Logique

L3 RI

Table des matières

1 Induction

Définition inductive : ensemble de base, des fonctions constructrices d'autres éléments. L'ensemble est alors défini comme le plus petit contenant la base et stable par les constructeurs.

Tout élément s'obtient alors depuis des éléments de base et un nombre fini de constructeurs. Représente sous forme d'arbre → hauteur d'un élément.

Pour prouver quelque chose, le faire sur la base et montrer la stabilité par les constructeurs.

Définition non-ambigüe : il existe un unique arbre d'induction pour chaque élément. Condition nécessaire et suffisante pour l'existence de fonctions définies inductivement.

2 Calcul propositionnel

2.1 Syntaxe et sémantique

Propositions définies par induction : variables et connecteurs. Non ambigu.

Valuation : fonction qui assigne à chaque variable 0 ou 1. Les valuations sont prolongeables sur l'ensemble des formules (en uitilisant la sémantique des connecteurs). Formule F satisfaite par la valuation φ : $\varphi(F) = 1$.

F et G formules sont équivalentes *ssi* $\forall \varphi, \varphi F) = \varphi(G)$. On note $F \equiv G$.

Soit Σ un ensemble de formules, F une formule. F conséquence de Σ ssi toute valuation qui satisfait Σ satisfait F, noté $\Sigma \models F$ (aussi équivalent à $\Sigma \cup F$ non satisfiable. Σ satisfait par une valuation si celle-ci satisfait toutes ses formules.

Substitution d'une variable p par une formule G dans une formule G noté $F[G/p]$. Opération syntaxique.

Support d'une formule : ensemble des variables qu'elle utilise.

2.2 Formes normales

Pour chaque table de vérité, on peut trouver une formule dont c'est la table.

Toute formule est équivalente à une formule sous forme normale disjonctive (FND) : union d'intersections de littéraux (littéral = variable ou négation d'une variable). De même, toute formule équivalente à une forme normale conjonctive (FNC).

Système complet de connecteur : ensemble de connecteurs qui permettent d'engendrer toutes les formules (sémantiquement). Ex : $\{nand\}$.

Théorème de compacité Ensemble Σ de formules. Σ satisfiable ssi Σ finiment satisfiable. Et Σ est finiment satisfiable si tout sous-ensemble fini de Σ est satisfiable.

Corollaire : $\Sigma \models F$ ssi il existe un sous-ensemble fini de Σ dont F est conséquence.

2.3 Systèmes de déduction

Système formel : alphabet, procédé de formation des formules, ensemble d'axiomes et ensemble de règles de déduction. Si on déduit F de Σ avec ces règles (syntaxiques), on note $\Sigma \vdash F$.

Un système de déduction est correct si $\Sigma \vdash F$ (syntaxique) implique $\Sigma \models F$ (sémantique). Un système de déduction est complet si $\Sigma \models F$ (sémantique) implique $\Sigma \vdash F$ (syntaxique).

2.4 Système de déduction naturelle

2.4.1 Logique minimale (NM)

$\frac{}{\Gamma, A \vdash A}\text{Ax}$

$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \implies B}$ intro $\implies$ $\frac{\Gamma \vdash A, \Delta \vdash A \implies B}{\Gamma, \Delta \vdash B}$ elim $\implies$

À compléter, j'ai la flemme là...

2.4.2 Logique intuitionniste (NJ)

On ajoute le symbole $\bot$, on ajoute une règle :

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash A}\text{elim } \bot$$

On enlève le connecteur $\neg$ et on en fait une macro : $\neg A = A \implies \bot$.

à compléter : elim intro non

2.4.3 Logique classique (NK)

On ajoute le Tiers exclus :

$$\frac{}{\Gamma \vdash A \vee \neg A}\text{Tiers exclus}$$

NK plus forte que NJ plus forte que NM. On est plus fort quand on permet démontrer au moins les mêmes choses.

2.5 Traduction de logiques

Soit $\mathcal{L}$ plus forte que $\mathcal{L}'$. Une traduction est une fonction φ qui transforme les formules : si $\vdash F$ dans $\mathcal{L}$ alors $\vdash \varphi(F)$ dans $\mathcal{L}'$.

HELP j'ai l'impression que c'est l'inverse. Je suis peut etre juste trop fatigué

2.6 Déduction par coupure

On s'intéresse aux clauses : disjonction de littéraux. Toute formule peut être mise sous forme de conjonction de clauses (FNC).

On met les clauses sous la forme $(a_1 \wedge \cdots \wedge a_n) \implies (b_1 \vee \cdots \vee b_m)$ où les a_i sont les variables négatives et les b_i les positives. On note aussi une clause $C = (\Gamma, \Delta)$ où $a_i \in \Gamma$ et $b_i \in \Delta$.

Règle de coupure :

$$\frac{(\Gamma_1, \Delta_1 \cup \{p\})\ (\Gamma_2 \cup \{p\}, \Delta_2)}{(\Gamma_1 \cup \Gamma_2, \Delta_1 \cup \Delta 2)}$$

Déduction par coupure correcte et complète.

Idée : On fait toutes les coupures sur une variable (on obtient le *résolvant*). Si un résolvant est nul, l'ensemble de clauses était non satisfiable.

Théorème : Tout ensemble de clause non satisfiable possède une réfutation par coupure.

2.7 Calcul des séquents

Un séquent : 2 suites finies de formules, noté $(H_1, \ldots, H_n) \vdash (C_1, \ldots, C_m)$ qui signifie $(H_1 \wedge \cdots \wedge H_n) \implies (C_1 \vee \cdots \vee C_m)$

2.7.1 Calcul des séquents classique (LK)

AJOUTER LES REGLES

Théorème fondamental (Hauptsatz) Pour toue preuve dans LK, il existe une preuve de même conclusion sans utiliser la règle de coupure.

Cohérence Un système de déduction est cohérent s'il ne permet pas de prouver A et $\neg A$ sans hypothèses. Cohérent ssi on ne peut pas prouver le séquent vide.

LK cohérent (grâce au Hauptsatz).

2.7.2 Calcul des séquents intuitionniste (LJ)

Restriction des séquents : une seule formule au plus à droite de $\vdash$.

Introductiçon du ou : $\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}$ $\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$.

3 Calcul des prédicats

1^{er} ordre : quantifier sur les objets (variables, constantes, fonctions appliquées). Second ordre : quantifier sur les relations.

Langage du 1^{er} ordre : variables, symboles logiques et quantificateurs ($\forall$ et $\exists$), constantes, fonctions, relations (d'arité définie).

Termes : variables, constantes et applications de fonctions. Non-ambigü : on connait l'arité des fonctions. Ex : $f(x_1)$.

Variables libres d'un terme : variables qui ne sont pas dans la portée d'un quantificateur. Ex : x_1 dans $f(x_1)$.

Formule atomique : Application d'un prédicat à des termes. Ex : $Rt_1 \ldots t_n$ où t_i terme.

Formule du premier ordre : par induction à partir des formules atomiques et stable par les connecteurs et les quantificateurs.

Clotûre universelle : $\forall x_1 \ldots \forall x_n F$ où les x_i sont les variables libres de F.

Structure $\mathcal{M}$ pour le langage L :

— Mensemble de base non vide

— un élément $c^{\mathcal{M}} \in M$ pour chaque symbole de constante (valuation des constantes).

— $f^{\mathcal{M}} : M^{\text{k}} \to M$ pour f d'arité k.

— sous ensemble $R^{\mathcal{M}}$ de M^{k} pour R d'arité k.

Formule universellement valide : valide dans toute sturcture. noté $\models {}^{*}F$.

Formule contradictoire : il n'existe pas de structure où la formule est valide.

F et G équivalents si $F \Leftrightarrow G$ uniersellement valide.

Toute formule est équivalente à une autre n'utilisant que $\neg, \vee$ et $\exists$ (système minimal.

F prénexe quand tout les quantificateurs sont devant. F prénexe polie quand il y a au plus une occurence de chaque variable dans le préfixe. Toute formule admet au moins une forme prénexe polie.

Formes de Skolem On passe sous forme prénexe polie. On enlève les $\exists$. Les variables quantifiées par ces $\exists$ deviennent l'image des variables quantifiées précédemment par des $\forall$ par une nouvelle fonction. Exemple : $\forall x \exists y Rxy$ devient $\forall x Rxfx$. y est devenu fx.

Une formule et sa forme de Skolem ne sont pas universellement équivalentes, mais il existe une structure qui valide la formule ssi il existe une structure qui valide sa forme de Skolem.

Une formule close admet un modèle ssi une de ses formes de Skolem en admet.

3.1 Théories

Théorie : ensemble de formules closes. Elle est consistante si elle admet au moins un modèle. Contradictoire sinon. Si toute partie finie de T admet un modèle, T admet un modèle.

Structure $\mathcal{M}$. $\mathcal{M} \models T$ ssi $\mathcal{M} \models F \forall F \in T$.

Formule F conséquence de T ssi $\forall \mathcal{M}, \mathcal{M} \models T$ implique $\mathcal{M} \models F$. On note $T \models {}^{*}F$. Équivalent à $T \cup \{\neg F\}$ contradictoire.

Inversement, si $T \cup \{\neg F\}$ contradictoire alors que F est universellement valide, T contradictoire.

Deux théories sont équivalentes si elles admettent les mêmes modèles.

L'ensemble des formules F telles que $T \vdash F$ est l'ensemble des théorèmes de T, Thm(T). T récursif si l'ensemble des formules de T est récursif : il existe un algorithme qui décide de l'appartenance à la théorie. T est décidablme si Thm(T) est récursif.

Contenir le prédicat binaire $=$: langage et théories égalitaires.

Théorème de Church Si L contient $=$ et un autre prédicat binaire, $\emptyset$ est indécidable.

3.2 Systèmes de déduction naturelle

(NK) et les règles suivantes :

AJOUTER LES REGLES

Théorie T cohérente : il n'existe pas de F telle que $T \vdash F$ et $T \vdash \neg F$. Si T a toutes ses parties finies cohérentes, T est cohérente.

Déduction complète et correcte. Donc T cohérente ssi T consistante.

3.3 Calcul des séquents au premier ordre

AJOUTER LES RÈGLES
Le Hauptsatz est valide. Le calcul des séquents est complet.

3.4 Arithmétique de Peano

$L = \{0, S, +, \times, =, <\}$
Axiomes de l'égalité, et :
ÉCRIRE TOUS LES AXIOMES DE PEANO
Il y a une infinité dénombrable de tels axiomes. Il existe une infinité de modèles de P non isomorphes à $\mathbb{N}$.

P(propriété) est axiomatisable s'il existe une théorie T telle que pour toute structure $\mathcal{M}$ sur le langage, $\mathcal{M}$ vérifié P ssi $\mathcal{M} \models T$. Finiment axiomatisable : T est finie.

"être un ensemble à au moins n éléments" finiment axiomatisable. "être un ensemble à exactement n éléments" finiment axiomatisable. Mais "être un ensemble fini" ou "être un ensemble infini" ne sont pas finiment axiomatisables.

Théorème (Ryll-Nardzewski) : Aucune extension consistante de Peano n'est finiment axiomatisable.

3.5 Décidabilité et arithmétique

L'arithmétique de Peano est indécidable.

Théorème d'incomplétude de Gödel Une théorie récursive et consistante contenant $\mathcal{P}_0$ n'est pas syntaxiquement complète (on peut construire un énoncé qui ne peut être ni prouvé ni réfuté syntaxiquement).

Une théorie syntaxiquement complète et récursive est décidable. Si T consistante et contient $\mathcal{P}$, T indécidable.

Deuxième théorème d'incomplétude de Gödel La cohérence de T, qui peut s'écrire comme une formule, ne peut pas être déduite de T.

3.6 Résolution

Extension de la preuve par coupure à la logique du premier ordre.
On met sous forme de clauses :
— Mise sous forme prénexe normale conjonctive
— Mises sous forme de Skolem
— Distribution des $\forall$ sur les $\wedge$
— Décomposition des conjonctions en ensembles de clauses quantifiées universellement

Puis on fait des unifications entre clauses : Choisir une substitution (remplacer variables par des termes).

Un ensemble dini de formules atomiques est unifiable s'il existe une substitution qui les rend égales. Pour un ensemble de couples de termes, on appelle cette substitution un unificateur. C'est un unificateur principal si tout autre unificateur s'obtient en composant avec une autre substitution. Deux unificateurs principaux sont égaux à permutation près des variables.

Pour trouver un unificateur principal, on effectue toutes les simplifications possibles (lorsqu'un couple commence par la même fonction), et on fait disparaître les variables une à une en créant l'unificateur au fur et à mesure.

Pour deux clauses $C_1 = (\Gamma_1, \Delta_1)$ et $C_2 = (\Gamma_2, \Delta_2)$ sans variables communes, C est un résolvant si il existe $P_1 \subseteq \Delta_1$ et $N_2 \subseteq \Gamma_2$ tels que $P_1 \cup N_2$ unifiable, avec σ unificateur principal et $C = (\sigma(\Gamma_1 \cup \Gamma_2 \backslash N_2), \sigma(\Delta_1 \backslash P_1 \cup \Delta_2))$.

S'il existe une réfutation par résolution d'un ensemble de clauses, alors cet ensemble n'a pas de modèle. Résolution complète et correcte.

Programmation logique On utilise la résolution. Une ligne de programme : 1 clause.

Clauses de Horn : au plus 1 littéral positif.
Exemples :
homme(Socrate) : Clause positive
?mort(Socrate) : Clause négative
empoisonné(X) :- boit(X,Y), poison(Y)

But : clause négative. 2 types de coupures possibles. Un but avec une clause de Horn pour un but éventuellement plus long, ou un but avec une clause positive pour un but plus court.

SLD (selective linear definite) resolution. Complet. On parcourt l'arbre de tous les résolvants possibles en profondeur