Logique des propositions

1. Formule logique

1.1. Syntaxe

On s'intéresse dans cette partie à la **syntaxe** de la logique des propositions.

Les symboles utilisés dans le calcul des propositions sont constitués

- de **variables propositionnelles**, en nombre dénombrable, représentées par les lettres a, b, c, ... éventuellement avec des indices a_1, a_2,...
- d'un connecteur unaire noté $\neg$ appelé connecteur de **négation**,
- de quatre connecteurs binaires notés $\wedge$, $\vee$, $\rightarrow$ et $\leftrightarrow$, appelés connecteurs de **conjonction** (et), **disjonction** (ou), **implication** et **équivalence**,
- des parenthèses ouvrantes et fermantes.

Formule logique

Une **formule**, ou encore proposition, est définie par les règles suivantes :

- toute variable est une formule,
- si φ est une formule, alors $(\neg\varphi)$ est une formule,
- si φ et ψ sont des formules, alors $(\varphi\wedge\psi)$, $(\varphi\vee\psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ sont des formules.

Remarques :

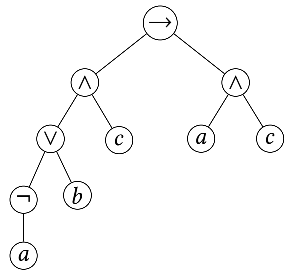
- Les choix des symboles utilisés diffèrent d'un auteur à l'autre : pour certains, on ne considère par l'équivalence $\leftrightarrow$.
- Les parenthèses permettent d'éviter toute ambiguïté, mais on s'autorise à en supprimer certaines en utilisant les règles de priorité suivantes : la négation est prioritaire devant tous les autres, la conjonction a priorité sur la disjonction, la disjonction et la conjonction sont prioritaires devant l'implication et l'équivalence.

  Par exemple, $(((\neg a) \vee b) \wedge c) \rightarrow (a \wedge c)$ peut se simplifier en $(\neg a \vee b) \wedge c \rightarrow a \wedge c$.

1.2. Arbre d'une formule

La définition inductive des formules conduit naturellement à une représentation sous forme d'arbre (la syntaxe *abstraite*) d'une formule propositionnelle : les feuilles sont les variables propositionnelles, et les nœuds internes les connecteurs.

Par exemple, la formule $(\neg a \vee b) \wedge c \rightarrow a \wedge c$ précédente est représentée par l'arbre



Cette représentation permet de définir les notions suivantes :

- la *hauteur* d'une formule est la hauteur de l'arbre ;
- la *taille* d'une formule est nombre de nœuds (feuilles et nœuds internes), c'est-à-dire la somme du nombre de variables et de connecteurs qui interviennent dans la formule ;
- une *sous-formule* est une formule représentée par un sous-arbre.

2. Sémantique

On cherche à donner un sens à une formule logique, plus précisément à lui attribuer une valeur de vérité booléenne : vrai ou faux. On doit pour cela donner une valeur à chacune des variables qui figurent dans la formule (ce que l'on appelle une valuation), et les règles d'évaluation de chacun des opérateurs (table de vérité).

On note V pour la valeur vraie, et F pour la valeur fausse.

2.1. Valeur d'une formule

Valuation

Une **valuation** (on dit aussi un **contexte**, ou un **environnement**, ou encore une **distribution de vérité**) est une application de l'ensemble des variables propositionnelles dans $\{V, F\}$.
Si $\mathcal{P}$ est un ensemble de variable, on note $\mathrm{Val}_{\mathcal{P}}$ l'ensemble des valuations sur $\mathcal{P}$ (ou simplement Val lorsqu'il n'y a pas d'ambiguïté sur les variables).

On peut noter $\mathrm{Val}_{\mathcal{P}} = \{V, F\}^{\mathcal{P}}$. En particulier, si une formule φ contient n variables, il y a 2^n valuations distinctes possibles pour évaluer φ.

Valeur de vérité d'une formule

Une valuation v étant donnée, on peut *évaluer* une formule logique pour lui attribuer une valeur de vérité. Le calcul se fait récursivement : la valeur de chaque variable est donnée par la valuation puis on utilise les **tables de vérité** des opérateurs logiques :

a	$\neg a$
F	V
V	F

a	b	$a \wedge b$
F	F	F
F	V	F
V	F	F
V	V	V

a	b	$a \vee b$
F	F	F
F	V	V
V	F	V
V	V	V

a	b	$a \rightarrow b$
F	F	V
F	V	V
V	F	F
V	V	V

a	b	$a \leftrightarrow b$
F	F	V
F	V	F
V	F	F
V	V	V

Exercice n°1

1. Donner la valeur de vérité de la formule $(\neg a \vee b) \wedge c \rightarrow a \wedge c$ avec la valuation qui associe V à b et à c, et F à a.
2. Donner la table de vérité de la formule $\neg(a \wedge b)$ (opérateur NAND, ou connecteur de Sheffer).
3. Donner la table de vérité du connecteur « ou exclusif » XOR.

2.2. Modèles

Modèle

Un **modèle** d'une formule φ est une valuation qui rend vraie cette formule.
Notations :

- on note $v \models \varphi$ pour signifier que la valuation v est un modèle de φ, et $v \not\models \varphi$ dans le cas contraire.
- on note $\mathrm{Mod}(\varphi)$ l'ensemble des modèles de φ.

Exercice n°2

Soit φ et ψ deux formules, exprimer $\mathrm{Mod}(\neg\varphi)$, $\mathrm{Mod}(\varphi \wedge \psi)$ et $\mathrm{Mod}(\varphi \vee \psi)$ en fonction de $\mathrm{Mod}(\varphi)$ et $\mathrm{Mod}(\psi)$.

Tautologie, satisfiabilité, antilogie

Une formule propositionnelle φ est

- une **tautologie** (on dit aussi une formule **valide**) lorsque φ est évaluée à V pour toute valuation.
  On note dans ce cas $\models \varphi$.
- **satisfiable** s'il existe une valuation qui évalue φ en V,
- une **antilogie** (on dit aussi une **contradiction)** lorsque φ n'est pas satisfiable.

Remarques :

- On peut donner des définitions équivalentes : φ est une tautologie lorsque $\mathrm{Mod}(\varphi) = \mathrm{Val}$, est satisfiable lorsque $\mathrm{Mod}(\varphi) \neq \emptyset$, est une antilogie lorsque $\mathrm{Mod}(\varphi) = \emptyset$.
- On peut aussi voir ces propriétés sur la table de vérité de la formule : φ est une tautologie lorsque la table ne contient que des V, est satisfiable si elle contient au moins un V, et est une antilogie si elle ne contient que des F.
- La formule φ est une antilogie si et seulement si $\neg\varphi$ est une tautologie.
- La formule φ est satisfiable si et seulement si $\neg\varphi$ n'est pas une tautologie.

Par exemple, pour toute formule φ :

- $\varphi \vee \neg\varphi$ est une tautologie (*principe du tiers exclu*)
- $\varphi \wedge \neg\varphi$ est une antilogie (*principe de non-contradiction*).

Exercice n°3

Montrer qu'on a : $\models (a \wedge (a \to b)) \to b$ (règle de *modus ponens*).

Formules constantes tautologique et antilogique

On peut ajouter à la syntaxe des formules logiques :

- la formule tautologique, notée $\top$,
- la formule antilogique, notée $\perp$.

On a donc $\mathrm{Mod}(\top) = \mathrm{Val}$ et $\mathrm{Mod}(\perp) = \emptyset$. On n'est pas obligé de considérer ces deux formules logiques, car on peut les remplacer par $a \vee \neg a$ et $a \wedge \neg a$ pour n'importe quelle variable a. Par exemple, il n'est pas utile en cas de preuve par induction structurelle sur les formules de les prendre en compte.

2.3. Équivalence sémantique

Équivalence sémantique

Deux formules propositionnelles φ et ψ sont **sémantiquement équivalentes** si leurs évaluations coïncident pour toute valuation.
On note alors $\varphi \equiv \psi$.

Dit autrement : $\varphi \equiv \psi$ lorsque $\mathrm{Mod}(\varphi) = \mathrm{Mod}(\psi)$, ou encore lorsque les deux formules ont la même table de vérité.

Par exemple, on a les équivalences classiques suivantes (a, b, c sont des variables)

- Éléments neutres :
  - $a \wedge \top \equiv a$
  - $a \vee \perp \equiv a$
- Idempotence
  - $a \vee a \equiv a$
  - $a \wedge a \equiv a$
- Associativité et commutativité :
  - $(a \wedge b) \wedge c \equiv a \wedge (b \wedge c)$
  - $(a \vee b) \vee c \equiv a \vee (b \vee c)$
  - $a \wedge b \equiv b \wedge a$
  - $a \vee b \equiv b \vee a$
- Distributivité
  - $a \wedge (b \vee c) \equiv (a \wedge b) \vee (a \wedge c)$
  - $a \vee (b \wedge c) \equiv (a \vee b) \wedge (a \vee c)$
- Lois de De Morgan :
  - $\neg (a \wedge b) \equiv \neg a \vee \neg b$
  - $\neg (a \vee b) \equiv \neg a \wedge \neg b$

- $\neg\neg a \equiv a$ (double négation)
- $(a \to b) \equiv (\neg a \vee b)$ (décomposition d'une implication)
- $(a \leftrightarrow b) \equiv (a \to b) \wedge (b \to a)$ (double implication)
- $(a \to b) \equiv (\neg b \to \neg a)$ (raisonnement par contraposée)
- $a \equiv (\neg a \to \bot)$ (raisonnement par l'absurde)

Exercice n°4

Montrer les équivalences :

- $(a \to b) \wedge (\neg a \to b) \equiv b$ (disjonction de cas)
- $(a \wedge b \to c) \equiv (a \to (b \to c)$ (exportation)

Substitution dans une formule

Soit φ et ψ deux formules et a une variable. La substitution de a par ψ dans φ, notée $\varphi\{a \mapsto \psi\}$, est la formule obtenue en remplaçant chaque occurence de a par ψ dans φ.

Par exemple, si $\varphi = a \wedge b$ alors $\varphi\{a \mapsto c \to d\} = (c \to d) \wedge b$.

On peut définir plus précisément cette notion récursivement :

- si $\varphi = a$ alors $\varphi\{a \mapsto \psi\} = \psi$
- si $\varphi = b$ avec $b \neq a$, alors $\varphi\{a \mapsto \psi\} = b$
- si $\varphi = \neg\varphi_1$, alors $\varphi\{a \mapsto \psi\} = \neg(\varphi_1\{a \mapsto \psi\})$
- si $\varphi = \varphi_1 \wedge \varphi_2$, alors $\varphi\{a \mapsto \psi\} = \varphi_1\{a \mapsto \psi\} \wedge \varphi_2\{a \mapsto \psi\}$
  avec des définitions analogues pour les connecteurs $\vee$, $\to$ et $\leftrightarrow$

Équivalence et substitutions

Soit φ, ψ_1 et ψ_2 des formules telles que $\psi_1 \equiv \psi_2$, et a une variable. Alors on a les équivalences :

- $\varphi\{a \mapsto \psi_1\} \equiv \varphi\{a \mapsto \psi_2\}$
- $\psi_1\{a \mapsto \varphi\} \equiv \psi_2\{a \mapsto \varphi\}$

Ce résultat permet de remplacer dans les équivalences classiques de la page précédente les variables a, b et c par des formules quelconques φ, ψ, θ.

Exercice n°5

Montrer que pour toutes formules φ, ψ et θ, on a l'équivalence : $\neg(\varphi \to \psi) \equiv \varphi \wedge \neg\psi$ (négation d'une implication)

Équivalence et tautologie

Pour toutes formules φ et ψ, on a :

$$\varphi \equiv \psi \text{ si et seulement si } \models \varphi \leftrightarrow \psi$$

Autrement dit : les formules φ et ψ sont sémantiquements équivalentes si et seulement si la formule $(\varphi \leftrightarrow \psi)$ est une tautologie.

2.4. Conséquence logique d'une formule

Conséquence logique entre deux formules

Une formule φ est une **conséquence logique** d'une formule ψ lorsque tout modèle de ψ est un modèle de φ.
On note dans ce cas $\psi \models \varphi$.

Dit autrement : $\psi \models \varphi$ lorsque $\mathrm{Mod}(\psi) \subset \mathrm{Mod}(\varphi)$, donc lorsque toute valuation qui satisfait ψ satisfait également φ.

Exercice n°6

Montrer les conséquences logiques :

- $a \wedge b \wedge c \models a \wedge b$
- $(a \to b) \wedge a \models b$

On a le résultat suivant : $\boxed{\psi \models \varphi \text{ si et seulement si} \models \psi \to \varphi}$.

Modèle d'un ensemble de formules

Soit Γ un ensemble de formules. Une valuation v est un **modèle** de Γ lorsque v est un modèle de chaque formule de Γ.
On note $\mathrm{Mod}(\Gamma)$ l'ensemble des modèles de Γ.

On dira que Γ est satisfiable lorsque $\mathrm{Mod}(\Gamma) \neq \varnothing$ et insatisfiable dans le cas contraire.

Exercice n°7

Montrer que si $\Gamma = \{\psi_1, \ldots, \psi_n\}$ est fini, alors $v \in \mathrm{Mod}(\Gamma)$ si et seulement si $v \models \psi_1 \wedge \ldots \wedge \psi_n$.

Conséquence logique d'un ensemble de formules

Soit Γ un ensemble de formules et φ une formule. On dit que φ est une **conséquence logique** de Γ lorsque $\mathrm{Mod}(\Gamma) \subset \mathrm{Mod}(\varphi)$.
On note alors $\Gamma \models \varphi$.

Exercice n°8

Soit $\psi_1, \ldots, \psi_n$ des formules, montrer que $\{\psi_1, \ldots, \psi_n\} \models \varphi$ si et seulement si $\models (\psi_1 \wedge \ldots \wedge \psi_n) \rightarrow \varphi$.

3. Formes normales

Littéral

On appelle **littéral** une variable booléenne a ou sa négation $\neg a$.

Clause

Une **clause** est une disjonction de littéraux, c'est-à-dire une formule de la forme $\bigvee_{i=1}^{n} l_i$, où l_i est un littéral.
On appelle **longueur** de la clause le nombre de littéraux qui la composent.

Forme normale conjonctive

Une formule φ est en forme normale conjonctive (FNC) si c'est une conjonctions de clauses, c'est-à-dire si elle est sous la forme

$$\varphi = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{n_i} l_{ij}$$

où chaque l_{ij} est un littéral.

Par exemple, $(\neg a \vee b \vee \neg c) \wedge (\neg b \vee c) \wedge a$ est en FNC.

Forme normale disjonctive

Une formule φ est en forme normale disjonctive (FND) si c'est une disjonction de conjonctions de littéraux, c'est-à-dire si elle est sous la forme

$$\varphi = \bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} l_{ij}$$

où chaque l_{ij} est un littéral.

Par exemple, $(a \wedge b) \vee (\neg b \wedge c) \vee \neg c$ est en FND.

Mise sous FNC et FND

Toute formule φ est équivalente à une formule sous FNC et à une formule sous FND.

Exercice n°9

Expliquer comment la table de vérité d'une formule φ permet d'obtenir une formule sous FND équivalente à φ.
Par exemple : mettre la formule $\varphi = a \text{ XOR } (\neg b \rightarrow c)$ sous FND et sous FNC, où XOR désigne le ou exclusif.