Homework 2 Solutions

Distributed Systems

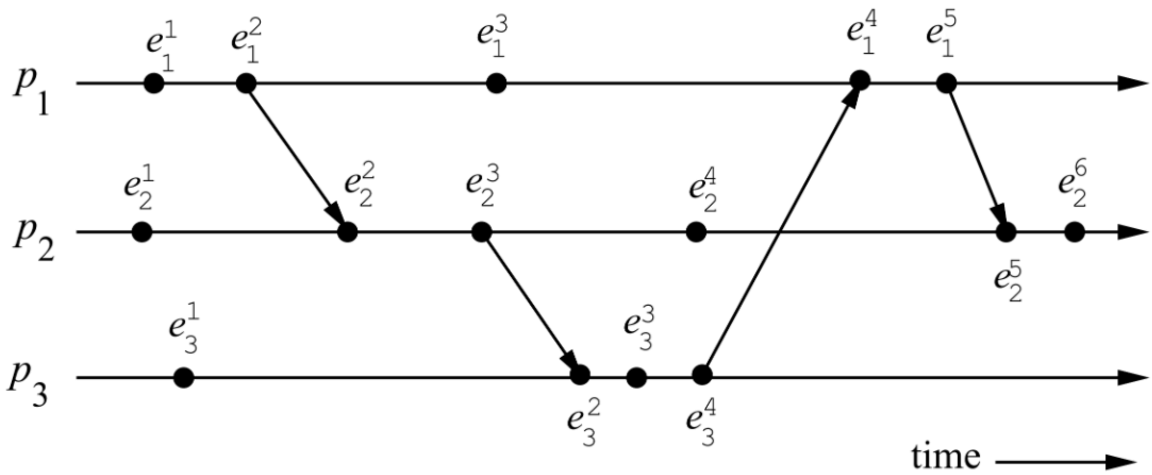
Monsoon 2024

Question 1. Consider a system with four processors and a distributed program with about 15 events.

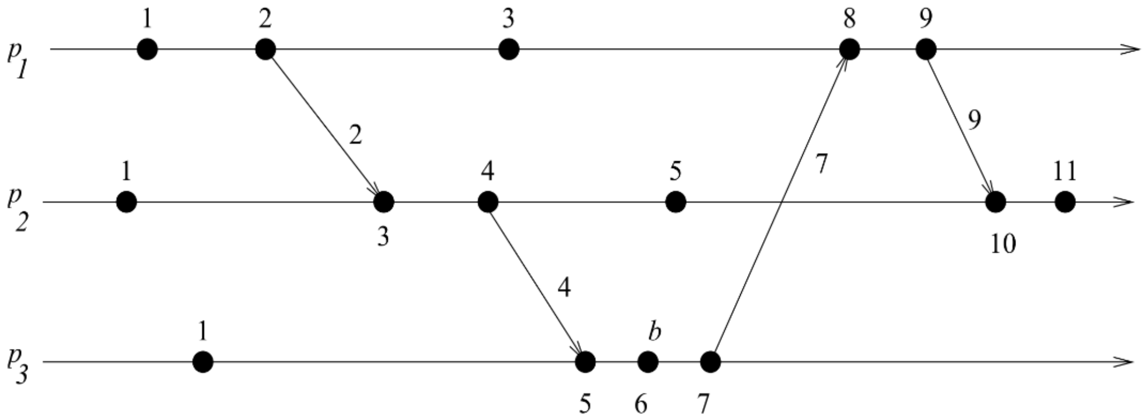
(a) **Draw the time diagram for such a program and mark the scalar time of the events. The set of events should include all possible events.**
(b) **Identify events that are logically concurrent.**
(c) **Identify events where strong monotonicity of the scalar time fails.**

Question 2. Repeat Question 1(a) and 1(b) with respect to vector time.

Solution of question 1 and 2:



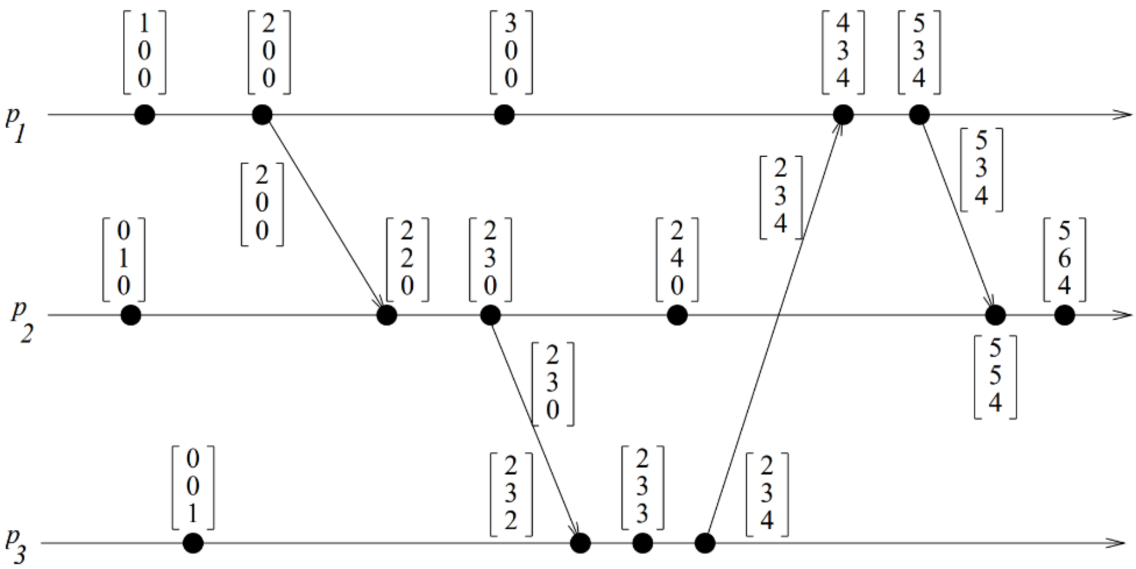
1a) Scalar time:



1b) Since scalar time is not strongly consistent we cant say which events are logically concurrent just from the scalar times

1c) Consider the third event at p1(e_1^3 , scalar time 3) and p2 (e_2^3 , scalar time 4), we cant say e_2^3 occured before e_1^3, they are actually logically concurrent)

2a)



2b) Example of logically concurrent events:

- e_1^3 and e_2^3 (vector times {3,0,0} and {2,3,0}. In first row $e_1^3 > e_2^3$, and in second row $e_1^3 < e_2^3$, so we they are concurrent)

Question 3. Show that the timestamps assigned by vector time are strongly consistent.

Solution:

Strong Consistent means $e_i \rightarrow e_j$ if and only if $C(e_i) < C(e_j)$

Case 1: if $e_i \rightarrow e_j$ then $C(e_i) < C(e_j)$

Events e_i and e_j may be directly causal or there might be events inbetween them which are directly causal such as $e_i \rightarrow e_{k1} \rightarrow e_{k2} ... \rightarrow e_{km} \rightarrow e_j$ in either case, by defintion(see relevant rules) of vector clocks for any two directly causal events $e_a \rightarrow e_b$, then $C(e_a) < C(e_b)$

Relavant rules: Rule 1, and step 2 in Rule 2:

- Rule 1: Before executing an event, process p_i updates its local logical time as follows:

  $$vt_i[i] := vt_i[i] + d \quad \text{where } (d > 0)$$

- Rule2: Each message m is piggybacked with the vector clock vt of the sender process at sending time. On the receipt of such a message (m,vt), process p_i executes the following sequence of actions:
  1. Update its global logical time as follows:
  2. $1 \leq k \leq n : vt_i[k] := \max(vt_i[k], vt[k])$
  3. Execute Rule1.
  4. Deliver the message m.

Case 2: if $C(e_i) < C(e_j)$ then $e_i \rightarrow e_j$

$C(e_i) < C(e_j)$:

- for all indices k: $C(e_i)[k] \leq C(e_j)[k]$
- and atleast for one index L, $C(e_i)[L] < C(e_j)[L]$

If the process in which both events e_i and e_j occur is same, then $C(e_i)[L] < C(e_j)[L]$ is enough to say that they are causally related. (Rule 1), so $e_i \rightarrow e_j$.

Now, let the event e_i be from process P1 and e_j from process P2. The only time process P2 will update the value $C(e)$[1] (for some event e on P2) is when it receives a message from P1 or from an event that is caused by an event on P1. Now, since $C(e_j)$[1] >= $C(e_i)$[1], we conclude that P2 had increased the value $C(e_j)$[1], meaning that at some point it must have received a message that is either e_i or from an event that is caused by e_i. Therefore, $e_i \rightarrow e_j$.