Concurrency: Theory, Languages and Programming

– Pi Calculus Examples –

Session 7 – December 3, 2003

Uwe Nestmann

EPFL-LAMP

Unbounded Buffers

$$
\begin{aligned}
B(\,i,o\,) &\stackrel{\text{def}}{=} i(x).C\langle\,x,i,o\,\rangle \\
C(\,x,i,o\,) &\stackrel{\text{def}}{=} \overline{o}\langle x\rangle.B\langle\,i,o\,\rangle \\
&\quad + \; i(y).(\;C\langle\,y,i,o\,\rangle \frown C\langle\,x,i,o\,\rangle\;)
\end{aligned}
$$

where

$$
X\langle\,i,o\,\rangle \frown Y\langle\,i,o\,\rangle \stackrel{\text{def}}{=} (\boldsymbol{\nu} m)\,(\;X\langle\,i,o\,\rangle[^m/_o] \mid Y\langle\,i,o\,\rangle[^m/_i]\;)
$$

- Follow the sequence $\xrightarrow{i1}\xrightarrow{i2}\xrightarrow{\bar{o}2}\xrightarrow{\cdots}$ to convince yourself that the buffer process is indeed a buffer (FIFO) and that it can grow unboundedly.
- Note the “type” of the stored values . . .
- Note the behavior of empty cells inside a buffer “chain”.

Elastic Buffers

Make the buffer elastic,
i.e., make empty cells disappear!

Several design decisions to be taken concern the question *when* an empty cell should cut itself out of a chain and die.

- ☐ if empty cell is next to a full/empty cell?
- ☐ if empty cell is left/right to a cell?
- ☐ should it be *allowed* (suicide)
  or *forced* (murder) to die?

One goal of this exercise is to make you think about how to argue for or against that the various design decisions above lead to equivalent solutions.

Elastic Buffers: Setup

$$
\begin{aligned}
B(\,i,l,o,r\,) &\stackrel{\text{def}}{=} i(x).C_x\langle\, i,l,o,r\,\rangle \\
&+ \ldots \\
C_x(\,i,l,o,r\,) &\stackrel{\text{def}}{=} \overline{o}\langle x\rangle.B\langle\, i,l,o,r\,\rangle \\
&+ i(y).(\; C_y\langle\, i,l,o,r\,\rangle \frown C_x\langle\, i,l,o,r\,\rangle\;) \\
&+ \ldots
\end{aligned}
$$

where

$$
\begin{aligned}
(\; X \frown Y\;)\langle\, i,l,o,r\,\rangle &\stackrel{\text{def}}{=} \\
X\langle\, i,l,o,r\,\rangle \frown Y\langle\, i,l,o,r\,\rangle &\stackrel{\text{def}}{=} \\
&\ldots
\end{aligned}
$$

Elastic Buffers: cut-when-left

$$
\begin{aligned}
B &\stackrel{\text{def}}{=} (i,l,o,r).\\
&\qquad i(x).C_x\langle\, i,l,o,r\,\rangle\\
&\quad + \ \ldots\\
C_x &\stackrel{\text{def}}{=} (i,l,o,r).\\
&\qquad \overline{o}\langle x\rangle.B\langle\, i,l,o,r\,\rangle\\
&\quad + \ i(y).\big(\,(C_y \frown C_x)\langle\, i,l,o,r\,\rangle\,\big)\\
&\quad + \ \ldots
\end{aligned}
$$

Elastic Buffers: cut-when-right

$$
\begin{aligned}
B \quad &\stackrel{\text{def}}{=} \quad (i, l, o, r).\\
&\qquad i(x).C_x\langle\, i, l, o, r \,\rangle\\
&\quad + \ \ldots\\
C_x \quad &\stackrel{\text{def}}{=} \quad (i, l, o, r).\\
&\qquad \overline{o}\langle x\rangle.B\langle\, i, l, o, r \,\rangle\\
&\quad + \ i(y).\big(\ (C_y \frown C_x)\langle\, i, l, o, r \,\rangle\ \big)\\
&\quad + \ \ldots
\end{aligned}
$$