

## Byzantine agreement

### The Byzantine agreement problem

- ▶ In the Byzantine agreement problem,  $n$  processors (or, say, Byzantine generals, ...) communicate with each other in order to reach an agreement on a binary value  $b$ .
- ▶ There are bad processors that may collaborate with each other in order to prevent an admissible agreement.  
At most a fraction of  $1/8$  of all processors are bad.
- ▶ Each processor has an initial binary value.  
The agreement must reflect to a certain extent the majority among the initial values. More precisely, the processors must reach an agreement that is *admissible* in the following sense.
  - All good processors must agree on the same value  $b$ .
  - In case all the good processors have the same initial value, then  $b$  must be equal to this value.

## Byzantine agreement

### The rules for the Byzantine agreement problem

- ▶ The communication is done in rounds.
- ▶ At the beginning of each round, each processor sends messages to all other processors.
- ▶ Processor  $i$  sees only the messages sent to Processor  $i$ .  
The messages sent by a processor to different receivers might differ.
- ▶ Before each round, the bad processors may agree on an arbitrarily complex pattern of messages for this round.
- ▶ At the beginning, the good processors know neither the bad processors nor their strategy.

The rationale for supposing collaboration among the bad processors is that a protocol that succeeds against concerted attacks is likely to succeed in the presence of random or unrelated faults.

## Byzantine agreement

### Protocol ByzantineAgreement (Processor $i$ )

Input: A binary value  $b_i$ .

Fix constants  $t_0 = \frac{5}{8}n$  and  $t_1 = \frac{6}{8}n$  and let  $v_i(1) = b_i$ .

For rounds  $s = 1, 2, \dots$  do the following.

Send  $v_i(s)$  to all other processors.

For all  $j \neq i$ , receive  $v_j(s)$  from Processor  $j$ .

For  $l = 0, 1$ , let  $c_l = |\{j : v_j(s) = l\}|$ .

If  $c_0 \geq c_1$  then  $u(s) = 0$  and  $c(s) = c_0$ ,  
else  $u(s) = 1$  and  $c(s) = c_1$ .

(The most frequent value among  $v_1(s), \dots, v_n(s)$  is  $u(s)$  and  $c(s)$  is its count.)

Obtain  $\tau(s) \in \{0, 1\}$  by tossing a fair coin.

(The random bit  $\tau(s)$  is the same for all processors.)

If  $c(s) \geq t_{\tau(s)}$ , then  $v_i(s+1) = u(s)$  else  $v_i(s+1) = 0$ .

If  $c(s) \geq \frac{7}{8}n$ , then assume an agreement on  $u(s)$  and  
let  $v_i(s+1) = v_i(s+2) = \dots = u(s)$ .

## Byzantine agreement

### Proposition

*Let a group of  $n$  processors communicate where all but at most  $n/8$  processors obey the Protocol ByzantineAgreement. Then an admissible agreement is reached with probability 1 and in an expected number of rounds that is constant.*

### Remark.

For deterministic protocols to solve the Byzantine agreement problem, matching lower and upper bounds are known:

under the given assumptions, any deterministic protocol will require at least  $n/8 + 1$  rounds in worst case,

there is a deterministic protocol that reaches an agreement in at most  $n/8 + 1$  rounds.

(for references see the corresponding section in the monograph by Motwani and Raghavan).

The proposition is immediate from the three following claims.

## Byzantine agreement

### Claim I

In case all good processors have the same initial value, an agreement on this value is reached at the end of the first round.

### Proof of Claim I.

In case all good processors have the same initial value  $b$ , then in the first round all good processors send  $b$  to all other processors. So we have for each good processor  $u(1) = b$  and  $c(1) = c_b \geq \frac{7}{8}n$ , hence the processor assumes an agreement on  $b$ .  $\square$

## Byzantine agreement

### Claim II

Let  $s$  be minimum such that at the end of round  $s$  some good processor assumes an agreement. Then all good processors assume an agreement on the same value at the end of round  $s + 1$ .

### Proof of Claim II.

Pick some processor that assumes an agreement on  $b$  at the end of round  $s$ . In round  $s$ , by minimality of  $s$  we have for this processor

$$c_b = c(s) \geq 7/8 n ,$$

hence at least  $\frac{6}{8}n$  good processors must have sent  $b$ .

Again by minimality of  $s$ , in round  $s + 1$  then each good processor will send  $b$ , assuming either an agreement on  $b$  or no agreement.

Consequently, all good processors will have reached an agreement on  $b$  at the end of round  $s + 1$ .  $\square$

## Byzantine agreement

### Claim III

Suppose no good processor assumes an agreement during any of the rounds 1 through  $s - 1$ . Then with probability at least  $1/2$  some processor assumes an agreement in round  $s + 1$ .

### Proof of Claim III.

If some good processor assumes an agreement at the end of round  $s$ , then we are done by Claim II, so we can assume otherwise. Then it suffices to show that with probability at least  $1/2$  in round  $s + 1$  all good processors send the same bit.

## Byzantine agreement

### Proof of Claim III (continued).

Let  $k_0$  and  $k_1$  be the number of good processors that send 0 and 1, respectively, during round  $s$  and let  $k = \max\{k_0, k_1\}$ .

We distinguish two cases and in both cases consider round  $s$ .

Case A:  $k < 5/8 n$ .

We have  $k_0, k_1 \leq k < 5/8 n$ .

Hence no matter what messages the at most  $n/8$  bad processors send, each processor will receive strictly less than  $6/8 n$  messages containing the same bit.

But with probability  $1/2$  the threshold  $t_{\tau(s)}$  will be equal to  $t_1 = 6/8 n$ , in which case all good processors send 0 in round  $s + 1$ .

## Byzantine agreement

Proof of Claim III (continued).

Case B:  $k \geq 5/8 n$ .

Choose  $b$  such that  $k = k_b$ .

No matter what messages the at most  $n/8$  bad processors send, each processor will receive at least  $5/8 n$  messages containing bit  $b$ .

But with probability  $1/2$  the threshold  $t_{\tau(s)}$  will be equal to  $t_0 = 5/8 n$ , in which case all good processors send  $b$  in round  $s + 1$ .

□