

## Logique

### Objectifs

Traiter formellement les notions de vérité et fausseté

Formaliser le "raisonnement logique", la "déduction logique"

### En S.I.

Spécifier les systèmes : invariants, conditions de fonctionnement, pré et post conditions (algorithmes).

Raisonner sur les systèmes : détecter les incohérences, prouver des propriétés.

Raisonner sur les données : interprétation logique des bases de données, logique pour l'interrogation de bd, expression des contraintes d'intégrité.

Systèmes basés sur la connaissance, systèmes experts, déduction automatique, bases de données déductives.

## Logique des propositions

Une logique parmi d'autres

Composée de

- un langage (écriture des formules)
- une sémantique (évaluation des formules)
- un système de preuve (axiomes, règles d'inférence)

## Aspect déductif / syntaxique

On s'intéresse aux preuves de certains énoncés

De

« quand il pleut Paul prend toujours son parapluie »

et

« aujourd'hui Paul a son parapluie »

peut-on conclure

« aujourd'hui il pleut » ?

## Aspect sémantique

On évalue la vérité ou fausseté de certains énoncés.

Est-il possible que

« Paul mange du chocolat » soit faux

et que simultanément

« Paul mange du pain ou du chocolat »

soit vrai ?

Les éléments de base de la logique sont les variables propositionnelles (également appelées atomes).

$p$  = « Paul aime le chocolat »,

$q$  = « Il pleut »

## Syntaxe des formules de logique propositionnelle

### Vocabulaire

- des symboles :  $\wedge, \vee, \Rightarrow, \Leftrightarrow, \neg, (, )$
- de symboles de variables propositionnelles :  $a, b, c, \dots, p, q, r, s$ , etc.

### Règles de grammaire

Formule  $\rightarrow$  Variable

Formule $\wedge$ Formule	(et)
Formule $\vee$ Formule	(ou)
Formule $\Rightarrow$ Formule	(implique)
Formule $\Leftrightarrow$ Formule	(implique bidirectionnel)
$\neg$ Formule	(non)
(Formule )	

Variable  $\rightarrow a \mid b \mid \dots \mid x \mid y \mid z$

## Exemples

$p$   
 $q \Rightarrow p$   
 $\neg \neg p$   
 $\neg (p \vee (r \wedge \neg s) \vee t)$   
 $(q \Rightarrow p) \Leftrightarrow (s \Rightarrow p)$

## Ambiguïtés

Deux manière d'analyser

$p \wedge q \vee r$

On utilise des parenthèses pour lever toute ambiguïté :

$(p \wedge q) \vee r$

n'a qu'un seul arbre syntaxique

## Sémantique

Objectif : attribuer une valeur logique **vrai** (v) ou **faux** (f) à chaque formule

= calculer l'**interprétation** de chaque formule

Il faut **fixer** une interprétation de chaque variable propositionnelle (atome).

Fonction  $I$  de  $\{p_1, p_2, \dots, p_n\}$  dans  $\{v, f\}$ .

$I(p_i)$  est la **valeur de vérité** (v ou f) attribuée à  $p_i$ .

## Interprétation d'une formule :

Extension de  $I$  aux formules :

$I(P \wedge Q) = v$  si  $I(P)$  **et**  $I(Q)$  sont  $v$ ,  
 $f$  sinon

$I(P \vee Q) = v$  si  $I(P)$  **ou**  $I(Q)$  est  $v$ ,  
 $f$  s'il sont les deux  $f$

$I(P \Rightarrow Q) = v$  si  $I(P)$  est  $f$  **ou bien** si  $I(Q)$  est  $v$ ,  
 $f$  sinon

$I(P \Leftrightarrow Q) = v$  si  $I(P) = I(Q)$ ,  
 $f$  sinon

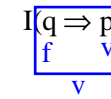
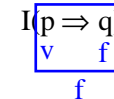
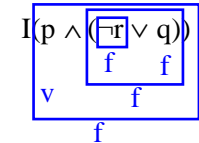
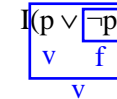
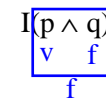
$I(\neg P) = v$  si  $I(P)$  est  $f$ ,  
 $f$  sinon

## Exemple (1)

$I(p) = v$  ;

$I(q) = f$  ;

$I(r) = v$ .

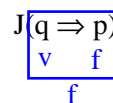
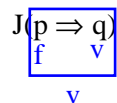
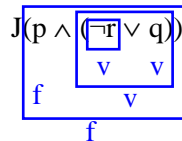
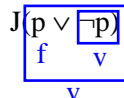
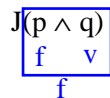


## Exemple (2)

$J(p) = f$  ;

$J(q) = v$  ;

$J(r) = f$ .



## Modèles

On considère qu'une interprétation  $I$  est un "monde possible".

Pour une formule  $\phi$  donnée

existe-t-il un monde dans lequel  $\phi$  est vraie ?

- Un modèle d'une formule  $\phi$  est une interprétation  $I$  telle que

$$I(\phi) = v$$

- Il peut exister zéro, un ou plusieurs modèles pour une formule donnée.
- Modèle d'une ensemble de formules  $F = \{\phi_1, \phi_2, \dots, \phi_k\} :=$  une interprétation qui rend vraie chaque formule  $\phi_1, \phi_2, \dots, \phi_k$ .

## Satisfaisabilité

S'il existe au moins un modèle de F  
on dit que F est « satisfaisable »  
(ou *consistant* ou *non contradictoire*).

sinon F est **inconsistante**.

## Exemple (1)

L'ensemble  $F = \{p \wedge q, q \vee r\}$  est satisfaisable

Il y a deux modèles :

I :  $I(p) = v, I(q) = v, I(r) = v$

J :  $I(p) = v, I(q) = v, I(r) = f$

## Exemple (2)

L'ensemble  $F = \{p \wedge q, q \vee r, \neg p\}$  est inconsistant

Dans tout modèle I on devrait avoir

$I(p \wedge q) = v$  donc  $I(p) = v$

et  $I(\neg p) = v$  donc  $I(p) = f$

[]

## Comment trouver un modèle ?

Etant donné une formule avec n variables

On ne connaît pas de meilleur algorithme que celui consistant à essayer successivement les  $2^n$  interprétations possibles.

=> complexité en temps exponentielle

Conséquence : les outils informatiques ne sont pas efficaces pour trouver des modèles.

p.ex. formule à 100 variables :

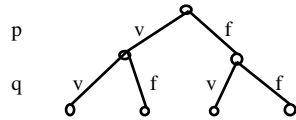
1267650600228229401496703205376 interprétations à tester

si on teste 1000000000 interprétations/sec.

40 196 936 841 331 années de calcul

## Arbre sémantique

- Autant de niveaux qu'il y a de variables dans la formule.
- De chaque noeud partent deux branches : 'vrai' et 'faux'
- Un chemin de la racine à une feuille représente une interprétation
- L'arbre représente toutes les interprétations possibles.



$p \wedge q$	v	f	f	f	(*)
$(p \wedge q) \vee p$	v	v	f	f	
$\neg(p \Rightarrow \neg q)$	v	f	f	f	(**)
$(p \wedge q) \Rightarrow q$	v	v	v	v	tautologie

## Tautologies

Les tautologies sont des formules vraies quelle que soit l'interprétation. Par exemple :

$$P \vee \neg P$$

$$P \Rightarrow P$$

$$((P \Rightarrow Q) \wedge P) \Rightarrow Q$$

$$(P \vee Q) \Leftrightarrow \neg(\neg P \wedge \neg Q)$$

etc.

Les tautologies sont des vérités universelles qui ne dépendent pas de l'état du monde.

## Equivalence

Deux formules E et F sont **équivalentes** si pour toute interprétation elles prennent les même valeurs de vérité.

p.ex.

$$p \wedge q \approx \neg(p \Rightarrow \neg q)$$

On peut voir que : E et F sont équivalentes si la formule

$$E \Leftrightarrow F$$

est une tautologie (est toujours vrai)

Les équivalences permettent des **manipulations syntaxiques** qui préservent la sémantique.

## Quelques équivalences:

Utiles pour simplifier les formules ou les restructurer.

A, B, C, ... sont des variables ou des formules quelconques

**double négation:**  $\neg(\neg A) \approx A$  ;

**associativité:**  $(A \wedge B) \wedge C \approx A \wedge (B \wedge C)$  ;  $(A \vee B) \vee C \approx A \vee (B \vee C)$  ;

**commutativité:**  $A \wedge B \approx B \wedge A$  ;  $A \vee B \approx B \vee A$  ;

**idempotence:**  $A \wedge A \approx A$  ;  $A \vee A \approx A$  ;

## Equivalences (suite)

**distributivité-ou:**  $A \vee (B \wedge C) \approx (A \vee B) \wedge (A \vee C)$  ;

**distributivité-et:**  $A \wedge (B \vee C) \approx (A \wedge B) \vee (A \wedge C)$  ;

**De Morgan:**  $\neg(A \wedge B) \approx \neg A \vee \neg B$  ;  $\neg(A \vee B) \approx \neg A \wedge \neg B$

**implication-ou:**  $A \Rightarrow B \approx \neg A \vee B$

**double implication:**  $A \Leftrightarrow B \approx (A \Rightarrow B) \wedge (B \Rightarrow A)$

donc  $\Rightarrow$  et  $\Leftrightarrow$  ne sont pas essentiels, toute formule peut être réécrite sans utiliser ces deux connecteurs.

**tiers exclu:**  $A \wedge \neg A \approx \text{faux}$      $A \vee \neg A \approx \text{vrai}$

## Formes normales

Toute formule peut se réécrire sous la forme

$$C_1 \wedge C_2 \wedge \dots \wedge C_n$$

où chaque  $C_i$  (appelé **clause**) est elle-même de la forme

$$p_1 \vee p_2 \vee \dots \vee p_m \vee \neg q_1 \vee \neg q_2 \vee \dots \vee \neg q_r$$

où les  $p_j$  et  $q_k$  sont des atomes.

Cette forme est appelée **forme normale conjonctive**.

Il y a un algorithme pour mettre en FNC

La FNC est nécessaire pour pouvoir appliquer certains algorithmes.

## Mise en forme normale conjonctive

$$((b \vee c) \Rightarrow a) \vee d$$

1. Appliquer les équivalence double-implique et implique-ou pour supprimer les  $\Rightarrow$  et  $\Leftrightarrow$ .

$$\approx (\neg(b \vee c) \vee a) \vee d$$

2. Appliquer De Morgan pour "descendre" les négation près des variables.

$$\approx ((\neg b \wedge \neg c) \vee a) \vee d$$

3. Appliquer la distributivité pour descendre les  $\vee$  et remonter les  $\wedge$

$$\approx ((\neg b \vee a) \wedge (\neg c \vee a)) \vee d$$

$$\approx ((\neg b \vee a) \vee d) \wedge ((\neg c \vee a) \vee d)$$

4. Appliquer l'associativité des  $\vee$  et  $\wedge$

$$\approx (\neg b \vee a \vee d) \wedge (\neg c \vee a \vee d)$$

## Autre forme normale

Il existe aussi une **forme normale disjonctive**  $\phi'''$  :

$$D_1 \vee D_2 \vee \dots \vee D_m$$

où chaque  $D_i$  est lui même de la forme

$$r_1 \wedge r_2 \wedge \dots \wedge r_u \wedge \neg s_1 \wedge \neg s_2 \wedge \dots \wedge \neg s_v$$

où les  $r_j$  et  $s_k$  sont des atomes.

## Autre notation (algèbre de Boole)

$$\neg A : \bar{A}$$

$$A \vee B : A + B$$

$$A \wedge B : A \cdot B$$

$$v : 1, f : 0$$

$$0+0 = 0, 0+1 = 1+0 = 1, 1+1 = 1$$

$$0 \cdot 0 = 0, 0 \cdot 1 = 1 \cdot 0 = 0, 1 \cdot 1 = 1$$

$$\text{clause} : x_1 + \dots + x_n + \bar{y}_1 + \dots + \bar{y}_n$$

$$\text{forme normale conjonctive} : C_1 \cdot C_2 \cdot \dots \cdot C_k$$

$$\text{monome} : x_1 \cdot \dots \cdot x_n \cdot \bar{y}_1 \cdot \dots \cdot \bar{y}_n$$

$$\text{forme normale disjonctive} : M_1 + M_2 \cdot \dots + M_r$$

## Conséquence logique

A est conséquence de B signifie

« A est nécessairement vraie lorsque B est vrai »

plus précisément

Si A est vraie dans une interprétation I alors B est vraie dans I

## Définition

f est une **conséquence logique** de  $\{f_1, f_2, \dots, f_n\}$

si tout modèle de  $f_1, f_2, \dots, f_n$  est forcément un modèle de f.

c-à-d pour tout I si  $I(f_1) = v, I(f_2) = v, \dots, I(f_n) = v$  alors  $I(f) = v$ ,

Notation  $\{f_1, f_2, \dots, f_n\} \models f$

Exemple.  $\{p \Rightarrow q, \neg q\} \models \neg p$ .

## Exemples de conséquences logiques

$$F = \{ p \Rightarrow q, q \Rightarrow r \}$$

vérifions que  $p \Rightarrow r$  est une conséquence logique de F.

Il y a quatre modèles de F:

modèles	p	q	r	$p \Rightarrow r$	$\neg r \vee p$
$I_1$	v	v	v	v	v
$I_2$	f	v	v	v	f
$I_3$	f	f	v	v	f
$I_4$	f	f	f	v	v

Donc  $F \models p \Rightarrow r$

mais  $F \not\models \neg r \vee p$

## Manipulations syntaxiques

Limites de l'approche purement sémantique :

1. Pour vérifier que  $E \models f$

il faut

- trouver tous les modèles de E
- pour chaque modèle  $I$  de E, vérifier que  $I(f) = v$

Si E utilise  $n$  propositions atomiques, le nombre de modèles potentiels est  $2^n$ .

2. Comment trouver les conséquences logiques de E ?

L'approche syntaxique (déductive) permet de calculer les conséquences logiques.

## T1. Une théorie formelle pour la LP (syst. de Hilbert)

### Axiomes

Il a une infinité d'axiomes, ce sont toutes les propositions de la forme

- $A \Rightarrow (B \Rightarrow A)$  ou
- $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$  ou
- $(\neg A \Rightarrow \neg B) \Rightarrow (B \Rightarrow A)$

où A, B et C sont des formules atomiques ou complexes.

### Règle d'inférence

Modus ponens

$$(A \Rightarrow B, A) \rightarrow B$$

ou Modus tollens

$$(A \Rightarrow B, \neg B) \rightarrow \neg A$$

## Exemple de preuve

axiome 2 avec  $B := (A \Rightarrow A)$  et  $C := A$  :

1.  $(A \Rightarrow ((A \Rightarrow A) \Rightarrow A)) \Rightarrow ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$

axiome 1 avec  $B := (A \Rightarrow A)$

2.  $A \Rightarrow ((A \Rightarrow A) \Rightarrow A)$

modus ponens 1, 2

3.  $((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$

axiome 1 avec  $B := A$

4.  $A \Rightarrow (A \Rightarrow A)$

modus ponens 3, 4

$(A \Rightarrow A)$

## Consistance de modus-ponens

Tous les modèles de  $F = \{A \Rightarrow B, A\}$

Dans tout modèle M de F on doit avoir

$$M(A) = v, M(B) = v$$

Comme  $M(B) = v$

$$\{A \Rightarrow B, A\} \models B$$

Donc

$$\{A \Rightarrow B, A\} \vdash B$$

est consistant.

On ne déduit que ce qui est conséquence logique



## Règles dérivées - règle de déduction

But: faciliter les preuves avec de nouvelles règles

Ces règles doivent être consistantes : ne pas produire plus que MP + axiomes

### Règle de déduction

U un ensemble de formules, A une formule

si

à partir de U et A on peut prouver B (avec MP et les axiomes)

alors

à partir de U on peut prouver  $A \Rightarrow B$

## T2. Une autre théorie [Hofstadter GEB]

Il n'y a pas d'axiomes !

Règles:

ASSOCIATION

si  $\vdash f$  et  $\vdash g$  alors  $\vdash f \wedge g$

DISSOCIATION

si  $\vdash f \wedge g$  alors  $\vdash f$  et  $\vdash g$

MODUS PONENS

si  $\vdash f \Rightarrow g$  et  $\vdash f$  alors  $\vdash g$

$\vdash f$  signifie « f est un théorème »

## Exemple

On part de trois théorèmes:

1.  $p \wedge q$  thm

2.  $q \Rightarrow r$  thm

3.  $q \Rightarrow s$  thm

On applique les règles:

4. q dissociation sur 1.

5. r modus ponens 2 et 4

6. s modus ponens 3 et 4

7.  $(r \wedge s)$  association 5 et 6

## Règles d'échange

DOUBLE NEGATION

on peut ajouter ou supprimer  $\neg\neg$  dans tout théorème

CONTRAPOSITION

$f \Rightarrow g$  et  $\neg g \Rightarrow \neg f$  sont interchangeables

DE MORGAN

$\neg f \wedge \neg g$  et  $\neg(f \vee g)$  sont interchangeables

OU-IMPLICATION

$f \Rightarrow g$  et  $\neg f \vee g$  sont interchangeables

## Exemple

$(f \Rightarrow g) \wedge \neg f$	
$\vdash (\neg f \vee g) \wedge \neg f$	échange ou-implique
$\vdash (\neg\neg f \vee \neg g) \wedge \neg f$	ajouts double négations
$\vdash \neg(\neg\neg f \wedge \neg g) \wedge \neg f$	échange par De-Morgan
$\vdash \neg(f \wedge \neg g) \wedge \neg f$	suppression double négation
$\vdash \neg((f \wedge \neg g) \vee f)$	encore De Morgan

## Règle « imagination » = règle de déduction

Création de théorèmes sans axiomes de départ

IMAGINATION:

si  $f \vdash g$  alors  $\vdash f \Rightarrow g$

Si on peut prouver  $g$  à partir de  $f$  (et des axiomes) alors que on a prouvé  $f \Rightarrow g$ .

Exemple

[	
$p$	hypothèse
$\neg\neg p$	double négation
]	
$\vdash p \Rightarrow \neg\neg p$	

## Imagination et importation

IMPORTATION:

On peut «importer» n'importe quel théorème du niveau supérieur dans une dérivation imaginaire.

[	imagination
$p$	hypothèse
[	imagination, 2
$q$	hypothèse
$p$	importation
$p \wedge q$	association
]	
$q \Rightarrow (p \wedge q)$	résultat imagination 2
]	
$p \Rightarrow (q \Rightarrow (p \wedge q))$	résultat imagination

## Exemple - la hache de Ganto

$(p \Rightarrow q) \wedge (\neg p \Rightarrow q)$	thm donné
$p \Rightarrow q$	dissociation
$\neg q \Rightarrow \neg p$	contraposition
$\neg p \Rightarrow q$	dissociation
$\neg q \Rightarrow \neg\neg p$	contraposition
[ $\neg q$	hypothèse
$\neg q \Rightarrow \neg\neg p$	import
$\neg\neg p$	modus ponens
$\neg q \Rightarrow \neg p$	import
$\neg p$	modus ponens
$\neg\neg p \wedge \neg p$	association
$\neg(\neg p \vee p)$	De Morgan
] $\neg q \Rightarrow \neg(\neg p \vee p)$	sortie imagination
[ $p$ ] $p \Rightarrow p, \neg p \vee p$	hypothèse, sortie imagination, échange ou-implique
$(\neg p \vee p) \Rightarrow q$	contraposition
$q$	modus ponens

## Complétude et Consistance des règles de déduction

C'est le lien entre syntaxe et sémantique

**Consistance** = tous les théorèmes sont des conséquences logiques des formules de départ = tout ce qu'on démontre est vrai =

$$\text{si } F \vdash g \text{ alors } F \models g$$

**Complétude** = on peut tout déduire = toute conséquence logique est un théorème = toutes les vérités sont démontrables =

$$\text{si } F \models g \text{ alors } F \vdash g$$

P.ex. la théorie T1 est consistante et complète.

## Principe de résolution (Robinson)

Encore une règle d'inférence

Inventée en 1965

But: mécanisation de la logique, trouver plus facilement des preuves

Travaille sur des clauses.

Il faut donc commencer par tout mettre en forme normale conjonctive.

$$\text{Clause}_1 \wedge \text{Clause}_2 \wedge \dots \wedge \text{Clause}_n$$

$$\text{Clause} = \text{Littéral}_1 \vee \text{Littéral}_2 \vee \dots \vee \text{Littéral}_m$$

$$\text{Littéral} = \text{Variable ou bien } \neg \text{Variable}$$

## Règle

Si on a deux clauses

$$C_1 = (p \vee L_1 \vee L_2 \vee \dots \vee L_m),$$

$$C_2 = (\neg p \vee L'_1 \vee L'_2 \vee \dots \vee L'_n)$$

le résolvant de  $C_1$  et  $C_2$  est la clause

$$L_1 \vee L_2 \vee \dots \vee L_m \vee L'_1 \vee L'_2 \vee \dots \vee L'_n$$

Si un littéral et son complément apparaissent dans deux clauses on peut produire une nouvelle clause à partir de celles-ci.

## Exemples

$$C1 = (p \vee q \vee \neg r \vee s)$$

$$C2 = (q \vee \neg p \vee t)$$

Résolution sur  $p$  et  $\neg p$ .

$$\text{Résolvant: } (q \vee \neg r \vee s \vee q \vee t),$$

On retrouve le modus ponens car

de  $p \Rightarrow q$ , équivalent à  $\neg p \vee q$ , et  $p$ , on déduit  $q$ .

## Exemple 2

$\{ (p \vee t) \Rightarrow q, r \Rightarrow (t \vee s), (q \wedge t) \Rightarrow u, p, \neg s, r \} \models u$

Mettre sous forme de clauses :

$(p \vee t) \Rightarrow q$

$\approx \neg(p \vee t) \vee q$

$\approx (\neg p \wedge \neg t) \vee q$

$\approx (\neg p \vee q) \wedge (\neg t \vee q)$  ..... 2 clauses :  **$(\neg p \vee q)$ ,  $(\neg t \vee q)$**

$r \Rightarrow (t \vee s)$

$\approx \neg r \vee (t \vee s)$

$\approx \neg r \vee t \vee s$

## (suite)

$(q \wedge t) \Rightarrow u$

$\approx \neg(q \wedge t) \vee u$

$\approx (\neg q \vee \neg t) \vee u$

$\approx \neg q \vee \neg t \vee u$

## Résolution

$\neg p \vee q$  (1)

$\neg t \vee q$  (2)

$\neg r \vee t \vee s$  (3)

$\neg q \vee \neg t \vee u$  (4)

$p$  (5)

$\neg s$  (6)

$r$  (7)

$\vdash q$  (résolution 1 & 5 sur p) (8)

$\vdash t \vee s$  (résolution 3 & 7 sur r) (9)

$\vdash t$  (résolution 6 & 9 sur s) (10)

$\vdash \neg t \vee u$  (résolution 4 & 8 sur q) (11)

$\vdash u$  (résolution 10 & 11 sur t) (12)

## Attention !

si on a

$C1 = (p \vee \neg q \vee \neg r \vee s)$

$C2 = (q \vee \neg p \vee t)$

on peut déduire

$\neg q \vee \neg r \vee s \vee q \vee t$  (résolution sur p)

ou

$p \vee \neg r \vee s \vee \neg p \vee t$  (résolution sur q)

mais pas

$\neg r \vee s \vee t$  !

## Utilisation

Théorème:

Le résolvant C est satisfaisable

si et seulement si les clauses parentes C1 et C2 le sont  
(simultanément)

Procédure de résolution

$S_i$  un ensemble de clauses

C1 et C2 deux clauses de  $S_i$  dont le résolvant est C

on définit  $S_{i+1} := S_i \cup \{C\}$

Si au cours du processus on trouve un  $C = []$ ,  $S_0$  était inconsistant

Si  $S_{i+1} = S_i$  pour tout choix de C,  $S_0$  était satisfaisable

## Preuves par réfutation

Une dérivation de  $[]$  à partir de S est une **réfutation**

Pour prouver que A est une conséquence logique de S

On cherche à réfuter  $S \cup \{\neg A\}$ .

Exemple

$S = \{p, \neg p \vee q, \neg p \vee \neg q \vee r\}$ ,  $A = r$

$S_0 = \{p, \neg p \vee q, \neg r, \neg p \vee \neg q \vee r\}$

$S_1 = \{p, \neg p \vee q, \neg r, \neg p \vee \neg q \vee r, \neg p \vee \neg q\}$

$S_2 = \{p, \neg p \vee q, \neg r, \neg p \vee \neg q \vee r, \neg p \vee \neg q, \neg p\}$

$S_3 = \{p, \neg p \vee q, \neg r, \neg p \vee \neg q \vee r, \neg p \vee \neg q, \neg p, []\}$

donc  $S \models r$

## Références

Cl. Benzaken. *Systèmes formels: introduction à la logique et à la théorie des langages*, Masson, Paris, 1991.

Bundy, A. *The Computer Modelling of Mathematical Reasoning*, Academic Press, London, 1983.

Cori, R., Lascar, D. *Logique mathématique*, Masson, 1993.