Compte Rendu Semaine 4

Andersson Calle Viera : 3521501

Exercice 2:
1)Les règles ci-dessus sont les règles de la subsomption :
- subsS1(C, C) est toujours vrai (C est subsumé par C)
- subsS1(C, D) :- subs(C, D), C\==D dans le cas ou C est différent de D
- subsS1(C, D) :- subs(C, E), subsS1(E, D) permet d'exprimer une relation d'ordre sur la subsomption.

2)
```
?- subsS1(canari, animal).
true .

?- subsS1(chat, etreVivant).
true .
```

3) la requête ne termine pas

4)
subsS(canari, animal).
subsS(chat, etreVivant).
subsS(chien, canide).
subsS(chien, chien).

5) « il existe » devant un rôle atomique nous donne un concept atomique, ainsi la requête subsS(souris,some(mange)). Fonctionne

6)
subsS(chat, humain).
subsS(chien, souris).

7)
subsS(chat, X).
X devrait prendre comme valeur chat, félin, mammifère, animal, etreVivant, some(mange).
C'est bien le cas.
subsS(X, mammifere).
X devrait prendre comme valeur mammifere, souris, félin, canidé, lion, chat, chien.
C'est le cas, mais lion apparait deux fois :
il apparait dans deux règles de notre base de connaissance qui participent à la résolution de cette requête.

8) fait

9) La requête renvoie faux avant ajout des règles car il n'y encore pas de déduction de équiv à subs

10) On a en effet intérêt à dériver de subs, car on réalise « plus tôt » des déductions sur subs, et ainsi les subsS résultant des subs sont plus nombreux.
Exercice 3:

1)
a) subsS(chihuahua,and(mammifere,some(aMaitre))). TRUE
b) subsS(and(chien,some(aMaitre)),pet). TRUE
c) subsS(chihuahua,and(pet,chien)). TRUE

2)
a) Cas où on subsume a une intersection: on regarde si les 2 subsomptions séparées sont vraies. subsS(chihuahua,and(mammifere,some(aMaitre))). ne fonctionne pas sans celle ci
b) permet de descendre dans les subsomptions avec intersections si elles existent
c) si C est une variable définie, il suffit juste de faire le subsS d'origine. subsS(and(X,X),pet). ne fonctionne pas sans celle ci
d) On vérifie si le premier membre de l'intersection subsume
e) Ou bien si le second membre de l'intersection subsume
f) Permet de remonter les subsomptions
g) inverse l'intersection pour faire le parcours de la ligne precedente sur le second membre de la subsomptions

3) oui

Exercice 4:

1) Règles:
subsS(C,all(R,D),L):- subs(E1,D),E=all(R,E1),not(member(E, L)),subsS(C,E,[E|L]).

subsS(all(R,C),D,L) :- subs(C,E1),E=all(R,E1),not(member(E, L)),subsS(E,D,[E|L]).

2)
a) subsS(lion,all(mange,etreVivant)). True car lion est un carnivoreExc
b) subsS(all(mange,canari),carnivoreExc). False car des êtres vivants non carnivoreExc peuvent manger des canaris.

3)
a) subsS(and(carnivoreExc,herbivoreExc),all(mange,nothing)). Renvoie True carnivoreExc et HerbivoreExc mangent respectivement animal et plante. Or plante et animal subsume nothing.

b) subsS(and(and(carnivoreExc,herbivoreExc),animal),nothing). renvoie True en ajoutant la règle subs(and(carnivoreExc,herbivoreExc),nothing). En effet carnivoreExc et herbivoreExc sont incompatibles

c) subsS(and(and(carnivoreExc,animal),herbivoreExc),nothing). Faux ça ne calcule pas le fait que (manger plante et animal) donne nothing

4) subsS(all(R, inst(_, C)), all(R, C), _).