4.4 Transformations de formules

4.4.1 Renommages et formules polies

Nous montrons ici que toute formule est équivalente à une formule polie.

Proposition 4.37. *Si $y \notin Var(\varphi)$ (une variable n'apparaissant pas dans φ, alors $\exists x\varphi \equiv \exists y(\varphi_{\{x\leftarrow y\}})$ où $\varphi_{\{x\rightarrow y\}}$ est obtenu en remplaçant toutes les occurrences de x par y.*

Démonstration. Pour toute structure $\mathcal{M}$ et toute valuation $\mathcal{V}$:

$$\begin{array}{lll} \mathcal{M}, \mathcal{V} \models \exists x\varphi & \text{ssi} & \text{il existe } d \in D_{\mathcal{M}},\ \mathcal{M}, \mathcal{V}[x := d] \models \varphi \\ & \text{ssi} & \text{il existe } d \in D_{\mathcal{M}},\ \mathcal{M}, \mathcal{V}[y := d] \models \varphi_{\{x\leftarrow y\}} \\ & \text{ssi} & \mathcal{M}, \mathcal{V} \models \exists y\varphi_{\{x\leftarrow y\}}. \end{array}$$

□

De la même façon, on peut prouver un résultat similaire pour la quantification universelle :

Proposition 4.38. *Si $y \notin Var(\varphi)$ (une variable n'apparaissant pas dans φ, alors $\forall x\varphi \equiv \forall y(\varphi_{\{x\leftarrow y\}})$ où $\varphi_{\{x\rightarrow y\}}$ est obtenu en remplaçant toutes les occurrences de x par y.*

Proposition 4.39 (Loi de **réalphabétisation** (renommage) des variables.)**.** *Si $y \notin Var(\varphi)$, alors la formule obtenue choisissant un quantificateur Qx dans la formule et en renommant toutes les occurrences de x qui lui sont liées par y (et en remplaçant Qx par Qy) est équivalente à φ.*

Par exemple, dans $\exists x(\forall xF(x, y) \Rightarrow (G(x) \vee q))$, on peut opérer deux renommages : on peut renommer d'abord l'occurrence de x dans $F(x, y)$ à t, en obtenant ainsi $\exists x(\forall tF(t, y) \Rightarrow (G(x) \vee q))$, et ensuite renommer le x restant à z ; on obtient $\exists z(\forall tF(t, y) \Rightarrow (G(z) \vee q))$.

Grâce à ces propositions, on pourra supposer désormais, sans perte de généralité, que toute formule est polie.

Proposition 4.40. *Pour toute formule φ, il existe réalphabétisation de φ donnant une formule polie ψ telle que $\varphi \equiv \psi$.*

4.4.2 Equivalences classiques

Les équivalences données dans le cadre du calcul propositionnel restent vraies. Nous en donnons d'autres ici, les preuves sont laissées en exercice.

— Lois de **conversion** des quantificateurs :

$$\neg\forall x\varphi \equiv \exists x\neg\varphi\,, \qquad \neg\exists x\varphi \equiv \forall x\neg\varphi\,.$$

— Lois de **distribution** des quantificateurs :

$$\forall x(\varphi \wedge \psi) \equiv (\forall x\varphi \wedge \forall x\psi)\,, \qquad \exists x(\varphi \vee \psi) \equiv (\exists x\varphi \vee \exists x\psi)\,.$$

— Lois de **permutation** des quantificateurs de même sorte :

$$\forall x\forall y\varphi \equiv \forall y\forall x\varphi\,, \qquad \exists x\exists y\varphi \equiv \exists y\exists x\varphi\,.$$

— Lois de **passage** : *si x n'est pas libre dans ψ*, on a les lois suivantes :

$$\begin{array}{ll} \forall x(\varphi \wedge \psi) \equiv (\forall x\varphi) \wedge \psi & \exists x(\varphi \wedge \psi) \equiv (\exists x\varphi) \wedge \psi \\ \forall x(\varphi \vee \psi) \equiv (\forall x\varphi) \vee \psi & \exists x(\varphi \vee \psi) \equiv (\exists x\varphi) \vee \psi \\ \forall x(\varphi \Rightarrow \psi) \equiv (\exists x\varphi) \Rightarrow \psi & \exists x(\varphi \Rightarrow \psi) \equiv (\forall x\varphi) \Rightarrow \psi \\ \forall x(\psi \Rightarrow \varphi) \equiv \psi \Rightarrow (\forall x\varphi) & \exists x(\psi \Rightarrow \varphi) \equiv \psi \Rightarrow (\exists x\varphi) \end{array}$$

Remarquez le changement de quantificateur dans les équivalences $\forall x(\varphi \Rightarrow \psi) \equiv (\exists x\varphi) \Rightarrow \psi$ et $\exists x(\varphi \Rightarrow \psi) \equiv (\forall x\varphi) \Rightarrow \psi$.

Notez également que dans chacune des 8 équivalences, si le membre droit est poli, alors la condition *si x n'est pas libre dans ψ* est vérifiée.

Exemple 4.41. Nous pouvons démontrer l'équivalence entre les formules $\neg\forall x\varphi(x)$ et $\exists x\neg\varphi(x)$ de la façon suivante. Soient $\mathcal{M}$ une $\mathcal{S}$-structure et $\mathcal{V}$ une valuation fixés.

(a) Supposons que $[\![\,\exists x\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$ et montrons que $[\![\,\neg\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$. De $\max_{a\in D_\mathcal{M}} [\![\,\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 1$, nous déduisons que $[\![\,\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 1$ pour un quelque $a \in D_\mathcal{M}$, donc $[\![\,\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 0$ et $[\![\,\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = \min_{a\in D_\mathcal{M}} [\![\,\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 0$, donc $[\![\,\neg\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$.

(b) Supposons, par contre, que $[\![\,\neg\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$ et montrons que $[\![\,\exists x\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$. On a bien que $[\![\,\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 0$; depuis

$$0 = [\![\,\forall x\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = \min_{a\in D_\mathcal{M}} [\![\,\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 0$$

nous déduisons que ce minimum est réalisé : donc $[\![\,\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 0$ pour un quelque $a \in D_\mathcal{M}$, d'où $[\![\,\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}[x:=a]} = 1$, et $[\![\,\exists x\neg\varphi\,]\!]_{\mathcal{M},\mathcal{V}} = 1$.

Exercice 4.42. Montrez que les formules $\forall x(\varphi \vee \psi)$ et $\forall x\varphi \vee \forall x\psi$ ne sont pas, en général, équivalentes. Argumentez de façon similaire pour $\exists x(\varphi \wedge \psi)$ et $\exists x\varphi \wedge \exists x\psi$.

4.4.3 Forme clausale

Pour la logique propositionnelle, nous avons vu que toute formule était équivalente à une formule sous forme clausale. Ce résultat était très utilisé puisque les formules propositionnelles sous formes clausales sont plus adaptées à la recherche de modèle.

Nous allons introduire ici la forme clausale du premier ordre. Si comme la forme propositionnelle, elle est plus adaptée à la recherche de modèle, il y a une différence importante : en générale la forme clausale φ_c d'une formule φ ne lui sera pas équivalente, mais equisatisfaisable : φ est satisfaisable sssi φ_c est satisfaisable.

Ainsi, même si les formules ne sont pas équivalente, prouver que φ est satisfaisable est équivalent à prouver que φ_c est satisfaisable.

Nous décrivons, à présent, les différentes étapes qui mènent à une représentation sous forme clausale.

4.4.3.1 Forme prénexe

Définition 4.43 (Forme prénexe)**.** Une formule φ est sous **forme prénexe** lorsqu'elle a la forme :

$$Q_1x_1Q_2x_2\ldots Q_nx_n\psi$$

où chaque Q_i est un quantificateur (existentiel ou universel) et ψ est sans quantificateurs. La partie $Q_1x_1Q_2x_2\ldots Q_nx_n$ est appelée le **préfixe** et ψ étant qualifiée de **matrice**.

Mettre une formule sous forme prénexe consiste donc à renvoyer tous les quantificateurs au début de la formule.

Théorème 4.44 (Forme prénexe équivalente)**.** *Pour toute formule $\varphi \in \mathcal{F}_{po}$ il existe une formule équivalente $\psi \in \mathcal{F}_{po}$ en forme prénexe.*

L'algorithme de mise sous forme prénexe suit les étapes suivantes :

Etape 1 : Renommer les variables pour obtenir une formule polie équivalente

Etape 2 : Appliquer tant que possible les substitutions suivantes : substitution du membre droit par le membre gauche pour toutes les lois de passage et de conversion des quantificateurs.

Exemple 4.45. Considérez la formule suivante :

$$\neg\exists x\forall yR(x,y) \wedge \forall x(\exists yR(x,y) \Rightarrow R(x,x))$$

La première étape, renommage des variables, donne la formule suivante :

$$\neg\exists z_0\forall z_1R(z_0,z_1) \wedge \forall z_2(\exists z_3R(z_2,z_3) \Rightarrow R(z_2,z_2))\,.$$

Nous appliquons ensuite la deuxième étape :

$$
\begin{aligned}
&\neg\exists z_0\forall z_1 R(z_0,z_1)\wedge\forall z_2(\exists z_3 R(z_2,z_3)\Rightarrow R(z_2,z_2))\\
&\quad\rightsquigarrow\quad \forall z_0\neg\forall z_1 R(z_0,z_1)\wedge\forall z_2(\exists z_3 R(z_2,z_3)\Rightarrow R(z_2,z_2))\\
&\quad\rightsquigarrow\quad \underline{\forall z_0}\,\underbrace{\exists z_1\neg R(z_0,z_1)}_{\varphi}\wedge\underbrace{\forall z_2(\exists z_3 R(z_2,z_3)\Rightarrow R(z_2,z_2))}_{\psi}\\
&\quad\rightsquigarrow\quad \forall z_0(\underline{\exists z_1}\,\underbrace{\neg R(z_0,z_1)}_{\varphi}\wedge\underbrace{\forall z_2(\exists z_3 R(z_2,z_3)\Rightarrow R(z_2,z_2))}_{\psi})\\
&\quad\rightsquigarrow\quad \forall z_0\exists z_1(\underbrace{\neg R(z_0,z_1)}_{\psi}\wedge\underline{\forall z_2}(\underbrace{\exists z_3 R(z_2,z_3)\Rightarrow R(z_2,z_2)}_{\varphi}))\\
&\quad\rightsquigarrow\quad \forall z_0\exists z_1\forall z_2(\neg R(z_0,z_1)\wedge(\underline{\exists z_3}\,\underbrace{R(z_2,z_3)}_{\varphi}\Rightarrow\underbrace{R(z_2,z_2)}_{\psi}))\\
&\quad\rightsquigarrow\quad \forall z_0\exists z_1\forall z_2(\underbrace{\neg R(z_0,z_1)}_{\psi}\wedge\underline{\forall z_3}(\underbrace{R(z_2,z_3)\Rightarrow R(z_2,z_2)}_{\varphi}))\\
&\quad\rightsquigarrow\quad \forall z_0\exists z_1\forall z_2\forall z_3(\neg R(z_0,z_1)\wedge(R(z_2,z_3)\Rightarrow R(z_2,z_2)))\,.
\end{aligned}
$$

Notez que c'est le fait d'avoir rendu la formule polie qui permet d'appliquer les lois de passage. Notez également qu'il y avait d'autres ordres possibles pour l'application des lois de passage, qui auraient abouti à des formules différentes (mais bien entendu équivalentes)

4.4.3.2 Forme de Skolem

Définition 4.46. Une formule est sous **forme de Skolem** lorsqu'elle est sous forme prenexe et qu'elle ne contient que des quantifications universelles.

Exemple 4.47. La formule $\forall x R(x,f(x))$ est sous forme de Skolem. La formule $\forall x\exists y R(x,y)$ est en forme prénexe, mais elle n'est pas sous forme de Skolem, car on trouve un quantificateur existentiel dans le préfixe.

Pour effectuer une *skolémisation*, on part donc d'une formule sous forme prenexe et on « supprime » les quantificateurs existentiels, en appliquant tant que possible la règle suivante :

Règle de Skolémisation.
Remplacer la formule $\forall x_1\cdots\forall x_n\exists y\psi$ par la formule $\forall x_1\cdots\forall x_n\psi_{\{y\leftarrow f(z_1,\ldots,z_m)\}}$, où
— $z_1,...,z_m$ sont les variables libres de la formule $\exists y\,\psi$, et
— f est un *nouveau* symbole de fonction (dit de Skolem) d'arité m,
— $\psi_{\{y\leftarrow f(z_1,\ldots,z_m)\}}$ est obtenue en remplaçant chaque occurrence de y par le terme $f(z_1,\ldots,z_m)$.

Définition 4.48. Une formule universelle (c'est-à-dire une formule contenant seulement des quantificateurs universels), obtenue de φ par mise en forme prenexe, puis application itérée de la règle de skolémisation est appelée **forme de skolem** ou **skolémisée** de φ.

Exemple 4.49.

1. Soit $\varphi=\forall x\exists y P(x,y)\Rightarrow\forall x\exists y P(y,x)$.
   La mise sous forme prénexe donne la formule équivalente $\varphi'=\exists x\forall y\forall x'\exists y'(P(x,y)\Rightarrow P(y',x'))$.
   La skolémisation se fait en 2 applications de la règle :

$$
\begin{array}{c|l}
\exists x\forall y\forall x'\exists y'(P(x,y)\Rightarrow P(y',x')) & \\ \hline
\forall y\forall x'\exists y'(P(c,y)\Rightarrow P(y',x')) & \text{substitution } x\leftarrow c \\ \hline
\forall x\forall x'(P(c,y)\Rightarrow P(f(y,x'),x'))) & \text{substitution } y'\leftarrow f(y,x')
\end{array}
$$

2. Considérons la formule $\exists x_1\forall x_2\forall x_3\exists x_4\forall x_5\exists x_6 P(x_1,x_2,x_3,x_4,x_5,x_6)$. On obtient sa formule de Skolem de la façon suivante :

$$\cfrac{\cfrac{\cfrac{\exists x_1 \forall x_2 \forall x_3 \exists x_4 \forall x_5 \exists x_6 P(x_1, x_2, x_3, x_4, x_5, x_6)}{\forall x_2 \forall x_3 \exists x_4 \forall x_5 \exists x_6 P(f_1, x_2, x_3, x_4, x_5, x_6)} \text{ substitution } x_1 \leftarrow f_1}{\forall x_2 \forall x_3 \forall x_5 \exists x_6 P(f_1, x_2, x_3, f_4(x_2, x_3), x_5, x_6)} \text{ substitution } x_4 \leftarrow f_4(x_2, x_3)}{\forall x_2 \forall x_3 \forall x_5 P(f_1, x_2, x_3, f_4(x_2, x_3), x_5, f_6(x_2, x_3, x_5))} \text{ substitution } x_6 \leftarrow f_6(x_2, x_3, x_5)$$

Dans la formule précédente, $\exists x_1$ n'est précédé par aucun quantificateur universel. C'est pourquoi on introduit une nouvelle constante f_1.

Remarque 4.50. La version skolémisée d'une formule *ne lui est **pas***, en général, *équivalente*. Le langage étant étendu, donc différent, les modèles des deux formules sont des structures pour des langages différents. Néanmoins :

— tout modèle de la formule skolémisée est modèle de la formule initiale ;

— tout modèle de la formule initiale peut s'étendre en un modèle de la formule skolémisée, obtenu en conservant les interprétations des symboles de la signature initiale, et en interprétant correctement les nouveaux symboles de fonction introduits pas la skolemisation ;

— une formule close et sa forme de Skolem sont dites *équisatisfaisables* : si l'une possède un modèle, l'autre également et réciproquement.

Proposition 4.51. *Si φ_s est obtenue par skolémisation à partir de φ alors φ_s est satisfaisable si et seulement si φ est satisfaisable.*

Le deux exemples suivants permettent d'expliquer cette proposition :

Exemple 4.52. Considérons une formule close $\varphi = \exists x \psi$ (on rappelle qu'une formule est *close* si elle ne contient pas de variables libres) écrite sur un langage $\mathcal{S}$. La règle de skolémisation donne la formule $\psi_{\{x \leftarrow c\}}$, où c est un nouveau symbole de constante. Les deux formules ne sont pas équivalentes, puisqu'elles ne sont pas définies sur le même langage : la formule $\psi_{\{x \leftarrow c\}}$ est définie sur $\mathcal{S}' := \mathcal{S} \cup \{ (c, 0) \}$ (où c est le nouveau symbole de constante). Nous allons toutefois vérifier que ces deux formules sont équisatisfaisables. Etant donnée une $\mathcal{S}$-structure $\mathcal{M}$, et $a \in D_{\mathcal{M}}$, nous définissons la $\mathcal{S}'$-structure $\mathcal{M}'_a$ obtenue depuis $\mathcal{M}$ en interprétant la constante c par $c^{\mathcal{M}'_a} := a$.

On a :

$$\begin{array}{rll} \mathcal{M} \models \varphi & \text{ssi} & \text{il existe } a \in D_{\mathcal{M}} \text{ tel que } \mathcal{M}, [x := a] \models \psi \\ & \text{ssi} & \text{il existe } a \in D_{\mathcal{M}} \text{ tel que } \mathcal{M}'_a \models \psi_{\{x \leftarrow c\}}. \end{array}$$

Donc φ est satisfaisable ssi $\psi_{\{x \leftarrow c\}}$ est satisfaisable.

Exemple 4.53. Considérons maintenant la formule $\varphi = \forall y \exists x \psi$ écrite sur un langage $\mathcal{S}$. L'application de la règle de skolémisation donne $\varphi_s = \forall y \psi_{\{x \leftarrow f(y)\}}$.

Encore une fois, φ et φ_s ne sont pas équivalentes puisque φ_s est définie sur le langage $\mathcal{S}' = \mathcal{S} \cup \{(f, 1)\}$.

Etant donné une $\mathcal{S}$-structure $\mathcal{M}$, pour toute fonction $\tau : D_{\mathcal{M}} \to D_{\mathcal{M}}$, nous définissons la $\mathcal{S}'$-structure $\mathcal{M}'_\tau$ obtenue depuis $\mathcal{M}$ en interprétant la fonction f par $f^{\mathcal{M}'_\tau} := \tau$. On a :

$$\begin{array}{rll} \mathcal{M} \models \varphi & \text{ssi} & \text{pour tout } a \in D_M, \text{ il existe } b \in D_M \text{ tel que } \mathcal{M}, [y := a, x := b] \models \psi \\ & \text{ssi} & \text{pour tout } a \in D_M, \text{ il existe } b \in D_{\mathcal{M}} \text{ tel que } \mathcal{M}, [y := a] \models \psi_{\{x \leftarrow b\}} \\ & \text{ssi} & \text{il existe une fonction } \tau : D_{\mathcal{M}} \to D_{\mathcal{M}} \text{ telle que pour tout } a \in D_M,\ \mathcal{M}, [y := a] \models \psi_{\{x \leftarrow \tau(a)\}} \\ & \text{ssi} & \text{il existe une fonction } \tau : D_{\mathcal{M}} \to D_{\mathcal{M}} \text{ telle que } \mathcal{M}'_\tau \models \forall y \psi_{\{x \leftarrow f(y)\}}. \end{array}$$

Encore une fois, on a donc que φ est satisfaisable ssi φ_s est satisfaisable. De plus, on peut construire un modèle de φ à partir d'un modèle de φ_s.

4.4.3.3 Forme clausale

Définition 4.54 (Forme clausale)**.** Une formule close est sous **forme clausale** si elle est

1. en forme prenexe,
2. elle est universelle (tous ses quantificateurs sont universels), et

3. sa matrice est sous forme normale conjonctive.

En utilisant la mise en forme prénexe, puis la Skolemisation, puis la mise en forme clausale du calcul propositionnel, toute formule peut se transformer dans une formule en forme clausale equisatisfiable.

Définition 4.55.
- Un **littéral** est une formule atomique ou la négation d'une formule atomique.
- Soit ψ une formule dont les variables libres sont $\{ x_1, \ldots, x_n \}$. Sa **fermeture universelle** est la formule
$$\forall x_1 \ldots \forall x_n \psi\,.$$
- Une **clause universelle** ou **clause de premier ordre** est la fermeture universelle d'une disjonction de littéraux.

Lorsqu'une formule est sous forme clausale, on peut ensuite la décomposer en une conjonction de clauses (du premier ordre) en appliquant la règle $\forall x(\varphi \wedge \psi) \equiv (\forall x\varphi) \wedge (\forall x\psi)$. On obtient ainsi un ensemble de clauses (de premier ordre) équisatisfiable à la formule donnée.

Théorème 4.56 (Satisfiabilité d'un ensemble de clauses)**.** *Soit S un ensemble de clauses résultant de la mise sous forme clausale d'une formule φ. Alors φ est satisfiable si et seulement si S est satisfiable.*

Ce théorème forme la base de nombreux démonstrateurs automatiques utilisant une représentation des formules sous forme de clauses. Il établit que la recherche de l'insatisfiabilité d'une formule φ est équivalente à la recherche d'insatisfiabilité de sa représentation sous forme clausale S. Cependant φ et S ne sont pas logiquement équivalentes : seule la satisfiabilité est préservée.

Exemple 4.57. Soit
$$\varphi_0 := \forall x \exists y P(x,y) \Rightarrow \forall x \exists y (P(y,x) \wedge Q(x))\,.$$
La mise sous forme prénexe donne la formule
$$\varphi_1 := \exists x \forall z \exists w \forall y (P(x,y) \Rightarrow (P(w,z) \wedge Q(z)))\,.$$
Par skolémisation, nous obtenons
$$\varphi_2 := \forall z \forall y (P(c,y) \Rightarrow (P(f(z),z) \wedge Q(z)))\,.$$
Nous pouvons ensuite mettre la matrice sous forme normale conjonctive :
$$\varphi_3 := \forall y \forall z ((\neg P(c,y) \vee P(f(z),z)) \wedge (\neg P(c,y) \vee Q(z)))\,,$$
et obtenir ainsi l'ensemble de clauses :
$$S := \{\ \forall y \forall z (\neg P(c,y) \vee P(f(z),z))\,, \forall y \forall z (\neg P(c,y) \vee Q(z))\ \}\,.$$
Cet ensemble est équisatisfiable avec la formule φ_0.