3.4 Le théorème de Herbrand

Dans l'espoir de résoudre le *Entscheidungsproblem*, nous avons déjà réduit le cas général à la satisfaisabilité d'un ensemble de formules purement universel. Nous allons plus loin ici en montrant qu'un ensemble de formules purement universel est satisfaisable si et seulement si il a un modèle dans une classe bien particulière : les structures de Herbrand. Ceci permettra ensuite de se ramener au calcul propositionnel (au prix de passer d'un ensemble fini de formules à un ensemble infini de formules) et de démontrer que l'insatisfaisabilité (ou la validité) est récursivement énumérable. Nous verrons que cela permet aussi de généraliser le théorème de compacité à la logique du premier ordre, par exemple.

Dans la suite, nous verrons malheureusement que le *Entscheidungsproblem* est indécidable et donc que l'ensemble des formules valides n'est pas récursif. Mais le théorème de Herbrand, avec des lemmes de relèvement, permettra d'utiliser des systèmes de preuves étudiés dans le cas propositionnel, en particulier de montrer que le relèvement de la résolution au premier ordre est réfutationnellement complet.

3.4.1 Les structures de Herbrand

Definition 3.4.1 *Soit $\mathcal{F}$ un ensemble de symboles de fonctions qui contient une constante, et $\mathcal{P}$ un ensemble de symboles de prédicats.*

1. *Le* $\mathcal{F}, \mathcal{P}$-univers de Herbrand *est* $T(\mathcal{F})$.
2. *Une* $\mathcal{F}, \mathcal{P}$-structure de Herbrand $\mathcal{H}$ *est une* $\mathcal{F}, \mathcal{P}$*-structure avec domaine* $T(\mathcal{F})$*, et telle que* $f_H(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$ *pour tous les* $f \in \mathcal{F}$ *et tous les* $t_1, \ldots, t_n \in T(\mathcal{F})$.
3. *La* $\mathcal{F}, \mathcal{P}$-base de Herbrand *est l'ensemble des atomes clos sur* $\mathcal{F}, \mathcal{P}$ *:* $\{P(t_1, \ldots, t_n\} \mid t_i \in T(\mathcal{F}), P \in \mathcal{P}\}$.

Étant données $\mathcal{F}, \mathcal{P}$, toutes les structures de Herbrand ont le même univers et interprètent les symboles de fonction de la même façon. La seule liberté qui reste dans le choix d'une structure de Herbrand est le choix des interprétations des symboles de prédicat.

On peut donc identifier chaque $\mathcal{F}, \mathcal{P}$-structure de Herbrand avec une partie de la $\mathcal{F}, \mathcal{P}$-base de Herbrand : On identifie une $\mathcal{F}, \mathcal{P}$-structure de Herbrand $\mathcal{H}$ avec la partie de la $\mathcal{F}, \mathcal{P}$-base de Herbrand qui consiste en tous les atomes clos qui sont vrais en $\mathcal{H}$. Ainsi, la structure de Herbrand où tous les prédicats sont toujours faux correspond à l'ensemble vide, et la structure de Herbrand où tous les prédicats sont toujours vrais correspond à la base de Herbrand elle-même

Exercice 99 (5)
Donner un exemple de formule satisfaisable, qui n'a pas de modèle fini et qui n'a pas de modèle de Herbrand.

Dans la suite nous supposons que $\mathcal{F}$ contient toujours une constante. Remarquons que, si S est un ensemble de $\mathcal{F}, \mathcal{P}$-formules et $\mathcal{F}$ ne contient pas de constante, alors S possède un $\mathcal{F}, \mathcal{P}$-modèle si et seulement si S possède un $\mathcal{F} \cup \{a\}, \mathcal{P}$-modèle où a est un symbole de constante.

Lemme 3.4.1 *Soit σ une application de $\mathcal{X}$ dans $T(\mathcal{F})$, $\mathcal{A}$ une $\mathcal{F}$-algèbre et $\sigma_{\mathcal{A}}$ l'affectation définie par $x\sigma_{\mathcal{A}} = [\![x\sigma]\!]_{\mathcal{A}}$, pour tout $x \in \mathcal{X}$. Alors, pour tout terme $t \in T(\mathcal{F}, \mathcal{X})$, $[\![t\sigma]\!]_{\mathcal{A}} = [\![t]\!]_{\sigma_{\mathcal{A}},\mathcal{A}}$.*

Preuve:
Par récurrence sur t : dans le cas de base, $[\![x\sigma]\!]_{\mathcal{A}} = x\sigma_{\mathcal{A}} = [\![x]\!]_{\sigma_{\mathcal{A}},\mathcal{A}}$ par définition et si $t = f(t_1, \ldots, t_n)$, $[\![f(t_1, \ldots, t_n)\sigma]\!]_{\mathcal{A}} = [\![f(t_1\sigma, \ldots, t_n\sigma)]\!]_{\mathcal{A}}$ (par morphisme) $= f_{\mathcal{A}}([\![t_1\sigma]\!]_{\mathcal{A}}, \ldots, [\![t_n\sigma]\!]_{\mathcal{A}}) = f_{\mathcal{A}}([\![t_1]\!]_{\sigma_{\mathcal{A}},\mathcal{A}}, \ldots, [\![t_n]\!]_{\sigma_{\mathcal{A}},\mathcal{A}})$ (par hypothèse de récurrence) $= [\![f(t_1, \ldots, t_n)]\!]_{\sigma_{\mathcal{A}},\mathcal{A}}$ (par morphisme). □

Une formule est *universelle* si elle est en forme prénexe et toutes ses variables sont quantifiées universellement.

Théorème 3.4.1 (Herbrand) *Soit S un ensemble de $\mathcal{F}, \mathcal{P}$-formules universelles. S est satisfaisable si et seulement si S a un modèle qui est une $\mathcal{F}, \mathcal{P}$-structure de Herbrand.*

Preuve:
Un seul sens de l'implication demande une preuve : supposons que $\mathcal{S} \models S$. On considère alors, pour chaque $P \in \mathcal{P}$ l'interprétation $P_{\mathcal{H}}$ dans l'univers de Herbrand :

$$P_{\mathcal{H}} = \{(t_1, \ldots, t_n) \mid ([\![t_1]\!]_{\mathcal{S}}, \ldots, [\![t_n]\!]_{\mathcal{S}}) \in P_{\mathcal{S}}\}$$

Montrons que la structure de Herbrand $\mathcal{H}$ satisfait S : si $\forall x_1, \ldots \forall x_m.\phi \in S$ et ϕ est sans quantificateur et $t_1, \ldots, t_m \in T(\mathcal{F})$, on montre, par récurrence sur ϕ, que $\mathcal{H}, \sigma \models \phi$ ssi $\mathcal{S}, \theta \models \phi$ où $\sigma = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ et θ est l'affectation $\{x_1 \mapsto [\![t_1]\!]_{\mathcal{S}}, \ldots, x_n \mapsto [\![t_n]\!]_{\mathcal{S}}\}$.

Si ϕ est une formule atomique, $\mathcal{H}, \sigma \models P(u_1, \ldots, u_m)$ ssi $(u_1\sigma, \ldots, u_m\sigma) \in P_{\mathcal{H}}$ ssi $([\![u_1\sigma]\!]_{\mathcal{S}}, \ldots, [\![u_m\sigma]\!]_{\mathcal{S}}) \in P_{\mathcal{S}}$ ssi $([\![u_1]\!]_{\theta,\mathcal{S}}, \ldots [\![u_m]\!]_{\theta,\mathcal{S}}) \in P_{\mathcal{S}}$ (par le lemme 3.4.1) ssi $\mathcal{S}, \theta \models P(u_1, \ldots, u_m)$.

L'étape de récurrence est immédiate (ϕ ne contient pas de quantificateur). □

Théorème 3.4.2 *Soit S un ensemble de $\mathcal{F}, \mathcal{P}$-formules closes. S est satisfaisable si est seulement s'il existe un $\mathcal{F}'$ avec $\mathcal{F} \subseteq \mathcal{F}'$ et une $\mathcal{F}', \mathcal{P}$-structure de Herbrand qui satisfait S.*

Preuve:
Pour montrer que quand S est satisfaisable alors S possède un $\mathcal{F}', P$-modèle de Herbrand pour un $\mathcal{F}' \supseteq \mathcal{F}$ (l'autre direction est triviale) :

Étant donné S, on construit l'ensemble S_{pr} des formes prénexes des formules de S. Les ensembles S et S_{pr} ont les mêmes modèles.

Puis, on construit l'ensemble S_{sk} de toutes les skolémisations des formules en S_{pr}. La skolémisation introduit des nouveaux symboles de fonction, on obtient donc une extension $\mathcal{F}'$ de $\mathcal{F}$ (Attention, $\mathcal{F}' - \mathcal{F}$ peut être infini dans le cas où S est infini). D'après la proposition 3.3.1, S_{pr} possède un $\mathcal{F}, \mathcal{P}$-modèle si et seulement si S_{sk} possède un $\mathcal{F}', \mathcal{P}$-modèle. En fait, chaque modèle de S_{sk} est un modèle de S_{nf} dans le langage étendu $(\mathcal{F}', \mathcal{P})$.

Si $\mathcal{A} \models S$ alors $\mathcal{A} \models S_{pr}$, et S_{sk} est satisfaisable. S_{sk} est un ensemble de formules universelles closes. Donc, par Théorème 3.4.1, S_{sk} possède un $\mathcal{F}', \mathcal{P}$-modèle de Herbrand, qui est dans le langage étendu alors aussi un modèle de S_{pr} et de S. □

Exercice 100 (3)
Soit $\mathcal{F} = \emptyset$, $\mathcal{P} = \{P(2)\}$, et S l'ensemble des deux formules

$$\forall x \exists y\, R(x, y)$$
$$\exists x \forall y (R(x, y) \rightarrow \exists z (R(x, z) \wedge \neg R(z, y)))$$

Construire un $\mathcal{F}'$ et une $\mathcal{F}', \mathcal{P}$-structure de Herbrand qui est un modèle de S.

3.4.2 Conséquences du théorème de Herbrand

Nous revenons à la question d'une méthode de sémi-décision de la validité (ou non-satifaisabilité) à la fin de cette section. D'abord nous donnons quelques conséquences du théorème de Herbrand.

Une première conséquence est le théorème de Skolem et Löwenheim, disant que chaque théorie satisfaisable a un « petit » modèle. Remarquez que le Théorème 3.4.1 ne s'applique qu'aux formules universelles.

Théorème 3.4.3 (Skolem et Löwenheim) *Soient $\mathcal{F}$ et $\mathcal{P}$ dénombrables. Si un ensemble S de $\mathcal{F}, \mathcal{P}$-formules a un modèle alors S a un modèle dénombrable (c'est-à-dire, un modèle avec un domaine dénombrable).*

Plus généralement si S a un modèle alors S à un modèle de cardinalité $\leq \max(\aleph_0, \mathit{card}(\mathcal{F} \cup \mathcal{P}))$.

(La cardinalité dénombrable est notée $\aleph_0$, prononcée « Aleph Zéro ».)

Preuve:
La preuve est un raffinement de la preuve du Théorème 3.4.2, il faut simplement « compter » les cardinalités des ensembles de formules construits dans cette preuve.

Dans le cas d'un langage dénombrable : Soient $\mathcal{F}$ et $\mathcal{P}$ dénombrables. L'ensemble de tous les $\mathcal{F}, \mathcal{P}$-formules est donc aussi dénombrable. En particulier, pour un ensemble S donné, sa forme prénexe S' est dénombrable. Puisque chaque formule ne contient qu'un nombre fini de quantificateurs, la skolémisation de S' peut introduire un ensemble dénombrable de nouvelles symboles de fonctions. La cardinalité de l'univers de Herbrand est donc dénombrable.

Pour le cas général : Soit $\alpha = \text{card}(\mathcal{F} \cup \mathcal{P})$. Il y a $\max(\alpha, \aleph_0)$ formules construites sur $\mathcal{F}, \mathcal{P}$. Pour un ensemble S donné, sa forme prénexe S' ne peut pas contenir plus que $\max(\alpha, \aleph_0)$ formules. Puisque chaque formule ne contient qu'un nombre fini de quantificateurs, la skolémisation de S' peut introduire au maximum $\max(\alpha, \aleph_0)$ nouvelles symboles de fonctions. La cardinalité de l'univers de Herbrand est donc au maximum $\max(\max(\alpha, \aleph_0), \aleph_0) = \max(\alpha, \aleph_0)$. □

Exercice 101
Supposant que $\mathcal{F}, \mathcal{P}$ sont dénombrables. Le théorème 3.4.3 est le théorème de Löwenheim-Skolem *descendant*. Montrer le théorème *ascendant* : si S est un ensemble de $\mathcal{F}, \mathcal{P}$-formules qui a un modèle de cardinal $\alpha \geq \aleph_0$, alors, pour tout $\beta \geq \alpha$, S a un modèle de cardinal β. (On rappelle que les cardinaux sont des classes d'équivalence d'ensembles pour la relation d'équivalence "être en bijection". L'ordre sur les cardinaux est défini par $\beta \geq \alpha$ s'il existe une surjection de β dans α ou, de manière équivalente, s'il existe une injection de α dans β).

Une autre conséquence du théorème de Herbrand est qu'on peut maintenant transférer certains théorèmes de la logique propositionnelle vers la logique du premier ordre. L'idée est, étant donnée une formule universelle, de construire ses *instances de Herbrand* :

$$H(\forall x_1, \ldots, x_n\, P) := \{P[x_1 \mapsto t_1, \ldots, x_n \mapsto t_n] \mid t_1, \ldots, t_n \in T(\mathcal{F})\}$$

et, pour un ensemble S, $H(S) := \{H(s) \mid s \in S\}$.

Théorème 3.4.4 *Soit S un ensemble de $\mathcal{F}, \mathcal{P}$-formules universelles, et soit $\mathcal{P}'$ la base de Herbrand sur $\mathcal{F}, \mathcal{P}$. S a un $\mathcal{F}, \mathcal{P}$-modèle si et seulement si l'ensemble de formules propositionnelles $H(S)$ a un modèle propositionnel, où $\mathcal{P}'$ est considéré comme l'ensemble des variables propositionnelles.*

Preuve:
Par Théorème 3.4.1. Remarquez que, pour toute structure de Herbrand $\mathcal{H}$ et toute formule universelle close ϕ, $\mathcal{H} \models \phi$ ssi $\mathcal{H} \models H(\phi)$. □

Théorème 3.4.5 (Compacité) *Un ensemble de formules du premier ordre est satisfaisable si et seulement si chacun des ses sous-ensembles finis est satisfaisable.*

Preuve:
Si $\mathcal{A}$ est un modèle de S alors $\mathcal{A}$ est évidemment aussi un modèle de chaque sous-ensemble (fini) de S.

Pour une formule ϕ, nous notons $su(\phi)$ sa forme prénexe et skolemisée. Supposons que T n'a pas de modèle. Donc, $su(T)$ n'a pas de modèle. Par le théorème 3.4.4, $H(su(T))$ n'a pas de modèle (propositionnel). Par la compacité de la logique propositionnelle, il y a un sous-ensemble fini $H' \subseteq H(su(T))$ qui n'a pas de modèle propositionnel. Donc, il existe un sous-ensemble fini $S' \subseteq su(T)$ tel que $H' \subseteq H(S')$. Évidemment, $H(S')$ n'a pas de modèle propositionnel, donc, par le théorème 3.4.4, S' n'a pas de modèle. Par conséquent, il y a un sous-ensemble fini $T' \subseteq T$ tel que $su(T') = S'$, et qui n'a pas de modèle. □

Le théorème de compacité a des conséquences sur les limites de l'expressivité de la logique du premier ordre (théorème de Lindström).

L'exercice suivant montre que le théorème de compacité échoue en théorie des modèles finis :

Exercice 102 (5)
Donner un ensemble S de formules du premier ordre (sur un alphabet fini) tel que tout sous-ensemble fini de S a un modèle fini, mais S n'a pas de modèle fini.

Exercice 103
Si ϕ est une formule propositionnelle sur les variables propositionelles $\mathcal{P}$, on considère, pour une variable $x \in \mathcal{X}$, la traduction $T(\phi, x)$ suivante dans $CP_1(\mathcal{P}, \mathcal{F}, \mathcal{X})(\emptyset, \mathcal{P} \cup \{\leq (2)\})$, où tous les éléments de $\mathcal{P}$ sont d'arité 1 :

— $T(\top, x) = \top$, $T(\bot, x) = \bot$
— $T(P, x) = P(x)$ si $P \in \mathcal{P}$
— $T(\phi \wedge \psi, x) = T(\phi, x) \wedge T(\psi, x)$
— $T(\phi \vee \psi, x) = T(\phi, x) \vee T(\psi, x)$
— $T(\neg\phi, x) = \forall y.(x \leq y \rightarrow \neg T(\phi, y))$
— $T(\phi \rightarrow \psi, x) = \forall y.(x \leq y \rightarrow (T(\phi, y) \rightarrow T(\psi, y))$

On considère de plus l'ensemble $\mathcal{E}$ de formules $\mathcal{E} = \{\forall x, \forall y.(P(x) \wedge x \leq y) \rightarrow P(y) \mid P \in \mathcal{P}\} \cup \{\forall x.x \leq x,\ \forall x.\forall y.\forall z.(x \leq y \wedge y \leq z) \rightarrow x \leq z\}$.

Montrer que ϕ est satisfaisable en logique propositionnelle intuitioniste ssi $\{\forall x.T(\phi, x)\} \cup \mathcal{E}$ est satisfaisable en logique du premier ordre. En déduire un théorème de compacité pour NJ_0.

3.4.3 Le cas des clauses de Horn

L'ordre d'inclusion sur les sous-ensembles de la base de Herbrand entraîne un ordre sur les structures de Herbrand : Une structure de Herbrand $\mathcal{H}_1$ est dite plus petite qu'une structure de Herbrand $\mathcal{H}_2$ si pour tout symbole de prédicat P, $P_{\mathcal{H}_1} \subseteq P_{\mathcal{H}_2}$. De plus, toutes les opérations sur les parties de la base de Herbrand, comme complément, union, intersections, se transfèrent maintenant de manière naturelle vers les structures de Herbrand. On peut donc parler de l'intersection, l'union, etc. de deux structures de Herbrand (sur le même $\mathcal{F}, \mathcal{P}$).

Exercice 104 (3)
Donner un exemple d'une formule qui possède plusieurs modèles de Herbrand minimaux (et qui n'a donc pas de plus petit modèle de Herbrand).

Si une formule purement universelle satisfaisable n'a pas nécessairement de plus petit modèle de Herbrand, en revanche les clauses de Horn ont cette propriété :

Definition 3.4.2 *Une* clause de Horn *est une clause (dont les variables sont quantifiées universellement) qui contient au plus un littéral positif.*

Théorème 3.4.6 *Un ensemble de clauses de Horn satisfaisable possède un plus petit modèle de Herbrand.*

Preuve:
On peut par exemple remarquer que l'intersection de modèles de Herbrand de E est encore un modèle de Herbrand de E (lorsque E ne contient que des clauses de Horn).

Le plus petit modèle de Herbrand peut aussi explicitement s'obtenir comme plus petit point fixe de l'opérateur de conséquence immédiate, ce qui est utile pour la suite : si E est un ensemble de clauses de Horn et S est un sous-ensemble de la base de Herbrand $\mathcal{B}$, on note

$$T_E(S) = \quad S \cup \{P(t_1,\ldots,t_n) \in \mathcal{B} \mid \quad \exists P(u_1,\ldots,u_n) \vee \neg A_1 \vee \ldots \vee \neg A_m \in E,\ \exists \sigma, \\ t_1 = u_1\sigma, \ldots, t_n = u_n\sigma, A_1\sigma \in S, \ldots, A_m\sigma \in S\}$$

Alors E est satisfaisable si et seulement si $LHM(E) = \bigcup_{n\in\mathbb{N}} T_E^n(\emptyset) \models E$. De plus, tout modèle de Herbrand de E contient $LHM(E)$.

En effet, tout d'abord, par récurrence sur n, si $A \in T_E^n(\emptyset)$, alors $E \models A$. Donc tout modèle de Herbrand de E contient bien $LHM(E)$.

Réciproquement, si $C \in E$ et C contient un littéral positif, alors $LHM(E)) \models C$, en effet, soit $C = A \vee \neg A_1 \vee \ldots \vee \neg A_n$. Pour toute affectation σ des variables de C dans $T(\mathcal{F})$, ou bien il existe un i tel que $A_i\sigma \notin LHM(E)$, ou bien pour tout i, il existe un n_i tel que $A_i\sigma \in T_E^{n_i}(\emptyset)$. Par définition on a alors $A\sigma \in T_E^{\max n_i+1}(\emptyset)$ et donc $A\sigma \in LHM(E)$.

Soit maintenant $C = \neg A_1 \vee \ldots \vee \neg A_m$ une clause purement négative de E. S'il existe une affectation σ telle que $A_i\sigma \in LHM(E)$ pour $i = 1, ...m$, alors E est insatisfaisable puisque $E \models A_i\sigma$ dès que $A_i\sigma \in LHM(E)$. Sinon, $LHM(E) \models C$.

3.5 La validité est récursivement énumérable et pas récursive

Théorème 3.5.1 *Si $\mathcal{F}$ et $\mathcal{P}$ sont récursivement énumérables, alors l'ensemble des formules valides est récursivement énumérable.*

Preuve:
Soit ϕ une formule. Puisque ϕ ne peut contenir qu'un nombre fini de symboles de fonction et de symboles de prédicats, nous pouvons supposer que $\mathcal{F}$ et $\mathcal{P}$ sont finis. Nous construisons, par mise en forme prénexe et skolémisation de la *négation de* ϕ, un ensemble de formules universelles U qui est non-satisfaisable si et seulement si ϕ est valide. Remarquons que l'ensemble U est fini, et que le nombre de symboles de fonction introduits par la skolémisation est fini. $H(U)$ est donc dénombrable. Par le Théorème 3.4.4, il suffit de tester si $H(U)$ est non-satisfaisable dans la logique propositionnelle.

Si, $H(U)$ est fini, il suffit de tester si $H(U)$ est non-satisfaisable.

Sinon, $H(U) = \{u_i \mid i \geq 0\}$, et cette énumération est effective. Par le théorème de compacité de la logique propositionnelle, il suffit de tester s'il y a un sous-ensemble fini de $H(U)$ qui est non-satisfaisable. Au lieu de faire ce test pour tous les sous-ensembles finis de $H(U)$, il est plus facile de tester successivement si les ensembles $\{u_i \mid 0 \leq i \leq n\}$ sont non-satisfaisable. L'algorithme peut donc être écrit en pseudo-code comme

```
T := ∅ ;
n := 0 ;
while satisfaisable(T) do begin
   T := T ∪ { u_n } ;
   n := n + 1 ;
end ;
print "la formule est valide" ;
```

Si $H(U)$ est satisfaisable alors le test de la boucle donne toujours vrai, et l'exécution de l'algorithme ne termine pas. Si $H(U)$ n'est pas satisfaisable alors l'exécution s'arrête dès qu'on tombe sur un sous-ensemble fini et non-satisfaisable de $H(U)$. □

Théorème 3.5.2 *Le problème suivant est indécidable :*

Donnée : *une formule ϕ de la logique du premier ordre*

Question : *ϕ est-elle satisfaisable ?*

On réduit le problème de l'arrêt (plus exactement son complémentaire, qui n'est pas décidable non plus). Si w est un mot de Σ^*, on note $\widetilde{w}$ son image miroir : $\widetilde{\epsilon} = \epsilon$ et $\widetilde{a \cdot w} = \widetilde{w} \cdot a$.

Si M est une machine de Turing et w un mot d'entrée, on considère l'ensemble $\mathcal{P} = Q \cup \{\mathbf{accept}, \mathbf{reject}\}$, ensemble des états de M comme ensemble de symboles de prédicats et $\mathcal{F} = \Sigma \cup \{0\}$ comme ensemble de symboles de fonction,

tous les éléments de Σ étant d'arité 1 et 0 étant d'arité 0. Si w est un mot, on note $\overline{w}$ le terme défini par $\overline{\epsilon} = 0$ et $\overline{a \cdot w} = a(\overline{w})$. On construit alors un ensemble de clauses $C(M, w)$ comme suit :

- $q_0(0, \overline{\$w})$ est une clause de $C(M, w)$, si q_0 est l'état initial de M.
- $\forall x, y.\neg\mathbf{accept}(x, y) \in C(M, w)$ et $\forall x, y.\neg\mathbf{reject}(x, y) \in C(M, w)$
- pour chaque $q \in Q$, $\forall x.(q(x, 0) \to q(x, B(0))) \in C(M, w)$.
- Pour chaque règle de transition $\delta(q, a) = (q', b, \to)$ de M, $\forall x, y.(q(x, a(y)) \to q'(b(x), y)) \in C(M, w)$.
- Pour chaque règle de transition $\delta(q, a) = (q', b, \downarrow)$ de M, $\forall x, y.(q(x, a(y)) \to q'(x, b(y))) \in C(M, w)$.
- Pour chaque règle de transition $\delta(q, a) = (q', b, \leftarrow)$ de M, $\forall x, y.(q(c(x), a(y)) \to q'(x, c(b(y))) \in C(M, w)$

La formule ϕ est la conjonction des formules de $C(M, w)$. Comme ϕ est purement universelle, d'après le théorème de Herbrand, ϕ est satisfaisable si et seulement si elle a un modèle de Herbrand.

Supposons d'abord que ϕ a un modèle de Herbrand H et montrons que la machine M ne s'arrête pas sur w. Par récurrence sur n on montre que $\gamma_0 \vdash^n \gamma_n$ où $\gamma_n = (w_n, q_n, w'_n)$ et $q_n(\widetilde{\overline{w_n}}, \overline{w'_n}) \in H$, $q_n \notin \{\mathbf{accept}, \mathbf{reject}\}$.

Pour $n = 0$, $\gamma_0 = (\epsilon, q_0, \$w)$ et comme $H \models q_0(0, \overline{\$w})$, qui est dans la base de Herbrand, on doit avoir $q_0(0, \overline{\$w}) \in H$.

Si $\gamma_n = (w_n, q_n, w'_n)$ et $q_n \notin \{\mathbf{accept}, \mathbf{reject}\}$, soit $w'_n = a_n w''_n$. On distingue suivant les transitions de la machine de Turing :

- si $\delta(q_n, a_n) = (q_{n+1}, b, \leftarrow)$, et $w_n = w'''_n \cdot c$, alors $H \models \forall x, y.(q_n(c(x), a_n(y)) \to q_{n+1}(x, c(b(y)))$ et comme, par hypothèse de récurrence, $q_n(\widetilde{\overline{w'''_n \cdot c}}, \overline{a_n \cdot w''_n}) = q_n(c(\widetilde{\overline{w'''_n}}), a_n(\overline{w''_n})) \in H$, on doit avoir $q_{n+1}(\widetilde{\overline{w'''_n}}, \overline{c \cdot b \cdot w''_n}) \in H$. Or il s'agit exactement du codage de la configuration $\gamma_{n+1} = (w'''_n, q_{n+1}, cbw''_n)$. De plus, $q_{n+1} \notin \{\mathbf{accept}, \mathbf{reject}\}$ puisque $q_{n+1}(\widetilde{\overline{w'''_n}}, \overline{c \cdot b \cdot w''_n}) \in H$. et $H \models \forall x, y.\neg\mathbf{accept}(x, y) \in C(M, w) \wedge \forall x, y.\neg\mathbf{reject}(x, y)$.
- Le raisonnement est le même pour les deux autres types de mouvements de la machine, avec un petit cas supplémentaire quand on a un mouvement à droite et que w'_n est réduit à une lettre : dans ce cas $\gamma_n = (w_n, q_n, a_n)$ et $\gamma_{n+1} = (w_n b, q_{n+1}, B)$. On obtient que $q_{n+1}(\widetilde{\overline{w_n b}}, 0) \in H$ et, comme $H \models \forall x.(q(x, 0) \to q(x, B(0)))$, $q_{n+1}(\widetilde{\overline{w_n b}}, \overline{B}) \in H$.

Réciproquement, si la machine M ne s'arrête pas sur w, l'ensemble des codages des configurations successives de la machine M est un modèle de Herbrand de ϕ. Plus précisément, si $\gamma_0 \vdash_M^n \gamma_n = (w_n, q_n, w'_n)$, alors on considère $H = \{q_n(\widetilde{\overline{w_n}}, \overline{w'_n}) \mid n \in \mathbb{N}\} \cup \{q_n(\widetilde{\overline{w_n}}, 0) \mid n \in \mathbb{N}, w'_n = B\}$. $H \models \phi$ car $q_n \notin \{\mathbf{accept}, \mathbf{reject}\}$ et pour toute les clauses binaires $\phi_1 \to \phi_2$, $H \models \phi_1\sigma$ ssi $\phi_1\sigma$ est le codage d'une configuration γ_n, au quel cas $\phi_2\sigma$ est le codage de la configuration γ_{n+1} (sauf quand un blanc est ajouté à droite, mais ce cas ne pose pas de difficulté) et donc $\phi_2\sigma \in H$.

Exercice 107
Montrer que le théorème 3.5.2 reste vrai si on impose en plus que l'alphabet de symboles de fonction est vide.

Exercice 108 (5)
Montrer que le probème suivant est indécidable :

Donnée : Un ensemble fini de formules du premier ordre $\mathcal{S}$ et une formule ϕ

Question : $\mathcal{S} \models \phi$ ou bien $\mathcal{S} \models \neg\phi$.

Corollaire 3.5.1 *L'ensemble des formules satisfaisables n'est pas récursivement énumérable.*

Preuve:
Supposons que l'ensemble des formules satisfaisables soit récursivement énumérable. L'ensemble des formules non-valides l'est alors aussi (par passage d'une formule à sa négation), ce qui est en contradiction avec les théorèmes 3.4.8 et 3.5.1. □

3.6 Unification

Nous avons ramené la satisafaisabilité d'une formule (ou d'un ensemble de formules) du premier ordre à la satisfaisabilité d'un ensemble de clauses (paragraphe 3.3). Puis, grâce au théorème de Herbrand (théorème 3.4.1), un ensemble de clauses est satisfaisable si et seulement si il a un modèle de Herbrand. Et donc, un ensemble de clauses est satisfaisable si et seulement si l'ensemble de ses instances closes est satisfaisable, chaque littéral clos étant vu comme une variable propositionnelle. Nous nous sommes ainsi ramenés à la satisfaisabilité en calcul propositionnel. Cependant, l'ensemble de formules (et de variables proporitionnel) est a priori infini. Il nous faut donc maintenant un moyen effectif de manipuler des ensembles infinis de clauses qui sont les instances closes d'un ensemble de clauses avec variables.

Ceci nous amène à étudier la représentation suivante d'ensembles infinis de termes : si $t \in T(\mathcal{F}, \mathcal{X})$, on note $[\![t]\!]$ l'ensemble des termes de $T(\mathcal{F})$ qui sont des instances de t :

$$[\![t]\!] \stackrel{\text{def}}{=} \{t\sigma \in T(\mathcal{F}) \mid \sigma \in \Sigma\}$$

où Σ est l'ensemble des morphismes de $T(\mathcal{F}, \mathcal{X})$ dans $T(\mathcal{F})$ (substitutions closes).

L'ensemble des termes $T(\mathcal{F}, \mathcal{X})$ (ou plutôt l'ensemble des termes à similarité près)est ainsi muni d'une structure de semi-treillis : $t \leq u$ si $[\![u]\!] \subseteq [\![t]\!]$ (lire "t est plus général que u"). Le treillis possède un élément minimal et le sup de deux éléments est calculable : $[\![t]\!] \cap [\![u]\!]$ est un ensemble obtenu par *unification de t et u*. C'est cette propriété fondamentale que nous utiliserons pour *relever* résolution et factorisation au premier ordre.

Nous rappelons que pour une substitution σ, son *domaine* est défini comme $Dom(\sigma) := \{x \mid \sigma(x) \neq x\}$.

La substitution obtenue par l'application d'abord de σ puis de τ est notée $\sigma\tau$. Dans cette notation, on a donc que $t(\sigma\tau) = (t\sigma)\tau$ pour tout terme t et toutes substitutions σ et τ.

Une substitution σ est *idempotente* si $\sigma = \sigma\sigma$.

Definition 3.6.1 *Un* problème d'unification *est soit $\bot$, soit une conjonction d'équations $s_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge s_n \stackrel{?}{=} t_n$ avec $s_i, t_i \in T(\mathcal{F}, \mathcal{X})$. (Par convention, on note $\top$ ce problème lorsque $n = 0$).*

Une substitution σ est une solution *(aussi appelée* unificateur*) d'un problème d'unification $s_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge s_n \stackrel{?}{=} t_n$ si, pour tout i, $s_i\sigma = t_i\sigma$.*

Une substitution μ est un unificateur le plus général *(angl. : most general unifier), abrégé* mgu*, d'un problème d'unification $P = (s_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge s_n \stackrel{?}{=} t_n)$ si*

1. *μ est un unificateur de P ;*
2. *μ est idempotent ;*
3. *pour chaque unificateur τ de P il existe une substitution ν telle que $\tau = \mu\nu$*

Attention, il existe dans la littérature des définitions d'un mgu qui sont légèrement différentes mais équivalentes à la notre, mais aussi des définitions légèrement différentes et pas équivalentes à la notre. En l'occurrence, on trouve aussi une définition qui n'exige pas que le mgu soit idempotent.

Proposition 3.6.1 *Une substitution σ est idempotente ssi $Dom(\sigma) \cap VIm(\sigma) = \emptyset$.*

Preuve:
Soit $y \in Dom(\sigma) \cap VIm(\sigma)$. Il existe alors $x \in Dom(\sigma)$ tel que $y \in Var(x\sigma)$. Puisque $y\sigma \neq y$, on obtient alors que $x\sigma\sigma \neq x\sigma$, donc σ n'est pas idempotent.

Soit $Dom(\sigma) \cap VIm(\sigma) = \emptyset$. Si $x \notin Dom(\sigma)$ alors $x\sigma = x$ et donc $x\sigma\sigma = x$. Si $x \in Dom(\sigma)$, alors $x\sigma\sigma = x\sigma$ puisque $Dom(\sigma) \cap Var(x\sigma) = \emptyset$. □

Exemple 3.6.1

1. Le problème d'unification $x \stackrel{?}{=} g(y,z) \wedge f(x) \stackrel{?}{=} f(a)$ n'a pas d'unificateur.
2. Le problème d'unification $x \stackrel{?}{=} f(x)$ n'a pas d'unificateur.
3. Soit le problème d'unification $f(x) \stackrel{?}{=} f(g(y))$. Les substitutions suivantes sont des unificateurs :

$$\begin{aligned} \sigma_1 = &\quad \{x \to g(y)\} \\ \sigma_2 = &\quad \{x \to g(a); y \to a\} \end{aligned}$$

   σ_1 est un mgu, σ_2 ne l'est pas. En fait, $\sigma_2 = \sigma_1 \circ \{y \to a\}$, mais il n'y a aucune substitution ν telle que $\sigma_1 = \sigma_2 \circ \nu$.
4. Soit le problème d'unification

$$f(x,x) \stackrel{?}{=} f(x,y) \wedge g(x) \stackrel{?}{=} g(g(z)) \wedge f(z,a) \stackrel{?}{=} f(z',a)$$

   Les substitutions suivantes sont toutes des mgu :

$$\{x \to g(z'); y \to g(z'); z \to z'\}$$
$$\{x \to g(z); y \to g(z); z' \to z\}$$

5. Soit le problème d'unification $g(x) \stackrel{?}{=} g(g(z)) \wedge f(a,z) \stackrel{?}{=} f(a,y)$. Les substitutions suivantes sont des unificateurs :

$$\begin{aligned} \sigma_1 = &\quad \{x \to g(z); y \to z\} \\ \sigma_2 = &\quad \{x \to g(w); y \to w; z \to w; w \to y\} \end{aligned}$$

   σ_1 est un mgu, en fait $\sigma_2 = \sigma_1 \circ \{z \to w, w \to y\}$. On aussi que $\sigma_1 = \sigma_2 \circ \{w \to z; y \to w\}$, mais σ_2 n'est pas idempotent (parce que $w \in Dom(\sigma_2) \cap VIm(\sigma_2)$) donc pas un mgu.

Exercice 109 (4)
Soit μ un mgu d'un problème d'unification P. Montrer que, pour toute substitution σ :

$$\sigma \text{ est un unificateur de } P \quad \Leftrightarrow \quad \sigma = \mu\sigma$$

Exercice 110
Montrer que le mgu est unique à renommage près :

Soient μ et ν deux mgus d'un problème d'unification P. Montrer qu'il y a un *renommage* ρ tel que $\mu = \nu\rho$.

Un système de règles pour les problèmes d'unification est donné dans la Figure 3.3. Dans ces règles, ont suppose que le symbole $\wedge$ est associatif et commutatif. De plus, on suppose que le symbole d'égalité est commutatif, c'est-à-dire on identifie $s \stackrel{?}{=} t$ avec $t \stackrel{?}{=} s$. On note $P \rightarrow_U P'$ quand on peut transformer par une des règles de la Figure 3.3 un problème d'unification P dans un problème d'unification P'.

Proposition 3.6.2 *Les règles de la Figure 3.3 sont correctes : Si $P \rightarrow_U P'$ alors P et P' ont les mêmes unificateurs.*

On dit qu'une variable x est *résolue* dans un problème d'unification P quand P est de la forme $x \stackrel{?}{=} t \wedge P'$, $x \notin \mathrm{Var}(t)$ et $x \notin \mathrm{Var}(P')$.

Proposition 3.6.3 *Les règles de la Figure 3.3 terminent : Il n'y a pas de séquence infinie $P_1 \rightarrow_U \ldots \rightarrow_U P_n \rightarrow_U \ldots$*

Avant de commencer la preuve, rappelons les définitions des extensions lexicographique et multi-ensemble.

Si $E_1, \ldots, E_n$ sont des ensembles ordonnés par $\geq_1, \ldots, \geq_n$, la *composée lexicographique* de $\geq_1, \ldots, \geq_n$ est la relation d'ordre définie sur $E_1 \times \ldots \times E_n$ par :

$$(a_1, \ldots, a_n) >_{lex} (b_1, \ldots, b_n) \text{ ssi } \exists i. a_1 = b_1, ..., a_{i-1} = b_{i-1}, a_i >_i b_i$$

Un ordre est *bien fondé* s'il n'y a pas de suite infinite strictement décroissante.

Exercice 111
Montrer que la composée lexicographique de $\geq_1, \ldots, \geq_n$ est bien fondée si et seulement si chacun des ordres $\geq_1, \ldots, \geq_n$ est bien fondé.

Un *multi-ensemble (fini)* sur E est une application de E dans $\mathbb{N}$, qui est nulle sauf pour un nombre fini d'éléments de E. On note en outre $M_1 + M_2$ l'application $(M_1 + M_2)(x) \stackrel{\text{def}}{=} M_1(x) + M_2(x)$, $\emptyset$ l'application nulle, $x \in M$ si $M(x) \geq 1$ et enfin $M \setminus N$ l'application $(M \setminus N)(x) \stackrel{\text{def}}{=} \max(M(x) - N(x), 0)$. On note $\{\underbrace{x_1, \ldots x_1}_{n_1}, \ldots, \underbrace{x_k, \ldots, x_k}_{n_k}\}$ le multi-ensemble M tel que $M(x_1) = n_1, \ldots, M(x_k) = n_k$.

Si E est un ensemble ordonné par $\geq$, l'*extension multi-ensemble* de $\geq$ est la relation $\geq_m$ est la plus petite relation transitive telle que :

1. pour tout M, $M \geq_m \emptyset$

Decompose $\phi \wedge f(s_1, \ldots, s_n) \stackrel{?}{=} f(t_1, \ldots, t_n) \rightarrow_U \phi \wedge s_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge s_n \stackrel{?}{=} t_n$

Clash $\phi \wedge f(s_1, \ldots, s_n) \stackrel{?}{=} g(t_1, \ldots, t_m) \rightarrow_U \bot$
si $f \neq g$

Trivial $\phi \wedge s \stackrel{?}{=} s \rightarrow_U \phi$

Replace $\phi \wedge x \stackrel{?}{=} t \rightarrow_U \phi[x \mapsto t] \wedge x \stackrel{?}{=} t$
si $x \in \mathrm{Var}(\phi)$, $t \notin \mathcal{X}$ et $x \notin \mathrm{Var}(t)$

Occur Check $\phi \wedge x \stackrel{?}{=} t \rightarrow_U \bot$
si $x \in \mathrm{Var}(t)$ et $t \notin \mathcal{X}$

Var Replace $\phi \wedge x \stackrel{?}{=} y \rightarrow_U \phi\{x \mapsto y\} \wedge x \stackrel{?}{=} y$
Si $x \in \mathrm{Var}(\phi)$, $x \neq y$ et $y \in \mathrm{Var}(\phi)$

General OC $\phi \wedge x_1 \stackrel{?}{=} t_1[x_2]_{p_1} \wedge \ldots \wedge x_n = t_n[x_1]_{p_n} \rightarrow_U \bot$
si au moins l'un des p_i est non vide

Fusion $\phi \wedge x = s \wedge x = t \rightarrow_U \phi \wedge x = s \wedge s = t$
si $s \notin \mathcal{X}$ et $|s| \leq |t|$

FIGURE 3.3 – Règles d'unification.

2. si $M \geq_m M'$, alors $M + N \geq_m M' + N$
3. si $\forall y \in N$, $x > y$ et $M \geq_m M'$, alors $M + \{x\} >_m M' + N$

Exercice 112
Montrer que $\geq_m$ est antisymétrique et réflexive.

Exercice 113
Montrer que $M \geq_m M'$ si et seulement si, il existe N, M_1, M'_1 tels que
— $M = M_1 + N$ et $M' = M'_1 + N$
— $\forall x \in M'_1, \exists y \in M_1,\ y > x$

Exercice 114
Montrer que $\geq$ est bien fondée si et seulement si $\geq_m$ est bien fondée.

Revenons à la preuve du lemme de terminaison :

Preuve:
On considère les fonctions d'interprétation suivantes des problèmes d'unification :
— $\phi_1(P)$ est le nombre de variables non résolues de P.
— Si $P \equiv s_1 = t_1 \wedge \ldots \wedge s_n = t_n$, alors $\phi_2(P)$ est le multi-ensemble $\{m_1, \ldots, m_n\}$ où $m_i = \max(|s_i|, |t_i|)$.
— $\phi_3(P)$ est le nombre d'équations de P dont l'un des membres au moins est une variable.

$\phi(P)$ est le triplet $(\phi_1(P), \phi_2(P), \phi_3(P))$. L'ensemble de ces triplets est muni de la composée lexicographique de trois ordres :

1. L'ordre habituel sur les entiers naturels
2. L'extension multi-ensemble de l'ordre sur les entiers naturels
3. L'ordre habituel sur les entiers naturels

Il est ainsi muni d'un ordre bien fondé puisque construit à partir d'extensions multi-ensemble et lexicographiques d'ordres bien fondés.

Il suffit maintenant de montrer que si $P \rightarrow_U Q$ alors $\phi(P) > \phi(Q)$. Le sens de variations des trois fonctions d'interprétation est résumé dans le tableau ci-dessous :

	ϕ_1	ϕ_2	ϕ_3
Trivial	$\preceq$	$\prec$	
Decompose	$\preceq$	$\prec$	
Clash	$\preceq$	$\prec$	
Replace	$\prec$		
Var replace	$\prec$		
Fusion	$\preceq$	$=$	$\prec$
Occur check	$\preceq$	$\prec$	
General OC	$\preceq$	$\prec$	

Il n'est pas difficile de vérifier qu'aucune règle ne crée de nouvelle variable non-résolue. Les conditions d'application des deux règles d'élimination et de remplacement de variables garantissent par ailleurs qu'une variable non résolue

(x dans la formulation des règles doit apparaitre dans P) devient résolue après leur application.

Notons également que la règle de décomposition fait bien décroître ϕ_2 puisqu'elle remplace $m = \max(|f(s_1, \ldots, s_n)|, |f(t_1, \ldots, t_n)|)$ par n entiers strictement inférieurs.

La règle de fusion conserve bien ϕ_2 à cause de la condition $|s| \leq |t|$: $\max(|x|, |t|) = \max(|s|, |t|)$. Enfin, ϕ_3 est bien décroissante par fusion à cause de la condition $s \notin x$ qui, avec $|s| \leq |t|$, garantit que $s = t$ est une équation dont aucun des membres n'est une variable.

Bien entendu certaines fonctions d'interprétation (ϕ_2 ou ϕ_3) peuvent croître par application de certaines règles, mais la composée lexicographique des trois interprétations est toujours décroissante comme le montre le tableau ci-dessus.

□

Definition 3.6.2 *Un problème d'unification est en* arbre-forme résolue *si c'est* $\bot$, $\top$ *ou bien* $x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$ *où* $x_1, \ldots, x_n$ *sont des variables et n'apparaissent qu'une seule fois dans le problème.*

Exercice 115
Montrer qu'un problème d'unification en arbre-forme résolue comme ci-dessus a pour plus général unificateur $\{x_1 \mapsto t_1; \ldots; x_n \mapsto t_n\}$.

Proposition 3.6.4 *Tout problème d'unification qui ne peut être simplifié par les règles de la figure 3.3 est en arbre-forme résolue.*

Exercice 116
Prouver la proposition 3.6.4. Indiquer un sous-ensemble des règles qui est nécessaire et suffisant pour obtenir ce résultat.

Théorème 3.6.1 (Robinson) *Il y a un algorithme qui, étant donné un problème d'unification P,*

- *soit répond « non », et dans ce cas P n'a pas d'unificateur ;*
- *soit donne une substitution σ, et dans ce cas σ est un mgu de P.*

Preuve:
Étant donné un problème d'unification P, on applique les règles de la Figure 3.3 dans un ordre quelconque. Par la Proposition 3.6.3, on obtient en temps fini un problème d'unification P' sur lequel aucune règle d'unification s'applique. Par la Proposition 3.6.2, P et P' ont les mêmes unificateurs. Par la Proposition 3.6.4 il y a deux possibilités :

1. P' est $\bot$. Par Proposition 3.6.2, P n'a pas d'unificateur.
2. P' est de la forme $x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$, et les variables $x_1, \ldots, x_n$ sont éliminées en P. Dans ce cas, $\sigma = \{x_1 \to t_1; \ldots, x_n \to t_n\}$ est un mgu de P', donc par Proposition 3.6.2 aussi de P. □

□

Exercice 117
Calculer un mgu pour les problèmes d'unification suivants :

1. $f(x, g(a, y)) \stackrel{?}{=} f(h(y), g(y, a)) \wedge g(x, h(y)) \stackrel{?}{=} g(z, z))$
2. $f(x, x) \stackrel{?}{=} f(g(y), z) \wedge h(z) \stackrel{?}{=} h(y)$

Definition 3.6.3 *Un problème d'unification est en DAG-forme résolue si c'est* $\top, \bot$ *ou bien* $x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$ *où* $x_1, \ldots, x_n$ *sont des variables distinctes telles que,* $\forall i, \forall j \geq i, x_i \notin Var(t_j)$.

Exercice 118

1. Montrer qu'une DAG forme résolue comme ci-dessus définit un mgu $\{x_1 \mapsto t_1\} \cdots \{x_n \mapsto t_n\}$.
2. Montrer que, pour tout n, il est possible que $|t_1| + ... + |t_n| = O(n)$ alors que tout mgu $\{y_1 \mapsto u_1; \ldots; y_k \mapsto u_k\}$ est exponentiellement plus grand $(|u_1| + \ldots + |u_k| = O(2^n))$.

Exercice 119
Donner un sous-ensemble minimal de règles de la figure 3.3 tel que tout problème irréductible pour ces règles soit en DAG-forme résolue (justifier à la fois la minimalité et le fait que les règles soient suffisantes).

Definition 3.6.4 *Un* cycle de variables *est une conjonction d'équations* $x_1 \stackrel{?}{=} x_2 \wedge \ldots \wedge x_n \stackrel{?}{=} x_1$.

Un problème d'unification est en RT-forme résulue si c'est $\bot, \top$ *ou bien une conjonction* $x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$ *qui ne contient pas de cycle de variable et tel que* $x_1, \ldots, x_n$ *sont des variables distinctes.*

Exercice 120

1. Donner un sous-ensemble minimal de règles de la figure 3.3 qui garantisse que les problèmes irréductibles sont en RT-forme résolue.
2. Montrer que ces règles sont correctes si on interprète les problèmes d'unification dans l'algèbre des termes finis ou infinis.
3. Montrer que toute RT-forme résolue a une solution dans la F-algèbre des termes finis ou infinis.
4. Comment définir un mgu dans cette algèbre ? Pourquoi l'obtient on avec les règles de la figure 3.3 ?

Exercice 121 (6)
On appelle *anti-unificateur* de deux termes u et v, un terme t tel qu'il existe une substitution σ et il existe une substitution θ telles que $u = t\sigma$ et $v = t\theta$.

Montrer que toute paire de termes possède un moins général (ou plus spécialisé) anti-unificateur. Donner un algorithme qui permette de le calculer.

(Note : en fait, modulo similarité, le préordre d'instanciation muni l'algèbre des termes d'un structure de treillis. L'inf est calculé par anti-unification et le sup par unification).

3.7 Résolution en logique du premier ordre

3.7.1 Motivation

Reprenons les explications données au début du paragraphe 3.6.

Un premier essai de transférer la résolution propositionnelle vers la logique du premier ordre est comme suit : Étant donné un ensemble fini S de formules, on peut le transformer en un ensemble de clauses universelles S', et puis construire l'ensemble des ces instances de Herbrand $H(S')$. Par le Théorème 3.4.4 et la complétude réfutationelle de la résolution propositionnelle, S est non-satisfaisable si et seulement si $H(S') \vdash_R \square$. Malheureusement, cela ne nous donne pas directement une énumération récursive des ensembles finis non-satisfaisables de formules puisqu'il s'agit d'un ensemble infini de clauses, et il n'est pas clair comment, en général, un algorithme peut travailler sur un ensemble infini de clauses.

En utilisant le fait qu'il y a des stratégies terminantes et complètes de résolution propositionnelle on peut quand-même tirer de ce premier essai une énumération récursive des ensembles finis non-satisfaisables de formules : Étant donné S, on construit une énumération effective $s_1, s_2, \ldots$ des instances de Herbrand de S. Soit $\vdash_{RS}$ la déduction par résolution binaire et factorisation selon la stratégie complète et terminante. On lance l'algorithme suivant :

T := $\emptyset$;
n := 0 ;
while $\square \notin T$ **do begin**
  $T := T \cup \{ s_n \}$;
  $n := n + 1$;
  while there exists a $w \notin T$ **such that** $T \vdash_{RS} w$ **do**
    $T := T \cup \{ w \}$;
end ;
print "non satisfaisable" ;

Puisque la stratégie est terminante, la boucle intérieure termine toujours après un nombre fini d'itérations. On dit que la boucle intérieure calcule une *saturation* par $\vdash_{RS}$. En un sens, cet algorithme est déjà un peu plus « intelligent » que l'algorithme présenté à la fin de la section précédente puisque, à chaque itération de la boucle principale, on réutilise le travail fait pendant les itérations précédentes.

La clef pour faire mieux est l'observation que, étant donnée une clause universelle U, son ensemble d'instances de Herbrand est un ensemble très régulier. L'idée est donc de chercher une représentation finie de cet ensemble infini de clauses. La représentation qui vient à l'esprit est d'utiliser une clause universelle en S' comme une représentation de l'ensemble infini $H(S')$. Puis, étant données deux clauses universelles C_1 et C_2, on cherche à exécuter d'un seul coup un pas de résolution sur toutes les instances de Herbrand de C_1 et toutes les instances de Herbrand de C_2, et de représenter l'ensemble résultant de clauses par une clause du premier ordre.

Pour illustrer cette idée, supposons données deux clauses du premier ordre $C_1 = \forall \bar{x}(P(s) \vee L)$ et $C_2 = \forall \bar{y}(\neg P(t) \vee K)$. On cherche une clause du premier ordre R, telle que les instances de Herbrand de R sont exactement les clauses propositionnelles qu'on peut obtenir par résolution propositionnelle sur P entre une instance de Herbrand de C_1 et une instance de Herbrand de C_2.

On a donc

$$\begin{aligned} H(C_1) &= \{(P(s) \vee L)\sigma \mid \sigma \in \Sigma_g\} \\ H(C_2) &= \{(\neg P(t) \vee K)\tau \mid \tau \in \Sigma_g\} \end{aligned}$$

Un pas de résolution est possible ssi il existe des substitutions σ et τ telles que $s\sigma = t\tau$. Comme C_1 et C_2 peuvent être supposées ne pas partager de variables (après éventuel renommage des variables liées), $s\sigma = t\tau$ revient à dire que s et t sont unifiables.

L'ensemble des clauses closes qu'on peut obtenir par un pas de résolution est donc

$$\{L\sigma \vee K\tau \mid \sigma, \tau \in \Sigma_g, s\sigma = t\tau\} \tag{3.1}$$

Quand $\bar{x} \cap \bar{y} = \emptyset$, cet ensemble est le même que

$$\{(L \vee K)\nu \mid \nu \in \Sigma_g, s\nu = t\nu\} \tag{3.2}$$

Quand $\bar{x} \cap \bar{y} = \emptyset$ on peut supposer que $Dom(\sigma) \cap Dom(\tau) = \emptyset$. Les ensembles (3.1) et (3.2) sont alors égaux parce que

— étant donnée σ et τ en (3.1), on peut définir ν en (3.2) par $\nu = \sigma \circ \tau$ $(= \tau \circ \sigma)$;

— étant donnée τ in (3.2), on peut définir en (3.2) $\sigma = \nu \mid_{\bar{x}}$ et $\tau = \nu \mid_{\bar{y}}$.

Par contre, cette égalité est en général fausse quand $\bar{x}$ et $\bar{y}$ ne sont pas disjoints. Par exemple, les termes x et $f(x)$ ont des instances communes (par exemple, $f(a)$), mais il n'y a aucune substitution ν telle que $x\nu = f(x)\nu$.

Si μ est un mgu de s et t, alors l'ensemble (3.2) est donc le même que

$$\{((L \vee K)\mu)\nu \mid \nu \in \Sigma_g\} \tag{3.3}$$

Le raisonnement pour la factorisation est analogue.

3.7.2 Résolution

On obtient donc les règles de résolution données sur la Figure 3.4. Dans les preuves par résolution, on n'écrit normalement pas les quantificateurs universels. Pourtant, il ne faut pas oublier que chaque clause est implicitement universellement quantifiée. Une conséquence importante est qu'on doit toujours assurer, avant d'appliquer un pas de résolution à deux clauses, que leur ensembles de variables soient disjoints. Si ce n'est pas le cas on est obligé de renommer les variables dans une des clauses. Finalement, si S est un ensemble de clauses du premier ordre, on note $S \vdash_{R1} C$ quand la clause C est obtenue à partir de S par une application des règles de la Figure 3.4.

Lemme 3.7.1 *Les règles de la résolution du premier ordre sont correctes : Si $S \vdash_{R1} C$ alors C est une conséquence logique de S.*