TECHNISCHE
UNIVERSITÄT
DARMSTADT

TK1: Distributed Systems - Programming & Algorithms

Prof. Dr. Max Mühlhäuser
Dr. Immanuel Schweizer

Jens Heuschkel, MSc.
Michael Stein, MSc.

TELEKOOPERATION
Fachbereich Informatik
Hochschulstr. 10
64289 Darmstadt

9. Theory Assignment
Submission Date: 3.02.2016

By handing in a solution you confirm that you are the exclusive author(s) of all the materials. Additional information can be found here: https://www.informatik.tu-darmstadt.de/de/sonstiges/plagiarismus/

Task 1: 'Snapshot'-Algorithm of Chandy and Lamport (5P)

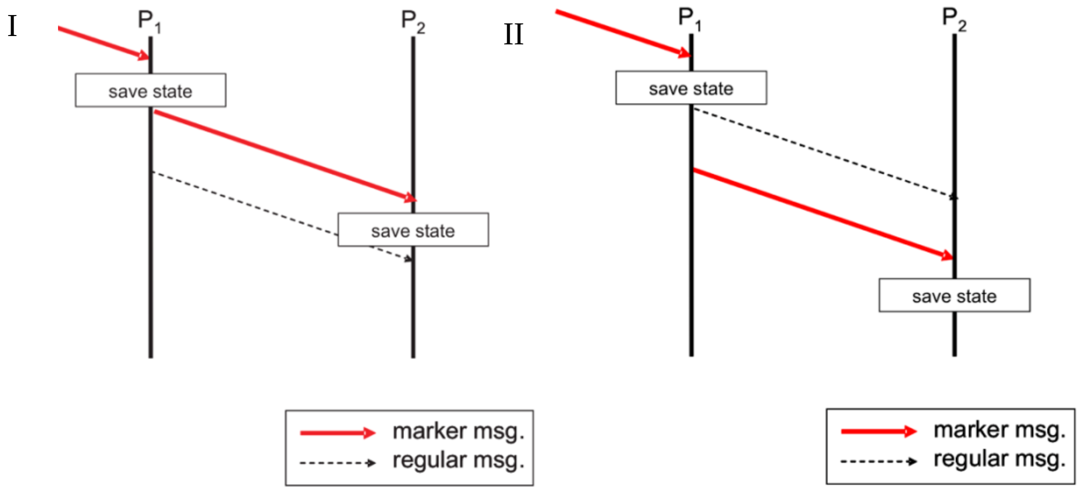

Figure 1: Synchronization of two processes using the 'snapshot'-algorithm by Chandy and Lamport.

a) Figures 1.I and 1.II both show two processes P_1 and P_2, whose common state will be determined by using the 'snapshot'-algorithm by Chandy and Lamport. For this, the *marker* messages and the *save state* actions are illustrated in figure 1. Which illustration (i.e. 1.I or 1.II) is correct? Explain your decision and give examples for both figures (one example per figure). The examples should show that the recorded state reflects a consistent or inconsistent recorded state, respectively.

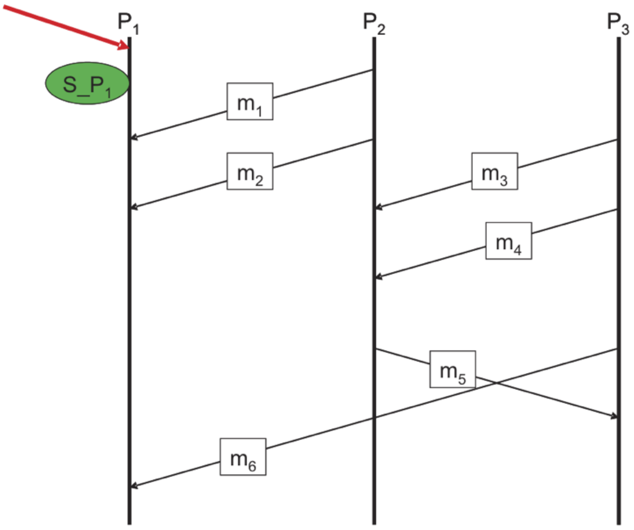

Figure 2: Synchronization of three processes with the 'snapshot'-algorithm by Chandy and Lamport.

b) Figure 2 shows the processes P_1, P_2 and P_3, and the messages m_1, m_2, …, m_6. Process P_1 receives a *marker* message and records its state in S_P_1. Specify two possible recorded states, which might exist upon termination of the algorithm. For both states, sketch the recorded marker messages, as well as the states S_P_2 and S_P_3. A communication channel from process P_i to process P_j is to be designated as $C(P_i,P_j)$.

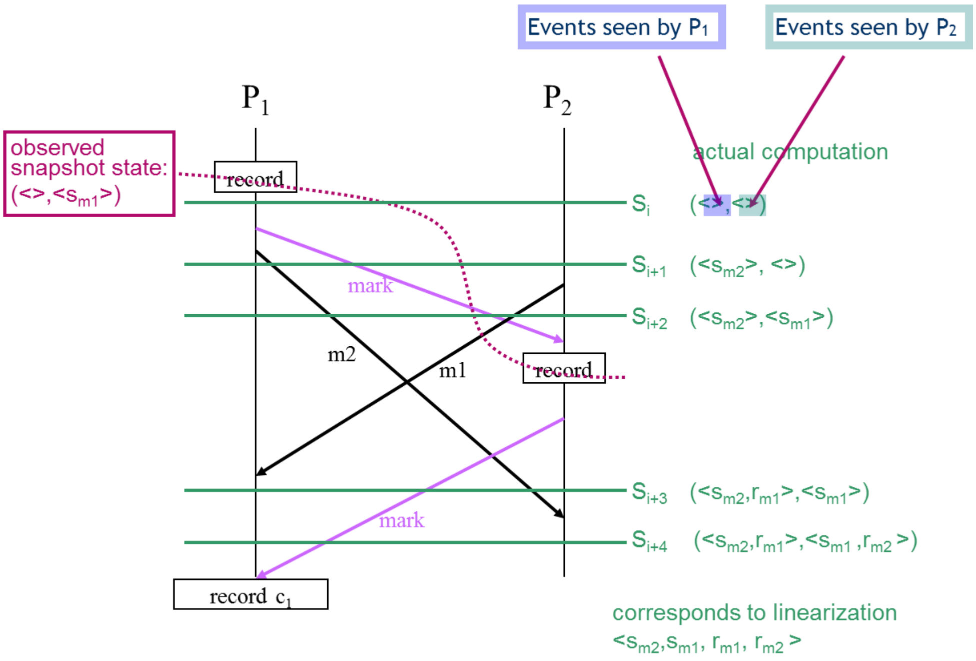

Figure 3: An example of a snapshot after applying the algorithm by Chandy and Lamport.

Task 2: Snapshot vs. Actual Program Flow (5P)

Figure 3 exemplifies the algorithm by Chandy and Lamport as discussed in the lecture. In addition to the real communication (messages m_1 and m_2), figure 3 depicts both *marker* messages and the recording of the states. The actual global system states are shown on the right, whereas the snapshot state is shown on the left.

Let $Sys = s_{m2}, s_{m1}, r_{m1}, r_{rm2}$ be a consistent run that transforms state S_i to S_{i+4}. Furthermore, let $S_{init} = S_i$ and $S_{final} = S_{i+4}$.

a) Determine the permutation $Sys' = e'_0, e'_1, e'_2, \ldots$, such that S_{snap} can be reached from S_{init} and S_{final} from S_{snap}, respectively specify S_{snap}, as well as both *pre-snap* and *post-snap* events.

b) Sketch the communication flow, which arises from Sys'.