

INFO0054 - Programmation fonctionnnelle

Projet : Tableaux Sémantiques

Groupe s180498-s170220: Martin RANDAXHE, Cyril RUSSE

Table des matières

1	Nouvelles règles	3
2	Représentation formules, tableaux et tableaux sémantiques	3
2.1	Formules	3
2.2	Tableaux	3
2.3	Tableaux sémantiques	3
2.4	Précision	3
3	Règles d'élimination	4
3.1	Ordre	4
3.2	NOT	4
3.3	OR	4
3.4	AND	5
3.5	IFTHEN	5
4	Extension	5
5	Modules	5
6	Extension	5

1 Nouvelles règles

Toutes les expressions EQUIV, XOR, NAND ET XNOR sont des expressions pouvant être réécrites en formules avec uniquement AND, OR et NOT. Donc à l'instar, de nos résolutions pour éliminer le IFTHEN expliqué dans la section 3, nous pourrions les transformer suivant ce procédé :

- $(\text{EQUIV } A \ B) \rightarrow (\text{AND } (\text{OR } A \ (\text{NOT } B)) \ (\text{OR } (\text{NOT } A) \ B))$
- $(\text{XOR } A \ B) \rightarrow (\text{OR } (\text{AND } A \ (\text{NOT } B)) \ (\text{AND } (\text{NOT } A) \ B))$
- $(\text{NAND } A \ B) \rightarrow (\text{OR } (\text{NOT } A) \ (\text{NOT } B))$
- $(\text{XNOR } A \ B) \rightarrow (\text{OR } (\text{AND } (\text{NOT } A) \ (\text{NOT } B)) \ (\text{AND } A \ B))$

2 Représentation formules, tableaux et tableaux sémantiques

Nous allons passer en revue les différents choix de représentation que nous avons décidé d'utiliser pour ce projet.

2.1 Formules

Les formules sont représentées par de simples listes dont le premier élément est un opérateur, et les suivants, les arguments. Ces arguments sont au nombre de un ou deux, en fonction de l'opérateur, s'il est respectivement unaire ou binaire. Les opérateurs "AND", "OR", "IFTHEN" ne peuvent être utilisés que de manière binaire.

Exemples :

- '(AND (IFTHEN (NOT b) a) c)
- '(OR a c)
- '(NOT (NOT b))

2.2 Tableaux

Nous utilisons des listes de formules pour représenter les tableaux.

Exemples :

- '(a (AND a b) (OR a b))
- '((OR a c))
- '((NOT (NOT b)) (IFTHEN a (NOT b)))

2.3 Tableaux sémantiques

Enfin, similairement aux formules et aux tableaux, les tableaux sémantiques sont représentées tels des listes de tableaux.

Exemples :

- ' ((a (AND a b) (OR a b)) ((OR a c)))
- ' (((NOT (NOT b)) (IFTHEN a (NOT b))))

2.4 Précision

Initialement dans notre réflexion, nous avons imaginé les fonctions de sorte à prendre en arguments un tableau sémantique et pas juste un tableau. Nous l'avons précisé correctement dans les spécifications des fonctions, mais cela peut porter à confusion étant donné que, par exemple, la fonction "semstab" prend en argument un tableau sémantique, donc un ensemble de tableaux contenant des formules et renverra un tableau sémantique contenant la concaténation

de tous les tableaux (qui constituaient le tableau sémantique donné) ayant été réduit à l'aide des règles d'élimination.

Par ailleurs, dans les cas où des fonctions telles que : "satisfiable?", qui prend un tableau en argument ou "tautology?", qui prend une formule en argument. Nous avons dû, intrinsèquement à celles-ci, modifier l'argument de sorte à le transformer en tableau sémantique ne contenant qu'un seul élément.

3 Règles d'élimination

Les règles d'élimination sont toutes utilisées dans la fonction "semtab" pour réduire les tableaux lui étant donné en argument en un tableau sémantique.

Nous avons créé trois principales fonctions auxiliaires d'élimination.

- semtab-aux-or : pour retirer les OR
- semtab-aux-and : pour retirer les AND
- semtab-aux-not : pour transformer, les NOT suivis d'un autre opérateur et les (IFTHEN A B) en (OR (NOT A) B)

3.1 Ordre

Chacune de ces fonctions boucle récursivement jusqu'à ce qu'il n'y ait plus aucun élément à transformer/retirer. Et semtab utilise ces fonctions dans l'ordre suivant :

semtab-aux-not > semtab-aux-or > semtab-aux-and.

Elle boucle, elle-même, également récursivement jusqu'à ce qu'il n'y ait plus d'élément à modifier, donc lorsqu'il est constitué que de tableaux contenant que des variables et des (NOT variables).

3.2 NOT

Les NOT suivi d'un autre NOT sont retirés et les NOT suivis d'un autre opérateur sont modifiés suivant les règles d'élimination données que l'on peut visualiser avec ces exemples :

Exemples :

- '(((NOT (NOT b)))) → '((b))
- '(((NOT (OR a b)))) → '(((AND (NOT a) (NOT b))))
- '(((NOT (AND a b)))) → '(((OR (NOT a) (NOT b))))
- '(((NOT (IFTHEN a b)))) → '(((AND a (NOT b))))

3.3 OR

La fonction OR va split les tableaux, du tableau sémantique donné en argument, contenant des OR de sorte à avoir deux tableaux similaires à l'exception de ce OR qui sera remplacé dans le premier tableau par le premier élément de ce OR et de même pour le second tableau avec le deuxième élément.

Exemple :

'((a (AND a b) (OR a b)) ((OR a c))) → '((a (AND a b) a) (a (AND a b) b) (a) (c))

3.4 AND

Les AND sont éliminés en dernier car ils sont les moins contraignants. En effet, ils nécessitent uniquement de retirer les AND.

Exemple :

'((a (AND a b) a) (a (AND a b) b) (a) (c)) → '((a a b a) (a a b b) (a) (c))

3.5 IFTHEN

Les IFTHEN sont modifiés comme expliqué précédemment.

Exemple :

'(((IFTHEN a b))) → '(((OR (NOT a) b)))

4 Extension

Nous n'avons malheureusement pas eu le temps de résoudre la partie 3 concernant les extensions.

5 Modules

Nous avons créé deux modules :

1. "semtab" : qui donne accès à la fonction semtab
2. "bibliothèque" : qui, à l'aide du module semtab, donne accès à six fonctions :
 - satisfiable ?
 - valid ?
 - tautology ?
 - contradiction ?
 - models
 - counterexamples

6 Extension

Nous n'avons malheureusement pas eu le temps de résoudre la partie 4 bonus concernant la visualisation.