Partiel de Programmation 2

12 avril 2011

Exercice 1. *LJ et LK*

1. On considère la formule $(A \to B) \to ((B \to C) \to A) \to B$. En donner une démonstration dans LK et un contre-modèle de Kripke en précisant les sous-formules forcées par les différents mondes.

2. Montrer que les trois règles suivantes sont équivalentes dans NJ.

$$\frac{\Gamma, A \to B \vdash A}{\Gamma \vdash A}(1) \qquad \frac{\Gamma, \neg A \vdash \bot}{\Gamma \vdash A}(2) \qquad \frac{\Gamma, A \vdash B \quad \Gamma, \neg A \vdash B}{\Gamma \vdash B}(3)$$

Exercice 2. *Élimination des coupures*

En déduction naturelle, une *coupure* dans une preuve est définie comme la succession d'une règle d'introdution d'un connecteur logique suivi par la règle d'élimination de ce même connecteur.

$$\frac{\dfrac{\pi_1 \quad \cdots \quad \pi_k}{\ldots \vdash \ldots \star \ldots}\star\text{-intro} \qquad \sigma_1 \quad \cdots \quad \sigma_h}{\ldots \vdash \ldots}\star\text{-élim}$$

1. Donner tous les types de coupures de NJ.

2. On notera par $\mathrm{ANJ}_{\to}$ le système de déduction naturelle minimale *affine* : on restreint les règles $\to$-intro à utiliser au plus une occurrence de l'hypothèse déchargée. Formellement, le contexte Γ doit être vu comme un multiensemble (c.-à-d. avec prise en compte des répétitions), et la règle $\to$-elim devient

$$\frac{\Gamma_1 \vdash A \to B \quad \Gamma_2 \vdash A}{\Gamma_1, \Gamma_2 \vdash B}\to\text{-elim}$$

tandis que les deux autres règles restent identiques [1]. Démontrer le théorème d'élimination des coupures pour $\mathrm{ANJ}_{\to}$:

Théorème. *Si la formule F est prouvable, alors il en existe une preuve sans coupure.*

---

1. $ANJ_{\to}$ est assez intuitif à définir dans une représentation à arbre avec déchargement des feuilles : là $\to$-intro a droit à décharger au plus une seule feuille.

Exercice 3. *Systèmes Acceptables de Programmation*

Soit $\langle .,. \rangle$ un codage primitif récursif de $\mathbb{N}^2$ dans $\mathbb{N}$ et $(\varphi_n)_{n\in\mathbb{N}}$ une énumération (c.-à-d. une suite surjective) des fonctions Turing-calculables. Elle est dite *universelle* s'il existe un entier U tel que pour tout x et y, $\varphi_U(\langle x, y\rangle) = \varphi_x(y)$ si $\varphi_x(y)$ est défini et $\varphi_U(\langle x, y\rangle)$ diverge sinon. Un *système acceptable de programmation* (SAP en abrégé) est une énumération universelle des fonctions Turing-calculables qui possède une fonction récursive primitive $c : \mathbb{N}^2 \to \mathbb{N}$ telle que :

$$\forall i, j, x : \varphi_{c(i,j)}(x) = \varphi_i(\varphi_j(y))$$

Elle représente intuitivement la possibilité de *calculer* le code de la composition de deux programmes en utilisant leurs codes.
On appelle *fonction s-m-n* une fonction récursive primitive $s : \mathbb{N}^2 \to \mathbb{N}$ telle que

$$\forall i, x, y : \varphi_{s(i,x)}(y) = \varphi_i(\langle x, y\rangle).$$

Une fonction s-m-n calcule le code d'une application partielle d'un programme qui attend deux arguments. Cela correspond à substituer la valeur x à l'entrée du programme, de façon algorithmique. On se propose de montrer l'équivalence entre l'existence de c et de s.

Théorème. *Une énumération universelle est un SAP ssi elle admet une fonction s-m-n.*

1. Montrer qu'il existe un entier F tel que pour tous entiers i, j, x on a $\varphi_F(\langle i, \langle j, x\rangle\rangle) = \varphi_i(\varphi_j(x))$. En déduire que de l'existence d'une fonction s-m-n découle celle de la fonction de composition.

Supposons avoir un SAP, on se propose de démontrer l'existence d'une fonction s-m-n.

2. Définir une fonction récursive primitive `cnst` qui sur l'entrée n donne un indice (dans un SAP) de la fonction constante égale à n :

$$\forall n, x : \varphi_{\texttt{cnst}(n)}(x) = n$$

(indication : on peut supposer d'avoir des entiers Z et S qui représentent des codes des fonctions à un argument constante 0 et successeur)

3. Montrer que dans un SAP on a une fonction primitive récursive `pairwith`(x) telle que

$$\forall x, y : \varphi_{\texttt{pairwith}(x)}(y) = \langle x, y\rangle.$$

4. Définir la fonction s-m-n s en utilisant c et les questions précédentes.

5. Pour un SAP quelconque, on définit le problème de l'arrêt comme l'ensemble :

$$H = \{\, \langle i, x\rangle \mid \varphi_i(x) \text{ est défini} \,\}.$$

Montrer qu'il est indécidable.

6. Démontrer le théorème du point fixe de Kleene pour un SAP quelconque.

Théorème.
Si $f : \mathbb{N} \to \mathbb{N}$ est une fonction calculable totale, alors il existe m tel que $\varphi_m = \varphi_{f(m)}$.

(indication : est-ce que $\varphi_{f(s(t,t))}(x)$ est calculable à partir de $\langle t, x\rangle$?)
Déduisez-en l'existence d'une quine, un programme p qui affiche son propre code (c.-à-d. $\varphi_p(x) = p$).

7. Faire de même pour le théorème de Rice.

8. Décrire informellement un codage $\langle M\rangle$ des machines de Turing $(Q, \{\bullet\}, \Gamma, E, q_0, F)$ dans $\mathbb{N}$ et expliquer pourquoi il forme un SAP.

9. À l'inverse, existe-il une énumération des machines de Turing qui ne soit pas un SAP ?

Exercice 4. *Fonctions élémentaires*

Soit $\mathcal{E}$ la plus petite classe de fonctions de $\mathbb{N}^k$ dans $\mathbb{N}$ contenant :
- les constantes $\underline{n}$ pour tout n, vues comme fonctions de $\mathbb{N}^0$ dans $\mathbb{N}$;
- l'addition $+ : \mathbb{N}^2 \to \mathbb{N}$,
- la soustraction naturelle $\dot{-} : \mathbb{N}^2 \to \mathbb{N}$ définie par $x \dot{-} y = \max(x - y, 0)$,
- les projections $P_i^k : \mathbb{N}^k \to \mathbb{N}$ avec $P_i^k(n_1, \dots, n_k) = n_i$,

et close par rapport aux opérations de
- composition $\texttt{comp}(g, h_1, \dots, h_k)$;
- pour $g : \mathbb{N}^{k+1} \to \mathbb{N}$ la somme $\Sigma g : \mathbb{N}^{k+1} \to \mathbb{N}$ qui vaut

$$\Sigma g(m, \vec{x}) = \sum_{i=0}^{m-1} g(i, \vec{x});$$

- pour $g : \mathbb{N}^{k+1} \to \mathbb{N}$ le produit $\Pi g : \mathbb{N}^{k+1} \to \mathbb{N}$ qui vaut

$$\Pi g(m, \vec{x}) = \prod_{i=0}^{m-1} g(i, \vec{x}).$$

Par convention, $\Sigma g(0, \vec{x})$ vaut toujours 0 et $\Pi g(0, \vec{x})$ vaut toujours 1.
On appelle $\mathcal{E}$ la classe des fonctions élémentaires (ou élémentaires de Kalmár d'après le mathématicien hongrois László Kalmár).

1. Montrer que le produit binaire $\cdot : \mathbb{N}^2 \to \mathbb{N}$ est dans $\mathcal{E}$ ainsi que la fonction

$$\texttt{if}(x, y, z) = \begin{cases} y & \text{si } x > 0, \\ z & \text{sinon} \end{cases}.$$

2. Prouver que $\mathcal{E} \subseteq \mathcal{RP}$, où $\mathcal{RP}$ sont les fonctions récursives primitives.

On va montrer que $\mathcal{E}$ est une sous-classe stricte de $\mathcal{RP}$, c.-à-d. $\mathcal{E} \subsetneq \mathcal{RP}$. Soit $T : \mathbb{N}^2 \to \mathbb{N}$ la tour exponentielle

$$T(m, x) = \overbrace{2^{2^{\cdot^{\cdot^{\cdot^{2^x}}}}}}^{m}$$

avec $T(0, x) = x$.

3. Prouver que $T \in \mathcal{RP}$, et que pour tout m la fonction $T_m(x) = T(m, x)$ est dans $\mathcal{E}$.

4. Prouver les propriétés suivantes de T :

- T est strictement croissante ;
- $x \leq T(m, x)$;
- $T(m, T(n, x)) = T(m + n, x)$;
- $2x \leq T(1, x)$;
- $x^2 \leq T(1, x)$;
- $x^k \leq T(2, \max(x, k))$.

5. Montrer que $T(m, x)$ majore toutes les fonctions dans $\mathcal{E}$, c.-à-d. que pour chaque $f \in \mathcal{E}$ il existe $m_f \in \mathbb{N}$ tel que $f(\vec{x}) < T(m_f, \max(\vec{x}))$ (où le max est 0 sur une suite vide). Déduisez-en que $T \notin \mathcal{E}$.

Pour votre curiosité : la classe $\mathcal{E}$ a la propriété remarquable suivante :

$$\mathcal{E} = \mathrm{DTIME}(\mathcal{E}) = \mathrm{DSPACE}(\mathcal{E}) = \mathrm{DTIME}(T_m) = \mathrm{DSPACE}(T_m),$$

c.-à-d. que les fonctions dans $\mathcal{E}$ sont exactement celles qui sont calculables en temps ou mémoire bornés par une fonction dans $\mathcal{E}$ (ou, ce qui est équivalent, par une tour exponentielle). Ainsi, $\mathcal{E}$ contient l'immense majorité des fonctions qui intéressent habituellement les informaticiens (P et NP notamment)...

$$\frac{}{\Gamma, A \vdash A}\text{Axiom}$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\to\text{-intro} \qquad \frac{\Gamma \vdash A \to B \quad \Gamma \vdash A}{\Gamma \vdash B}\to\text{-elim}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\wedge\text{-intro} \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\wedge\text{-elim}_1 \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\wedge\text{-elim}_2$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\vee\text{-intro}_1 \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\vee\text{-intro}_2 \qquad \frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\vee\text{-elim}$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash A}\bot\text{-elim}$$

Figure 1 – Petit rappel des règles de NJ

$$\frac{}{A \vdash A}\text{Axiom}$$

$$\frac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2, A \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\text{Cut}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}W_g \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta}W_d$$

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}C_g \qquad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}C_d$$

$$\frac{\Gamma_1, B \vdash \Delta_1 \quad \Gamma_2 \vdash A, \Delta_2}{\Gamma_1, \Gamma_2, A \to B \vdash \Delta_1, \Delta_2}\to_g \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta}\to_d$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}\wedge_g \qquad \frac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2 \vdash B, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \wedge B, \Delta_1, \Delta_2}\wedge_d$$

$$\frac{\Gamma_1, A \vdash \Delta_1 \quad \Gamma_2, B \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \vee B \vdash \Delta_1, \Delta_2}\vee_g \qquad \frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta}\vee_d$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}\neg_g \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta}\neg_d$$

Figure 2 – Idem pour LK