These are the notes of the Distributed Computing course of Monday October 1 (3 pm to 6pm).

The 6 following pages are the notes of the course.
The 2 last pages are the notes of the exercise session.

About the course:

If you found it difficult, here are the important points to retain:

- The notion of connectivity (we will use it in the next course)
- Proof by induction
- Proof by contradiction
- Basic probabilistic calculation (the "Random failures" part)

About the exercise session:

We will finish this exercise during the next exercise session. Forget about the last 10 minutes ("STEP 2" of the proof), I will redo it in a more clear way.

As I said, we did all the "easy things" during the course. This is the simplest exercise I could find with only crash failures, but it is relatively complicated. You do not need to memorize all the steps of reasoning, of course. This is just to give you an example of "longer proof" in distributed computing (some proofs can take 10 or 20 PDF pages in research papers).

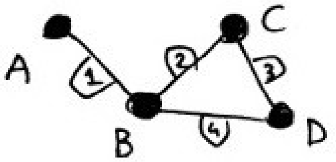
1 SETTING

graph (V, E)

set of nodes — set of edges

(1 node = 1 process)

example:



$V = \{A, B, C, D\}$

$E = \{ \underbrace{\{A, B\}}_{1}, \underbrace{\{B, C\}}_{2}, \underbrace{\{C, D\}}_{3}, \underbrace{\{B, D\}}_{4} \}$

- 2 nodes p and q are neighbor if $\{p, q\} \in E$

- $X \subseteq V$: set of crashed nodes (the other nodes are correct nodes)

- Path : sequence of nodes $(p_1, p_2, \dots, p_m)$ such that, $\forall i \in \{1, \dots, m-1\}$, p_i and p_{i+1} are neighbors

- 2 nodes p and q are connected if there exists a path $(p_1, \dots, p_m)$ such that $p = p_1$ and $q = p_m$.

ALGORITHM

each node p holds a message m_p and a set $p.R$

GOAL: for two nodes p and q, to have $(p, m_p) \in q.R$ and $(q, m_q) \in p.R$

(2) (We then say that p and q "communicate reliably")

ALGO for each node p:

(1) initially:

- send (p, m_p) to all neighbors.

(2) when receives a tuple (v, m):

- add (v, m) to $p.R$ ($p.R := p.R \cup \{(v,m)\}$)
- send (v, m) to all neighbors.

PRELIMINARY RESULTS

"If p and q are connected by a path $(p_1, \dots, p_m)$ of correct nodes, then p and q communicate reliably."

PROOF: proof by induction.

→ We want to prove a property $\mathcal{P}_k$, $\forall k \in \{1, \dots, m\}$.

(1) prove that $\mathcal{P}_1$ is true

(2) suppose $\mathcal{P}_k$ true, for $k \in \{1, \dots, m-1\}$

→ prove that $\mathcal{P}_{k+1}$ is true.

(3) conclusion: $\mathcal{P}_k$ is true $\forall k \in \{1, \dots, m\}$

HERE:

$\mathcal{P}_k$: "p_k receives (p, m_p)"

(for $k \in \{2, \dots, m\}$)

(1) prove $\mathcal{P}_2$:

ATTA = According To The Algorithm

③ ATTA, $p = p_1$ initially sends (p, m_p) to p_2, so p_2 receives (p, m_p) from p_1

→ $\mathcal{P}_2$ is true.

(2) suppose $\mathcal{P}_k$ true for $k \in \{2, \dots, m-1\}$: "p_k receives (p, m_p)"

Then, ATTA, p_k sends (p, m_p) to p_{k+1}

→ p_{k+1} receives (p, m_p) from p_k

→ $\mathcal{P}_{k+1}$ is true

(3) conclusion:

$\mathcal{P}_m$ is true:

$p_m = q$ receives (p, m_p)

→ ATTA, $(p, m_p) \in q.R$

Symmetrical proof (starting from p_m): $(q, m_q) \in p.R$

→ p and q communicate reliably (the result)

$\boxed{2}$ CONNECTIVITY

DEF: disjoint paths

- let p and q be two nodes;
- let $(p_1, \dots, p_{\underline{m}})$ and $(q_1, \dots, q_{\underline{m}})$ be two paths connecting p and q;

$(p = p_1 = q_1$ and $q = p_m = q_m)$

→ these 2 paths are disjoint if

$\{p_1, \dots, p_m\} \cap \{q_1, \dots, q_m\} = \{p, q\}$

(p and q are the only two nodes in common)

(4) DEF: connectivity

the graph is "k-connected" if $\forall \{p, q\} \subseteq V$, there exists k disjoint paths between p and q

---

ROBUSTNESS PROPERTY

HYP: at most k nodes are crashed

→ If the graph is $(k+1)$-connected, then all correct nodes communicate reliably

---

PROOF:

proof by contradiction

[SPEAK]

- We want to prove P

(1) suppose the opposite of P

(2) show a contradiction

HERE:

suppose the opposite: the graph is $(k+1)$-connected BUT there exists 2 correct nodes p and q that do not communicate reliably.

- As $(k+1)$-connected → there exists $(k+1)$ disjoint paths $(P_1, \dots, P_{k+1})$ connecting p and q.

- as p and q do not communicate reliably, ALL paths are "cut" by a crashed node.

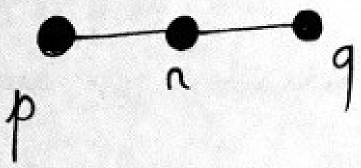
- as the paths are disjoint, it requires $k+1$ crashed nodes to cut them all

→ CONTRADICTION

→ the result.

⚠ TO MEMORIZE
- proof by INDUCTION
- proof by CONTRADICTION

5 RANDOM FAILURES

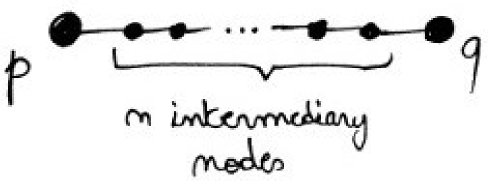


p and q correct

n : proba. f to fail

- $\overbrace{\text{proba}}^{P}$ that p and q communicate reliably ?

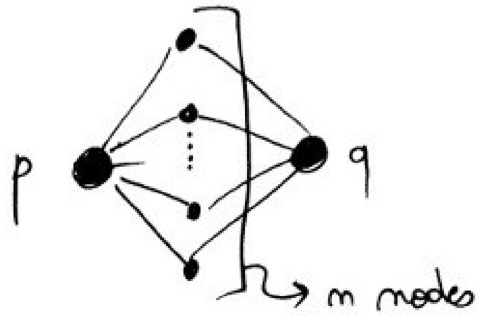
→ $\boxed{P = 1 - f}$ (obvious)



P ?

p and q connected when <u>ALL</u> nodes are "not crashed"

→ $\boxed{P = (1-f)^m}$



P ?

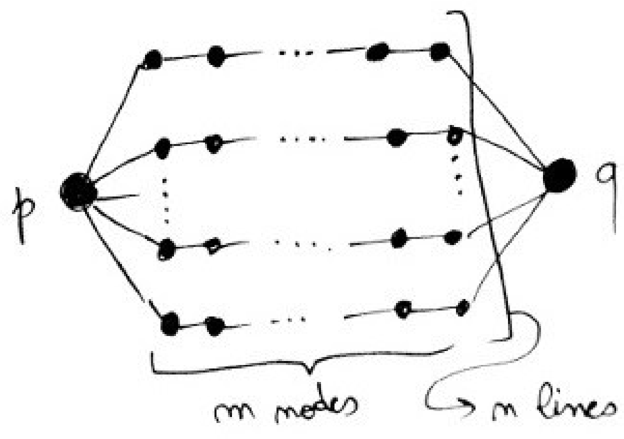
P' = proba that p and q are <u>NOT</u> connected

→ only when <u>ALL</u> nodes are crashed

→ $P' = f^m$

→ $P = 1 - P'$

$\boxed{P = 1 - f^m}$



(6) P?

P' = proba that p and q are NOT connected

→ only when ALL lines are cut

- proba. that a line is NOT cut :

$$P_1 = (1-f)^m$$

- proba that a line is cut :

$$P_2 = 1 - P_1 = 1-(1-f)^m$$

$$\rightarrow P' = P_2^{\,n}$$
$$= [1-(1-f)^m]^n$$

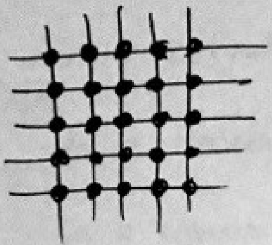
$$P = 1 - P'$$

$$\boxed{P = 1 - [1-(1-f)^m]^n}$$

(7)

INFINITE GRID

GRID topology :



QUESTION:

suppose ...

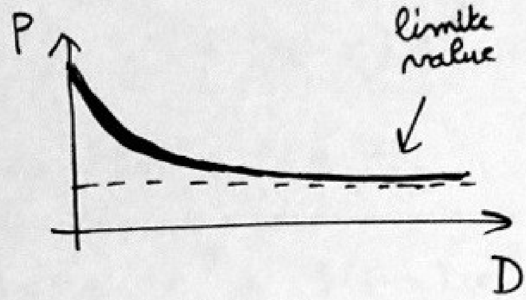
- a very large grid + random failures
- two distant nodes p and q (distance D)
- P : proba that p and q communicate reliably

$$\boxed{\lim_{D \to +\infty} P \ldots ?}$$

(0 or not ?)

case 1 :

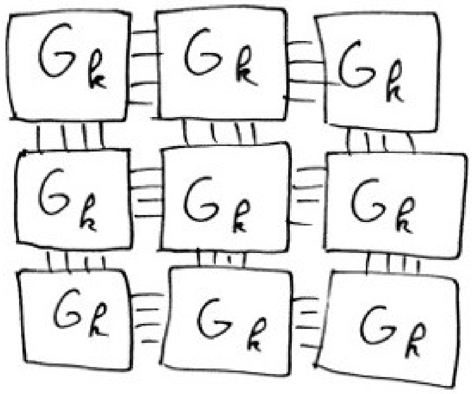
case 2 :



GOAL: show that we are in case 2

Sequence of grids G_k :

- G_0 = 1 node
- G_{k+1} : grid of "9 grids G_k"



f : proba of failure of each node

$f = 0.01$

DEF: "correct" grid

G_{k+1} is "correct" iff at least 8 of its grids G_k are "correct"

(8) **DEF** : "meta-connect" mode

consider a grid G_m

p is "meta-connect" if p is ...

- in a connect grid G_m

AND

- in a connect grid G_{m-1}

AND

- in a connect grid G_{m-2}

$\vdots$

- in a connect grid G_0

ADMITTED RESULT :

ALL meta-connect modes are connected

now: proof in 3 steps

STEP 1

x : proba that G_k is connect

$P(x)$: proba that G_{k+1} is connect

$\rightarrow P(x)$ .... ?

G_{k+1} connect if :

- 9 "G_k" are connect (P_1)

OR

- 8 "G_k" are connect (P_2)

$$P_1 = x^9$$

$$P_2 = 9(1-x)x^8$$

$$\rightarrow P(x) = P_1 + P_2$$

$$\boxed{P(x) = x^9 + 9(1-x)x^8}$$