

Homework 5

EE 290n - Advanced Topics in Systems Theory

Edward A. Lee

1. Suppose V is some set and $S = V^{**}$ is the set of finite and infinite sequences of elements of V . This exercise explores some of the properties of the CPO S^n with the pointwise prefix order, for some non-negative integer n .
 - (a) Show that any two elements $a, b \in S^n$ that have an upper bound have a least upper bound.
 - (b) Let $U \subset S^n$ be such that no two distinct elements of U are joinable. Prove that for all $s \in S^n$ there is at most one $u \in U$ such that $u \sqsubseteq s$.
 - (c) Given $s \in S^n$, suppose that $Q(s) \subset S^n$ is a joinable set where for all $q \in Q(s)$, $q \sqsubseteq s$. Then show that there is an s' such that $s = (\bigvee Q(s)).s'$.

2. Consider the model shown in figure 1. Assume that data types are all $V = \{0, 1\}$. Assume f is a dataflow that implements an identity function and that Const is an actor that produces an infinite sequence $(0, 0, 0, \dots)$. Obviously, the overall output of this model should be this same infinite sequence. The box labeled g indicates a composite actor. Find firing rules and firing function g for the composite actor to satisfy conditions 1 and 3 covered in class. Note that the composite actor has one input and two outputs.

3. **Extra credit.** In theory, dataflow models with only boolean data types, switch, select, and logic functions are Turing complete. A simple function that should be implementable, but is not easy to implement using such primitives, is one that, given a sequence (v_1, v_2, \dots) produces a sequence where every block of five inputs is reversed, yielding

$$(v_5, v_4, v_3, v_2, v_1, v_{10}, v_9, \dots).$$

I am looking for elegant dataflow models using the dynamic dataflow (DDF) director in Ptolemy II (under ExperimentalDirectors). An extension of this would use integer data types

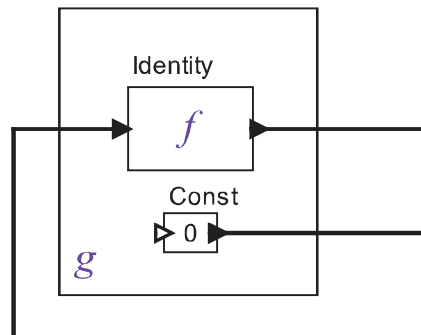


Figure 1: A model.

and given three sequences $v = (v_1, v_2, \cdot)$, (n_1, n_2, \cdot) , and (m_1, m_2, \cdot) that would behave as follows: for every integer $i > 0$, it would consume n_i tokens from v and push them onto a stack, then pop m_i tokens from the stack (reversing their order) and produce them on the output. I am looking for an elegant dataflow model that performs this function. Note that I do not have a solution to this problem.