Chapitre III:

Représentation des Connaissances

Plan

- Types & Nature des connaissances
- Représentation des connaissances
  - Logique des prédicats
  - Réseaux sémantiques
  - Frames
  - …..

Plan

- Types & Nature des connaissances
- Représentation des connaissances
  - Logique des prédicats
  - Réseaux sémantiques
  - Frames
  - …..

Types de Connaissances

- Connaissance causale
- Connaissance déclarative
- Connaissance procédurale
- Connaissance stratégique

Types des connaissances

- Connaissance Causale:

Connaissance qu'on peut utiliser pour une inférence considérée comme explicative par l'utilisateur.

<u>Exemple:</u> fermeture de la porte dont la serrurerie est rouillée.

Types de Connaissances

- Connaissances déclaratives :
  - connaissances théoriques reconnues par une communauté de pratique à un moment déterminé,
  - s'expriment le plus souvent sous forme de règles
  - indique ce qu'il faut faire sans préciser comment le faire
- Exemple: Dans le code de la route, on doit se tenir sur le côté droit de la chaussée

  Mais pas comment le faire

Types de Connaissances

- Connaissances procédurales:
  - correspondent, aux savoir-faire.
  - s'actualisent dans des séquences d'actions et répondent à la question « comment faire ? ».
  - c'est un savoir *algorithmique*

Exemple:

- Procédure de diagnostic d'une panne
- Procédure de correction d'erreur dans un prog.
- Proc. de contrôle d'un processus industriel
- Proc. de dépannage d'un circuit électronique

Types de Connaissances

- Connaissances stratégiques:
  - liées à l'opportunité ou à la nécessité d'utiliser un savoir, un savoir-faire, une stratégie
  - Indiquent comment et quand utiliser la connaissance du domaine en fonction d'un but à accomplir
  - Elles sont en relation avec les questions « quand ? » et « pourquoi ? »

- Elles doivent permettre notamment d’adapter les stratégies de résolution à une situation ou à une tâche
- elles sont parfaitement contextualisées

- Exemple:

- Il est utile de commencer par la 1ère règle puis passer à la 2ème règle

⇒ Méta-connaissance

Nature des connaissances

Connaissance évolutive

Connaissance approximative

Connaissance de définition

Connaissance d'exception

Connaissance de proximité, de transitivité

Métaphore

Nature des connaissances

- Connaissance de définition :

<u>*Exemple:*</u> *« Un quadrilatère est un polygone ayant* exactement quatre côtés »

$\Rightarrow$ *Fait certain*

Nature des connaissances

- Connaissance évolutive :

<u>*Exemple:*</u> *« Sami est un étudiant en 3Ing GLSI»*

⇒ Fait certain, valeur de vérité peut être modifiée

Nature des connaissances

- Connaissance incertaine, approximative:

Exemple: «Il est probablement vrai que ...»

⇒ *Fait incertain, comment gérer l'imprécision?*

Nature des connaissances

- Connaissance d'exception :

Exemple: « les oiseaux volent sauf les autruches »

$\Rightarrow$ *comment modéliser l'exception?*

Nature des connaissances

- Connaissance Métaphore :

Exemple: « Le roi des animaux »

$\Rightarrow$ *suggestion d'informations*

Nature des connaissances

- Connaissance de proximité, continuité, transitivité

Exemple:

- *« Un tas de sable auquel on enlève un grain de* sable, reste un tas de sable »
- « Si X est voisin de Y et Y est voisin de Z alors X est voisin de Z »

⇒ Nécessité d'une bonne représentation pour une mise à jour facile et une bonne exploitation

Questions :

- Quelles sont les représentations possibles ?
- Quelle représentation choisir?

Bonne représentation

- Pouvoir représenter tout ce qui est nécessaire de façon concise et efficace.
- Etre manipulable pour déduire d 'autres connaissances.
- Facilité d'ajout de connaissances.
- Problèmes liés: représentation et utilisation

- Une représentation de connaissances : c’est un système définissant une série de symboles et une série d'opérations sur ces symboles.
- La représentation des connaissances est un problème qui se pose à tout système expert.
- Le choix d’une bonne représentation est fondamental et détermine toute la suite du développement du système.

Différentes Représentations

- Aspects Relationnel :
  - Représentation Logique
  - Règles de production

- Aspect Procédural :
  - Représentation Procédurale

- Aspect Objet :
  - Réseaux sémantiques
  - Frames
  - Dépendances conceptuelles

Différentes Représentations

- Aspects Relationnel :
  - Représentation Logique
  - Règles de production
- Aspect Procédural :
  - Représentation Procédurale
- Aspect Objet :
  - Réseaux sémantiques
  - Frames
  - Dépendances conceptuelles

Représentation Logique

- Logique Propositionnelle (ou de prédicats d'ordre 0 – LP0)
- Logique de Prédicats d'ordre 1 - LP1

Logique Propositionnelle – LP0

- Etude des relations entre des énoncés appelés propositions ou encore des formules.
- Ces relations peuvent être exprimées par l'intermédiaire de connecteurs logiques

Logique Propositionnelle – LP0
Syntaxe

Alphabet

- Variables : $V = \{p, q, r, \ldots\}$ ensemble dénombrable de lettres appelées variables propositionnelles. ⇒ propositions atomiques
- Constantes : vrai, faux.
- Connecteurs logiques : $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\equiv$
- Séparateurs : (,)

Logique Propositionnelle – LP0
Syntaxe

Formules Bien Formées (FBF)

- Peut être atomique ou complexe

- Règle 1: les propositions atomiques : p, q, r,…
- Règle 2: si A est une FBF alors ¬A est une FBF
- Règle 3: en utilisant les connecteurs logiques

Logique Propositionnelle – LP0
Syntaxe

- Si *E* et *F* sont des FBF, *E* $\wedge$ *F* est une FBF (*conjonction*)
- Si *E* et *F* sont des FBF, *E* $\vee$ *F* est une FBF (*disjonction*)
- Si *E* et *F* sont des FBF, *E* $\Rightarrow$ *F* est une FBF (*implication*)
- Si *E* et *F* sont des FBF, *E* $\Leftrightarrow$ *F* est une FBF (*équivalence*)

Logique Propositionnelle – LP0
Syntaxe

- Exemple:

F1: $(((\neg p \leftrightarrow q) \vee \neg(r \wedge s)) \rightarrow q)$

F2 : $p\neg qr \rightarrow t($

Ces formules sont elles des FBF ??

Logique Propositionnelle – LP0
Sémantique

- S'intéresser à la sémantique: c'est déterminer la valeur de vérité d'un énoncé,

  ⇒ On parle de l'interprétation d'une formule

- Interprétation : il s'agit d'affecter une valeur vrai ou faux à chacune des variables propositionnelles qui la compose.

- Pour une formule à n variables, il y a 2^n mondes possibles.

Logique Propositionnelle – LP0
Sémantique

- La valeur de vérité d'une FBF est déterminé en fonction des atomes qui la composent

Exemple:

Table de vérité des connecteurs logiques:

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

Logique Propositionnelle – LP0
Consistance / Validité

- Modèle : une interprétation pour laquelle une formule est vraie.

  Exemple: I(p = faux, r = vrai, s = faux) est un modèle de $(p \rightarrow (r \wedge s))$.

- Consistance : une formule A est consistante, ou satisfiable, si et seulement si A a un modèle.

  $(p \rightarrow (r \wedge s))$ est une formule consistante.

- Inconsistance : (ou insatisfiable) une formule A est inconsistante si elle n'a pas de modèle

Logique Propositionnelle – LP0
Consistance / Validité

- Validité : A est dite valide si elle est vraie pour n'importe quelle interprétation de ses variables propositionnelles

$\Rightarrow$ tautologie

Logique Propositionnelle – LP0
Conséquence Logique

- Le principe du raisonnement logique : c'est l'idée qu'un énoncé découle logiquement d'un autre énoncé.

    $\Rightarrow$ conséquence logique

Logique Propositionnelle – LP0
Conséquence Logique

Exemple :

« vous savez qu'à la dernière séance du cours vous aurez un contrôle.

Vous savez également qu'en général un contrôle aboutit à une note.

Vous en déduisez logiquement que vous aurez une note à la fin du cours. »

$$\frac{\text{Contrôle},\ \text{Contrôle} \rightarrow \text{Note}}{\text{Note}}$$

Logique Propositionnelle – LP0
Conséquence Logique

Définition

- Soit $\{F1, \ldots, Fn\}$ un ensemble de formules.
- Une formule C est une conséquence logique de l'ensemble $\{F1, \ldots, Fn\}$, noté $\{F1, \ldots, Fn\} \models C$,

ssi tout modèle de $\{F1, \ldots, Fn\}$ est un modèle de C.

- Exemple : $\{Contrôle, Contrôle \rightarrow Note\} \models Note$.

Logique Propositionnelle – LP0
Principe de Résolution

- Théorème de déduction

C est une formule déduite de F

$F \models C$ ssi $\models (F \rightarrow C)$ est une tautologie

- Théorème de réfutation

C est une formule déduite de F

$F \models C$ ssi $F \cup \{\neg C\}$ est inconsistant

$\Rightarrow$Ce théorème découle du premier

Logique Propositionnelle – LP0
Résolution par déduction

- Règles d'inférence: 2 règles particulière
  - Modus ponens: si A est vraie et $A \rightarrow B$ alors B est vraie

  Dans l'exemple :

  {Contrôle, Contrôle $\rightarrow$ Note} |= Note,

  le schéma utilisé s'appelle *modus ponens*

  - Modus tollens : si $\neg B$ est vraie et $A \rightarrow B$, alors $\neg A$ est vraie

Logique Propositionnelle – LP0
Résolution par déduction

- En général: le principe de résolution s'applique à un ensemble de clauses.

- Une clause est une disjonction de littéraux ($p \vee \neg q \vee r$). On note $\perp$ la clause vide.

  Un littéral est une variable prop. ou sa négation.

- La résolvante de deux clauses : c'est la clause résultant de l'application de la règle de résolution sur ces deux clauses.

- D’une manière générale, le principe de résolution est basée sur la règle de résolution par déduction suivante :

Si $X \vee A$ et $\neg X \vee B$ alors $A \vee B$ peut être déduite

$$\frac{X \vee A, \neg X \vee B}{A \vee B}$$

Logique Propositionnelle – LP0
Résolution par déduction

- Exemple:
  - Julien est à la Maison ou il est au Cinéma,
  - Julien n'est pas à la Maison ou il est au Travail
- Formellement:
  - Cinema v Maison
  - Travail v ¬Maison
- La résolvante de ces deux clauses est alors
  - Cinema v Travail

Logique Propositionnelle – LP0
Résolution par déduction

<u>Cas général</u>

- Le calcul de la résolvante est l'unique règle d'inférence pour le principe de résolution.
- Pour cela, une transformation en forme Normale Conjonctive (FNC)est obligatoire: Conjonction de clauses

Logique Propositionnelle – LP0
Résolution par déduction

Exemple

- **Résolution d'une énigme par la logique des propositions.**

Vous êtes perdus sur une piste dans le désert.

Vous arrivez à une bifurcation. Chacune des deux pistes est gardée par un sphinx que vous pouvez interroger.

Les pistes peuvent soit conduire à une oasis, soit se perdre dans le désert profond (au mieux, elle conduisent toutes à une oasis, au pire elles se perdent toutes les deux).

- A. *Le sphynx de droite vous répond : « Une au moins des deux pistes conduit à une oasis. »*
- B. *Le sphynx de gauche vous répond : « La piste de droite se*
- *perd dans le désert. »*
- C. *Vous savez que les sphynx disent tous les deux la vérité, ou*
- *bien mentent tous les deux.*

- On pose :
  D : « Il y a une oasis au bout de la route de droite. »
  G : « Il y a une oasis au bout de la route de gauche. »
- Ainsi :
  - $A = D \vee G$
  - $B = \neg D$
  - $A \leftrightarrow B$ (formule F qu'on cherche à vérifier)

- Mise sous forme normale conjonctive :

$$
\begin{aligned}
F &= A \leftrightarrow B \\
&= (A \rightarrow B) \wedge (B \rightarrow A) \\
&= ((D \vee G) \rightarrow \neg D) \wedge (\neg D \rightarrow (D \vee G)) \\
&= (\neg(D \vee G) \vee \neg D) \wedge (\neg\neg D \vee (D \vee G)) \\
&= ((\neg D \wedge \neg G) \vee \neg D) \wedge (D \vee (D \vee G)) \\
&= (\neg D \vee \neg D) \wedge (\neg G \vee \neg D) \wedge (D \vee D \vee G) \\
&= (\neg D) \wedge (\neg D \vee \neg G) \wedge (D \vee G) \\
&= (\neg D) \wedge (D \vee G)
\end{aligned}
$$

- L'ensemble de clauses obtenu: $C = \{\neg D, D \vee G\}$.
- L'application de la règle de résolution nous indique que la route de gauche conduit effectivement
- à une oasis :

$$\frac{\neg D, D \vee G}{G}$$

Logique Propositionnelle – LP0
Résolution par réfutation

- Les formules doivent être en forme conjonctive normale

- Une formule C est la conséquence logique d'un ensemble de formules :

$F = \{H1, \ldots, Hn\}$ ssi $F \cup \{\neg C\}$ est inconsistant

- F et $\neg C$ doivent être en FNC

Logique Propositionnelle – LP0
Résolution par réfutation

- Algorithme de résolution

1. Choisir deux clauses dont on peut calculer la résolvante et la calculer

effectivement.

2. Si La résolvante produite est la clause vide : dans ce cas, F a pour conséquence logique C.
3. Sinon revenir à 1

Logique Propositionnelle – LP0
Résolution par réfutation

__Exemple:__

- *F1: Si Didier est l'auteur de ce bruit, il est stupide ou dépourvu de principes.*
- *F2: Didier n'est ni stupide ni dépourvu de principes.*
- *F3: Didier n'est pas l'auteur de ce bruit.*

Montrer par réfutation que F3 est déduite logiquement de F1 et F2

Exemple Considérons l'exemple de Didier. On dispose des connaissances suivantes : $D \rightarrow (S \vee P)$, $\neg S$ et $\neg P$. Ramené sous forme clausale, on a $\{\neg D \vee S \vee P, \neg S, \neg P\}$. On cherche à déduire $\neg D$. On en prend donc la négation, soit D, et on vérifie la consistance de $\{\neg D \vee S \vee P, \neg S, \neg P, D\}$.

$$\dfrac{\dfrac{\dfrac{\neg D \vee S \vee P \qquad \neg S}{\neg D \vee P} \qquad \neg P}{\neg D} \qquad D}{\bot}$$

Logique Propositionnelle – LP0

- Inconvénients:
  - Absence de quantificateurs
  - Absence de variables
  - Absence de fonctions

$\Rightarrow$ Peu expressive

Logique Propositionnelle – LP0

- Par exemple la formalisation du raisonnement suivant lui échappe :

  – *Certains étudiants assistent à tous les cours.*
  – *Aucun étudiant n'assiste à un cours inintéressant.*
  – *Le cours d'IA est intéressant*

Peut-on conclure que tous les cours sont intéressants ?
Peut-on conclure que les étudiants assistent au cours d'IA ?

Représentation Logique

- Logique Propositionnelle (ou de prédicats d’ordre 0 – LP0)
- Logique de Prédicats d’ordre 1- LP1

Logique des prédicats d'ordre1 – LP1

- Plus expressive
- Construite à partir de LP0
- S'inspire du langage naturel pour définir les entités du monde réel, à savoir :
  - Objets
  - Relations
  - Fonctions

Logique des prédicats d'ordre1 – LP1

- Objets : personnes, maisons, nombres, couleurs, match de foot, guerres…
- Relations :
  - relations unaires ou propriétés : rouge, arrondi, faux, premier…
  - relations *n-aires* : frère-de, plus-grand-que, est-de-couleur, possède…
- Fonctions : une seule "valeur" pour une "entrée" donnée : père de, meilleur ami, …

Logique des prédicats d'ordre1 – LP1

- Syntaxe:
  - Constantes : 2, *Jean*, *X*, ...
  - Prédicats : *Frere*, >, *Avant*, ...
  - Fonctions : *RacineCarre*, ...
  - Variables : *x*, *y*, *a*, *b*, ...
  - Connecteurs : $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$
  - Egalité : =
  - Quantificateurs : $\forall$, $\exists$

Logique des prédicats d'ordre1 – LP1

- Terme :
  - Constante,
  - Variable
  - Fonction($t1, \ldots, tn$) où $t1, \ldots tn$ sont des termes

- Atome : Prédicat($t1, \ldots, tn$) où $t1, \ldots tn$ sont des termes

Exemple :

Frere(Richard, Jean)

Marie(Pere(Richard), Mere(Jean))

Logique des prédicats d'ordre1 – LP1

- Les formules composées (ou FBF) : construites à partir des:
  - atomes,
  - des quantificateurs,
  - et des connecteurs logiques

Exemples:

- $\neg E, E1 \wedge E2, E1 \vee E2, E1 \Rightarrow E2, E1 \Leftrightarrow E2$
- *Frere(John, Richard)* $\wedge$ *Frere(Richard, John)*
- $> (1,2) \vee \leq (1,2)$

Logique des prédicats d'ordre1 – LP1

- La valeur de vérité d'une formule est déterminée par une interprétation des symboles :
  - Symboles de constantes/variables → objets
  - Symboles de prédicats → relations
  - Symboles de fonctions → fonctions

Logique des prédicats d'ordre1 – LP1

- Quantificateur Universel
  - Syntaxe:

    ∀<*variables*> <*formule*>

  - Exemple: Tous les étudiants sont intelligents :

$$\forall x \; Etudiant(x) \Rightarrow Intelligent(x)$$

- *∀xP est vrai dans une interprétation ssi P est vrai pour tous les objets x*

Logique des prédicats d'ordre1 – LP1

- *$\forall x$ P est équivalent à la conjonction de toutes les instanciations de P* **:**

$$
\begin{aligned}
& \mathit{Etudiant(Paul)} \Rightarrow \mathit{Intelligent(Paul)} \\
\wedge\ & \mathit{Etudiant(Pierre)} \Rightarrow \mathit{Intelligent(Pierre)} \\
\wedge\ & \mathit{Etudiant(Sophie)} \Rightarrow \mathit{Intelligent(Sophie)} \\
\wedge\ & \mathit{Etudiant(Julie)} \Rightarrow \mathit{Intelligent(Julie)} \\
\wedge\ & \\
& \ldots
\end{aligned}
$$

Logique des prédicats d'ordre1 – LP1

- Le connecteur principal à utiliser avec ∀ est l'implication ⇒
- *Erreur fréquente* : utiliser la conjonction ∧ comme connecteur principal avec ∀

$$\forall x\ Etudiant(x) \wedge Intelligent(x)$$

signifie

"*tout le monde est étudiant et est* intelligent"

Logique des prédicats d'ordre1 – LP1

- Quantificateur Existentiel
  - Syntaxe:

    $\exists$<*variables*> <*formule*>

  - Exemple: Un étudiant est intelligent :

    $\exists x\ Etudiant(x) \wedge Intelligent(x)$
- $\exists xP$ est vrai dans une interprétation ssi P est vrai pour un objet x

Logique des prédicats d'ordre1 – LP1

- $\exists x\ P$ est équivalent à la *disjonction de toutes les instanciations de P* :

$$
\begin{aligned}
& Etudiant(Paul) \wedge Intelligent(Paul) \\
\vee\ & Etudiant(Pierre) \wedge Intelligent(Pierre) \\
\vee\ & Etudiant(Sophie) \wedge Intelligent(Sophie) \\
\vee\ & Etudiant(Julie) \wedge Intelligent(Julie) \\
\vee\ &
\end{aligned}
$$

Logique des prédicats d'ordre1 – LP1

- Le connecteur principal à utiliser avec ∃ est la conjonction ∧

- *Erreur fréquente* : utiliser l'implication ⇒comme connecteur principal avec ∃

$$\exists x\ (Etudiant(x) \Rightarrow Intelligent(x))$$

est vrai s'il existe quelqu'un qui n'est pas étudiant!

Logique des prédicats d'ordre1 – LP1

Propriétés des quantificateurs

- $\forall x \forall y \Leftrightarrow \forall y \forall x$
- $\exists x \exists y \Leftrightarrow \exists y \exists x$
- $\exists x \forall y \neq \forall y \exists x$
  - $\exists y \forall x$ *Aime(x,y) "Il existe une personne qui est aimé par tout le monde"*
  - $\forall x \exists y$ *Aime(x,y) "Tout le monde aime quelqu'un" (pour toute personne, il* existe quelqu'un qu'il aime)

Logique des prédicats d'ordre1 – LP1

- Liens entre ∀ et ∃ : ils sont liés par la négation :

  - $\forall x$ *Aime (x,Glace)* « *tout le monde aime Glace* »
  - $\neg\exists x \neg$*Aime(x,Glace)* « *il n'existe pas qq'un qui n'aime..* »

  $\Rightarrow$ équivalentes

  - $\exists x$ *Aime(x,Brocoli)* « *il y a qq'un qui aime Brocoli* »
  - $\neg\forall x \neg$*Aime(x,Brocoli)* « *ce n'est pas tt le monde qui n'aime pas ..* »

  $\Rightarrow$ équivalentes

Logique des prédicats d'ordre1 – LP1

- Interprétation d'une formule F:

pour un domaine non vide et fini de valeurs D, c'est une affectation de :

  - Un élément de D à chaque variable
  - D'une fonction $D^n \rightarrow D$ pour chaque fonction n-aire
  - D'une valeur booléene {T, F} pour chaque proposition
  - D'une application $D^n \rightarrow \{T, F\}$ pour chaque prédicat n-aire

Logique des prédicats d'ordre1 – LP1

- <u>Exemple 1</u> :

  Soit la formule :$(\forall x)\ H(x)$

  et soit l'interprétation suivante :

  $D = \{1, 2\}, H(1) = F, H(2) = T$

- <u>Exemple 2</u> :

  Soit la formule :$(\forall x)\ (R(x) \rightarrow G(X, a))$

  et soit l'interprétation suivante :

  $D = \{1, 2\}, a = 2, R(1) = F, R(2) = T$

  $G(1, 2) = F, G(2, 2) = F$

Logique des prédicats d'ordre1 – LP1

- Le principe de résolution n'est possible qu'avec des FBF transformée en forme clausale

- Clause: la disjonction de littéraux ou de conjonctions
  - Exemple:

    $A \vee D \vee \neg K$ est une clause

Logique des prédicats d'ordre1 – LP1

- Forme clausale: comment l'obtenir?
  1. Éliminer les symboles : implication → et équivalence ↔
  2. Accoler la négation aux atomes
  3. Distinguer entre les variables $(\exists x)G(x) \wedge (\forall x) Q(x)$
  4. Mettre les quantificateurs à gauche
  5. Eliminer ∃ (par skolémisation)
  6. Eliminer ∀
  7. Mettre sous forme de conjonction de disjonction
  8. Eliminer les symboles de conjonction et les remplacer par ,

Logique des prédicats d’ordre1 – LP1

- ***Skolémisation*** : Remplacer les quantificateurs existentiels par des fonctions de Skolem (nouvelles constantes ou nouvelles fonctions d'arité non nulle) en procédant de la gauche vers la droite.

Logique des prédicats d'ordre1 – LP1

Conséquence Logique

- soient F1, F2, …Fn et G des formules.

- G est une conséquence logique de F1, F2, …Fn

ssi

Tout modèle de F1, F2, …Fn est modèle de G.

Notation : F1, F2, …Fn $\models$ G est valide

ssi

(F1, F2, …Fn)^ ˥G est inconsistante

(clause vide□)

Logique des prédicats d'ordre1 – LP1

- Durant la résolution par réfutation, l'unification des termes se fait uniquement en remplaçant:
  - Une variable par une variable
  - Une variable par une constante
  - Une variable par une fonction

Logique des prédicats d'ordre1 – LP1

- Exemple:
- On suppose les clauses suivantes :
  - C1: $\neg p(X) \lor p(Y)$
  - C2: $p(Z) \lor q(Z)$
  - C3: $\neg p(C)$
  - C4: $q(D)$
- Montrer que $\{C1,C2,C3\} \models C4$

Logique des prédicats d'ordre1 – LP1

⇒ il suffit de montrer que :

{C1,C2,C3} v ¬ C4 est inconsistant

Résolution par réfutation : il faut trouver la clause vide

Différentes Représentations

- Aspects Relationnel :
  - Représentation Logique
  - Règles de production
- Aspect Procédural :
  - Représentation Procédurale
- Aspect Objet :
  - Réseaux sémantiques
  - Frames
  - Dépendances conceptuelles

Règles de production

- **Syntaxe:**

Si conditions alors conséquence

$\Rightarrow$ Les conditions sont des conjonctions de faits ou de prédicats qui ont une valeur logique (vrai ou fausse).

Règles de production

- **Exemples:**

Si la température du réacteur dépasse 800°C alors

descendre les barres de contrôle.

Si X est un chien alors X est un mammifère.

Règles de production

Faits :

H et K sont vrais

Règles:

- R1: Si A alors E
- R2: Si B alors D
- R3: Si H alors A
- R4: Si E et G alors C
- R5: Si E et K alors B
- R6: Si E et D et K alors C
- R7: Si G et F et K alors A

Différentes Représentations

- Aspects Relationnel :
  - Représentation Logique
  - Règles de production
- Aspect Procédural :
  - Représentation Procédurale
- Aspect Objet :
  - Réseaux sémantiques
  - Frames
  - Dépendances conceptuelles

Représentation Procédurale

- C'est la première et la plus connue des techniques de programmation utilisée pour implanter des algorithmes.
- Offre un certain nombre de structures de contrôle prédéfinies : la séquence, le test, la boucle.
- Les représentations procédurales sont liées à la description des conditions d'exécution d'une tâche (la réponse à la question « comment ? »)
- La force des travaux dans les représentations procédurales était de tenter de trouver les moyens pour exprimer l'information de contrôle.

Différentes Représentations

- Aspects Relationnel :
  - Représentation Logique
  - Règles de production
- Aspect Procédural :
  - Représentation Procédurale
- Aspect Objet :
  - Réseaux sémantiques
  - Frames
  - Dépendances conceptuelles