Topics in Concurrency

The syntax of parallel commands is given by:

\[
\begin{array}{c}
  c ::= X := a \mid c_0; c_1 \mid c_0 \| c_1 \mid \text{if } b \text{ then } c \mid \text{while } b \text{ do } c
\end{array}
\]

where \(X\) ranges over locations, \(a\) over arithmetic expressions, and \(b\) over boolean expressions.

\(a\) Give an operational semantics to parallel commands, assuming an operational semantics for arithmetic and boolean expressions. [5 marks]

\(b\) This part is concerned with a Petri net semantics for parallel commands.

There are to be two kinds of conditions: data conditions, pairs of locations and integers, which specify the contents of locations, and control conditions, which specify the local control points in parallel components of commands.

A parallel command is to be represented by a basic net (where every condition has capacity one) in which a subset of control conditions \(I\) is to be distinguished as its initial conditions and another subset \(T\) is to be distinguished as its terminal conditions; the initial conditions are precisely those control conditions which hold at the start of execution of the command; the terminal conditions are precisely those control conditions which hold if and when the command terminates.

A diagrammatic account suffices for answers to the questions below.

\(i\) Describe an (infinite) net for \(X := X + 1\). [2 marks]

\(ii\) Describe a construction on nets for \(c_0; c_1\). [Hint: Replace the terminal conditions \(T_0\) of \(c_0\) and the initial conditions \(I_1\) of \(c_1\) with their product \(T_0 \times I_1\).] [4 marks]

\(iii\) Describe a construction on nets for \(c_0 \| c_1\). [2 marks]

\(iv\) Describe a construction on nets for \(\text{if } X > 0 \text{ then } c\). [2 marks]

\(v\) Describe a construction on nets for \(\text{while } X > 0 \text{ do } c\). [5 marks]