La Logique des Prédicats

ou *logique du premier ordre*

3 Introduction

En logique des prédicats,

- on ajoute les quantificateurs
- on généralise les valeurs que peuvent prendre les variables (0,1 pour la logique propositionnelle, des valeurs sur un ensemble quelconque pour la logique des prédicats)
- on ajoute des relations (appelés prédicats) pour décrire certaines relations entre ces valeurs ($R(x, y)$ veut dire que x et y sont en relation)
- on ajoute des symboles de fonctions à la syntaxe

Exemple : $\forall x \forall y \cdot \mathit{PremierEntreEux}(x, y) \leftrightarrow \exists x' \exists y' \cdot x.x' + y.y' = 1$

- $\mathit{PremierEntreEux}$ est un prédicat à deux arguments,
- 1 est appelée *constante*
- $x.x' + y.y'$ est un *terme* formé avec les fonctions $\times$ et $.$

4 Langages du premier ordre

- on parle ici de langages car les objets syntaxiques qui seront à la base de la construction des formules sont déterminés par des vocabulaires qu'on fixera (symboles de relations, de fonctions, ...). Il existera donc une grande variété de langages du premier ordre selon le vocabulaire choisi.
- L'expression "premier ordre" différencie ces langages des langages "d'ordre supérieur", dans lesquels il est possible de quantifier sur d'autres objets que les variables (par ex. sur les fonctions, prédicats, ...)

5 Des exemples de formules

- $\forall x,\ p(x,x)$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \rightarrow x = y$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \rightarrow x = y$
- $\forall x, [(p(x) \rightarrow \neg p(S(x))) \wedge (\neg p(x) \rightarrow p(S(x))]$

Comment, dans les entiers, interpréter p et S pour que cette formule soit vraie ? On pourrait interpréter p par "être pair" et S par $n \longmapsto n+1$ $\forall n \in \mathbb{Z}$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \to p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \to x = y$
- $\forall x, [(p(x) \to \neg p(S(x))) \wedge (\neg p(x) \to p(S(x))]$
- $\exists x \forall y,\ \neg(S(y) = x)$

↳ dans $\mathbb{Z}$, on pourrait interpréter
S par $n \mapsto n^2+1$ $\forall n \in \mathbb{Z}$

↳ dans $\mathbb{N}$, la même interprétation rend
vraie la formule, ainsi que $n \mapsto n+1$.

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \rightarrow x = y$
- $\forall x, [(p(x) \rightarrow \neg p(S(x))) \wedge (\neg p(x) \rightarrow p(S(x))]$
- $\exists x \forall y,\ \neg(S(y) = x)$
- $\forall y,\ \neg(S(y) = c)$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \rightarrow x = y$
- $\forall x, [(p(x) \rightarrow \neg p(S(x))) \wedge (\neg p(x) \rightarrow p(S(x))]$
- $\exists x \forall y,\ \neg(S(y) = x)$
- $\forall y,\ \neg(S(y) = c)$
- $\forall x \forall y,\ [f(x,y) = f(y,x) \wedge f(x,S(y)) = S(f(x,y)) \wedge f(x,c) = x]$

5 Des exemples de formules

- $\forall x,\ p(x,x)$
- $\forall x \forall y,\ p(x,y) \rightarrow p(y,x)$
- $\forall x \forall y,\ f(x) = f(y) \rightarrow x = y$
- $\forall x, [(p(x) \rightarrow \neg p(S(x))) \wedge (\neg p(x) \rightarrow p(S(x))]$
- $\exists x \forall y,\ \neg(S(y) = x)$
- $\forall y,\ \neg(S(y) = c)$
- $\forall x \forall y,\ [f(x,y) = f(y,x) \wedge f(x,S(y)) = S(f(x,y)) \wedge f(x,c) = x]$

Les ingrédients pour construire les formules sont : connecteurs Booléens, quantificateurs, symboles de relations, de fonctions, constantes, et termes.
Pour satisfaire une formule, il faudra définir une *interprétation* des symboles (relations, fonctions, constantes) dans un *domaine*.

6 Langages du premier ordre : alphabet

L'alphabet d'un langage du premier ordre comporte d'abord les symboles suivants qui sont *communs à tous ces langages* :

- les connecteurs $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$;
- les parenthèses $(,)$;
- le *quantificateurs universel* $\forall$ et le *quantificateur existentiel* $\exists$;
- un ensemble infini V de symboles de *variables* $x, y, z, \ldots$.

7 Langages du premier ordre : alphabet

Un *langage* $\mathcal{L}$ de la logique du premier ordre est caractérisé par :

- des *symboles de relations* (appelés aussi *prédicats*), notés $p, q, r, s...$;
- des *symboles de fonctions* , notés $f, g, h, ...$;
- des *symboles de constantes* , notés $c, d, e, ...$

A chaque prédicat p, respectivement fonction f, on associe un entier strictement positif appelé l'*arité* de p, respectivement de f, c'est-à-dire le nombre d'arguments de p, respectivement f. On notera parfois $p|_n$ ou $f|_n$ pour signifier que p (resp. f) est un symbole de relation (resp. de fonction) d'arité n.

7 Langages du premier ordre : alphabet

Un *langage* $\mathcal{L}$ de la logique du premier ordre est caractérisé par :

- des *symboles de relations* (appelés aussi *prédicats*), notés $p, q, r, s...$;
- des *symboles de fonctions* , notés $f, g, h, ...$;
- des *symboles de constantes* , notés $c, d, e, ...$

A chaque prédicat p, respectivement fonction f, on associe un entier strictement positif appelé l'*arité* de p, respectivement de f, c'est-à-dire le nombre d'arguments de p, respectivement f. On notera parfois $p|_n$ ou $f|_n$ pour signifier que p (resp. f) est un symbole de relation (resp. de fonction) d'arité n.
On utilise le prédicat "$=$" pour dénoter l'égalité et on supposera qu'il est toujours présent (même si on ne l'indique pas dans $\mathcal{L}$).

8 Langages du premier ordre : alphabet

Exemples de langages :

- $\mathcal{L}_1 = \{r|_1, c\}$ contient un prédicat unaire r et une constante c ;
- $\mathcal{L}_2 = \{r|_2, f|_1, g|_2, h|_2, c, d\}$ contient un prédicat binaire r, une fonction unaire f, deux symboles de fonctions binaires g et h, et deux constantes c et d.

9 Langages du premier ordre : construction des termes

L'ensemble des *termes d'un langage* $\mathcal{L}$ est le plus petit ensemble qui contient les symboles de constantes et de variables et qui est clos par application des fonctions.
L'ensemble des termes, noté $\mathcal{T}$, est le plus petit ensemble satisfaisant :

1. tout symbole de constante ou variable est un terme ;
2. si f est un symbole de fonction d'arité n et $t_1, t_2, \ldots, t_n$ sont des termes alors $f(t_1, t_2, \ldots, t_n)$ est un terme.

Remarque : les prédicats ne fournissent pas des termes ; ils serviront pour construire les formules.

10 Langages du premier ordre : construction des termes

Exemples :

- les seuls termes du langage $\mathcal{L}_1$ sont la constante c et les variables ;
- les expressions suivantes sont des termes du langage $\mathcal{L}_2$:
  - $f(c)$;
  - $f(h(f(c), d))$;
  - $f(y)$;
  - $f(h(f(x), f(d)))$.

Un terme est *clos* s'il est sans variable. Par exemple $f(c)$ est clos.

11 Langages du premier ordre : construction des formules

L'ensemble des *formules atomiques* d'un langage $\mathcal{L}$ est l'ensemble des formules de la forme :

- $p(t_1, t_2, \ldots, t_n)$ ou p est un prédicat d'arité n et $t_1, t_2, \ldots, t_n$ sont des termes du langage $\mathcal{L}$;

Lorsque p est le prédicat $=$, on notera $t_1 = t_2$ au lieu de $= (t_1, t_2)$.

12 Langages du premier ordre : construction des formules

Définition

L'ensemble des *formules du langage* $\mathcal{L}$, que l'on désigne par $\mathcal{F}(\mathcal{L})$, est défini par la grammaire suivante :

$$\phi ::= p(t_1, \ldots, t_n) \mid \phi \wedge \phi \mid \phi \vee \phi \mid \neg\phi \mid \phi \rightarrow \phi \mid \phi \leftrightarrow \phi \mid \exists x \cdot \phi \mid \forall x \cdot \phi \mid (\phi)$$

- $t_1, \ldots, t_n$ sont des termes
- p est un symbole de relation
- $\exists x$ est appelé *quantificateur existentiel*
- $\forall x$ est appelé *quantificateur universel*

13 Langages du premier ordre : exemples de formules

La formule $r(c) \vee \neg\exists x \cdot r(x)$ est une formule du langage $\mathcal{L}_1$.
Exemples de formules du langage $\mathcal{L}_2$:

- $\forall x \cdot \exists y(g(x,y) = c \wedge g(y,x) = c)$
- $\forall x \cdot \neg(f(x) = c)$

Exemples de formules du langage $\mathcal{L}_3 = \{p\}$ où p est un symbole de prédicat binaire :

- $\forall x \cdot \forall y \cdot (p(x,y) \rightarrow p(y,x))$
- $\forall x \cdot p(x,x)$
- $\forall x \cdot \forall y \cdot (p(x,y) \vee p(y,x))$
- $\forall x \cdot \forall y \cdot \forall z \cdot (p(x,y) \wedge p(y,z) \rightarrow p(x,z))$
- Comment exprimer que p doit être interprétée par une fonction ?

$$\forall x \ \forall y_1 \forall y_2 \left(\left(p(x,y_1) \wedge p(x,y_2) \right) \longrightarrow y_1 = y_2 \right)$$

13 Langages du premier ordre : exemples de formules

La formule $r(c) \vee \neg\exists x \cdot r(x)$ est une formule du langage $\mathcal{L}_1$.
Exemples de formules du langage $\mathcal{L}_2$:

- $\forall x \cdot \exists y(g(x,y) = c \wedge g(y,x) = c)$
- $\forall x \cdot \neg(f(x) = c)$

Exemples de formules du langage $\mathcal{L}_3 = \{p\}$ où p est un symbole de prédicat binaire :

- $\forall x \cdot \forall y \cdot (p(x,y) \rightarrow p(y,x))$
- $\forall x \cdot p(x,x)$
- $\forall x \cdot \forall y \cdot (p(x,y) \vee p(y,x))$
- $\forall x \cdot \forall y \cdot \forall z \cdot (p(x,y) \wedge p(y,z) \rightarrow p(x,z))$
- Comment exprimer que p doit être interprétée par une fonction ?
- $\forall x \cdot \forall y \cdot \forall z \cdot (p(x,y) \wedge p(x,z) \rightarrow y = z)$.

14 Langages du premier ordre : règles de précédence

- pour les connecteurs Booléens, on garde les mêmes règles de précédence que dans la logique propositionnelle
- les quantificateurs ont la même priorité que la négation

Exemple :

- $\forall x \cdot \neg p(x, y) \vee p(y, x) \equiv (\forall x \cdot \neg(p(x, y))) \vee p(y, x)$

15 Langages du premier ordre : variables libres et liées

Une *occurrence d'une variable* dans une formule est un couple constitué de cette variable et d'une place effective, c'est-à-dire qui ne suit pas un quantificateur.
Par exemple, dans la formule

$$r(x, z) \rightarrow \forall z \cdot (r(y, z) \vee y = z)$$

la variable x possède une occurrence, la variable y deux et la variable z trois.

16 Langages du premier ordre : variables libres et liées

Définition

- Une occurrence d'une variable x dans une formule ϕ est une *occurrence libre* si elle ne se trouve dans aucune sous-formule de ϕ, qui commence par une quantification $\forall x$ ou $\exists x$.
- Dans le cas contraire, l'occurrence est dite *liée*.
- Une variable est *libre* dans une formule si elle a au moins une occurrence libre dans cette formule.
- Une *formule close* est une formule sans variable libre.
- On note $\mathit{Libres}(\phi)$ l'ensemble des variables libres de ϕ.

16 Langages du premier ordre : variables libres et liées

Définition

- Une occurrence d'une variable x dans une formule ϕ est une *occurrence libre* si elle ne se trouve dans aucune sous-formule de ϕ, qui commence par une quantification $\forall x$ ou $\exists x$.
- Dans le cas contraire, l'occurrence est dite *liée*.
- Une variable est *libre* dans une formule si elle a au moins une occurrence libre dans cette formule.
- Une *formule close* est une formule sans variable libre.
- On note $\mathit{Libres}(\phi)$ l'ensemble des variables libres de ϕ.

Exemples :

- dans $\exists x \cdot p(x, y)$, l'occurrence de x est liée et celle de y est libre
- dans $r(x, z) \rightarrow \forall z \cdot (r(y, z) \vee y = z)$, la première occurrence de z est libre et les deux suivantes sont liées

16 Langages du premier ordre : variables libres et liées

Définition

- Une occurrence d'une variable x dans une formule ϕ est une *occurrence libre* si elle ne se trouve dans aucune sous-formule de ϕ, qui commence par une quantification $\forall x$ ou $\exists x$.
- Dans le cas contraire, l'occurrence est dite *liée*.
- Une variable est *libre* dans une formule si elle a au moins une occurrence libre dans cette formule.
- Une *formule close* est une formule sans variable libre.
- On note $\mathit{Libres}(\phi)$ l'ensemble des variables libres de ϕ.

Exemples :

- dans $\exists x \cdot p(x, y)$, l'occurrence de x est liée et celle de y est libre
- dans $r(x, z) \rightarrow \forall z \cdot (r(y, z) \vee y = z)$, la première occurrence de z est libre et les deux suivantes sont liées

On note souvent $\phi(x_1, \ldots, x_n)$ pour indiquer que ϕ possède exactement $x_1, \ldots, x_n$ comme variables libres.

17 Interprétation des formules

- On va interpréter les formules dans des structures.
- Une structure $\mathcal{M}$ pour un langage $\mathcal{L}$ se compose d'un ensemble non vide M, appelé le *domaine* et d'une interprétation des symboles de prédicats par des relations sur M, des symboles de fonctions par des fonctions de M, et des constantes par des éléments de M.
- Plus précisément, une structure se aussi composée :
  - d'un sous-ensemble de M^n, noté $r^{\mathcal{M}}$, pour chaque symbole de prédicat r d'arité n dans $\mathcal{L}$;
  - d'une fonction **totale** de M^m dans M, notée $f^{\mathcal{M}}$, pour chaque symbole de fonction f d'arité m dans $\mathcal{L}$;
  - d'un élément de M, noté $c^{\mathcal{M}}$, pour chaque symbole de constante c dans $\mathcal{L}$.

18 Structures et langages : Exemples

- Pour $\mathcal{L}_1 = (r|_1, c)$, la structure $\mathcal{M}_1 = (\mathbb{N}, r^{\mathcal{M}_1}, c^{\mathcal{M}_1})$ avec $r^{\mathcal{M}_1}$ l'ensemble des nombres premiers et $c^{\mathcal{M}_1} = 2$ est une interprétation de $\mathcal{L}_1$.

18 Structures et langages : Exemples

- Pour $\mathcal{L}_1 = (r|_1, c)$, la structure $\mathcal{M}_1 = (\mathbb{N}, r^{\mathcal{M}_1}, c^{\mathcal{M}_1})$ avec $r^{\mathcal{M}_1}$ l'ensemble des nombres premiers et $c^{\mathcal{M}_1} = 2$ est une interprétation de $\mathcal{L}_1$.
- Parfois, on écrira directement les interprétations dans le n-uplet comme dans l'exemple suivant :
- Pour $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$, on peut prendre la structure sur les réels

$$\mathcal{M}_2 = (\mathbb{R}, \leq, +1, +, \times, 0, 1)$$

avec $+1$ la fonction qui a x associe $x + 1$

19 Interprétation des termes dans une structure

Etant donnés un ensemble de variables $\mathcal{V}$ et un domaine M, une *valuation* pour les variables de $\mathcal{V}$ dans M est une fonction $v : \mathcal{V} \to M$ qui attribue à chaque variable $x \in \mathcal{V}$, une valeur $v(x) \in M$.

L'interprétation d'un terme t (dont les variables sont dans $\mathcal{V}$) dans une structure de domaine M et selon une valuation v est un élément $t^{\mathcal{M},v} \in M$, inductivement défini de la façon suivante :

- si $t = c$, alors $t^{\mathcal{M},v} = c^{\mathcal{M}}$;
- si $t = x$, alors $t^{\mathcal{M},v} = v(x)$ est $v(x)$;
- si $t = f(t_1, t_2, \ldots, t_n)$ alors $t^{\mathcal{M},v} = f^{\mathcal{M}}(t_1^{\mathcal{M},v}, \ldots, t_n^{\mathcal{M},v})$;

20 Exemples

Soit $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$ et $\mathcal{M}_3 = (\mathbb{N}, \leq, +1, +, \times, 0, 1)$.
L'interprétation dans $\mathcal{M}_3$ du terme

$$t_1 \equiv g(y, h(c, x))$$

selon la valuation v telle que $v(x) = 3$, $v(y) = 4$, $v(z) = 6$ est :

20 Exemples

Soit $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$ et $\mathcal{M}_3 = (\mathbb{N}, \leq, +1, +, \times, 0, 1)$.
L'interprétation dans $\mathcal{M}_3$ du terme

$$t_1 \equiv g(y, h(c, x))$$

selon la valuation v telle que $v(x) = 3$, $v(y) = 4$, $v(z) = 6$ est :

$$t_1^{\mathcal{M}_3, v} = 4 + (0 \times 3) = 4$$

L'interprétation du terme

$$t_2 \equiv f(g(d, h(y, z)))$$

est :

20 Exemples

Soit $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$ et $\mathcal{M}_3 = (\mathbb{N}, \leq, +1, +, \times, 0, 1)$.
L'interprétation dans $\mathcal{M}_3$ du terme

$$t_1 \equiv g(y, h(c, x))$$

selon la valuation v telle que $v(x) = 3$, $v(y) = 4$, $v(z) = 6$ est :

$$t_1^{\mathcal{M}_3, v} = 4 + (0 \times 3) = 4$$

L'interprétation du terme

$$t_2 \equiv f(g(d, h(y, z)))$$

est :

$$t_2^{\mathcal{M}_3, v} = (1 + (4 \times 6)) + 1 = 26$$

21 Interprétation des formules

Une formule ϕ construite sur un langage $\mathcal{L}$ est satisfaite dans une structure $\mathcal{M}$ et pour une valuation v donnant une valeur aux variables de l'ensemble $\mathcal{V}$, noté $\mathcal{M}, v \models \phi$, si et seulement si :

- si $\phi \equiv r(t_1, t_2, \ldots, t_n)$ et $t_i^{\mathcal{M},v} = b_i$ pour $i = 1, 2, \ldots, n$, alors ϕ est vraie ssi $(b_1, b_2, \ldots, b_n) \in r^{\mathcal{M}}$;

22 Interprétation des formules

- si $\phi \equiv \neg\psi_1$, $\phi \equiv \psi_1 \vee \psi_2$, $\phi \equiv \psi_1 \wedge \psi_2$, $\phi \equiv \psi_1 \rightarrow \psi_2$, $\phi \equiv \psi_1 \leftrightarrow \psi_2$ alors la valeur de ϕ est calculée à partir des valeurs de ψ_1 et ψ_2 comme dans le cas propositionnel ;
- si $\phi \equiv \exists x \cdot \psi$, alors ϕ est vraie ssi **il existe** une valuation v' telle que $\mathcal{M}, v' \models \psi$ et v' est d'accord [1] avec v sur $\mathit{Libres}(\phi)$.

1. Cela veut dire que $v'(x) = v(x)$ pour tout $x \in \mathit{Libres}(\phi)$

22 Interprétation des formules

- si $\phi \equiv \neg\psi_1$, $\phi \equiv \psi_1 \vee \psi_2$, $\phi \equiv \psi_1 \wedge \psi_2$, $\phi \equiv \psi_1 \rightarrow \psi_2$, $\phi \equiv \psi_1 \leftrightarrow \psi_2$ alors la valeur de ϕ est calculée à partir des valeurs de ψ_1 et ψ_2 comme dans le cas propositionnel ;
- si $\phi \equiv \exists x \cdot \psi$, alors ϕ est vraie ssi **il existe** une valuation v' telle que $\mathcal{M}, v' \models \psi$ et v' est d'accord [1] avec v sur $\mathit{Libres}(\phi)$.
- si $\phi \equiv \forall x \cdot \psi$, alors ϕ est vraie ssi **pour toute** valuation v' qui est d'accord avec v sur $\mathit{Libres}(\phi)$, on a $\mathcal{M}, v' \models \psi$.

1. Cela veut dire que $v'(x) = v(x)$ pour tout $x \in \mathit{Libres}(\phi)$

23 Structures et satisfaction des formules

- Lorsque $\mathcal{M}, v \models \phi$, on dit que $\mathcal{M}, v$ satisfait ϕ, ou encore que $(\mathcal{M}, v)$ est un modèle de ϕ
- Lorsque ϕ est une formule close, alors sa valeur de vérité dans un couple $(\mathcal{M}, v)$, ne dépend pas de v. On omettera de mentionner v dans ce cas.

24 Exemples

- Prenons $\mathcal{L}_1 = \{r|_2, c\}$. La formule close suivante

$$\begin{array}{l} \forall x \cdot r(x,x) \\ \wedge \forall x \cdot \forall y \cdot (r(x,y) \rightarrow r(y,x)) \\ \wedge \forall x \cdot \forall y \cdot \forall z \cdot (r(x,y) \wedge r(y,z) \rightarrow r(x,z)) \end{array}$$

  exprime qu'une structure (D, R, a) est un modèle de la formule si et seulement si R est une relation

24 Exemples

- Prenons $\mathcal{L}_1 = \{r|_2, c\}$. La formule close suivante

$$\begin{array}{l} \forall x \cdot r(x,x) \\ \wedge \forall x \cdot \forall y \cdot (r(x,y) \rightarrow r(y,x)) \\ \wedge \forall x \cdot \forall y \cdot \forall z \cdot (r(x,y) \wedge r(y,z) \rightarrow r(x,z)) \end{array}$$

  exprime qu'une structure (D, R, a) est un modèle de la formule si et seulement si R est une relation d'équivalence.
- la formule $\forall y \cdot r(x,y)$ est vraie dans la structure $(\mathbb{N}, \leq, +1, +, \times, 0, 1)$ et la valuation v ssi $v(x) = 0$ (Rappel : r est interprété par $\leq$)

25 Exemples

- est-ce que $\exists x \cdot \forall y \cdot r(x, y)$ est vraie dans $(\mathbb{N}, \leq)$?

"est-ce qu'il existe un entier $x \in \mathbb{N}$ tel que pour tout $y \in \mathbb{N}$, $x \leq y$"

Oui, on prend $x = 0$.

- est-ce qu'elle est vraie dans $(\mathbb{Z}, \leq)$?

Non, il n'y a pas d'élément minimal dans $\mathbb{Z}$

25 Exemples

- est-ce que $\exists x \cdot \forall y \cdot r(x, y)$ est vraie dans $(\mathbb{N}, \leq)$? oui.
- Sur le langage $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$, prenons la formule close

$$\forall x \cdot \forall z \cdot \exists y \cdot (x = c \vee g(h(x, y), z) = c)$$

...

  - est-ce que $(\mathbb{R}, \leq, +1, +, \times, 0, 1)$ en est un modèle ?

La formule est vraie ssi pour tout réel $x \in \mathbb{R}$, tout réel $z \in \mathbb{R}$, il existe $y \in \mathbb{R}$, tel que $x = 0$ ou $xy + z = 0$.
Oui, si $x \neq 0$, alors on prend $y = -\frac{z}{x}$

25 Exemples

- est-ce que $\exists x \cdot \forall y \cdot r(x, y)$ est vraie dans $(\mathbb{N}, \leq)$? oui.
- Sur le langage $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$, prenons la formule close

$$\forall x \cdot \forall z \cdot \exists y \cdot (x = c \vee g(h(x, y), z) = c)$$

  - est-ce que $(\mathbb{R}, \leq, +1, +, \times, 0, 1)$ en est un modèle ? oui.
  - est-ce que $(\mathbb{N}, \leq, +1, +, \times, 0, 1)$ en est un modèle ?

25 Exemples

- est-ce que $\exists x \cdot \forall y \cdot r(x, y)$ est vraie dans $(\mathbb{N}, \leq)$? oui.
- Sur le langage $\mathcal{L}_2 = (r|_2, f|_1, g|_2, h|_2, c, d)$, prenons la formule close

$$\forall x \cdot \forall z \cdot \exists y \cdot (x = c \vee g(h(x, y), z) = c)$$

  - est-ce que $(\mathbb{R}, \leq, +1, +, \times, 0, 1)$ en est un modèle ? oui.
  - est-ce que $(\mathbb{N}, \leq, +1, +, \times, 0, 1)$ en est un modèle ? non.

26 Formules satisfaisables, valides et équivalentes

qui ne contient pas de variables libres.

- une formule ϕ close est **satisfaisable** si elle a un modèle[2]
- elle est **valide** si toutes les structures sont des modèles de ϕ
- deux formules ϕ_1, ϕ_2 telles que $\mathit{Libres}(\phi_1) = \mathit{Libres}(\phi_2)$ sont équivalentes si la formule $\forall x_1 \ldots \forall x_n (\phi_1 \leftrightarrow \phi_2)$ est valide, avec $\{x_1, \ldots, x_n\} = \mathit{Libres}(\phi_1)$.

2. Malheureusement, il a été démontré qu'il n'existe aucun algorithme permettant de tester la satisfaisabilité d'une formule.

27 Formules valides, formules équivalentes

Les couples de formules suivants sont des exemples de formules équivalentes :

- $\forall x \cdot (\phi \wedge \psi)$ et $(\forall x \cdot \phi) \wedge (\forall x \cdot \psi)$
- $\exists x \cdot (\phi \vee \psi)$ et $(\exists x \cdot \phi) \vee (\exists x \cdot \psi)$
- $\neg \forall x \cdot \phi$ et $\exists x \cdot \neg \phi$
- $\neg \exists x \cdot \phi$ et $\forall x \cdot \neg \phi$

Exercice 1 (NON FAIT EN COURS)

Soit un langage $\mathcal{L} = (p, q, r, s, t, f)$ où p, q sont des prédicats unaires, r, s, t sont des prédicats binaires, et f est une fonction unaire. Modélisez en logique des prédicats les propriétés suivantes :

1. le prédicat s contient le produit cartésien de p et q
2. le prédicat t est égal au produit cartésien de q et p
3. La fonction f est surjective

CORRECTION

1. $\forall x \forall y \left(p(x) \wedge q(y) \longrightarrow s(x,y) \right)$
2. $\forall x \forall y \left(q(x) \wedge p(y) \right) \longleftrightarrow t(x,y)$
3. $\forall x \exists y \cdot f(y) = x$

Exercice 2

Soit un langage $\mathcal{L} = \{f\}$ où f est une fonction binaire, et soit une formule ϕ de $\mathcal{L}$ telle que $\phi \equiv \exists y \cdot z = f(x, y)$. La formule ϕ est-elle vraie dans la structure $\mathcal{M}$, en utilisant la valuation v ?

1. $\mathcal{M} = (D \equiv \mathbb{N}, p \equiv \leq, f \equiv +, g \equiv \times)$ et $v \equiv (x \mapsto 5, z \mapsto 3)$
2. $\mathcal{M} = (D \equiv \mathbb{Z}, p \equiv \leq, f \equiv +, g \equiv \times)$ et $v \equiv (x \mapsto 5, z \mapsto 3)$
3. $\mathcal{M} = (D \equiv \mathbb{N}, p \equiv \leq, f \equiv \times, g \equiv +)$ et $v \equiv (x \mapsto 5, z \mapsto 3)$
4. $\mathcal{M} = (D \equiv \mathbb{Z}, p \equiv \leq, f \equiv \times, g \equiv +)$ et $v \equiv (x \mapsto 5, z \mapsto 3)$
5. $\mathcal{M} = (D \equiv \mathbb{Z}_6, p \equiv \leq, f \equiv \times, g \equiv +)$ et $v \equiv (x \mapsto 5, z \mapsto 3)$

1. $\exists y \in \mathbb{N} \cdot 3 = 5 + y$ NON car on est dans $\mathbb{N}$
2. VRAI, il faut prendre $y = -2$
3. $\exists y \in \mathbb{N} \cdot 3 = 5y$ NON
4. NON
5. OUI, il faut prendre $y = 3$ car $3 = 5 \times 3 \bmod 6$.

Énigme d'Einstein : qui possède la poisson rouge ?

Il y a cinq maisons de cinq couleurs différentes, alignées le long d'une route. Dans chacune de ces maisons vit une personne de nationalité différente. Chacune de ces personnes boit une boisson différente, fume une marque de cigare différente et a un animal domestique différent.

1. L'Anglais vit dans la maison rouge.
2. Le Suédois a des chiens.
3. Le Danois boit du thé.
4. La maison verte est directement à gauche de la maison blanche.
5. Le propriétaire de la maison verte boit du café.
6. La personne qui fume des Pall Mall élève des oiseaux.
7. Le propriétaire de la maison jaune fume des Dunhill.
8. La personne qui vit dans la maison du centre boit du lait.
9. Le Norvégien habite dans la première maison en partant de la gauche.
10. L'homme qui fume des Blend vit à côté de celui qui a des chats.
11. L'homme qui a un cheval est le voisin de celui qui fume des Dunhill.
12. Celui qui fume des Bluemaster boit de la bière.
13. L'Allemand fume des Prince.
14. Le Norvégien vit juste à côté de la maison bleue.
15. L'homme qui fume des Blend a un voisin qui boit de l'eau.

Résolution de l'énigme avec un solveur du premier ordre (Paradox)

On va modéliser les indices en logique du premier ordre, et appeler un solveur qui va tester la satisfiabilité de la solution. Un modèle de la formule devra nous donner la réponse.
https://www.tptp.org/cgi-bin/SystemOnTPTP

Langage utilisé

On a :

- 5 nationalités : Anglais, Suédois, Danois, Allemand, Norvégien
- 5 boissons : thé, café, lait, bière, eau
- 5 marques de cigares : Pall Mall, Dunhill, Blend, Bluemaster, Prince
- 5 animaux : oiseaux, chats, chiens, poisson, cheval
- 5 couleurs : rouge, bleue, verte, jaune, blanche

Chaque maison va correspondre à un élément du domaine. On aura un prédicat unaire par caractéristique :

$$\{bleue|_1, verte|_1, \ldots, the|_1, cafe|_1, \ldots, oiseaux|_1, \ldots, pallmall|_1, \ldots\}$$

Contraintes d'unicité

1. On doit exprimer qu'il y a une unique couleur par maison, un unique type d'animal par maison, etc..
2. On doit aussi exprimer qu'il existe au moins une maison de chaque couleur, au moins un animal de chaque type, ...

$$\forall x \, \forall y \left[\text{rouge}(x) \wedge x \neq y \rightarrow \neg \text{rouge}(y) \right]$$

"au plus une maison rouge"

$$\exists x \, \text{rouge}(x)$$

"au moins une maison rouge"

Définition de l'ordre entre les maisons

On va définir :

1. un prédicat $gauche(x, y)$ vrai si et seulement si x est à gauche de y, avec x, y différents (autrement dit, que x "$\leq$" y).
2. Un prédicat $succgauche(x, y)$ vrai si et seulement si y est le successeur direct de x sur sa gauche
3. Un prédicat $succdroit(x, y)$ vrai si et seulement si y est le successeur direct de x sur sa droite
4. Un prédicat $acote(x, y)$ vrai si et seulement si y est un successeur direct de x, à gauche ou à droite.

1. Pour 1, on va exprimer que gauche (x,y) est une relation d'ordre (transitive, réflexive et antisymétrique)

(A.) transitivité

$$\forall x \forall y \forall z \left[\text{gauche}(x,y) \wedge \text{gauche}(y,z) \rightarrow \text{gauche}(x,z) \right]$$

(B.) réflexivité : $\forall x \ \text{gauche}(x,x)$

(C.) antisymétrie : $\forall x \forall y \ (\text{gauche}(x,y) \wedge \text{gauche}(y,x)) \rightarrow x = y$

2. $\forall x \forall y \ \text{succgauche}(x,y) \leftrightarrow \text{gauche}(x,y) \wedge \neg \exists z \ (\text{gauche}(x,z) \wedge \text{gauche}(z,y) \wedge x \neq z \neq y)$

4. $\forall x \forall y \ \text{acote}(x,y) \leftrightarrow (\text{succgauche}(x,y) \vee \text{succdroit}(x,y))$

L'Anglais vit dans la maison rouge

$$\forall x \left[\text{anglais}(x) \rightarrow \text{rouge}(x) \right]$$

Le Suédois a des chiens.

$$\forall x \left[\text{Suédois}(x) \longrightarrow \text{chiens}(x) \right]$$

Le Danois boit du thé.

$$\forall x \quad \text{Danois}(x) \rightarrow \text{the}(x)$$

La maison verte est directement à gauche de la maison blanche.

$$\forall x \forall y \left(\text{succgauche}(x, y) \wedge \text{verte}(x) \right) \rightarrow \text{blanche}(y)$$

Le propriétaire de la maison verte boit du café.

$$\forall x \quad \text{verte}(x) \rightarrow \text{cafe}(x)$$

La personne qui fume des Pall Mall élève des oiseaux.

$$\forall x \quad \text{PallMall}(x) \longrightarrow \text{oiseau}(x)$$

Le propriétaire de la maison jaune fume des Dunhill.

$$\forall x \quad \text{jaune}(x) \rightarrow \text{Dunhill}(x)$$

La personne qui vit dans la maison du centre boit du lait.

On introduit un symbole de constante 'centre'

et les formules :

$$\exists x_1 \exists x_2 \exists x_3 \exists x_4 \; . \; \text{succdroit}(x_1, x_2) \wedge \text{succdroit}(x_2, \text{centre}) \wedge \text{succdroit}(\text{centre}, x_3) \wedge \text{succdroit}(x_3, x_4)$$

puis la formule :

$$\text{lait}(\text{centre})$$

Le Norvégien habite dans la première maison en partant de la gauche.

"pas de maison à gauche de x"

$$\forall x \left(\neg \exists y \ \text{gauche}(y, x) \right) \rightarrow \text{Norvegien}(x)$$

L'homme qui fume des Blend vit à côté de celui qui a des chats.

$$\forall x \left(\text{Blend}(x) \rightarrow \exists y \ \text{acote}(x,y) \wedge \text{chats}(y) \right)$$

L'homme qui a un cheval est le voisin de celui qui fume des Dunhill.

$$\forall x \left(\text{cheval}(x) \longrightarrow \exists y \; \text{acote}(x,y) \wedge \text{Dunhill}(y) \right)$$

Celui qui fume des Bluemaster boit de la bière.

$\forall x \quad \text{biere}(x) \rightarrow \text{Bluemaster}(x)$

L'Allemand fume des Prince.

$\forall x \quad \text{Allemand}(x) \longrightarrow \text{Prince}(x)$

Le Norvégien vit juste à côté de la maison bleue.

$$\forall x \left(\text{Norvegien}(x) \rightarrow \exists y \ \text{acote}(x, y) \wedge \text{bleue}(y) \right)$$

L'homme qui fume des Blend a un voisin qui boit de l'eau.

Questions

En appliquant le solveur Paradox aux formules (données dans le fichier sur l'UV), le solveur retourne une solution où l'Allemand possède le poisson.

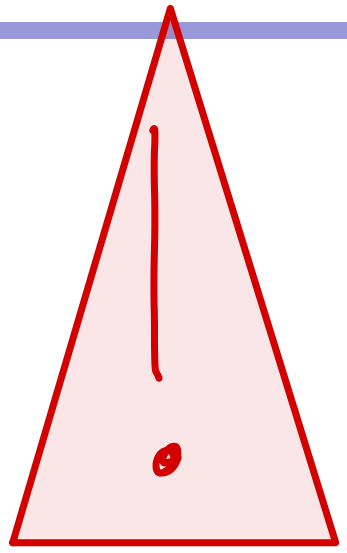
1. Comment tester que la solution est unique ?
2. Comment tester que chaque indice est nécessaire ?

On teste la satisfaisabilité de $\forall x\ \text{Allemand}(x) \rightarrow \text{poisson}(x)$

↳ Paradox retourne que c'est insatisfaisable, donc la solution Allemand / poisson est unique!

On commente l'indice et on teste l'unicité de la solution. Si ce n'est pas unique, alors l'indice est nécessaire. Sinon il ne l'est pas. On se rend compte que le dernier n'est pas *.

51 Indécidabilité de la logique du premier ordre



- Nous allons montrer qu'il n'existe pas d'algorithme A qui prend en entrée une formule de la logique du premier ordre et retourne 1 si elle est valide, 0 sinon.

À PARTIR D'ICI JUSQU'À LA FIN, CETTE PARTIE, NON VUE EN COURS, NE SERA PAS ÉVALUÉE.

52 Problème de la Correspondance de Post (PCP) – Rappel

Soit $\Sigma = \{0, 1\}$. Un mot u sur Σ est une séquence finie d'éléments de Σ. Deux mots u et v peuvent être concaténés pour former un nouveau mot noté uv. L'élément neutre pour la concaténation est noté ϵ (mot vide). Par exemple : $u = 01$ est un mot, $v = 110$ est un mot, $uv = 01110$, $\epsilon u = u\epsilon = u = 01$.

Le problème de la Correspondance de Post (PCP) se formule de la manière suivante :

Entrée

$(u_1, v_1), \ldots, (u_n, v_n)$, $n \geq 1$, n paires de mots (possiblement vides) sur Σ.

Sortie

1 si et seulement si il existe une séquence finie d'indices $i_1, \ldots, i_k \in \{1, \ldots, n\}$ telle que $k \geq 1$ et

$$u_{i_1} u_{i_2} \ldots u_{i_k} = v_{i_1} v_{i_2} \ldots v_{i_k}$$

53 PCP : exemples

- Instance : $(u_1, v_1) = (100, 00),\ (u_2, v_2) = (0, 01)$,

53 PCP : exemples

- Instance : $(u_1, v_1) = (100, 00),\ (u_2, v_2) = (0, 01)$, Solution (parmi d'autres) : 2,1

$$u_2 u_1 = 0100 = v_2 v_1$$

- Instance : $(u_1, v_1) = (1, 100),\ (u_2, v_2) = (0, \epsilon)$,

53 PCP : exemples

- Instance : $(u_1, v_1) = (100, 00),\ (u_2, v_2) = (0, 01)$, Solution (parmi d'autres) : 2,1

$$u_2 u_1 = 0100 = v_2 v_1$$

- Instance : $(u_1, v_1) = (1, 100),\ (u_2, v_2) = (0, \epsilon)$, Solution : 1,2,2

$$u_1 u_2 u_2 = 100 = v_1 v_2 v_2$$

- Instance : $(u_1, v_1) = (1, 0),\ (u_2, v_2) = (0, 1)$,

53 PCP : exemples

- Instance : $(u_1, v_1) = (100, 00),\ (u_2, v_2) = (0, 01)$, Solution (parmi d'autres) : 2,1

$$u_2 u_1 = 0100 = v_2 v_1$$

- Instance : $(u_1, v_1) = (1, 100),\ (u_2, v_2) = (0, \epsilon)$, Solution : 1,2,2

$$u_1 u_2 u_2 = 100 = v_1 v_2 v_2$$

- Instance : $(u_1, v_1) = (1, 0),\ (u_2, v_2) = (0, 1)$, Pas de solution
- Instance : $(u_1, v_1) = (0, 100),\ (u_2, v_2) = (01, 00),\ (u_3, v_3) = (110, 11)$?

53 PCP : exemples

- Instance : $(u_1, v_1) = (100, 00),\ (u_2, v_2) = (0, 01)$, Solution (parmi d'autres) : 2,1

$$u_2 u_1 = 0100 = v_2 v_1$$

- Instance : $(u_1, v_1) = (1, 100),\ (u_2, v_2) = (0, \epsilon)$, Solution : 1,2,2

$$u_1 u_2 u_2 = 100 = v_1 v_2 v_2$$

- Instance : $(u_1, v_1) = (1, 0),\ (u_2, v_2) = (0, 1)$, Pas de solution
- Instance :
  $(u_1, v_1) = (0, 100),\ (u_2, v_2) = (01, 00),\ (u_3, v_3) = (110, 11)$?

$$u_3 u_2 u_3 u_1 = (110)(01)(110)(0) = 110011100 = (11)(00)(11)(100) = v_3 v_2 v_3 v_1$$

54 Théorèmes

Théorème

Le problème de Correspondance de Post est indécidable.

Preuve admise.

Théorème

Le problème de validité en logique du premier ordre est indécidable.

Nous allons montrer ce résultat en réduisant PCP.

55 Preuve

Nous considérons le langage du premier ordre L contenant un prédicat binaire p (et l'égalité $=$), deux symboles de fonction unaires f_0 et f_1, et un symbole de constante a. Etant donné un mot u sur Σ et un terme g, on définit le terme $t_u(g)$ inductivement par :

$$t_\epsilon(g) = g \qquad t_{bv}(g) = f_b(t_v(g)) \text{ pour } b \in \{0,1\},\ v \in \{0,1\}^*$$

Remarquez que pour tout $u, v \in \{0,1\}^*$ et tout terme g, on a $t_{uv}(g) = t_u(t_v(g))$, et $t_u(g) = t_v(g)$ ssi $u = v$.

Soit $I = \{(u_1, v_1), \ldots, (u_n, v_n)\}$ une instance de PCP. Nous allons construire une formule ϕ_I de la logique du premier ordre, sur le langage L, et montrer que ϕ_I est valide si et seulement si I a une solution. La formule ϕ_I se décompose comme suit :

$$\phi_I = (\rho \wedge \sigma) \rightarrow \tau$$

Premièrement,

$$\rho \equiv p(t_{u_1}(a), t_{v_1}(a)) \wedge p(t_{u_2}(a), t_{v_2}(a)) \wedge \cdots \wedge p(t_{u_n}(a), t_{v_n}(a))$$

56 Preuve

Cela signifie qu'on met dans la relation p les termes qui correspondent aux paires de mots. On peut étendre cette relation aux concaténations (paires à paires) de couples de mots de I. En particulier, la formule suivante signifie que si x et y sont en relation, alors on peut concaténer u_i à x et v_j à y, pourvu que $i = j$.

$$\sigma \equiv \forall x \forall y.(p(x, y) \rightarrow \bigwedge_{i=1}^{n} p(t_{u_i}(x), t_{v_i}(y)))$$

Enfin, τ signifie l'existence de deux éléments égaux en relation :

$$\tau \equiv \exists z.p(z, z) \wedge z \neq a$$

Clairement, cette réduction est effective (on peut écrire un algorithme qui l'implémente). Montrons qu'elle est correcte.

57 Preuve

Supposons que I a une solution $i_1, \ldots, i_k$ et montrons que ϕ_I est valide. Pour cela, soit $\mathcal{M}$ une structure sur L. Supposons que $\mathcal{M} \models \rho \wedge \sigma$. On sait par hypothèse que

$$u_{i_1} \ldots u_{i_k} = v_{i_1} \ldots v_{i_k}$$

Donc $t_{u_{i_1} \ldots u_{i_k}}(a) = t_{v_{i_1} \ldots v_{i_k}}(a)$, dont on déduit

$$t_{u_{i_1}}(t_{u_{i_2}}(\ldots(t_{u_{i_k}}(a))) = t_{v_{i_1}}(t_{v_{i_2}}(\ldots(t_{v_{i_k}}(a)))$$

Notons g ce terme. Par hypothèse, on sait que $\mathcal{M} \models p(t_{u_{i_k}}, t_{v_{i_k}})$ et comme $\mathcal{M} \models \sigma$, on obtient aussi que
$\mathcal{M} \models p(t_{u_{i_{k-1}}}(t_{u_{i_k}}(a)), t_{v_{i_{k-1}}}(t_{v_{i_k}})(a))$, et plus généralement, que
$\mathcal{M} \models p(g, g)$. Donc $\mathcal{M} \models \exists z.p(z, z)$. De plus, on peut supposer $z \neq a$ car $k \geq 1$. Donc $\mathcal{M} \models \exists z.p(z, z) \wedge z \neq a$.

58 Preuve

Réciproquement, supposons que ϕ_I est valide et montrons que I a une solution. On définit une structure H comme ceci :

- son domaine est l'ensemble des termes clos sur L
- les fonctions et constantes sont interprétées par elles-mêmes ($f^H(t) = f(t)$)
- l'interprétation p^H de p est définie inductivement : $p^H(a, a) = 1$ et pour tous termes clos h et g, $p^H(t_{u_i}(g), t_{v_j}(h)) = 1$ si et seulement si $i = j$ et $p^H(g, h) = 1$.

Clairement, $H \models \rho \wedge \sigma$, donc $H \models \tau$ puisque ϕ_I est valide. Donc il existe un terme clos t tel que $H \models p(t, t) \wedge t \neq a$. Par définition de H, il est facile de voir que t se décompose nécessairement en $t = t_{u_{i_1}}(t_{u_{i_2}}(\ldots(t_{u_{i_k}}(a))) = t_{v_{i_1}}(t_{v_{i_2}}(\ldots(t_{v_{i_k}}(a)))$. Donc

$$u_{i_1} \ldots u_{i_k} = v_{i_1} \ldots v_{i_k}$$

et I a une solution. □