}(C1, C2) = \mu_{term} * Sim_{term} + \mu_{att} * Sim_{att} + \mu_{struc} * Sim_{struc} . \quad (3)$$

La similarité partielle Sim_{att} est égale à zéro si l'un des deux nœuds à comparer ou bien les deux sont de type individus (concept global).

Le résultat de cette phase est une matrice de similarité globale, qui contient tous les couples d'entités comparées avec une valeur de similarité compris entre [0, 1].

5.3 Post-Alignement

Filtrage des couples candidats à base de seuil. Cette étape est la première à effectuer dans la phase de post-alignement. Pour définir les nœuds les plus similaires nous déterminons une valeur de seuil située entre 0 et 1 dans ce cas. Après le calcul de similarité entre chaque couple de nœuds provenant de deux ontologies en entrée, on a une matrice contenant ces valeurs de similarité. Le filtrage consiste à éliminer les couples dont la valeur de similarité est inférieure à ce seuil. Nous obtenons donc une matrice des couples les plus similaires. Où nous pouvons trouver qu'un nœud du $MPV-Graph_1$ peut apparier à zéro, un ou plusieurs nœuds du $MPV-Graph_2$ (i.e., appariement multivoque).

Optimisation d'alignement. Afin de trouver les meilleures correspondances entre les deux ontologies, nous formalisons notre problème sous forme d'un problème de sélection de sous-ensemble et nous adaptons l'algorithme d'optimisation par colonies de fourmis (ACO) proposé dans [11] pour ce type de problème.

Bien que l'algorithme de colonie de fourmis est conçu au départ pour le problème du voyageur de commerce, il offre finalement beaucoup de souplesse, et il a été possible de l'adapter à un grand nombre de problèmes combinatoire (appariement de graphes).

Formalisation du problème. Dans notre contexte, le problème d'alignement d'une ontologie classique avec une ontologie multi-points de vue, est un problème d'appariements multivoques de deux graphes $MPV-Graph_s$, ce problème se ramène au problème de sélection de sous-ensemble [11], et nous pouvons le formaliser par le triplet $(S, S_{consistent}, f)$ tel que :

- S contient l'ensemble des couples appariant un nœud de $MPV-Graph_1$ à un nœud de $MPV-Graph_2$ « la matrice de similarité globale »;
- $S_{consistent} = P(S)$ contient tous les sous-ensembles de S et aussi les appariements multivoques ;
- f est définie par la fonction score, la fonction score est définie dans [10] par la formule :

$$score(S') = f(MPV-Graph_1 \cap_S MPV-Graph_2) - g(splits(S')). \quad (4)$$

Où $splits(S')$ est l'ensemble des nœuds qui sont appariés à plus d'un nœud. Donc, le résultat est un sous-ensemble consistant de couples de nœuds $S' \in S_{consistent}$, tel que la fonction score soit maximal.

Description de l'algorithme. A chaque cycle de l'algorithme, chacune des fourmis construit un sous-ensemble. En partant d'un sous-ensemble S' vide, les fourmis ajoutent à chaque itération un couple de nœuds de la matrice de similarité globale à S' choisi parmi l'ensemble des couples non encore sélectionnés. Le couple de nœuds à ajouter à S' est choisi selon une probabilité qui dépend des traces des phéromones et deux facteurs heuristiques, l'un vise à favoriser les couples qui ont la similarité la plus forte et l'autre vise à favoriser les couples qui font le plus augmenter la fonction score.

Une fois que chaque fourmi a construit son sous-ensemble, une procédure de recherche locale est lancée afin d'essayer d'améliorer la qualité du meilleur sous-

ensemble trouvé lors de ce cycle. Les traces de phéromone sont par la suite mises à jour en fonction de ce sous-ensemble amélioré.

Les fourmis arrêtent leur construction quand tous les couples de nœuds candidats font décroître le score de sous-ensemble ou quand les trois derniers ajouts n'ont pas permis d'accroître se score.

Construction d'une solution par une fourmi. Le code suivant décrit la procédure suivie par les fourmis pour construire un sous-ensemble.

Procédure construire sous-ensemble

Entrée : un SS-problème $(S, S_{\text{consistant}}, f)$ et une fonction heuristique associée : $S * P(S) \rightarrow \mathbb{R}^+$;

Une stratégie phéromonale et un facteur phéromonal τ , et deux facteurs heuristiques $\varphi 1, \varphi 2$.

Deux paramètres à valeurs numériques : α, β_1, β_2

Sortie : un sous-ensemble consistant d'objets $S' \in S_{\text{consistant}}$

Initialiser les traces de phéromone à τ_{max}

Répéter

Pour chaque fourmi k dans $1..nbFourmis$,
 construire une solution S_k comme suit :
 Choisir aléatoirement le premier nœud $o_i \in S$
 $S_k \leftarrow \{o_i\}$, **Candidat** $\leftarrow \{o_j \in S \mid S_k \cup \{o_j\} \in S_{\text{consistant}}\}$
Tant que **Candidats** $\neq \emptyset$ **faire**
 Choisir $o_i \in \text{Candidats}$ avec la probabilité

$$P_{o_i} = \frac{[\tau_{\text{facteur}}(o_i, S_k)]^\alpha \cdot [\varphi 1_{\text{facteur}}(o_i, S_k)]^{\beta_1} \cdot [\varphi 2_{\text{facteur}}(o_i, S_k)]^{\beta_2}}{\sum_{o_j \in \text{Candidat}} [\tau_{\text{facteur}}(o_j, S_k)]^\alpha \cdot [\varphi 1_{\text{facteur}}(o_j, S_k)]^{\beta_1} \cdot [\varphi 2_{\text{facteur}}(o_j, S_k)]^{\beta_2}}$$

$S_k \leftarrow S_k \cup \{o_i\}$

Enlever de **Candidats** chaque couple qui viole les contraintes.

Fin tant que

Fin pour

Mettre à jour les traces de phéromone en fonction de $\{S_1, \dots, S_{nbfourmis}\}$

Si une trace de phéromone est inférieure à τ_{min} **alors** la mettre à τ_{min}

Si une trace de phéromone est supérieure à τ_{max} **alors** la mettre à τ_{max}

Jusqu'à nombre maximal de cycles atteint ou solution trouvée.

Stratégie phéromonale. Le choix d'une stratégie phéromonale est un point clé lors du développement d'un algorithme à base de colonies de fourmis². Deux stratégies sont distinguées : « Vertex » mémorise l'expérience concernant chaque couple de nœuds individuellement et donc l'expérience de la colonie concernant l'intérêt d'apparier un nœud du premier graphe à un autre nœud du deuxième graphe. Et la stratégie « Edge » qui mémorise l'expérience concernant chaque couples de nœuds et donc l'expérience d'apparier deux arcs entre eux lors de la construction de nouveaux appariements.

Les résultats expérimentaux présentés dans [10] sur les problèmes d'appariement de graphes ont montré que la stratégie « Vertex » donne de meilleurs résultats que la stratégie « Edge ». ainsi la stratégie « Vertex » est plus rapide que la stratégie « Edge ».

La procédure de Recherche locale. Elle est appliquée sur le meilleur sous-ensemble trouvé lors de chaque cycle. Et permet d'offrir un bon compromis entre la quantité des solutions trouvées et son temps d'exécution. La procédure permet à chaque itération de sélectionner le couple de nœuds qui essaye d'améliorer la solution en explorant le voisinage. Les voisins d'un sous-ensemble sont les couples qui peuvent être obtenus en ajoutant ou en supprimant un couple de nœuds à ce sous-ensemble.

Résultat de l'optimisation. Le résultat de l'algorithme d'optimisation est un graphe de construction sur lequel les fourmis déposent de la phéromone. Il est construit à partir de deux MPV-Graphs. Les nœuds de ce graphe sont les combinaisons de deux nœuds, l'un du graphe de l'ontologie mono-point de vue et l'autre du graphe de l'ontologie multi-points de vue. Ces combinaisons représentent la meilleure solution que les fourmis peuvent sélectionner lors de la construction de leurs solutions. Cette solution représente les meilleures correspondances entre les deux ontologies.

Résultat de l'alignement. A partir du graphe de construction, nous déduisons une représentation graphique de l'alignement de deux MPV-Graphs, elle est composée des représentations des deux graphes et d'un ensemble de correspondances entre des nœuds issus respectivement des deux graphes.

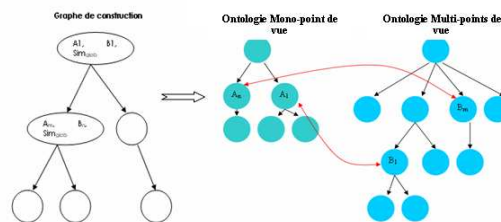


Fig. 2. Résultat de l'alignement

² Une stratégie phéromonale consiste à décider où les traces de phéromone doivent être déposées et comment elles doivent être exploitées et mises à jour.

6 Conclusion

Dans ce papier, nous avons adapté l'algorithme d'optimisation par les colonies de fourmis pour optimiser le résultat d'un alignement d'une ontologie multi-points de vue et une ontologie classique.

De ce fait, nous avons tout d'abord proposé une approche d'alignement d'une ontologie classique avec une ontologie multi-points de vue. L'approche proposée se compose de trois phases principales : (i) la phase de pré-alignement qui permet d'identifier les formats des ontologies en entrée de les adapter et de les transformer en des structures de graphes afin de résoudre le problème d'hétérogénéité de représentation. (ii) Le processus d'alignement qui combine les techniques et les méthodes d'appariement syntaxique, linguistiques, basé sur les attributs pertinents selon les différents points de vue et les techniques structurelles. (iii) La phase de post-alignement : dans cette phase, l'optimisation de l'appariement est effectuée par une technique de méta-heuristique de colonies de fourmis.

Dans l'approche proposée l'ordre des ontologies en entrée a une grande importance. Par ailleurs, l'hybridation des ACO/ recherche locale dans notre adaptation, nous permis de trouver un compromis entre le temps et la qualité d'alignement.

Références

1. Zghal, S., Kamoun, K., Ben Yahia S., Mephu-Nguifo, E., Slimani, Y. :EDOLA : Une Nouvelle Méthode d'Alignement d'Ontologies OWL-Lite - Dans CORIA'2007, pp.351-367 (2007)
2. Hemam, M., Boufaïda, Z. : Prise en Compte des Points de Vue dans la Construction des Ontologies en Logique de Description. 6^{ème} Colloque International sur l'Optimisation et les Systèmes d'Information, Algérie (2008)
3. Dorigo M., Caro G. D. :The Ant Colony Optimization Meta-Heuristic. : In New Ideas in Optimization, pp. 11–32. McGraw Hill, London, UK, (1999)
4. Hemam, M., Boufaïda, Z. : Raisonement par classification sur une ontologie multi-points de vue. Actes de la 3èmes Journée Francophone sur les Ontologies (JFO'09), Poitiers (France): ACM Edition. (2009)
5. Hemam, M., Boufaïda, Z. : Représentation d'Ontologies Multi-points de Vue : une Approche Basée sur la Logique de Description. Conférence en Ingénierie des Connaissances, 25-29 mai, Hammamet, Tunisie, (2009)
6. Falquet, G., Mottaz, J.C.L. : Navigation Hypertexte dans une Ontologie Multi-Points de Vue. NîmesTIC'2001, Nîmes, France, (2001)
7. Miller, A.M. :WordNet : A Lexical Database for English.Communications of the ACM, 38(11) :39-41, (1995)
8. Gilles, B. : La similarité: une notion symbolique/numérique. Apprentissage symbolique-numérique (tome 2). Eds Moulet, Brito. Editions CEPADUES. pp. 169-201. (2000)
9. Wu, Z., Palmer, M.: Verb Semantics and Lexical Selection'', Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics, pp 133-138 (1994)
10. Touzani, M.: Alignement des Ontologies OWL-Lite. Master's thesis, University of Montreal, (2005)
11. Solnon, C. : Contributions à la Résolution Pratique de Problèmes Combinatoires – des Fourmis et des Graphes-. Mémoire pour l'Obtention de l'Habilitation à Diriger des Recherches, Université Claude Bernard Lyon I, (2005).