Calcul de processus
Examen final

21 mars 2017

Durée de l'épreuve : 3 heures.
Tous les documents sont autorisés. Le barème est marqué dans chaque exercice.

Exercice 1 (6 points). On considère le langage $Imp_{\|}$ introduit en cours. Soient :

$$\begin{aligned} S &:= \quad x := y \\ P &:= \quad z := y;\ x := z \\ Q &:= \quad \mathsf{new}\ z = 0\ \mathsf{in}\ (z := y;\ x := z) \end{aligned}$$

Déterminer si les équivalences suivantes sont vraies ou fausses :

1. (**3 points**) $S \simeq^{c}_{IO} P$?
2. (**3 points**) $S \simeq^{c}_{IO} Q$?

Exercice 2 (9 points). On considère les trois processus de CCS suivants :

$$\begin{aligned} S &:= \quad a.b \\ P &:= \quad a \mid \tau.b \\ Q &:= \quad \nu c(a.\bar{c} \mid c.b) \\ R &:= \quad \nu c(a.R_0 \mid c.b) \qquad\qquad \text{où } R_0 \overset{\text{rec}}{=} \tau.(R_0 \mid \bar{c}). \end{aligned}$$

Déterminer si les bisimilarités suivantes sont vraies ou fausses :

1. (**3 points**) $P \approx S$?
2. (**3 points**) $Q \approx S$?
3. (**3 points**) $R \approx S$?

Exercice 3 (5 points). Soit

$$H \overset{\text{rec}}{=} r.b.H + g.b.H$$

Trouver un processus P de CCS de la forme

$$\nu(\widetilde{c})(K_r \mid K_g \mid K_b \mid V)$$

où K_r ne contient pas b, g, K_g ne contient pas r, b, K_b ne contient pas r, g et V ne contient pas r, g, b, tel que $P \approx H$.

Corrigé

Exercice 1.

1. On a $S \not\simeq^c_{IO} P$ (on a vu cela en TD, c'était dans le sujet d'examen de l'année passée, avec des noms de variables légèrement différents). Un contexte qui les sépare est par exemple
$$C := [\cdot] \mid z := 0.$$
En effet, si s est une mémorie dans laquelle y vaut $n \neq 0$, $(C[S], s)$ a deux exécutions qui terminent (selon qui entre $x := y$ et $z := 0$ est executé en premier), les deux terminant avec la mémoire $s[x \mapsto n, z \mapsto 0]$; en particulier, la valeur finale de x est différente de 0. En revanche, $(C[P], s)$ a une execution terminante de la forme
$$\begin{aligned}(C[P], s) &\to^* (x := z \mid z := 0, s[z \mapsto n]) \to (x := z \mid \mathsf{skip}, s[z \mapsto 0]) \\ &\to (\mathsf{skip} \mid \mathsf{skip}, s[x \mapsto 0, z \mapsto 0])\end{aligned}$$
où la valeur finale de x est 0. Cela montre que $[\![C[P]]\!]^{IO} \neq [\![C[S]]\!]^{IO}$, car $(s, s[x \mapsto 0, z \mapsto 0]) \in [\![C[P]]\!]^{IO} \setminus [\![C[S]]\!]^{IO}$.
2. On a bien $S \simeq^c_{IO} Q$, car $[\![S]\!]^{TE} = [\![Q]\!]^{TE}$. En effet, les traces de $[\![S]\!]^{TE}$ on toutes la forme
$$(s_1, s_1) \cdots (s_n, s_n)(s_{n+1}, s_{n+1}[x \mapsto k])(s_{n+2}, s_{n+2}) \cdots (s_m, s_m)$$
où k est la valeur de y dans s_{n+1}. Dans Q, l'instruction $z := y$ n'est pas visible de l'extérieur, car z est locale : on a
$$(Q, s) \to (\mathsf{new}\, z = k \,\mathsf{in}\, (\mathsf{skip}; x := z)\ ,\ s) \to (\mathsf{new}\, z = k \,\mathsf{in}\, x := z\ ,\ s),$$
c'est-à-dire, après l'exécution de l'affectation $z := y$ (on suppose que y a la valeur k dans s) et la suppression du skip qui en résulte, la variable locale z a la valeur k, mais cette valeur n'est pas visible de l'extérieur, la mémoire reste égale à s. L'exécution de cette affectation est donc assimilable à l'écution d'un skip. Après, comme z est locale, personne ne peut en modifier la valeur, donc l'affectation $x := z$ a pour effet de donner la valeur k à x, exactement comme dans l'instruction $x := y$ (car et k est la valeur de y dans s). Les traces de $[\![Q]\!]^{TE}$ ont donc la même forme que celles de $[\![S]\!]^{TE}$.

Exercice 2.

1. On a $P \not\approx S$. En effet, l'opposant a une stratégie gagnante : il suffit qu'il joue le τ dans P. Voici la vérification du fait que cette stratégie est gagnante :
   - O : $P \xrightarrow{\tau} a \mid b$
   - J : obligé de rester sur S (S n'a pas de transition τ)
   - O : $a \mid b \xrightarrow{b} a \mid \mathbf{0}$
   - J : perdu ! (S n'a pas de transition b)
2. On a $Q \approx S$. Voici une bisimulation, en utilisant les couleurs pour le couplage :

$$S \xrightarrow{a} b \xrightarrow{b} \mathbf{0} \qquad\qquad Q \xrightarrow{a} \nu c(\bar{c} \mid c.b) \xrightarrow{\tau} b \xrightarrow{b} \mathbf{0}$$

3. On a $R \approx S$. Écrivons d'abord le système de transitions étiquetées de R. Comme il est infini, il vaut mieux de commencer par fixer des notations :

$$
\begin{aligned}
T_n &:= \nu c(R_0 \mid \overbrace{\overline{c} \mid \cdots \mid \overline{c}}^{n} \mid c.b) \\
U_n &:= \nu c(R_0 \mid \overbrace{\overline{c} \mid \cdots \mid \overline{c}}^{n} \mid b) \\
V_n &:= \nu c(R_0 \mid \overbrace{\overline{c} \mid \cdots \mid \overline{c}}^{n})
\end{aligned}
$$

Voici donc le système de transitions étiquetées de R, les couleurs indiquant le couplage avec celui de S pour la bisimulation :

$$
\begin{tikzcd}
R \arrow[r, "a"] & T_0 \arrow[d, "\tau"'] & & \\
& T_1 \arrow[r, "\tau"] \arrow[d, "\tau"'] & U_0 \arrow[r, "b"] \arrow[d, "\tau"'] & V_0 \arrow[d, "\tau"'] \\
& T_2 \arrow[r, "\tau"] \arrow[d, "\tau"'] & U_1 \arrow[r, "b"] \arrow[d, "\tau"'] & V_1 \arrow[d, "\tau"'] \\
& T_3 \arrow[r, "\tau"] \arrow[d, "\tau"'] & U_2 \arrow[r, "\tau"] \arrow[d, "\tau"'] & V_2 \arrow[d, "\tau"'] \\
& \vdots & \vdots & \vdots
\end{tikzcd}
$$

Exercice 3. Voici une solution possible :

$$
\begin{aligned}
K_r &\overset{\text{rec}}{=} \overline{p}.(\overline{v}.K_r + r.\overline{c}.c.\overline{v}.K_r) \\
K_g &\overset{\text{rec}}{=} \overline{p}.(\overline{v}.K_g + g.\overline{c}.c.\overline{v}.K_g) \\
K_b &\overset{\text{rec}}{=} c.b.\overline{c}.K_b \\
V &\overset{\text{rec}}{=} p.v.V
\end{aligned}
$$

avec $\widetilde{c} = p, v, c$. On reconnaitra que V n'est rien d'autre qu'un verrou. La bisimulation est donné par le couplage suivant (indiqué par les couleurs) :

$$
\begin{tikzcd}
H \arrow[r, shift left, "r"] \arrow[r, shift right, "g"'] & b.H \arrow[l, bend right=80, "b"']
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
R_1 \arrow[r, "r"] \arrow[d, bend right, "\tau"'] & R_2 \arrow[r, "\tau"] & R_3 \arrow[r, "b"] & R_4 \arrow[r, "\tau"] & R_5 \arrow[dllll, bend left=15, "\tau"'] \\
P \arrow[u, bend right, "\tau"'] \arrow[d, bend left, "\tau"] & & & & \\
G_1 \arrow[u, bend left, "\tau"] \arrow[r, "g"] & G_2 \arrow[r, "\tau"] & G_3 \arrow[r, "b"] & G_4 \arrow[r, "\tau"] & G_5 \arrow[ullll, bend right=15, "\tau"]
\end{tikzcd}
$$

Il n'est pas nécessaire de spécifier les processus $R_1, \dots, R_5$ et $G_1, \dots, G_5$ (on peut les retrouver facilement à partir des transitions). Cependant, les voici pour complétude :

$$
\begin{array}{ll}
R_1 := \nu\widetilde{c}(\overline{v}.K_r + r.\overline{c}.c.\overline{v}.K_r \mid K_g \mid K_b \mid v.V) & G_1 := \nu\widetilde{c}(K_r \mid \overline{v}.K_g + g.\overline{c}.c.\overline{v}.K_g \mid K_b \mid v.V) \\
R_2 := \nu\widetilde{c}(\overline{c}.c.\overline{v}.K_r \mid K_g \mid K_b \mid v.V) & G_2 := \nu\widetilde{c}(K_r \mid \overline{c}.c.\overline{v}.K_g \mid K_b \mid v.V) \\
R_3 := \nu\widetilde{c}(c.\overline{v}.K_r \mid K_g \mid b.\overline{c}.K_b \mid v.V) & G_3 := \nu\widetilde{c}(K_r \mid c.\overline{v}.K_g \mid b.\overline{c}.K_b \mid v.V) \\
R_4 := \nu\widetilde{c}(c.\overline{v}.K_r \mid K_g \mid \overline{c}.K_b \mid v.V) & G_4 := \nu\widetilde{c}(K_r \mid c.\overline{v}.K_g \mid \overline{c}.K_b \mid v.V) \\
R_5 := \nu\widetilde{c}(\overline{v}.K_r \mid K_g \mid K_b \mid v.V) & G_5 := \nu\widetilde{c}(K_r \mid \overline{v}.K_g \mid K_b \mid v.V)
\end{array}
$$