Logique des propositions

F. Alexandre

École Supérieure d'Informatique et Applications de Lorraine

December 17, 2008

Plan

- Définition des formules de la logique des propositions
- Sémantique de la logique des propositions (liens avec les fonctions booléennes)
- Systèmes formels du calcul des propositions
- Atomes, littéraux, clauses

Definition (inductive)

Soit P un ensemble de symboles (appelés les variables propositionnelles), soit $C = \{\mathcal{V},\ \mathcal{F},\ \neg,\ \wedge,\ \vee,\ \Rightarrow,\ \Leftrightarrow\}$ l'ensemble des connecteurs logiques et $D = \{(,\ )\}$.
L'ensemble des formules de la logique des propositions construites sur P, noté $Prop(P)$ est défini inductivement par

- la base $B = P \cup \{\mathcal{V},\ \mathcal{F}\}$
- les opérations :

$$(\forall p \in Prop(P))\ (\forall q \in Prop(P)) \qquad \begin{array}{l} \neg p \in Prop(P) \text{ et} \\ (p \wedge q) \in Prop(P) \text{ et} \\ (p \vee q) \in Prop(P) \text{ et} \\ (p \Rightarrow q) \in Prop(P) \text{ et} \\ (p \Leftrightarrow q) \in Prop(P) \end{array}$$

Remarques

- les éléments de P peuvent être considérés comme des faits élémentaires (dans la suite les éléments de P prendront la valeur 0 ou 1 (sémantique))
- $Prop(P)$ est un ensemble de mots construits sur l'alphabet $P \cup C \cup D$.
- cette définition inductive de $Prop(P)$ permet de faire des preuves par récurrence sur $Prop(P)$ et de définir certains concepts par récurrence

Remarques

- parfois on utilise seulement deux connecteurs $\neg$ et $\vee$, les autres sont alors considérés comme des abréviations (il y a alors moins de cas à considérer lorsque l'on fait des preuves)
  - $x \Rightarrow y$ est une abréviation pour $\neg x \vee y$
  - $x \wedge y$ pour $\neg(\neg x \vee \neg y)$,
  - $x \Leftrightarrow y$ pour $(x \Rightarrow y) \wedge (y \Rightarrow x) \ldots$
- cette définition conduit à "surparenthéser" les formules de la logique des propositions

Definition

Soit P un ensemble, une formule du calcul des propositions sur P est un mot engendré par la grammaire algébrique suivante : $G = (\{X\}, \{\mathcal{V}, \mathcal{F}, a, \neg, \vee, \wedge, \Rightarrow, \Leftrightarrow, (,)\}, \rightarrow, X)$ où
$X \rightarrow \mathcal{V} \mid \mathcal{F} \mid a \mid \neg X \mid X \vee X \mid X \wedge X \mid X \Rightarrow X \mid X \Leftrightarrow X \mid (X)$
où a est un élément quelconque de P. $Prop(P)$ est l'ensemble des formules engendrées par cette grammaire.

Remarques

- La grammaire donnée n'est pas "optimale", elle est ambiguë et ne tient pas compte des priorités des différents connecteurs logiques les uns par rapport aux autres.
- *Priorité des opérateurs, ordre de priorité décroissante :* $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$
  Suppression des parenthèses autour des variables propositionnelles
  $((\neg(p)) \wedge (q))$ *devient* $\neg p \wedge q$
  $((\neg(p)) \vee (q)) \Rightarrow (r)$ *devient* $\neg p \vee q \Rightarrow r$
  $((\neg p)) \vee ((q) \Rightarrow (r))$ *devient* $\neg p \vee (q \Rightarrow r)$

Example

$P = \{y,\ z,\ t,\ u,\ v\}$, un ensemble de variables propositionnelles

- $\mathcal{V} \in Prop(P)$
- $\mathcal{F} \in Prop(P)$
- $u \in Prop(P)$
- $X \rightarrowtail (X) \rightarrowtail (X \Rightarrow X) \rightarrowtail ((X) \Rightarrow X) \rightarrowtail ((X) \Rightarrow (X))$
  $\rightarrowtail ((X \Rightarrow X) \Rightarrow (X)) \rightarrowtail ((X \Rightarrow X) \Rightarrow (X \Rightarrow X))$
  $\rightarrowtail ((y \Rightarrow X) \Rightarrow (X \Rightarrow X)) \rightarrowtail ((y \Rightarrow u) \Rightarrow (X \Rightarrow X))$
  $\rightarrowtail ((y \Rightarrow u) \Rightarrow (X \Rightarrow z)) \rightarrowtail ((y \Rightarrow u) \Rightarrow (y \Rightarrow z))$
  $((y \Rightarrow u) \Rightarrow (y \Rightarrow z)) \in Prop(P)$.
- $((u \Rightarrow t) \wedge ((z \vee \neg y) \Rightarrow u)) \Rightarrow (t \vee u) \in Prop(P)$
- $(x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z)) \in Prop(P)$

Definition

On définit une application [ ] de $Prop(P)$ vers $\{0,\ 1\}^P$ de la façon suivante :

- $[\mathcal{V}] \ = \ 1$
- $[\mathcal{F}] \ = \ 0$
- $[x] \ = \ x \quad (x \in P)$
- $[\neg\alpha] \ = \ \overline{[\alpha]} \quad (\alpha \in Prop(P))$
- $[\alpha \vee \beta] \ = \ [\alpha] + [\beta] \qquad (\alpha \in Prop(P) \text{ et } \beta \in Prop(P))$
- $[\alpha \wedge \beta] \ = \ [\alpha].[\beta] \qquad (\alpha \in Prop(P) \text{ et } \beta \in Prop(P))$
- $[\alpha \Rightarrow \beta] \ = \ [\alpha]{\Rightarrow}[\beta] \qquad (\alpha \in Prop(P) \text{ et } \beta \in Prop(P))$
- $[\alpha \Leftrightarrow \beta] \ = \ [\alpha]{\Leftrightarrow}[\beta] \qquad (\alpha \in Prop(P) \text{ et } \beta \in Prop(P))$

Definition

On appelle valuation des variables propositionnelles une application $\delta : P \longrightarrow \{0,\ 1\}$

Definition (sémantique)

Soient une formule α de $Prop(P)$ et δ une valuation de P. La sémantique de la formule α sur la valuation δ, que l'on note $[\alpha](\delta)$ (ou $\delta(\alpha)$) est la valeur obtenue en remplaçant dans $[\alpha]$ chaque variable propositionnelle x par $\delta(x)$.

- $\beta = x \wedge \neg y$
  $[\beta]$ est la fonction booléenne $(x,\ y) \mapsto x.\overline{y}$
- $\alpha = x \vee (\neg y \Rightarrow z)$
  - $[\alpha]$ est la fonction booléenne $(x,\ y,\ z) \mapsto x + (\overline{y} \Rightarrow z)$
  - soit δ_1 tel que $\delta_1(x) = 1$, $\delta_1(y) = 1$, $\delta_1(z) = 0$, $[\alpha](\delta_1) = 1$
  - soit δ_2 tel que $\delta_2(x) = 0$, $\delta_2(y) = 0$, $\delta_2(z) = 0$, $[\alpha](\delta_2) = 0$
  - soit δ_3 tel que $\delta_3(x) = 0$, $\delta_3(y) = 0$, $\delta_3(z) = 1$, $[\alpha](\delta_3) = 1$

Definition

α est une formule de $Prop(P)$ et $\delta : P \longrightarrow \{0,\ 1\}$ une valuation des variables propositionnelles de P.

- δ est un modèle de α si et seulement si $[\alpha](\delta) = 1$
- α est une tautologie si et seulement si **pour toute valuation** δ, $[\alpha](\delta) = 1$
- α et β sont sémantiquement équivalentes si et seulement si $\alpha \Leftrightarrow \beta$ est une tautologie.
- α est contradictoire si et seulement si α ne possède pas de modèle
- $\mathcal{A}$ un ensemble de formules de $Prop(P)$, δ est un modèle de $\mathcal{A}$ si et seulement si δ est un modèle de chacune des formules de $\mathcal{A}$, c'est à dire $\forall \alpha \in \mathcal{A}\ \ [\alpha](\delta) = 1$
- $\mathcal{A}$ un ensemble de formules de $Prop(P)$, $\mathcal{A}$ est contradictoire si et seulement si $\mathcal{A}$ ne possède pas de modèle, c'est à dire que $\forall \delta : P \longrightarrow \{0,\ 1\}\ \ \exists \alpha \in \mathcal{A}\ \ [\alpha](\delta) = 0$

Exemple (modèle)

$\alpha = x \wedge (y \vee z)$, on a $[\alpha] = f$ tel que $f(x,\ y,\ z) = x(y+z)$,
Table de vérité de α

x	y	z	$y \vee z$	$\alpha = x \wedge (y \vee z)$
1	1	1	1	1
1	1	0	1	1
1	0	1	1	1
1	0	0	0	0
0	1	1	1	0
0	1	0	1	0
0	0	1	1	0
0	0	0	0	0

Les modèles de α sont les valuations δ_1, δ_2 et δ_3 telles que

- $\delta_1(x) = 1,\ \delta_1(y) = 1,\ \delta_1(z) = 1$
- $\delta_2(x) = 1,\ \delta_2(y) = 1,\ \delta_2(z) = 0$
- $\delta_3(x) = 1,\ \delta_3(y) = 0,\ \delta_3(z) = 1$

$\alpha = x \Rightarrow (y \Rightarrow x)$

- Table de vérité de α

x	y	$y \Rightarrow x$	$x \Rightarrow (y \Rightarrow x)$
1	1	1	1
1	0	1	1
0	1	0	1
0	0	1	1

  La formule α est une tautologie car toute valuation est un modèle de α.

- Comment montrer que α est une tautologie sans construire la table de vérité de α.
  Supposons qu'il existe une valuation δ telle que $[\alpha](\delta) = 0$.
  Si $[x \Rightarrow (y \Rightarrow x)](\delta) = 0$
  on a $\delta(x) = 1$ et $[(y \Rightarrow x)](\delta) = 0$,
  d'où $\delta(x) = 1$ et $\delta(y) = 1$ et $\delta(x) = 0$
  d'où contradiction, ce qui montre qu'il n'existe pas de valuation δ telle que $[\alpha](\delta) = 0$, α est donc une tautologie.

Exemple (formule contradictoire)

$\alpha = x \wedge \neg(y \Rightarrow x)$

- Table de vérité de α

x	y	$y \Rightarrow x$	$\neg(y \Rightarrow x)$	α
1	1	1	0	0
1	0	1	0	0
0	1	0	1	0
0	0	1	0	0

  La formule α est une formule contradictoire car aucune valuation n'est un modèle de α

- Comment montrer que α est contradictoire sans construire la table de vérité de α.
  Supposons qu'il existe une valuation δ telle que $[\alpha](\delta) = 1$.
  Si $[x \wedge \neg(y \Rightarrow x)](\delta) = 1$
  on a $\delta(x) = 1$ et $[\neg(y \Rightarrow x)](\delta) = 1$,
  d'où $\delta(x) = 1$ et $[y \Rightarrow x](\delta) = 0$,
  d'où $\delta(x) = 1$ et $\delta(y) = 1$ et $\delta(x) = 0$
  d'où contradiction, ce qui montre qu'il n'existe pas de valuation δ telle que $[\alpha](\delta) = 1$, α est donc contradictoire.

Exemple (modèle d'un ensemble de formules)

$\alpha = x \Rightarrow y$, $\beta = x \vee z$ et $\gamma = y \vee z$
Table de vérité de α, β, γ

x	y	z	$x \Rightarrow y$	$x \vee z$	$y \vee z$
1	1	1	1	1	1
1	1	0	1	1	1
1	0	1	0	1	1
1	0	0	0	1	0
0	1	1	1	1	1
0	1	0	1	0	1
0	0	1	1	1	1
0	0	0	1	0	0

Les modèles de l'ensemble de formules $\{\alpha,\ \beta,\ \gamma\}$ sont les valuations δ_1,δ_2, δ_3 et δ_4 suivantes :

- $\delta_1(x) = 1,\ \delta_1(y) = 1,\ \delta_1(z) = 1$
- $\delta_2(x) = 1,\ \delta_2(y) = 1,\ \delta_2(z) = 0$
- $\delta_3(x) = 0,\ \delta_3(y) = 1,\ \delta_3(z) = 1$
- $\delta_4(x) = 0,\ \delta_4(y) = 0,\ \delta_4(z) = 1$

Definition

Soi $\mathcal{A}$ un ensemble de formules de $Prop(P)$ et α une formule de $Prop(P)$, on dit que α se déduit "sémantiquement" de $\mathcal{A}$ si et seulement si tout modèle de $\mathcal{A}$ est un modèle de α. On note

$$\mathcal{A} \models \alpha$$

.

Remarque : si $\mathcal{A} = \emptyset$, on note $\models \alpha$ au lieu de $\emptyset \models \alpha$ (cela signifie que α est une tautologie)

Propriétés : $\mathcal{A} \subset Prop(P)$, $\mathcal{B} \subset Prop(P)$, $\alpha \in Prop(P)$, $\beta \in Prop(P)$.

- Si $\mathcal{A} \subset \mathcal{B}$ alors $\forall \alpha \in Prop(P)$ (si $\mathcal{A} \models \alpha$ alors $\mathcal{B} \models \alpha$).
- $\mathcal{A} \models \alpha$ si et seulement si l'ensemble $\mathcal{A} \cup \{\neg\alpha\}$ est contradictoire (c'est le principe du raisonnement par contradiction).
- $\mathcal{A} \cup \{\alpha\} \models \beta$ si et seulement si $\mathcal{A} \models \alpha \Rightarrow \beta$ (c'est le lemme de détachement).

Théorème

$\mathcal{A} \subset Prop(P)$ et $\alpha \in Prop(P)$

- Forme 1 : $\mathcal{A}$ admet un modèle si et seulement si tout sous-ensemble fini de $\mathcal{A}$ admet un modèle.
- Forme 2 : α se déduit sémantiquement de $\mathcal{A}$ si et seulement si α se déduit sémantiquement d'un sous-ensemble fini de $\mathcal{A}$.
- Forme 3 : $\mathcal{A}$ est contradictoire si et seulement si l'un de ses sous-ensembles finis est contradictoire.

Remarque : intuitivement cela signifie que l'activité de déduction est intrinsèquement finie et qu'il est impossible de tenir compte d'une infinité d'hypothèses pour déduire une formule.

Definition (Système formel)

Un système formel est un triplet $(E,\ \mathcal{A},\ \mathcal{R})$ où

- E est un ensemble non vide (de formules)
- $\mathcal{A} \subset E$, l'ensemble des axiomes
- $\mathcal{R}$ est un ensemble des règles de déduction de la forme $\dfrac{e_1, \ldots, e_n}{e_{n+1}}(r)$
  cette règle se lit : " e_{n+1} se déduit de $e_1 \ldots e_n$ par la règle r

Definition (Démonstration dans un système formel)

Soit $(E,\ \mathcal{A},\ \mathcal{R})$ un système formel, et $\mathcal{H}$ un sous-ensemble de E. Une démonstration dans $\mathcal{S}$ avec hypothèses dans $\mathcal{H}$ est une suite finie $e_1, \ldots, e_n$ de formules de E telles que:

- soit $e_i \in \mathcal{A}$ (e_i est un axiome)
- soit $e_i \in \mathcal{H}$ (e_i est une hypothèse)
- soit il existe une règle de déduction r de $\mathcal{R}$ telle pour des indices $j_1, \ldots, j_k$ tous strictement inférieurs à i on a ait $\dfrac{e_{j_1}, \ldots, e_{j_k}}{e_i}(r)$

On note $\mathcal{H} \ \vdash_{\mathcal{S}} \ e_n$, on dit que la formule e_n se démontre dans le système formel $\mathcal{S}$ en utilisant les hypothèses de $\mathcal{H}$

Notation

$\mathcal{H} \ \vdash_{\mathcal{S}} \ e$
si $\mathcal{H} = \emptyset$, on note

$$\vdash_{\mathcal{S}} \ e$$

au lieu de $\emptyset \ \vdash_{\mathcal{S}} \ e$
on dit que e est un **théorème** du système formel $\mathcal{S}$

Proposition

Soit $\mathcal{S} = (E,\ \mathcal{A},\ \mathcal{R})$ un système formel, on a les résultats suivants :

- si $\mathcal{H} \subset \mathcal{H}'$ alors $\mathcal{H}\ \vdash_{\mathcal{S}}\ e$ implique $\mathcal{H}'\ \vdash_{\mathcal{S}}\ e$
  la logique des systèmes formels est monotone, plus on a d'hypothèses plus on peut démontrer de formules
- si $\mathcal{H}'\ \vdash_{\mathcal{S}}\ e$ et $\mathcal{H}'' \cup \{e\}\ \vdash_{\mathcal{S}}\ e'$ alors $\mathcal{H}' \cup \mathcal{H}'' \vdash_{\mathcal{S}}\ e'$

Definition (Propriétés : validité, complétude)

Soit $\mathcal{S} = (Prop(P),\ \mathcal{A},\ \mathcal{R})$ un système formel sur le calcul des propositions

- $\mathcal{S}$ est un système formel valide si et seulement si pour tout $\mathcal{H}$ tel que $\mathcal{H} \subset Prop(P)$ et pour toute formule $\alpha \in Prop(P)$ on a :

$$\mathcal{H} \vdash_{\mathcal{S}}\ \alpha\ \ \textit{implique}\ \ \mathcal{H} \models\ \alpha$$

- $\mathcal{S}$ est un système formel complet si et seulement si
  pour tout $\mathcal{H}$ tel que $\mathcal{H} \subset Prop(P)$ et pour toute formule $\alpha \in Prop(P)$ on a :

$$\mathcal{H} \vdash_{\mathcal{S}}\ \alpha\ \ \textit{équivaut}\ \ \mathcal{H} \models\ \alpha$$

Proposition

Soit $\mathcal{S} = (Prop(P),\ \mathcal{A},\ \mathcal{R})$ un système formel du calcul des propositions, $\mathcal{S}$ est valide si et seulement si :

- les axiomes de $\mathcal{S}$ sont des tautologies
- pour chaque règle de la forme $\dfrac{e_1, \ldots, e_n}{e_{n+1}}(r)$ de $\mathcal{S}$ on a $\{e_1, \ldots, e_n\} \models e_{n+1}$ on dit que chaque règle est valide.

Remarque : la complétude est en général plus difficile à démontrer.

Soit le système formel $\mathcal{S} = (Prop(P),\ \mathcal{A},\ \mathcal{R})$ sur le calcul des propositions tel que

- $\mathcal{A} = \{\mathcal{V},\ \neg\mathcal{F},\ a_1,\ a_2,\ a_3,\ a_4\}$ où
  $a_1 = x \Rightarrow (y \Rightarrow x)$,
  $a_2 = (x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z))$,
  $a_3 = x \Rightarrow (\neg x \Rightarrow y)$,
  $a_4 = (x \Rightarrow y) \Rightarrow ((\neg x \Rightarrow y) \Rightarrow y)$
- $\mathcal{R} = \{ \dfrac{\alpha,\quad \alpha \Rightarrow \beta}{\beta} (\textit{modus ponens}) \}$

Remarque : les axiomes $a_1,\ a_2,\ a_3,\ a_4$ sont des schémas d'axiomes : on peut remplacer chaque variable propositionnelle x, y, z, par une formule quelconque du calcul des propositions.

Mettons en évidence une démonstration de $x \Rightarrow x$,
(a_2) $(x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z))$
(f_1) $(x \Rightarrow ((x \Rightarrow x) \Rightarrow x)) \Rightarrow ((x \Rightarrow (x \Rightarrow x)) \Rightarrow (x \Rightarrow x))$ (substitution)
(a_1) $x \Rightarrow (y \Rightarrow x)$,
(f_2) $x \Rightarrow ((x \Rightarrow x) \Rightarrow x)$
(f_3) $((x \Rightarrow (x \Rightarrow x)) \Rightarrow (x \Rightarrow x))$ (modus ponens avec f_1 et f_2)
(a_1) $x \Rightarrow (y \Rightarrow x)$,
(f_4) $x \Rightarrow (x \Rightarrow x)$, (substitution)

(f_5) $x \Rightarrow x$ (modus ponens avec f_3 et f_4)

$a_1 = x \Rightarrow (y \Rightarrow x)$ est une tautologie :

(a_1)

x	y	$y \Rightarrow x$	$x \Rightarrow (y \Rightarrow x)$
1	1	1	1
1	0	1	1
0	1	0	1
0	0	1	1

$a_2 = (x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z))$
$a_2 = \quad b_2 \quad \Rightarrow \quad c_2$

x	y	z	$y \Rightarrow z$	b_2	$x \Rightarrow y$	$x \Rightarrow z$	c_2	a_2
1	1	1	1	1	1	1	1	1
1	1	0	0	0	1	0	0	1
1	0	1	1	1	0	1	1	1
1	0	0	1	1	0	0	1	1
0	1	1	1	1	1	1	1	1
0	1	0	0	1	1	1	1	1
0	0	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1

(a_2) est une tautologie

$a_3 = x \Rightarrow (\neg x \Rightarrow y)$
$a_4 = (x \Rightarrow y) \Rightarrow ((\neg x \Rightarrow y) \Rightarrow y)$
sont des tautologies (à démontrer)

Le modus ponens est une règle valide
$\dfrac{\alpha, \quad \alpha \Rightarrow \beta}{\beta}$(*modus ponens*) , on doit démontrer que

$$\{\alpha,\ \alpha \Rightarrow \beta\} \models \beta$$

α	β	$\alpha \Rightarrow \beta$	β
1	1	1	1
1	0	0	
0	1	0	
0	0	1	

Definition

Soit P un ensemble de variables propositionnelles, un atome (ou littéral) construit sur P et soit une variable propositionnelle, soit une négation de variable propositionnelle.

Example

Soit $P = \{p,\ q,\ r,\ s\}$ un ensemble de variables propositionnelles,

- $\neg p$, q, $\neg r$, s sont des atomes construits sur P
- q, s sont des atomes positifs
- $\neg p$ et $\neg r$ sont des atomes négatifs.

Definition

Soit P un ensemble de variables propositionnelles, une clause sur P est une formule de $Prop(P)$ de la forme

$$a_1 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_{k+r}$$

où les a_i sont des éléments tous **distincts** de P. On note $CL(P)$ l'ensemble des clauses sur P.

Remarque

$a \vee \neg a$ (sémantiquement égale à la formule $\mathcal{V}$) n'est pas une clause

Notations

Suivant la notation on peut noter la clause c sous la forme suivante :

- Définition : $\mathbf{c} = a_1 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_{k+r}$
- Formes implicatives :
  - $\mathbf{c} = \big(a_1 \vee \ldots \vee a_k\big) \Leftarrow \big(a_{k+1} \wedge \ldots \wedge a_{k+r}\big)$
  - $\mathbf{c} = \big(a_{k+1} \wedge \ldots \wedge a_{k+r}\big) \Rightarrow \big(a_1 \vee \ldots \vee a_k\big)$
- Forme ensembliste :
  $\mathbf{c} = (\{a_1, \ldots a_k\},\ \{a_{k+1}, \ldots, a_{k+r}\}) = (c^+,\ c^-)$

Si $k = 0$ et $r = 0$ la clause n'a pas d'atomes (sous forme implicative : conjonction vide "implique" disjonction vide, donc $1 \Rightarrow 0$, c'est à dire $\mathcal{F}$), on la note $\Box$.

Definition (Subsomption)

Soient deux clauses c et c' définies sur P, on dit que c subsume c' si et seulement si une des deux conditions équivalentes suivantes est vérifiée :

- (condition syntaxique) : tout littéral de c apparaît dans c'
- (condition sémantique) : $\{c\} \models c'$ (i.e. pour toute valuation δ des variables de P, $[c](\delta) = 1$ implique $[c'](\delta) = 1$)

- $\mathbf{c} = p \vee q \vee \neg t$ subsume $\mathbf{c}' = p \vee q \vee r \vee \neg s \vee \neg t$
- pour toute clause **c**, la clause vide $\Box$ subsume **c**
- si l'on pose $c \succeq c'$ si et seulement c subsume c', $\succeq$ est une relation d'ordre partiel sur les clauses
  - $\Box$ est le plus grand élément pour cette relation
  - Il n'y a pas de plus petit élément pour cette relation.
    Si P est fini tel que $card(P) = n$ les éléments minimaux sont les clauses comportant n littéraux.
    Si P est infini il n'existe pas d'éléments minimaux

Proposition

Soit α une formule de $Prop(P)$, α est (sémantiquement) équivalente à une formule β de la forme $\beta = c_1 \wedge \ldots \wedge c_n$ où les c_i sont des clauses.

Démonstration

La forme clausale d'une formule de $Prop(P)$ correspond à la forme canonique conjonctive disjonctive des fonctions booléennes $f(x_1, \ldots, x_n) = \prod_{(\varepsilon_1, \ldots, \varepsilon_n) \in \overline{S_n(f)}} (x_1^{\overline{\varepsilon_1}} + \ldots + x_n^{\overline{\varepsilon_n}})$

Si l'on remplace $+$ par $\vee$, . par $\wedge$ ($\prod$ par $\bigwedge$), on obtient la forme clausale $\bigwedge_i \bigvee_j a_{i,\ j}$

Definition

Mettre une formule α sous forme clausale c'est trouver un ensemble de clauses $C(\alpha)$ dont la conjonction est équivalente à la formule α.

- $\alpha = p \Rightarrow q$, on a $\alpha = \neg p \vee q$ d'où $C(\alpha) = \{\neg p \vee q\}$
- $\alpha = p \wedge \neg q$, on a $C(\alpha) = \{p,\ \neg q\}$
- $\alpha = p \Leftrightarrow q$, $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow p)$,
  $\alpha = (\neg p \vee q) \wedge (\neg q \vee p)$, d'où $C(\alpha) = \{\neg p \vee q,\ \neg q \vee p\}$
- $\alpha = (p \wedge (q \Rightarrow r)) \Rightarrow s$,
  $= (p \wedge (\neg q \vee r)) \Rightarrow s$
  $= \neg(p \wedge (\neg q \vee r)) \vee s$
  $= (\neg p \vee \neg(\neg q \vee r)) \vee s$
  $= (\neg p \vee (q \wedge \neg r)) \vee s$
  $= ((\neg p \vee q) \wedge (\neg p \vee \neg r)) \vee s$
  $= (\neg p \vee q \vee s) \wedge (\neg p \vee \neg r \vee s)$
  $C(\alpha) = \{\neg p \vee q \vee s,\ \neg p \vee \neg r \vee s\}$

Proposition

Soient α et β deux formules de $Prop(P)$, soient $C(\alpha)$ et $C(\beta)$ les ensembles de clauses associées respectivement à α et β, on a :

- $C(\alpha \wedge \beta) = C(\alpha) \cup C(\beta)$
- $C(\alpha \vee \beta) = C(\alpha) \otimes C(\beta)$ où $E \otimes E' = \{c \vee c'; c \in E \text{ et } c' \in E'\}$
  ($E \otimes E'$ est l'ensemble des clauses obtenues à partir des clauses de E et E' en effectuant des disjonctions, on élimine les formules contenant un atome et la négation de cet atome, car ces formules ne sont pas des clauses)

Démonstration

Soit $C(\alpha) = \{c_1, \ldots, c_m\}$ et $C(\beta) = \{d_1, \ldots, d_n\}$, on a
$\alpha \Leftrightarrow c_1 \wedge \ldots \wedge c_m$ et $\beta \Leftrightarrow d_1 \wedge \ldots \wedge d_n$, donc

- $\alpha \wedge \beta \Leftrightarrow c_1 \wedge \ldots \wedge c_m \wedge d_1 \wedge \ldots \wedge d_n$
  $C(\alpha \wedge \beta) = \{c_1, \ \ldots, \ c_m, \ d_1, \ \ldots, d_n\} = C(\alpha) \cup C(\beta)$
- $\alpha \vee \beta \Leftrightarrow (c_1 \wedge \ldots \wedge c_m) \vee (d_1 \wedge \ldots \wedge d_n)$
  en appliquant la distributivité de $\vee$ par rapport à $\wedge$
  $(c_1 \vee d_1) \wedge \ldots \wedge (c_1 \vee d_n) \wedge \ldots \wedge (c_m \vee d_1) \wedge \ldots \wedge (c_m \vee d_n)$

Ensemble vide de clauses et clause vide

Chaque formule α de $Prop(P)$ est équivalente à la conjonction d'un ensemble de clauses $C(\alpha) = \{c_1, \ldots, c_n\}$

$$\alpha \Leftrightarrow c_1 \wedge \ldots \wedge c_n$$

- la formule $\mathcal{V}$ est équivalente à l'ensemble vide de clauses $\emptyset$ (car une conjonction vide est équivalente à $\mathcal{V}$ l'élément neutre de $\wedge$)
- la formule $\mathcal{F}$ est la clause $\Box$, elle est donc équivalente à l'ensemble $\{\Box\}$