Advanced Network Security 2017

Agreement & consensus - crash failures
Homework Lecture 2, Question 3

In $[i]$, $\text{num}[i] = 0$, $\text{num}[k] = 0$

Choosing $[i] = \text{true}$

\[
\text{num}[i] \leq 1 \\
\text{max}_{j \neq i} \text{num}[j] \\
\Rightarrow \text{max}_{j \neq i} \text{num}[j] + \text{num}[i] = 1
\]

Choosing $[i] = \text{false}$

\[
\text{node } L: \quad \text{num}[i] = 0 \\
\text{node } i: \quad \text{num}[k] = 0
\]

So $k$ will also enter C.S.
Consensus

input → 0

\{0, 1\} → attach

\text{or attack}

\text{or attack}

0 → input

genralz

need to agree on a plan when to attach.

\underline{agreement}: all generals to decide on the same value
Faults / Failures
- shopping / crash failures: work doesn’t work
- omission failures:
- Byzantine failures:

Diagram:

Sender: I = 1
0: I = 0

1: receivers
0
Decision problems

- private input $C[i].in$
- private output $C[i].decision$

**Termination conditions** (for correct processors)
- deterministic
- probabilistic — decide $p=1$
- implicit / stabilising decision value can change → becomes stable after a while

**Consistency condition**
- global predicate over all input & all decisions

**Context**
- topology $G = (V, E) \ n = |V|$
- we assume certain types of faults
- we assume $f < n$ faulty nodes at most → sometimes a fixed processor is assumed not to fail
- we assume senders & receiver know their identities (2 know this about messages)
Decision problem #1: replicated server

- Two nodes p, q hold the same data/input
- Consistency condition:
  - All correct processors decide on the input of p and q
- Deterministic termination
- Crash failures
- At most one of p and q fails

$p(q) \quad \text{for all } r \in \{p, q, 3\} \text{ do send input to } r$

$\text{C[}p, q, 3\text{].decision} \leftarrow \text{C[}p, q\text{].in}$

$\text{receive } u \leftarrow \text{C[}r, j\text{].decision} \leftarrow u$
Decision problem #2: weak broadcast

- one server $p$ holds an input bit $\{0,1\}$
- consistency condition:
  - all non-faulty processors decide on $p$'s input
- termination condition:
  - stabilising
- assumption:
  - crash failures

TBD next lecture
The consensus problem

- all processors have a binary input value

- consistency
  - all correct processors decide on the same value (agreement)
  - if all processors have the same input, then all correct processors decide on that value (validity)

- termination: deterministic
Solve consensus, using atomic/strong broadcast

initialise vector \( V[p] \)
\[ V[p] \leftarrow C[p], \text{ in broadcast } C[p], \text{ in for all } r \neq p \text{ do receive } V[r] \]

\( C[p], \text{ decision } \leftarrow \text{ majority in } V[r] \)

So assume:
- no broadcast!
- \( f < n \) crash failures
- synchronous protocol
  - at the start of round \( r \), processors send their values
  - before going to round \( r+1 \), processors receive all messages sent to them in round \( r \).
Consensus: main approach

- each processor is going to build a tree $T_p$ which is going to contain values (inputs) from other processors, together with "votes"

$\Rightarrow v_1, v_2, \ldots, v_{10} \rightarrow$ processor 9, 1, 101 told processor 9, 101, ... to a processor 92 that 9, 1 value is $v$

Initially $v^p = CPJ, in$

$V_e = CPJ, in$

$T_p$

$V_e \leftarrow$ to all other processors

$V^p \rightarrow 2, 3, \ldots, n$ except $p$
- before round 1
  - initialise the tree; set all $v_0^p = 1$, $v_{-1}^p = V[p].$in

- round $r$: $1 \leq r \leq f+1$
  - for all $\sigma$ with $|\sigma| = r-1$ and $p \neq \sigma$
    - send $v_0^p$ to all other processors $q$
  - receive all $m_0^p$ addressed to $p$ and
    - store this in the tree as $v_0^p$

- tell me that $x$ tells me that $v_0^p$
  
  $\sigma = \sigma_0: \sigma_1: \ldots$
How to decide on a value after $f+1$ rounds, when the tree $T_p$ is fully constructed.

Let $\hat{V_p} = \{ v \mid v = v_P \in T_p \land v \neq \bot \}$

$\hat{v}$, only possible inputs

$V_p$ could be: $\{3, -3, 13, -13, 0, 1\}$

**Decision rule** is

- If $|V_p| = 1$ → decide on $V$

($V_p = \{v3\}$)

- If not → decide on a default value $v_{def}$ (say $\phi$)
Agreement: $V_p = V_q$ for all non-faulty $p, q$

**Lemma**: Suppose $p, q$ correct. Then if $v \in V_p$, then $v \in V_q$.

**Proof**:
- if $v \in V_p$ then $v = v^p_\sigma$ for some $\sigma$ s.t. $p \in \sigma$
- if $|\sigma| < f+1$
  
  In this case $p$ will send $M_{\sigma;p}^q = V^p_\sigma$ to $q$. Therefore $v^q_\sigma; p = v^p_\sigma = v \implies v \in V_q$
- if $|\sigma| = f+1$ then there is a non-faulty $z$ s.t.
  
  $\sigma = \alpha \cup z \cup \beta$

  $V^p_\sigma = v \implies V^z_\alpha = v$

  $V^p_\alpha; z; \beta \implies V^z_\alpha \implies v^q_\alpha; z; \beta \implies \vdots$

  In round $|\sigma| + 1$, $z$ sub $v^q_\alpha; z$ to $q \implies v^q_\alpha; z; \beta$
Agreement follows

if $|V_p| > 1$ then $|V_q| > 1 \to$ both decide on $v_{def}$

if $|V_p| = 1$ \quad $V_r = V_q = \{v\} \to$ both decide on $v$