Examen 2: Corrigé

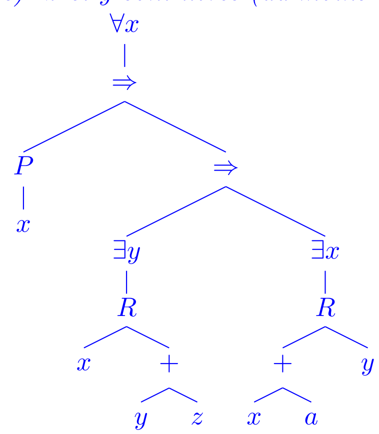
Exercice 1. ⟦ *3,5 pts* ⟧

Soit la formule à priorité

$$F = \forall x(P(x) \Rightarrow \exists y R(x, y+z) \Rightarrow \exists x R(x+a, y))$$

1. *Donner la signature associée à la formule F.* *La signature associée est $a^{f0}, +^{f2}, P^{r1}, R^{r2}$. Autrement dit: on a une constante a, une fonction binaire +, et un prédicat unaire P et un prédicat binaire R.*

2. *Donner la structure syntaxique de F sous forme d'arbre, et précisez les variables libres et les variables liées de F.* *x est liée. (Toutes les occurences sont liées). z et y sont libres (au moins une occurence libre).*



3. *Donner $G = F < y := z >$ et $H = F < y := x >$. Que peut-on dire des deux formules $G \Rightarrow \exists y F$ et $H \Rightarrow \exists y F$?*

- $G = F < y := z >= \forall x(P(x) \Rightarrow \exists y R(x, y+z) \Rightarrow \exists x R(x+a, z))$.

- $H = F < y := x >= \forall x(P(x) \Rightarrow \exists y R(x, y+z) \Rightarrow \exists x R(x+a, x))$.

- *La $G \Rightarrow \exists y F$ est valide. Parce que on a remplacé y par un terme libre pour y dans F. (Théorème d'instanciation).*

- *On ne peut rien dire sur $H \Rightarrow \exists y F$. Car on a remplacé y par un terme(x) qui n'est pas libre pour y dans A.*

Exercice 2. *(3 pts.)*

1. *Montrer que ce raisonnement est incorrect par la méthode des expansions finies.* $\forall x(P(x) \vee Q(x)) \models \forall x P(x) \vee \forall x Q(x)$.

Il suffit de trouver une assignation(interpétation puisque les formules sont fermées) qui soit modèle de $\forall x(P(x) \vee Q(x))$ mais contre modèle de $\forall x P(x) \vee \forall x Q(x)$.

Avec $D = \{0\}$. Trouver une interprétation de P et de Q telle que $P(0)+Q(0) = 1$ et $P(0)+Q(0) = 0$ est impossible. Considérons $D = \{0, 1.\}$ On doit avoir

$$\begin{cases} P(0) + Q(0) = 1 \\ P(1) + Q(1) = 1 \\ P(0)P(1) = 0 \\ Q(0)Q(1) = 0. \end{cases}$$

Posons $P(0) = 1$. On déduit successivement alors: $P(1) = 0$ donc $Q(1) = 1$ et $Q(0) = 0$. On conclut que $D = \{0, 1\}, P_I = 0, Q_I = \{1\}$ est un contre modèle du raisonnement.

2. *Montrer que ce raisonnement est correct :* $\forall x A \vee \forall x B \models \forall x(A \vee B)$.

Supposons que (I, e) est un modèle de $\forall x A \vee \forall x B$: $[\forall x A \vee \forall x B]_{(I,e)} = 1$.
On a $[\forall x A]_{(I,e)} = 1$ ou $[\forall x B]_{(I,e)} = 1$.
On a deux cas à étudier.
1er cas. $[\forall x A]_{(I,e)} = 1$. *Donc* $\prod_{d \in D} [A]_{(I,e[x=d])} = 1$ *Donc* $\prod_{d \in D} ([A \vee B]_{(I,e[x=d])}) = 1$.
On conclut $[\forall x(A \vee B)]_{(I,e)} = 1$. Le deuxième cas se traite d'une manière similaire. On vient de montrer que tout modèle de $\forall x A \vee \forall x B$ est un modèle de $\forall x(A \vee B)$. Ce qui est la définition du fait que le raisonnement est correct.

Exercice 3. ***(4,5 pts)*** *Soit*

$$F = \forall x R(a,x) \wedge \forall x R(x,b) \Rightarrow \exists x Q(x) \wedge \forall x \exists y R(y, f(x))$$

1. *Donner la forme clausale de F en précisant les étapes du calcul.*

Formes prénexes.

$$\begin{aligned} F &\equiv (\forall x R(a,x) \wedge \forall x R(x,b)) \Rightarrow (\exists x Q(x) \wedge \forall x \exists y R(y, f(x))) \\ &\equiv \forall (x R(a,x) \wedge R(x,b)) \Rightarrow (\exists x Q(x) \wedge \forall x \exists y R(y, f(x))) \\ &\equiv \forall x (R(a,x) \wedge R(x,b)) \Rightarrow (\exists z Q(z) \wedge \forall w \exists y R(y, f(w))) \\ &\equiv \exists x \exists z \forall w \exists y \Big((R(a,x) \wedge R(x,b)) \Rightarrow (Q(z) \wedge R(y, f(w))) \Big) \end{aligned}$$

Skolémisation

$$F_s = \forall w \Big((R(a,c) \wedge R(c,b)) \Rightarrow (Q(d) \wedge R(g(w), f(w))) \Big)$$

On a introduit deux nouvelles constantes c et d. On a introduit aussi un nouveau symbole de fonction g

Forme Clausale

$$\begin{aligned} F_s &\equiv \forall w \neg (R(a,c) \wedge R(c,b)) \vee (Q(d) \wedge R(g(w), f(w))) \\ &\equiv \forall w \neg R(a,c) \vee \neg R(c,b)) \vee (Q(d) \wedge R(g(w), f(w))) \\ &\equiv \forall w (\neg R(a,c) \vee \neg R(c,b) \vee Q(d)) \wedge (\neg R(a,c) \vee \neg R(c,b)) \vee R(g(w), f(w)) \end{aligned}$$

$$S = \{\neg R(a,c) \vee \neg R(c,b) \vee Q(d) \quad , \quad \neg R(a,c) \vee \neg R(c,b)) \vee R(g(w), f(w))\}$$

2. *Quelle est la relation entre F et S l'ensemble des clauses obtenues? F et S(sa fermeture universelle) sont equisatisfaisable. C'est à dire: F est contradictoire ssi S est contradictoire. Autrement dit: F est satsisfaisable ssi S est satisfaisable.*

Exercice 4. ***(4 pts.)*** *Pour cet ensemble des clauses, donner la signature associée, le domaine de Herbrand, le déploiement de Herbrand:$H(S)$ et dire si l'ensemble(sa fermeture universelle) est satisfaisable ou non.*

$$S = \{P(x) \vee Q(x) \vee R(x), \neg P(a) \vee Q(x), \neg Q(b), \neg R(b)\}.$$

$\Sigma = a^{r0}, b^{r0}, P^{r1}, R^{r1}, Q^{r1}$
$H = \{a, b\}$. *Car il n'y a pas de symbole de fonction.*
$A = \{P(a), Q(a), R(a), P(b), Q(b), R(b)\}$.
$H(S) = \{P(a) \vee Q(a) \vee R(a), \neg P(a) \vee Q(a), \neg Q(b), \neg R(b), P(b) \vee Q(b) \vee R(b), \neg P(a) \vee Q(b)\}$
$H(S)$ est fini. Le théorème de Herbrand nous dit que S est satisfaisable ssi $H(S)$ est satisfaisable.
En appliquant la méthode de Davis Putnam sur $H(S)$ on obtient: Par RU sur $\neg Q(b)$ et $\neg R(b)$ on obtient:

$$\{P(a) \vee Q(a) \vee R(a), \neg P(a) \vee Q(a), P(b), \neg P(a)\}$$

En appliquant RU sur $\neg P(a)$ on obtient:

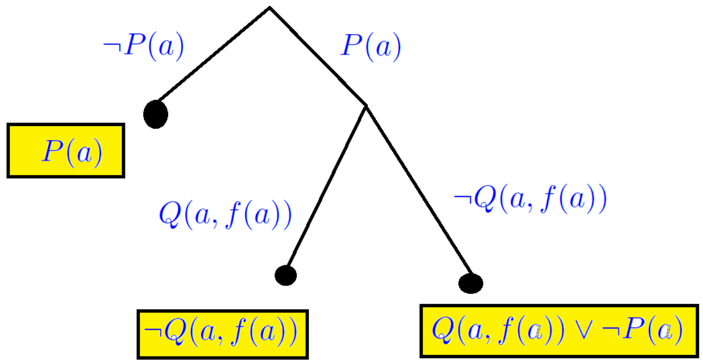
$$\{Q(a) \vee R(a), P(b)\}$$

On fait RI sur $Q(a), R(a), P(b)$ donne l'ensemble vide. Donc $H(S)$ est satisfaisable.

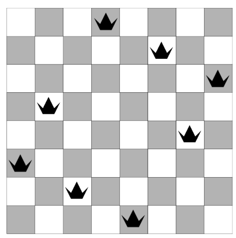
Exercice 5. ***(3 pts.)***
Considérons l'ensemble S de clauses. Montrer que l'ensemble (sa fermeture universelle) est insatisfaisable en donnant un arbre sémantique clos.

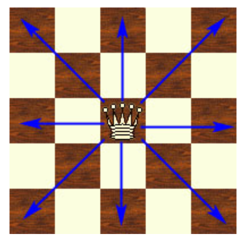
$$S = \{P(x), Q(x, f(x)) \vee \neg P(x), \neg Q(y, z)\}$$

Le domaine de Herbrand : $H = \{a, f(a), f(f(a)), \ldots\}$
La base de Herbrand : $A = \{P(a), Q(a,a), P(f(a)), Q(a, f(a)), Q(f(a), a), Q(f(a), f(a)), \ldots\}$.
L'ensemble des instance de base : $H(S) = \{P(a), \neg Q(a,a), Q(a, f(a)) \vee \neg P(a), P(f(a)), \neg Q(a, f(a)), \ldots\}$



Exercice **6**. ***(3pts)*** *Formaliser le problème des 8 reines en calcul des prédicats. Rappelons que le problème est de placer huit dames(reines) d'un jeu d'échecs sur un échiquier de 8×8 cases sans que les dames ne puissent se menacer mutuellement, conformément aux règles du jeu d'échecs.*





Les dames ne sont pas numérotées. Notons $P(i, j)$ le prédicat qui est vrai ssi une dame existe dans la case (i, j). Le domaine est $\{1, 2, 3, 4, 5, 6, 7, 8\}$.

- $\forall i \exists j P(i, j)$. *Dans chaque ligne il existe une dame. (Au minimum il y aura donc 8 dames).*
- $\forall i \forall x \forall y ((P(i, x) \wedge P(i, y) \Rightarrow x = y))$. *Au plus une dame par ligne.*
- $\forall j \forall x \forall y ((P(x, j) \wedge P(y, j) \Rightarrow x = y))$. *Au plus une dame par colonne.*
- $\forall i \forall j \forall x \forall y \Big(P(i, j) \wedge P(x, y) \wedge (|x - y| = |i - j|) \Rightarrow (x = i \wedge y = j) \Big)$. *Au plus une dame par diagonale.*

Question Bonus (? pts) Peut-on faire un algorithme(semi décidable) de recherche de contre-modèles qui étant donné une formule F du calcul de prédicats, répond en un temps fini par oui *dans le cas d'existence de contre modèle* ? Si c'est possible alors expliquer en bref comment ont doit procéder, sinon justifier.

La réponse est non.

On sait qu'il existe un programme prog1, semi décidable qui s'arrête si F est valide. (En appliquant la méthode de Herbrand sur $\neg F$ par exemple).

Par l'absurde; supposant qu'il existe un programme prog2 qui s'arrête si F a un contre modèle. (Donc il va toujours répondre en un temps fini si F n'est pas valide).

On peut alors programmer un programme prog3 qui execute (lance en parallèle par exemple) les deux programmes prog1 et prog2 sur F et s'arrête quand l'un des deux programmes s'arrêtent. Si F est valide alors prog1 s'arrête en un temps fini donc prog3 s'arrête en un temps fini. Si F n'est pas valide alors prog2 s'arrête en un temp fini donc prog3 s'arrête en un temps fini à tout les coups.

Donc on aurait un programme décidable qui étudie la validité de toute formule de calcul des prédicats. Cela est impossible selon le théorème de Church et Turing. **Conclusion**: Le prog2 n'existe pas.