I2DS24 exercise 2

Wolfgang J. Paul

March 3, 2024

1 Formalizing multisets (20 Pt)

Formalize $S \in \mathbb{M}(\mathcal{M})$ as a mapping

$$S : \mathcal{M} \rightarrow \mathbb{N}$$

where $S(m) = n$ means, that message m is in multiset S exactly n times.[1] How do you formalize for multisets S and T then

1. S contains the single message m. $S(x) = ?$
2. $S \cup T$. $(S \cup T)(x) = ?$
3. $S \cap T$
4. $S \setminus T$

2 Simulating asynchronous communication by synchronous communication (20 + 20 Pt)

We denote components X of the system with asynchronous communivation as on the slides and their counterparts (if existing) in the simulating system with synchronous communication by $\tilde{X}$.

[1] Recall that we denote with $\mathbb{N}$ the natural numbers including 0.

- As set of states we choose

$$\tilde{Z}_p = Z_p \times \mathbb{M}(\mathcal{M})$$

  i.e. we include the simulated message buffer in the state set of the simulating process.

- for a send transition

$$c \rightarrow_p^s (x,m,d) \text{ with } q = link_p(x)$$

  and corresponding receive transition

$$(e,m) \rightarrow_q^r f$$

  of the simulated system we choose for all S,T

$$(S,c) \tilde{\rightarrow}_p^s (x,m,S) \text{ and } ((e,T),m) \tilde{\rightarrow}_q^r (f, T \cup \{m\})$$

Specify for the simulating systems

1. the send relation $\tilde{\rightarrow}_p^s$
2. the internal transition relation $\tilde{\rightarrow}_p^s$. Hint: you need rules for the simulated internal steps and for the simulated receive steps
3. state a simulation theorem between computations of the two systems (20 Bonus Points). Hint: don't forget to couple the the start configurations of the systems in the simulation relation.

3 Schedules (20 Pt)

Consider the system from exercise 1 of sheet 1. Prove or disprove

- in every run of the system and for every natural number n we eventually have $n \in S$

4 Norm Functions (20 Pt)

Lemma 2 from the slides: There is a typo on the slides. The following condition is part of the definition of norm functions.

- if E terminates in a state γ, the $P(\gamma)$ holds in that state.

Now prove

- Let f be a norm function for predicate P. Then for each execution of E of system S predicate $P(\gamma)$ holds in some configuration of E.

5 Making the leader known

Modify the deterministic leader election algorithm with UIDs on a ring, such that

- computations on all nodes terminate (not necessarily at the same time).
- each node knows the UID of the elected leader at the time when its computation has terminated.

For ring size N estimate

1. the total number of messages sent
2. the number of steps until the last nodes terminates.