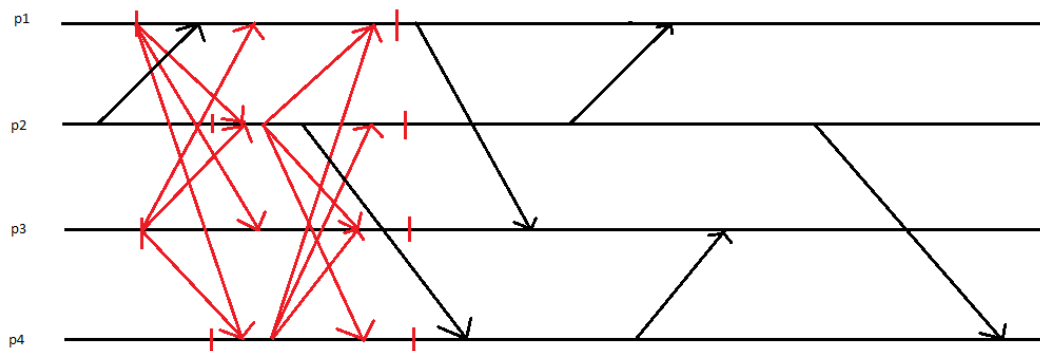


Der Chandy-Lamport-Algorithmus erzeugt auch dann konsistente Snapshots, wenn mehrere Prozesse ihn starten.

Im Bild unten sind die zwischen den Prozessen versendeten Nachrichten schwarz, die Markernachrichten rot und die roten vertikalen Striche deuten den Start eines Snapshots an. Es wird von einem vollständig vernetzten Prozessgraphen ausgegangen, sodass jeder Prozess jedem anderen Prozess Nachrichten schicken kann.



p1 und p3 starten simultan einen Snapshot. Sie verschicken auf allen ausgehenden Kanälen Markernachrichten und nehmen ihren Zustand auf. p2 und p4 empfangen die Markernachrichten etwas später, nehmen ebenfalls ihren Zustand auf und schicken auf allen Kanälen Markernachrichten. Damit ist der Snapshot dann auch schon beendet, da damit jeder Prozess auf jedem Eingangskanal eine Markernachricht empfangen hat und den Snapshot abschließt. Der Snapshot ist konsistent, es wird keine happens-before Relation verletzt, also keine Nachricht vor dem Snapshot empfangen, die erst im Snapshot versendet wird. Es wird auch im Snapshot keine Nachricht empfangen, die erst danach verschickt wird.

Das gleiche gilt auch für den Fall, dass jeder Prozess einen Snapshot startet. Wenn alle Prozesse gleichzeitig einen Snapshot starten, ist dieser sehr schnell beendet, wenn man davon ausgeht, dass jeder Kommunikationskanal die gleiche Zeit benötigt, um eine Nachricht zu versenden. Dann ist der Snapshot beendet, sobald die Markernachrichten bei den Prozessen ankommen, da bereits jeder Prozess eine solche Nachricht versendet hat. Daraus folgt, dass auf jedem eingehenden Kanal jedes Prozesses ein Marker ankommt, damit wird der Snapshot beendet.

Da dies relativ analog zum obigen Bild geschieht, wird auf ein weiteres Bild verzichtet.

Für die obigen Angaben wird davon ausgegangen, dass sich die Markernachrichten nicht unterscheiden lassen.

Wenn sie sich unterscheiden ließen, würden die Snapshots unabhängig voneinander ausgeführt. Da der Chandy-Lamport Algorithmus für einzelne Snapshots schon Konsistenz garantiert, wären weitere Ausführungen hier nicht nötig.