Langage logique

Chap. 5

Plan

- Logique des propositions
- Logique des prédicats
- Règles de production

Calcul propositionnel

- Comment écrire les formules ?
  - *Aspects syntaxiques*
- Comment déterminer la valeur de vérité d'une formule ?
  - *Aspects sémantiques*
- Comment démontrer de nouveaux résultats ?
  - *Aspects déductifs*

Syntaxe d'une formule

- Données
  - *un ensemble **P** de variables propositionnelles*
    P = { p, q, r, … } *énoncés élémentaires*
  - *un ensemble **C** de connecteurs*
    $C = \{ \neg, \wedge, \vee, \rightarrow, \leftrightarrow \}$
- Formules
  - ***p*** *est une formule si* $p \in P$
  - $\neg$ ***(H)*** *est une formule si **H** est une formule*
  - ***(H)*** Δ ***(K)*** *est une formule si **H** et **K** sont des formules et si* $\Delta \in C$
- Règles de suppression des parenthèses

Sémantique d'une formule

- Logique bi-valuée
  - *faux (**0**)*
  - *vrai (**1**)*
- Notion d'**interprétation**
  - *donner une valeur de vérité à une variable*

$$\delta\,(p) \in \{\,0\,,1\,\}$$

  - *extension à un opérateur*
  - *extension à une formule*

5

Tables de vérité : opérateurs

p	$\neg p$
0	1
1	0

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\rightarrow$	0	1
0	1	1
1	0	1

$\leftrightarrow$	0	1
0	1	0
1	0	1

6

Formules particulières

- **Tautologies** : formules toujours vraies
  - *La table de vérité ne contient que des 1*
  - *exemple :* $p \vee \neg p$

p	$(\neg p)$	$(p \vee \neg p)$
0	1	1
1	0	1

$\vee$	0	1
0	0	1
1	1	1

7

$\vdash$ **?** $\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$

p	q	$\neg$ p	$\neg$ q	p $\wedge$ q	$\neg$ (p $\wedge$ q)	($\neg$ p $\vee$ $\neg$ q)	F
0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	1	0	1	1	1
1	1	0	0	1	0	0	1

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\leftrightarrow$	0	1
0	1	0
1	0	1

8

Catégories de formules

- **Tautologies** : formules toujours vraies

- **Formules inconsistantes**
  - *formules toujours fausses*
  - *la table de vérité ne contient que des* 0
  - exemple : $p \wedge \neg p$

- **Formules consistantes**
  - *formules non toujours fausses*

$$\exists \delta, \quad \delta(F) = 1$$

9

Formules particulières

- **Formules tautologiquement équivalentes**
  - *les tables de vérité sont les mêmes*

$$\forall \delta, \quad \delta(F) = \delta(H)$$

  - *Condition nécessaire et suffisante :*

    (F) $\leftrightarrow$ (H) est une tautologie $\vdash$ *(F) $\leftrightarrow$ (H)*

10

Quelques équivalences utiles

$F \leftrightarrow G \quad =_{tauto} \quad (F \rightarrow G) \wedge (G \rightarrow F)$

$F \vee G \quad =_{tauto} \quad G \vee F$

$F \vee \neg F \quad =_{tauto} \quad 1$

$\neg(\neg F) \quad =_{tauto} \quad F$

$F \rightarrow G \quad =_{tauto} \quad \neg F \vee G$

$F \wedge G \quad =_{tauto} \quad G \wedge F$

$F \wedge \neg F \quad =_{tauto} \quad 0$

$\neg(F \vee G) \quad =_{tauto} \quad \neg F \wedge \neg G$

$\neg(F \wedge G) \quad =_{tauto} \quad \neg F \vee \neg G$

lois de **De Morgan**

- **Propriétés** de $\vee$ et $\wedge$
  - *associativité*
  - *distributivité (dans les 2 sens)*
  - *éléments neutres (**0** pour $\vee$ et **1** pour $\wedge$)*
  - *éléments absorbants (**1** pour $\vee$ et **0** pour $\wedge$)*

11

Formes normales

- **But** avoir une représentation uniforme des formules du calcul propositionnel
  - ***limiter*** *le nombre de connecteurs différents utilisés*
  - ***limiter*** *l'allure des formules rencontrées*

12

Formes normales

Une formule F est dite sous **forme normale disjonctive** ssi F est une *disjonction de conjonctions* de variables propositionnelles et de leur négation $\bigvee \bigwedge$

Toute formule du calcul propositionnel est tautologiquement équivalente à une formule sous forme normale disjonctive

Une formule F est dite sous **forme normale conjonctive** ssi F est une *conjonction de disjonctions* de variables propositionnelles et de leur négation $\bigwedge \bigvee$

Toute formule du calcul propositionnel est tautologiquement équivalente à une formule sous forme normale conjonctive

Aspects déductifs

- notion de conséquence logique
- notion de démonstration
- notion de règles de déduction

Conséquence logique

- Soit $A = \{F_1, \ldots, F_n\}$ un ensemble de n formules

$$A \vdash G \text{ ssi } \vdash (F_1 \wedge \ldots \wedge F_n) \rightarrow G$$

- Notion de **réfutation**
  - *démonstration par l'absurde*

$$A \vdash G \text{ ssi } \quad F_1 \wedge \ldots \wedge F_n \wedge \neg G \quad \text{est inconsistante}$$

15

Quelques règles de déduction classiques

- **modus ponens**
  - $p,\ p \rightarrow q \vdash q$
- **modus tollens**
  - $p \rightarrow q,\ \neg q \vdash \neg p$
- **syllogisme**
  - $p \rightarrow q,\ q \rightarrow r \vdash p \rightarrow r$

16

Une autre règle d'inférence

- Principe de **résolution** (Robinson)
  - *Définitions*
    - littéral **positif** ou **négatif**
    - une **clause** est une disjonction de littéraux
    - la **résolvante** de $C_1 = l \vee C'_1$ et de $C_2 = \neg l \vee C'_2$ est

$$\boxed{C'_1 \vee C'_2}$$

  - *Principe de résolution*

$$\boxed{l \vee C'_1 , \neg l \vee C'_2 \vdash_{réso} C'_1 \vee C'_2}$$

    - *le principe de résolution est **valide***
    - *le principe de résolution généralise les autres règles*

17

Validité du principe de résolution

- Il faut montrer que :

$$l \vee C'_1 , \neg l \vee C'_2 \vdash C'_1 \vee C'_2$$

ie $(l \vee C'_1) \wedge (\neg l \vee C'_2) \rightarrow (C'_1 \vee C'_2)$

- Il suffit de montrer que
  si $(l \vee C'_1) \wedge (\neg l \vee C'_2)$ **vrai** alors $(C'_1 \vee C'_2)$ n'est **pas faux**

- Deux cas se présentent
  - *l est vrai*
    - nécessairement C'_2 vrai et donc $(C'_1 \vee C'_2)$ aussi
  - *$\neg l$ est vrai*
    - nécessairement C'_1 vrai et donc $(C'_1 \vee C'_2)$ aussi

18

Propriétés du calcul propositionnel

- Le calcul propositionnel muni du principe de résolution est **correct** et **complet**
- Un ensemble **S** de clauses est **insatisfaisable** ssi

  $\mathbf{S} \vdash_{reso}$

- Démonstration : par l'absurde (**réfutation**)

  $\mathbf{S} \vdash \mathbf{C}$ ssi $\mathbf{S} \wedge \{\neg \mathbf{C}\} \vdash_{reso}$

Rappel	On peut toujours se ramener à une forme normale conjonctive (forme clausale)

19

Ce qu'il faut retenir

- Intérêt d'une **forme normale**
- **Conséquence logique** vs **démonstration**
- Principe de **résolution**
- Preuve par **réfutation**

20

Une énigme policière

- Un meurtre a été commis au laboratoire, le corps se trouve dans la salle de conférences…
- On dispose des informations suivantes :
  - *La secrétaire déclare qu'elle a vu l'ingénieur dans le couloir qui donne sur la salle de conférences*
  - *Le coup de feu a été tiré dans la salle de conférences, on l'a donc entendu de toutes les pièces voisines*
  - *L'ingénieur affirme n'avoir rien entendu*
- On souhaite démontrer que si la secrétaire dit vrai, alors l'ingénieur ment

21

Formalisation en calcul des propositions

- p : la secrétaire dit vrai
- q : l'ingénieur était dans le couloir au moment du crime
- r : l'ingénieur était dans une pièce voisine de la salle de conférences
- s : l'ingénieur a entendu le coup de feu
- t : l'ingénieur dit vrai

22

Résolution de l’énigme

- Les informations de l’énoncé se traduisent par les implications :

  $p \rightarrow q$, $q \rightarrow r$, $r \rightarrow s$, $t \rightarrow \neg s$
- Il s’agit de prouver la validité de la formule :

  $(p \rightarrow q \wedge q \rightarrow r \wedge r \rightarrow s \wedge t \rightarrow \neg s) \rightarrow (p \rightarrow \neg t)$

23

Démonstration

$(p \rightarrow q \wedge q \rightarrow r \wedge r \rightarrow s \wedge t \rightarrow \neg s) \rightarrow (p \rightarrow \neg t)$

- La formule ne peut être fausse que si
  - *$(p \rightarrow \neg t)$ est faux, soit p et t vrais*
  - *la prémisse est vraie, soit toutes les implications vraies*
- Comme t doit être vrai, s doit être faux, donc r faux, donc q faux, donc p faux, et il y a contradiction

24

Calcul des prédicats

- Comment écrire les formules ?
  - *Aspects syntaxiques*
- Comment déterminer la valeur de vérité d'une formule ?
  - *Aspects sémantiques*
- Comment démontrer de nouveaux résultats ?
  - *Aspects déductifs*

25

Limites du calcul propositionnel

- Modéliser
  - *Les chandelles sont faites pour éclairer*
  - *Quelques chandelles éclairent très mal*
  - *Quelques objets qui sont faits pour éclairer le font très mal*

Impossible

26

Une modélisation

- *Les chandelles sont faites pour éclairer*

$$\forall x, \quad chandelle(x) \rightarrow éclaire(x)$$

- *Quelques chandelles éclairent très mal*

$$\exists x, \quad chandelle(x) \wedge éclaireMal(x)$$

- *Quelques objets qui sont faits pour éclairer le font très mal*

$$\exists x, \quad éclaire(x) \wedge éclaireMal(x)$$

Syntaxe

- des **connecteurs** ($\neg$, $\wedge$, $\vee$, $\rightarrow$ et $\leftrightarrow$)
- des **quantificateurs** ($\forall$ et $\exists$)
- des **variables** (x,y, …)
- des relations (**prédicats**) (R, S, éclaire, …)
- des symboles de **fonctions** (f, g, …)
  - *les fonctions d'**arité** 0 sont appelées des **constantes***

Définitions

- Terme :
  - Une variable est un terme — X
  - Une constante est un terme — tom
  - Si t1, t2, …, tn sont des termes, alors f(t1,t2,…,tn) est un terme — mere(tom)
- Atome :
  - Si t1, t2, …, tn sont des termes, alors p(t1,t2,…,tn) est un atome — humain(socrate)

29

Construction d'une formule

- V, F sont des formules
- Un atome est une formule
- Si F1 et F2 sont les formules, alors $\neg$F1, F1$\wedge$F2, F1$\vee$F2, F1$\rightarrow$ F2 sont des formules
- Si F est une formule, $\forall$ x F et $\exists$x F sont des formules

- Remarque : la logique des propositions est un cas particulier de la logique des prédicats

humain(socrate)

$\forall$ X. humain(X) $\rightarrow$ mortel(X)

30

Exemples de formules valides

- $\forall x \neg A \Leftrightarrow \neg \exists x\, A$
- $\forall x\, A \Leftrightarrow \neg \exists x \neg A$

Preuve et démonstration

- Comment **prouver** une formule du calcul des prédicats ?
  - Prouver qu’elle est vraie
    - passer en revue **toutes** les interprétations !
  - Prouver qu’elle est fausse
    - **trouver** une interprétation qui invalide la formule

Unification

- Deux termes t1 et t2 sont unifiables s'il existe une substitution s des variables de t1 et t2 telle que s(t1) = s(t2)
- Exemples :
  - pere(X,jean) s'unifie avec pere(Y,Z) si X|Y et jean|Z
  - pere(jean,mere(X)) s'unifie avec pere(Y,mere(pierre)) si jean|Y et X|pierre

33

Règles de production

→Règle: couple situation-action

reconnue **exécutée**

Action= conclusion à tirer ou traitement à exécuter

Exemple:
Si *X est un chien alors X est un mammifère*

34

Règles de production

- Règles de production
  - Syntaxe
    - **SI** *conditions* **ALORS** *conclusions*
    - **SI** *événements* **ALORS** *actions*

35

Règles de production

- Employées pour représenter des connaissances très variées :

❑ **Connaissances « heuristiques »**

– L'appendicite provoque généralement une douleur vive dans la partie droite de l'abdomen.

❑ **Connaissances « profondes »**

– L'appendice se trouve généralement dans la partie droite de l'abdomen. L'inflammation d'un organe cause généralement une douleur locale.

❑ **Connaissances « stratégiques »**

(méta connaissances : sur l'utilisation des connaissances)

– Si un diagnostic ne peut être atteint par l'usage des connaissances heuristiques, alors essayer les connaissances profondes.

36

Règles de production

- une règle= morceau indépendant de connaissances
  - Rien n'empêche son exécution sauf sa condition
  - Une règle ne peut jamais en appeler une autre
- Règles en vrac:
  - sans savoir comment seront utilisées (pas d'ordre)
  - Interpréteur= seul décideur

37

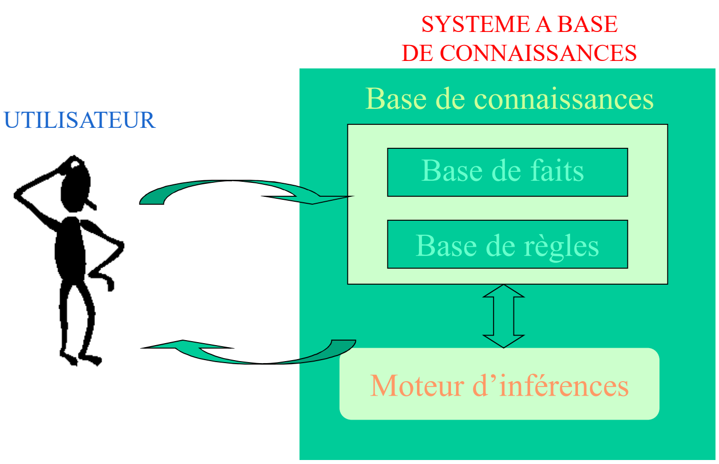
Représentation des connaissances

- Règles de production
  - Elles sont exprimées dans l'une des deux logiques mathématiques : ***logique des propositions***, ***logique des prédicats***
  - Le terme production vient du mécanisme qui consiste à produire des faits à partir des faits initiaux et des règles d'inférences
  - formalisme le plus répandu dans le domaine des systèmes experts (systèmes à base de connaissances)

38

Règles de production

- Architecture des systèmes à base de règles de production ou à base de connaissances



Moteur d'inférences : Chaînage avant

- Saisie des faits initiaux
- Début
  - ➢ Phase de filtrage => Détermination des règles applicables
  - ➢ Tant que ensemble de règles applicables n'est pas vide ET que le problème n'est pas résolu Faire
    - Phase de choix => Résolution des conflits
    - Appliquer la règle choisie (exécution)
    - Modifier (éventuellement) l'ensemble des règles applicables
  - ➢ Fin faire
- Fin

40

Exemple : les règles

- REGLE r1
  SI animal vole ET animal pond des oeufs
  ALORS animal est un oiseau
- REGLE r2
  SI animal a des plumes
  ALORS animal est un oiseau
- REGLE r3
  SI animal est un oiseau ET animal a un long cou ET animal a de longues pattes
  ALORS animal est une autruche

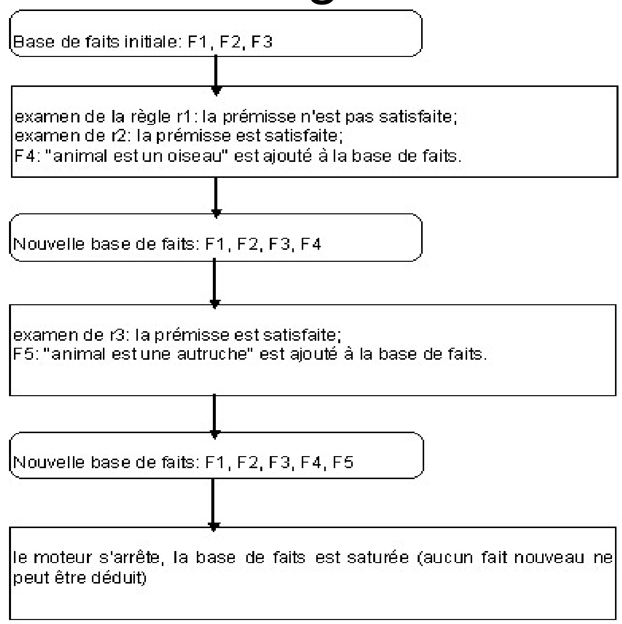
41

Exemple : les faits

- F1 : animal a des plumes
- F2 : animal a un long cou
- F3 : animal a de longues pattes

42

Chaînage avant



43

Chaînage arrière

- Le principe est le suivant :
  - ➢ Le moteur recherche les règles qui concluent sur le but à vérifier, et s'assurent que ces règles sont "déclenchables".
  - ➢ La règle est déclenchable si ses prémisses sont vérfiées.
  - ➢ Si parmi les règles sélectionnées, une règle est déclenchable, alors le but est vérifié.
  - ➢ Si ce n'est pas le cas, alors les prémisses à vérifier deviennent de nouveaux buts, appelés sous-buts, et le processus est réitéré.
- Les principales conditions d'arrêt :
  - ➢ L'ensemble des sous-buts est vide (succès) = tous les sous-buts ont été vérifiés et le problème est résolu
  - ➢ Impasse ou échec : Soit un des sous - buts n'est pas vérifiable avec la règle courante et il faut choisir une nouvelle règle pour le vérifier, et si cela n'est pas possible, alors il y a échec.

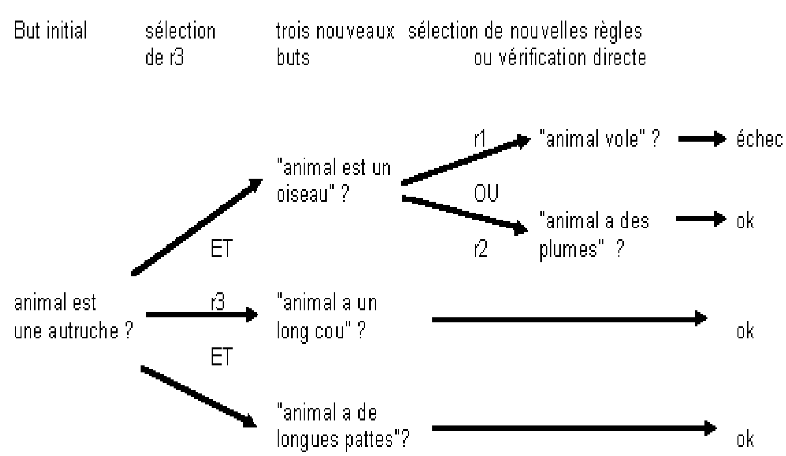
44

Moteur d'inférences : Chaînage arrière

- Phase de filtrage
- Si l'ensemble des règles sélectionnées est vide Alors questionner l'utilisateur
- Sinon
  - Tant que le but n'est pas résolu ET qu'il reste des règles sélectionnées Faire
    - Phase de choix
    - Ajouter les sous-buts (partie gauche de la règle choisie)
    - Si un sous-but n'est pas résolu Alors mettre le sous-but en but à résoudre
  - Fin faire

45

Chaînage arrière



46

Règles de production

Base des règles

- Avantages:
  - ➢ Modularité: modifications n'affectent pas le reste
  - ➢ Plus de règles = puissance + conclusions fines
- Dangers:
  - ➢ Evolution= perte de cohérence (redondance, contradictions)

47