La méthode de résolution pour le calcul des prédicats

HLIN602 Logique II Christian Retoré

confinement 16 mars 2020 $\longrightarrow$ ...

1. Complétude

Un objectif majeur du cours est de mettre en rapport les deux notions centrales vues dans la première partie :

— les preuves (calcul des séquents ou résolution)
— les modèles

Théorème de complétude (Gödel, 1929) : $X_1, \ldots, X_n \vdash F$ **est démontrable (par exemple dans le calcul des séquents, ou par résolution) si et seulement si toute interprétation qui rend** $X_1, \ldots, X_n$ **vrais rend** F **vraie.**

Une manière de montrer cela est :

Model Existence Lemma : Si un ensemble de formules est cohérent (consistent, c.-à-d.s'il ne démontre pas $\perp$) alors il admet un modèle.

2. Insatifiabilité et prouvabilité

Une autre manière de voir ce lemme *model existence lemma*, est de regarder sa contraposée, qui s'exprime en termes d'insatisfiabilité et d'incohérence.

Si un ensemble de formules n'admet pas de modèle (insatisfiable) alors cet ensemble de formules entraîne $\perp$ (dans le calcul des séquents, ou par résolution).

La contraposée du *model existence lemma* est plus facile à établir avec la méthode de *résolution*, car la résolution est précisément un mode de démonstration basé sur la réfutation qui permet de montrer qu'un ensemble de formule entraîne$\perp$.

3. Résolution

La résolution est un mécanisme de démonstration qui n'utilise que la règle de coupure du calcul des séquents et les règles de quantification universelles, et qui lui est équivalent, alors qu'elle procède très différemment du calcul des séquents.

La résolution est le mécanisme d'évaluation du langage PROLOG, langage phare de l'Intelligence Artificielle, né avec l'IA, et qui revient à le mode.

Cela fait au moins deux bonnes raisons d'étudier la méthode de résolution.

Plus tard dans le cours nous verrons le *model existence lemma* avec le calcul des séquents et la complétude de la méthode de résolution ce qui aura pour conséquence :

> F **est réfutable par résolution**
> si et seulement si
> $\neg F$ **est démontrable dans le calcul des séquents**
> si et seulement si
> F **est fausse dans toute interprétation**

4. Résolution

Pour présenter la méthode de résolution (une autre technique de preuve que le calcul des séquents, basée sur la réfutation), nous allons (re)voir :

— mise sous forme prénexe
— Skolemisation
— mise sous forme clausale
— unification
— résolution proprement dite

On définira la résolution, on en donnera des exemples,

mais nous montrerons la complétude de la méthode de résolution dans un autre chapitre, car il faut des rappels sur le calcul propositionnel.

5. Distributivité de $\forall$ sur &

Pour toutes formules F et G et pour toute variable x, on a

$$(1): \quad ((\forall x F) \wedge (\forall x G)) \equiv (\forall x(F \wedge G)) \quad :(2)$$

Cela se comprend intuitivement.

Si on a (1), on a $(1a): \quad (\forall x F)$ et $(1b): \quad (\forall x G)$. Pour n'importe quelle valeur de x, on à la fois G et F, et donc $F \wedge G$. Comme cela est vrai pour n'importe quelles valeur de x on a $(\forall x(F \wedge G))$.

Réciproquement, supposons (2). On a $F \wedge G$ qui est vraie pour toute valeur de x. Et donc pour toute valeur de x F est vraie, soit $(1a): \quad (\forall x F)$, et de même G est vraie pour toute valeur de x soit $(1b): \quad (\forall x G)$. On a donc $(1a) \wedge (1b)$ soit (1).

6. Attention

En général on n'a pas

$$(1*) : \quad ((\forall x F) \vee (\forall x G)) \not\equiv (\forall x (F \vee G)) \quad : (2*)$$

Il suffit pour cela de trouver deux formules F et G et une interprétation qui rende l'une vraie et l'autre fausse. Considérons la formule $F = P(x)$ et $G = I(x)$. Prenons pour domaine les entiers et interprétons $P(x)$ par *x est pair* et $I(x)$ par *x est impair*. On a bien (2$*$) (tout entier est pair ou impair) mais on n'a pas (1$*$) il est faux que tout entier soit pair, et il est égalent faux que tout entier soit pair.

Exercice : montrer que si on a (1$*$), on a toujours (2$*$).

7. Distributivité de $\exists$ sur $\vee$

Pour toutes formules F et G et pour toute variable x on a

$$(1') : \quad ((\exists x F) \vee (\exists x G)) \equiv (\exists x (F \vee G)) \quad : (2')$$

Cela se comprend pareillement.

Si $(1')$ est vraie, un au moins des deux énoncés $(1'a)$: $(\exists x F)$ et $(1'b)$: $(\exists x F)$. Si $(1'a)$ est vraie, soit a une valeur de x telle que F soit vraie ; alors $F \vee G$ est vraie pour cette même valeur a de x, et par conséquent $(2')$ est vraie. Si $(1'b)$ est vraie, soit b une valeur de x telle que G soit vraie ; alors $F \vee G$ est vraie pour cette même valeur b de x, et par conséquent $(2')$ est vraie. Dans les deux cas, $(2')$ est vraie.

Si $(2')$ est vraie, alors il existe une valeur c de x telle que $(F \vee G)$ soit vraie. pour cette valeur c de x l'une au moins des deux formules F ou G est vraie. Si F est vraie pour cette valeur de x, alors $(\exists x F)$ est vraie, et par suite $(\exists x F) \vee (\exists x G)$: (1) est vraie. Si G est vraie pour cette valeur de x, alors $(\exists x G)$ est vraie, et par suite $(\exists x F) \vee (\exists x G)$: (1) est vraie.

8. Attention

En général on n'a pas

$$(1'*) : \quad ((\exists x F) \wedge (\exists x G)) \not\equiv (\exists x(F \wedge G)) \quad : (2'*)$$

Il suffit pour cela de trouver deux formules F et G et une interprétation qui rende l'une vraie et l'autre fausse. Considérons la formule $F = P(x)$ et $G = I(x)$. Prenons pour domaine les entiers et interprétons $P(x)$ par *x est pair* et $I(x)$ par *x est impair*. On a bien $(1'*)$ (il existe un entier pair et il existe un entier impair) mais on n'a pas $(2'*)$ car il n'existe pas d'entier qui soit pair et aussi impair.

Exercice : montrer que si on a $(2'*)$, on a toujours $(1'*)$.

9. Formes prénexes : commutations

$$\begin{array}{rcl} \neg(\forall x F) & \equiv & (\exists x(\neg F)) \\ \neg(\exists x F) & \equiv & (\forall x(\neg F)) \end{array}$$

cela se comprend : dire qu'on n'a pas pour tout x la propriété F c'est dire qu'il existe un x qui satisfait $(\neg F)$. De même dire qu'il n'existe pas de x satisfaisant F, c'est dire que tout x satisfait $(\neg F)$.

10. Formes prénexes : "remontée" des quantificateurs

Soit x une variable qui n'apparaisse pas dans G (qui ne soit pas libre dans G suffit, mais on peut toujours s'y ramener). Alors on a les équivalence suivantes :

$$
\begin{array}{rcl}
((\forall x F) \wedge G) & \equiv & (\forall x (F \wedge G)) \\
((\exists x F) \wedge G) & \equiv & (\exists x (F \wedge G)) \\
\hline
((\forall x F) \vee G) & \equiv & (\forall x (F \vee G)) \\
((\exists x F) \vee G) & \equiv & (\exists x (F \vee G)) \\
\hline\hline
((\forall x F) \Rightarrow G) & \equiv & (\exists x (F \Rightarrow G)) \\
((\exists x F) \Rightarrow G) & \equiv & (\forall x (F \Rightarrow G)) \\
(G \Rightarrow (\forall x F)) & \equiv & (\forall x (F \Rightarrow G)) \\
(G \Rightarrow (\exists x F)) & \equiv & (\exists x (F \Rightarrow G))
\end{array}
$$

11. Commentaires,... 1/2

Les équivalences pour l'implication s'obtiennent (exercice possible) à partir de celles sur la disjonction et la négation, puisque $(A \Rightarrow B) \equiv ((\neg A) \vee B)$. Le changement de quantificateur lorsque le quantificateur porte sur le premier argument, est bien sûr dû à la négation cachée dans l'implication.

Il faut bien prendre garde à la condition : *x est une variable qui n'apparaît pas dans G*. Soit par exemple la formule : $(\exists x((\exists x P(x)) \wedge I(x)))$ Comme $(\exists x P(x))$ est vraie la formule se ramène à $(\exists x(\top \wedge I(x)))$ c'est-à-dire à $(\exists x I(x))$ qui est vraie.

12. Commentaires,...2/2

Par contre, si on applique la transformation au $\exists$ interne, sans se soucier de la condition, on a $(\exists x P(x)) \wedge I(x) \equiv (\exists x(P(x) \wedge I(x)))$ et par remplacement d'une sous formule par une sous formule équivalente, la formule devient $(\exists x(\exists x(P(x) \wedge I(x))))$ et, comme il n'existe pas d'entier qui soit pair et impair, elle devient $(\exists x \bot)$ qui est bien sûr faux.

Pour pouvoir appliquer ce schéma il faut tout d'abord renommer les variables liées, ce qui fait apparaître la dépendance entre occurrences de variables et quantificateurs : $(\exists x((\exists y P(y)) \wedge I(x)))$. On obtient alors $(\exists x(\exists y(P(y) \wedge I(x))))$ qui est bien équivalent à la formule de départ. Il existe un entier x et un entier y tels que x soit pair et y soit impair est bien équivalent à : il existe un entier x tel que x soit pair et qu'il existe un entier pair (par forcément le même).

13. Mise sous forme prénexe, fin

Il n'est pas difficile de voir qu'en utilisant les équivalences de la gauche vers la droite (et en opérant au préalable les renommages nécessaires) toute formule est équivalente à une formule où tous les quantificateurs sont en tête.

Une formule est dite en forme prénexe si elle s'écrit :

$$(Q_1 x_1 (Q_2 x_2 (\ldots (Q_n x_n F) \ldots)))$$

où les Q_i sont des quantificateurs ($\exists$ ou $\forall$)

14. Exemple

$((\forall x(\exists y F(x,y))) \Rightarrow (\forall x(\exists y G(x,y))))$ devient successivement :

1. $((\forall x(\exists y F(x,y))) \Rightarrow (\forall z(\exists w G(z,w))))$ (renommage des variables liées)

2. $(\exists x((\exists y F(x,y)) \Rightarrow (\forall z(\exists w G(x,y)))))$ ($\exists x$ et $\Rightarrow$)

3. $(\exists x(\forall y(F(x,y) \Rightarrow (\forall z(\exists w G(x,y))))))$ ($\forall y$ et $\Rightarrow$)

4. $(\exists x(\forall y(\forall z(F(x,y) \Rightarrow (\exists w G(x,y))))))$ ($\forall z$ et $\Rightarrow$)

5. $(\exists x(\forall y(\forall z(\exists w(F(x,y) \Rightarrow G(x,y))))))$ ($\exists w$ et $\Rightarrow$)

Tous les quantificateurs sont en tête.

On aurait pu procéder dans un autre ordre et par exemple passer de 1 à 2' :

2'. $(\forall z((\forall x(\exists y F(x,y))) \Rightarrow (\exists w G(z,w))))$

mais le résultat obtenu aurait été équivalent.

15. Formes de Skolem

Étant donnée une formule sous forme prénexe (tous les quantificateurs sont en tête), on va enrichir le langage de manière à l'écrire plus simplement.

Supposons que la formule prénexe F soit satisfiable , c.-à-d.qu'il existe une interprétation qui la rende vraie.

Lorsqu'on a un quantificateur $\exists s$, la ou les valeurs du s qui rendent la formule vraie ne dépendent que des variables quantifiées universellement ($\forall$) qui précèdent $\exists s$.

Par exemple pour une formule $\exists x \forall y \exists z \forall t \exists u F$ le x ne dépend de rien, le y qui rend la formule vraie ne dépend que de y, et le u ne dépend que de y et t (et de z mais celui-ci est fonction de y).

16. Fonctions de Skolem

D'où l'idée, étant donnée une formule, de la transformer en une formule où on remplace chaque variable quantifiée existentiellement s par un nouveau symbole de fonction appliqué aux variables quantifiées universellement avant s , ou par une constante s'il n'y en a pas.

Prenons un exemple :

$$\exists x \quad \exists y \quad \forall z \quad \exists w \quad \forall u \quad \exists v((R(x,y) \Rightarrow ((\neg F(z,u,v)) \vee K(w))) \wedge G(v))$$

se transforme en :

$$((R(a,b) \Rightarrow ((\neg F(z,u,f(z,u))) \vee K(g(z)))) \wedge G(f(z,u)))$$

où a et b sont de nouvelles constantes, et f un nouveau symbole de fonction à deux arguments, et g un nouveau symbole de fonction à un argument.

17. Elimination des quantificateurs

Comme les variables sont toutes quantifiées universellement (puisqu'on a remplacé les quantifications existentielles) on omet souvent les quantificateurs, mais une telle formule doit être comprise comme sa clôture universelle c.-à-d. :

$$\forall z \quad \forall u \quad ((R(a,b) \Rightarrow ((\neg F(z,u,f(z,u))) \vee K(g(z)))) \wedge G(f(z,u)))$$

— on remarquera que l'ordre des quantifications universelles est sans importance, tout autre ordre conduisant à une formule équivalente.

18. Forme de Skolem et satisfiabilité 1/5

Soit G une formule sous forme prénexe et G' sa forme de Skolem ; G est satisfiable si et seulement si G' est satisfiable.

satisfiable : on dit aussi consistante ou non contradictoire : elle est vraie pour une interprétation

En particulier,
si F est une formule prénexe
et si F' est sa forme de Skolem,
alors F est satisfiable si et seulement si F' est satisfiable ,
ou, en d'autres termes,
F est insatisfiable (ou contradictoire)
si et seulement si
F' est insatisfiable (ou contradictoire).

19. Forme de Skolem et satisfiabilité 2/5

S'il existe un domaine D et une interprétation dans D qui rend G vrai, alors définissons une interprétation de même domaine pour le langage de G' sa forme de Skolem comme suit.

Le domaine est le même, et le langage commun reçoit la même interprétation. Comme pour toutes les valeurs des variables universelles, il existe des valeurs des variables existentielles situées après elles qui rendent la formule vraie, on peut définir une fonction qui associent à un n-uplets de valeurs des variables universelles la valeur qui rend la formule vraie.

En interprétant les nouveaux symboles fonctionnels ainsi, on obtient un domaine et une interprétation qui rend la formule de Skolem vraie.

20. Forme de Skolem et satisfiabilité 3/5

Réciproquement, si la formule de Skolem G' est vraie pour une interprétation alors la formule non transformée est également vraie pour cette même interprétation.

Le $\exists s$ sera précisément garanti par l'élément du domaine f appliqué aux variables universelles le précédant.

21. Forme de Skolem et satisfiabilité 4/5

Un exemple : soit la formule $\forall x \exists y P(x, y)$, elle est vraie avec pour domaine les entiers et pour interprétation de $P(x, y)$ la relation $x < y$. Sa transformée de Skolem est $P(x, f(x))$ qui se comprend comme $\forall x P(x, f(x))$.

Comme pour tout entier il existe un entier qui lui est supérieur, on peut définir une fonction qui réalise cette existence, par exemple $f(x) = x + 5$ (ou $f(x) = 2^x$ ou $f(x) = 2x$ si x est impair et x^2 si x et pair, ou.. $\ldots$). Il est clair qu'en gardant la même interprétation pour P et en donnant au symbole f l'une des interprétations possibles on obtient une interprétation dans laquelle la forme de Skolem est vraie.

22. Forme de Skolem et satisfiabilité 5/5

Reprenosn notre exemple : soit la formule $\forall x \exists y P(x, y)$, elle est vraie avec pour domaine les entiers et pour interprétation de $P(x, y)$ la relation $x < y$. Sa transformée de Skolem est $P(x, f(x))$ qui se comprend comme $\forall x P(x, f(x))$.

Réciproquement, supposons que nous ayons une interprétation qui rende $\forall x P(x, f(x))$ vraie. Cela veut dire qu'on a un domaine D, une relation binaire $P \subset D \times D$ sur ce domaine, et une fonction de $D \times D$ dans D qui rend la formule vraie. Par exemple si D est l'ensemble des entiers, P est $x < y$ et f est $f(x) = x + 1$, on a bien $\forall x \quad x < x + 1$. En gardant cette même interprétation la formule $\forall x \exists y P(x, y)$ est vraie. Pourquoi ? On sait que $f(x)$ est définie pour tout x de D, et $y = f(x)$ est une valeur pour laquelle $P(x, y)$ est vraie.

23. Formes clausales

Lorsqu'une formule est sous forme de Skolem, la partie sans quantificateur peut se mettre sous forme normale conjonctive, puisque les lois de de Morgan, et la distributivité entre $\wedge$ et $\vee$ s'appliquent.

On se retrouve alors avec une formule équivalente qui a la forme :

$$\forall x \forall y \ldots (C_1 \wedge \cdots \wedge C_N)$$

où chaque C_i est une disjonction de formules atomiques ou de négations de formules atomiques.

Comme les $\forall$ et $\wedge$ commutent, la formule est équivalente à une formule

$$(\forall x \forall y \ldots C_1) \wedge \cdots (\forall x \forall y \ldots C_n)$$

ce qui s'appelle une forme clausale.

24. Théorème de Herbrand

Supposons qu'on veuille montrer qu'une formule F est insatisfiable . D'après ce qui précède, cela revient à montrer que sa forme clausale est insatisfiable , et pour cela on dispose du théorème de Herbrand :

Une forme clausale est insatisfiable si et seulement s'il est possible de remplacer dans chaque clause les variables par des termes de Herbrand de sorte que l'ensemble des clauses (propositionnelles) obtenues soit inconsistant.

Quels sont les termes de Herbrand ? Ce sont tous les termes obtenus à partir des constantes et des fonctions — les termes sans variable. Par exemple si les fonctions sont f (d'arité 2) et g (d'arité 1) et h (d'arité 3) et les constantes a, b, c, les termes suivants sont des termes de Herbrand : $h(f(a, a), g(b), h(a, b, f(a, c)))$, $f(g(a), h(a, c, f(a, b)))$,...

25. Une méthode pour chosir : l'unification

Pour montrer l'inconsistance de ces ensembles de clauses propositionnelles, on peut bien sûr utiliser la méthode de résolution en engendrant librement les termes de Herbrand.

En fait la méthode de résolution (et Prolog) procède différemment : on part des clauses, et on les instancie pour obtenir une paire de clauses à laquelle appliquer la résolution, en utilisant pour cela l'unification des termes.

26. Unification

Unifier deux termes t_1 et t_2 consiste à trouver une substitution des variables qui les rendent égaux. Vous avez sans doute déjà vu un algorithme qui calcule l'unificateur le plus général ou affirme qu'il n'en existe pas. QUelques exemples devraient suffire à vous raffraîchir la mémoire :

— Soient $t_1 = f(x, g(y))$ et $t_2 = f(g(u), g(z))$.
La substitution $\sigma : x \mapsto g(u), z \mapsto y$ donne $\sigma t_1 = f(g(u), g(z)) = \sigma t_2$.

— Soient $t_1 = f(x, g(y))$ et $t_2 = f(g(u), h(z))$.
Il n'y a pas de substitution qui unifie ces deux termes, car cela imposerait $g = h$ et seules les variables sont substituables.

— Soient $t_1 = h(x, g(x), x)$ et $t_2 = h(g(u), g(g(z)), z)$.
Pour unifier ces termes, il faut que $x = g(u) = z$. Mais il faut aussi que $g(x) = g(g(z))$ et donc que $g(z) = x$, ce qui entraîne $z = g(z)$, ce qui n'est pas possible.

27. Unification de formules

L'unification sur les formules permet de transformer deux formules atomiques $R(\ldots)$ et $(\neg R(\ldots))$ de sorte que l'une soit la négation de l'autre.

Par exemple unifier

$(\neg R(f(u, g(v)), g(u)))$ et

$R(f(x, x)), g(z))$

conduit à

$u = z$ et $u = x = g(v)$

et les deux formules deviennent :

$R(f(g(v), g(v)), g(g(v)))$ et

$(\neg R(f(g(v), g(v)), g(g(v))))$.

28. Méthode de résolution

Pour prouver par résolution que

- — la conjonction des C_i entraîne la disjonction des B_i
  c.-à-d.
- — $C_1, ..., C_n \vdash B_1, ..., B_n$
  c.-à-d.
- — $(\wedge C_i) \Rightarrow (\vee B_j)$

on cherche une contradiction dans la conjonction

$$C_1, ..., C_n, (\neg B_1), ..., (\neg B_n)$$

Pour ce faire, la méthode de résolution consiste à :

1. écrire chaque C_i et chaque $(\neg B_j)$ comme une clause $\mathcal{C}_k$, et utiliser des variables différentes dans les diverses clauses.
2. unifier éventuellement plusieurs atomes positifs d'une clause et plusieurs atomes négatifs d'une autres clauses (plusieurs : sinon la méthode n'est pas complète)
3. simplifier les deux clauses unifiées

C'est plus simple sur un exemple.

29. Attention : résolution sur plusieurs atomes

Considérons les clauses :
$\forall u \forall v.\ P(u) \vee P(v)$
$\forall x \forall y.\ \neg P(x) \vee \neg P(y)$

Ces clauses ne sont pas satisfiables, la première entraine $\forall v.\ P(v)$ (faire $u := v$) et la seconde $\forall y.\ \neg P(y)$ (faire $x := y$).
Cependant il faut simplifier **simultanément**
$P(u)$ avec $\neg P(x)$
ET
$P(v)$ avec $\neg P(y)$
pour arriver à la clause vide : $x := u$ et $y := v$ par exemple.

(merci à Marie-Laure Mugnier pour cet exemple "percutant" repris de ses notes de cours)

30. Exemple résolution

1. Toutes les personnes qui entrent en voiture dans la faculté doivent avoir une carte ou être accompagnées par un membre du personnel.
2. Certains étudiants entrent en voiture dans la faculté sans être accompagnés de personnes qui ne sont pas des étudiants.
3. Aucun étudiant n'a de carte.

Montrer que certains étudiants sont membres du personnel

31. Formalisation en logique des prédicats

On considère les prédicats suivants :

— $V(x)$ x entre en voiture dans la faculté.
— $C(x)$ x possède une carte
— $P(x)$ x est membre du personnel de la faculté.
— $E(x)$ x est étudiant.
— $A(x, y)$ x est accompagné par y

32. Traduction logique

1. $\forall x \quad \left(V(x) \Rightarrow \left[C(x) \vee \left(\exists y \quad [A(x,y) \wedge P(y)] \right) \right] \right)$
2. $\exists x \quad \left(E(x) \wedge V(x) \wedge \neg \left[\exists y \quad A(x,y) \wedge \left(\neg E(y) \right) \right] \right)$
3. $\forall x \quad \left(E(x) \Rightarrow \left[\neg C(x) \right] \right)$

   Le but se traduit par :
4. $\exists x \quad \left(E(x) \wedge P(x) \right)$

33. Transformation des énoncés et de la négation du but en clauses

Il faut transformer chaque énoncé (en fait la conjonction des énoncés) en un ensemble de clauses. Le dernier énoncé est bien sûr la négation du but : lorsqu'on procède par résolution, on montre que les hypothèses augmentées de la négation du but sont contradictoires, ce qui montre que la conjonction des hypothèses entraîne le but.

34. Enoncé 1 $\rightarrow$ forme clausale 1/3

1. $\forall x \quad \Big(V(x) \Rightarrow \Big[C(x) \vee \big(\exists y \quad [A(x,y) \wedge P(y)]\big)\Big]\Big)$
2. $\forall x \quad \Big(V(x) \Rightarrow \Big[\exists y \quad \big(C(x) \vee [A(x,y) \wedge P(y)]\big)\Big]\Big)$
   (remontée du $\exists y$)
3. $\forall x \quad \Big(\exists y \quad \Big[V(x) \Rightarrow \big(C(x) \vee [A(x,y) \wedge P(y)]\big)\Big]\Big)$
   (remontée du $\exists y$)
4. $\forall x \quad \Big(\exists y \quad \Big[\big(\neg V(x)\big) \vee C(x) \vee \big(A(x,y) \wedge P(y)\big)\Big]\Big)$
   (transformation de $A \Rightarrow B$ en $(\neg A) \vee B$)

35. Enoncé 1 $\rightarrow$ forme clausale 2/3

5. $\forall x \quad \left(\left[\left(\neg V(x) \right) \vee C(x) \right] \vee \left[A(x, f(x)) \wedge P(f(x)) \right] \right)$
   (mise sous forme de Skolem : remplacement du y de $\forall x \exists y$, par $f(x)$ où f est un nouveau symbole de fonction)
6. $\forall x \quad \left(\left[\left(\neg V(x) \right) \vee C(x) \right] \vee \left[A(x, f(x)) \wedge P(f(x)) \right] \right)$
   (reparenthésage)

36. Enoncé 1 $\rightarrow$ forme clausale 3/3

7. $\forall x \quad \left(\left[\left(\neg V(x) \right) \vee C(x) \vee A(x, f(x)) \right] \right.$
$\left. \wedge \left[\left(\neg V(x) \right) \vee C(x) \vee P(f(x)) \right] \right)$
(distributivité de $\vee$ sur $\wedge$

8. $\left(\forall x \quad \left[\left(\neg V(x) \right) \vee C(x) \vee A(x, f(x)) \right] \right)$
$\wedge \left(\forall x \quad \left[\left(\neg V(x) \right) \vee C(x) \vee P(f(x)) \right] \right)$
(distributivité de $\forall x$ sur $\wedge$)

37. Enoncé 2 $\rightarrow$ forme clausale 1/2

1. $\exists x \quad \left(E(x) \wedge V(x) \wedge \left[\neg \left(\exists y \quad [A(x,y) \wedge \neg E(y)] \right) \right] \right)$
2. $\exists x \quad \left(\left[E(x) \wedge V(x) \right] \wedge \left[\forall y \quad \left([\neg A(x,y)] \vee E(y) \right) \right] \right)$
   (de Morgan : $(\neg(\exists y \quad (X \wedge Y))) \equiv (\forall y \quad ((\neg X) \vee (\neg Y))))$
3. $\exists x \quad \left(\forall y \quad \left[\left(E(x) \wedge V(x) \right) \wedge \left([\neg A(x,y)] \vee E(y) \right) \right] \right)$
   (remontée du $\forall y$)

38. Enoncé 2 $\rightarrow$ forme clausale 2/2

4. $\forall y \quad \left(\left[E(a) \wedge V(a)\right] \wedge \left[\left(\neg A(a,y)\right) \vee E(y)\right]\right)$

   (mise sous sforme de Skolem, nouvelle constante a remplaçant le x de $\exists x$)

5. $\left(\forall y \quad \left[E(a) \wedge V(a)\right]\right) \wedge \left(\forall y \quad \left[\left(\neg A(a,y)\right) \vee E(y)\right]\right)$

   (distributivité de $\forall$ sur $\wedge$)

6. $E(a) \wedge V(a) \wedge \left(\forall y \quad \left[\left(\neg A(a,y)\right) \vee E(y)\right]\right)$

   (Si la formule X ne contient pas de y alors $(\forall y \quad X) \equiv X$)

39. Enoncé 3 $\rightarrow$ forme clausale 1/1

1. $\forall x \quad \left(E(x) \Rightarrow \left[\neg C(x) \right] \right)$
2. $\forall x \quad \left(\left[\neg E(x) \right] \vee \left[\neg C(x) \right] \right)$
   (transformation de $A \Rightarrow B$ en $(\neg A) \vee B$)

40. Enoncé 4 $\rightarrow$ forme clausale 1/1

1. $\neg\left(\exists x \quad \left[E(x) \wedge P(x)\right]\right)$
2. $\forall x \quad \left(\left[\neg E(x)\right] \vee \left[\neg P(x)\right]\right)$

   (de Morgan)

41. Mise sous forme clausale

La conjonction des quatre formules (trois hypothèses et la négation du but) donne donc la liste de clauses suivantes.

Vous remarquerez remarquez que la première formule donne deux clauses et la deuxième formule trois.

42. Clauses correspondant à l'énoncé 1

I. $\forall x \quad \left(\left[\neg V(x)\right] \vee C(x) \vee A(x, f(x))\right)$

II. $\forall x \quad \left(\left[\neg V(x)\right] \vee C(x) \vee P(f(x))\right)$

43. Clauses correspondant à l'énoncé 2

III. $E(a)$

IV. $V(a)$

V. $\forall y \quad \left(\left[\neg A(a, y) \right] \vee E(y) \right)$

44. Clauses correspondant à l'énoncé 3

VI. $\forall x \quad \left(\left[\neg E(x) \right] \lor \left[\neg C(x) \right] \right)$

45. Clauses correspondant à l'énoncé 4

VII. $\forall x \quad \left(\left[\neg E(x) \right] \vee \left[\neg P(x) \right] \right)$

46. Résolution

8. $\neg C(a)$ (résolution sur les clauses III et VI)
9. $C(a) \vee A(a, f(a))$ (résolution sur les clauses I et IV)
10. $A(a, f(a))$ (résolution sur les clauses 8 et 9)
11. $C(a) \vee P(f(a))$ (résolution sur les clauses II et IV)
12. $P(f(a))$ (résolution sur les clauses 8 et 11)
13. $E(f(a))$ (résolution sur les clauses V et 10)
14. $\neg P(f(a))$ (résolution sur les clauses VII et 13)
15. $\perp$ (résolution sur les clauses 12 et 14)

les trois hypothèses plus la négation du but sont contradictoires. Les trois hypothèses entraînent le but.

47. Autre solution directe, sans résolution

On suppose les trois formules 1, 2 3, montrons le but 4.

1. $\forall x \quad \left(V(x) \Rightarrow \left[C(x) \vee \left(\exists y \quad [A(x,y) \wedge P(y)]\right)\right]\right)$
2. $\exists x \quad \left(E(x) \wedge V(x) \wedge \neg\left[\exists y \quad A(x,y) \wedge \left(\neg E(y)\right)\right]\right)$
3. $\forall x \quad \left(E(x) \Rightarrow \left[\neg C(x)\right]\right)$
4. (BUT) $\exists x \quad \left(E(x) \wedge P(x)\right)$

On sait d'après la deuxième formule qu'il existe x_0 tel que $(E(x_0) \wedge V(x_0) \wedge \neg(\exists y \quad A(x_0, y) \wedge (\neg E(y))))$.

Comme on a $E(x_0)$, à cause de la troisème formule on a $\neg C(x_0)$. A cause de la première formule, comme $V(x_0$, et $\neg C(x_0)$, il existe y_0 tel que $A(x_0, y_0)$ et $P(y_0)$. Comme $\neg(\exists y \quad A(x_0, y) \wedge (\neg E(y))))$ et que $A(x_0, y_0)$ on a $E(y_0)$. **On a donc un y_0 tel que $P(y_0)$ et $E(y_0)$.**

48. Le mécanisme de Prolog 1/5

Un programme Prolog est une liste de p clauses de Horn (j variant de 1 à p) — éventuellement il n'y a pas de F^j :

$$Clause^j : \quad F^j :- H_1^j, \ldots, H_{n_j}^j$$

Chacune se comprend comme une formule :

$$C^j : \quad \forall x, \forall y, \ldots (H_1^j \wedge \ldots \wedge H_{n_j}^j \Rightarrow F^j) = \forall x, \forall y, \ldots \neg H_1^j \vee \ldots \vee \neg H_{n_j}^j \vee F^j)$$

où $x, y, \ldots$ sont toutes les variables de la formule/clause.

On observe qu'une seule formule est positive par clause.

Le programme n'est autre que la conjonction de ces clauses.

49. Le mécanisme de Prolog 2/5

Lorsqu'on cherche à réaliser le but G, c.-à-d.$G = \exists u, v, \ldots B$, Prolog cherche à trouver des valeurs pour u, $v \ldots$ telle que

$$\bigwedge_{j=1,\ldots,P} C^j \Rightarrow G$$

ce qui revient à réfuter

$$W = (\neg G) \wedge \left(\bigwedge_{j=1,\ldots,P} C^j \right)$$

c.-à-d.à montrer

$$(\neg G) \wedge \left(\bigwedge_{j=1,\ldots,P} C^j \right) \vdash$$

On observe alors que $\neg G$ est aussi de la forme $\forall u \forall v \ldots (\neg B)$. C'est aussi une clause, mais sans formule atomique positive.

On a donc un ensemble de clauses dont une est particulière (pas de formule positive) et dont toutes les autres ont au plus une formule positive.

50. Le mécanisme de Prolog 3/5

Le théorème de Herbrand affirme alors que si cet ensemble de clauses est inconsistant alors il existe des instanciations des variables universelles (on les instancie par clauses, mais les clauses utilisent des variables distinctes) par des termes de Herbrand qui réfutent cette clause et la réfutation est faite par la méthode de résolution.

Plutôt que d'énumérer successivement tous les termes de Herbrand, on résout les clauses en utilisant l'unification, ce qui donne les termes de Herbrand nécessaires. C'est cette propriété de commutation entre unification et résolution qui est à la base Prolog.

51. Le mécanisme de Prolog 4/5

Une seconde optimisation utilisée par Prolog, et que pour ces ensembles de clauses particulières (une clause négative et une seule formule positive dans les autres clauses) on peut partir du but (la clause négative) et construire par résolution et unification une suite de clauses ainsi :

— le but
— résolvant du but et d'une des clauses initiales
— résolvant de la clause précédente et d'une des clauses initiales
— résolvant de la clause précédente et d'une des clauses initiales
— ...
— résolvant de la clause précédente et d'une des clauses initiales
— $\perp$ (la clause vide établissant la contradiction)

Dans cette suite, une même clause initiale peut bien sûr être utilisée plusieurs fois.

52. Le mécanisme de Prolog 5/5

L'un des intérêt de Prolog est qu'il ne se contente pas de dire « il y a une solution » mais qu'il calcule une solution en fonction des constantes (et des fonctions) présentes dans le but et dans les clauses.

Cela est possible et même facile pour les clauses particulières utilisées et avec la méthode ci-dessus en gardant mémorisant les substitutions, comme on va le voir sur un exemple.

53. Un exemple de résolution à la Prolog 1/6

Montrons ainsi l'inconsistance de l'ensemble des trois clauses :

Clause 1 $(\neg F(x, y) \vee \neg F(y, z) \vee G(x, z))$

Clause 2 $F(g(x'), x'))$

Clause 3 $\neg G(x'', a))$ — *but*

54. Un exemple de résolution à la Prolog 2/6

Il faut trouver une clause qui se résolve, par unification, avec le but. On peut unifier $G(x'', a)$ de 3 et $G(x, z)$ de 1 ce qui conduit à la substitution $\sigma(x) = x''$ et $\sigma(z) = a$ et donc aux clauses (il faut appliquer la substitution à toute la clause qu'on instancie) $(\neg F(x'', y) \vee \neg F(y, a) \vee G(x'', a))$ et $\neg G(x'', a))$ dont le résolvant à ajouter aux clauses initiales, est la

Clause 4 $(\neg F(x'', y) \vee \neg F(y, a)$

55. Un exemple de résolution à la Prolog 3/6

On peut ensuite unifier $F(y, a)$ avec $F(g(x'), x'))$ (clause 2) par la substitution $\tau(x') = a$ et $\tau(y) = g(a)$, ce qui donne les clauses $(\neg F(x'', g(a)) \vee \neg F(g(a), a)$ (d'après la clause 4) et $F(g(a), a)$ (d'après la clause 2) dont le résolvant est la clause :

Clause 5 $\neg F(x'', g(a))$

56. Un exemple de résolution à la Prolog 4/6

On peut maintenant unifier pour appliquer la résolution à cette clause et à une autre instance de la clause initiale $F(g(x'), x')$ (clause 2).

La substitution obtenue est : $\phi(x') = g(a)$,$\phi(x'') = g(g(a))$,

et les deux clauses deviennent :

$F(g(g(a)), g(a))$ (d'après la clause 2) et

$\neg F(g(g(a)), g(a))$ (d'après la clause 5)

dont le résolvant est la clause vide ($\perp$),
on a donc dérivé une contradiction.

57. Un exemple de résolution à la Prolog 5/6

On peut observer que les substitutions utilisées déterminent la valeur à donner au variables (dans l'univers de Herbrand) pour obtenir la contradiction. Ici on voit que la seule substitution sur x'' est $\phi(x'') = g(g(a))$

La résolution vue ci-dessus correspond au calcul qu'effectue Prolog lorsque les clauses sont :

$$G(x, z) :- F(x, y), F(y, z).$$

$$F(g(x'), x').$$

et que le but est :

$$? - G(x'', a).$$

58. Un exemple de résolution à la Prolog 6/6

On peut suivre plus directement le calcul sans mettre sous forme clausale :

$G(x'', a)$ est produit par $G(x, z) :- F(x, y), F(y, z)$. avec $x = x''$ et $z = a$. Il faut donc produire $F(x'', y)$ et $F(y, a)$ pour une *même* valeur de y.

Pour produire $F(y, a)$, on peut utiliser la clause $F(g(x'), x')$. qui conduit à $x' = a$ et $y = g(a)$. On a à ce moment-là déterminé une valeur possible de y : $g(a)$.

Il faut donc réussir à montrer $F(x'', g(a))$. En utilisant la clause $F(g(x'), x')$. on y parvient et cela conduit à $x'' = g(g(a))$.

59. Une méthode de résolution 1/3 – règles

On peut formuler les règles de résolution ainsi, qui prennent en compte, avec les règles $fact^{+}$ et $fact^{-}$, la nécessité de simplifier plusieurs atomes simultanément (comme vu dans le transparent 29).

Règle de Résolution

$$\frac{A \vee C \qquad \neg B \vee D}{\sigma(C) \vee \sigma(D)} \text{res où } \sigma(A) = \sigma(B).$$

Règles de factorisation

$$\frac{A \vee B \vee C}{\sigma(B) \vee \sigma(C)} \text{fact}^{+} \qquad \frac{\neg A \vee \neg B \vee C}{\neg\sigma(B) \vee \sigma(C)} \text{fact}^{-} \quad \text{où } \sigma(A) = \sigma(B).$$

60. Une méthode de résolution 2/3 – mise an garde

Soit S un ensemble de clauses à réfuter. On peut appliquer les règles de résolution ci-dessus suivant une stratégie particulière. C'est ce que fait la méthode *given-clause* présentée ci-dessous comme un algorithme, mais qui n'est **pas un algorithme** pour la raison suivante :
"S est insatisfiable" est une question indécidable : il n'y a pas d'algorithme terminant assurément et qui conclut par S satisfiable ou S instatisfiable.

Néanmoins, en pratique cette méthode fonctionne bien, et termien souvent soit par "S satisfiable soit par $\perp$ (S insatisfiable).

61. Une méthode de résolution 3/3 – "given-clause"

Sat contient l'ensemble des clauses déjà traitées et permet de ne pas remettre dans S des clauses (lorsqu'on génère de nouveaux résolvants) qu'on a déjà vues. Cette méthode (ici : version D. Delahaye) maintient l'invariant que toutes les inférences non redondantes entre deux clauses dans Sat ont été appliquées.

$Sat := \emptyset$;
tant que $S \neq \emptyset$ faire
  choisir $C \in S$;
  $S := S \setminus \{C\}$;
  si $C = \square$ alors retourner « insatisfiable » ;
  si C est une tautologie alors ; (* passer à la clause suivante *)
  sinon, si $C \in Sat$ alors ; (* idem *)
  sinon pour tout résolvant C_1 entre C
  et une clause de $Sat \cup \{C\}$ faire
    $S := S \cup \{C_1\}$;
  $Sat := Sat \cup \{C\}$;
retourner « satisfiable ».

62. Correction et complétude de la résolution

Nous verrons dans la suite de ce cours que la méthode de résolution est correcte et complète :

> **si un ensemble de clause est insatisfiable, alors il existe une suite finie de résolutions qui dérive une contradiction**
>
> (la réciproque est évidente)

plus généralement

> **une formule (ou un séquent) est démontrable**
>
> si et seulement si
>
> **il est vrai pour toute interprétation**

La complétude du calcul des prédicats n’empêche pas que

> **le calcul des prédicats est indécidable : il n’y a pas d’algorithme qui décide en un temps fini si un séquent (ou une formule) est démontrable, si une formule est satisfiable, si un ensemble de clauses est réfutable etc.**