Logique du premier ordre
Deuxième partie :
Interprétation d'une formule

Benjamin Wack

Université Grenoble Alpes

Mars 2025

Rappels : formalisation au premier ordre

$a^{r2}(x,y)$ signifie « x aime y », $c^{f1}(x)$ dénote le conjoint de x

Traduire en logique du premier ordre :

- Il y a des gens qui s'aiment.

  $\exists x\ \exists y\ \ (a(x,y) \wedge a(y,x))$

- Si deux personnes s'aiment l'une l'autre, alors elles sont conjointes.

  $\forall x\ \forall y\ \ (a(x,y) \wedge a(y,x) \Rightarrow c(x) = y \wedge c(y) = x)$

- On ne peut pas aimer deux personnes à la fois.

  $\forall x\ \forall y\ \ (a(x,y) \Rightarrow \forall z\ \ (a(x,z) \Rightarrow y = z))$

  OU : $\forall x\ \forall y\ \forall z\ \ (a(x,y) \wedge a(x,z) \Rightarrow y = z)$

Plan

Plan

Sens des formules

Interprétation et substitution

Interprétation finie par expansion : compléments

Equivalences remarquables

Conclusion

Rappel : Interprétation

Définition 4.3.16

Une interprétation I sur une signature Σ est définie par :

- un domaine D non vide
- pour chaque symbole $s^{gn} \in \Sigma$:

(constante)	s_I^{f0} est un élément de D
(fonction)	s_I^{fn} est une fonction de $D^n \rightarrow D$
(variable propositionnelle)	s_I^{r0} vaut 0 ou 1
(relation)	s_I^{rn} est un ensemble de n-uplets dans D^n

État, assignation

Une interprétation définit seulement le sens de la signature (les symboles), pas celui des variables ni des formules.

Définition 4.3.21

Un état e d'une interprétation associe à chaque variable un élément du domaine D.

Définition 4.3.22

Une assignation est un couple (I, e) composé d'une interprétation I et d'un état e.

Exemple 4.3.23

Soient le domaine $D = \{1,2,3\}$ et l'interprétation I
$ami_I^{r2} = \{(1,2),(1,3),(2,3)\}$

L'interprétation I ne suffit pas à déterminer le sens de $ami(x,y)$.

Soit e l'état qui associe 2 à x et 1 à y.

L'assignation (I,e) rend la formule $ami(x,y)$ fausse.

Remarque 4.3.24

- Pour une formule avec des variables libres, nous avons besoin d'une assignation (I, e) dont l'état est précisé.
- Pour une formule sans variable libre, il suffit de donner une interprétation I des symboles de la formule.

  En effet (I, e) et (I, e') donneront la même valeur à toutes les formules :
  on assimile donc (I, e) et I.

Attention : Interprétation à deux niveaux

- Pour une **formule** A, sa signification $[A]$ est un **booléen** (dans une assignation donnée).
- Pour un **terme** t, on calculera $[\![t]\!]$ (un **élément du domaine**).

Définition 4.3.25 Évaluation d'un terme

Définition inductive :

1. si t est une variable, alors $[\![t]\!]_{(I,e)} = e(t)$
2. si t est une constante alors $[\![t]\!]_{(I,e)} = t_I^{f0}$
3. si $t = s(t_1, \ldots, t_n)$ avec s un symbole de fonction, alors $[\![t]\!]_{(I,e)} = s_I^{fn}([\![t_1]\!]_{(I,e)}, \ldots, [\![t_n]\!]_{(I,e)})$

Exemple 4.3.26

Soit la signature a^{f0}, f^{f2}, g^{f2}.
Soit I l'interprétation de domaine $\mathbb{N}$ dans laquelle :

- a est interprété par l'entier 1 ;
- f est interprétée comme le produit ;
- g est interprétée comme la somme.

Soit e l'état tel que $e(x) = 2$ and $e(y) = 3$.
Calculons $[\![f(x, g(y, a))]\!]_{(I,e)}$.

$$
\begin{array}{rcl}
[\![f(x, g(y, a))]\!]_{(I,e)} & = & [\![x]\!]_{(I,e)} * ([\![y]\!]_{(I,e)} + [\![a]\!]_{(I,e)}) \\
 & = & 2 * (3 + 1) = 8
\end{array}
$$

Formules

Définition 4.3.27 Sens des formules atomiques

Selon la forme de la formule atomique, on pose :

1. Une constante propositionnelle : $[\top]_{(I,e)} = \mathit{vrai}$ et $[\bot]_{(I,e)} = \mathit{faux}$
2. Une variable propositionnelle : $[s]_{(I,e)} = s_I^{r0}$
3. Un terme $A = s(t_1, \ldots, t_n)$ avec s un symbole de relation :
   - si $([\![t_1]\!]_{(I,e)}, \ldots, [\![t_n]\!]_{(I,e)}) \in s_I^{rn}$ alors $[A]_{(I,e)} = \mathit{vrai}$
   - sinon $[A]_{(I,e)} = \mathit{faux}$

Exemple 4.3.19

On considère la signature suivante :

- $Anne^{f0}$, $Bernard^{f0}$ et $Claude^{f0}$: constantes
- a^{r2} : relation à deux arguments ($a(x,y)$ signifie « x aime y »)
- c^{f1} : fonction à un argument ($c(x)$ dénote le conjoint de x)

Une interprétation possible sur cette signature est l'interprétation I de domaine $D = \{0,1,2\}$ où :

- $Anne_I^{f0} = 0$, $Bernard_I^{f0} = 1$, et $Claude_I^{f0} = 2$
- $a_I^{r2} = \{(0,1),(1,0),(2,0)\}$
- c_I^{f1} est une fonction de D dans D qu'on définit par

x	0	1	2
$c_I^{f1}(x)$	1	0	2

Exemple 4.3.29

Nous obtenons :

- $[a(Anne, Bernard)]_I =$

  $\boxed{vrai \text{ car } (0,1) \in a_I^{r2}.}$

- $[a(Anne, Claude)]_I =$

  $\boxed{faux \text{ car } (0,2) \notin a_I^{r2}.}$

Ici on a utilisé *vrai* et *faux* au lieu des valeurs de vérité $0, 1$ pour les distinguer des éléments $0, 1$ du domaine (attention suivant le contexte).

Exemple 4.3.29

Soit e l'état $x = 0, y = 2$. Nous avons :

- $[a(x, c(x))]_{(I,e)} =$

  $\boxed{\textit{vrai} \ \text{ car } \ c_I^{f1}(0) = 1 \ \text{ et } \ (0,1) \in a_I^{r2}.}$

- $[a(y, c(y))]_{(I,e)} =$

  $\boxed{\textit{faux} \ \text{ car } \ c_I^{f1}(2) = 2 \ \text{ et } \ (2,2) \not\in a_I^{r2}.}$

Exemple 4.3.29

Nous avons :

- $[(Anne = Bernard)]_I =$

  $\boxed{\textit{faux}, \text{ car } (0,1) \notin =_I^{r2}.}$

- $[(c(Anne) = Anne)]_I =$

  $\boxed{\textit{faux}, \text{ car } c_I^{f1}(0) = 1 \text{ et } (1,0) \notin =_I^{r2}.}$

- $[(c(c(Anne)) = Anne)]_I =$

  $\boxed{\textit{vrai}, \text{ car } c_I^{f1}(c_I^{f1}(0)) = 0 \text{ et } (0,0) \in =_I^{r2}.}$

Sens des formules non atomiques 4.3.30

1. Les connecteurs propositionnels ont le même sens qu'en logique propositionnelle.
2. Notons $e[x = d]$ l'état identique à l'état e, sauf pour x.

$$[\forall xB]_{(I,e)} = min_{d \in D}[B]_{(I,e[x=d])} = \prod_{d \in D}[B]_{(I,e[x=d])},$$

   vrai si $[B]_{(I,f)} = vrai$ pour tout état f identique à e, sauf pour x.
3.

$$[\exists xB]_{(I,e)} = max_{d \in D}[B]_{(I,e[x=d])} = \sum_{d \in D}[B]_{(I,e[x=d])},$$

   vrai s'il y a un état f identique à e, sauf pour x, tel que $[B]_{(I,f)} = vrai$.

Exemple 4.3.32

Utilisons l'interprétation I donnée dans l'exemple 4.3.19.
(Rappel $D = \{0,1,2\}$)

- $[\exists x\ a(x,x)]_I$

$$\boxed{= [a(0,0)]_I + [a(1,1)]_I + [a(2,2)]_I = \mathit{faux} + \mathit{faux} + \mathit{faux} = \mathit{faux}.}$$

- $[\forall x \exists y\ a(x,y)]_I$

$$= [\exists y\ a(0,y)]_I \ .\ [\exists y\ a(1,y)]_I \ .\ [\exists y\ a(2,y)]_I$$

$$\begin{aligned} = ([a(0,0)]_I + [a(0,1)]_I + [a(0,2)]_I) \\ .\ ([a(1,0)]_I + [a(1,1)]_I + [a(1,2)]_I) \\ .\ ([a(2,0)]_I + [a(2,1)]_I + [a(2,2)]_I) \end{aligned}$$

$$= (\mathit{faux} + \mathit{vrai} + \mathit{faux})\ .\ (\mathit{vrai} + \mathit{faux} + \mathit{faux})\ .\ (\mathit{vrai} + \mathit{faux} + \mathit{faux})$$

$$= \mathit{vrai}.\mathit{vrai}.\mathit{vrai} = \mathit{vrai}.$$

Exemple 4.3.32

- $[\exists y \forall x \ a(x,y)]_I$

$$\boxed{\begin{aligned} &= [a(0,0)]_I.[a(1,0)]_I.[a(2,0)]_I + [a(0,1)]_I.[a(1,1)]_I.[a(2,1)]_I \\ &\quad + [a(0,2)]_I.\ [a(1,2)]_I.\ [a(2,2)]_I \\ &= \mathit{faux}.\mathit{vrai}.\mathit{vrai} + \mathit{vrai}.\mathit{faux}.\mathit{faux} + \mathit{faux}.\mathit{faux}.\mathit{faux} \\ &= \mathit{faux} + \mathit{faux} + \mathit{faux} = \mathit{faux}. \end{aligned}}$$

Remarque 4.3.33

Les formules $\forall x \exists y \ a(x,y)$ et $\exists y \forall x \ a(x,y)$ n'ont pas la même valeur.
En intervertissant un $\exists$ et un $\forall$, on ne préserve pas le sens des formules.

Modèle, validité, conséquence, équivalence

Ces notions sont définies comme en logique propositionnelle mais...

... on utilise une interprétation au lieu d'une assignation.

Pour donner une valeur à une formule

- **En logique propositionnelle** : assignation $V \to \{0, 1\}$
- **En logique du premier ordre** : (I, e) où
  - I est une interprétation des symboles
  - e un état des variables.

La valeur d'une formule dépend :

- de l'état de ses variables libres
- ET de l'interprétation de ses symboles.

Plan

Au niveau propositionnel

Rappel : substituer une variable propositionnelle dans une formule valide donne une formule valide.

Exemple :

Soit $\sigma(p) = \forall x\ q(x)$.
$p \vee \neg p$ est valide, il en est de même de la formule

$$\sigma(p \vee \neg p) = \forall x\ q(x) \vee \neg \forall x\ q(x)$$

Le principe de remplacement est encore valable aussi car :

Pour toutes formules A et B et toute variable x :

- $(A \Leftrightarrow B) \models (\forall x A \Leftrightarrow \forall x B)$
- $(A \Leftrightarrow B) \models (\exists x A \Leftrightarrow \exists x B)$

Instanciation d'une variable dans un terme

Définition 4.3.34

$A < x := t >$ est la formule obtenue en remplaçant dans A toute occurrence libre de x par t.

Exemple 4.3.35

Soit A la formule $(\forall x P(x) \vee Q(\mathbf{x}))$, la formule $A < x := b >$ vaut

$(\forall x P(x) \vee Q(b))$ car seule l'occurrence en gras de x est libre.

Mais on ne peut pas substituer n'importe quoi :

Exemple 4.3.37

Soit A la formule $\exists y p(x, y)$.

- $A < x := y >= \boxed{\exists y p(y, y)}$ (phénomène de capture)

La capture change le sens :

Exemple 4.3.37

Soit p une relation binaire interprétée sur $\{0,1\}$ par $p_I = \{(0,1)\}$
Soit e un état où $x = 0$ (et $y = 0$).

- $[A < x := y >]_{(I,e)} =$

  $\boxed{[\exists y p(y,y)]_{(I,e)} = [p(0,0)]_{(I,e)} + [p(1,1)]_{(I,e)} = faux + faux = faux.}$

- Alors que $[A]_{(I,e)} =$

  $\boxed{[\exists y p(x,y)]_{(I,e)} = [p(0,0)]_{(I,e)} + [p(0,1)]_{(I,e)} = faux + vrai = vrai.}$

Ainsi, $[A < x := y >]_{(I,e)} \neq [A]_{(I,e)}$, même si $e(x) = e(y)$.

Instanciation d'une variable dans un terme : précautions

Solution : notion de terme t libre pour une variable

Définition 4.3.34 (suite)

t est libre pour x dans A si aucune variable de t n'est liée dans A aux emplacements (libres) de x.

(Si aucune capture n'advient en remplaçant x par t dans A.)

Exemple 4.3.35

- Le terme $f(z)$ est libre pour x dans la formule $\exists y p(x, y)$.
- Par contre les termes y ou $g(y)$ ne sont pas libres pour x dans cette formule.
- Par définition, le terme x est libre pour x dans toute formule.

Propriétés

Théorème 4.3.36

Soient A une formule et t un terme libre pour la variable x dans A.
Pour toute assignation (I, e) nous avons
$[A < x := t >]_{(I,e)} = [A]_{(I,e[x=d])}$ où $d = [\![t]\!]_{(I,e)}$.

Corollaire 4.3.38

Soient A une formule et t un terme libre pour x dans A.
Les formules $\forall x A \Rightarrow A < x := t >$ et $A < x := t > \Rightarrow \exists x A$ sont valides.

Plan

William McCune (1953-2011)

- Auteur de plusieurs systèmes de raisonnement automatisés : Otter, Prover9, Mace4

MACE

- **expansion** des formules du premier ordre
- **algorithmes performants** comme `DPLL`

`http://www.cs.unm.edu/~mccune/mace4/examples/2009-11A/mace4-misc/`

- 1996 : Preuve de la conjecture de Robbins à l'aide du logiciel de démonstration automatisée EQP
  - 8 jours de calcul sur un processeur à 66 MHz, 30 Mo de mémoire
  - production d'un témoin de preuve par Otter, vérifié par un troisième programme

(Conjecture sans réponse depuis 1933)

Rappels à propos de la méthode des expansions

Recherche de modèles à n éléments **par réduction au cas propositionnel**
Cas simple : formule n'ayant ni symbole de fonction ni constante.

Construction du modèle à n éléments

1. suppression des quantificateurs par expansion sur un domaine à n éléments ;
2. remplacement des égalités par leur valeur ;
3. recherche d'une assignation propositionnelle des formules atomiques qui soit modèle de la formule.

Recherche d'un modèle fini d'une formule fermée **avec** symbole de fonction

Soit A une formule fermée quelconque.

Procédure

- Remplacer A par sa n-expansion.
- Enumérer les choix des valeurs des symboles, en propageant le plus possible chacun des choix effectués.

Similaire à *l'algorithme de* `DPLL`.

Exemple 4.3.46 : $A = \exists y P(y) \Rightarrow P(a)$

On cherche un contre-modèle à 2 éléments.

2-expansion de A

$$P(0) + P(1) \Rightarrow P(a)$$

Il reste à trouver les valeurs de $P(0)$, $P(1)$ et a.
On choisit (arbitrairement) $a = 0$:

$$P(0) + P(1) \Rightarrow P(0)$$

$P(0) = \mathit{faux}, P(1) = \mathit{vrai}$ est un contre-modèle propositionnel,

ou encore I de domaine $\{0, 1\}$ telle que $P_I = \{1\}$ et $a_I = 0$.

Assignation VS interprétation

Soit A une formule fermée, sans quantificateur ni symbole de fonction.
Soit P l'ensemble des formules atomiques de A.

Théorème 4.3.42 et 4.3.44

Pour toute assignation propositionnelle $v : P \rightarrow \{faux, vrai\}$
il existe une interprétation I de A telle que $[A]_I = [A]_v$.
Pour toute interprétation I
il existe une assignation $v : P \rightarrow \{faux, vrai\}$ telle que $[A]_I = [A]_v$.

Exemple 4.3.43

La formule $(p(0) + p(1)) \Rightarrow (p(0).p(1))$ est rendue fausse par :

- l'assignation v définie par $[p(0)]_v = vrai$ et $[p(1)]_v = faux$
- ou l'interprétation I définie par $p_I = \{0\}$

v et I sont deux façons de présenter la même interprétation.

Exemple 4.3.47 : $P(a), \forall x(P(x) \Rightarrow P(f(x))), \neg P(f(b))$

1. 2-expansion :
   $F = \{P(a),\ (P(0) \Rightarrow P(f(0))).(P(1) \Rightarrow P(f(1))),\ \neg P(f(b))\}$.
2. On cherche des valeurs de $P(0)$, $P(1)$, a, b, $f(0)$ et $f(1)$ modèle de F.
3. Choisissons $a = 0$
   - De $P(a) = vrai$ et $a = 0$, on déduit : $P(0) = vrai$
   - De $P(0) = vrai$ et de $(P(0) \Rightarrow P(f(0))) = vrai$, on déduit : $P(f(0)) = vrai$
   - De $P(f(b)) = faux$ et de $P(f(0)) = vrai$, on déduit $f(0) \neq f(b)$ donc que $b \neq 0$, donc : $b = 1$ et $P(f(1)) = faux$
   - De $P(f(1)) = faux$ et $P(0) = vrai$, on déduit $f(1) \neq 0$ donc : $f(1) = 1$ et $P(1) = faux$
   - De $P(f(0)) = vrai$ et $P(1) = faux$, on déduit : $f(0) = 0$

Modèle : $a = 0, b = 1, P = \{0\}, f(0) = 0, f(1) = 1$

Correction de la méthode

$$\begin{array}{ccccccc} & n\text{-expansion} & & \text{simplifications} & & & \\ A & \longrightarrow & B & \longrightarrow & C & \longrightarrow & C \\ \text{(1er ordre)} & \equiv_n & \text{(sans } \forall\exists\text{)} & \equiv & \text{(sans } =\text{)} & \approxeq & \text{(propositionnelle)} \end{array}$$

- $[A]_I = [B]_I$ pour toute I sur un domaine de taille n
- $B \equiv C$ par construction (donc $[B]_I = [C]_I$ pour toute I)
-
  - À toute v correspond une I telle que $[C]_I = [C]_v$.
  - À toute I correspond une v telle que $[C]_I = [C]_v$.

A a donc un modèle I sur un domaine de taille n
si et seulement si
C a un modèle v.

(Et v permet de retrouver I si besoin.)

Plan

Lois de De Morgan généralisées : $\forall$ et $\exists$

Lemme 4.4.1

Soient A une formule et x une variable :

$$\begin{aligned}
\neg\forall x A &\equiv \exists x \neg A \\
\forall x A &\equiv \neg\exists x \neg A \\
\neg\exists x A &\equiv \forall x \neg A \\
\exists x A &\equiv \neg\forall x \neg A
\end{aligned}$$

Déplacement des quantificateurs

Soient x, y deux variables et A, B deux formules.

1. $\forall x \forall y A \equiv \forall y \forall x A$
2. $\exists x \exists y A \equiv \exists y \exists x A$
3. $\forall x (A \wedge B) \equiv (\forall x A \wedge \forall x B)$
4. $\exists x (A \vee B) \equiv (\exists x A \vee \exists x B)$
5. Soient Q un quantificateur et $\circ$ un des connecteurs $\wedge, \vee$
   Si x n'est pas une variable libre de A alors :
   5.1 $Qx\ A \equiv A$,
   5.2 $Qx\ (A \circ B) \equiv A \circ (Qx\ B)$

Exemple 4.4.2

Nous éliminons de ces deux formules les quantificateurs inutiles :

- $\forall x \exists x P(x) \equiv$

  $\boxed{\exists x P(x)}$

- $\forall x (\exists x P(x) \vee Q(x)) \equiv$

  $\boxed{\exists x P(x) \vee \forall x Q(x)}$

Changement de variables liées (1/3)

Théorème 4.4.3

Soit Q un quantificateur.
Si y ne figure pas dans $Qx\ A$ alors : $Qx\ A \equiv Qy\ A < x := y >$

Exemple 4.4.4

- $\forall x\ p(x,z) \equiv \forall y\ p(y,z)$
- $\forall x\ p(x,z) \not\equiv \forall z\ p(z,z)$

Plan

Aujourd'hui

- Évaluer une formule, c'est choisir une interprétation pour ses **symboles** et un état pour ses **variables**
- Méthode des expansions pour trouver un **(contre-)modèle**
- Équivalences remarquables à propos des quantificateurs (attention, **pas de notion de forme normale** utilisable pour trouver les modèles)

La prochaine fois

- Skolémisation
- Semi-algorithme pour montrer qu'une formule est insatisfaisable.

À chercher

Tous les hommes sont mortels.
Socrate est un homme.
Donc Socrate est mortel.

- Rechercher un contre-modèle par 1-expansion puis par 2-expansion.
- Que peut-on en conclure ?