Notions de dépendances pour les règles existentielles

Jean-François Baget, Marie-Laure Mugnier, and Michaël Thomazo

Université Montpellier 2
LIRMM
INRIA - GraphIK

Résumé : Nous nous intéressons à un problème de décision consistant à déterminer s'il existe une réponse à une requête conjonctive dans une base de connaissances composée d'une base de données et d'une ontologie décrite par un ensemble de règles contenant des variables existentielles en conclusion, équivalentes aux Tuple-Generating Dependencies (TGD) en bases de données. Ce problème indécidable impacte de nombreux domaines (logiques de description, bases de données, web sémantique...). Dans Baget (2004) et Baget *et al.* (2009) il a été proposé une notion de dépendance entre règles pour définir de nouveaux cas décidables. Cependant, si cette notion de dépendance est optimale à un instant donné du raisonnement, elle ne prend pas en compte l'historique de celui-ci. Dans cet article, nous proposons un raffinement de cette notion, la k-dépendance, qui tient compte d'un historique de taille bornée.
Mots-clés : Règles existentielles, TGD, Datalog+/-, Dépendance

1 Introduction

Un thème de recherche à la croisée des bases de données et de la représentation des connaissances reçoit actuellement une attention croissante : il s'agit de l'interrogation de bases de connaissances munies d'ontologies expressives par des requêtes conjonctives. Dans ce domaine, le langage Datalog+/- (Calì *et al.* (2009)) code l'ontologie par des contraintes, des EGDs (Equality Generating Dependencies) et des TGDs (Tuple Generating Dependencies). Ce sont ces derniers objets que nous appelons règles existentielles, et que nous étudions sous leur forme logique, qui sont des formules closes du type $\forall x_1 \dots x_p(H \rightarrow (\exists z_1 \dots z_q C))$, où H et C sont des conjonctions d'atomes. Le problème fondamental est celui de la conséquence logique : étant donnée une base de connaissances $K = (F, \mathcal{R})$ et Q, où F et Q sont deux conjonctions d'atomes existentiellement closes, appelés respectivement le fait et la requête, et $\mathcal{R}$ est un ensemble de règles, est-il vrai que $K \models Q$?

Les connaissances de nature ontologique sont traditionnellement représentées par des logiques de description. Cependant, la nécessité de prendre en compte des requêtes conjonctives (et non plus seulement des requêtes réduites à un atome) a profondément modifié le paysage de ces logiques. De nouvelles logiques, moins expressives,

sont étudiées afin de répondre à ce problème avec un meilleur compromis expressivité/complexité.

L'approche présentée ici provient d'un courant parallèle de recherche, où certaines contraintes inhérentes aux logiques de description (arité des prédicats inférieure ou égale à deux, pas de cycle dans les règles) n'ont plus lieu d'être. Cependant, sans contraintes sur les règles, la réponse à des requêtes conjonctives est un problème indécidable. Il est donc important de définir des cas particuliers pour lesquels ce problème devient décidable. La plupart des cas décidables connus reposent sur un processus de marche avant, appelé "chase" en base de données (Maier *et al.* (1979)). A l'aide des règles, on opère une saturation de la base de faits, c'est-à-dire que l'on ajoute des informations aux faits en appliquant les règles, tant que ces informations sont nouvelles. Lorsque ce processus s'arrête (Fagin *et al.* (2005), Baget & Mugnier (2002), Baget (2004), Spezzano & Greco (2010)), la conséquence logique est décidable. Si ce processus ne s'arrête pas, mais génère des faits dont la structure est quasi-arborescente (Calì *et al.* (2008), Calì *et al.* (2009), Baget *et al.* (2010)), le problème est également décidable. Enfin, une autre condition de décidabilité repose sur la finitude d'un mécanisme de marche arrière. Malheureusement, décider si un ensemble de règles vérifie l'un des trois critères sus-cités est un problème indécidable. Une zoologie de classes de règles décidables ayant ces propriétés a donc été proposée.

Cependant, ces classes concrètes ne se comportent pas bien les unes par rapport aux autres. Si $\mathcal{R}_1$ et $\mathcal{R}_2$ sont des classes décidables, il est généralement faux que $\mathcal{R}_1 \cup \mathcal{R}_2$ soit décidable. De manière à pouvoir décomposer un ensemble de règles en sous-ensembles plus simples étudiables de manière (presque) indépendante, une notion de dépendance a été introduite dans Baget (2004) et Baget *et al.* (2009). Intuitivement, une règle R_2 dépend d'une règle R_1 s'il existe un fait F tel qu'une application de R_1 sur F peut permettre une application de R_2 qui n'était pas possible sur F uniquement. Toutefois, si cette notion de dépendance est optimale à un instant donné du raisonnement, elle ne prend pas en compte l'historique de celui-ci. Dans cet article, nous proposons un raffinement de cette notion, la k-dépendance, qui tient compte d'un historique de taille bornée par k.

En section 2, nous formalisons le problème de conséquence logique pour les règles existentielles, et introduisons les principales définitions. En section 3, nous rappelons la définition de dépendance et certaines de ses propriétés. Nous étudions les k-dépendances en section 4. Nous concluons et présentons quelques pistes à explorer en section 5.

2 Préliminaires

On considère un langage logique du premier ordre avec constantes mais sans autre symbole de fonction. Un atome est donc de la forme $p(t_1, \ldots, t_k)$ où p est un prédicat d'arité k et t_i sont des variables ou des constantes. Un fait F est une conjonction d'atomes fermée existentiellement, que l'on verra comme un ensemble d'atomes. Les variables d'une conjonction d'atomes seront notées $\mathrm{var}(F)$ et ses termes $\mathrm{term}(F)$. Une requête conjonctive a la même forme qu'un fait, et l'on identifiera donc les deux notions. Une règle existentielle (et simplement règle par la suite) est une formule close de la forme $\forall x_1 \ldots \forall x_p (H \rightarrow (\exists z_1 \ldots \exists z_q C))$ où H et C sont deux conjonctions non

vides d'atomes appelées respectivement l'hypothèse et la conclusion de la règle R. Par la suite, les quantificateurs seront implicites. Une base de connaissances est un couple composé d'un ensemble de faits et d'un ensemble de règles. L'union de deux faits étant un fait, on considérera l'ensemble des faits comme un seul fait. On note donc $(F, \mathcal{R})$ une base de connaissances, où F est un fait et $\mathcal{R}$ un ensemble de règles. Nous nous intéressons au problème suivant : "étant donnés une base de connaissances $K = (F, \mathcal{R})$ et un fait (aussi appelé requête conjonctive) Q, est-il vrai que $F, \mathcal{R} \models Q$, i.e. que Q est conséquence logique de la conjonction des formules composant F et $\mathcal{R}$?"

Substitution et homomorphisme. Etant donné un ensemble de variables V et un ensemble de termes T, une *substitution* de V par T est une application de V dans T. Soit $\sigma : V \to T$ une substitution, et F un fait. $\sigma(F)$ est le fait obtenu à partir de F en remplaçant chaque occurrence de $x \in V \cap \text{term}(F)$ par $\sigma(x)$. Une substitution est dite *sûre* s'il s'agit d'une bijection entre V et un ensemble de variables *fraîches* (c'est-à-dire n'apparaissant nulle part dans les formules du raisonnement). Etant donnés deux faits F et Q, un homomorphisme π de Q vers F est une substitution de $\text{var}(Q)$ par $\text{term}(F)$ telle que $\pi(Q) \subseteq F$. L'homomorphisme est correct et complet par rapport à la conséquence logique dans le fragment des faits : étant donné deux faits F et Q, $F \models Q$ ssi il y a un homomorphisme de Q dans F.

Pour la facilité de lecture, nous rappelons quelques définitions provenant de Baget *et al.* (2011a).

Définition 1 (Application d'une règle)
Soit F un fait et $R = (H, C)$ une règle. R est dite applicable *à F s'il existe un homomorphisme π de H dans F. Dans ce cas, l'application de R à F selon π produit un fait $\alpha(F, R, \pi) = F \cup \pi^s(C)$, où $\pi^s = \pi$ sur le domaine de π, et est une substitution sûre des autres variables. Les* atomes créés *par l'application de R à F selon π sont les éléments de $\pi^s(C) \backslash F$. Un atome créé par R est dit d'*origine R.

Exemple 1
Soit $F = \{r(a, b), r(b, c)\}$ un fait, $R_1 : r(x, y) \wedge r(y, z) \to r(x, z)$ et $R_2 : r(x, y) \to r(y, z)$ deux règles. R_1 s'applique par $\pi(x) = a, \pi(y) = b, \pi(z) = c$. Le fait obtenu est alors $F' = \{r(a, b), r(b, c), r(a, c)\}$. R_2 s'applique à F' de trois manières différentes : par $\pi_1(x) = a, \pi_1(y) = b$, $\pi_2(x) = b, \pi_2(y) = c$ et $\pi_3(x) = a, \pi_3(y) = c$. Si l'on applique R_1 à F' par π_3, on obtient $F'' = \{r(a, b), r(b, c), r(a, c), r(c, x_1)\}, x_1$ étant existentiellement quantifiée.

Nous considérons la saturation d'un fait, qui est un mécanisme de marche avant en largeur. En notant F_0 le fait initial, F_i étant le fait à l'étape i, on obtient F_i à partir de F_{i-1} en calculant tous les homomorphismes d'une hypothèse de règle dans F_{i-1} qui n'ont pas déjà été appliqués, puis en les appliquant.

Définition 2 (Saturé d'ordre k, ordre et origine d'un atome)
Soit F un fait et $\mathcal{R}$ un ensemble de règles existentielles. $\Pi(\mathcal{R}, F)$ désigne l'ensemble des homomorphismes de l'hypothèse d'une règle de $\mathcal{R}$ dans F : $\Pi(\mathcal{R}, F) = \{(R, \pi) | R = (H, C)$ et π est un homomorphisme de H dans $F\}$. La saturation *de F par $\mathcal{R}$ à l'ordre k est notée $F^k = \alpha_k(F, \mathcal{R})$ et est définie récursivement comme suit :*

- $\alpha_0(F, \mathcal{R}) = F$
- $\forall i > 0, F^i = F^{i-1} \cup_{(R=(H,C),\pi)\in\Pi(F^{i-1},\mathcal{R})\setminus\Pi(F^{i-2},\mathcal{R})} \pi^s(C)$, *où par convention,* $F^{-1} = \emptyset$.

*Un atome est dit d'*ordre k *(par rapport à* $F, \mathcal{R}$*) s'il est créé par une application de règle appartenant à* $F^k \setminus F^{k-1}$.

On note $\alpha_\infty(F, \mathcal{R}) = \cup_{k\in\mathbb{N}}\alpha_k(F, \mathcal{R})$.

Théorème 1
Soit F *et* Q *deux faits et* $\mathcal{R}$ *un ensemble de règles.* $F, \mathcal{R} \models Q$ *si et seulement s'il existe* $k \in \mathbb{N}$ *et un homomorphisme de* Q *dans* $\alpha_k(F, \mathcal{R})$.

Définition 3 (Ensemble à expansion finie)
Un ensemble de règles $\mathcal{R}$ *est dit à expansion finie si et seulement s'il existe un entier* k *tel que* $F^k = \alpha_k(F, \mathcal{R}) \equiv \alpha_{k+1}(F, \mathcal{R})$.

Avec un ensemble de règles à expansion finie, le problème de conséquence logique devient décidable. L'algorithme 1 est en effet assuré de terminer avec un ensemble $\mathcal{R}$ à expansion finie.

Données: $\mathcal{R}$ un ensemble de règles, F un fait et Q une requête
Résultat: Vrai si Q est conséquence logique de $F, \mathcal{R}$, faux ou ne s'arrête pas sinon
$F_c = F$;
tant que $\alpha_(F_c, \mathcal{R}) \not\equiv F_c$ **faire**
  $F_c = \alpha(F_c, \mathcal{R})$;
  si $F_c \models Q$;
  alors
    retourner *Vrai*;
  fin
fin
retourner *Faux*;

Algorithm 1: L'algorithme générique de marche avant

En effet, il suffit d'appliquer des étapes de saturation de F, c'est-à-dire calculer $\alpha_k(F, \mathcal{R})$ jusqu'à ce que $\alpha_k(F, \mathcal{R}) \equiv \alpha_{k+1}(F, \mathcal{R})$. Un tel k existe par hypothèse : il suffit alors de tester la conséquence logique entre faits.

3 Rappels sur la notion de dépendance

La recherche de cas décidables pour la conséquence logique, c'est-à-dire de contraintes sur les ensembles de règles assurant la décidabilité a mené à la définition de nombreuses classes de règles. Malheureusement, ces différents critères sont généralement incompatibles, dans la mesure où si $\mathcal{R}_a$ et $\mathcal{R}_b$ appartiennent à des classes décidables, il est généralement faux que $\mathcal{R}_a \cup \mathcal{R}_b$ appartienne également à une classe décidable (Baget *et al.* (2010)). Cette propriété empêche la décomposition d'un ensemble de règles en sous-ensembles plus simples, que l'on souhaiterait étudier de manière séparée.

Une réponse partielle à ce problème a été proposée dans Baget (2004) et Baget *et al.* (2009) : la notion de dépendance. Intuitivement, une règle R_j dépend d'une règle R_i si une application de R_i est susceptible de déclencher une nouvelle application de R_j. Cette notion est formalisée ci-dessous. Les définitions et propriétés de cette section sont tirées de Baget *et al.* (2009).

3.1 Définitions et propriétés

Définition 4 (Dépendance de R_j par rapport à R_i)
Soit R_i, R_j deux règles. $R_j = (H_j, C_j)$ dépend de $R_i = (H_i, C_i)$ s'il existe un fait F, un homomorphisme $\pi : H_i \to F$ et un homomorphisme $\pi' : H_j \to \alpha(F, R_i, \pi)$ tel que $\pi'(H_j) \not\subseteq F$.

Exemple 1 (suite)
L'application π_3 de R_2 n'était pas possible dans F, mais a été rendue possible dans F' grâce à une application de R_1. R_2 dépend donc de R_1. Dans F'' obtenu en appliquant R_2, il y a une nouvelle application de R_1 rendue possible. R_1 dépend donc de R_2.

Cette notion abstraite de dépendance peut se calculer effectivement par un test d'unification entre H_j et C_i (cette notion étant relativement complexe, nous invitons le lecteur intéressé à se reporter à Baget *et al.* (2009)). Ce test est NP-complet, mais cette étape peut être considérée comme étant faite hors-ligne dans le cas de réponses à des requêtes conjonctives. Les relations de dépendance sont alors compilées dans le *graphe de dépendance* des règles.

Définition 5 (Graphe de dépendance)
Soit $\mathcal{R}$ un ensemble de règles. Le graphe de dépendance de règles de $\mathcal{R}$ est le graphe ayant pour sommets les éléments de $\mathcal{R}$, et qui possède un arc de R_i vers R_j ssi R_j dépend de R_i. Il est noté $GRD(\mathcal{R})$ (de l'anglais Graph of Rule Dependencies*).*

Avoir un graphe de dépendance sans circuit est une condition suffisante pour être à expansion finie :

Propriété 1 (GRD sans circuit)
Soit $\mathcal{R}$ un ensemble de règles. Si $GRD(\mathcal{R})$ est sans circuit, alors $\mathcal{R}$ est à expansion finie.

La propriété clé suivante permet (potentiellement) de *décomposer* un ensemble de règles en sous-ensembles plus petits, qui peuvent être étudiés séparément.

Propriété 2 (Décomposition en composantes fortement connexes)
Soit $\mathcal{R}$ un ensemble de règles. Si toutes les composantes fortement connexes de $GRD(\mathcal{R})$ sont à expansion finie, alors $\mathcal{R}$ est à expansion finie.

3.2 Limites de cette notion de dépendance

Nous présentons maintenant plusieurs limites de cette notion de dépendance. Pour des raisons historiques, il est parfois demandé que les règles aient des conclusions atomiques. C'était en effet le cas dans le Datalog [1] de base, où cela peut être supposé sans perte de généralité, car toute règle Datalog avec plusieurs atomes en conclusion peut être décomposée en un ensemble de règles avec même hypothèse et un seul atome en conclusion. Dans le cas des règles existentielles, la transformation est également possible, mais fait appel à un prédicat auxiliaire par règle, comme présenté ci-dessous.

Définition 6 (Décomposition atomique $\mathcal{T}_a$)
Soit $R = (H, C)$ une règle possédant k atomes en conclusion. $\mathcal{T}_a(R)$ est un ensemble de $k + 1$ règles. Une première règle a pour hypothèse H et pour conclusion $p_R(x_1, \ldots, x_n)$, où var$(C) = \{x_i\}_i$, et p_R est un prédicat frais. Les k autres règles ont pour hypothèse $p_R(x_1, \ldots, x_n)$, et pour conclusion l'un des atomes qui est en conclusion de R (une règle par atome). On définit enfin $\mathcal{T}_a(\mathcal{R}) = \cup_{R \in \mathcal{R}} \mathcal{T}_a(R)$.

Exemple 2
Les règles ci-dessous montrent la décomposition de $\mathcal{R} = \{R_1, R_2\}$ en $\mathcal{T}_a(\mathcal{R}) = \{R_1^0, R_1^1, R_1^2, R_1^3, R_2^0, R_2^1\}$:

$R_1 : p(x) \rightarrow r(x,y) \wedge r(y,z) \wedge r(z,x)$	$R_2 : r(x,y) \wedge r(y,x) \rightarrow p(y)$
$R_1^0 : p(x) \rightarrow p_{R_1}(x,y,z)$	$R_2^0 : r(x,y) \wedge r(y,x) \rightarrow p_{R_2}(x,y)$
$R_1^1 : p_{R_1}(x,y,z) \rightarrow r(x,y)$	$R_2^1 : p_{R_2}(x,y) \rightarrow p(y)$
$R_1^2 : p_{R_1}(x,y,z) \rightarrow r(y,z)$	
$R_1^3 : p_{R_1}(x,y,z) \rightarrow r(z,x)$	

Propriété 3
Soit $K = (F, \mathcal{R})$ une base de connaissances et Q une requête conjonctive. $\mathcal{T}_a(\mathcal{R})$ est tel que $\mathcal{R}, F \models Q$ ssi $\mathcal{T}_a(\mathcal{R}), F \models Q$.

Cependant, les graphes de dépendance de ces deux ensembles de règles peuvent avoir des comportements différents : $GRD(\mathcal{R})$ peut être sans circuit alors que $GRD(\mathcal{T}_a(\mathcal{R}))$ en possède, comme le montre l'exemple suivant.

Exemple 2 (suite)
En reprenant l'exemple précédent, les graphes de dépendance de $\mathcal{R}$ et $\mathcal{T}_a(\mathcal{R})$ sont visibles dans la Figure 1. Du fait de la dépendance de R_2^0 envers R_1^1, R_1^2 et R_1^3, on a maintenant un cycle dans le GRD. En découpant la conclusion de R_1, on a ainsi introduit de nouvelles dépendances.

Plus généralement, dans le cas où R_c dépend de R_b qui dépend elle-même de R_a, il n'est pas certain qu'une application de R_c puisse être déclenchée par une application de R_b qui a été déclenchée par une application de R_a. Ce cas de figure est illustré par l'exemple 2, après décomposition atomique (en considérant par exemple $R_a = R_1^0, R_b = R_1^1, R_c = R_2^0$). Alors que la dépendance calcule très exactement la

1. On a alors des règles sans variable existentielle en conclusion, i.e., pour toute $R = (H, C)$, var$(C) \subseteq$ var(H).

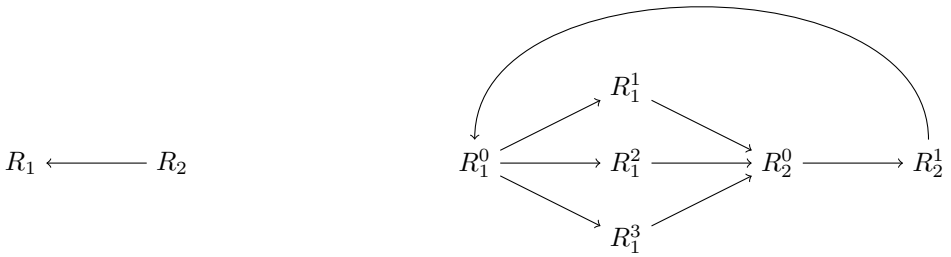

FIGURE 1 – Les graphes de dépendances de $\mathcal{R}$ et de $\mathcal{T}_a(\mathcal{R})$

possibilité pour une règle de déclencher l'application d'une autre sur un fait dont on ne sait rien, l'utilisation de la dépendance à une étape $k > 1$ de saturation n'est plus optimale, car nous avons une information supplémentaire sur le fait : il a été produit par $k - 1$ étapes de saturation. C'est cette nouvelle notion de dépendance que nous nous proposons d'étudier dans la suite de cet article.

4 Une nouvelle famille de dépendances

Pour cela, nous proposons dans cette section un raffinement de la notion de dépendance : la dépendance à l'ordre k ($k \geq 1$), encore appelée k-dépendance. L'intuition derrière cette notion est d'exclure des cas de dépendance celles qui font "fortement" appel au fait initial, au sens où elles ne peuvent avoir lieu "tard", c'est-à-dire sur un fait qui est un saturé à l'ordre k d'un autre fait. Il devient donc nécessaire (même si la notation n'en fait pas explicitement mention) de prendre en compte l'ensemble $\mathcal{R}$ tout entier, et plus seulement deux règles pour décider de la dépendance.

Définition 7 (k-dépendance)
Soit $\mathcal{R}$ un ensemble de règles, $k \geq 1$, et $R_i, R_j \in \mathcal{R}$. On dit que $R_j = (H_j, C_j)$ dépend à l'ordre k (ou k-dépend) de $R_i = (H_i, C_i)$ s'il existe F et un homomorphisme $\pi : H_j \to \alpha_k(F, \mathcal{R})$ tel qu'un atome de H_j ait pour image par π un atome d'ordre k et d'origine R_i. Un tel fait F est appelé un k-témoin de la dépendance

Comme pour la dépendance usuelle, nous synthétisons les dépendances au sein d'un graphe de dépendance, le $GRD_k(\mathcal{R})$:

Définition 8 ($GRD_k(\mathcal{R})$)
Soit $\mathcal{R}$ un ensemble de règles. Le graphe de dépendance à l'ordre k de $\mathcal{R}$ a pour sommets les éléments de $\mathcal{R}$, et un arc de R_i vers R_j ssi R_j dépend à l'ordre k de R_i. Il est noté $GRD_k(\mathcal{R})$.

Nous commençons par démontrer que l'objet que nous définissons est bien "plus petit" que le graphe usuel, ce qui était notre but initial :

Propriété 4
Soit $\mathcal{R}$ un ensemble de règles. On a, pour $1 \leq k \leq k'$, les inclusions suivantes :

$$GRD_{k'}(\mathcal{R}) \subseteq GRD_k(\mathcal{R}) \subseteq GRD_1(\mathcal{R}) = GRD(\mathcal{R})$$

Preuve : Soit $1 \leq k \leq k'$. Comme le saturé à l'ordre k' de F est égal au saturé à l'ordre k de $\alpha_{k'-k}(F, \mathcal{R})$, nous en déduisons que $GRD_{k'}(\mathcal{R}) \subseteq GRD_k(\mathcal{R})$.

Dans quels cas le $GRD_k(\mathcal{R})$ est-il strictement plus petit que le $GRD(\mathcal{R})$? Pour en obtenir une intuition, il est intéressant de considérer l'exemple suivant : $p(x) \wedge r(x, y) \rightarrow r(y, z)$. Cette règle dépend d'elle même, mais ne 2-dépend pas d'elle même (car elle ne produit pas d'atome de prédicat p). Cet exemple est généralisable, en considérant des chaînes d'atomes de prédicat r de longueur k en hypothèse, la conclusion apportant un $k+1^{\text{ème}}$ élément à la chaîne. Cette règle serait k dépendante d'elle-même, mais pas $k + 1$-dépendante.

Nous montrons maintenant que les propriétés du $GRD(\mathcal{R})$ concernant la marche avant peuvent être généralisées au $GRD_k(\mathcal{R})$.

Propriété 5 (Acyclicité de $GRD_k(\mathcal{R})$)
Soit $\mathcal{R}$ un ensemble de règles. S'il existe k tel que $GRD_k(\mathcal{R})$ soit sans circuit, $\mathcal{R}$ est à expansion finie.

Preuve : Soit F un fait quelconque. Soit F^k le saturé à l'ordre k de F. Soit $\mathcal{R}_1$ l'ensemble des règles qui s'appliquent de façon nouvelle à F^k. Les règles susceptibles de s'appliquer à F^{k+l} sont celles avec un chemin de longueur l partant d'une règle de $\mathcal{R}_1$. Comme $GRD_k(\mathcal{R})$ est sans circuit, il existe m tel que m soit la longueur d'un plus long chemin dans $GRD_k(\mathcal{R})$. On a alors $F_{k+m} = F_{k+m+1}$, et $\mathcal{R}$ est donc à expansion finie.

Propriété 6 (Décomposition en c.f.c.)
Soit $\mathcal{R}$ un ensemble de règles. Si toutes les composantes fortement connexes de $GRD_k(\mathcal{R})$ sont à expansion finie, alors $\mathcal{R}$ est à expansion finie.

Preuve : Similaire à celle de la propriété 5.

Il existe également des propriétés du GRD_k qui ne sont pas directement le pendant d'une propriété du GRD. En effet, la notion de k-dépendance possède une certaine résistance aux transformations syntaxiques, par exemple à la décomposition atomique présentée dans la section précédente :

Propriété 7 (Résistance à la décomposition atomique)
Soit $\mathcal{R}$ un ensemble de règles. Si $GRD_k(\mathcal{R})$ est sans circuit, alors $GRD_{2k}(\mathcal{T}_a(\mathcal{R}))$ est sans circuit.

Preuve : Voir Baget *et al.* (2011b).

L'intérêt de la notion de k-dépendance ne se limite pas à la définition de classes décidables plus générales qu'auparavant. Nous présentons ici de possibles optimisations pratiques que peut apporter la connaissance du $GRD_k(\mathcal{R})$, avec la présentation de l'algorithme 2. En effet, par définition de la k-dépendance, une fois effectuées k étapes

de saturation, il n'est pas nécessaire de tester les applications potentielles de toutes les règles, mais uniquement de celles étant successeurs dans le $GRD_k(\mathcal{R})$ d'une règle appliquée au rang précédent.

Données: F un fait, $\mathcal{R}$ une base de connaissances, Q une requête, $GRD_k(\mathcal{R})$
Résultat: Vrai si Q est conséquence logique de $F, \mathcal{R}$, faux ou ne s'arrête pas sinon
$F_b = \alpha_{k-1}(F, \mathcal{R})$;
$F_c = \alpha(F_b, \mathcal{R})$;
tant que $F_b \not\equiv F_c$ **faire**
  $F_b = F_c$;
  $\mathcal{R}_b =$ l'ensemble des règles ayant été appliquées à F_b;
  $\mathcal{R}_c =$ l'ensemble des successeurs des règles de $\mathcal{R}_b$ dans $GRD_k(\mathcal{R})$;
  $F_c = \alpha(F_c, \mathcal{R}_c)$;
  si $F_c \models Q$ **alors**
    retourner *Vrai*;
  fin
fin
retourner *Faux;*

Algorithm 2: L'algorithme de marche avant avec un graphe de k-dépendance précompilé.

Enfin, nous mentionnons un résultat de complexité. Soit la famille de problèmes de décision DÉDUCTION$_k$, paramétrée par $k \geq 1$:

- ENTRÉE : un ensemble de règles $\mathcal{R}$, $R_i, R_j \in \mathcal{R}$
- QUESTION : R_j k-dépend de R_i ?

On peut montrer (Baget *et al.* (2011b)) que pour tout $k \geq 1$, le problème DÉDUCTION$_k$ est NP-complet.

5 Conclusion et travaux ultérieurs

Nous nous sommes intéressés au problème de l'interrogation de bases de connaissances munies de règles expressives à l'aide de requêtes conjonctives. Pour ce problème indécidable dans le cas général, il a été proposé dans les nombreuses publications récentes, un grand nombre de classes décidables de règles. Cependant, ces différentes classes de règles sont incompatibles, au sens où l'union de deux ensembles de règles décidables n'est plus nécessairement décidable. Le seul outil à la connaissance des auteurs permettant de combiner des résultats de décidabilité est le graphe de dépendance des règles, qui souffre cependant de certaines faiblesses. Nous avons proposé un raffinement de la notion de dépendance, à savoir la k-dépendance, qui possède de meilleures propriétés. Il reste cependant des pistes à explorer. Premièrement, l'algorithme (non présenté ici, voir Thomazo (2010),Baget *et al.* (2011b)) pour caractériser la k-dépendance entre deux règles repose sur de la force brute. Deuxièmement, il serait bon d'adapter les résultats obtenus également dans le cas de la marche arrière. Pour ces deux pistes, étudier les liens entre la k-dépendance et les unificateurs (qui servent pour la marche

arrière et pour caractériser la dépendance usuelle) est la démarche envisagée. Enfin, l'étude de la k-dépendance invite à considérer des réécritures de règles, de manière à obtenir des ensembles de règles équivalentes ayant de bonnes propriétés syntaxiques.

Références

BAGET J.-F. (2004). Improving the forward chaining algorithm for conceptual graphs rules. In *KR*, p. 407–414.

BAGET J.-F., LECLÈRE M. & MUGNIER M.-L. (2010). Walking the decidability line for rules with existential variables. In *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning, Toronto, Canada* : AAAI Press.

BAGET J.-F., LECLÈRE M., MUGNIER M.-L. & SALVAT E. (2009). Extending decidable cases for rules with existential variables. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI 2009), Pasadena, California, USA,*, p. 677–682.

BAGET J.-F., LECLÈRE M., MUGNIER M.-L. & ÉRIC SALVAT (2011a). On rules with existential variables : Walking the decidability line. *To appear in Artificial Intelligence Journal*.

BAGET J.-F. & MUGNIER M.-L. (2002). The Complexity of Rules and Constraints. *J. Artif. Intell. Res. (JAIR)*, **16**, 425–465.

BAGET J.-F., MUGNIER M.-L. & THOMAZO M. (2011b). *Towards Farsighted Dependencies for Existential Rules*. LIRMM Research Report 11016, LIRMM.

CALÌ A., GOTTLOB G. & KIFER M. (2008). Taming the infinite chase : Query answering under expressive relational constraints. In *Proceedings of the Eleventh International Conference on the Principles of Knowledge Representation and Reasoning (KR 2008), Sydney, Australia*, p. 70–80.

CALÌ A., GOTTLOB G. & LUKASIEWICZ T. (2009). A general datalog-based framework for tractable query answering over ontologies. In *Proceedings of the 28th symposium on Principles of database systems (PODS'09)*, p. 77–86, New York, NY, USA : ACM.

FAGIN R., KOLAITIS P. G., MILLER R. J. & POPA L. (2005). Data exchange : semantics and query answering. *Theor. Comput. Sci.*, **336**(1), 89–124.

MAIER D., MENDELZON A. O. & SAGIV Y. (1979). Testing implications of data dependencies. *ACM Trans. Database Syst.*, **4**, 455–469.

SPEZZANO F. & GRECO S. (2010). Chase termination : A constraints rewriting approach. *PVLDB*, **3**(1), 93–104.

THOMAZO M. (2010). Interrogation de bases de connaissances avec des règles expressives : décidabilité et dépendance entre règles. *Rapport de M2 au MPRI*.