General Byzantine consensus

Impossibility results

Reference: [Lyn96], Sections 6.4, 6.5.

Theorem: Byzantine consensus is not solvable in complete, synchronous network of n processes if number of possible failures is $\geq \frac{n}{3}$

Proof: Assume you can solve this problem i.e algorithm A exists. Will show that we would then be able to solve problem for $n = 3, f = 1$, contradicting previous theorem.

Design algorithm for 3 processes by having each one simulate $\leq \lceil \frac{n}{3} \rceil$ threads of A. (We'll call the simulated processes "threads" to distinguish between the two systems.)

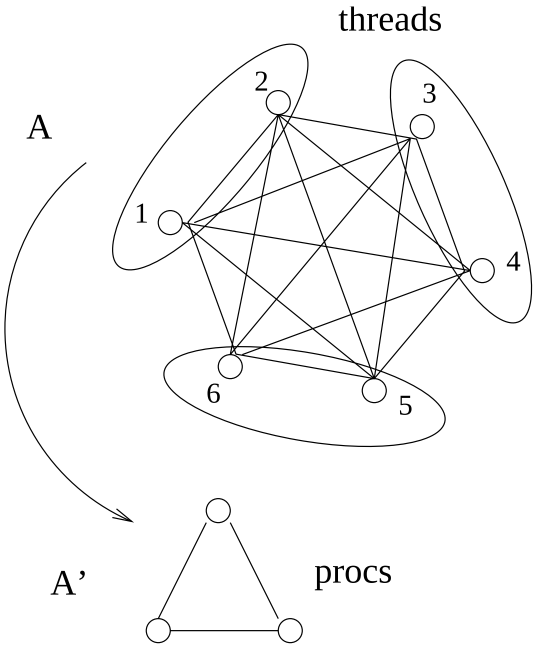

Figure 1: Comparing A and A' algorithm

The simulating process does all the internal computation of threads and sends a copy of any message the threads would send to the process simulating the recipient threads.

When process P simluates a bunch of threads (=simulated processes) the input values of those threads will be input value of P.

When threads produce an output (all will be the same) the process simulating those threads outputs that value.

We assumed A solves concensus if up to f $\geq \frac{n}{3}$ processes fail.

We want to show that the algorithm A' works if 1 process fails. If 1 process fails in A' then number of threads that start misbehaving in A is $\leq \lceil \frac{n}{3} \rceil \leq f$. Thus the simulated threads must still terminate satisfying weak agreement and weak validity.

Therefore A' must also correctly solve consensus for up to 1 failure - an impossibility.

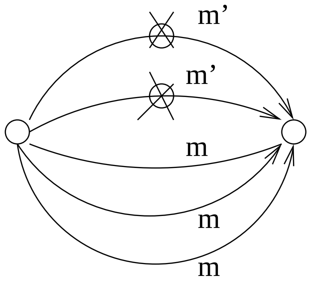

Figure 2: Graph connectivity

Question: What if the network is not complete?

The connectivity of a graph is defined as the minimum number of nodes that must be removed in order to disconnect the graph.

Theorem: Byzantine consensus is impossible if $f \geq \frac{c}{2}$ where f is number of possible failures and c is connnectivity of the graph.

Proof: First let's look at the particluar graph on the left of the figure 3, which has connectivity of 2. We will show Byzantine consensus is impossible in this graph when $f = 1$.

Suppose we have an algorithm for the 4-process system that tolerates one failure. Imagine running the algorithm in a system of 8 processes with the input values as shown in the right half of Figure 3.

To P, Q and S, it looks as though they are running a system with a faulty process R, so they must output 0 (by weak validity). Similarly, P', Q' and S' must output 1 (by weak validity). To processes P', R and S, it appears that the processes are running with a faulty process Q, so the three processes must all output the same result (by weak agreement). This contradicts the fact that S outputs 0 and P' outputs 1.

Generalizing this to complete the proof of the theorem is your task in Homework Exercise 3.

Algorithm for Byzantine consensus

We now give algorithm for Byzantine consensus in synchronous system, complete graph.

Reference: Berman and Garay, Mathematical Systems Theory, 1993 or [AW98], Section 5.2.5.

Assume number of failures $f < \frac{n}{4}$. Note that is also possible (but harder) to show give an algorithm that works under the weaker assumption that $f < \frac{n}{3}$ (See Garay & Moses, SIAM J. Computing, 1998.)

An algorithm outline:

1. $f + 1$ phases
2. each phase has two steps
3. each process keeps track of its preference for output (initially use own input value)

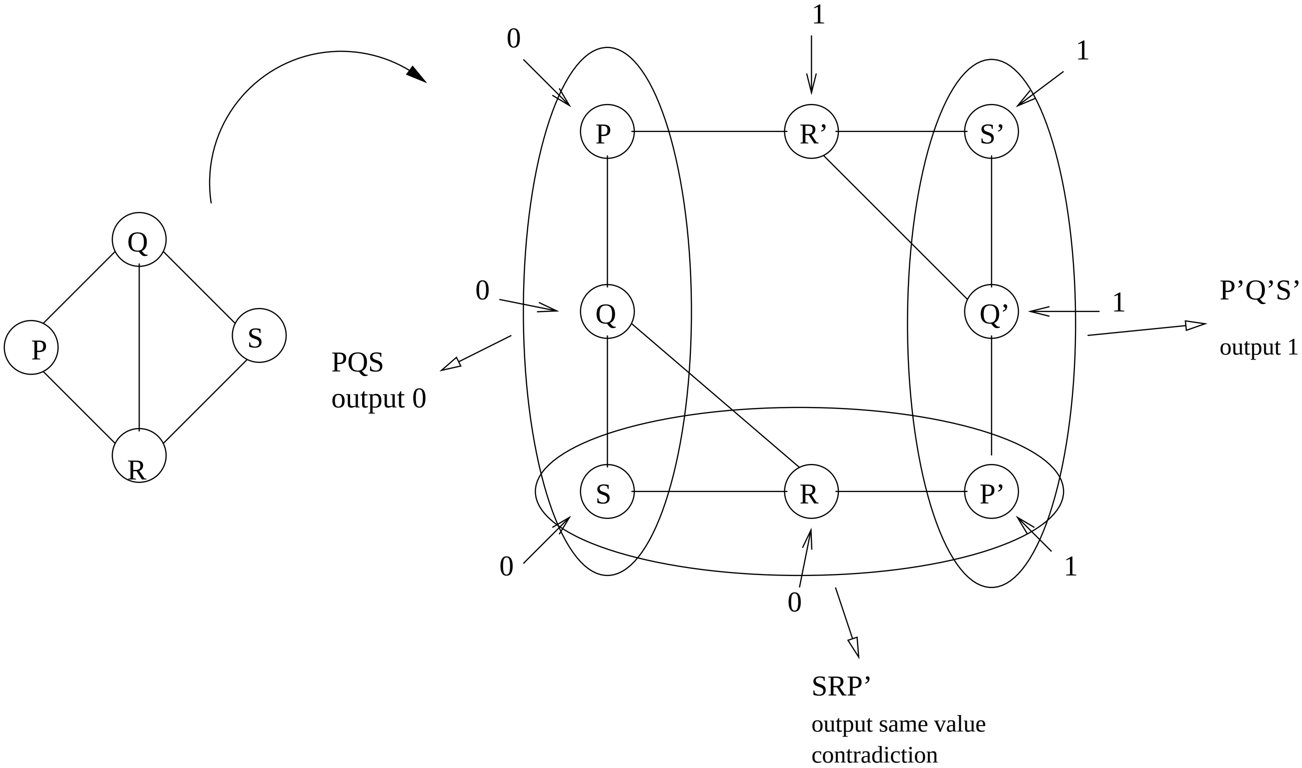

Figure 3: Impossibility of Byzantine consensus

First step of each phase - every process sends its preference to every process (including itself).

Second step of each phase k - only process k sends messages to everyone, containing the majority value it got in previous step. If no majority value, send arbitrary value , say k's input value.

Each process updates its preference on each phase as follows:

If process gets $> f + \frac{n}{2}$ copies of the same value in step 1 it uses that value as new preference, else use value received in second step.

Lemma: If all correct processes have same preference at some point they never change it thereafter.

Proof: Number of correct processes $\geq n - f > f + \frac{n}{2}$ (Since $n - f > f + \frac{n}{2} \iff \frac{n}{2} > 2f \iff n > 4f$)

If all correct processes agree on their preferred value, x, at the beginning of a phase, they will all send x to every process in step one of the phase. Thus, each process will receive more than $f + \frac{n}{2}$ copies of x in step 1 of the phase, so every correct process will set its preference to x at the end of the phase.

Weak validity follows - if every correct process starts with same input no correct process will ever change it's preference ad they will all output that value.

We run the algorithm for f+1 phases. So a correct process will do step 2 of at least one of the phases. We show next day that the weak agreement property follows.