INFf3
Programmation logique

Cours 6 : Fondements logiques + déboguage

Benoît Lemaire

Université Grenoble Alpes
L2 - MIASHS
Grenoble – France

Trace

Tracer l’appel courant : **<c>**reep (ou ESPACE ou ENTREE)
Ne pas tracer l’appel courant : **<s>**kip
Recommencer l’appel précédent : **<r>**etry
Arrêter : **<a>**bort

?- trace, tictactoe.
   Call: (9) tictactoe ? **creep**
   Call: (10) tictactoe(lui, [v, v, v, v, v, v, v, v|...]) ? **creep**
   Call: (11) ligne([v, v, v, v, v, v, v, v|...], _694) ? **skip**
   Fail: (11) ligne([v, v, v, v, v, v, v, v|...], _694) ? **retry**
[retry]
   Call: (11) ligne([v, v, v, v, v, v, v, v|...], _694) ? **creep**
...

Il est aussi possible de définir dans le programme le passage en mode trace :

```
p(…) :- N=1, ….
P(…) :- N=2, trace, ...
```

Logique des prédicats

- Représenter des connaissances
- Inférer de nouvelles connaissances
- Exemple :
  - Les poissons n’ont pas de poumons
  - Certains poissons n’ont pas d’écailles
  - Tous les mammifères ont des poumons

  - Peut-on inférer qu’il y a des animaux sans écailles qui ne sont pas des mammifères ?

Représentation logique

- Les poissons n'ont pas de poumons

  $\forall x \ poisson(x) \Rightarrow \neg \ aDesPoumons(x)$

- Certains poissons n'ont pas d'écailles

  $\exists x \ poisson(x) \wedge \neg \ aDesEcailles(x)$

- Tous les mammifères ont des poumons

  $\forall x \ mammifere(x) \Rightarrow aDesPoumons(x)$

- Peut-on inférer qu'il y a des animaux sans écailles qui ne sont pas des mammifères ?

  $\exists x \ \neg \ aDesEcaille(x) \wedge \neg \ mammifere(x)$

Règles d'inférences

- **Modus ponens**

$A \Rightarrow B$	s'il pleut, les escargots sortent
A	il pleut

B	donc les escargots sortent

- **Règle de résolution**

P1 v P2 v ... v Pn

¬P1 v Q2 v ... v Qp

P2 v ... v Pn v Q2 v ... v Qp

Exemples

A
$\neg A \vee B \quad (A \Rightarrow B)$

B

$A \vee B$
$\neg A \vee B$

B

$\neg A \vee B \quad (A \Rightarrow B)$
$\neg B \vee C \quad (B \Rightarrow C)$

$\neg A \vee C \quad (A \Rightarrow C)$

A
$\neg A$
----
nil (contradiction)

La résolution fonctionne sur des clauses

- Clauses = disjonction de littéraux
- On peut toujours transformer une formule en une ou plusieurs clauses
  - Eliminer les $\Rightarrow$
  - Réduire la portée des négations
  - Standardiser les variables
  - Supprimer les $\exists$
  - Mettre sous forme normale conjonctive

Exemple : mettre sous forme de clauses cette formule

$((\exists x\ P(x) \Rightarrow \exists x\ R(x)\ \vee\ \forall y\ P(y))\ \vee\ \neg(\exists y \forall x(R(y) \Rightarrow P\ (x))))$

Exemple : syllogisme de Socrate

Tous les hommes sont mortels

Socrate est un homme

Socrate est mortel

$\downarrow$ Représentation logique

$\forall x$ homme(x) $\Rightarrow$ mortel (x)

homme(Socrate)

$\downarrow$ Mise sous forme de clauses

$\neg$homme(x) $\vee$ mortel (x)

homme(Socrate)

x/Socrate

$\downarrow$ Application de la résolution

mortel(Socrate)

Exemple : syllogisme de Socrate

Tous les hommes sont mortels

Socrate est un homme

Socrate est mortel

↓ Représentation logique

$\forall x\ homme(x) \Rightarrow mortel(x)$

homme(Socrate)

↓ Mise sous forme de clauses

$\neg homme(x) \vee mortel(x)$

homme(Socrate)

Unification !

x/Socrate

↓ Application de la résolution

mortel(Socrate)

Unification

- p(x,y) et p(A,A): yes, {x/A, y/A}
- p(x,x) et p(A,B): fail
- p(x,x) et p(y,z): yes, {x/y, x/z}
- p(x,f(x)) et p(A,y): yes, {x/A, y/f(A)}
- p(x,x) et p(y,f(y)): fail (x ne peut s'unifier avec f(x))

Convention en logique mathématique

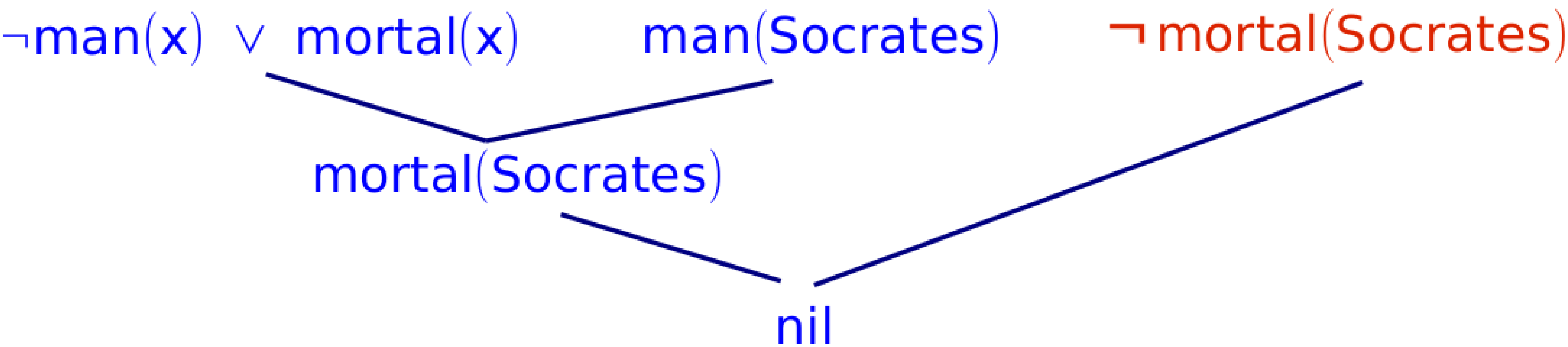
x,y,z : variables

A,B,C.. : constantes

Preuve par réfutation

- Exemple :
  - Les poissons n'ont pas de poumons
  - Certains poissons n'ont pas d'écailles
  - Tous les mammifères ont des poumons
  - Peut-on inférer qu'il y a des animaux sans écailles qui ne sont pas des mammifères ?
- Méthode pour prouver W à partir d'un ensemble de formules S
  - Montrer que $S \cup \neg W$ aboutit à une contradiction (nil) en utilisant la règle de la résolution

Exemple 1


¬man(x) ∨ mortal(x)
man(Socrates)
¬ mortal(Socrates)
mortal(Socrates)
nil

Exemple 2

- Les poissons n'ont pas de poumons

  $\forall x \text{ poisson}(x) \Rightarrow \neg \text{ aDesPoumons}(x)$

  ¬poisson(x) ∨ ¬ aDesPoumons(x)

- Certains poissons n'ont pas d'écailles

  $\exists x \text{ poisson}(x) \wedge \neg \text{ aDesEcailles}(x)$

  poisson(nemo) **¬ aDesEcailles(nemo)**

- Tous les mammifères ont des poumons

  $\forall x \text{ mammifere}(x) \Rightarrow \text{aDesPoumons}(x)$

  ¬mammifere(x) ∨ aDesPoumons(x)

- Peut-on inférer qu'il y a des animaux sans écailles qui ne sont pas des mammifères ?

  $\neg\ (\exists x\ \neg \text{ aDesEcaille}(x) \wedge \neg \text{ mammifere}(x))$

  aDesEcaille(x) ∨ mammifere(x))

Exemple 2

- Les poissons n'ont pas de poumons

  ∀x poisson(x) ⇒ ¬ aDesPoumons(x)

  ¬poisson(x) ∨ ¬ aDesPoumons(x)

- Certains poissons n'ont pas d'écailles

  ∃x poisson(x) ∧ ¬ aDesEcailles(x)

  poisson(nemo) **¬ aDesEcailles(nemo)**

- Tous les mammifères ont des poumons

  ∀x mammifere(x) ⇒ aDesPoumons(x)

  ¬mammifere(x) ∨ aDesPoumons(x)

- Peut-on inférer qu'il y a des animaux sans écailles qui ne sont pas des mammifères ?

  ¬ (∃x ¬ aDesEcaille(x) ∧ ¬ mammifere(x))

  aDesEcaille(x) ∨ mammifere(x))

¬ aDesPoumons(nemo)

nil

aDesPoumons(nemo)

mammifere(nemo)

Algorithme

```
action resolutionRefutation(W,S) {
    clauses ← S
    while (nil ∉ clauses) {
        selectionner Ci and Cj ∈ clauses
        calculer la résolvante Rij
        clauses = clauses ∪ Rij
    }
}
```

Prolog

- On ne peut utiliser que des clauses qui ont zéro ou un seul littéral positif (appelées clauses de Horn)

$P1 \lor \neg P2 \lor \ldots \lor \neg Pn$

$= \neg(\neg P2 \lor \ldots \lor \neg Pn) \Rightarrow P1$

$= P2 \land \ldots \land \neg Pn \Rightarrow P1$

= P1 :- P2, ..., Pn

Exercice

- Quiconque possède un smartphone a une connexion Internet
- Quiconque n’est pas malade et a une connexion Internet peut se connecter à Moodle
- Igor possède un smartphone et n’est pas malade
- Laura possède un smartphone et elle est malade
- Quiconque peut se connecter à Moodle peut travailler ses TD à l’avance
- Quelqu’un peut-il travailler ses TD à l’avance ?