TD 12 : Résolution (suite)

Nicolas Margulies nicolas.margulies@lmf.cnrs.fr
Théo Vignon theo.vignon@lmf.cnrs.fr

1 Arbres sémantiques

Soit $\mathcal{L}$ le langage contenant pour symboles de fonctions, a d'arité nulle, f d'arité 1, et deux prédicats unaires P et Q.

Soit $E = \{P(x), \neg P(f(x)) \vee \neg Q(a), Q(a) \vee \neg P(f(a))\}$ un ensemble de clauses, et considérons l'énumération suivante des atomes clos : $P(a), Q(a), P(f(a)), Q(f(a)), \ldots$.

Construisez l'arbre sémantique associé.

2 Stratégie négative

Dans cet exercice, on se concentre sur les formules propositionnelles. Le langage contient seulement des symboles de prédicat constants $P_1, P_2, \ldots$.

Une clause est *négative* si elle ne contient que des littéraux négatifs. On étudie la stratégie de résolution appelée *stratégie négative* : les applications de la règle de résolution sont limitées au cas où une des prémisses est négative. On écrit $\vdash_\neg$ la relation de déduction associée.

1. Soit $E = \{ \neg P \vee Q \,,\, P \vee Q \,,\, P \vee \neg Q \,,\, \neg P \vee \neg Q \}$. Montrez que $E \vdash_\neg \bot$.

On appelle *interprétation partielle* un sommet d'un arbre sémantique associé à l'énumération $P_0, P_1, \ldots$, representée par l'ensemble des littéraux choisis à ce point.

Si I et J sont des interprétations partielles, on écrit $I >_{lex} J$ quand il existe $k \geq 1$ tel que :

— pour tout $j < k$, $P_j \in I$ et $P_j \in J$, ou $\neg P_j \in I$ et $\neg P_j \in J$;

— $P_k \in I$ et $\neg P_k \in J$.

2. Quelle est l'interprétation partielle associée à :
   (a) la racine ?
   (b) ses enfants ?
   (c) les enfants d'un sommet représenté par l'interprétation partielle I de taille k ?
3. Montrez que $\geq_{lex}$ est un ordre et que pour toutes interprétations partielles I et J, on a $I \subseteq J$, $J \subseteq I$, $I \leq_{lex} J$, ou $J \leq_{lex} I$.
4. Soit A l'arbre sémantique associé à un ensemble de clauses E. En supposant que A est fini et non vide, montrez qu'il existe une unique interprétation partielle maximale pour $\leq_{lex}$ étant une feuille et n'invalidant pas de clause négative de E.
5. Prouvez la complétude de $\vdash_\neg$ en utilisant des arbres sémantiques.
   Indice : considérez la feuille maximale pour $\leq_{lex}$ n'invalidant aucune des clauses de E^* dans l'arbre de ce dernier, avec $E^* = \{ C \;:\; E \vdash_\neg C \}$.

3 Stratégie d'entrée

De retour au premier ordre, considérons une signature $\mathcal{P}, \mathcal{F}$.

On considère une autre stratégie de résolution : la règle de résolution est restreinte au cas où au moins une des prémisses est dans l'ensemble de clauses original. On appelle cette stratégie la stratégie *d'entrée*.

1. Montrez que $\bot$ ne peut pas être dérivé depuis $E = \{ P \vee Q \, , \, \neg P \vee Q \, , \, P \vee \neg Q \, , \, \neg P \vee \neg Q \}$ en utilisant cette strategie. Que pouvez-vous en déduire ?
2. Une *clause de Horn* est une clause contenant au plus un littéral positif. Montrez que de telles clauses sont stables par résolution et factorisation.

Soient P, P', Q, Q' des littéraux, et σ l'unificateur le plus général du problème d'unification $\{ P \stackrel{?}{=} P' \, , \, Q \stackrel{?}{=} Q' \}$, i.e. une substitution σ telle que $\sigma(P) = \sigma(P')$ et $\sigma(Q) = \sigma(Q')$, et pour toute autre substitution θ unifiant ce problème, il existe η telle que $\theta = \eta \circ \sigma$.

3. On pose $\sigma_P = mgu(P, P')$ et $\sigma_Q = mgu(\sigma_P(Q), \sigma_P(Q'))$. Montrez que $\sigma = \sigma_Q \circ \sigma_P$.

Le but de cet exercice est de montrer que la stratégie d'entrée est complète pour les clauses de Horn. Fixons un ensemble de clauses de Horn E. On veut alors montrer par récurrence que pour toute preuve par résolution π d'une clause C à partie de E, il existe une autre preuve π' de C depuis E qui suit la stratégie d'entrée. On définit dans ce but :

— $N(\pi)$, le nombre de règles de π,

— $H(\pi)$, le nombre de règles dans la sous-preuve gauche de π (si π commence par une factorisation, il s'agit du nombre de règles de sa seule sous-preuve).

4. Prouvez que la stratégie d'entrée sur un ensemble de clauses de Horn est complète. Procédez par récurrence sur $(N(\pi), H(\pi))$, ordonné par l'ordre lexicographique.

   Astuce : Le cas problématique est celui où la première règle de π est une résolution. Dans ce cas, faites une disjonction de cas sur la première règle R' à gauche, et permutez ces deux règles afin de réduire la taille de la sous-preuve de gauche.

4 La stratégie de sélection et une application à la sécurité

4.1 La stratégie de sélection

Soit f une fonction qui étant donné une clause, retourne l'un des littéraux, appelé le *littéral choisi* de la clause. La *stratégie de sélection* associée à la fonction f restreint la règle de résolution de sorte que les littéraux sur lesquels la résolution est effectuée soient les littéraux choisis dans leurs clauses respectives.

1. Montrez que cette stratégie n'est pas complète.
2. Est-elle complète si f choisit un littéral négatif dès que possible ?

Quelle que soit la fonction choisie, la stratégie de sélection est complète pour des clauses de Horn (i.e. qui contiennent au plus un littéral positif).

4.2 Sécurité

On veut représenter des protocoles cryptographiques en utilisant des clauses de Horn. Pour cela, on utilise la signature suivante :

— Les termes représentent les messages échangés par les participants.

— Les primitives cryptographiques sont représentées par des symboles de fonction.

- pair(2) et aenc(2) sont des symboles de fonction binaires représentant respectivement les paires de messages et le chiffrement d'un message en utilisant une clé.
- pk(1) est le symbole de fonction unitaire pour symboliser la clé publique d'un participant.
- $s(0)$ est un symbole de fonction constant symbolisant le secret.
- $a(0)$, $b(0)$, $i(0)$ sont des symboles de fonction constants représentant les clés secrètes de 3 participants (Alice,Bob et l'imposteur (l'attaquant)).

— L'attaquant et ses capacités sont représentées par le prédicat unaire att(1).

Par exemple, l'attaquant peut construire et déconstruire les paires, représenté par des clauses de Horn de la façon suivante :

$$\begin{aligned} \text{att}(x) \wedge \text{att}(y) &\Rightarrow \text{att}(\text{pair}(x, y)) \\ \text{att}(\text{pair}(x, y)) &\Rightarrow \text{att}(x) \\ \text{att}(\text{pair}(x, y)) &\Rightarrow \text{att}(y) \end{aligned}$$

Il peut également chiffrer des messages avec une clé publique connue :

$$\text{att}(m) \wedge \text{att}(k) \Rightarrow \text{att}(\text{aenc}(m, k))$$

Pour déchiffrer des messages, il a besoin de la clé secrete du participant :

$$\text{att}(\text{aenc}(m, \text{pk}(p))) \wedge \text{att}(p) \Rightarrow \text{att}(m)$$

On suppose également que l'attaquant a accès aux clés publiques des autres participants, c'est-à-dire que les clauses att(pk(a)) et att(pk(b)). il a également accès a ses clés secrète et publique, i.e. les clauses att(i) et att(pk(i)) sont valides. On nomme A l'ensemble des 9 clauses que l'on vient juste de décrire.

1. Prouvez que si l'attaquant a accès à un secret chiffré mais pas à la clé secrète associée, il ne peut pas avoir accès au secret, i.e. on ne peut pas dériver $\bot$ depuis $A \cup \{$ att(aenc$(s, \text{pk}(a)))$, $\neg$att(s) $\}$ en utilisant la résolution.

   Astuce : Utilisez la résolution par sélection, avec la fonction de sélection qui choisit les litéraux de la forme, att(t) ou $\neg$att(t) avec t qui n'est pas une variable lorsque c'est possible, sinon un litéral positif, et un littéral quelconque sinon.

Voici un protocole cryptographique :

— Le participant A contacte le participant B, chiffrant avec la clé publique de B le secret et l'envoie avec sa clé publique :

$$A \to B \ : \ \text{pair}(\text{pk}(a), \text{aenc}(s, \text{pk}(b)))$$

— Le participant B lui répond avec le secret chiffré avec la clé publique de A :

$$B \to A \ : \ \text{aenc}(s, \text{pk}(a))$$

L'attaquant peut intercepter des messages échanger durant le protocole, et transmet des messages à Alice et Bob, ce qui est réprésenté par les clauses de Horn :

$$\begin{aligned} & & \text{att}(\text{pair}(\text{pk}(a), \text{aenc}(s, \text{pk}(b)))) \\ \text{att}(\text{pair}(x, \text{aenc}(y, \text{pk}(b)))) &\Rightarrow & \text{att}(\text{aenc}(y, x)) \end{aligned}$$

On appelle P l'ensemble des 11 clauses contenant A et les deux clauses précédentes.

2. Prouvez qu'il y a une attaque sur ce protocole, i.e. on peut dériver $\bot$ depuis $P \cup \{ \neg\text{att}(s) \}$.

3. Une façon d'empêcher cette attaque est de chiffrer à la fois la clé publique de A et le secret dans le premier message, ce qui donne le nouveau protocole :

$$\begin{array}{rcl} A \to B & : & \mathsf{aenc}(\mathsf{pair}(\mathsf{pk}(a), s), \mathsf{pk}(b)) \\ B \to A & : & \mathsf{aenc}(s, \mathsf{pk}(a)) \end{array}$$

avec les clauses associées

$$\begin{array}{rcl} & & \mathsf{att}(\mathsf{aenc}(\mathsf{pair}(\mathsf{pk}(a), s), \mathsf{pk}(b))) \\ \mathsf{att}(\mathsf{aenc}(\mathsf{pair}(x, y), \mathsf{pk}(b))) & \Rightarrow & \mathsf{att}(\mathsf{aenc}(y, x)) \end{array}$$

Soit P' l'ensemble des 11 clauses contenant A et les deux clauses précédente. Montrez que l'on ne peut pas dériver $\bot$ depuis $P' \cup \{ \neg\mathsf{att}(s) \}$.

Astuce : même chose que la question 1.