Exercise 10: Consensus

Task 1: More Value(s)

In this task, we have a look at *multivalued consensus,* which is like binary consensus, except that inputs and outputs are from a larger domain $[O]$ for some $2 < O \in \mathbb{N}$. Our goal is an efficient reduction to binary consensus.

a) Give a 1-round algorithm with the following property. If all correct nodes have the same input, each node outputs this value. Otherwise, there is a single value $o \in [O]$ such that all nodes output either o or 0.

b) We replace all inputs with the output computed in a). Then we use binary consensus to decide whether to use output 0 or output o. What is the problem with this naive approach?

c) Fix the problem from b) in a way that still guarantees that if *all* correct nodes had opinion o after a), then the binary consensus routine will decide in favor of o. Exploit that if this is not the case, it is perfectly fine to output 0!

d) Plug these results together to obtain the desired reduction. Show that it costs $\mathcal{O}(1)$ rounds with messages of size $\mathcal{O}(\log O)$.

e*) If $f \ll n$, this reduction costs $\omega(fn)$ messages. Can you adapt the solution so that $\mathcal{O}(fn)$ messages (and $\mathcal{O}(fn \log O)$ bits) are sent by correct nodes in total?

Task 2: As Fast as it Gets

Consider the task of binary consensus with up to $f < n$ crash faults.

a) Describe and prove correct an $(f + 1)$-round algorithm using 1-bit broadcasts for communication.

Task 3*: Timing Issues

In this task, the goal is to transfer synchronous algorithms to the bounded-delay model with faults. Therefore, the setting is the same as in Chapter 4 of the lecture.

a) Use the Srikanth-Toueg algorithm to *simulate* synchronous execution of some given R-round synchronous algorithm in $\mathcal{O}(Rd)$ time, assuming that the execution is triggered by events at the individual nodes that are at most $\mathcal{O}(Rd)$ time apart (for a known bound).

b) Can you formalize what the term "simulates" here means precisely?

c) Things get a bit messy with randomization, as there typically are additional model assumptions needed for randomization to be useful. Figure out what these might be and whether this poses a problem!

d) Agree on your findings in the TA session!