Concurrent Models of Computation for Embedded Software

Edward A. Lee
Professor, UC Berkeley
EECS 290n – Advanced Topics in Systems Theory
Fall, 2004

Lecture 14: Dataflow Process Networks

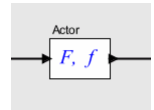
Firings

Dataflow is a variant of Kahn Process Networks where a process is computed as a sequence of atomic *firings*, which are finite computations enabled by a *firing rule*.

In a firing, an actor consumes a finite number of input tokens and produces a finite number of outputs.

A possibly infinite sequence of firings is called a *dataflow process.*

Firing Rules



Let $F : S^n \to S^m$ be a dataflow process.

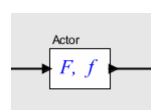
Let $U \subset S^n$ be a set of *firing rules* with the constraints:

1. Every $u \in U$ is finite, and
2. No two elements of U are joinable.

This implies that for all $s \in S^n$ there is at most one $u \in U$ where $u \sqsubseteq s$. (exercise)

When $u \sqsubseteq s$ there is a unique s' such that $s = u.s'$ where the period denotes concatenation of sequences.

Firing Function



Let $f : S^n \to S^m$ be a (possibly partial) firing function with the constraint that for all $u \in U$, $f(u)$ is defined and is finite.

Then the dataflow process $F : S^n \to S^m$ is given by

$$F(s) = \begin{cases} f(u).F(s') & \text{if there is a } u \in U \text{ such that } s = u.s' \\ \perp_n & \text{otherwise} \end{cases}$$

where $\perp_n \in S^n$ is the n-tuple of empty sequences.

Note that this is self referential. Seek a fixed point F.

Fixed Point Definition of Dataflow Process (cf. Lifting Formulation in SR)

Define $\phi : [S^n \to S^m] \to [S^n \to S^m]$ by:

$$(\phi(F))(s) = \begin{cases} f(u).F(s') & \text{if there is a } u \in U \text{ such that } s = u.s' \\ \bot_n & \text{otherwise} \end{cases}$$

Fact: ϕ is continuous (see handout). This means that it has a unique least fixed point, and that we can constructively find that fixed point by starting with the bottom of the CPO. The bottom of the CPO is the function $F_0 : S^n \to S^m$ that returns $\bot_n$.

Executing a Dataflow Process is the Same as Finding the Least Fixed Point

Suppose $s \in S^n$ is a concatenation of firing rules,

$$s = u_1 . u_2 . u_3 \quad \dots$$

Then the procedure for finding the least fixed point of ϕ yields the following sequence of approximations to the dataflow process:

$$F_0(s) = \bot_n$$

$$F_1(s) = (\phi(F_0))(s) = f(u_1)$$

$$F_2(s) = (\phi(F_1))(s) = f(u_1).f(u_2)$$

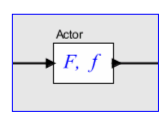
$$\dots$$

This exactly describes the operational semantics of repeated firings governed by the firing rules!

The LUB of this Sequence of Functions is Continuous

The chain $\{F_0(s), F_1(s), \ldots\}$ will be finite for some s (certainly for finite s, but also for any s for which after some point, no more firing rules match), and infinite for other s. Since each F_i is a continuous function, and the set of continuous functions is a CPO, then the LUB is continuous, and hence describes a valid Kahn process that guarantees determinacy, and can be put into a feedback loop.

Example 1

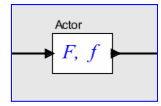


Suppose $V = \{0, 1\}$ and $S = V^{**}$ is the set of finite and infinite sequences of elements from V.

Consider a dataflow process with one input and one output, $F : S \to S$. Its firing rules are $U \subset S$. The following are all valid firing rules:

$$U = \{\perp\}$$
$$U = \{(0)\}$$
$$U = \{(0), (1)\}$$
$$U = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$$

Example 2 : Valid Firing Rule?



Suppose $V = \{0, 1\}$ and $S = V^{**}$ is the set of finite and infinite sequences of elements from V.

Consider a dataflow process with one input and one output, $F : S \to S$. Its firing rules are $U \subset S$. Is the following set a valid set of firing rule?

$$U = \{\perp, (0), (1)\}$$

Example 2 : Valid Firing Rule?

Suppose $V = \{0, 1\}$ and $S = V^{**}$ is the set of finite and infinite sequences of elements from V.

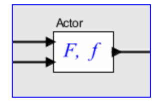
Consider a dataflow process with one input and one output, $F : S \to S$. Its firing rules are $U \subset S$. Is the following set a valid set of firing rule?

$$U = \{\perp, (0), (1)\}$$

No. There are joinable pairs.

Intuition: The same input sequence can lead to multiple executions. Nondeterminacy!

Example 3



Consider $F : S^2 \to S$. Its firing rules are $U \subset S^2$. Which of the following are valid sets of firing rules?

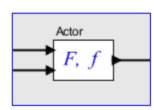
$$\{((0), (0)), ((0), (1)), ((1), (0)), ((1), (1))\}$$

$$\{((0), \bot), ((1), \bot), (\bot, (0)), (\bot, (1))\}$$

$$\{((0), \bot), ((1), (0)), ((1), (1))\}$$

$$\{((0), \bot), ((1), \bot)\}$$

Example 3



Consider $F : S^2 \to S$. Its firing rules are $U \subset S^2$. Which of the following are valid sets of firing rules?

$$\{((0), (0)), ((0), (1)), ((1), (0)), ((1), (1))\}$$

Yes. Consume one token from each input.

$$\{((0), \bot), ((1), \bot), (\bot, (0)), (\bot, (1))\}$$

No. Nondeterminate merge.

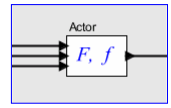
$$\{((0), \bot), ((1), (0)), ((1), (1))\}$$

Yes. Consume from the second input if the first is 1.

$$\{((0), \bot), ((1), \bot)\}$$

Yes. Consume only from the first input.

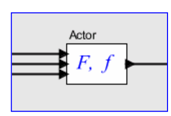
Example 4



Consider $F : S^3 \to S$. Its firing rules are $U \subset S^3$. Is the following a valid set of firing rules?

$$\{((1), (0), \perp), ((0), \perp, (1)), (\perp, (1), (0))\}$$

Example 4



Consider $F : S^3 \to S$. Its firing rules are $U \subset S^3$. Is the following a valid set of firing rules?

$$\{((1), (0), \perp), ((0), \perp, (1)), (\perp, (1), (0))\}$$

Yes. Dataflow version of the Gustave function!

Conclusions and Open Issues

- Dataflow processes are Kahn processes composed of atomic *firings*.
- Firing rules that are not joinable lead to simple fixed point semantics.