Logique des propositions

TELECOM Nancy (1A)

Mathématiques Appliquées pour l'Informatique

2019-2020

Plan

- *Syntaxe* : 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
- Algorithme de mise sous forme clausale
- Système formel de Robinson (résolution)
- Références :
  - MATHÉMATIQUES DISCRÈTES. Automates, langages, logique et décidabilité. Pierre Marchand. DUNOD
  - Outils logiques pour l'INTELLIGENCE ARTIFICIELLE. J.P. Delahaye. EYROLLES. 1988.

Définition des formules du calcul des propositions (1)

Définition (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 : $\forall p, q \in Prop(P)$, les cinq expressions

  $$(p \vee q) \qquad (p \Rightarrow q) \qquad (p \wedge q) \qquad (p \Leftrightarrow q) \qquad \neg p$$

  sont dans $Prop(P)$.

Remarques

- Les éléments de P peuvent être considérés comme des faits élémentaires (dans la suite, lorsque l'on étudiera la sémantique, les éléments de P prendront la valeur 0 ou la valeur 1).
- $Prop(P)$ 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 induction sur $Prop(P)$.

Remarques sur la définition

Remarques

- Cette définition conduit à “surparenthéser” les formules de la logique des propositions
- Parfois on utilise seulement deux connecteurs $\neg$ et $\vee$, les autres sont alors considérés comme des abréviations :
  - $x \Rightarrow y$ est une abréviation pour $(\neg x) \vee y$
  - $x \wedge y$ pour $\neg(\neg x \vee \neg y)$
  - $x \Leftrightarrow y$ pour $(x \Rightarrow y) \wedge (y \Rightarrow x)$

  Il y a alors moins de cas à considérer lorsque l'on fait des preuves.

Définition

Soit P un ensemble, une formule du calcul des propositions sur P est un mot engendré par la grammaire algébrique suivante :

$$G = (\{X\},\ \{\mathcal{V},\ \mathcal{F},\ a,\ \neg,\ \vee,\ \wedge,\ \Rightarrow,\ \Leftrightarrow,\ (,\ )\}, \rightarrow,\ X)$$

$$X \rightarrow \quad \mathcal{V} \mid \mathcal{F} \mid a \mid \neg X \mid X \vee X \mid X \wedge X \mid X \Rightarrow X \mid X \Leftrightarrow X \mid (X)$$

où a est un élément quelconque de P. On définit : $Prop(P) = L(G)$.

Remarques

- La grammaire donnée n'est pas "optimale." Elle est ambiguë et ne tient pas compte des priorités entre les différents connecteurs logiques.
- 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 $\neg p \vee (q \Rightarrow r)$

Exemples

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

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

Sémantique d'une formule du calcul des propositions

Définition

On définit une application [ ] de $Prop(P)$ vers $\mathbb{F}$ (l'ensemble des fonctions booléennes dont les variables appartiennent à P) de la façon suivante :

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

Définition

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

Définition (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 $\delta(x)$.

Exemples (sémantique d'une formule)

- $\beta = x \wedge \neg y$
  $[\beta]$ 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 $\delta_3(x) = 0$, $\delta_3(y) = 0$, $\delta_3(z) = 1$, $[\alpha](\delta_3) = 1$

Définition

Soient α une formule de $Prop(P)$, $\delta : P \to \{0,\ 1\}$ une valuation des variables propositionnelles de P et $\mathcal{A}$ un ensemble de formules de $Prop(P)$.

- 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
- 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 si $\forall \alpha \in \mathcal{A}\ \ [\alpha](\delta) = 1$
- $\mathcal{A}$ est contradictoire si et seulement si $\mathcal{A}$ ne possède pas de modèle, c'est-à-dire si $\forall \delta : P \to \{0,\ 1\}\ \ \exists \alpha \in \mathcal{A}\ \ [\alpha](\delta) = 0$

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

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

Exemple (tautologie)

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

- **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)

$$\mathcal{A} = \{\alpha = x \Rightarrow y, \quad \beta = x \vee z, \quad \gamma = y \vee z\}$$

Table de vérité de α, β et 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 $\mathcal{A}$ sont les quatres valuations suivantes :

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

Définition

Soit $\mathcal{A}$ 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 : Soient $\mathcal{A}, \mathcal{B} \subset$ *Prop*(P) et $\alpha, \beta \in$ *Prop*(P).

- Si $\mathcal{A} \subset \mathcal{B}$ alors pour toute proposition α de *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 de finitude (ou compacité)

Théorème de compacité

$\mathcal{A} \subset Prop(P)$ 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.

Remarques :

- Intuitivement cela signifie que l'activité de déduction est intrinséquement finie et qu'il est impossible de tenir compte d'une infinité d'hypothèses pour déduire une formule.
- La démonstration de ce théorème fait intervenir des notions de topologie. On admet ce théorème.

Définition (système formel)

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

- E est un ensemble non vide de formules (bien formées).
- $\mathcal{A} \subset E$, l'ensemble des axiomes
- $\mathcal{R}$ est un ensemble de règles de déduction de la forme $\dfrac{e_1, \ldots, e_n}{e_{n+1}}(r)$

  où les e_i sont des formules de E et r est le nom de la règle.
  Cette règle se lit : "e_{n+1} se déduit de $e_1 \ldots e_n$ par la règle r".

Définition (démonstration dans un système formel)

Soit $\mathcal{S} = (E,\ \mathcal{A},\ \mathcal{R})$ 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 e_i est telle qu'il existe une règle de déduction $r \in \mathcal{R}$ et des indices j_1 à j_k tous strictement inférieurs à i vérifiant $\dfrac{e_{j_1}, \ldots, e_{j_k}}{e_i}(r)$

Notation

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 $\mathcal{H}$ (e_n est la dernière formule de la démonstration).

Si $\mathcal{H} = \emptyset$, on note

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

au lieu de

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

Dans ce cas, 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 $\mathcal{H}' \cup \mathcal{H}'' \vdash_{\mathcal{S}}\ e'$

Définitions (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 :

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

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.

Exemple : système formel du calcul des propositions (Hilbert)

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
  $a_1 = x \Rightarrow (y \Rightarrow x)$,
  $a_2 = (x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z))$,
  $a_3 = x \Rightarrow (\neg x \Rightarrow y)$,
  $a_4 = (x \Rightarrow y) \Rightarrow ((\neg x \Rightarrow y) \Rightarrow y)$
- $\mathcal{R} = \left\{ \dfrac{\alpha, \quad \alpha \Rightarrow \beta}{\beta} (\mathit{modus\ ponens}) \right\}$

Remarque : $a_1,\ a_2,\ a_3,\ a_4$ 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 :

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

$a_2 = (x \Rightarrow (y \Rightarrow z)) \Rightarrow ((x \Rightarrow y) \Rightarrow (x \Rightarrow z))$ est une tautologie :

$a_2 = \quad b_2 \quad \Rightarrow \quad c_2$

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

Validité du système formel de Hilbert (preuve suite)

$a_3 = x \Rightarrow (\neg x \Rightarrow y)$
$a_4 = (x \Rightarrow y) \Rightarrow ((\neg x \Rightarrow y) \Rightarrow y)$
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	1	
0	0	1	

Il suffit d'examiner la première ligne de la table ci-dessus.

Définition

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

Exemple

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.

Définition

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 équivalente 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 :
  - $\mathbf{c} = (a_1 \vee \ldots \vee a_k) \Leftarrow (a_{k+1} \wedge \ldots \wedge a_{k+r})$
  - $\mathbf{c} = (a_{k+1} \wedge \ldots \wedge a_{k+r}) \Rightarrow (a_1 \vee \ldots \vee a_k)$
- 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 $\square$.

Définition (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 $\mathbf{c}$, la clause vide $\Box$ subsume $\mathbf{c}$
- si l'on pose $c \succeq c'$ si et seulement si 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 booléennees

$$f(x_1, \ldots, x_n) = \prod_{(\varepsilon_1, \ldots, \varepsilon_n) \in \overline{S_n(f)}} (x_1^{\overline{\varepsilon_1}} + \ldots + x_n^{\overline{\varepsilon_n}})$$

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

Définition

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.

Exemple de formules sous forme clausale

- $\alpha = p \Rightarrow q$, 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 $E \otimes E' = \{c \vee c'; c \in E \textit{ et } c' \in E'\}$ ($E \otimes E'$ 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 clause et clause vide

Remarques

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 $\square$, elle est donc équivalente à l'ensemble $\{\square\}$

Proposition

Soit α une formule de $Prop(P)$, α est une tautologie si et seulement l'ensemble $C(\alpha)$ obtenue par l'algorithme de mise sous forme clausale est égal à $\emptyset$.

Règle de résolution, méthode de résolution

- Objectif : montrer que $\mathcal{A} \models \alpha$
- Ce qui est équivalent à montrer que $\mathcal{A} \cup \{\neg\alpha\}$ est contradictoire
- En considérant les formes clausales si $\mathcal{A} = \{\beta_1, \ldots, \beta_n\}$, on veut montrer que l'ensemble de clauses $K = (\bigcup_{i=1}^{n} C(\beta_i)) \cup C(\neg\alpha)$ est contraditoire
- Pour cela, on veut montrer que $K \models \mathcal{F}$
- Dans un système formel $\mathcal{S}$ valide, cela revient à trouver $K \vdash_{\mathcal{S}} \square$, à partir des clauses de K et de la règle de résolution

Système formel de Robinson (basé sur les clauses)

Définition

Soit P un ensemble de variables propositionnelles et $\mathcal{C}(P)$ l'ensemble des clauses construites sur P. Le système formel $(\mathcal{C}(P),\ \emptyset,\ \mathcal{R})$

$$\text{où } \mathcal{R} = \{ \frac{c_1 \lor a \lor c_2, \quad c_3 \lor \neg a \lor c_4}{c_1 \lor c_2 \lor c_3 \lor c_4}(\mathit{resolution}),$$

$$\frac{c_1 \lor a \lor c_2 \lor a \lor c_3}{c_1 \lor a \lor c_2 \lor c_3}(\mathit{fact}+),$$

$$\frac{c_1 \lor \neg a \lor c_2 \lor \neg a \lor c_3}{c_1 \lor \neg a \lor c_2 \lor c_3}(\mathit{fact}-)$$

$$\}$$

est un système basé sur la règle de résolution (c_1, c_2, c_3, c_4 sont des clauses a est une variable propositionnelle).

Remarque

La règle la plus importante du système formel de Robinson est la règle de résolution, les règles de factorisations (positive et négative) sont souvent appliquées de façon implicite.

Proposition

Le système formel de Robinson est valide.

Démonstration : Il faut montrer que les trois règles du système formel de Robinson sont valides.

- La régle de résolution est valide, c'est-à-dire que
  $\{c_1 \vee a \vee c_2, \quad c_3 \vee \neg a \vee c_4\} \models c_1 \vee c_2 \vee c_3 \vee c_4$
  Soit δ telle que $[c_1 \vee a \vee c_2](\delta) = 1$ et $[c_3 \vee \neg a \vee c_4](\delta) = 1$
  on a $[c_1 \vee c_2](\delta) = 1$ ou $\delta(a) = 1$
    - si $[c_1 \vee c_2](\delta) = 1$ alors $[c_1 \vee c_2 \vee c_3 \vee c_4](\delta) = 1$
    - si $\delta(a) = 1$ alors $[\neg a](\delta) = 0$ comme $[c_3 \vee \neg a \vee c_4](\delta) = 1$
      on a $[c_3 \vee c_4](\delta) = 1$
      et donc $[c_1 \vee c_2 \vee c_3 \vee c_4](\delta) = 1$
- Les règles de factorisation sont trivialement valides.

Validité et complétude du système formel de Robinson

Théorème de Robinson

Soit C un ensemble de clauses, C est contradictoire si et seulement s'il existe une démonstration de $\square$ avec hypothèses dans C. On a donc les équivalences suivantes :

$$\mathcal{A} \models \alpha$$
$$\textit{ssi}$$
$$C(\mathcal{A}) \cup C(\neg\alpha) \textit{ est contradictoire}$$
$$\textit{ssi}$$
$$C(\mathcal{A}) \cup C(\neg\alpha) \vdash_{Resolution} \square$$

Démonstration

Ce théorème établit la validité et la complétude du système formel basé sur la résolution. La complétude exprime que si un ensemble de clauses est contradictoire il est possible de trouver une démonstration menant à la clause vide $\square$. (voir démonstration livre PM)

Exemple de “puzzle” logique

On a le fait suivant :
Si l’assemblée nationale refuse de voter la loi alors la grève ne s’arrêtera pas à moins qu’elle ne dure depuis plus d’un an et que le président de la firme démissionne.
Peut-on déduire que la grève n’arrêtera pas si l’assemblée nationale refuse de voter la loi et si la grève vient juste de commencer ? Détermination des variables propositionnelles suivantes p, q, r et s suivantes :

- p : l’assemblée refuse de voter la loi
- q : la grève est finie
- r : le président de la firme démissionne
- s : la grève dure depuis plus d’un an

Exemple de “puzzle” logique (suite et fin)

On exprime le fait par $p \to (\neg q \vee (r \wedge s))$
peut-on déduire que $(p \wedge \neg s) \to \neg q$,
c'est-à-dire que $\{p \to (\neg q \vee (r \wedge s))\} \models (p \wedge \neg s) \to \neg q$
c'est-à-dire $\{p \to (\neg q \vee (r \wedge s)),\ \neg((p \wedge \neg s) \to \neg q)\}$ est contradictoire

- Mise sous forme clausale des deux formules : $C(p \to (\neg q \vee (r \wedge s))) = \{\neg p \vee \neg q \vee r,\ \ \neg p \vee \neg q \vee s\}$
  $C(\neg((p \wedge \neg s) \to \neg q)) = \{p,\ \neg s,\ q\}$
- On montre en utilisant la règle de résolution que l'ensemble de clauses $\{\neg p \vee \neg q \vee r,\ \ \neg p \vee \neg q \vee s,\ p,\ \neg s,\ q\}$ est contradictoire :
- On a $\dfrac{\neg p \vee \neg q \vee s,\quad p}{\neg q \vee s}(res)$

$$\frac{\neg q \vee s,\quad q}{s}(res)$$

$$\frac{s,\quad \neg s}{\square}(res)$$