Nom :

Note : on donnera les réponses dans les cases prévues à cet effet. Ceci a pour double but de limiter la taille des réponses (et donc d'éviter les arguments inutilement complexes), et d'indiquer la difficulté relative des questions. Les questions sont d'autant plus notées qu'elles sont difficiles.

Preuve par instantiation et tests SAT

Lee et Plaisted (1992) ont remarqué que la règle de résolution (binaire, ici) :

$$\frac{C \vee +A \quad C' \vee -A'}{C\sigma \vee C'\sigma}\ \sigma = \mathrm{mgu}\ (A \doteq A')$$

mélange deux effets. Le premier est de calculer un unificateur le plus général, ce qui trouve très efficacement la forme générale des instances des clauses en prémisses qui seront utiles, soit $(C \vee +A)\sigma$ et $(C' \vee -A')\sigma$; d'autre part, d'appliquer la règle de résolution propositionnelle. Or cette dernière est un test, relativement inefficace, de satisfiabilité propositionnelle. L'idée de Lee et Plaisted est donc de remplacer la règle de résolution (générale) par, d'une part, une règle de *génération d'instances* :

InstGen

$$\frac{C \vee +A \quad C' \vee -A'}{(C \vee +A)\sigma \quad (C' \vee -A')\sigma}$$

où $\sigma = \mathrm{mgu}\ (A \doteq A')$.

Ceci signifie que l'on produit les deux clauses $(C \vee +A)\sigma$ et $(C' \vee -A')\sigma$, mais pas leur résolvant, en particulier. On ne produit pas non plus de facteur. En revanche, comme en résolution, on supposera implicitement que les deux prémisses $C \vee +A$ et $C' \vee -A'$ sont renommées de sorte à n'avoir aucune variable en commun.

D'autre part, au lieu de chercher à dériver la clause vide (ce qui ne sera pas possible en général), la procédure de Lee et Plaisted cherchera à produire un ensemble S de clauses propositionnellement insatisfiable. Plus précisément, on note $\star$ une constante n'apparaissant pas dans S, et $\tau_\star$ la substitution qui à toute variable associe $\star$. On appliquera la règle **InstGen** non pas jusqu'à dériver la clause vide, mais jusqu'à ce que l'ensemble de clauses obtenu S soit tel que $S\tau_\star$ soit propositionnellement insatisfiable. (Notons que $S\tau_\star$ est un ensemble de clauses closes, et un modèle propositionnel de $S\tau_\star$ est juste une affectation de valeurs de vérité à chaque atome, clos, qui apparaît dans $S\tau_\star$.)

1. Montrer que, si $S\tau_\star$ est propositionnellement insatisfiable, alors S est insatisfiable (au premier ordre).

2. Notons $S \to S'$ s'il existe deux clauses (possiblement la même) dans S qui sont prémisses de la règle **InstGen**, et telles que S' vaille S union les deux conclusions de la règle. Déduire de la question précédente que la procédure de Lee et Plaisted est correcte : si $S_0 \to^* S$, et $S\tau_\star$ est propositionnellement insatisfiable, alors S_0 est insatisfiable.

3. On s'attaque maintenant à la complétude. Soit $(T, C_\bullet, \theta_\bullet)$ un arbre décoré (fini) pour l'ensemble S sur l'ensemble d'atomes clos $A_1^0, \ldots, A_n^0$, et supposons $S\tau_\star$ propositionnellement satisfiable. Soit $I_\star$ une interprétation sur $A_1^0, \ldots, A_n^0$, telle que $I_\star \models S\tau_\star$.

   On construit une interprétation I (une branche de l'arbre) par forcing, comme dans le cours, mais avec une subtilité : on va autoriser certains atomes à être indéfinis dans I. Appelons *interprétation partielle* I sur $A_1^0, \ldots, A_i^0$ un ensemble de littéraux parmi $+A_1^0, -A_1^0, \ldots, +A_i^0, -A_i^0$, qui est non contradictoire au sens où I ne contient pas à la fois $+A_j^0$ et $-A_j^0$, pour aucun j, $1 \leq j \leq i$. Un atome A_j^0 est *vrai* dans I si $+A_j^0 \in I$, *faux* dans I si $-A_j^0 \in I$, et *indéfini* dans I sinon. Pour une clause close C, on dit que C est *vraie* dans I si et seulement si C contient un littéral vrai (donc défini, aussi) dans I.

   On construit une interprétation partielle I pas à pas comme suit :

   (a) Initialement, $I := \emptyset$;

   (b) Tant qu'il existe une clause C_N (N nœud d'échec) telle que $C_N\theta_N$ n'est pas vrai dans I et C_N contient un littéral $\pm A$ tel que $A\theta_N$ soit indéfini dans I mais $\pm A\tau_\star$ est vrai dans $I_\star$, alors ajouter $\pm A\theta_N$ à I.

   Cette procédure termine en au plus n étapes (où l'on choisit N arbitrairement dans la deuxième étape au cas où il y aurait plusieurs C_N possibles), et définit une interprétation partielle I_1, la valeur finale de I.

   On considère n'importe quelle branche de T qui fait les mêmes choix que I_1, autrement qui descend à gauche de A_i^0 si A_i^0 est faux dans I_1, à droite si A_i^0 est vrai dans I_1, et de

n'importe quel côté si A_i^0 est indéfini. Cette branche se termine sur un nœud d'échec, que nous appellerons N_1.

Montrer qu'il existe un littéral $\pm A$ dans C_{N_1} tel que $\pm A\tau_\star$ soit vrai dans $I_\star$ mais $\pm A\theta_N$ soit faux dans I_1.

4. En déduire que, si $S\tau_\star$ est propositionnellement satisfiable, il existe un (autre) nœud d'échec N_2 tel que l'on peut appliquer la règle **InstGen** sur les clauses C_{N_1} et C_{N_2}. Autrement dit, C_{N_1} est de la forme $C \vee \pm A$, C_{N_2} est de la forme $C' \vee \mp A'$ ($\mp$ étant le signe opposé de $\pm$), et $\sigma = \mathrm{mgu}\ (A \doteq A')$ existe.

5. Montrer que l'on peut demander de plus que $\pm A'\tau_\star$ soit faux dans $I_\star$.

6. En déduire que $A\tau_\star \neq A'\tau_\star$.

7. Déduire de la question 6 que $\sigma = \mathrm{mgu}\ (A \doteq A')$, tel que défini en question 4, est tel qu'il existe une variable x, dans A ou dans A', telle que $x\sigma$ n'est pas une variable.

8. Dans la situation de la question 4, on construit le nouvel arbre décoré $(T, C'_\bullet, \theta'_\bullet)$ pour S' égal à l'union de S et des deux clauses conclusions de **InstGen**, en posant $C'_N = C_N$, $\theta'_N = \theta_N$ si $N \neq N_1, N_2$; en posant $C'_{N_1} = C_{N_1}\sigma$ et $C'_{N_2} = C_{N_2}\sigma$, où $\sigma = \mathrm{mgu}\ (A \doteq A')$, et θ'_{N_1}, θ'_{N_2} définies de sorte que $C'_{N_1}\theta'_{N_1} = C_{N_1}\theta_{N_1}$, $C'_{N_2}\theta'_{N_2} = C_{N_2}\theta_{N_2}$.
   Montrer que ceci est sensé, autrement dit que θ'_{N_1} et θ'_{N_2} existent.

9. Les questions précédentes définissent un système de réécriture d'arbres décorés, qui réécrit $(T, C_\bullet, \theta_\bullet)$ en l'arbre $(T', C'_\bullet, \theta'_\bullet)$, défini à la question 8. En raisonnant (pour l'essentiel) sur les tailles des clauses (C_N, resp. C'_N) étiquetant les nœuds d'échec des arbres décorés, montrer que ce système de réécriture termine.

10. En déduire la complétude de la procédure de Lee et Plaisted.