Logique et Programmation Logique

Quentin Bramas

Largement inspiré par le cours de J.Narboux, C.Dubois et d'autres

RT-INOC 1A

September 12, 2018

Table des matières I

Calcul des propositions

Le calcul des propositions (aussi appelé logique propositionnelle, ou bien CP0) est une des logiques les plus simples, elle ne comporte que des *variables* et des *connecteurs* logiques.

Les formules

Les formules représentent des propriétés

- exemple en algèbre, une formule peut représenter le fait qu'un espace vectoriel est dimension 3.

Les formules

Les formules représentent des propriétés

- exemple en algèbre, une formule peut représenter le fait qu'un espace vectoriel est dimension 3.
- exemple en analyse, une formule peut exprimer la continuité d'une fonction.

Les formules

Les formules représentent des propriétés

- exemple en algèbre, une formule peut représenter le fait qu'un espace vectoriel est dimension 3.
- exemple en analyse, une formule peut exprimer la continuité d'une fonction.
- exemple en théorie des ensembles, une formule peut exprimer l'inclusion de deux ensembles.

Les formules

Les formules représentent des propriétés

- exemple en algèbre, une formule peut représenter le fait qu'un espace vectoriel est dimension 3.
- exemple en analyse, une formule peut exprimer la continuité d'une fonction.
- exemple en théorie des ensembles, une formule peut exprimer l'inclusion de deux ensembles.

Dans ce cours, ce qu'elles représentent n'a pas d'importance, on veut définir ce qu'est une formule, de manière générale, *au sens syntaxique*.

Syntaxe vs sémantique

Syntaxe

La *syntaxe* d'un langage formel est l'ensemble des règles définissant si un mot est correct dans le langage.

Syntaxe vs sémantique

Syntaxe

La *syntaxe* d'un langage formel est l'ensemble des règles définissant si un mot est correct dans le langage.

Sémantique

La *sémantique* d'une langage est une relation entre les formules syntaxiquement correctes et leurs signification.

Syntaxe vs sémantique

Syntaxe

La *syntaxe* d'un langage formel est l'ensemble des règles définissant si un mot est correct dans le langage.

Sémantique

La *sémantique* d'une langage est une relation entre les formules syntaxiquement correctes et leurs signification.

Exemple en C

0x10 et 16 désignent l'entier mathématique: 16

Méta-théorie vs théorie

Il ne faut pas confondre la logique dans laquelle on va raisonner et la logique (appelée théorie objet ou langage objet) que l'on étudie.

- Théorie dans laquelle on raisonne : méta-théorie
- La logique : théorie objet

La méta-théorie ne sera pas bien définie ce sont *les mathématiques usuelles*, tandis que la théorie objet sera définie avec précision.

On va définir la logique et prouver des méta-théorème la concernant.

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \bot\}$ *inductivement*

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \bot\}$ *inductivement* par les éléments de base $V \cup \{\bot\}$ et les règles suivantes :

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \bot\}$ *inductivement* par les éléments de base $V \cup \{\bot\}$ et les règles suivantes :

- Si B une formule alors $\neg B$ aussi.

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \bot\}$ *inductivement* par les éléments de base $V \cup \{\bot\}$ et les règles suivantes :

- Si B une formule alors $\neg B$ aussi.
- Si B et C sont des formules, alors $B \wedge C$ aussi.

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \perp\}$ *inductivement* par les éléments de base $V \cup \{\perp\}$ et les règles suivantes :

- Si B une formule alors $\neg B$ aussi.
- Si B et C sont des formules, alors $B \wedge C$ aussi.
- Si B et C sont des formules, alors $B \vee C$ aussi.

Syntaxe

Soit V un ensemble dénombrable de symboles, dits variables propositionnelles.

$$V = \{a, b, c, d, e, \ldots\}$$

L'ensemble des formules de la logique propositionnelle est défini sur l'alphabet $V \cup \{\Rightarrow, \wedge, \vee, \neg, \perp\}$ *inductivement* par les éléments de base $V \cup \{\perp\}$ et les règles suivantes :

- Si B une formule alors $\neg B$ aussi.
- Si B et C sont des formules, alors $B \wedge C$ aussi.
- Si B et C sont des formules, alors $B \vee C$ aussi.
- Si B et C sont des formules, alors $B \Rightarrow C$ aussi.

Exemples

Exemples de formules:

- a
- $a \Rightarrow b$
- $\neg(a \Rightarrow b)$
- $a \wedge b \vee c$

Exemples qui ne sont pas des formules:

- $a \Rightarrow\Rightarrow a$
- $a\neg b$

Parenthèsage

- Ordre des priorités: $\neg \ > \ \wedge \ > \ \vee$.
- Associtivité:
  - $\vee$ et $\wedge$ sont associatifs à gauche.
  - $\Rightarrow$ est associatif à droite.

Parenthèsage

- Ordre des priorités: $\neg \ > \ \wedge \ > \ \vee$.
- Associtivité:
  - $\vee$ et $\wedge$ sont associatifs à gauche.
  - $\Rightarrow$ est associatif à droite.

Exemples

- $A \Rightarrow \ \ B \Rightarrow \ \ C \Rightarrow D$
- $A \vee B \ \ \vee C$

Parenthèsage

- Ordre des priorités: $\neg\ >\ \wedge\ >\ \vee$.
- Associtivité:
  - $\vee$ et $\wedge$ sont associatifs à gauche.
  - $\Rightarrow$ est associatif à droite.

Exemples

- $A \Rightarrow (B \Rightarrow (C \Rightarrow D))$
- $A \vee B\ \ \vee C$

Parenthèsage

- Ordre des priorités: $\neg \ > \ \wedge \ > \ \vee$.
- Associtivité:
  - $\vee$ et $\wedge$ sont associatifs à gauche.
  - $\Rightarrow$ est associatif à droite.

Exemples

- $A \Rightarrow (B \Rightarrow (C \Rightarrow D))$
- $(A \vee B) \vee C$

Equivalence, *etc.*

On considère que l'équivalence $A \Leftrightarrow B$ est une *notation* pour $A \Rightarrow B \wedge B \Rightarrow A$.

Equivalence, *etc.*

On considère que l'équivalence $A \Leftrightarrow B$ est une *notation* pour $A \Rightarrow B \wedge B \Rightarrow A$.

Remarque

On aurait pu choisir un autre ensemble de connecteurs.

- Avoir plus de connecteurs rendent les meta-theorèmes plus longs: plus de cas à traiter.
- Avoir moins de connecteurs obligent à définir des connecteurs à partir d'autres connecteurs.

Equivalence, *etc.*

On considère que l'équivalence $A \Leftrightarrow B$ est une *notation* pour $A \Rightarrow B \wedge B \Rightarrow A$.

Remarque

On aurait pu choisir un autre ensemble de connecteurs.

- Avoir plus de connecteurs rendent les meta-theorèmes plus longs: plus de cas à traiter.
- Avoir moins de connecteurs obligent à définir des connecteurs à partir d'autres connecteurs.

Exercice

Quels sont les ensembles de connecteurs minimal ?

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0,1\}$.

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0, 1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0, 1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\bot) = 0$
- $I(\neg A) = 1$ ssi

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0, 1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\perp) = 0$
- $I(\neg A) = 1$ ssi $I(A) = 0$
- $I(A \vee B) = 1$ ssi

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0, 1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\bot) = 0$
- $I(\neg A) = 1$ ssi $I(A) = 0$
- $I(A \vee B) = 1$ ssi $I(A) = 1$ ou $I(B) = 1$
- $I(A \wedge B) = 1$ ssi

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0,1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\bot) = 0$
- $I(\neg A) = 1$ ssi $I(A) = 0$
- $I(A \vee B) = 1$ ssi $I(A) = 1$ ou $I(B) = 1$
- $I(A \wedge B) = 1$ ssi $I(A) = 1$ et $I(B) = 1$
- $I(A \Rightarrow B) = 1$ ssi

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0, 1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\bot) = 0$
- $I(\neg A) = 1$ ssi $I(A) = 0$
- $I(A \vee B) = 1$ ssi $I(A) = 1$ ou $I(B) = 1$
- $I(A \wedge B) = 1$ ssi $I(A) = 1$ et $I(B) = 1$
- $I(A \Rightarrow B) = 1$ ssi $I(A) = 0$ ou $I(B) = 1$

On dit que I satisfait A ssi $I(A) = 1$.
On dit que I falsifie A ssi $I(A) = 0$.

Sémantique

On appelle interprétation (ou valuation) une application I de V dans $\{0,1\}$. L'application I est étendue aux formules de la logique propositionnelle par récurrence sur la structure des formules:

- $I(\perp) = 0$
- $I(\neg A) = 1$ ssi $I(A) = 0$
- $I(A \vee B) = 1$ ssi $I(A) = 1$ ou $I(B) = 1$
- $I(A \wedge B) = 1$ ssi $I(A) = 1$ et $I(B) = 1$
- $I(A \Rightarrow B) = 1$ ssi $I(A) = 0$ ou $I(B) = 1$

On dit que I satisfait A ssi $I(A) = 1$.
On dit que I falsifie A ssi $I(A) = 0$.

Exercice

$I(A \Leftrightarrow B) = 1$ ssi ...

Tables de vérités

Pour une interprétation I donnée, on peut calculer rapidement l'interprétation d'une formule F avec une ligne représentant les étapes successives du calcul.

Tables de vérités

Pour une interprétation I donnée, on peut calculer rapidement l'interprétation d'une formule F avec une ligne représentant les étapes successives du calcul.

Exemple: Calcluler la valeur de $F = (A \vee B) \Rightarrow (\neg A \wedge B)$ lorsque $I(A) = 1$ et $I(B) = 0$

A	B	$A \vee B$	$\neg A$	$\neg A \wedge B$	F
1	0	1	0	0	0

Tables de vérités

Définition

On appelle table de vérité la représentation sous forme d'un tableau de la fonction qui à chaque interprétation I associe la valeur $I(F)$ pour des formules F données.

A	B	$A \wedge B$	$A \vee B$	$A \Rightarrow B$	$\neg A$
0	0	0	0	1	1
0	1	0	1	1	1
1	0	0	1	0	0
1	1	1	1	1	0

Tables de vérités (Exemple précédent)

A	B	$A \vee B$	$\neg A$	$\neg A \wedge B$	F
0	0	0	1	0	1
1	0	1	0	0	0
0	1	1	1	1	1
1	1	1	0	0	0

Les 16 connecteurs logiques d'arité 2

A	B	$\bot$	$\wedge$	?	?	?	?	xor	$\vee$	?	$\Leftrightarrow$	?	?	?	$\Rightarrow$	?	$\top$
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

L'implication

Remarque:

L'implication ne représente pas un lien de cause à effet.

Exemple:

L'augmentation de la consommation de glaces entraine une augmentation du nombre de noyades.

Formules satisfiables, insatisfiables

satisfiable: Une formule A est satisfiable s'il existe **au moins une** interprétation I qui satisfait A.

satisfiable: Un ensemble de formules Γ est satisfiable s'il existe **au moins une** interprétation I qui satisfait toutes les formules de Γ.

insatisfiable: Une formule A est insatisfiable (contradictoire) si elle n'est pas satisfiable.

insatisfiable: Un ensemble de formules Γ est insatisfiable (contradictoire) si il n'est pas satisfiable.

Formules valides, conséquence

validité Une formule A est valide (tautologie) si **toute** interprétation satisfait A.
On note: $\models A$

conséquence Une formule A est conséquence logique d'un ensemble de formules Γ si toute interprétation qui satisfait Γ satisfait A.
On note: $\Gamma \models A$

Formules valides, conséquence

validité Une formule A est valide (tautologie) si **toute** interprétation satisfait A.
On note: $\models A$

conséquence Une formule A est conséquence logique d'un ensemble de formules Γ si toute interprétation qui satisfait Γ satisfait A.
On note: $\Gamma \models A$

Exemples

$$\models P \vee \neg P$$

$$\{P, Q\} \models Q \wedge P$$

Exercices

Montrer que:

$$\models (A \Rightarrow B) \vee \neg(A \Rightarrow B)$$

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

Equivalence sémantique

Définition de l'équivalence sémantique

Deux formules A et B sont équivalentes (noté $A \equiv B$) ssi $\{A\} \models B$ et $\{B\} \models A$.

Meta-théorème de la déduction

Meta-théorème de la déduction

$\Gamma \models F \Rightarrow G$ ssi $\Gamma, F \models G$

Meta-théorème de la déduction

Meta-théorème de la déduction

$\Gamma \models F \Rightarrow G$ ssi $\Gamma, F \models G$

Preuve

- Supposons que $\Gamma \models F \Rightarrow G$ montrons que $\Gamma, F \models G$. Soit I une interprétation telle que I satisfait Γ, F. I satisfait Γ donc I satisfait $F \Rightarrow G$. Or $I(F) = 1$ donc $I(G) = 1$.
- Supposons que $\Gamma, F \models G$ montrons que $\Gamma \models F \Rightarrow G$. Soit I une interprétation qui satisfait Γ. Si I satisfait F alors I satisfait G et donc I satisfait $F \Rightarrow G$. Sinon I ne satisfait pas F. Donc I satisfait $F \Rightarrow G$.

Remarque

$A \equiv B$ ssi $\models A \Leftrightarrow B$

Quelques équivalences utiles

Lois de De Morgan

- $\models \neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$
- $\models \neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$

Lois de ditributivité

- $\models (A \vee B) \wedge C \Leftrightarrow (A \wedge C) \vee (B \wedge C)$
- $\models (A \wedge B) \vee C \Leftrightarrow (A \vee C) \wedge (B \vee C)$

Associativité/Commutativité

- $\models ((A \vee B) \vee C) \Leftrightarrow (A \vee (B \vee C))$
- $\models ((A \wedge B) \wedge C) \Leftrightarrow (A \wedge (B \wedge C))$
- $\models (A \vee B) \Leftrightarrow (B \vee A)$
- $\models (A \wedge B) \Leftrightarrow (B \wedge A)$

Quelques équivalences utiles

Règle de l'implication

- $\models A \Rightarrow B \Leftrightarrow \neg A \vee B$

Règles de simplification

- $\models \neg\neg A \Leftrightarrow A$
- $\models A \wedge \neg A \Leftrightarrow \bot$
- $\models A \wedge \bot \Leftrightarrow \bot$
- $\models A \wedge \top \Leftrightarrow A$
- $\models A \vee \neg A \Leftrightarrow \top$
- $\models A \vee \bot \Leftrightarrow A$
- $\models A \vee \top \Leftrightarrow \top$
- $\models A \wedge A \Leftrightarrow A$
- $\models A \vee A \Leftrightarrow A$

Quelques équivalences utiles

Implications imbriquées

- $\models (A \Rightarrow (B \Rightarrow G)) \Leftrightarrow (A \wedge B \Rightarrow G)$

Quelques équivalences utiles

Implications imbriquées

- $\models (A \Rightarrow (B \Rightarrow G)) \Leftrightarrow (A \wedge B \Rightarrow G)$
- $\models (H_1 \Rightarrow (H_2 \Rightarrow (H_3 \Rightarrow G))) \Leftrightarrow (H_1 \wedge H_2 \wedge H_3 \Rightarrow G)$

Quelques équivalences utiles

Implications imbriquées

- $\models (A \Rightarrow (B \Rightarrow G)) \Leftrightarrow (A \wedge B \Rightarrow G)$
- $\models (H_1 \Rightarrow (H_2 \Rightarrow (H_3 \Rightarrow G))) \Leftrightarrow (H_1 \wedge H_2 \wedge H_3 \Rightarrow G)$

Simplification

- $\models (A \vee P) \wedge (A \vee \neg P) \iff A$

Substitution

Si F et G sont des formules et p une variable propositionnelle. $F[p := G]$ est la formule obtenue à partir de F en substituant G à p.

Substitution

Si F et G sont des formules et p une variable propositionnelle. $F[p := G]$ est la formule obtenue à partir de F en substituant G à p.

Formellement, on définit $F[p := G]$ par induction :

- $\bot[p := G] = \bot$
- $q[p := G] = G$ si $q = p$; q sinon
- $(\neg F)[p := G] = \neg(F[p := G])$
- $(F_1 \wedge F_2)[p := G] = (F_1[p := G]) \wedge (F_2[p := G])$
- $(F_1 \vee F_2)[p := G] = (F_1[p := G]) \vee (F_2[p := G])$
- $(F_1 \Rightarrow F_2)[p := G] = (F_1[p := G]) \Rightarrow (F_2[p := G])$

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\perp[p := G]) = I'(\perp) = 0 = I(\perp)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\perp[p := G]) = I'(\perp) = 0 = I(\perp)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.
3. hypothèses d'induction :
   $I'(F_1[p := G]) = I(F_1)$ et $I'(F_2[p := G]) = I_0(F_2)$.

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\perp[p := G]) = I'(\perp) = 0 = I(\perp)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.
3. hypothèses d'induction :
   $I'(F_1[p := G]) = I(F_1)$ et $I'(F_2[p := G]) = I_0(F_2)$.
   alors:
   $I'((F_1 \wedge F_2)[p := G]) = I'(F_1[p := G] \wedge F_2[p := G]) = 1$
   ssi $I'(F_1[p := G]) = 1$ et $I'(F_2[p := G]) = 1$

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\bot[p := G]) = I'(\bot) = 0 = I(\bot)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.
3. hypothèses d'induction :
   $I'(F_1[p := G]) = I(F_1)$ et $I'(F_2[p := G]) = I_0(F_2)$.
   alors:
   $I'((F_1 \wedge F_2)[p := G]) = I'(F_1[p := G] \wedge F_2[p := G]) = 1$
   ssi $I'(F_1[p := G]) = 1$ et $I'(F_2[p := G]) = 1$
   ssi $I(F_1) = 1$ et $I(F_2]) = 1$

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\bot[p := G]) = I'(\bot) = 0 = I(\bot)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.
3. hypothèses d'induction :
   $I'(F_1[p := G]) = I(F_1)$ et $I'(F_2[p := G]) = I_0(F_2)$.
   alors:
   $I'((F_1 \wedge F_2)[p := G]) = I'(F_1[p := G] \wedge F_2[p := G]) = 1$
   ssi $I'(F_1[p := G]) = 1$ et $I'(F_2[p := G]) = 1$
   ssi $I(F_1) = 1$ et $I(F_2]) = 1$
   ssi $I(F_1 \wedge F_2) = 1$.

Lemme

Soit I et I' deux interprétations et p une variable. Si $I'(q) = I(q)$ pour toute variable $q \neq p$ et si $I'(G) = I(p)$, alors $I'(F[p := G]) = I(F)$.

Par induction sur F. La lemme est vrai sur les cas de base:

1. $I'(\perp[p := G]) = I'(\perp) = 0 = I(\perp)$
2. soit q une variable propositionnelle.
   si $p = q$, $I'(p[p := G]) = I'(G) = I(p)$.
   si $p \neq q$, $I'(q[p := G]) = I'(q) = I(q)$.
3. hypothèses d'induction :
   $I'(F_1[p := G]) = I(F_1)$ et $I'(F_2[p := G]) = I_0(F_2)$.
   alors:
   $I'((F_1 \wedge F_2)[p := G]) = I'(F_1[p := G] \wedge F_2[p := G]) = 1$
   ssi $I'(F_1[p := G]) = 1$ et $I'(F_2[p := G]) = 1$
   ssi $I(F_1) = 1$ et $I(F_2]) = 1$
   ssi $I(F_1 \wedge F_2) = 1$.
4. de même pour F de la forme $F_1 \vee F_2$.
5. de même pour F de la forme $F_1 \Rightarrow F_2$.
6. de même pour F de la forme $\neg F_1$.

Principe d'induction sur les formules

1. Pour toute variable x, $P(x)$.
2. Pour toute formule F, si $P(F)$ alors $P(\neg F)$.
3. Pour toutes formules F_1 F_2, si $P(F_1)$ et $P(F_2)$ alors $P(F_1 \wedge F_2)$.
4. Pour toutes formules F_1 F_2, si $P(F_1)$ et $P(F_2)$ alors $P(F_1 \vee F_2)$.
5. Pour toutes formules F_1 F_2, si $P(F_1)$ et $P(F_2)$ alors $P(F_1 \Rightarrow F_2)$.

alors

Pour toute formule F, on a $P(F)$.

Lemmes de substitution

Corollaires

1. Si $\models F$ alors $\models F[p := G]$
2. Si $F \equiv F'$ alors $F[p := G] \equiv F'[p := G]$

Lemme

- Si $G \equiv G'$ alors $F[p := G] \equiv F[p := G']$

Forme clausale

Littéral

Un littéral est une variable x ou sa négation $\neg x$ (souvent noté $\overline{x}$).

Vocabulaire

On dit qu'un littéral est positif s'il ne comporte pas de négation $\neg$, négatif sinon.

Clause

Une clause est une disjonction de littéraux.

Formes canoniques

Forme normale conjonctive

On dit que F est sous forme normale conjonctive si F est une conjonction de disjonctions de littéraux. (*une conjonction de clauses)*

Forme normale disjonctive

On dit que F est sous forme normale disjonctive si F est une disjonction de conjonction de littéraux.

Exemples de formules en forme normale conjonctive

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

$$A$$

$$A \wedge B$$

Exemples de formules qui ne sont pas en forme normale

$$\neg(A \wedge B)$$

$$A \Rightarrow B$$

Exercice

Une formule peut-elle être sous forme normale disjonctive et normale conjonctive en même temps?

Un méta-théorème

Mise sous forme normale conjonctive

Pour toute formule F il existe une formule F' en FNC (forme normale conjonctive) telle que $F \equiv F'$.

Preuve = méthode

1. On élimine les $\Leftrightarrow$ en transformant $A \Leftrightarrow B$ par $A \Rightarrow B \wedge B \Rightarrow A$.
2. On élimine les $\Rightarrow$ en transformant $A \Rightarrow B$ par $\neg A \vee B$.
3. On propage les négations vers les feuilles avec les règles:
   - $\neg(A \vee B)$ devient $\neg A \wedge \neg B$
   - $\neg(A \wedge B)$ devient $\neg A \vee \neg B$
4. On simplifie les double-négations en transformant $\neg\neg A$ en A.
5. On distribue le $\wedge$ sur le $\vee$ avec les règles:
   - $A \vee (B \wedge C)$ devient $(A \vee B) \wedge (A \vee C)$
   - $(B \wedge C) \vee A$ devient $(A \vee B) \wedge (A \vee C)$
6. On simplifie.

Exemple

$\neg(A \Leftrightarrow B)$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

$$\equiv$$

$$(\neg(\neg A \vee B)) \vee (\neg(\neg B \vee A))$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

$$\equiv$$

$$(\neg(\neg A \vee B)) \vee (\neg(\neg B \vee A))$$

$$\equiv$$

$$(A \wedge \neg B) \vee (B \wedge \neg A)$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

$$\equiv$$

$$(\neg(\neg A \vee B)) \vee (\neg(\neg B \vee A))$$

$$\equiv$$

$$(A \wedge \neg B) \vee (B \wedge \neg A)$$

$$\equiv$$

$$(A \vee (B \wedge \neg A)) \wedge (\neg B \vee (B \wedge \neg A))$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

$$\equiv$$

$$(\neg(\neg A \vee B)) \vee (\neg(\neg B \vee A))$$

$$\equiv$$

$$(A \wedge \neg B) \vee (B \wedge \neg A)$$

$$\equiv$$

$$(A \vee (B \wedge \neg A)) \wedge (\neg B \vee (B \wedge \neg A))$$

$$\equiv$$

$$((A \vee B) \wedge (A \vee \neg A)) \wedge ((\neg B \vee B) \wedge (\neg B \vee \neg A))$$

Exemple

$$\neg(A \Leftrightarrow B)$$

$$\equiv$$

$$\neg(A \Rightarrow B \wedge B \Rightarrow A)$$

$$\equiv$$

$$\neg((\neg A \vee B) \wedge (\neg B \vee A))$$

$$\equiv$$

$$(\neg(\neg A \vee B)) \vee (\neg(\neg B \vee A))$$

$$\equiv$$

$$(A \wedge \neg B) \vee (B \wedge \neg A)$$

$$\equiv$$

$$(A \vee (B \wedge \neg A)) \wedge (\neg B \vee (B \wedge \neg A))$$

$$\equiv$$

$$((A \vee B) \wedge (A \vee \neg A)) \wedge ((\neg B \vee B) \wedge (\neg B \vee \neg A))$$

$$\equiv$$

$$(A \vee B) \wedge (\neg B \vee \neg A)$$

Attention !

La formule peut grandir de manière exponentielle.

Méta-théorème

Décidabilité

La logique propositionnelle est décidable: il existe un algorithme qui termine dans tous les cas et qui permet de déterminer si une formule est satisfiable/valide.