

Distributed Algorithms

Fall 2020

Links & Gossip

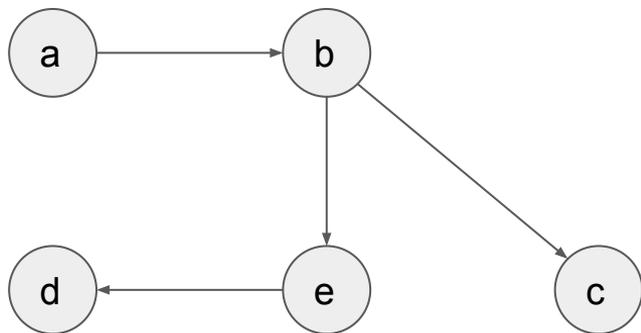
1st exercise session, 28/09/2020

Matteo Monti <matteo.monti@epfl.ch>

Jovan Komatovic <jovan.komatovic@epfl.ch>

Graphs

A **graph** is a couple (V, E) where V is a set of **vertices** and $E \subseteq V^2$ is a set of **edges**.



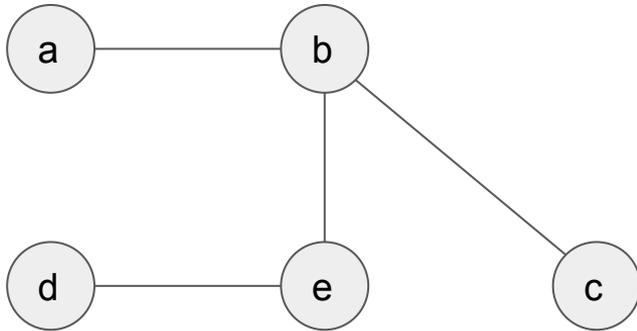
Example graph (V, E) :

- $V = \{a, b, c, d, e\}$
- $E = \{(a, b), (b, c), (b, e), (e, d)\}$

Two vertices are **adjacent** (or **neighbors**) iff an edge exists between them. In the example, a and b are adjacent; a and d are not adjacent.

Graphs (undirected)

An **undirected graph** is a graph (V, E) such that $(a, b) \in E$ if and only if $(b, a) \in E$.



Example graph (V, E) :

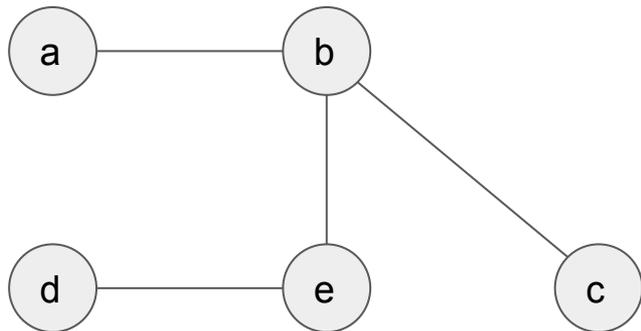
- $V = \{a, b, c, d, e\}$
- $E = \{(a, b), (b, a), (b, c), (c, b), (b, e), (e, b), (e, d), (d, e)\}$

We use undirected graphs to model networks of processes:

- Each vertex represents a process
- Two vertices are neighbors iff the corresponding processes can directly exchange messages.

Paths

A **path** is a sequence of *distinct* vertices (v_1, \dots, v_N) such that, for all $i \in [1, N - 1]$, v_i and v_{i+1} are adjacent.



Some paths in (V, E) :

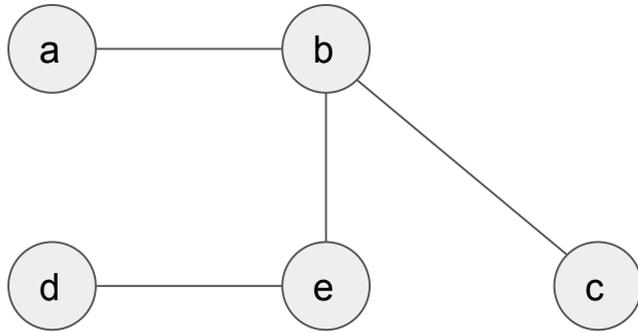
- (a, b)
- (a, b, c)
- (a, b, e, d)

While

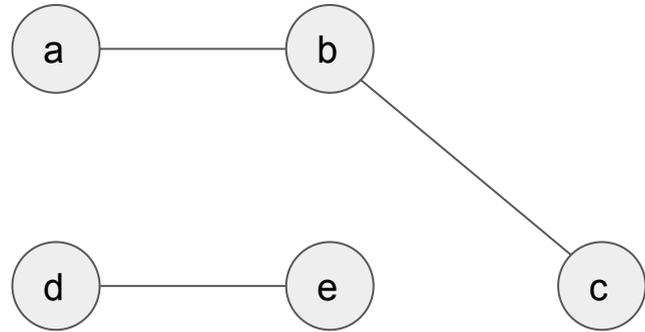
- (a, c, e) is **not** a path: *a and c are not adjacent!*

Connectivity

Two distinct vertices a and z are **connected** if and only if at least one path (a, \dots, z) exists in the graph. A graph is connected if any two distinct vertices are connected.



A connected graph



A disconnected graph

Exercise 1 (connectivity)

Prove that **connectivity** is a **symmetric property** on an undirected graph: let a, b be vertices such that a is connected with b . Prove that b is connected with a .

Hint: you can do it constructively.

Exercise 2 (connectivity)

Prove that **connectivity** is a **transitive property** on an undirected graph: let a , b , c be vertices such that a is connected with b and b is connected with c . Prove that a is connected with c .

Hint: double-check the definition of a path.

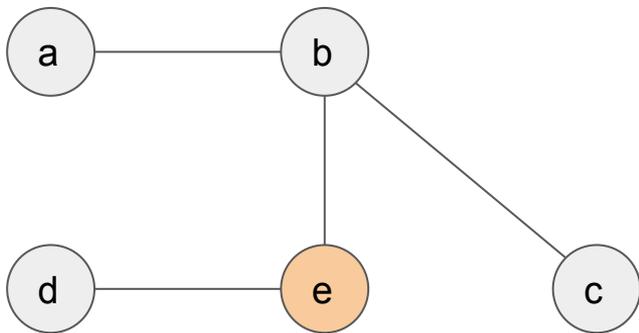
Exercise 3 (connectivity)

Write a procedure (pseudocode or any programming language) that inputs an undirected graph $G = (V, E)$ and outputs *true* if and only if the G is connected.

Hint: use the results from Exercises 1 and 2.

Gossip

We use an undirected graph to represent which processes can communicate. Upon receiving a new message m , a process forwards m to all its neighbors.

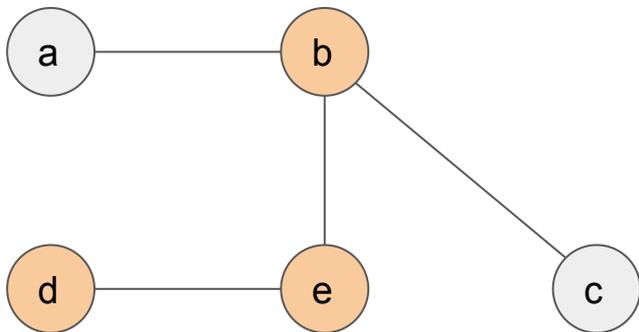


Example: diffusion of a message m from process e .

- *e issues m*

Gossip

We use an undirected graph to represent which processes can communicate. Upon receiving a new message m , a process forwards m to all its neighbors.

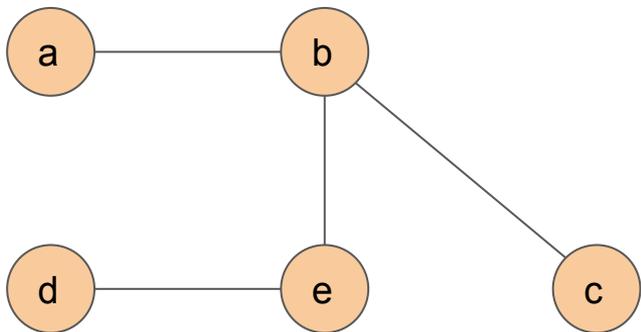


Example: diffusion of a message m from process e .

- *e issues m .*
- *b and d receive m .*

Gossip

We use an undirected graph to represent which processes can communicate. Upon receiving a new message m , a process forwards m to all its neighbors.



Example: diffusion of a message m from process e .

- *e issues m .*
- *b and d receive m .*
- *a and c receive m .*

Gossip is **correct** if and only if, if the sender is correct, every correct process eventually receives the message.

Exercise 4 (gossip)

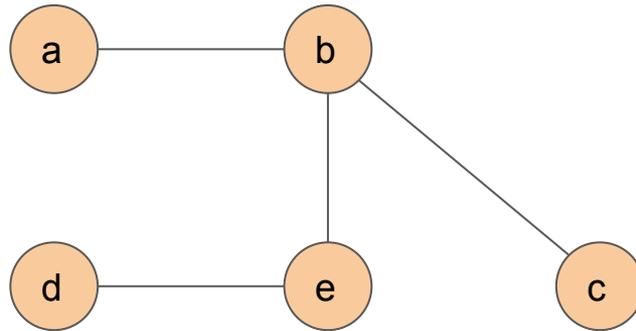
Prove that gossip is correct if and only if the subgraph of correct processes is connected.

Note: prove **both** directions of the implication!

Hint: induction is your friend.

Exercise 5 (gossip)

In the following system, exactly one process crashes. What is the minimum number of edges we need to add so that gossip is always correct?



k -connectivity

Two paths p, p' connecting two vertices a and z are **disjoint** if they have no vertex in common, except a and z :

$$p = (a, b, \dots, y, z)$$

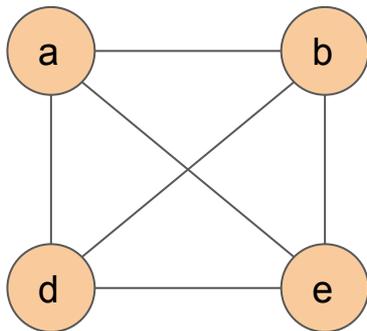
$$p' = (a, b', \dots, y', z)$$

$$\{a, b, \dots, y, z\} \cap \{a, b', \dots, y', z\} = \{a, z\}$$

A graph is **k -connected** if and only if k disjoint paths exist between any two vertices of the graph.

Robustness

Gossip is **robust** to k failures if and only if it is always correct, as long as no more than k nodes are crashed.



A fully connected gossip graph is robust to N failures, where N is the number of processes.

Exercise 6 (robustness)

Prove that, if the gossip graph is $(k+1)$ -connected, then gossip is k -robust.

Is the converse also true? Find a counterexample if not.

Hint: contradiction is your friend.

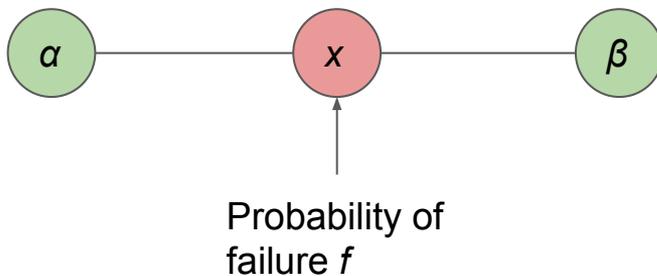
Random failures

Suppose that processes can fail independently with probability f .

What is the probability that two *correct* processes can communicate in the presence of failures?

It depends on their connectivity!

e.g.



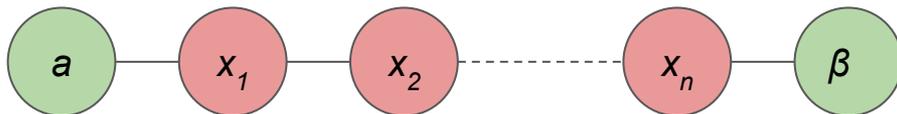
α, β can communicate iff x has not failed \Rightarrow

α, β communicate with probability $1-f$.

Exercise 7 (random failures on series topology)

Suppose that processes x_i , $i=1, \dots, n$ can fail independently with probability f .

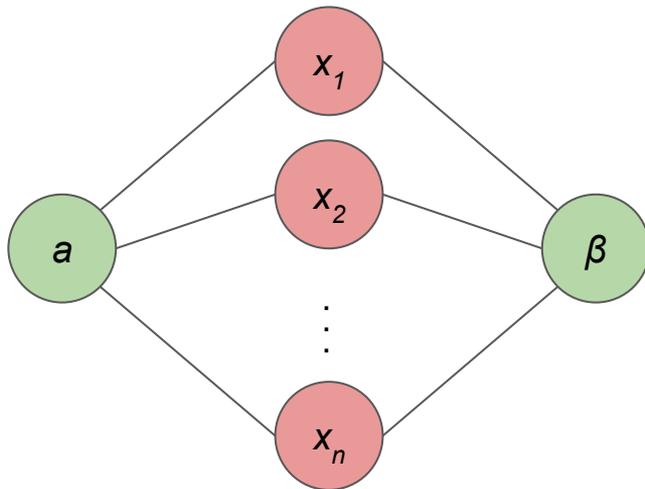
What is the probability that a and b can communicate?



Exercise 8 (random failures on parallel topology)

Suppose that processes x_i , $i=1, \dots, n$ can fail independently with probability f .

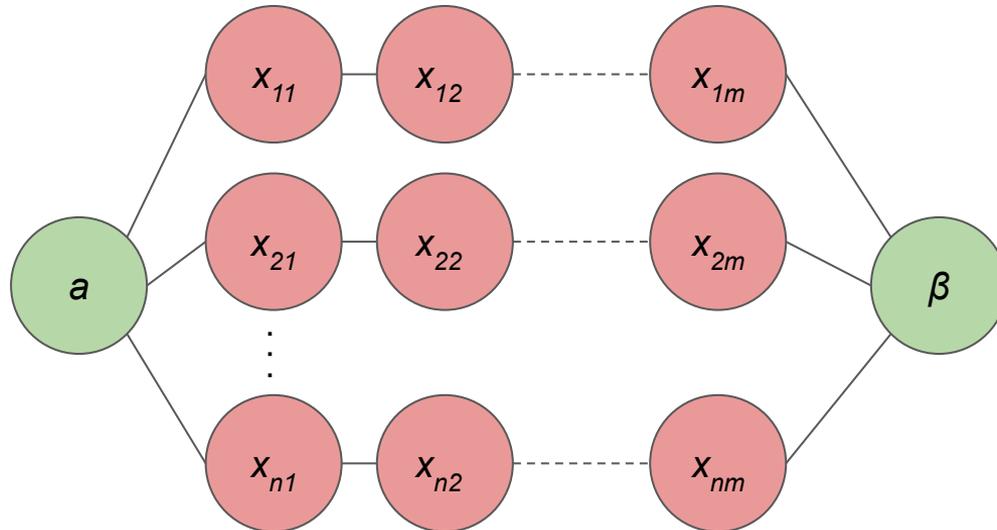
What is the probability that a and b can communicate?



Exercise 9 (random failures on series/parallel topology)

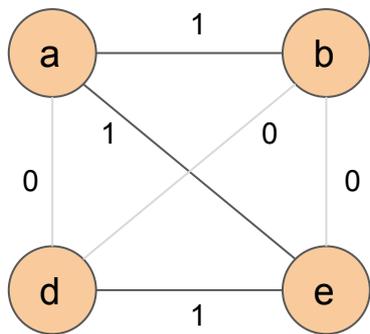
Suppose that processes x_{ij} , $i=1, \dots, n$, $j=1, \dots, m$ can fail independently with probability f .

Prove that a and b can communicate with probability $1 - [1 - (1-f)^m]^n$.



Erdős-Renyi graphs

An Erdős-Renyi graph $G(N, p)$ is a random undirected graph with N vertices, such that any two distinct vertices have an independent probability p of being adjacent.



Example graph
 $G(4, \frac{1}{2})$

An Erdős-Renyi graph is defined by the values of $N(N - 1)/2$ independent Bernoulli random variables:

$$E_{ij} \sim \text{Bernoulli}(p)$$
$$E_{ij} = E_{ji}$$

with $i, j \in V$. Vertices i and j are adjacent iff $E_{ij} = 1$.

Bonus Exercise 10 (Erdős-Renyi graphs)

What distribution underlies the number of edges in an Erdős-Renyi $G(N, p)$?

What distribution underlies the degree (i.e., number of links) of any vertex?

Are the degrees of any two vertices independently distributed?

Hint: how is the sum of Bernoulli variables distributed?

Connectivity of $G(N, p)$

Let $C(N, p)$ denote the probability of a random graph $G(N, p)$ being connected. It is possible to prove that:

$$\begin{aligned} \lim_{N \rightarrow \infty} C(N, p) &= 0 && \text{iff } p < \ln(N) / N \\ \lim_{N \rightarrow \infty} C(N, p) &= 1 && \text{iff } p > \ln(N) / N \end{aligned}$$

A large Erdős-Renyi graph is almost surely connected, as long as each vertex has an expected degree larger than $\ln(N)$.

We can use Erdős-Renyi graphs to build probabilistic gossip with
logarithmic communication complexity!

Bonus Exercise 11 (Erdős-Renyi graphs)

Write a distributed procedure that runs on N processes to build an Erdős-Renyi graph $G(N, \ln(N)/N)$. We assume no failures. Each process can invoke:

- A procedure $rand(x)$ that returns a real number between 0 and x , independently picked with uniform probability.
- A procedure $connect(i)$ to connect to the i -th process.

Is it possible for the procedure to have $O(\ln(N))$ computation complexity?