9-26-19

5. Distributed Consensus despite Faulty Links.

Each process has a local input value v (can be either 0 or 1)
After executing a protocol (consensus) or a distributed algorithm they must agree on a decision value, satisfying the following 3 conditions:

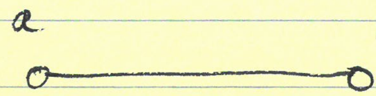
① Termination
② Agreement: All processes agree on the same decision value.
③ Validity: If all processes start with input value 0 then 0 is the only possible decision value.
If all processes start with input value 1, then 1 is the only possible decision value

No failures? No problem: many ways to solve this.:
Leader can collect all input values
or
All processes broadcast their input values

Apply a function on the n input values (max, AND, OR, majority--)

Link Failures?

processes are non-faulty
Faulty link drops messages



2 processes connected by a link that may be faulty.

<u>Theorem 5.1</u> G is a 2-node, 1-link graph.
~~There~~ There does not exist an algorithm that solves the coordinated attack (consensus) problem.

<u>Proof:</u> By contradiction; Assume Algorithm A exists.

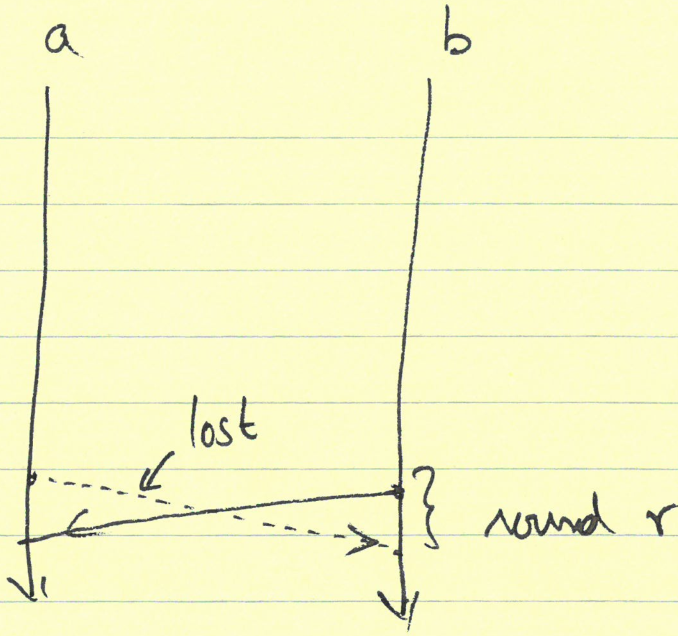
Run algorithm A. Both terminate after r rounds.

Consider run α when both start with 1 and all messages are delivered, ~~at~~ Both terminate after r rounds.
Both decide on 1.

α_1 = Same as α except all messages after r rounds are lost.

In α_1 both decide on 1.

3



α_2 = Same as α_1 except ~~that~~ the message sent by a to b in round r is lost.

$\alpha_1 \overset{a}{\approx} \alpha_2$; a decided on 1 in α_1;
a decides on 1 in α_2
(runs α_1 & α_2 are identical to process a)
$\Rightarrow$ b decides on 1 in α_2
(agreement)

α_3: message to a in round r is lost but is same as α_2 otherwise.

$\alpha_2 \overset{b}{\approx} \alpha_3$ (runs α_2 & α_3 are identical from b's point of view)

b decides on 1 ~~in~~ in α_3 (like in α_2)
a . . . . . 1.

⋮

α' : Both start with 1 and all messages are lost

Both decide on 1 in α'

α'' : a starts with 1,
b 0
all messages are lost

$\alpha' \overset{a}{\approx} \alpha''$ a decides on 1 in α''
b .. 1 in α''
(agreement condition)

α''': Both start with 0 & all messages are lost

$\alpha'' \overset{b}{\approx} \alpha'''$: b decides on 1 in α'''
a ... 1 in α'''
(agreement)

violation of validity condition. $\blacksquare$

What do we do?

Randomized Version

Allow probability of error:
~~Run~~ Run for r rounds.

Algorithm: Process 1 (a "leader") chooses a key randomly in the range 1.. r (uniformly) process 1 sends key in every message.

key is piggybacked on all messages

Initial values of all processes known to "me" is piggy backed on all messages.

V : Value vector (input value)
L: Level vector (Level of knowledge about other processes)

r rounds:
- Send (L, V, key) in each round
- when a message containing (L', V', key) is received, update the local variable L, V, key

After r rounds.
the process decides on 1 if ~~key~~ its Level $\geq$ key and all input values are 1
else decide on 0.

Level? How do I find my Level at a round?

<u>9-26-19:</u> Quiz #2.
After FloodMax till today