

# Cryptographic Protocols

## Exercise 9

### 9.1 Consensus: An Example

Four players  $P_1, \dots, P_4$  execute the **Consensus** protocol from the lecture with  $t=1$ , where  $P_1$  is corrupted and has the following strategy: in any step of the protocol where the players are instructed to send a value to all other players,  $P_1$  always sends the value 1 to  $P_2$  and  $P_3$ , and the value 0 to  $P_4$ . Below, you can find a table with the outputs of the players in the sub-protocols in an execution of the **Consensus** protocol in the above setting.

	$P_1$	$P_2$	$P_3$	$P_4$
Input	–	1	0	0
WeakConsensus	–	⊥	⊥	0
GradedConsensus	–	(0, 0)	(0, 0)	(0, 0)
KingConsensus $_{P_1}$	–	1	1	0
WeakConsensus	–	1	1	⊥
GradedConsensus	–	(1, 1)	(1, 1)	(1, 0)
KingConsensus $_{P_2}$	–	1	1	1

a) Fill, similarly to the above example, the following tables (where the strategy of  $P_1$  is as described above):

**Scenario 1:**

	$P_1$	$P_2$	$P_3$	$P_4$
Input	–	1	1	0
WeakConsensus	–			
GradedConsensus	–			
KingConsensus $_{P_1}$	–			
WeakConsensus	–			
GradedConsensus	–			
KingConsensus $_{P_2}$	–			

**Scenario 2:**

	$P_1$	$P_2$	$P_3$	$P_4$
Input	–	1	1	1
WeakConsensus	–			
GradedConsensus	–			
KingConsensus $_{P_1}$	–			
WeakConsensus	–			
GradedConsensus	–			
KingConsensus $_{P_2}$	–			

- b) Can  $P_1$  make the honest parties output 0 in Scenario 1? Describe the corresponding strategy for  $P_1$  or justify why such a strategy does not exist. What about Scenario 2?
- c) Assume that  $P_1, P_2$ , and  $P_3$  all have input 1 and  $P_4$  has input 0. Use the properties of the sub-protocols to show that when at most one player is corrupted, then the construction

WeakConsensus; GradedConsensus; KingConsensus $_{P_4}$

achieves the properties of Consensus.

## 9.2 Variations of GradedConsensus

- a) Amélie has an idea for improving the **GradedConsensus** protocol from the lecture: in Step 3, each player  $P_j$  computes the value  $y_j$  as follows:

$$y_j = \begin{cases} 0 & \text{if } \#\text{zeros} \geq n - t, \\ 1 & \text{otherwise.} \end{cases}$$

What do you think about this suggestion? Prove or disprove whether the new protocol achieves **GRADED CONSENSUS**.

- b) Cindy also has a suggestion, where  $y_j$  is computed as follows:

$$y_j = \begin{cases} 0 & \text{if } \#\text{zeros} > t, \\ 1 & \text{if } \#\text{ones} > t, \\ \text{random} & \text{otherwise.} \end{cases}$$

What do you think about Cindy's suggestion? Is the new protocol well-defined? Prove or disprove whether the new protocol achieves **GRADED CONSENSUS**.

- c) Hans has yet another idea: compute  $y_j$  as in the protocol from the lecture, but  $g_j$  is computed as follows:

$$g_j = \begin{cases} 1 & \text{if } \#y_j\text{'s} > n/2, \\ 0 & \text{otherwise.} \end{cases}$$

Prove or disprove whether Hans' protocol achieves **GRADED CONSENSUS**.

## 9.3 Two-Threshold Consensus

In the lecture, we have seen a consensus protocol based on the phase-king paradigm. In each of the phases, the persistency and consistency properties hold simultaneously as long as at most  $t$  parties are corrupted, for any  $t < \frac{n}{3}$ .

In this exercise, we consider separate thresholds for persistency and for consistency. More precisely, we want the protocol to achieve persistency as long as at most  $t_p$  parties are corrupted ( $t_p$ -persistency), and consistency as long as at most  $t_c$  parties are corrupted ( $t_c$ -consistency).

- a) Consider the weak consensus protocol presented in the lecture, where  $t$  can be seen as a protocol parameter:

**Protocol WeakConsensus**  $(x_1, \dots, x_n) \rightarrow (y_1, \dots, y_n)$ :

1.  $\forall P_i$ : send  $x_i$  to each  $P_j$
2.  $\forall P_j$ :  $y_j = \begin{cases} 0 & \text{if } \#\text{zeros} \geq n - t \\ 1 & \text{if } \#\text{ones} \geq n - t \\ \perp & \text{else} \end{cases}$
3.  $\forall P_j$ : return  $y_j$

For a fixed  $t$ , find the thresholds  $t_p$  such that the above protocol achieves  $t_p$ -persistency. Repeat the analysis for  $t_c$ -consistency. Give a condition on  $t_p$  and  $t_c$  such that one can set the parameter  $t$  to achieve simultaneously  $t_p$ -persistency and  $t_c$ -consistency.

- b) (Optional) Analyze the protocol graded consensus with separate thresholds  $t_p$  for persistency and  $t_c$  for consistency.