Institute of Computer Science, University of Bern
Distributed Algorithms, Spring 2020

Prof. Christian Cachin
Jovana Mićić

Exercise 8

8.1 Total-order broadcast using consensus (4pt)

Consider Algorithm 6.1 (Consensus-Based Total-Order Broadcast) and observe that the payload messages proposed to consensus are represented as a set.

a) What happens if the messages decided by consensus are not sorted deterministically before *tob-delivering* them?

b) Suppose one leaves out the sorting of decided payload messages. How can one modify the algorithm so that every process nevertheless outputs the same ordered sequence of messages?

8.2 Atomic register as a replicated state machine (4pt)

One of the simplest state machines is the register abstraction discussed earlier. Implement an (N, N)-atomic register using a primitive for total-order broadcast.

The replicated state is the current value of the register and the relevant commands are *write*(v) and *read*$() \to v$. Use Module 4.3 [CGR11] and show that the implemented register satisfies the respective properties.

8.3 Replicated register with local read (2pt)

The atomic-register implementation of Ex. 8.2 sends every operation through total-order broadcast. Describe a modification of the protocol, where *reads* are executed locally, without being *tob-broadcast* to all processes. One often finds this optimization in practical systems. Does the resulting register still satisfy *atomicity*? Either argue why the property holds or demonstrate an execution that violates atomicity.

References

[CGR11] C. Cachin, R. Guerraoui, and L. Rodrigues, *Introduction to reliable and secure distributed programming (Second Edition)*, Springer, 2011.