

Programmation des Systèmes Experts

Deuxième Année ENSIMAG 2006/2007

James L. Crowley

Exercice 3

14 mars 2007

Raisonnement Temporel en CLIPS

Il s'agit de programmer en CLIPS 6 un système de propagation de contraintes sur des relations temporelles entre intervalles de temps. Votre système doit utiliser les définitions de templates RELATIONS et TEMP ci-dessous. Le tableau de transitivité est défini par le deffacts "TABLE-DE-TRANSITIVITE" ci-dessous. Vous disposez des fonctions UNION et INTERSECTION définies ci-dessous.

Dans la liste de relations temporelles, il faut remplacer les symbols ">", "<", et "=" par "b" (before), "bi" (before-inverse) et "e" (equals).

```
(deftemplate RELATIONS (slot int1) (slot int2) (multislot relations))
(deftemplate TEMP (slot nom) (slot int1) (slot int2) (multislot relations))

(deftemplate TABLE (slot r1) (slot r2) (multislot relations) )
(deffacts TABLE-DE-TRANSITIVITE
  (TABLE (r1 b) (r2 b) (relations b))
  (TABLE (r1 b) (r2 bi) (relations b bi d di o oi m mi s si f fi e))
  (TABLE (r1 b) (r2 d) (relations b o m d s))
  (TABLE (r1 b) (r2 di) (relations b))
  (TABLE (r1 b) (r2 o) (relations b))
  (TABLE (r1 b) (r2 oi) (relations b o m d s))
  (TABLE (r1 b) (r2 m) (relations b))
  (TABLE (r1 b) (r2 mi) (relations b o m d s))
  (TABLE (r1 b) (r2 s) (relations b))
  (TABLE (r1 b) (r2 si) (relations b))
  (TABLE (r1 b) (r2 f) (relations b o m d s))
  (TABLE (r1 b) (r2 fi) (relations b))
  .....
  (TABLE (r1 fi) (r2 fi) (relations fi))
)

(deffunction UNION (?r1 ?r2)
  (progn$ (?e ?r1)
    (if (not (member ?e ?r2)) then (bind ?r2 (insert$ ?r2 1 ?e))))
  (return ?r2)
)

(deffunction INTERSECTION (?r1 ?r2)
  (bind ?r3 (create$))
  (progn$ (?e ?r1)
    (if (member ?e ?r2) then (bind ?r3 (insert$ ?r3 1 ?e)) ))
)
```

```
(return ?r3)  
)
```

Soit trois intervalles ?A, ?B et ?C et une liste de relations possibles ?RAB entre ?A et ?B et une liste de relations possibles ?RBC entre ?B et ?C. La génération de la liste de relations entre ?A et ?C peut se faire avec trois règles : R1, R2 et R3.

a) (2 points) Pour chaque couple de relations (?rab, ?rbc) tel que ?rab \in ?RAB et ?rbc \in ?RBC, la règle R1 génère un fact de type TEMP en utilisant la table de transitivité. Ecrire la règle R1.

b) (2 points) Quand il existe deux facts de type TEMP avec la même valeur pour int1 et int2, mais ayant les noms différents il faut les combiner. Ecrire la règle R2 qui effectue cette combinaison. (aide : supprimer les deux facts et créer un nouveau fact de type TEMP avec une liste de relations qui est l'UNION des relations des deux facts)

c) (2 points) Une fois que tous les facts de type TEMP sont combinés, il faut recopier la liste de RELATIONS dans un fact de type RELATIONS et supprimer le fact de type TEMP. Ecrire la règle R3 qui effectue cette opération. (aide : assurez-vous que R2 a fini de générer tous les facts de type TEMP avant que R3 s'exécute.)

d) (2 points) Ecrire la règle R4 qui détecte quand deux listes de relations sont générées pour une paire d'intervalles. Cette règle doit supprimer les deux facts de type RELATIONS et créer un nouveau fact de type RELATIONS qui contient l'intersection des listes de relations.