1 De la logique des propositions à la logique du premier ordre

1.1 Introduction

Logique = science du raisonnement
Raisonnement = à partir d'hypothèses déduire / obtenir une conclusion

Exemple

Un vol a été commis. A, B et C ont été appréhendés. Les faits suivants sont affirmés par la police

1. Nul autre que A, B ou C ne peut être impliqué
2. A ne travaille jamais sans un complice
3. C est innocent

Put-on trouver le(s) coupable(s), c'est à dire : chacun des raisonnements : de 1, 2 et 3 on peut (respectivement) A (resp. B resp. C) est coupable (resp. innocent) est-il correct ?

— conclure *C est innocent* est un raisonnement correct (et C coupable incorrect)
— on ne peut rien dire sur A : tout raisonnement qui conclue sur A est incorrect
— sur B :
deux cas, A innocent ou pas , dans les deux cas B est coupable
Formaliser

1. Nul autre que A, B ou C ne peut être impliqué $A \vee B \vee C$ (H_1)
2. A ne travaille jamais sans un complice $A \rightarrow B \vee C$ (H_2)
3. C est innocent $\neg C$ (H_3)
4. $H_1 \wedge H_2 \wedge H_3 \models A \wedge \neg C$

Remarque on ne s'intéresse pas à la véracité des hypothèses

1.2 la logique des propositions

1.2.1 les fbf

définition inductive de la syntaxe autorisée
arborescence liée à une formule logique
formalisation de l'exemple.

1.2.2 sémantique

interprétation

- intuition : un monde possible
- compréhension fruste : table de vérité
  Table de vérité de l'exemple : 2^3 lignes nom ses colonnes
  | A | B | C | H_1 | H_2 | H_3 | Conclusion |
- définition mathématique : application de S (symbole propositionnels) dans B (booléens)

Définition inductive de la valeur de vérité d'une formule pour une interprétation donnée

1.2.3 Vocabulaire

valide, contingent, insatisfiable

1.2.4 conséquence logique $\models$

validité d'un raisonnement
définition ; sur l'exemple et la table de vérité

1.2.5 méthodes de calcul

1. toutes les interprétations (table de vérité)
2. théorème fondamental et manipulations syntaxique
3. résolution
4. méthode des tableaux

1.3 Les problèmes d'expressivité de la logique des propositions

1.3.1 Exemple introductif

La logique des propositions a une expressivité limitée (par rapport par exemple au langage naturel)
Exemple :

- $A =_{def}$ Tout étudiant possède un bac
- $B =_{def}$ Pierre est un étudiant
- $C =_{def}$ Pierre possède un bac
- Comment avoir $\{A, B\} \models C$

1.3.2 De la logique des propositions simples ...

1. modifier l'ennoncé à modéliser :
   $A_{Pierre} =_{def}$ Si Pierre est un étudiant alors Pierre possède un bac
   On obtient bien $\{A_{Pierre}, B\} \models C$
2. Conséquence :

   Idée 1 : Introduction d'un domaine : *l'ensemble des objets du monde considéré*
   - Domaine : l'ensemble des personnes : $\{p_1, p_2, \ldots \text{ Pierre } \ldots\}$
   - La modélisation de A devient :
     - $A_1 =_{def}$ Si p_1 est un étudiant alors p_1 possède un bac
     - $A_2 =_{def}$ Si p_2 est un étudiant alors p_2 possède un bac
     - ...
3. Problèmes :
   - Infinité potentielle du domaine
   - Accroissement inutile du nombre de symboles propositionnels
   - Comment représenter *il y a au moins un étudiant* ?

1.3.3 ... aux propositions paramétrées : les symbole propositionnel d'arité donnée ...

Idée 2 : utiliser des énoncés paramétrés par des variables dont les valeurs seront prises dans le domaine

Penons un autre exemple :
Chercons à modéliser

- François n'est pas coupable
- François est le pote à Emile
- Emile est le pote à Denis
- Denis est le pote à Charles
- Charles est le pote à Bernard
- Bernard est le pote à Albert
- le pote à un coupable est coupable
- Donc Albert n'est pas coupable

modélisation en logique des propositions

modélisation pure on doit modifier l'énoncé à modeliser

- $\neg C_F$
- $pote_{FG} \ \wedge \ pote_{EF} \ \wedge \ pote_{DE} \ \wedge \ pote_{CD} \ \wedge \ pote_{AB}$
- $A_{coup} \wedge pote_{AB} \rightarrow B_{coup}$
- $B_{coup} \wedge pote_{BC} \rightarrow C_{coup}$
- $C_{coup} \wedge pote_{CD} \rightarrow D_{coup}$
- $D_{coup} \wedge pote_{DE} \rightarrow E_{coup}$
- $E_{coup} \wedge pote_{EF} \rightarrow F_{coup}$
- $F_{coup} \wedge pote_{FG} \rightarrow G_{coup}$
- $\models \neg A_{coup}$

modélisation paramétrée

- $\neg Coup(F)$
- $Pote(F,E) \ \wedge \ Pote(E,D) \ \wedge \ Pote(D,C) \ \wedge \ Pote(C,B) \ \wedge \ Pote(B,A)$
- $Coup(A) \wedge Pote(A,B) \rightarrow Coup(B)$
- $Coup(B) \wedge Pote(B,C) \rightarrow Coup(C)$
- $Coup(C) \wedge Pote(C,D) \rightarrow Coup(D)$
- $Coup(D) \wedge Pote(D,E) \rightarrow Coup(E)$
- $Coup(E) \wedge Pote(E,F) \rightarrow Coup(F)$
- $\models \neg Coup(A)$

comparaison maintenant *Coup* est un symbole propositionnel d'arité 1 et *Pote* est un symbole propositionnel d'arité 2.

1.3.4 ... avec des quantificateurs ...

Idée 3 : utiliser des quantificateurs précisant les valeurs qui peuvent être prises par les variables

- Tous les éléments :
  - Quantification universelle : $\forall$
  - Exprimant : pour tout, quelque soit, chaque ...
- Au moins un élément :
  - Quantification Existentielle : $\exists$
  - Exprimant : il y a, certains, quelque ...
- Exemple :
  - $A =_{def} \forall x \ (EtreUnEtudiant(x) \rightarrow AvoirUnBac(x))$
  - $B =_{def} EtreUnEtudiant(Pierre)$
  - $C =_{def} AvoirUnBac(Pierre)$
  - On a bien $\{A, B\} \models C$

1.4 Prédicat vs. Propositions

- La logique des propositions veut représenter, *interpréter le monde* par des faits
  On dispose *dans le langage* donc de symboles propositionnels $pote_{AB}$, des atomes.
  Mais s'il est possible de modéliser à la main un problème qui nécessite peu d'atomes, ou de modéliser automatiquement des relations de dependance entre les différentes fonction d'un logiciel (la fonction f_1 appelle f_2) on ne peut pas y arriver dés que l'ensemble est infini (tout nombre entier à un successeur)
- La logique des prédicats veut représenter, interpréter le monde par
  - des objets : les éléments du domaine p_1, p_2 Pierre ...
  - des propriétés sur ces objets et relations entre ces objets : EtreUnEtudiant, PossséderUnBac ... Pair, <, EtreLePgcdDe ...
    Ces propriétés et ces relations sont appelées des *prédicats* car ce sont des fonctions dont l'image est un booléen.
  - et des fonctions entre ces objets : PèreDe ... Succ, Carré, + ... (on n'étudiera pas cet aspect dans la première partie)

  On va pour cela disposer dans le langage
  - des *constantes* qui représentent, dénotent des objets déterminés (une constante donnée ne peut représenter qu'un seul objet, mais deux constantes différentes peuvent représenter le même objet) : p_{jean} $v_{montpellier}$
  - des *variables* qui représentent, dénotent des objets non précisés (une variable peut représenter plusieurs objets) : x
  - des *symboles de prédicat* qui représentent, dénotent
    - des propriétés des objets et qui s'appliquent à des variables ou des constantes : $pair(n_2)$, $EtreUnEtudiant(x)$
    - des relations entre les objets qui s'appliquent à des listes ordonnées de variables ets constantes : $EstLaSommeDe(x, n_2, n_3)$
  - des quantificateurs $\forall$, $\exists$ qui s'appliquent à des variables $\forall\ x$, $\exists\ y$

1.5 Exemples de modélisation en logique des prédicats

On veut modéliser le problème suivant, extrait du livre de R. Smullyan "le livre qui rend fou" :

100 hommes politiques se réunissent pour constituer un nouveau parti. Sachant que

1. *parmi eux il y a au moins un homme honnête*
2. *chacun d'eux est soit un homme honnête, soit une franche canaille*
3. *chaque fois qu'on prend un couple de ces hommes, un au moins est malhonnête*

combien d'entre eux sont honnêtes et combien sont des canailles ?

en logique des propositions :

H_i sera la proposition : l'homme politique numéro i est honnête

On va s'intéresser à ce problème avec seulement 3 hommes politiques.
Nos atomes sont alors H_1, H_2 et H_3 et nos hypothèses :

— $\mathcal{H}_a$: $H_1 \vee H_2 \vee H_3$
— $\mathcal{H}_b$: $\neg H_1 \vee \neg H_2$
— $\mathcal{H}_c$: $\neg H_1 \vee \neg H_3$
— $\mathcal{H}_d$: $\neg H_2 \vee \neg H_3$

Si on rajoute un homme politique (c'est à dire si on passe de 3 à 4) il faut modifier l'hypothèse $\mathcal{H}_1$ (car il se peut que ce dernier homme politique soit honnête et que ce soit le seul) mais combien faut-il rajouter d'hypothèses nouvelles ?

Comment augmente le nombre d'interprétations lors de ce rajout ?

en logique des prédicats

on utilisera les prédicats

— $HP(x)$
— $Hnt(x)$
— $\neq (x, y)$

et la constante h_h (qui représentera l'homme politique honnête).
La partie la plus intéressante de la modélisation ne dépend pas du nombre d'hommes politique :

— $\mathcal{H}_1$: $HP(h_h) \wedge Hnt(h_h)$
— $\forall x : HP(x) \to Hnt(x) \vee \neg Hnt(x)$ sans intérêt
— $\mathcal{H}_2$: $\forall x \ \forall y$: $HP(x) \wedge HP(y) \wedge \neq (x, y) \to \neg Hnt(x) \vee \neg Hnt(y)$

Pour représenter les 100 hommes politiques, on n'y gagne cependant pas grand chose, car il faudra une constante h_i pour chaque homme politique et il faudra indiquer que chacune des constantes est différente de toutes les autres.
Et on ne peut **pas** écrire $\forall i,\ j \in \{1 \ldots 100\}\ i \neq j \rightarrow h_i \neq h_j$
Mais indépendament de ça, comment arriver à la conclusion qui nous interesse : ?
$\forall x\ :\ HP(x) \wedge \neq (x, h_h) \rightarrow \neg Hnt(x)$

Les mécanismes de calcul de la logique des prédicats.
il nous suffira d'un mécanisme (l'instantiation) qui nous permette de dire que dans $\mathcal{H}_2$ "on a le droit de remplacer" y par h_h.
On aura alors $\forall x\ :\ HP(x) \wedge HP(h_h) \wedge \neq (x, h_h) \rightarrow \neg Hnt(x) \vee \neg Hnt(h_h)$
et en appliquant un mécanisme de résolution semblable à celui déjà vu en logique des propositions on obtiendra
$\forall x\ :\ HP(x) \wedge \neq (x, h_h) \rightarrow \neg Hnt(x)$

Remarque
si on n'avait pas triché en utilisant la constante h_h mais qu'on avait pris pour $\mathcal{H}_1\ :\ \exists x\ :\ HP(x) \wedge Hnt(x)$, il aurait fallu un autre mécanisme (la skolémisation) qui nous permette de "créer" la constante h_h.

1.6 un problème classique de logique des prédicats, la différence entre

- $\mathcal{F}_1$ $\forall x \exists y P'(x) \wedge P'(y) \rightarrow R(x, y)$ et
- $\mathcal{F}_2$ $\exists y \forall x P'(x) \wedge P'(y) \rightarrow R(x, y)$

Si on peut choisir comme interprétation pour $P'(x)$:

1. $\mathcal{I}_1$: x est un entier positif ou nul, ou
2. $\mathcal{I}_2$: x est un réèl

et que pour $R(x, y)$ on choisit $x \leq y$.

$\mathcal{F}_1$ est vrai pour les deux interprétations, $\mathcal{F}_2$ n'est vraie que pour $\mathcal{I}_1$ (0 est plus petit que tous les autres nombres).
Il faudra qu'un mécanisme de logique (toujours la skolémisation) explicite cette différence.

1.7 l'introduction des fonctions en deuxième partie

pour l'instant, si on veut formaliser la conjecture de Goldbach, il faut écrire
$\forall x\ Pair(x) \rightarrow \exists y \exists z\ :\ Premier(y) \wedge Premier(z) \wedge EstLaSomme(x, y, z)$
Pour écrire $\forall x\ Pair(x) \rightarrow \exists y\ :\ Premier(y) \wedge Premier(y - x)$ il faut avoir la fonction $Moins$ et écrire
$\forall x\ Pair(x) \rightarrow \exists y\ :\ Premier(y) \wedge Premier(Moins(y, x))$ mais ce qui apparaît comme une simple commodité d'écriture (l'introduction de fonction) pose en réalité de sérieux problèmes théoriques, car on introduit des symboles de fonction qu'il faudra aussi interpréter.