$$
\begin{array}{rcl}
\phi \to \psi & \Rightarrow & (\neg\phi) \vee \psi \\
\neg\neg\phi & \Rightarrow & \phi \\
\neg(\phi \wedge \psi) & \Rightarrow & (\neg\phi) \vee (\neg\psi) \\
\neg(\phi \vee \psi) & \Rightarrow & (\neg\phi) \wedge (\neg\psi) \\
(\phi \wedge \psi) \vee \theta & \Rightarrow & (\phi \vee \theta) \wedge (\psi \vee \theta) \\
\theta \vee (\phi \wedge \psi) & \Rightarrow & (\theta \vee \phi) \wedge (\theta \vee \psi) \\
\top \vee \phi & \Rightarrow & \top \\
\phi \vee \top & \Rightarrow & \top \\
\phi \vee \perp & \Rightarrow & \phi \\
\perp \vee \phi & \Rightarrow & \phi \\
\phi \vee \phi & \Rightarrow & \phi
\end{array}
$$

FIG. 2.2 – Règles de mise en forme clausale

2.3 Forme clausale

On considère les règles de simplification de la figure 2.2.

Proposition 2.3.1 *Les règles de la figure 2.2 transforment des formules en des formules logiquement équivalentes.*

Proposition 2.3.2 *Les règles de la figure 2.2 se terminent (quel que soit l'ordre dans lequel elles sont appliquées).*

Preuve:
On interprète comme suit les formules dans les entiers :

- $f(\perp) = f(\top) \stackrel{\text{def}}{=} 2$
- $f(P) \stackrel{\text{def}}{=} 2$ si P est une variable propositionnelle.
- $f(\phi \wedge \psi) \stackrel{\text{def}}{=} g_1(f(\phi), f(\psi))$ où $g_1(x, y) \stackrel{\text{def}}{=} x + y + 1$
- $f(\phi \vee \psi) \stackrel{\text{def}}{=} g_2(f(\phi), f(\psi))$ où $g_2(x, y) \stackrel{\text{def}}{=} x \times y$
- $f(\neg\phi) \stackrel{\text{def}}{=} g_3(f(\phi))$ où $g_3(x) \stackrel{\text{def}}{=} 2^x + 1$
- $f(\phi \to \psi) \stackrel{\text{def}}{=} g_4(f(\phi), f(\psi))$ où $g_4(x, y) \stackrel{\text{def}}{=} 2^{1+x+y}$

On montre d'abord par récurrence sur ϕ que $f(\phi) \geq 2$. Ensuite, toutes les fonctions g_i sont strictement croissantes dans leurs deux arguments pour $x, y \geq 2$. Il en résulte que, si $f(\phi) > f(\psi)$, alors $f(C[\phi]) > f(C[\psi])$ pour tout contexte C.

Il suffit ainsi de démontrer que, pour toutes les formules ϕ, ψ, chacune des règles fait décroître f. Pour plus de clarté, on notera $x = f(\phi)$, $y = f(\psi)$, $z = f(\theta)$ dans ce qui suit.

- $f(\phi \to \psi) = 2^{1+x+y}$ et $f(\neg\phi \vee \psi) = (2^x + 1) \times y$. Mais, pour $y \geq 0$, $2^y > y$ et, pour $x \geq 1$, $2^{1+x} > 2^x + 1$ donc $2^{1+x+y} > (2^x + 1) \times y$.
- $f(\neg\neg\phi) = 2^{2^x+1} + 1 > x$
- $f(\neg(\phi \wedge \psi)) = 2^{x+y+1} + 1$ et $f(\neg\phi \vee \neg\psi) = (2^x + 1) \times (2^y + 1) = 2^{x+y} + 2^x + 2^y + 1$. Or, pour $x, y \geq 1$, $2^{x+y} > 2^x + 2^y + 1$ (par convexité de l'exponentiation), donc $2^{x+y+1} + 1 = 2^{x+y} + 2^{x+y} + 1 > (2^x + 1)(2^y + 1)$

$$\begin{array}{rcl} \phi \wedge \phi & \to & \phi \\ \phi \wedge \bot & \to & \bot \\ \phi \wedge \top & \to & \phi \\ \top \wedge \phi & \to & \phi \end{array}$$

FIG. 2.3 – Simplification des conjonctions de clauses

- $f(\neg(\phi \vee \psi)) = 2^{x \times y} + 1$ et $f(\neg\phi \wedge \neg\psi) = 2^x + 1 + 2^y + 1 + 1 = 2^x + 2^y + 3$. Or $2^{xy} \geq 2^x \times 2^y + 2^x \times 2^y$ pour $x, y \geq 1$ et $2^x \times 2^y \geq 2^x + 2^y$ pour $x, y \geq 1$ et $2^x \times 2^y \geq 3$ pour $x, y \geq 1$.
- $f((\phi \wedge \psi) \vee \theta) = f(\theta \vee (\phi \wedge \psi)) = (x + y + 1) \times z$ et $f((\phi \vee \theta) \wedge (\psi \vee \theta)) = x \times z + y \times z + 1$ qui est strictement inférieur à $x \times z + y \times z + z$ pour $z \geq 2$.
-

□

Definition 2.3.1 *Un* littéral *est une variable propositionnelle ou la négation d'une variable propositionnelle.*

Une clause *est une disjonction de littéraux ou bien* $\bot$.

Proposition 2.3.3 *Toute formule irréductible pour les règles de la figure 2.2 est une conjonctions de clauses ou bien* $\top$.

Definition 2.3.2 *Une* forme clausale *d'une formule* $\phi \in \mathcal{F}_0(\mathcal{P})$ *est la forme normale par les règles de la figure 2.3 d'une des formes irréductibles de* ϕ *pour les règles de la figure 2.2.*

Exercice 20 (3)
Donner un exemple de formule ayant deux formes clausales distinctes.

L'exercice suivant montre que les formes clausales peuvent inévitablement conduire à une croissance exponentielle de la formule.

Exercice 21 (6)
Si $\phi \in \mathcal{F}_0(\mathcal{P})$, on note $\tau(\phi)$ la taille minimale d'une forme clausale de ϕ.

Donner un exemple d'une famille de formules ϕ_n telles que $\lim_{n \to +\infty} |\phi_n| = +\infty$ et $\lim_{n \to +\infty} \frac{\tau(\phi_n)}{2^{|\phi_n|}} > 0$.

$$\text{Résolution binaire} \quad \frac{\neg A \vee C \quad A \vee C'}{C \vee C'}$$

$$\text{Factorisation binaire} \quad \frac{L \vee L \vee C}{L \vee C}$$

FIG. 2.4 – Règle de résolution

2.4 Résolution

Dans cette partie, on ne considère que des formes clausales.

Décider de la satisfaisabilité d'une formule en forme clausale n'est pas (algorithmiquement) facile. C'est un problème NP-complet (voir calculabilité).

Les règles d'inférence de la figure 2.4 ont en prémisse une ou deux clauses et en conclusion une clause. Dans ces règles, C est ou bien une disjonction de littéraux, ou bien la clause vide $\perp$ (on suppose que $C \vee \perp = C$) et L est un littéral.

On note $E \vdash_R C$ lorsque la clause C peut être obtenue à partir de E. par une application d'un nombre quelconque de règles de la figure 2.4. De manière équivalente, $E \vdash C$ s'il existe un arbre dont les noeuds sont étiquetés par des clauses, la racine est étiquetée par C, les étiquettes des feuilles sont dans E et, chaque noeud qui n'est pas une feuille

- ou bien a un seul fils et, dans ce cas, son étiquette est obtenue par factorisation à partir de l'étiquette de son fils
- ou bien a deux fils et, dans ce cas, son étiquette est obtenue par résolution binaire à partir des étiquettes de ses deux fils.

Exemple 2.4.1 Si $E = \{P \vee \neg Q \vee R, P \vee \neg R\}$ alors $E \vdash_R P \vee \neg Q$ et voici un arbre de preuve :

$$\cfrac{\cfrac{P \vee \neg Q \vee R \quad P \vee \neg R}{P \vee P \vee \neg Q}}{P \vee \neg Q}$$

Exercice 22 (2)
Montrer comment les règles de la figure 2.4 permettent de dériver la clause vide $\perp$ de l'ensemble de clauses

$$E = \{P \vee \neg Q, \neg P \vee \neg Q, P \vee Q, \neg P \vee Q\}$$

Exercice 23 (2)
Montrer qu'une même clause peut avoir plusieurs arbres de preuve distincts (à permutation près des fils).

Proposition 2.4.1 *Les règles de la figure 2.4 sont correctes. ($\vdash_R \subseteq \models$).*

Preuve:
Il suffit de montrer que, pour chacune des deux règles, lorsqu'une interprétation satisfait les prémisses, elle satisfait aussi la conclusion, puis de faire une récurrence sur la longueur de la preuve (taille de l'arbre de preuve). Les détails sont laissés en exercice. □

Pour démontrer le théorème qui suit, on utilisera les *arbres sémantiques* que nous définissons maintenant formellement (après les avoir utilisés dans la preuve du théorème 2.2.1).

On suppose, ainsi que dans toute la suite, que l'ensemble des variables propositionnelles est dénombrable : $\mathcal{P} = \{P_i, i \in \mathbb{N}\}$. Une *interprétation partielle* est une application de $\{P_1, \ldots, P_n\}$ dans $\{0, 1\}$. $\{P_1, \ldots, P_n\}$ est alors le domaine de l'interprétation partielle. Les interprétations partielles sont ordonnées par prolongement : $I \leq J$ si $Dom(I) \subseteq Dom(J)$ et $\forall x \in Dom(I), I(x) = J(x)$. Si $Dom(I) \neq \mathcal{P}$, il existe exactement deux interprétations partielles I_0 et I_1 telles que $I_0, I_1 > I$ et $\forall J, J > I \Rightarrow J \geq I_0$ ou $J \geq I_1$. I_0 et I_1 sont les *successeurs* de I. Si $Dom(I) = \{P_1, \ldots, P_n\}$, I_0 est l'interprétation qui prolonge I par $I_0(P_{n+1}) = 0$. C'est le *fils droit* de I. I_1 prolonge I par $I_1(P_{n+1}) = 1$. C'est le *fils gauche* de I.

Une interprétation partielle I *falsifie* une clause C si toutes les variables propositionnelles de C sont dans le domaine de I et $I \not\models C$.

Si E est un ensemble de clauses, l'*arbre sémantique* $A(E)$ est défini comme suit. On confond les chemins finis de l'arbre, les noeuds de l'arbre et les interprétations partielles qui correspondent. On précise ci-dessous la correspondance.

- La racine correspond au chemin vide, c'est à dire à l'interprétation de domaine vide.
- Si N est un noeud de l'arbre correspondant à l'interprétation I de domaine $\{P_1, \ldots, P_n\}$,
  - Ou bien il existe une clause $C \in E$ telle que I falsifie C et N est un *noeud d'échec*. C'est alors une feuille de l'arbre et on l'étiquette par une clause de E falsifiée par I
  - Ou bien on n'est pas dans le premier cas et I est une interprétation totale sur $Var(t)$. Dans ce cas N est une feuille de l'arbre. C'est un *noeud de succès*.
  - Dans les autres cas, N a deux fils qui correspondent aux deux successeurs de I.

Exemple 2.4.2 Soit $E = \{P_1, \neg P_2 \vee \neg P_1, P_3 \vee \neg P_1\}$. L'arbre sémantique de E est représenté dans la figure 2.5. L'arbre comporte un noeud de succès (et un seul) qui correspond à la seule interprétation qui satisfait E.

Lemme 2.4.1 *Si E est un ensemble de clauses, alors E est satisfaisable si et seulement si ou bien $A(E)$ contient un noeud de succès, ou bien $A(E)$ contient un chemin infini.*

Preuve:

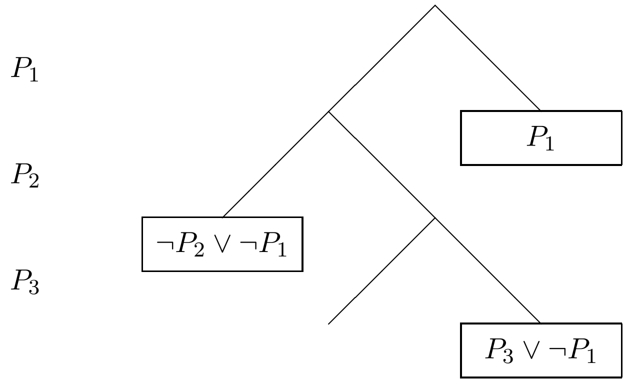

FIG. 2.5 – Arbre sémantique de l'ensemble E de l'exemple 2.4.2

Si E est satisfaisable, alors l'interprétation I qui satisfait E définit ou bien un chemin conduisant à un noeud de succès (cas où $\mathcal{P}$ est fini) ou bien un chemin infini de $A(E)$ (cas où $\mathcal{P}$ est infini).

Réciproquement, si $A(E)$ contien un noeud de succès, alors l'interprétation I qui lui correspond est totale et ne falsifie aucune clause de E et donc $I \models E$. Si $A(E)$ contient un chemin infini, il existe une suite infinie d'interprétations partielles I_n (les noeuds de ce chemin) telles que $Dom(I_n) = \{P_1, \ldots, P_n\}$, $I_n < I_{n+1}$ et, pour tout n, I_n ne falsifie aucune clause de E. On définit alors I par $I(P_i) = I_i(P_i)$ pour tout i. Pour toute clause $C \in E$, si n_C est l'indice maximal d'une variable propositionnelle de C, $I_{n_C} \models C$ par construction, et, puisque I_{n_C} prolonge I_k pour $k \leq n_C$, I et I_C coïncident sur le domaine de I_{n_C}. Il en résulte que $I \models C$. $\Box$

Exercice 24 (2)
Construire l'arbre sémantique associé à $E = \{\neg P_2, P_1 \vee P_2 \vee P_3, P_1 \vee \neg P_3, \neg P_1 \vee P_2 \vee \neg P_3, \neg P_1 \vee P_3\}$ E est-il satisfaisable ? Pourquoi ?

Théorème 2.4.1 (Complétude réfutationnelle) *Un ensemble de clauses E est insatisfaisable si et seulement si $E \vdash_R \bot$.*

Preuve:
Si $E \vdash_R \bot$, alors E est insatisfaisable par le résultat de correction.

Supposons maintenant que E est insatisfaisable et montrons que $E \vdash_R \bot$.

Soit E^* l'ensemble des clauses C telles que $E \vdash_R C$. Comme $E \subseteq E^*$ est insatisfaisable, il en est de même pour E^*. Donc, d'après le lemme 2.4.1, $A(E^*)$ ne contient ni noeud de succès, ni chemin infini. Raisonnons alors par l'absurde et supposons que $A(E^*)$ n'est pas réduit à la racine. Soit alors N un noeud maximal ayant deux fils. Ces deux fils sont des feuilles, par maximalité, donc des noeuds d'échec. Soit I l'interprétation correspondant à N et I_0, I_1 ses deux successeurs, obtenus en interprétant la variable $P \notin Dom(I)$. Il existe des clauses de E^* qui sont falsifiées respectivement par I_0 et I_1. Soient $C_0, C_1 \in E^*$

de taille minimale qui sont falsifiées respectivement par I_0 et I_1. Comme I ne falsifie ni C_0 ni C_1, $C_0 = P \vee C'_0$ et $C_1 = \neg P \vee C'_1$. De plus si $C'_0 = P \vee C''_0$, alors, par factorisation, $C_0 \vdash_R P \vee C''_0$ et I_0 falsifie $P \vee C''_0$, ce qui contredit la minimalité de C_0. De plus C'_0 ne peut s'écrire $\neg P \vee C''_0$, car I_0 falsifie C_0. Il en résulte que $Var(C'_0) \subseteq Dom(I)$. De même, $Var(C'_1) \subseteq Dom(I)$. Par résolution, $C_0, C_1 \vdash_R C'_0 \vee C'_1$, donc $C'_0 \vee C'_1 \in E^*$. De plus I_0 falsifie C'_0, donc I falsifie C'_0 et, de même, I_1 falsifie C'_1, donc I falsifie C'_1. Il en résulte que I falsifie $C'_0 \vee C'_1$ et donc que N est un noeud d'échec. Absurde.

On en conclut que l'arbre sémantique de E^* est réduit à la racine : la racine est un noeud d'échec, ce qui ne peut se produire que si $\bot \in E^*$, c'est à dire $E \vdash_R \bot$. $\square$

Le résultat suivant montre que l'on peut obtenir le théorème de compacité comme corollaire au théorème de complétude.

Corollaire 2.4.1 *Si E est un ensemble de clauses insatisfaisables, alors E contient un sous-ensemble fini de clauses insatisfaisable.*

Preuve:
Si E est insatisfaisable, alors $E \vdash_R \bot$. Or la preuve correspondante est un arbre fini. Si E_0 est le sous-ensemble des clauses qui étiquettent des feuilles de l'arbre, $E_0 \subseteq E$, E_0 est fini et E_0 est insatisfaisable. $\square$

La *longueur* d'une preuve est le nombre de sous-arbres distincts dans cette preuve.

Une preuve Π est *sans boucle* si, pour toute clause C, tout sous-arbre de Π dont la racine est étiquetée par C ne contient pas lui-même de sous-arbre propre dont la racine est étiquetée par C.

Exercice 25 (6)
Soit $\mathcal{P}$ un ensemble fini de variables propositionnelles, de cardinal n. On appelle 2-clause toute clause contenant au plus deux littéraux.

1. Montrer que, si E est ensemble de 2-clauses insatisfaisable, toute preuve sans boucle de $\bot$ est de longueur polynômiale en n
2. Montrer par contre que la taille peut être exponentielle : donner un exemple d'ensemble de 2-clauses E tel que E est insatisfaisable et une preuve sans boucle de $\bot$ à partir de E dont la taille est exponentielle en n.
3. Donner un algorithme polynômial en n pour décider de la satisfaisabilité d'un ensemble de 2-clauses

Exercice 26 (6)
Soit $\mathcal{P}$ un ensemble de variables propositionnelles fini. Donner un ensemble de clauses E qui est insatisfaisable et tel qu'il existe une preuve de $\bot$ de taille exponentielle en $|E|$ et ne contenant aucune redondance (i.e. deux noeuds distincts de l'arbre de preuve qui ne sont pas des feuilles sont étiquetés par des formules distinctes).