Des logiques à la programmation en logique

P. De Loor – deloor@enib -

Programmer en Logique

- Une définition de l 'I.A. [ALL94] : *faire exécuter par l'ordinateur des tâches pour lesquelles l 'homme est meilleur que la machine.*
- 3 courants de pensée (voire annexes)
  - **Approche cognitive (source de la PL)**
  - **Approche pragmatique (utilisation de la PL)**
  - Approche connexionniste

Plan

- 1. Programmer en logique ?
- 2. Origines des Logiques
- 3. Systèmes formels
- 4. Calcul propositionnel
- 5. Calcul des prédicats
- 6. Logiques non-classique
- 7. Démonstration automatique
- 8. Programmation en logique

Programmer en Logique ?

- Paradigmes de programmation
  - mémoire/instruction : impératif
  - mathématique : fonctionnelle
  - génie logiciel : objet
  - raisonnement : logique
  - ...

Un programme en PL exprimera des règles (et des faits) l 'interpréteur « raisonnera » dessus

2. Les origines de la logique

D ’où viennent **les** logiques

- Origines philosophiques
- Caractériser les raisonnements valides
- Qu’est ce qu’un savoir permet de faire
  - classer, trier les « mots clés » du savoir
  - trouver des règles de fonctionnement
  - utiliser ces règles pour démontrer des possibilités et impossibilités
  - étudier ces règles pour discuter de leur bien-fondé

D ’où viennent les logiques

- Aristote (384-322 avant J.C)
  - *l ’Organon* (instrument (du savoir))
  - étude des propriétés des propositions *catégoriques*

Certains chimpanzés sont des femelles

quantité sujet qualité « prédicat »

– énoncé déclaratif : vrai ou faux

D ’où viennent les logiques

- Les quatre propositions catégoriques :

  A - L’universelle affirmative : *Tout homme est mortel*

  E - L’universelle négative : *Aucun homme n’est mortel*

  I - La particulière affirmative : *Quelque homme est mortel*

  O - La particulière négative : *Quelque homme n’est pas mortel*

  Aff**i**rmo / n**E**g**O**

- 5 propriétés

D ’où viennent les logiques

- Exemple de propriété : *l ’inférence directe*
  - Si **A** est vraie, **E** est fausse, **I** est vraie et **O** est fausse
  - Si **O** est vrai, on ne sait rien sur **E** et **I**

D ’où viennent les logiques

- Exemple de propriété : *la contraposition*

Tous les lézards sont verts,

donc :

Tout ce qui n ’est pas vert n ’est pas un lézard

D ’où viennent les logiques

- Exemple de propriété : *le syllogisme*

aucune vulgarisation n ’est aisée

quelques travaux sont des vulgarisations

quelques travaux ne sont pas aisés

(EIO)

  - Approche systématique (64 syllogismes)
  - prémisses des approches formelles

Historique

450 av. J.C - Stoïciens - logique propositionnelle inférence.
322 av. J.C - Aristote - propositions catégoriques, syllogismes (règles d 'inférences) quantificateurs.

1565 - Cardano - théorie des probabilités (logique des propositions & incertitude)

1666 - Leibniz - logique symbolique contemporaine
1847 - Boole - logique des propositions mise sous la forme d 'algèbre

1879 - Frege - logique du premier ordre

1930 - Gödel - $\exists$ algorithme complet pour la logique du premier ordre
1930 - Herbrand - algorithme complet pour premier ordre
1931 - Gödel - $\neg\exists$ algorithme complet pour l 'arithmétique
1960 - Davis/Putnam - algorithme utile pour la logique propositionnelle
1965 - Robinson -algorithme « utilisé » pour la logique du premier ordre : principe de résolution

Prolog
CLP,ILP,MOLOG...

D ’où viennent les logiques

- Clés de voûte de la logique

  – *modus ponens*

  si le premier **alors** le second

  or le premier **donc** le second

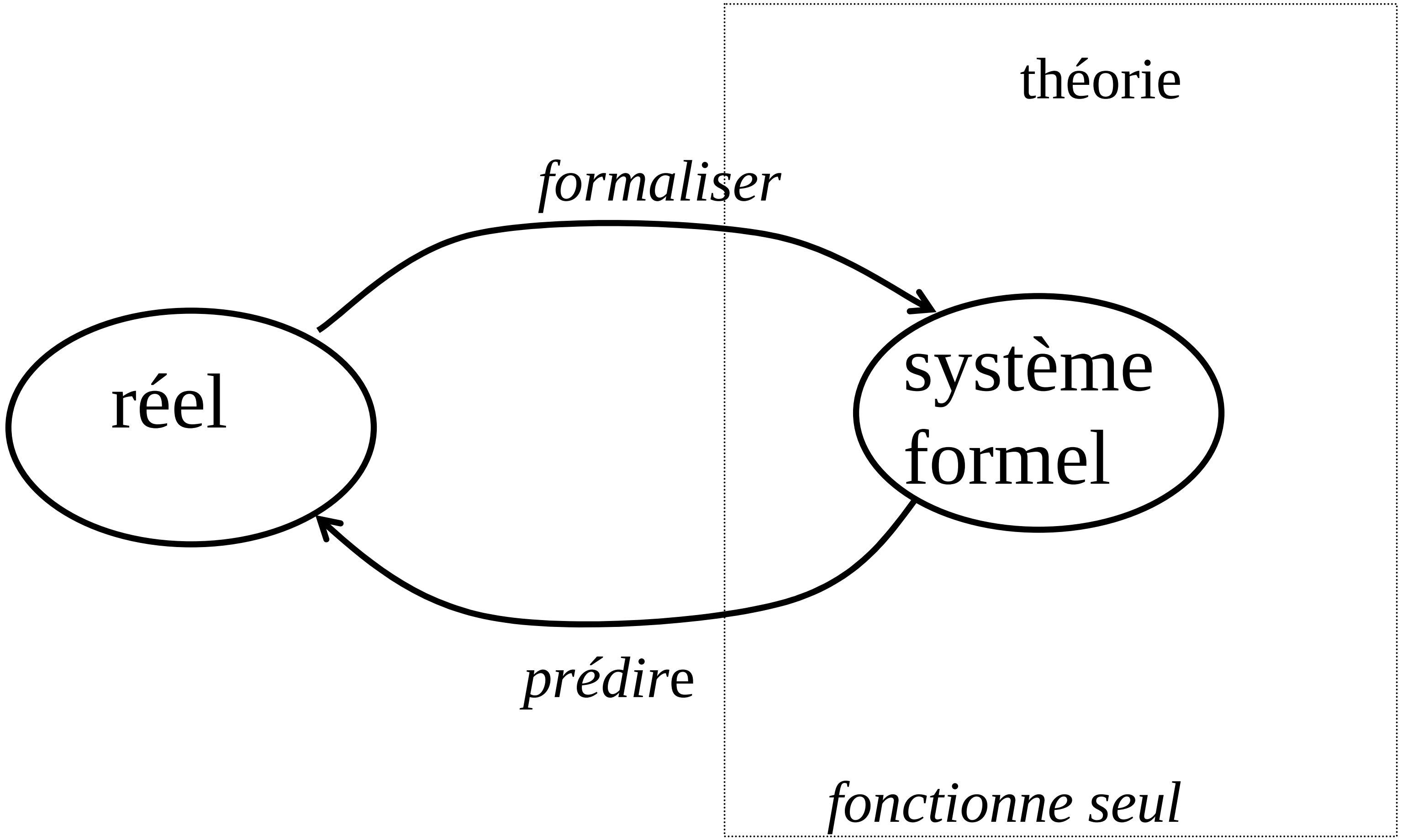
  – *modus tollens*

  si le premier **alors** le second

  or pas le second **donc pas** le premier

3. Systèmes formels

Formaliser


théorie
formaliser
réel
système
formel
prédire
fonctionne seul

Limite de la formalisation

– simplification/abstraction de la réalité

– système formel logique

- formalisation du raisonnement,
- **pas du *sens***

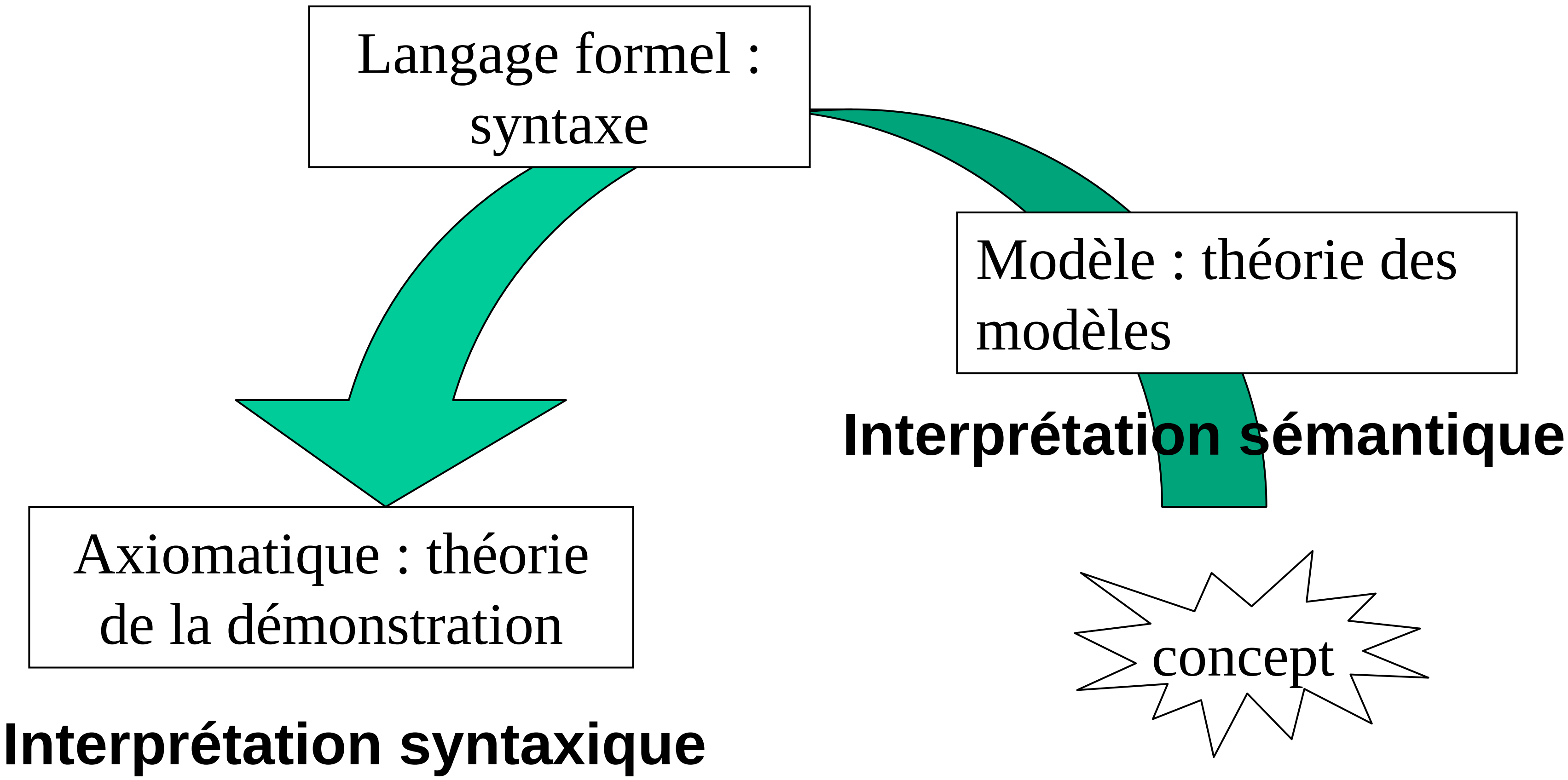
– paradoxe du menteur

– théorème d ’incomplétude

Système formel

- **Un langage formel**
  - mots clés, règles de grammaire (formule bien formée)
- **Un modèle formel**
  - interprétation ≈ sens (vrai faux ou autres symboles)
  - simplification de la réalité (danger !!)
  - relié à un concept (propositions, temps ...)
- **Un système déductif formel : axiomatique**
  - permet de faire du « calcul » : Calcul formel
  - règles de déductions (axiomes, inférence)
  - se suffit à lui même

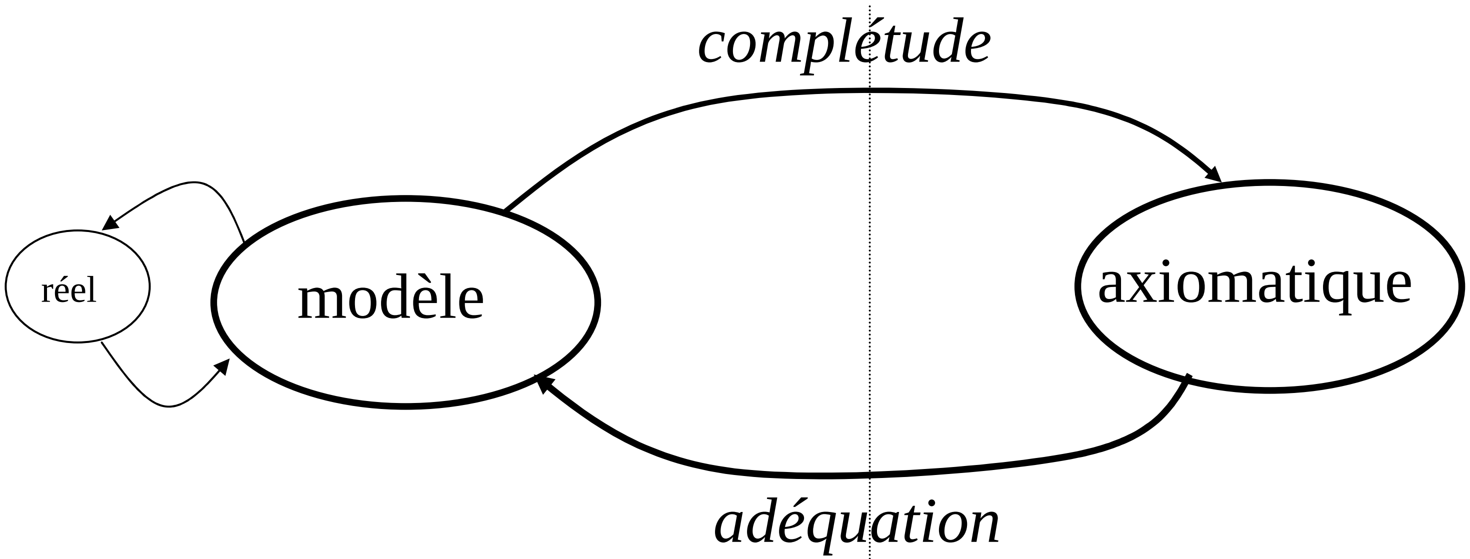
Mots clés des formalistes



Un *bon* système formel

- Modèle
  - cohérent avec le concept que l'on s'est fixé
- Langage
  - simplicité: pas d 'ambiguité
  - expressivité
- Axiomatique
  - cohérente avec le modèle
  - minimale : pas de redondance
  - exécutable « automatiquement »

Les liens modèle-axiomatique



Notion de vérité {v,f}
de validité

Notion de preuve :
théorèmes

Propriétés importantes

– **consistance :** ***avec mon langage, je ne peux pas trouver une chose vraie et fausse à la fois***

– **décidabilité :** ***mon langage peut il répondre en un temps fini à une question ?***

– complétude : *lien modèle-axiomatique, tout ce que mon modèle considère valide, mon langage peut le prouver*

– adéquation : *lien axiomatique-modèle : tout ce que je peux déduire de mon langage est valide (toujours vrai dans mon modèle)*

4. Calcul propositionnel

Ou logique d 'ordre 0

Calcul propositionnel

Langage : propositions, connecteurs, règle d ’équivalence

Axiomatique : 3 axiomes, 1 règle d ’inférence

Modèle : tables de vérité

Proposition langage naturel

Calcul propositionnel

- Exemple de *cognition* :
  - S 'il pleut, alors la route est mouillée
  - Donc si la route est sèche c'est qu'il ne pleut pas
- Eléments clés :
  - propositions : `la_route_est_sèche, il_pleut`
  - négation : `sèche` ⇔ `mouillée, pleut` ⇔ `ne_pleut_pas`
  - règle : `si alors`
  - une déduction : `donc`
- Début de formalisation
  - `si X alors Y, donc si non Y alors non X`

Calcul propositionnel : le langage

- Les mots :
  - propositions dénombrables ou formules atomiques ou atomes (p, q, r, …)
  - opérateurs (connecteurs) :

$\neg$	négation
$\wedge$	conjonction
$\vee$	disjonction
$\rightarrow$	implication
$\leftrightarrow$	bi-implication
false	contradiction

  - ponctuations : ( )

Calcul propositionnel : le langage

- Les phrases (formules)
  - règles définissant les formules « bien formées »
  - exemple :
    - si A est une formule atomique alors A est une formule
    - FALSE est une formule
    - (A v B) est une formule si A et B sont des formules
    - …
    - énoncé mal formé : $(\vee\ p)q$

Récursif !

Calcul propositionnel : le modèle

- Les propositions peuvent posséder 2 valeurs
  - ☞ {0,1} ou {t,f} ou {titi, toto} …

- Les opérateurs définissent la valeur d 'une formule
  - ☞ table de vérité
  - ☞ les formalistes appellent ça une Interprétation I

ps : Si I(A) = t (true), on écrit $I \models A$ (satisfiabilité)

Calcul propositionnel : le modèle

- Interprétation des connecteurs *dyadiques*

A	B	$(A \leftrightarrow B)$	$(A \rightarrow B)$	$(A \wedge B)$	$(A \vee B)$
t	t	t	t	t	t
t	f	f	f	f	t
f	t	f	t	f	t
f	f	t	t	f	f

- négation

A	$\neg A$
t	f
f	t

Calcul propositionnel : le modèle

- Exemple d'interprétation
  - propositions = {p1, p2, ...}.
  - interprétation I : I(p1) = v, I(p2), = I(p3) = ... = f

$I(p1 \vee p2) = v$

$I((p1 \wedge p2) \rightarrow (p1 \wedge p1)) = v$

$I(p1 \wedge (p2 \vee \neg p1)) = f$

Calcul propositionnel : le modèle

- tautologie ou formule valide

  formule toujours vraie quelle que soit l 'interprétation (on note $\models A$)

- exemple

$$p \rightarrow (q \rightarrow p)$$

- formules satisfiables : au moins une interprétation vraie.

Calcul propositionnel : l ’axiomatique

- Purement syntaxique

« vérité » non discutable

Axiomes

mécanismes déductifs

règles d ’inférence

Nouvelles formules valides : théorèmes

technique copier/coller

substitution

Calcul propositionnel : l 'axiomatique

- Les Axiomes

  1) $A \rightarrow (B \rightarrow A)$

  2) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$

  3) $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$

- remarque : les connecteurs $(\wedge, \vee, \leftrightarrow)$ ne sont que des synonymes
  - exemple $(A \wedge B)$ est synonyme de $\neg(A \rightarrow \neg B)$

Calcul propositionnel : l 'axiomatique

- Règle d 'inférence
  - modus-ponens

    si A est « vraie » et si A implique B alors B est « vraie »

$$\frac{\vdash A\ , \quad \vdash A \to B}{\vdash B} \qquad \begin{array}{l}\text{prémisses} \\ \text{conclusion}\end{array}$$

$\vdash$: Est un théorème (ou un axiome)

Calcul propositionnel : l 'axiomatique

- La substitution
  – exemple

axiome donc tautologie

$$(\neg P \rightarrow \neg Q) \rightarrow (Q \rightarrow P)$$

substitution

$$a \vee b \rightarrow a$$

  – remplacement d 'une proposition/formule par une formule, préserve la tautologie

Calcul propositionnel : l 'axiomatique

- Démonstration :

partant des axiomes, enchaînement de subsitution/inférence pour déduire des théorèmes (fastidieux, exemple en annexe)

- Déduction :

démonstration à partir d 'hypothèses supplémentaires (formules)

théorème de Herbrand (1930) : si A|-B alors |-(A→B)

Calcul propositionnel : propriétés importantes

- Adéquat *(si $\vdash A$ alors $\models A$)*
- Complet *(complétude forte : si $E \models A$ alors $E \vdash A$)*
- Consistant *(il n 'existe pas de formule telle que $\vdash A$ et $\vdash \neg A$)*
- Décidable

5. Calcul des prédicats

Ou logique du premier ordre

Calcul des prédicats

- Limite des propositions

la_bière_est_blonde

- On aimerait exprimer

estBlonde(X), si X est une Leffe

Calcul des prédicats

- Notion d ’ensemble
  - Ensemble complet (pour tout)
  - Eléments particuliers d ’un ensemble (il existe)

TOUT être humain est mortel
OR Socrate est un être humain
DONC Socrate est mortel

Calcul des prédicats

- Notion d 'ordres

`la_biere_est_blonde`	ordre 0
`est_blonde(X)`	ordre 1
`est_blonde(X(Y))`	ordre 2
	...

Calcul des prédicats : le langage

- On ajoute au calcul propositionnel
  - des constantes : *a*,*b*,*c*,..
  - des symboles de fonctions (arité ≥ 0) : *f*,*g*,..
  - des symboles de prédicats (arite≥0) : *p*,*q*,..
  - deux quantificateurs : $\exists$, $\forall$

les prédicats d 'arité 0 sont des variables propositionnelles

Les arguments des prédicats sont des termes

Les fonctions d 'arité 0 sont des constantes

Calcul des prédicats : le langage

- Terme
  - une variable est un terme
  - un symbole de constante est un terme
  - si *f* est un symbole de fonction et *t1*,*t2*…*tn* sont des termes alors *f(t1*,*t2..tn)* est un terme
  - les termes sont les arguments des prédicats

Récursif !!

Calcul des prédicat : le langage

- Atome
  - si p est un symbole de prédicat et t1..tn sont des termes, alors p(t1,t2..tn) est un atome
  - un atome s 'appelle aussi une formule atomique

Calcul des prédicats : le langage

- Formules
  - toute formule atomique est une formule
  - si A et B sont des formules alors $A \rightarrow B$ est une formule
  - …
  - si A est une formule est si x est une variable quelconque alors $\exists x\ A$ est une formule

Calcul des prédicats : le langage

- Variables libres et liées
  - exemple, soit la formule

$$\forall x \; p(x,y)$$

  - x est liée à la formule
  - y est libre
  - définition complète : quelles sont les variables libres et liées d 'une formule composée de formules...

Calcul des prédicats : axiomatique

- Extension de l 'axiomatique du calcul propositionnel
- Quelques axiomes en plus
- Règle d 'inférence supplémentaire : généralisation

$$\frac{\vdash A}{\vdash \forall x\, A}$$

- Démonstration/déduction
  - découle du calcul propositionnel

Calcul des prédicats : le modèle

- Exemple : interprétation du prédicat *estBleu(x)*

x	I(estBleu)(x)
martien	f
homme	f
schtroumpf	t

- Définition formelle -termes-formules : voir littérature

Calcul des prédicats : propriétés importantes

- Adéquat
- Complet *(Gödel)*
- Consistant
- Semi-décidable :

  si A est valide alors la procédure s'arrête et retourne `oui'

  sinon ou bien la procédure s'arrête et retourne `non', ou bien elle ne s'arrête pas.

6. Logiques non classiques

Logiques non-classiques

- Autres modèles (donc autre langage)
  - logiques faibles
    - absolue, minimale, positive, intuitionniste
  - logiques modales
    - aléthique
    - temporelle
    - déontique, connaissance, croyance
    - premier ordre
  - logiques multivaluées
    - floues

Logiques faible

- Question sur la négation
  - logique absolue, positive : pas de négation
  - logique minimale : $\neg$ devient « est réfutable »
    - $p \rightarrow \neg\neg p$, mais pas l 'inverse
  - logique intuitionniste : $\neg$ devient « est absurde »

Logiques modales

- Problème du modèle de la logique classique : l 'implication matérielle

A	B	$(A \leftrightarrow B)$	$(A \rightarrow B)$	$(A \wedge B)$	$(A \vee B)$
t	t	t	t	t	t
t	f	f	f	f	t
f	t	f	t	f	t
f	f	t	t	f	f

Si A = f, $A \rightarrow B$ est vrai (quelque soit B) !!

Implication matérielle et bon sens

- Lien avec le *réel* :
  - `La_bière_est_verte`... est une proposition fausse : f
    - donc `la_bière_est_verte` implique que 1 + 1 = 3 est vraie.
  - Si on décide que « A→B » est faux dès que A = f

    alors « → » devient la même chose que « ∧ »
  - Or
    - <u>*si* il fait beau j'irais me promene</u>r … vraie : t
      - *donc* s'il ne fait pas beau je n'irais pas,
      - mais reste vraie

Logiques modales

- Nouvelle implication
  - > : implication stricte

possible

$$A > B =_{def} \neg\Diamond(A \wedge \neg B)$$

A implique strictement B si il est impossible que A soit vraie et B faux

- L ’inverse de la possibilité est la nécessité :

nécessaire

$$A =_{def} \neg\Diamond\neg$$

Logiques modales : le modèle

Si `ciel_bleu` est vrai alors

`ciel_bleu` $\vee$ `ciel_pas_bleu` est vrai

`ciel_bleu` est faux

(`ciel_bleu` $\vee$ `ciel_pas_bleu`) est vrai

- Table de vérité = impossible
- Le modèle est un graphe représentant une évolution des propositions (mondes possibles)

Logiques temporelles

- Logiques temporelles modales
  - le graphe représente l 'évolution dans le temps
- Notions représentées
  - avant, après, toujours, parfois (o,P,U,…)
- Diversité
  - temps discrets/continu
  - temps linéaire/arborescent
  - temps métré/symbolique

Autres logiques modales

- Déontique
  - O : obligation
  - P : permission
  - informatique juridique
- Dynamique
  - séquence, choix non déterministe, parallélisme

Logiques multi-valuées

- Logique multivaluée (Lukasiewicz)
  - Poids à la notion de vérité (très vrai, peu probable)
  - Valeur de vérité sur un intervalle [0,1]
  - Exemple du $\wedge$: $p \wedge q = \min(p , q)$
- Logique floue (Zadeh)
  - Partition dénombrable de sous-intervalles de vérité
  - Vrai, faux, pas vrai, pas faux, plus ou moins vrai, plutôt vrai…
  - Exemple d 'inférence :

$$\frac{\text{a petit, a et b presque semblables}}{\text{b est plus ou moins petit}}$$

  - certains auteurs ne la considèrent pas comme une logique (axiomatique)

7. Démonstration automatique

Démontrer ...

- Montrer que quelque chose est *vrai*
- calcul formel = prouver un théorème
- calcul à partir du langage
    - définir un ensemble de règles d ’inférences
    - définir un ensemble d ’axiomes
    - répététivement choisir un règle & des axiomes pour arriver progressivement au théorème voulu
    - Long et difficile (quels règles et axiomes choisir ?)

Démonstration automatique

- Déduction en logique propositionnelle
  - table de vérité - Wittgenstein - 1922
  - arbres sémantiques
  - algorithme de Quine
  - algorithme de Davis/Putman
  - <u>principe de résolution de Robinson -1965</u>
  - <u>clauses de Horn</u>

efficacité, simplicité

Principe de résolution

– Une seule règle pour montrer qu'une phrase logique est vraie ou non (simplicité)

– Réduction des formules logiques sous forme « normale conjonctive »

– Résolution par réfutation (raisonnement par l 'absurde)

- prendre un système formel
- considérer le contraire du théorème à prouver
- montrer que l 'association des 2 est insatisfiable

ensemble vide

démontrer que $\{p \rightarrow r, q \rightarrow r\} \models (p \vee q \rightarrow r)$

revient à démontrer $\{p \rightarrow r, q \rightarrow r, \neg(p \vee q \rightarrow r)\} \models \emptyset$

Principe de résolution

- Forme Normale Conjonctive (FNC)
  - que des disjonctions $\vee$
  - négations que sur les atomes

$$\neg \text{hiver} \vee \text{froid}$$

Principe de résolution

- La règle d 'inférence
  - modus ponens vue d 'une autre façon

$$\frac{\boxed{A \lor C}, \neg C \lor B}{A \lor B}$$

clause

forme normale

  - trouver deux clauses telles que
    - on y trouve une proposition positive dans l 'une
    - on y trouve la même proposition négative dans l 'autre
    - fusionner ces deux clauses = création d 'une clause

Principe de résolution

- Exemple

$$\text{été} \vee \text{hiver}, \neg\text{hiver} \vee \text{froid}$$

$$\downarrow$$

$$\text{été} \vee \text{froid}$$

Principe de résolution

- Prouver que des prémisses impliquent une proposition S.
  - nier S et convertir le résultat en FNC
  - convertir les prémisses en FNC
  - répéter jusqu'à ce qu 'une contradiction apparaisse
    - sélectionner 2 clauses
    - les résoudre mutuellement
    - si le résolvent est la clause vide, une contradiction est trouvée
    - sinon, ajouter le résolvent aux prémisses

Principe de résolution : exemple

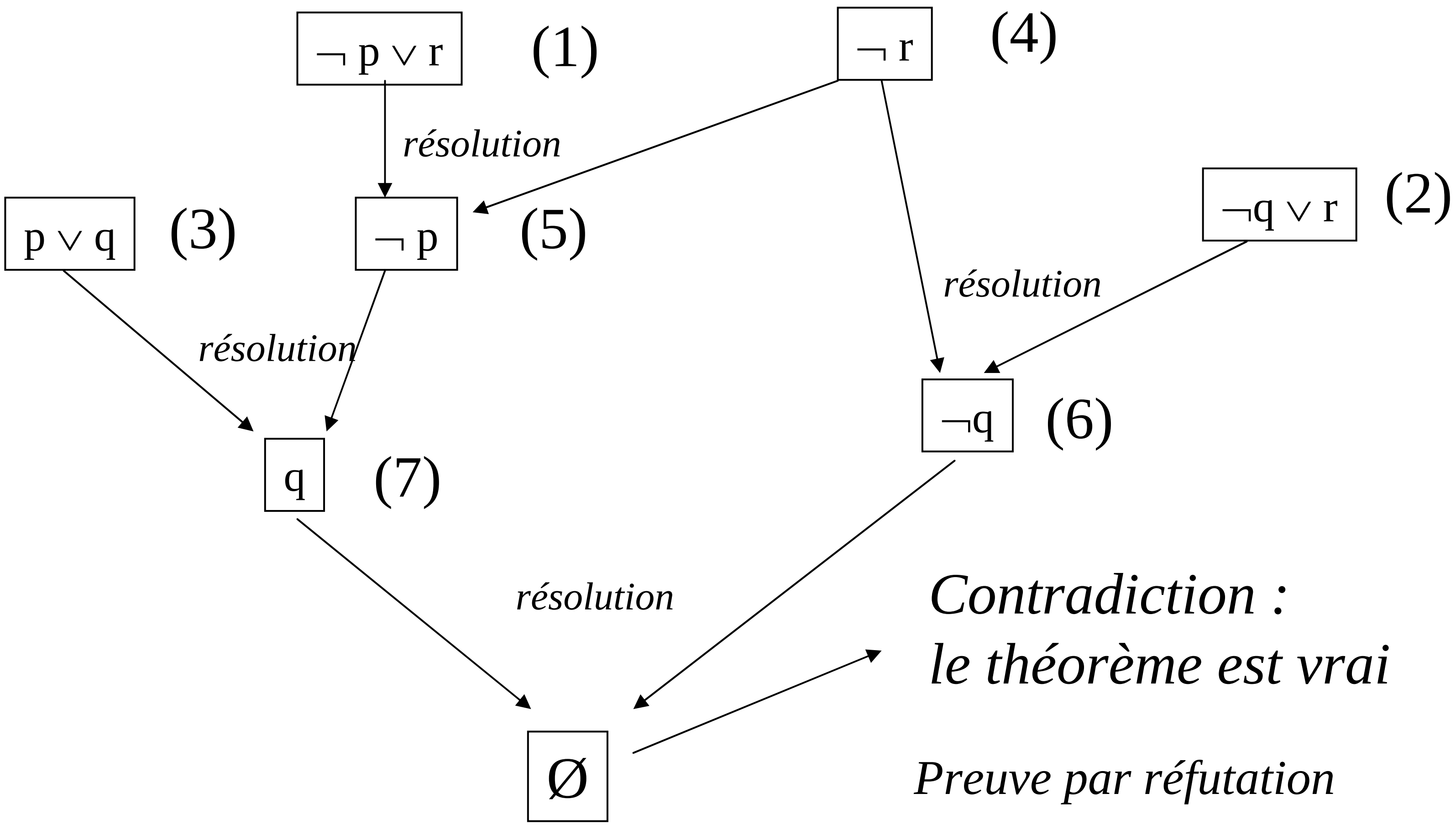
démontrer que $\{p \rightarrow r, q \rightarrow r\} \models (p \vee q \rightarrow r)$

revient à démontrer que $\{p \rightarrow r, q \rightarrow r, \underbrace{\neg(p \vee q \rightarrow r)}_{\textit{négation de la conclusion}}\} \models \varnothing\}$

Conversion en FNC :

$p \rightarrow r$	$\Rightarrow$	$\neg p \vee r$	(1)
$q \rightarrow r$	$\Rightarrow$	$\neg q \vee r$	(2)
$\neg(p \vee q \rightarrow r)$	$\Rightarrow$	$p \vee q$	(3)
		$\neg r$	(4)

Principe de résolution : exemple



Règles d 'obtention d 'une FNC

- Dans l 'ordre

$A \leftrightarrow B$	$(A \rightarrow B) \wedge (B \rightarrow A)$
$A \rightarrow B$	$\neg A \vee B$
$\neg(A \wedge B)$	$\neg A \vee \neg B$
$\neg(A \vee B)$	$\neg A \wedge \neg B$
$\neg\neg A$	A
$A \vee (B \wedge C)$	$(A \vee B) \wedge (A \vee C)$

Correspond au '$\wedge$'

  - Exemple : $p \leftrightarrow (q \rightarrow r)$ donne

$$\underline{(\neg p \vee \neg q \vee r)},(q \vee p) ; (\neg r \vee p)$$

clause

Clause de Horn

- *Le principe de résolution est très rapide si les clauses sont sous la forme de clauses de Horn.*
  (suppressions de clauses équivalentes ou « conséquences valides »)

- Une clause de Horn
  - contient 1 litteral positif et au moins 1 négatif (clause de Horn stricte)
  - 1 seul litteral positif (clause de Horn positive)
  - que des litteraux négatifs (clause de Horn négative)

Résolution sur clauses de Horn

- Exemple
  - inconsistance de $\{\neg p \vee r, \neg r \vee s, p, \neg s\}$
  - 1. Sélection de p et $\neg p \vee r$: $\{r, \neg r \vee s, p, \neg s\}$
  - 2. Sélection de r et $\neg r \vee s$: $\{r, s, p, \neg s\}$
  - 3. Sélection de $\neg s$ et s : $\{r, s, p, \emptyset\}$

inconsistance

Interprétation des clauses de Horn

– clauses de Horn positives : $p \vee q \vee ..$
  - faits, vérités logiques
– clauses de Horn strictes : $q \vee \neg p1 \vee \neg p2 \vee \ldots$
  - équivalent à $\{p1, p2, \ldots\} \models q$
  - règle **si** .. **alors**
– clauses de Horn négatives $\neg p \vee \neg q \vee \neg r \ldots$
  - si on cherche à prouver $\{H1, H2..\} \models (p \wedge q \wedge r)$
  - revient à prouver $\{H1, H2,.., \neg p \vee \neg q \vee \neg r\} \models \varnothing$
  - but à atteindre, question

hypothèses

but

Démonstration automatique :
et pour la logique du premier ordre ?

- En logique propositionnelle

àLaMaison ∨ auTravail

¬àLaMaison

auTravail

- En logique des prédicats

∀x (àLaMaison(x) ∨ auTravail(x))

¬àLaMaison(y)

?

Démonstration automatique

et pour la logique du premier ordre ?

- Le Problème
  - en logique propositionnelle, les propositions sont considérées « vraies »
  - en logique des prédicats :
    - elles dépendent de variables
    - les clauses sont formées à partir de prédicats communs

Démonstration automatique

et pour la logique du premier ordre ?

- La solution
  - mêmes principes qu 'en calcul propositionnel : preuve de l 'inconsistance (par résolution)
  - mise des formules sous forme « prénexe »
  - mise des formules sous forme de skolem
  - mise sous forme de clauses
  - rechercher les valeurs « cohérentes » des variables : *UNIFICATION* || résolution

Démonstration automatique

- La forme prénexe

$$\exists x\ \forall y\ \exists z\ (\neg p(x) \vee \neg q(y)) \vee (p(z) \wedge q(z))$$

Plus d 'implication

Quantificateurs à l 'extérieur

$\Uparrow$ équivalent

$$\forall x\ p(x) \wedge \exists y\ q(y) \rightarrow \exists y (p(y) \wedge q(y))$$

Démonstration automatique

- Mise sous forme prénexe

$A \leftrightarrow B$	$(A \rightarrow B) \wedge (B \rightarrow A)$
$A \rightarrow B$	$\neg A \vee B$
$\neg(A \wedge B)$	$\neg A \vee \neg B$
$\neg(A \vee B)$	$\neg A \wedge \neg B$
$\neg \neg A$	A
$A \vee (B \wedge C)$	$(A \vee B) \wedge (A \vee C)$
$\neg \forall x\, A$	$\exists x\, \neg A$
$\neg \exists x\, A$	$\forall x\, \neg A$

Démonstration automatique

- Mise sous forme prénexe
  - respecter l 'ordre suivant :

$\forall x\ p(x) \wedge \exists y\ q(y) \rightarrow \exists y(p(y) \wedge q(y))$

1. suppression de $\rightarrow$, $\leftrightarrow$: $\neg(\forall x\ p(x) \wedge \exists y\ q(y)) \vee \exists y(p(y) \wedge q(y))$

2. renommage des variables : $\neg(\forall x\ p(x) \wedge \exists y\ q(y)) \vee \exists z(p(z) \wedge q(z))$

3. transfert des $\neg$ vers l 'intérieur : $(\exists x \neg\ p(x) \vee \forall y \neg q(y)) \vee \exists z(p(z) \wedge q(z))$

4. déplacement des quantificateurs :

$\exists x\ \forall y\ \exists z\ ((\neg\ p(x) \vee \neg\ q(y)) \vee (p(z) \wedge q(z)))$

Démonstration automatique

- Mise sous forme de Skolem
  – suppression des quantificateurs
    - existentiels : création d 'une fonction
    - universels : sous-entendus

ps : la forme de skolem n 'est pas strictement équivalente à la formule initiale, mais si une forme de skolem est satisfiable, la formule dont elle est issue l 'est aussi

Démonstration automatique

- Forme de Skolem, exemple
  - formule initiale : $\forall x\ p(x) \wedge \exists y\ q(y) \rightarrow \exists y(p(y) \wedge q(y))$
  - forme prénexe : $\exists x\ \forall y\ \exists z\ ((\neg\ p(x) \vee \neg\ q(y)) \vee (p(z) \wedge q(z)))$
  - forme de skolem : $\forall y(\neg\ p(a) \vee \neg\ q(y)) \vee (p(f(y)) \wedge q(f(y))))$

Constante, car x ne peut dépendre d 'aucune variable

fonction, car z peut dépendre de y

  - forme normale : idem au calcul propositionnel (distribution du $\vee$)

    $(p(f(y)) \vee \neg\ p(a) \vee \neg\ q(y)) \wedge (\ q(f(y) \vee \neg\ p(a) \vee \neg\ q(y))\ )))$
  - forme clausale :

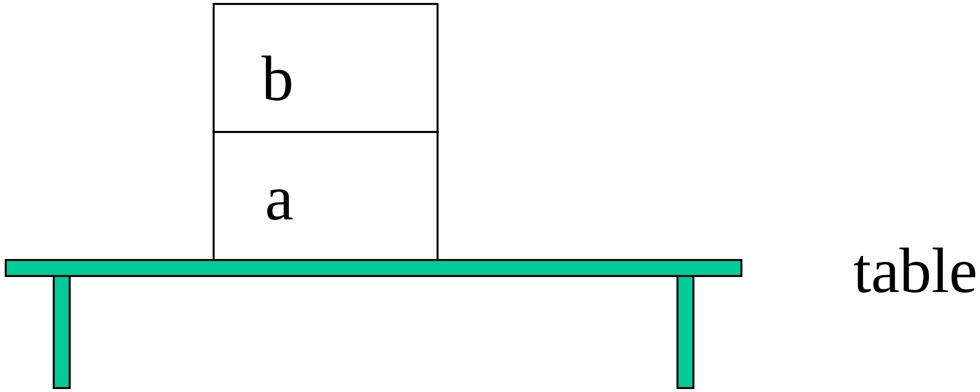
    $\{p(f(y)) \vee \neg\ p(a) \vee \neg\ q(y)\ ,\ \ q(f(y) \vee \neg\ p(a) \vee \neg\ q(y))\ ))\}$

Démonstration automatique

- Résolution par unification
  - Recherche d 'un prédicat
    - positif dans une clause
    - négatif dans une autre
  - tenter de les rendre égaux :
    - Une variable peut être remplacée par une constante
    - Une variable peut être remplacée par une variable
    - Une variable peut être remplacée par une fonction qui ne la contient pas
    - Une substitution/instanciation qui rend réductible deux clauses est une UNIFICATION

Démonstration automatique

- L 'unification, exemple :



Sachant que b est sur a et a est sur table, déduire que b est au-dessus de la table

Démonstration automatique

- Note : les variables s ’écrivent désormais X, Y , … les constante a, b, table
- Axiomes :

  $\forall$ X $\forall$ Y [sur(X,Y)$\rightarrow$auDessus(X,Y)]

  $\forall$ X $\forall$ Y $\forall$ Z [auDessus(X,Y) $\wedge$auDessus(Y,Z)$\rightarrow$auDessus(X,Z)]

- Mis sous forme normale

  $\neg$sur(U,V) $\vee$ auDessus(U,V) (1)

  $\neg$auDessus(X,Y) $\vee$ $\neg$auDessus(Y,Z) $\vee$ $\neg$auDessus(X,Z) (2)

- Deux axiomes supplémentaires spécifiques à l ’exemple

  sur(b,a) (3)

  sur(a,table) (4)

Démonstration automatique

- On veut démontrer : auDessus(b, table)
- déduction / résolution

$\neg$sur(U,V) $\vee$ auDessus(U,V) (1)

$\neg$auDessus(X,Y) $\vee$ $\neg$auDessus(Y,Z) $\vee$ auDessus(X,Z) (2)

sur(b,a) (3)

sur(a,table) (4)

$\neg$auDessus(b, table) (5)

réfutation

$\neg$auDessus(b,Y) $\vee$ $\neg$auDessus(Y,table) (2) et (5) -> (6)

Unification

$\neg$auDessus(X,Y) $\vee$ $\neg$auDessus(Y,Z) $\vee$ auDessus(X,Z) (2)

$\neg$auDessus(b, table) (5)

unification : X=b
Z=table

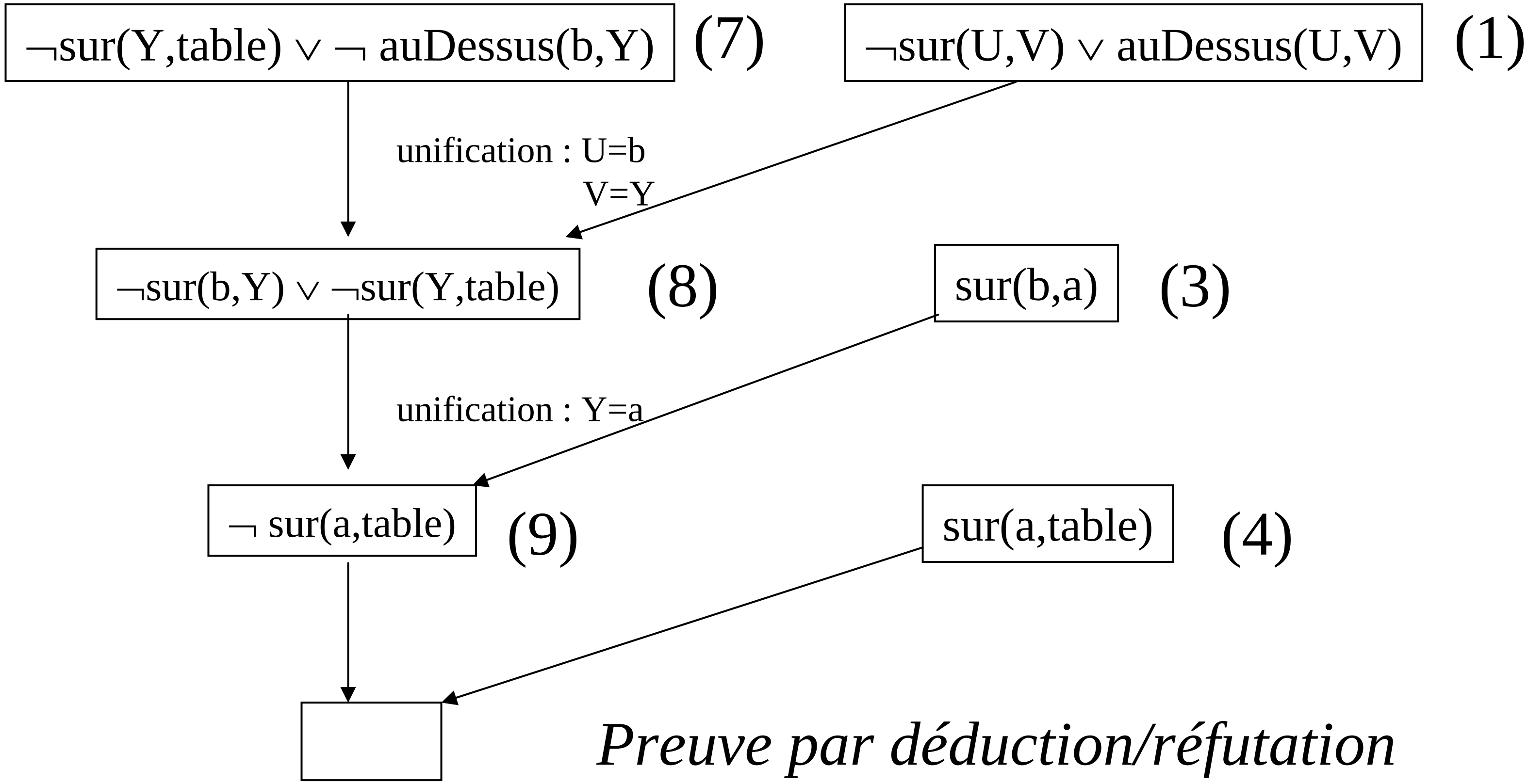
$\neg$auDessus(b,Y) $\vee$ $\neg$auDessus(Y,table) (6)

$\neg$sur(U,V) $\vee$ auDessus(U,V) (1)

unification : U=Y
V=table

$\neg$sur(Y,table) $\vee$ $\neg$ auDessus(b,Y) (7)

Unification



Preuve par déduction/réfutation

8. Programmer en logique

Un premier programme PROLOG

```
% l ’exemple des blocs

sur(a,table).
sur(b,a).
auDessus(X,Z):- sur(X,Z).
auDessus(X,Z):- auDessus(X,Y), auDessus(Y,Z).

Question à prolog
-? auDessus(b,table).

Yes
...
```

Bibliographie

- **Ouvrages**
- Intelligence artificielle & informatique théorique, J.M. Alliot & T. Schiex, cepadus edition, 1994.
- Approche logique de l 'intelligence artificielle, A. Thayse et al, Dunod Informatique, 1988.
- Introduction à la calculabilité, Pierre Wolper, Interedition, 1991.
- Outils logiques pour l 'Intelligence Artificielle, J.P. De lahaye, 1986.

- **Web :**
- http://www-lipn.univ-paris13.fr/~uvf/apprenant/chapitres.htm
- http://cui.unige.ch/ScDep/Cours/IA/ia00-01/transparents/notes-de-cours/

Limite du calcul propositionnel

- il fait beau ou il pleut
- il pleut ou il fait beau

pareil

- il prit peur et il tua l 'intrus
- il tua l 'intrus et il pris peur

pas pareil

Calcul propositionnel : le modèle

- Conséquence valide

  Soit $A = (p \rightarrow q) \wedge (p \vee r)$

  $B = (q \vee r)$

  $C = (p \rightarrow r)$
- Si A est vraie B est vraie (C ?) (table de vérité)
- B est une conséquence valide de A, $(A \models B)$
- Utile lors des démonstrations

Calcul propositionnel : axiomatique

- Exemple démonstration de |- $(A \rightarrow A)$

1. |- $((A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C)))$ - axiome 2
2. |- $((A \rightarrow ((A \rightarrow A) \rightarrow A)) \rightarrow ((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A)))$
   subtitution de B par $A \rightarrow A$ et de C par A
3. |- $A \rightarrow (B \rightarrow A)$ - axiome 1
4. |- $A \rightarrow ((A \rightarrow A) \rightarrow A)$ - substitution de B par $(A \rightarrow A)$ dans 3
5. |- $((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A))$: inférence entre 4 et 2
6. |- $A \rightarrow (A \rightarrow A)$ - axiome 1 avec substitution de B par A
7. |- $A \rightarrow A$ inférence entre 5 et 6

☒ c ’est fastidieux

Calcul propositionnel : propriétés importantes

- Adéquation

tout ce que je peux déduire de ma logique est valide

  - si $\vdash A$ alors $\models A$
  - tout théorème est une formule valide
  - le calcul propositionnel est adéquat

Calcul propositionnel : propriétés importantes

- Consistance

avec ma logique, je ne peux pas trouver une chose vraie et fausse à la fois

  – il n ’existe pas de formule telle que |-A et |- ¬A
  – le calcul propositionnel est consistant

Calcul propositionnel : propriétés importantes

- Complétude

  lien modèle-langage, tout ce que mon modèle considère valide, mon langage peut le prouver

  – le calcul propositionnel est complet

  ps : différentes familles de complétude

  – forte : si E |=A alors E|–A
  – faible : quand |=A alors |–A (les règles de raisonnement permettent de démontrer toutes les formules valides)
  – syntaxique : pour tout A on a soit |–A soit |–¬A

Calcul propositionnel : Propriétés fondamentales

- Décidabilité

Une logique est décidable si on connaît une procédure permettant de savoir, en un temps fini, si une formule donnée est ou n'est pas un théorème

– Le calcul propositionnel est décidable

Calcul des prédicats : le langage

- Exemple de traduction d ’énoncés en formules
  - tout est relatif

$$\forall x\ relatif(x)$$

  - Il y a des peines, il y a des plaisirs, mais aucune peine n'est un plaisir

$$(\exists x\ peine(x)) \wedge (\exists x\ plaisir(x)) \wedge (\forall x\ (peine(x) \rightarrow \neg plaisir(x)))$$

  - il existe un plus grand entier

$$(\exists x\ entier(x)) \wedge (\forall y(entier(y) \rightarrow plusgrand(x,y)))$$

Calcul des prédicats : le langage

- Les variables libres sont sources de sophisme logique

  – différence entre identité et égalité conditionnelle

    - $(x + 1)^{\wedge 2} = x^{\wedge 2} + 2x + 1$
    - $x^{\wedge 2} - 4x + 4 = 0$

Approche cognitive

- noms et dates
- arguments :
  - fonctions cognitives humaines = algorithmes
- proposition :
  - formaliser ces fonctions ↔ langages / calculs
- base :
  - logiques formelles
- résultats :
  - PROLOG, d 'autres langages expérimentaux

Approche pragmatique

- noms et dates
- arguments :
  - fonctions cognitives humaines trop complexe
  - machine = contrainte matérielle
- proposition :
  - faire au mieux pour que ça marche / bon sens
- base :
  - programmation logique, calcul de complexité
- résultats :
  - systèmes experts, heuristiques, Allen, SHDRLU ...

Approche connexionniste

- noms et dates
- arguments :
  - fonctions cognitives humaines = trop complexe
- proposition :
  - imitons le cerveau au niveau « hard »
- base :
  - biologie
  - mathématiques
- résultats :
  - réseau de neurones, ...

Calcul de prédicats : le langage

- Preuve par réfutation
- Preuve par résolution

Logique propositionnelle : le langage

- Les synonymes (règles d 'équivalence)
  - exemple :

  si on prend juste {$\neg$, $\vee$, $\wedge$} comme connecteurs primitifs

    - FALSE peut être défini comme p $\wedge$ $\neg$ p
    - A $\rightarrow$ B peut être défini comme $\neg$ A v B.

Logique propositionnelle : axiomatique

- La substitution
  - F est une formule contenant des atomes p1, p2,..pn
  - si $\models$ F
  - si F* est F dans laquelle on remplace p1, p2.. par des formules A1, A2…
  - alors : $\models$ F*
- Utiliser pour les démonstrations

Calcul des prédicats : le langage

- Formule fermée : toutes les variables sont liées

$$\forall x \forall y ((p(x) \vee \exists x\, p(x)) \wedge q(y))$$

- Formule ouverte

$$\forall y ((p(x) \vee \exists x\, p(x)) \wedge q(y))$$

Note sur les systèmes formels

- Prouver la complétude et décidabilité…??
- Outil : Machines de Turing
- exemple :
  - incomplétude de l ’arithmétique formelle
  - indécidabilité de l ’arithmétique formelle

Langage naturel : syntaxe, sémantique, pragmatique

Il avait le nez collé à un mur haut, large et épais ...

- Il avança à un cha : lexicalement incorrect
- il avança à un chat : lexicalement correct, mais syntaxiquement incorrect ;
- il avança d'un chat : syntaxiquement correct, mais sémantiquement incorrect ;
- il avança d'un pas : sémantiquement correct, mais pragmatiquement incorrect (à cause du mur) ;
- il recula d'un pas : pragmatiquement correct.