Logique : le calcul propositionnel

Sophie Pinchinat
sophie.pinchinat@irisa.fr

IRISA, Université de Rennes 1

UE LOG – année 2015-2016

Syntaxe

Définition

Le langage du calcul propositionnel est formé de :

- variables propositionnelles/propositions $\text{Prop} = \{p, q, p_1, p_2, \ldots\}$
- connecteurs logiques $\{\neg, \vee, \wedge, \rightarrow\}$
- symboles auxiliaires : $\{(,), \sqcup\}$.

L'ensemble $\mathcal{F}_{cp}$ des formules du calcul propositionnel est le plus petit ensemble tel que :

- si $p \in \text{Prop}$, alors $p \in \mathcal{F}_{cp}$,
- si $\varphi \in \mathcal{F}_{cp}$, alors $\neg\varphi \in \mathcal{F}_{cp}$,
- si $\varphi, \psi \in \mathcal{F}_{cp}$, alors $\varphi \vee \psi, \varphi \wedge \psi, \varphi \rightarrow \psi$ appartiennent à $\mathcal{F}_{cp}$.

Symboles auxiliaires

- Les symboles auxiliaires (parenthèses et espace) sont utilisés pour lever les ambiguïtés possibles.
- On convient de l'ordre de priorité suivant entre les connecteurs :

$$\neg \text{ "précède" } \wedge \text{ "précède" } \vee \text{ "précède" } \rightarrow$$

  De sorte que

$$\neg q \vee p \wedge \neg r$$

  signifie

$$((\neg q) \vee (p \wedge (\neg r))$$

- On écrira $((p \vee q) \wedge r)$ au lieu de $p \vee q \wedge r$

Exemple

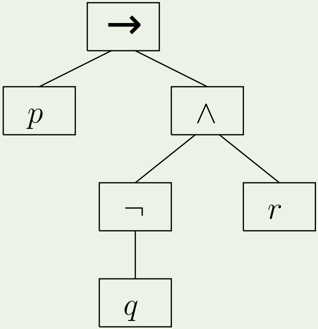
p, $p \rightarrow (q \vee r)$ et $p \vee q$ sont des formules (du calcul) propositionnelles; $\neg(\vee q)$ et $f(x) \rightarrow g(x)$ n'en sont pas.

Représentation arborescente des formules

De par la définition inductive de $\mathcal{F}_{cp}$, une formule possède une structure d'arbre dont les *feuilles* sont des propositions et les *noeuds internes* sont des connecteurs.

Exemple

La formule $p \to (\neg q \wedge r)$ correspond à l'arbre :



Notations

- On utilise souvent en plus le connecteur binaire $\leftrightarrow$ comme abréviation : $\varphi \leftrightarrow \psi$ est l'abréviation de $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$.

- On ajoute les symboles $\perp$ et $\top$ qui correspondent à "faux" et "vrai" en langue naturelle, respectivement.

  Les symboles $\perp$ et $\top$ sont aussi des abréviations, ils ne sont pas indispensables : $\perp$ correspond à $p \wedge \neg p$ et $\top$ correspond à $p \vee \neg p$.

Sous-formules strictes, sous-formules

Définition

L'ensemble $SFs(\varphi)$ des *sous-formules strictes* d'une formules φ est défini par induction de la façon suivante.

- $SFs(p) = \emptyset$;
- $SFs(\neg\varphi) = \{\varphi\} \cup SFs(\varphi)$;
- $SFs(\varphi \vee \psi) = SFs(\varphi \wedge \psi) = SFs(\varphi \to \psi) = \{\varphi, \psi\} \cup SFs(\varphi) \cup SFs(\psi)$.

L'ensemble $SF(\varphi)$ des *sous-formules* d'une formules φ est $\{\varphi\} \cup SFs(\varphi)$

Exemple

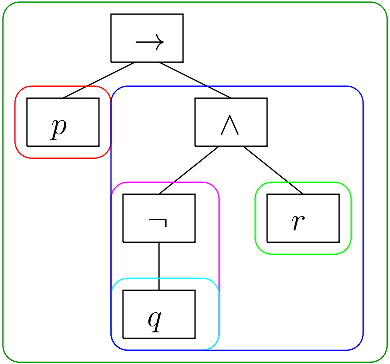
$SF(p \to (q \wedge r)) = \{p, q, r, q \wedge r, p \to (q \wedge r)\}$

Exercice

Faire le lien entre sous-formules strictes et sous-arbres des formules.

Représentation arborescente des sous-formules

Par exemple pour les sous-formules de $p \rightarrow (\neg q \wedge r)$:



Valuations

C'est à partir des valuations qu'on peut évaluer si une formule est vraie ou fausse. Une valuation détermine valeur de vérité des variables propositionnelles.

Définition

Une valuation est une application $\nu : \text{Prop} \to \{0, 1\}$.
La valeur 0 signifie "faux" et la valeur 1 signifie "vrai".

Une valuation sera souvent donnée sous forme d'un tableau :

Exemple

Soient $\text{Prop} = \{p, q\}$ et la valuation $\nu : p \mapsto 1, q \mapsto 0$. On écrit plus simplement

ν :

p	q
1	0

Valeur d'une formule

Définition

Soit ν : Prop $\to \{0,1\}$ une valuation.
On définit $\widehat{\nu} : \mathcal{F}_{cp} \to \{0,1\}$ par induction sur la structure des formules par :

$$
\begin{array}{lll}
\widehat{\nu}(p) = \nu(p) & & \text{si } p \in \text{Prop} \\
\widehat{\nu}(\neg\varphi) = 1 & \text{ssi} & \widehat{\nu}(\varphi) = 0 \\
\widehat{\nu}(\varphi \vee \psi) = 1 & \text{ssi} & \widehat{\nu}(\varphi) = 1 \text{ ou } \widehat{\nu}(\psi) = 1 \\
\widehat{\nu}(\varphi \wedge \psi) = 1 & \text{ssi} & \widehat{\nu}(\varphi) = 1 \text{ et } \widehat{\nu}(\psi) = 1 \\
\widehat{\nu}(\varphi \to \psi) = 0 & \text{ssi} & \widehat{\nu}(\varphi) = 1 \text{ et } \widehat{\nu}(\psi) = 0
\end{array}
$$

On remarque que $\widehat{\nu}(\neg\varphi) = 0$ ssi $\widehat{\nu}(\neg\varphi) \neq 1$ ssi $\widehat{\nu}(\varphi) \neq 0$ ssi $\widehat{\nu}(\varphi) = 1$

Exemple

L'énoncé "si tu dis la Vérité alors je tombe à la Renverse" que l'on peut écrire $V \to R$ se voit attribuer la valeur de vérité 1 même si tu mens. Cet énoncé n'est faux que si tu dis la Vérité et que je ne tombe pas à la Renverse : $V \wedge \neg R$.

Remarque

On écrira simplement ν au lieu de $\widehat{\nu}$.

Tables de vérité

On résume la manière d'évaluer une formule en fonction des valeurs de ses sous-formules au moyen des tables de vérité des connecteurs logiques suivantes :

φ	$\neg\varphi$
1	0
0	1

φ	ψ	$\varphi \vee \psi$
1	1	1
1	0	1
0	1	1
0	0	0

φ	ψ	$\varphi \wedge \psi$
1	1	1
1	0	0
0	1	0
0	0	0

φ	ψ	$\varphi \to \psi$
1	1	1
1	0	0
0	1	1
0	0	1

Remarques sur le connecteur $\rightarrow$

- Le connecteur "$\rightarrow$" s'entend en général comme un "si ... alors ...".
- D'après la table de vérité du connecteur "$\rightarrow$", l'énoncé

  "*si* $1+1=1$ *alors la capitale de la France est Rennes*"

  est vrai, puisque tout énoncé $\varphi \rightarrow \psi$ est vrai dès lors que φ est fausse.
- On convient de ce caractère peu naturel, car dans le langage courant, on ne s'intéresse à la vérité d'un tel énoncé que lorsque la condition est vraie : "s'il fait beau, je vais à la pêche" n'a d'intérêt pratique que s'il fait effectivement beau.

Autre formulation de la valeur d'une formule

On profite d'opérations arithmétiques sur le domaine $\{0, 1\}$, en posant :

Définition

$$
\begin{array}{rcl}
\nu(\neg\varphi) & = & 1 - \nu(\varphi) \\
\nu(\varphi \vee \psi) & = & max\{\nu(\varphi), \nu(\psi)\} \\
\nu(\varphi \wedge \psi) & = & min\{\nu(\varphi), \nu(\psi)\} \\
\nu(\varphi \rightarrow \psi) & = & \nu(\neg\varphi \vee \psi)
\end{array}
$$

Exercice

Vérifier que l'on obtient bien les mêmes résultats qu'avec la définition originelle.

Ensemble des valuations

Définition

On note Val_{Prop}, ou juste Val, l'ensemble des valuations de Prop dans $\{0,1\}$.

Exemple

$Val_{\{p,q,r\}}$ est représenté par le tableau à 2^3 lignes suivant :

p	q	r
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Chaque ligne est une valuation.

Remarque

Si on a n propositions dans Prop,
alors Val_{Prop} a 2^n éléments.

Modèles d'une formule

Définition

Soit $\varphi \in \mathcal{F}_{cp}$.

Un **modèle** de φ est une valuation ν telle que $\nu(\varphi) = 1$. On le notera :

$$\nu \models \varphi$$

On note $mod(\varphi)$ l'ensemble des modèles de φ.

Exemple

Soit Prop $= \{p, q, r\}$. L'ensemble des modèles de $\varphi = (p \vee q) \wedge (p \vee \neg r)$ est :

$mod(\varphi) =$

p	q	r
1	1	1
1	1	0
1	0	1
1	0	0
0	1	0

Exercice

Vérifier.

Satisfaisabilité, Tautologie

Définition

Une formule φ est satisfaisable si elle admet un modèle (*i.e.* si il existe une valuation ν telle que $\nu \models \varphi$). Sinon elle est insatisfaisable.

Exercice

- Proposer une formule satisfaisable et une formule insatisfaisable.
- Exprimer la propriété "φ est satisfaisable" en foncion de $mod(\varphi)$.

Définition

Une formule φ est valide/une tautologie si toute valuation est un modèle de φ. On le note $\models \varphi$.

Exercice

- Proposer une formule valide et une formule non valide.
- Exprimer la propriété "φ est valide" en foncion de $mod(\varphi)$.

Propriétés des ensembles de modèles

Proposition

1. $mod(\neg\varphi) = Val \setminus mod(\varphi)$;
2. $mod(\varphi \vee \psi) = mod(\varphi) \cup mod(\psi)$;
3. $mod(\varphi \wedge \psi) = mod(\varphi) \cap mod(\psi)$;
4. $\models \varphi \to \psi$ ssi $mod(\varphi) \subseteq mod(\psi)$.

Exercice

La preuve est laissée en exercice. Il faut la faire !

Une sélection de tautologies (1/3)

Exercice

Essayez d'en montrer quelques unes. Quelle est votre technique systématique ?

(Associativité et la commutativité de $\vee$, $\wedge$)

- $((\varphi \vee \psi) \vee \theta) \leftrightarrow (\varphi \vee (\psi \vee \theta))$
- $((\varphi \wedge \psi) \wedge \theta) \leftrightarrow (\varphi \wedge (\psi \wedge \theta))$
- $(\varphi \vee \psi) \leftrightarrow (\psi \vee \varphi)$
- $(\varphi \wedge \psi) \leftrightarrow (\psi \wedge \varphi)$

(Lois de distributivité)

- $(\varphi \wedge (\psi \vee \theta)) \leftrightarrow ((\varphi \wedge \psi) \vee (\varphi \wedge \theta))$
- $(\varphi \vee (\psi \wedge \theta)) \leftrightarrow ((\varphi \vee \psi) \wedge (\varphi \vee \theta))$

Une sélection de tautologies (2/3)

(Négation)

- $((\neg(\neg\varphi)) \leftrightarrow \varphi)$
- $((\neg(\varphi \to \psi)) \leftrightarrow (\varphi \wedge (\neg\psi)))$
- $((\neg(\varphi \leftrightarrow \psi)) \leftrightarrow ((\varphi \wedge (\neg\psi)) \vee ((\neg\varphi) \wedge \psi)))$

(Lois de De Morgan (1806-1871))

$((\neg(\varphi \wedge \psi)) \leftrightarrow ((\neg\varphi) \vee (\neg\psi)))$ et $((\neg(\varphi \vee \psi)) \leftrightarrow ((\neg\varphi) \wedge (\neg\psi)))$

(Autres)

- Tiers exclu : $(\varphi \vee (\neg\varphi))$ (vraie ou fausse, sans valeur intermédiaire)
- Non-contradiction : $(\neg(\varphi \wedge (\neg\varphi)))$
- Contraposée : $((\varphi \to \psi) \leftrightarrow ((\neg\psi) \to (\neg\varphi)))$
- Exportation : $(((\varphi \wedge \psi) \to \theta) \leftrightarrow (\varphi \to (\psi \to \theta)))$

Notion de formules équivalentes

Définition

On dit que les formule φ et ψ sont équivalente si elles ont les mêmes modèles (*i.e.* $mod(\varphi) = mod(\psi)$). On note alors $\varphi \equiv \psi$.

Exercice

- Traduire l'associativité et la commutativité de $\wedge$ et $\vee$ en termes d'équivalences de formules.
- Montrer que d'après leurs tables de vérité, les connecteurs $\wedge$, $\vee$ sont associatifs-commutatifs.
- Expliquer/montrer pourquoi pour toutes formules φ et ψ, on a :

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

Substitution

Définition

Soient $\varphi, \psi, \psi' \in \mathcal{F}_{cp}$. On souhaite définir $\varphi[\psi \hookleftarrow \psi']$ comme la formule résultant du remplacement dans φ de toutes les occurrences de ψ par ψ'.

Exemple

Soient $\varphi = (\neg p \vee q) \wedge (\neg p \vee \neg r)$, $\psi = \neg p$ et $\psi' = q \rightarrow p$. Alors,
$\varphi[\psi \hookleftarrow \psi'] = ((q \rightarrow p) \vee q) \wedge ((q \rightarrow p) \vee \neg r)$.

Exercice

Proposez une définition de $\varphi[\psi \hookleftarrow \psi']$ par induction sur la formule φ.

Proposition

Soient $\varphi, \psi, \psi' \in \mathcal{F}_{cp}$. Si $\psi \equiv \psi'$, alors $\varphi \equiv \varphi[\psi \hookleftarrow \psi']$.

Exercice

Démontrez la proposition ci-dessus.

Équivalences classiques

On peut aussi reformuler toutes les tautologies basées sur le connecteur $\leftrightarrow$ précédemment listées (Associativité et la commutativité de $\vee$ et $\wedge$, Lois de distributivité, etc.) en termes d'équivalences.

Proposition

Pour toutes formules φ et ψ, affirmer que $\models \varphi \leftrightarrow \psi$, c'est la même chose qu'affirmer $\varphi \equiv \psi$.

Autrement dit, toutes les valuations sont modèles de $\varphi \leftrightarrow \psi$ si, et seulement si, les deux formules s'évaluent à 1 exactement pour les mêmes valuations.

Exemple

- Comme $\models ((\varphi \vee \psi) \vee \theta) \leftrightarrow (\varphi \vee (\psi \vee \theta))$, on a $((\varphi \vee \psi) \vee \theta) \equiv (\varphi \vee (\psi \vee \theta))$.
- $(\neg(\neg\varphi)) \equiv \varphi$, $(\neg(\varphi \vee \psi)) \equiv ((\neg\varphi) \wedge (\neg\psi))$
- etc.

Exercice

Comment montreriez-vous la proposition ? Par induction ?

Conséquence logique

Définition

Une formule φ est **conséquence logique** d'une formule ψ, noté $\psi \models \varphi$, si tout modèle de ψ est un modèle de φ (*i.e.* $mod(\psi) \subseteq mod(\varphi)$).

Remarque

Attention à ne pas confondre deux notations qui se ressemblent :

- $\nu \models \varphi$ où ν est une valuation, et
- $\psi \models \varphi$ où ψ est une formule.

Proposition

1. $\varphi \models \psi$ ssi $\models \varphi \to \psi$ (*i.e.* , $\varphi \to \psi$ est une tautologie)
2. $\models \varphi \leftrightarrow \psi$ ssi $mod(\varphi) = mod(\psi)$

Preuve Au tableau.

Traitement de la conséquence logique

Les formules propositionnelles peuvent être vues comme un ensemble de contraintes sur les propositions. Ainsi, $p \wedge q$ contraint p et q à être vraies. Il est donc très courant de considérer des ensembles de formules propositionnelles pour modéliser des problèmes de satisfaction de contraintes. On étend les définitions vues précédemment aux ensembles de formules.

Définition

Soit Γ un ensemble de formules. Un modèle de Γ est une valuation ν qui est modèle de chaque formule de Γ.
Par extension, on note $mod(\Gamma)$ l'ensemble des modèles de Γ.

Remarque

Si Γ est un ensemble fini, on peut le voir comme le $\bigwedge$ de toutes ses formules.

$$\bigwedge_{\varphi \in \Gamma} \varphi \tag{1}$$

Lorsque Γ est infini, ce n'est plus possible car l'expression (1) n'est pas une formule.

Notions naturelles sur les ensembles de formules

Définition

Un ensemble de formules Γ est consistant (ou satisfaisable) s'il admet au moins un modèle (*i.e.* , si $mod(\Gamma) \neq \emptyset$).

Définition

Un ensemble de formules Γ est contradictoire (ou insatisfaisable) s'il n'admet aucun modèle (*i.e.* , si $mod(\Gamma) = \emptyset$), on note alors $\Gamma \models \perp$.

Exemple

L'ensemble $\{p \vee q, \neg p \vee r\}$ est satisfaisable mais l'ensemble $\{p, \neg p\}$ est contradictoire.

Exercice

Si Γ est vu comme un ensemble de contraintes (par exemple celles d'affectation de salles de cours à des groupes d'étudiants), trouver une façon simple d'expliquer ces notions à un enfant.

Conséquences logiques d'un ensemble de formules

Définition

Une formule φ est conséquence logique d'un ensemble de formules Γ, noté $\Gamma \models \varphi$, lorsque toute valuation modèle de Γ est aussi modèle de φ (*i.e.* , si $mod(\Gamma) \subseteq mod(\varphi)$).
On note $cons(\Gamma)$ l'ensemble des conséquences logiques de Γ.

Exercice

Choisissez un Γ et réfléchissez à ce qu'est $cons(\Gamma)$, pensez aux tautologies !

Remarque

Attention à la confusion dans les notations :

- $\nu \models \varphi$ où ν est une valuation,
- $\psi \models \varphi$ où ψ est une formule,
- $\Gamma \models \varphi$ où Γ est une ensemble de formules.

Conséquences logiques d'un ensemble de formules

Remarque

Parmis les conséquences logiques φ d'un ensemble Γ, il y a les nouvelles contraintes déduites directement de Γ. Puisqu'elles découlent de Γ, elles ne peuvent pas apporter de contraintes supplémentaires que celles déjà apportées par Γ.

Proposition

Si $\Gamma \models \varphi$, les modèles de Γ sont exactement les mêmes que ceux de $\Gamma \cup \{\varphi\}$.

Exercice

Montrez-le, en établissant $mod(\Gamma) = mod(\Gamma \cup \{\varphi\})$.

Exemple

De $\Gamma = \{(p \rightarrow s) \vee q, \neg q\}$, on peut en déduire $p \rightarrow s$, *i.e.* , $p \rightarrow s \in cons(\{(p \rightarrow s) \vee q, \neg q\})$, ce qui s'écrit encore $\{(p \rightarrow s) \vee q, \neg q\} \models p \rightarrow s$.

Conséquences logiques : Propriétes

Proposition

$\Gamma \models \varphi$ ssi $\Gamma \cup \{\neg\varphi\}$ est contradictoire.

Preuve Au tableau.

Proposition

Pour tous ensembles de formules Σ et Γ,

$$mod(\Sigma \cup \Gamma) = mod(\Sigma) \cap mod(\Gamma)$$

En particulier, si $\Sigma \subseteq \Gamma$ alors $mod(\Gamma) \subseteq mod(\Sigma)$.

Preuve Au tableau.

Autres résultats

Exercice

Proposition

Si $\Gamma' \subseteq \Gamma$ et $\Gamma' \models \varphi$ alors $\Gamma \models \varphi$.

Montrez-le.

Proposition

$\{\varphi_1, ...\varphi_n\} \models \varphi$ ssi $\models (\varphi_1 \wedge \ldots \wedge \varphi_n) \rightarrow \varphi$.

Montrez-le.

Décidabilité du calcul propositionnel

On rappelle qu'une logique est décidable s'il existe un algorithme (calcul réalisable sur un ordinateur qui termine toujours pour toute donnée) qui permet de savoir pour chaque formule si elle est ou non valide (*i.e.* si $\models \varphi$).

Théorème

Le calcul propositionnel est décidable.

Preuve On fonde l'agorithme simplement sur le calcul des tables de vérité.

Algorithm 1 TAUTOLOGIE(φ)

for all $\nu \in Val$ **do**
  if $\nu(\varphi) = 0$ **then**
    return "φ n'est pas une tautologie"; BREAK
  end if
end for
return "φ est une tautologie"

Complexité de TAUTOLOGIE(φ) : on exécute la boucle $O(2^n)$ fois, où n est la taille de φ et le calcul de $\nu(\varphi)$ est en $O(|\varphi|)$

Remarques

- Sur la complexité en $O(2^n)$: si $n = 80$, il faut remplir 2^{80} ligne pour calculer la table de vérité complète.
  C'est long comment s'il nous faut une microseconde par ligne ?
  Converti en années, 2^{80} microsecondes est de l'ordre de 38 milliards d'années.
  En comparaison, l'âge de l'univers est d'environ 15 billion milliards d'années.

- Sur la validité :

Proposition

$\models \varphi$ ssi $\neg\varphi$ n'est pas satisfaisable.

Preuve $\models \varphi$ ssi $mod(\varphi) = Val$ ssi $Val \setminus mod(\varphi) = \emptyset$ ssi $mod(\neg\varphi) = \emptyset$ ssi $\neg\varphi$ n'est pas satisfaisable.

- Sur la satisfaisabilité : c'est pour ce problème qu'on dispose le plus d'outils informatiques, cherchez "SAT solvers" sur le net, voir par exemple `http://www.satcompetition.org/`.

La satisfaisabilité des formules du calcul propositionnel

Définition

Le problème SAT est un problème de décision (réponse oui/non) qui consiste à déterminer si une formule $\varphi \in \mathcal{F}_{cp}$ admet ou non un modèle.

On vient de voir que φ est satisfaisable ssi $\not\models \neg\varphi$.

Corollaire

Le problème SAT est décidable.

Preuve On a un algorithme pour SAT en coût exponentiel en invoquant $\textsc{Tautologie}(\neg\varphi)$.

Algorithme pour la satisfaisabilité des formules du calcul propositionnel

Algorithm 2 TAUTOLOGIE(φ)

for all $\nu \in$ *Val* **do**
  if $\nu(\varphi) = 0$ **then**
    return "non (φ n'est pas une tautologie, et la valuation ν la falsifie)" ; BREAK
  end if
end for
return "oui (φ est une tautologie)"

L'algorithme pour SAT répond "non" si TAUTOLOGIE($\neg\varphi$) retourne "oui (φ est une tautologie)", et "oui, avec le modèle ν" si TAUTOLOGIE($\neg\varphi$) répond "non (φ n'est pas une tautologie, et la valuation ν la falsifie)"

Remark

Il y a de meilleurs algorithmes pour résoudre SAT, *mais ils ont tous un coût exponentiel dans le pire cas (si la formule n'est pas satisfaisable).*

Formes normales

La plupart des algorithmes pour SAT débutent par une première phase de normalisation de la formule, c'est à dire qu'on modifie la syntaxe de la formule de manière à la mettre sous une forme normalisée, tout en conservant sa sémantique (c'est-à-dire l'ensemble de ses modèles).

Notations

Si $\varphi_1, \varphi_2, \ldots, \varphi_n \in \mathcal{F}_{cp}$, on utilise les notations :

$$\bigwedge_{i=1}^{n} \varphi_i = \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \text{ et } \bigvee_{i=1}^{n} \varphi_i = \varphi_1 \vee \varphi_2 \vee \ldots \vee \varphi_n$$

Si $n = 0$ on convient que :

$$\bigwedge_{i=1}^{0} \varphi_i = \top \text{ et } \bigvee_{i=1}^{0} \varphi_i = \perp \text{ (la clause vide)}$$

Littéraux, Clause, Forme clausale

Définition (Littéraux et Clauses)

Un littéral est une formule atomique ou la négation d'une formule atomique. Autrement dit, c'est une formule ℓ de la forme p ou $\neg p$, où $p \in \mathrm{Prop}$.

Une clause (disjonctive) est une disjonction de littéraux : $\bigvee_{i=1}^{n} \ell_i$ où les ℓ_is sont des littéraux. On notera $\perp$ pour la clause vide (sans aucun littéraux), car elle correspond à la formule 0 toujours fausse.

Définition (Forme clausale)

Une formule sous forme clausale (ou conjonctive ou sous forme normale conjonctive (FNC)) est une conjonction non vides de clauses (non vides):

$$\bigwedge_{j=1}^{m} C_j, \text{ où les } C_j\text{s sont des clauses } \bigvee_{i=1}^{n_j} \ell_i^j$$

Si l'ensemble de clausesest vide

Exemple

La formule $(\neg p \vee q \vee r) \wedge (\neg q \vee p) \wedge s$ est sous forme clausale, avec 3 clauses.

"Toute formule peut être mise sous forme clausale"

Proposition

Pour toute $\varphi \in \mathcal{F}_{cp}$, il existe $\tilde{\varphi} \in \mathcal{F}_{cp}$ sous forme clausale telle que $\varphi \equiv \tilde{\varphi}$.

Preuve On propose l'algorithme suivant :

- Appliquer tant que possible la substitution suivante :

$$\varphi \rightarrow \psi \hookleftarrow \neg\varphi \vee \psi \tag{2}$$

- Appliquer tant que possible les substitutions suivantes :

$$\begin{array}{ll} \neg\neg\varphi \hookleftarrow \varphi & \\ \neg(\varphi \wedge \psi) \hookleftarrow (\neg\varphi \vee \neg\psi) & \neg(\varphi \vee \psi) \hookleftarrow (\neg\varphi \wedge \neg\psi) \end{array} \tag{3}$$

- Appliquer tant que possible les substitutions suivantes

$$\varphi \vee (\psi_1 \wedge \psi_2) \hookleftarrow (\varphi \vee \psi_1) \wedge (\varphi \vee \psi_2) \tag{4}$$

Exercice

Écrire proprement l'algorithme. Pourquoi termine-t-il ? Pourquoi est-il correct ?

Mise sous forme clausale

Exemple

Soit $\varphi = (\neg(p \wedge (\neg q \vee (r \vee s)))) \wedge (p \vee q)$.

- Étape 1 : Il n'y a rien à faire
- Étape 2 :
  1. $\varphi \rightsquigarrow (\neg p \vee \neg(\neg q \vee (r \vee s))) \wedge (p \vee q)$
  2. $\rightsquigarrow (\neg p \vee (\neg\neg q \wedge \neg(r \vee s))) \wedge (p \vee q)$
  3. $\rightsquigarrow (\neg p \vee (q \wedge \neg r \wedge \neg s)) \wedge (p \vee q)$
  4. $\rightsquigarrow (\neg p \vee q) \wedge (\neg p \vee \neg r) \wedge (\neg p \vee \neg s) \wedge (p \vee q)$

  sous forme clausale, avec 4 clauses.

On en déduit que

$$\varphi \equiv (\neg p \vee q) \wedge (\neg p \vee \neg r) \wedge (\neg p \vee \neg s) \wedge (p \vee q)$$

Remark

On pourra écrire simplement $\{\{\neg p, q\}, \{\neg p, \neg r\}, \{\neg p, \neg s\}, \{p, q\}\}$

Remarques sur la mise sous forme clausale

Exercice

Trouver deux formules distincts en forme clausales qui sont équivalentes (leur équivalence ne venant pas uniquement que de la commutativité des connecteurs).

Remarque

On peut améliorer l'algorithme de mise sous forme clausale en appliquant les conseils suivants :

- Les clauses comportant deux littéraux opposés sont valides (tiers-exclu) et peuvent donc être supprimées (par ex. $p \vee q \vee \neg r \vee \neg q$).
- On peut aussi supprimer les répétitions d'un littéral au sein d'une même clause (par ex. $\neg p \vee q \vee \neg r \vee \neg p$ équivaut à $\neg p \vee q \vee \neg r$).
- Si dans une formule clausale une clause C_i est "incluse" dans une clause C_j alors la clause C_j peut être supprimée (la valeur de la conjonction des deux clauses ne dépend que de la valeur de C_i). On dit que C_i *subsume* C_j.

Exemple

$\{\{p, q, r\}\,\{p, \neg s, t, q, r\}\}$ est équivalent à $\{\{p, q, r\}\}$.

Systèmes formels

Définition

Un système formel S définit des règles de calcul entre formules qui permettent de déduire des conséquences logiques. Il est défini par :

- le langage (les propositions), et les formules sur ce langage,
- les axiomes : formules supposées vraies
- les règles d'inférence permettant de déduire une nouvelle formule (conclusion) à partir d'un ensemble de formules (prémisses ou hypothèses)

On note $\vdash_S \varphi$ si φ peut se déduire dans le système formel considéré, *i.e.* , si il est possible de l'obtenir à partir des axiomes, en appliquant les règles du système.

Soit Γ un ensemble de formules, on note $\Gamma \vdash_S \varphi$ si φ peut se déduire dans le système formel considéré, à partir des axiomes et des éléments de Γ.

Un système formel pour le calcul des proposition : la résolution

Introduite en 1965 par Robinson, la résolution est un système formel qui utilise des formules sous forme clausale. Elle n'a pas d'axiomes, et comporte deux règles d'inférence.

Definition

Le système de résolution

- La règle de coupure

$$\frac{C \vee \ell \quad C' \vee \neg\ell}{C \vee C'} \text{ coupure}$$

- La règle de factorisation.

$$\frac{C \vee \ell \vee \ell}{C \vee \ell} \text{ factorisation}$$

Exemple de résolution

$$\frac{C \vee \ell \quad C' \vee \neg\ell}{C \vee C'} \text{ coupure} \qquad \frac{\ell \quad \neg\ell}{\bot} \text{ coupure (cas particulier)}$$

$$\frac{C \vee \ell \vee \ell}{C \vee \ell} \text{ factorisation}$$

Exemple

Soit

$$\mathcal{C} = \{(p \vee r \vee s), (r \vee \neg s), \neg r, \neg p\}$$

Voici une dérivation par résolution de cet ensemble de clauses :

$$\cfrac{\cfrac{\cfrac{\cfrac{p \vee r \vee s \quad r \vee \neg s}{p \vee r \vee r}\text{ coupure}}{p \vee r}\text{ factorisation} \quad \neg r}{p}\text{ coupure} \quad \neg p}{\bot}\text{ coupure}$$

Utilisation du système de résolution

La résolution est réfutationnelle : pour prouver φ, on suppose $\neg\varphi$ et on montre que cela conduit à une absurdité.
On procède en trois étapes :

1. Mettre $\neg\varphi$ sous forme clausale :
$$\neg\varphi = C_1 \wedge \ldots \wedge C_n$$
2. Remplacer la conjonction de clauses $C_1 \wedge \ldots \wedge C_n$ par l'ensemble $\mathcal{C} = \{C_1, \ldots, C_n\}$
3. Saturer l'ensemble en rajoutant à $\mathcal{C}$ les clauses qu'on peut déduire à partir des deux règles de résolution.

On s'arrête

- soit quand l'application des règles ne modifie plus l'ensemble de clauses
- soit quand on vient d'ajouter la clause vide $\perp$ et donc la formule initiale est insatisfaisable.

Preuve par réfutation

Soit $\varphi\psi = (p \vee r \vee s) \wedge (r \vee \neg s) \wedge \neg r \wedge \neg p$, déjà sous forme clausale.

Remarque

Puisqu'on arrive à dériver la clause vide, la formule ψ est insatisfiable.
Donc φ est une tautologie.

Propriétés d'un système formel

Définition

Si une formule φ se dérive en une formule ψ par application des règles de résolution (coupure ou factorisation), on note $\varphi \vdash_R \psi$.
Et on note $\vdash_R \psi$ lorsque $\neg\psi \vdash_R \bot$.

Deux questions fondamentales sont systématiquement posées pour relier les calculs dans un système formel S et la sémantique dans une logique : existe-il un système formel tel que :

1. Correction, *i.e.* si $\vdash_S \varphi$, alors $\models \varphi$ (φ est valide)
2. Complétude, *i.e.* si $\models \varphi$, alors $\vdash_S \varphi$?

Il faut impérativement la Correction sinon le système permet de déduire n'importe quoi.

Théorème (Correction et complétude de la résolution)

- La résolution est correcte, *i.e.* , si $\vdash_R \psi$, alors $\models \psi$.
- La résolution est complète, *i.e.* , si $\models \psi$ alors $\vdash_R \psi$.