The Model

Node properties:
- No shared memory
- No global clock

Channel properties:
- FIFO
- loss free
- non-dupli-cat-ing

The Problem

\[ C_1: \text{empty} \rightarrow C_2: \text{empty} \]
\[ $500 \rightarrow $200 \]

\[ C_1: \text{transfer $50} \rightarrow C_2: \text{empty} \]
\[ $450 \rightarrow $200 \]

\[ C_1: \text{empty} \rightarrow C_2: \text{empty} \]
\[ $450 \rightarrow $250 \]

Distributed Snapshot (Global State Recording)

Problems:
- recording a “consistent” state of the global computation
- checkpointing for fault tolerance (rollback, recovery)
- testing and debugging
- monitoring and auditing
- detecting stable properties in a distributed system via snapshots.
  A property is “stable” if, once it holds in a state, it holds in all
  subsequent states.
  - termination
  - deadlock
  - garbage collection

Definitions

Local State and Actions:

- local state: \( \text{LS}_i \)
- message send: \( \text{send}(m_{ij}) \)
- message receive: \( \text{rec}(m_{ij}) \)
- time: \( \text{time}(x) \)
- \( \text{send}(m_{ij}) \in \text{LS}_i \text{ iff time(\text{send}(m_{ij})) < time(\text{LS}_i)} \)
- \( \text{rec}(m_{ij}) \in \text{LS}_j \text{ iff time(\text{rec}(m_{ij})) < time(\text{LS}_j)} \)

Predicates:

- \( \text{transit}(\text{LS}_i, \text{LS}_j) = \{ m_{ij} | \text{send}(m_{ij}) \in \text{LS}_i \land \text{rec}(m_{ij}) \in \text{LS}_j \} \)
- \( \text{inconsistent}(\text{LS}_i, \text{LS}_j) = \{ m_{ij} | \text{send}(m_{ij}) \in \text{LS}_i \land \text{rec}(m_{ij}) \in \text{LS}_j \} \)

Consistent Global State:

\( \forall i, \forall j : 1 \leq i, j \leq n : \text{inconsistent}(\text{LS}_i, \text{LS}_j) = \emptyset \)

Global-State-Detection Algorithm

**Marker-Sending Rule for a Process p:**

For each channel \( c \), incident on, and directed away from \( p \): \( p \) sends one marker along \( c \) after \( p \) records its state and before \( p \) sends further messages along \( c \).

**Marker-Receiving Rule for a Process q:**

if \( (q \) has not recorded its state) then

begin \( q \) records its state;
\( q \) records the state of \( c \) as the empty sequence;
end
else \( q \) records the state of \( c \) as the sequence of message
received along \( c \) after \( q \)’s state was recorded and before
\( q \) received the marker along \( c \).

**Detecting a Stable Property**

begin
record a global snapshot, \( S^* \);
test for the stable property in \( S^* \);
end;
Snapshot/State Recording Example

\[500 \quad p \quad c_1 \quad c_2 \quad q \quad 500\]

\[r\]

\[c_3 \quad c_4\]

\[M = \text{Marker}\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>{}</td>
</tr>
<tr>
<td>q</td>
<td>{}</td>
</tr>
<tr>
<td>r</td>
<td>{}</td>
</tr>
</tbody>
</table>

---

Snapshot/State Recording Example (Step 1)

\[490 \quad p \quad c_1 \quad 20 \quad c_2 \quad q \quad 470\]

\[r\]

\[c_3 \quad c_4\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>490 {}</td>
</tr>
<tr>
<td>q</td>
<td>{}</td>
</tr>
<tr>
<td>r</td>
<td>{}</td>
</tr>
</tbody>
</table>

---

Snapshot/State Recording Example (Step 2)

\[470 \quad p \quad 20 \quad c_1 \quad c_2 \quad q \quad 480\]

\[r\]

\[c_3 \quad c_4\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>490 {}</td>
</tr>
<tr>
<td>q</td>
<td>480 {}</td>
</tr>
<tr>
<td>r</td>
<td>{}</td>
</tr>
</tbody>
</table>

---

Snapshot/State Recording Example (Step 3)

\[470 \quad p \quad 20 \quad c_1 \quad c_2 \quad q \quad 480\]

\[r\]

\[c_3 \quad c_4\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>490 {}</td>
</tr>
<tr>
<td>q</td>
<td>480 {}</td>
</tr>
<tr>
<td>r</td>
<td>485 {}</td>
</tr>
</tbody>
</table>

---

Snapshot/State Recording Example (Step 4)

\[490 \quad p \quad 25 \quad c_1 \quad c_2 \quad q \quad 500\]

\[r\]

\[c_3 \quad c_4\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>490 {20}</td>
</tr>
<tr>
<td>q</td>
<td>480 {}</td>
</tr>
<tr>
<td>r</td>
<td>485 {}</td>
</tr>
</tbody>
</table>

---

Snapshot/State Recording Example (Step 5)

\[515 \quad p \quad 485 \quad c_1 \quad c_2 \quad q \quad 500\]

\[r\]

\[c_3 \quad c_4\]

<table>
<thead>
<tr>
<th>Node</th>
<th>Recorded state</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>490 {20}</td>
</tr>
<tr>
<td>q</td>
<td>480 {25}</td>
</tr>
<tr>
<td>r</td>
<td>485 {}</td>
</tr>
</tbody>
</table>