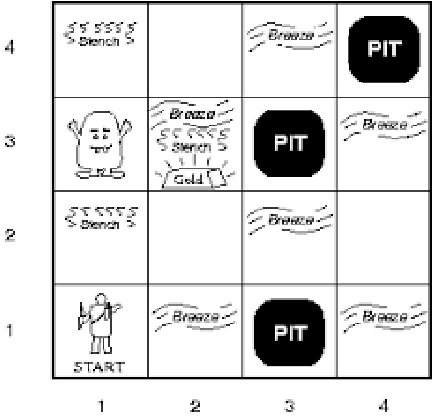
Logique propositionnelle

Rym Guibadj

LISIC, ULCO, EILCO

Un premier exemple : le monde du Wumpus

- Environnement
  - Grille 4x4
  - Depart en [1,1]
- Effecteurs
  - Pivoter droite
  - Pivoter gauche
  - Avancer
  - Saisir
  - Tirer
- Capteur
  - Odeur (Wumpus)
  - Brise (Puits)
  - Lueur (or)
  - Choc (mur)
  - Cri (Mort)
- But
  - Trouver l'or et sortir



Un premier exemple : conclusion

Propriété

Si l'agent tire une conclusion en s'appuyant sur les informations disponibles, alors cette conclusion est garantie correcte si les informations sont elles-mêmes correctes

Suite du cours

Comment construire un tel agent ?

Logique propositionnelle

- Formules propositionnelles
  - Définies à l'aide de **constantes**, de **variables** et de **connecteurs**
- Constantes
  - **vrai** ou **faux**
- Variables
  - Ensemble dénombrable
  - Représentées par des lettres : **p, q, r, s** ...
- Connecteurs
  - Unaire : $\neg$ not
  - Binaire : $\wedge$ (et), $\vee$ (ou), $\Rightarrow$ (implique), $\Leftrightarrow$(ssi)
  - Priorité : $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$ (parenthèses possibles)
- Exemple :
  - $p \wedge q \vee \neg r \Rightarrow Vrai$

Sémantique de la LP

- But de la sémantique
  - Associer une signification aux formules
  - Définir la valeur de vérité (vrai ou faux)
- Table de vérité

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
faux	faux					
faux	vrai					
vrai	faux					
vrai	vrai					

Sémantique de la LP

- But de la sémantique
  - Associer une signification aux formules
  - Définir la valeur de vérité (vrai ou faux)
- Table de vérité

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
faux	faux	vrai	faux	faux	vrai	vrai
faux	vrai	vrai	faux	vrai	vrai	faux
vrai	faux	faux	faux	vrai	faux	faux
vrai	vrai	faux	vrai	vrai	vrai	vrai

Sémantique de la LP

Validité

Une proposition est valide si elle est vraie pour toutes les assignations

Satisfiabilité

Une proposition est satisfiable si elle est vraie pour au moins une assignation

Exemple

Soit la formule : $F = (\neg p \Rightarrow q) \wedge (p \Rightarrow r)$
F est valide ? F est satisfiable ?

Sémantique de la LP

Exemple

Soit la formule : $F = (\neg p \Rightarrow q) \wedge (p \Rightarrow r)$
F est valide ? F est satisfiable ?

p	q	r	$\neg p$	$\neg p \Rightarrow q$	$p \Rightarrow r$	$(\neg p \Rightarrow q) \wedge (p \Rightarrow r)$
faux	faux	faux	vrai	faux	vrai	faux
faux	faux	vrai	vrai	faux	vrai	faux
faux	vrai	faux	vrai	vrai	vrai	vrai
faux	vrai	vrai	vrai	vrai	vrai	vrai
vrai	faux	faux	faux	vrai	faux	faux
vrai	faux	vrai	faux	vrai	vrai	vrai
vrai	vrai	faux	faux	vrai	faux	faux
vrai	vrai	vrai	faux	vrai	vrai	vrai

$\Rightarrow$

F est satisfiable mais non valide

Conséquence logique

Définition

- Soit $E = F_1, ..., F_n$, un ensemble non vide de formules et G une formule
- G est une conséquence logique de E ($F_1, ..., F_n \models G$) ssi toutes les interprétations pour lesquelles $F_1, ..., F_n$ sont vraies, G est vraie

Observations

- $F_1, ..., F_n \models G$ ssi $(F_1 \wedge .... \wedge F_n) \Rightarrow G$ est valide
- si E est insatisfiable alors $F_1, ..., F_n \models G$
- si G est valide, elle est C. L. de n'importe quel ensemble de formule
- $E \models G$ ssi $E \cup \neg G$ est insatisfiable

Exemple : conséquences logiques de F ?

- $F = (p \Rightarrow q) \wedge (p \vee r)$
- $G = q \vee r$
- $H = p \Rightarrow r$

Exemple : conséquences logiques de F ?

- $F = (p \Rightarrow q) \wedge (p \vee r)$
- $G = q \vee r$
- $H = p \Rightarrow r$

p	q	r	F	G	H
faux	faux	faux	faux	faux	vrai
faux	faux	vrai	vrai	vrai	vrai
faux	vrai	faux	faux	vrai	vrai
faux	vrai	vrai	vrai	vrai	vrai
vrai	faux	faux	faux	faux	faux
vrai	faux	vrai	faux	vrai	vrai
vrai	vrai	faux	vrai	vrai	faux
vrai	vrai	vrai	vrai	vrai	vrai

- $F \models G$
- H n'est pas conclusion logique de F

Conséquence logique

Notion

La notion de conséquence logique permet de vérifier certains raisonnements formulés en langage naturel

Exemple

”Si je vous paie mon installation TV c’est qu’elle marche. Or, elle ne marche pas, donc, je ne vous paierai pas”

Conséquence logique

Notion

La notion de conséquence logique permet de vérifier certains raisonnements formulés en langage naturel

Exemple

”Si je vous paie mon installation TV c’est qu’elle marche. Or, elle ne marche pas, donc, je ne vous paierai pas”

On formalisme :

- p : ”je vous paierai votre installation TV”
- m : ”mon installation marche”

On écrit le raisonnement : $\{p \Rightarrow m, \neg m\} \models \neg p$
$p \Rightarrow m$ et $\neg m$ sont les prémisses et $\neg p$ la conclusion

Equivalence logique

Equivalence logique

si $F \models G$ et $G \models F$ alors F et G sont logiquement équivalentes
$F \Leftrightarrow G$

Quelques équivalences

- $F \Leftrightarrow G \equiv (F \Rightarrow G) \wedge (G \Rightarrow F)$
- $F \Rightarrow G \equiv \neg F \vee G$
- $F \wedge G \equiv G \wedge F$
- $F \vee G \equiv G \vee F$
- $\neg(F \wedge G) \equiv \neg F \vee \neg G$
- $\neg(F \vee G) \equiv \neg F \wedge \neg G$
- $(F \wedge G) \vee H \equiv (F \vee H) \wedge (F \vee H)$
- $(F \vee G) \wedge H \equiv (F \wedge H) \vee (F \wedge H)$
- $F \Rightarrow G \equiv \neg G \Rightarrow \neg F$

Récapitualtif

La logique est un langage formel permettant de représenter l'information de sorte qu'on puisse en déduire des conclusions, et donc de nouvelles informations

Syntaxe

définit l'alphabet et la structure des phrases du langage

Sémantique

définit le sens des phrases, la vérité d'un énoncé

D'un point de vue d'un agent en IA

Un agent dispose d'une base de connaissance et d'un moyen de faire des déductions pour choisir ses actions

Base de connaissances

Un ensemble de faits ou de formules vraies

Introduction à la déduction

- Notations
  - $p_{i,j}$ est vrai s'il y a un puits en $[i, j]$
  - $b_{i,j}$ est vrai s'il y a une brise en $[i, j]$
- Connaissance
  - $R_1 : \neg p_{1,1}$
  - $R_2 : \neg b_{1,1}$
  - $R_3 : b_{1,2}$
  - $R_4 : b_{1,1} \Leftrightarrow (p_{1,2} \vee p_{2,1})$
  - $R_5 : b_{1,2} \Leftrightarrow (p_{1,1} \vee p_{2,2} \vee p_{1,3})$
- Objectif
  - On veut savoir s'il y a un puits en $[2, 1], [2, 2], [1, 3]$
  - On cherche la vérité de $\neg p_{2,1}, \neg p_{2,2}, \neg p_{1,3}$

Méthodes de preuves

- Vérification de modèles
  - Enumération (par table de vérité)
  - Par réfutation
  - Algorithme de résolution
- Application de règles d'inférence.
  - Génération de nouvelles formules à partir de formules vraies.
  - Preuve : séquence d'application de régles d'inférence

Démonstration par réfutation

Principe

- Une démonstration par l'absurde
- Pour montrer $KB \models q$ on peut montrer que : $KB \cup \neg q \Rightarrow \emptyset$
- Montrer que $KB \wedge \neg q$ n'est pas valide

Remarques

- Il faut vérifier toutes les interprétations possibles donc 2^n cas si on a n variables propositionnelles
- Algorithme de complexité exponentiel $\Rightarrow$ en pratique peu utilisable

Algorithme de résolution

Règle de résolution

A partir de $(p_1 \vee \ldots \vee p_m)$,$(q_1 \vee \ldots \vee q_n)$, si $p_i = \neg q_j$, on déduit :
$(p_1 \vee \ldots \vee p_{i-1} \vee p_{i+1} \vee \ldots \vee p_m \vee q_1 \vee \ldots \vee q_{j-1} \vee q_{j+1} \vee \ldots \vee q_n)$

Principe

Pour montrer $\{F_1, \ldots, F_n\} \models G$, il faut :

- Mettre la formule $F_1 \wedge \ldots \wedge F_n \wedge \neg G$ sous forme normale conjonctive.
- Appliquer la règle de la résolution sur les clauses jusqu'à ce que on trouve la clause vide ϕ

Règles d'inférence

Principe

- Appliquer un schéma de règles qui conduisent au but recherché
- $F_1, ..., F_n/p$: à partir de $F_1, ..., F_n$ on peut montrer p

Quelques règles d'inférence

$p \Rightarrow q, p/q$	(le modus ponens)
$p \wedge q/q$	(élimination de la conjonction)
$p \Leftrightarrow q/(p \Rightarrow q) \wedge (q \Rightarrow p)$	(élimination de l'équivalence)
$(p \Rightarrow q) \wedge (q \Rightarrow p)/p \Leftrightarrow q$	(élimination de la double implication)