Concurrency Theory (WS 2016) Out: Thu, 02 Feb Due: Wed, 08 Feb

Exercise Sheet 13

D'Osualdo, Lederer, Schneider Technische Universität Kaiserslautern

Definition Recall that $\mathcal{R}_\Delta(P)$ denotes the set of all processes reachable from P using reactions and the definitions Δ; moreover we defined in class the set

$$\mathbb{D}_k^X := \{P \mid \mathrm{depth}(P) \leq k, \mathrm{fn}(P) \subseteq X\}$$

the set of all processes with depth at most k and free names in a finite set X. A π-term P is *k-bounded* (under Δ) if $\mathcal{R}_\Delta(P) \subseteq \mathbb{D}_k^{\mathrm{fn}(P)}$. Further, P is *depth-bounded* if it is k-bounded for some $k \in \mathbb{N}$.

Problem 1: Invariants in π-calculus

In class we defined $\mathrm{Bunch}_n := \boldsymbol{\nu} a.(\boldsymbol{\nu} x.\mathsf{A}[a, x])^n$ which denotes a process parametric in $n \in \mathbb{N}$, thanks to the notation

$$P^n := \underbrace{P \parallel \cdots \parallel P}_{n \text{ times}}.$$

It is clear from its structure that Bunch_n has depth at most 2 *for every* n.

a) Show that putting arbitrarily many copies of Bunch_n together, each with a different n, still defines processes with depth at most 2. Do this by using a suitable nested variant of the P^n notation.

b) Now consider the definitions of the "Servers" example seen in the lecture:

$$\begin{aligned}
\mathsf{E}[s] &:= \tau.\boldsymbol{\nu} m.(\mathsf{C}[s, m] \parallel \mathsf{E}[s]) + \tau.\boldsymbol{\nu} s'.(\mathsf{S}[s'] \parallel \mathsf{E}[s'] \parallel \mathsf{E}[s]) \\
\mathsf{S}[s] &:= s(x).\boldsymbol{\nu} d.(\overline{x}\langle d\rangle \parallel \mathsf{S}[s]) \\
\mathsf{C}[s, m] &:= \tau.(\overline{s}\langle m\rangle \parallel \mathsf{C1}[s, m]) \\
\mathsf{C1}[s, m] &:= m(x).\mathsf{C}[s, m]
\end{aligned}$$

Using a notation similar to the one used for the previous point, find a forward inductive invariant that can be used to show that $\boldsymbol{\nu} s.(\mathsf{S}[s] \parallel \mathsf{E}[s])$ is depth-bounded.

Recall that a forward inductive invariant is a set (in this context, of processes) that is closed under reactions. If a process P is an element of a forward inductive invariant $\mathcal{I}$, then $\mathcal{R}_\Delta(P) \subseteq \mathcal{I}$.

Problem 2: Resettable Counters

The goal of this exercise is to prove the undecidability of reachability in depth-bounded π-calculus. For this purpose, we will consider resettable counters, which respond to the actions inc, dec and $reset$. Their intended behaviour is the same as reset nets counters: they can be incremented, decremented when above zero and reset to zero; but they cannot be tested for zero.

a) Give a depth-bounded π-calculus implementation of resettable counters (using finitely many definitions).

b) **OPTIONAL:** Use your encoding to show that reachability in depth-bounded π-calculus is undecidable. In other words, show that the following problem is undecidable: given a π-calculus process P which is k-bounded and a process Q, determine if $P \to^* Q$.

   [Hint: *Use the same trick we used to show undecidability of reachability for reset nets.*]

Problem 3: Terminating terms are depth-bounded

We call a process P *terminating* if it does not start an infinite reaction sequence. Show that every terminating π-calculus process is depth-bounded.

Problem 4: CCS with Bang is depth-bounded

Recall from the lecture that the bang operator satisfies $!P \equiv P \parallel !P$. The process algebra $\mathsf{CCS}^!$ is defined as follows:

$$\mathsf{CCS}^! \ni P ::= P \parallel P \;\mid\; \sum_{i\in I} \alpha_i.P_i \;\mid\; \boldsymbol{\nu} a.P \;\mid\; !P$$

Note the absence of process calls and, therefore, of definitions. The reaction rules are dictated by the same rules as the ones for CCS; the bang operator's semantics is determined by the STRUCT rule in conjuction with the bang's structural congruence law above.

Prove that every $\mathsf{CCS}^!$-term is depth-bounded.

[Hint: *In bounding the depth, try avoiding having to use structural congruence as much as possible. Keep also in mind that you only need to show* some *bound, not necessarily the most precise one.*]