Logique

Cours de Licence de Sciences du Langage (L2)
Alain Lecomte – Professeur, Université Paris 8

3 Logique propositionnelle

3.1 Interprétation propositionnelle de l'algèbre de Boole

3.1.1 Calcul propositionnel

- Une candidate à la signification qu'on peut accorder à une variable booléenne est la notion logique de **proposition**,
- Une proposition est une entité qui est soit **vraie** (1) soit **fausse** (0)
- Le **calcul propositionnel** est donc une « interprétation concrète » du calcul booléen quand **1** est interprété comme **Vrai** (*v*) et **0** comme **Faux** (*f*)
- Le calcul propositionnel est donc une algèbre de Boole où les variables, appelées *variables propositionnelles*, représentent des **propositions**, c'est-à-dire des *entités ayant pour valeurs possibles: le vrai (v) ou le faux (f)*
- Les **expressions booléennes** contenant de telles variables s'interprètent aisément.

3.1.2 Des expressions booléennes aux formules de logique propositionnelle

- Il est d'usage de noter p, q, r, … les variables propositionnelles,
- Il est d'usage aussi de noter $\vee$ ce qu'on avait noté +, $\wedge$ pour $\times$, $\neg$ pour ~
- Ces symboles sont appelés des **connecteurs**

3.1.3 Conjonction, disjonction et négation

- Ainsi les connecteurs $\wedge$ et $\vee$ permettent de définir les propositions $p \wedge q$ et $p \vee q$, dont les valeurs de vérité sont calculées au moyen des tables suivantes :

p	q	**p∨q**	p	q	**p∧q**
v	v	**v**	v	v	**v**
v	f	**v**	v	f	**f**
f	v	**v**	f	v	**f**
f	f	**f**	f	f	**f**

- Ainsi, $p \vee q$ est vrai si et seulement si l'un des deux (ou les deux) de p et de q est vrai
- $p \wedge q$ est vrai si et seulement si p et q sont vrais simultanément
- D'où la lecture qu'on donne à ces symboles. $p \vee q$ se lit : *p ou q* $p \wedge q$ se lit : *p et q*
- De même, on peut définir la proposition $\neg p$, qui, bien sûr, s'interprète comme la *négation* de p (qui se lit non-p)

p	**¬p**
v	**f**
f	**v**

- Les exemples précédents montrent qu'une nouvelle proposition est construite à partir de deux propositions p et q en déterminant **quelle est sa valeur de vérité** pour **chaque situation** concernant les valeurs de vérité de p et q,
- Il y a 4 situations possibles pour deux propositions :

(v,v), (v, f), (f, v), (f, f)

- **Exercice** : *combien y en a-t-il quand 3 propositions entrent en jeu ? quand n propositions entrent en jeu ?*
- Il y a donc autant de propositions obtenues à partir de p et q (donc autant de connecteurs binaires) qu'il y a de *fonctions* associant v ou f à chacune de ces situations.
- Ces fonctions sont appelées **fonctions de vérité**, elles sont représentées par des tables: **tables de vérité**.
- Comme nous n'avons pas de moyens de distinguer deux propositions hormis par les valeurs de vérité qu'elles prennent dans les mêmes situations, nous sommes amenés à identifier une proposition avec sa fonction de vérité: c'est ce qu'on appelle le **principe d'extensionnalité**.
- De ce qui précède, on déduit qu'on peut facilement procéder au recensement de toutes les propositions composées à partir de deux propositions,
- *Effectuer ce recensement…*
- *Idem pour les propositions obtenues à partir **d'une seule** proposition*

3.1.4 Définition de l'implication

- Une autre manière de « découvrir » des connecteurs consiste à combiner entre eux ceux que nous connaissons déjà…
- Ainsi, il est bien connu que… dire « qu'il n'y a pas de fumée sans feu » revient à dire que « s'il y a de la fumée (quelque part) alors il y a du feu (pas loin!) »
- D'où l'idée de définir un connecteur correspondant à « si… alors… », noté $\Rightarrow$, par:

$$\boxed{p \Rightarrow q =\text{def} \neg(p\wedge \neg q)}$$

implication

- Il est facile d'en déduire la table de vérité de ce connecteur:

p	q	**p⇒q**
v	v	**v**
v	f	**f**
f	v	**v**
f	f	**v**

- Bien noter que $p \Rightarrow q$ **n'est faux que si p est vrai et que q est faux**
- En particulier, $p \Rightarrow q$ est **vrai** lorsque **p est faux**,
- $p \Rightarrow q$ est **vrai** également lorsque q est **vrai,**
- Vérifier qu'on aurait pu tout aussi bien définir $p \Rightarrow q$ par : $\neg p\vee q$
- Dans l'expression $p \Rightarrow q$, on dit souvent que p est la **condition suffisante** de q, ou que q est la **condition nécessaire** de p,

- En français, la condition suffisante s'exprime généralement par un « si », exemple: « **si** la température dépasse 37°2 (p) alors le patient est malade (q)»,
- La condition nécessaire s'exprime généralement par « seulement si» ou « que si », exemple: « le patient n'est malade (q) **que si** sa température dépasse 37°2 (p)»
- Dans le premier cas, la proposition p : « la température dépasse 37°2 » est condition suffisante (de la maladie, c'est-à-dire q), dans le deuxième cas, elle est condition nécessaire, donc dans le premier cas, on a $p \Rightarrow q$, et dans le second, on a $q \Rightarrow p$,
- Bien sûr, le connecteur $\Rightarrow$ n'est pas symétrique ($p \Rightarrow q \neq q \Rightarrow p$), c'est tout l'intérêt de sa table de vérité!

3.1.5 Autres connecteurs

- Fabriquer les tables de vérité des connecteurs obtenus des manières suivantes:
- $P \Leftrightarrow Q =_{def} (P \Rightarrow Q) \wedge (Q \Rightarrow P)$
- $PWQ =_{def} (P \vee Q) \wedge \neg(P \wedge Q)$
- $P \mid Q =_{def} \neg P \wedge \neg Q$
- Leur donner des interprétations intuitives

3.2 Le langage propositionnel

Nous avons désormais un stock de symboles utilisés:

  - Les variables propositionnelles,
  - Les connecteurs : $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$, W
  - Des signes de ponctuation (les parenthèses)

- Nous pouvons les utiliser pour définir un **langage**: le langage de la logique propositionnelle LP

Définition:

  - Toute variable propositionnelle est une expression de ce langage,
  - Si P est une expression de ce langage, alors $\neg P$ l'est aussi
  - Si P et Q sont deux expressions de ce langage, alors $(P \wedge Q)$, $(P \vee Q)$,$(P \Rightarrow Q)$, $(P \Leftrightarrow Q)$, (PWQ) le sont aussi,
  - Rien d'autre n'est une expression de ce langage hormis par les trois clauses précédentes.

Théorème fondamental

> Soit $P=\{p_1, p_2, \ldots, p_n\}$ un ensemble de variables propositionnelles et soit L(P) l'ensemble des expressions qui ne contiennent que les variables incluses dans P.
> Pour toute assignation δ de valeurs de vérité à $p_1, p_2, \ldots, p_n$ il existe une et une seule fonction val_δ de L(P) dans $\{0, 1\}$ qui coïncide avec δ sur P et qui soit telle que, pour toutes expressions a et b dans L(P)
> $val_\delta (b \wedge a) = 1$ si et seulement si $val_\delta (a) = val_\delta (b) = 1$
> $val_\delta (b \vee a) = 0$ si et seulement si $val_\delta (a) = val_\delta (b) = 0$
> $val_\delta (a \Rightarrow b) = 0$ si et seulement si $val_\delta (a) = 1$ et $val_\delta (b) = 0$
> $val_\delta (a \Leftrightarrow b) = 1$ si et seulement si $val_\delta (a) = val_\delta (b)$
> $val_\delta (\neg a) = 1$ si et seulement si $val_\delta (a) = 0$

- *Ce théorème dit simplement que quelle que soit l'expression du langage propositionnel que nous construisons au moyen des symboles donnés, elle aura une interprétation définie de lanière unique au moyen des tables de vérités élémentaires associées aux connecteurs* $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$.

- Deux propositions (expressions du langage propositionnel) seront dites équivalentes si leurs fonctions de vérité sont identiques. On écrira alors :

$$\boxed{\varphi \equiv \psi}$$

Pour dire que φ et ψ sont équivalentes

3.3 La notion de loi en logique propositionnelle

3.3.1 Tautologies et contradictions

- Un cas particulier de fonction de vérité est donné par *une fonction constante*. C'est soit le cas où, à toute situation de vérité est associé 1, soit le cas où, à toute telle situation est associé 0. Wittgenstein a appelé ***tautologie*** une proposition qui tombe sous le premier cas et ***contradiction*** une proposition qui tombe sous le second. Une tautologie est donc une proposition ϕ telle que pour toute assignation possible de valeurs de vérité δ, on ait:

$$\mathrm{val}_\delta(\phi) = 1$$

- Nous écrirons:

$$\boxed{\models A}$$

pour dire que "A est une tautologie"

- **Théorème 1** : *on a les tautologies suivantes:*

$\models p \Rightarrow p$	(loi d'identité pour l'implication)
$\models p \Leftrightarrow p$	(loi d'identité pour l'équivalence)
$\models p \vee \neg p$	(loi du tiers exclu)
$\models \neg (p \wedge \neg p)$	(loi de non-contradiction)
$\models \neg (p \wedge q) \Leftrightarrow (\neg p \vee \neg q)$	
$\models \neg (p \vee q) \Leftrightarrow (\neg p \wedge \neg q)$	(lois de De Morgan)

- Noter que les lois de De Morgan peuvent aussi bien s'écrire:

$$\boxed{\begin{array}{c} \neg (p \wedge q) \equiv (\neg p \vee \neg q) \\ \neg (p \vee q) \equiv (\neg p \wedge \neg q) \end{array}}$$

- D'une façon générale, on démontrera que:

Théorème 2 : $\models$ *(A $\Leftrightarrow$ B) si et seulement si A $\equiv$ B. (où A et B sont n'importe quelles propositions construites à partir de propositions élémentaires).*

- Etant donné une tautologie, on peut en construire une infinité à partir d'elle. Pour cela il suffit d'appliquer le théorème suivant:

Théorème 3 : *Soit une proposition composée de variables propositionnelles: a_1, a_2, ... a_n, de connecteurs et de parenthèses. Nous la noterons: $A(a_1, a_2, ..., a_n)$. Si $A(a_1, a_2, ... a_n)$ est une tautologie, alors en substituant dans $A(a_1, a_2,... a_n)$ des propositions quelconques A_1, A_2, ... A_n aux symboles a_1, a_2...a_n, on obtient encore une tautologie.*

Exemple: $((a \wedge b) \Rightarrow c) \vee \neg ((a \wedge b) \Rightarrow c)$ est une tautologie.

3.3.2 *Règles d'inférence*

- **Définition**: Soit A_1, A_2, ..., A_n des propositions composées de variables propositionnelles, de connecteurs et de parenthèses. Une proposition B sera dite **être une conséquence logique** de A_1, A_2, ... A_n et nous écrirons:

  $$\{A_1 , ... , A_n\} \models B$$

  si et seulement si: pour toute assignation de valeurs de vérité aux variables propositionnelles occurrant dans A_1, A_2, ...A_n et B, chaque fois que A_1, A_2, ...A_n sont vraies, alors B est aussi vraie.

- **Attention**: voici un nouvel usage du signe "|=". Cette fois, il établit un lien entre un ensemble de propositions et une proposition. Nous allons plus loin établir le rapport entre cet usage et l'usage de "|=" qui est fait pour indiquer la présence d'une tautologie (cf théorème 4). *Noter aussi que bien évidemment, "|=" n'est pas un connecteur, puisqu'il n'est pas au même niveau que les connecteurs*. Ce n'est pas un signe avec lequel on *construit* des propositions, c'est un signe avec lequel on exprime *une relation entre* propositions. (C'est, si on veut, un "méta-signe").

- **Exemple**: règle du détachement, ou du *Modus Ponens* :

$$\boxed{\{A , (A \Rightarrow B)\} \models B}$$

- Cette règle d'inférence est la plus fameuse. Frege a démontré qu'elle était suffisante dans le cadre d'un système formel du calcul propositionnel pour démontrer tous les théorèmes. Mais elle n'est pas toujours la plus commode à employer. De nos jours, en informatique, les ***systemes-experts*** ont introduit les règles d'inférence dans de nombreuses applications. Ainsi un ***moteur d'inferences*** est un programme qui applique à une ***base de connaissance*** des ***regles d'inference*** qui permettent de déduire des "théorèmes". La méthode consistant à utiliser le ***modus ponens*** est assimilable à ce qu'on appelle le ***chainage avant*** : pour établir si un "fait" est vrai, on part des faits déjà admis puis on "avance" en appliquant un nombre indéterminé de fois la règle du détachement. On voit tout de suite l'inconvénient: une telle règle est appliquée à l'aveugle. Si on n'arrive pas à démontrer le fait considéré, ce n'est pas nécessairement parce qu'il est faux ... mais peut-être parce que la chaine d'inférence à accomplir est plus longue que celle où nous nous sommes arrêtés ou bien parce qu'on est mal parti...etc. D'où l'intérêt d'autres règles.

- Autre règle: le *Modus Tollens* :

$$\boxed{\{(\neg B \Rightarrow \neg A) , A\} \models B}$$

- L'application de cette règle correspond au CHAINAGE ARRIERE. Pour démontrer le fait B, on part de sa négation et on "remonte" jusqu'à ce qu'on ait trouvé la négation d'un fait antérieurement admis.

- *Reductio ad absurdum*

$$\boxed{\{\neg B \Rightarrow (A \wedge \neg A)\} \models B}$$

C'est la règle classiquement employée dans ce qu'on nomme le "raisonnement **par l'absurde**" (improprement, car il n'y a rien d'"absurde" là dedans!)

3.3.3 Tautologie et relation de conséquence

- On démontre facilement:

Théorème 4 :

a) $\models (A \Rightarrow B)$ *si et seulement si* $A \models B$

b) $\{A_1, A_2, ..., A_k\} \models B$ *si et seulement si* $\{A_1, A_2, ..., A_{k-1}\} \models (A_k \Rightarrow B)$

- Ce théorème justifie qu'on emploie le même signe pour indiquer la tautologie et pour symboliser la relation de déduction: dire qu'une proposition est une tautologie, c'est dire qu'elle est vraie sans aucune hypothèse particulière. On peut donc dire que "|= A" signifie en fait: "Ø |= A".

3.3.4 Retour sur l'implication.

- Nous avons dit plus haut que la définition donnée de l'implication était adéquate pour le raisonnement logico-mathématique. C'est ce que nous sommes maintenant en mesure de voir grâce à l'introduction des règles d'inférence. Considérons en effet la règle "universelle" qui est celle du MODUS PONENS. Si nous voulons déduire Q à partir de (P ⇒ Q), *elle impose que P soit vrai.* Il n'y a donc aucun danger dans l'acceptation des situations 0 ⇒ 1 et 0 ⇒ 0 dans la valeur de vérité "1" de p ⇒ q. D'une proposition fausse nous pouvons déduire n'importe quoi, certes. Mais ... une proposition fausse ne saurait être vraie! Aucune règle d'inférence ne nous permet donc d'utiliser une proposition fausse pour déduire quelque chose. Ici se marque bien la différence entre **déduction** et **implication**, différence cruciale pour toute la théorie logique. (Voir aussi les fameux sophismes de Lewis Carroll, et particulièrement celui intitulé: "Ce que se dirent Achille et la Tortue"...)

- *Remarque*: on se convaincra facilement que tout autre choix de valeurs de vérité pour les deux dernières lignes de la table du "⇒" conduirait soit à une symétrie des deux côtés du "⇒" soit à la possibilité de déduire quelque chose de la fausseté de l'antécédent, ce que nous ne voulons pas. Le propre de l'implication (p ⇒ q) est de dire que lorsque p est vrai, nécessairement q l'est aussi, mais que **lorsque p est faux, on ne peut *rien* dire**.