ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

SS 2004 Prof. R. Wattenhofer / Dr. C. Cachin / F. Kuhn / R. O'Dell

Principles of Distributed Computing
Exercise 8: Sample Solution

1 Multi-Valued Agreement

The following protocol implements asynchronous multivalued Byzantine agreement according to Definition 9.8 (relaxed for randomized protocols). It uses digital signatures and calls a (randomized) protocol for binary Byzantine agreement.

Algorithm 1 Multi-Valued Agreement

```
upon propose(v):
send the signed message (value, v) to all servers

upon receiving 2t + 1 messages (value, *) with proper signatures:
let m be the value v that occurs most often in the received value messages
let Π be the set of received value messages
send the message (majority, m, Π) to all servers

upon receiving n − t messages (majority, *) with valid proofs:
if all values m in the received majority messages are the same then
   let M be the majority value
   propose 1 for binary agreement
else
   propose 0 for binary agreement
end if

upon deciding for b in binary agreement:
if b = 1 then
   decide for M
else
   decide for a default value
end if
```

The protocol satisfies the standard validity condition because if all honest servers propose the same value, all honest servers obtain a unique m, all valid *majority* messages contain m, and all honest servers propose 1 for binary agreement.

Agreement and termination follow from a standard argument and from the properties of the binary agreement protocol.

2 Strong Agreement

The standard validity condition of Binary Byzantine agreement requires a particular outcome only if *all* honest servers propose the same value; but the complement is that some honest server proposed the opposite value, hence any decision "makes sense" because some honest server proposed it.

Let D denote the agreement domain with m values and $H \subset D$ the set of values proposed by the honest servers. The values in H are called *valid*. Towards a contradiction, suppose that $n \leq (m+1)t$ and $|H| = m-1$. Let the set of all honest servers be partitioned into A and B such that $|A| \leq (m-1)t$ and $|B| = t$, such that for every $v \in H$ there are at most t servers who propose v.

The adversary now causes all corrupted servers to follow the protocol with the invalid input $u \in D \setminus H$. The adversary isolates the servers in B by delaying all messages from servers in B. Then the servers in A must reach agreement together with the corrupted servers. But since the corrupted servers follow the protocol, they cannot be distinguished from honest servers and the protocol will decide on u with some non-negligible probability. Since u is not valid, this contradicts strong validity.

Reference: M. Fitzi and J. A. Garay. Efficient player-optimal protocols for strong and differential consensus. In *Proc. 22nd ACM Symposium on Principles of Distributed Computing (PODC)*, 2003.