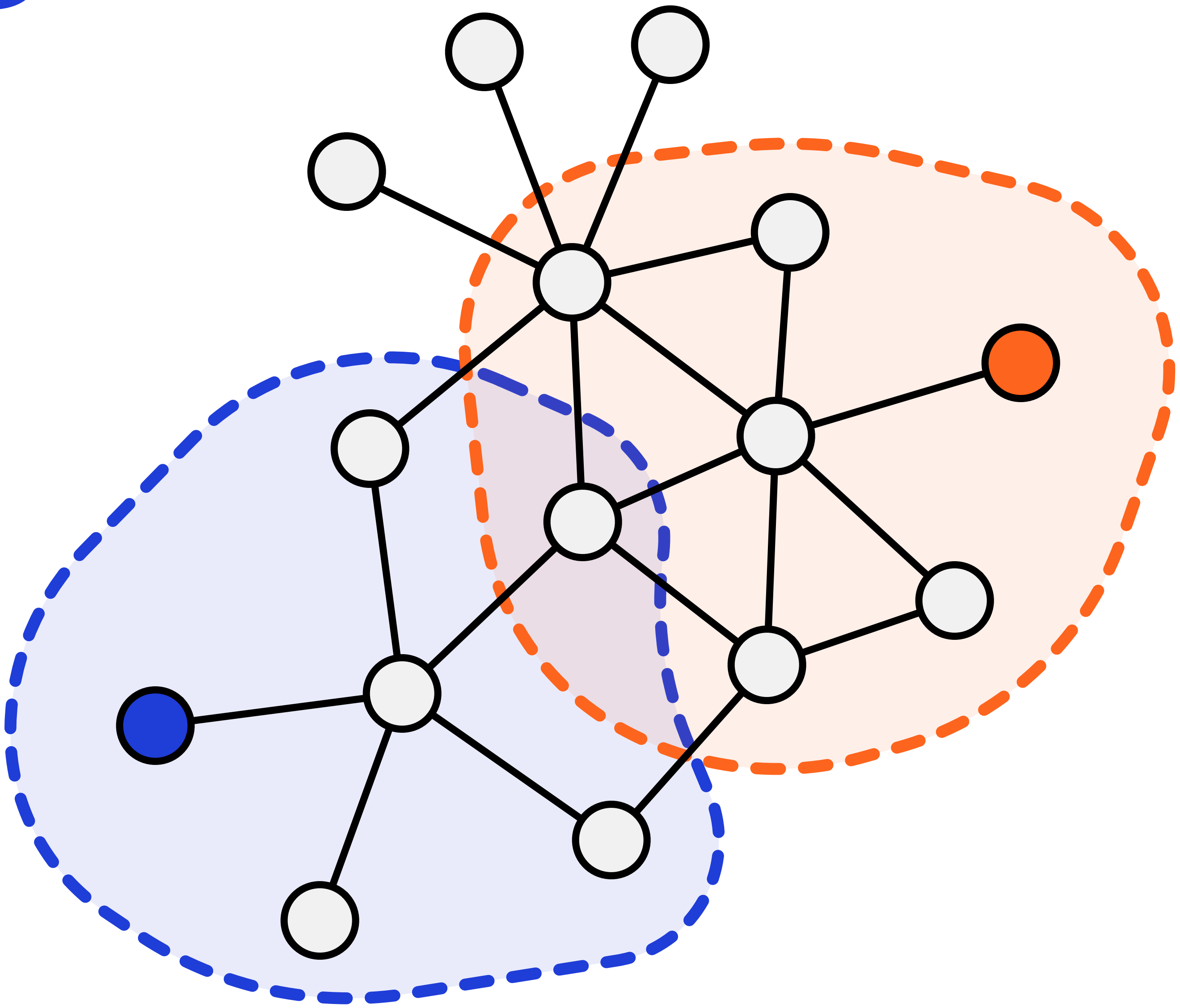
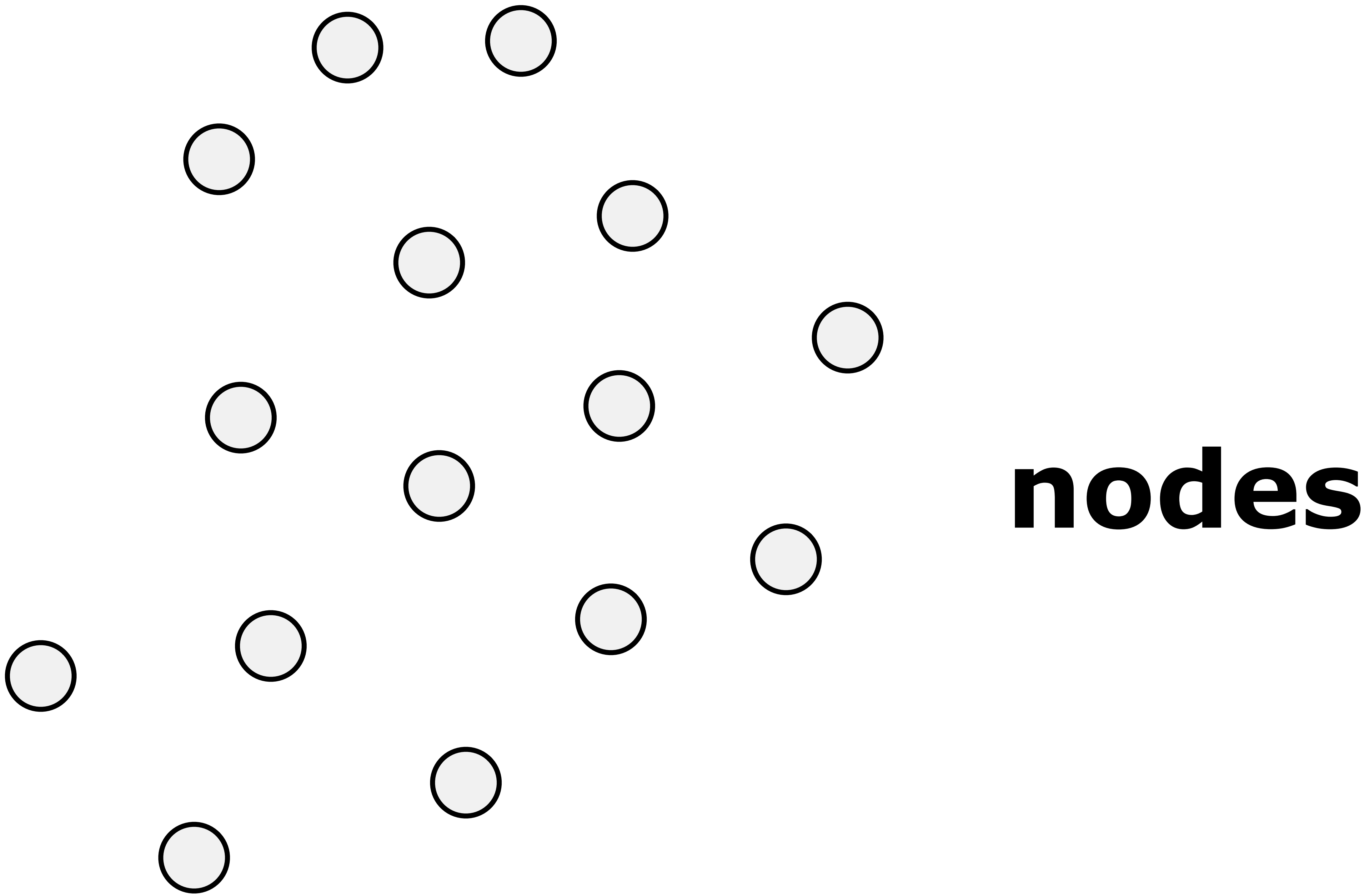
Introduction to **Distributed Graph Algorithms**

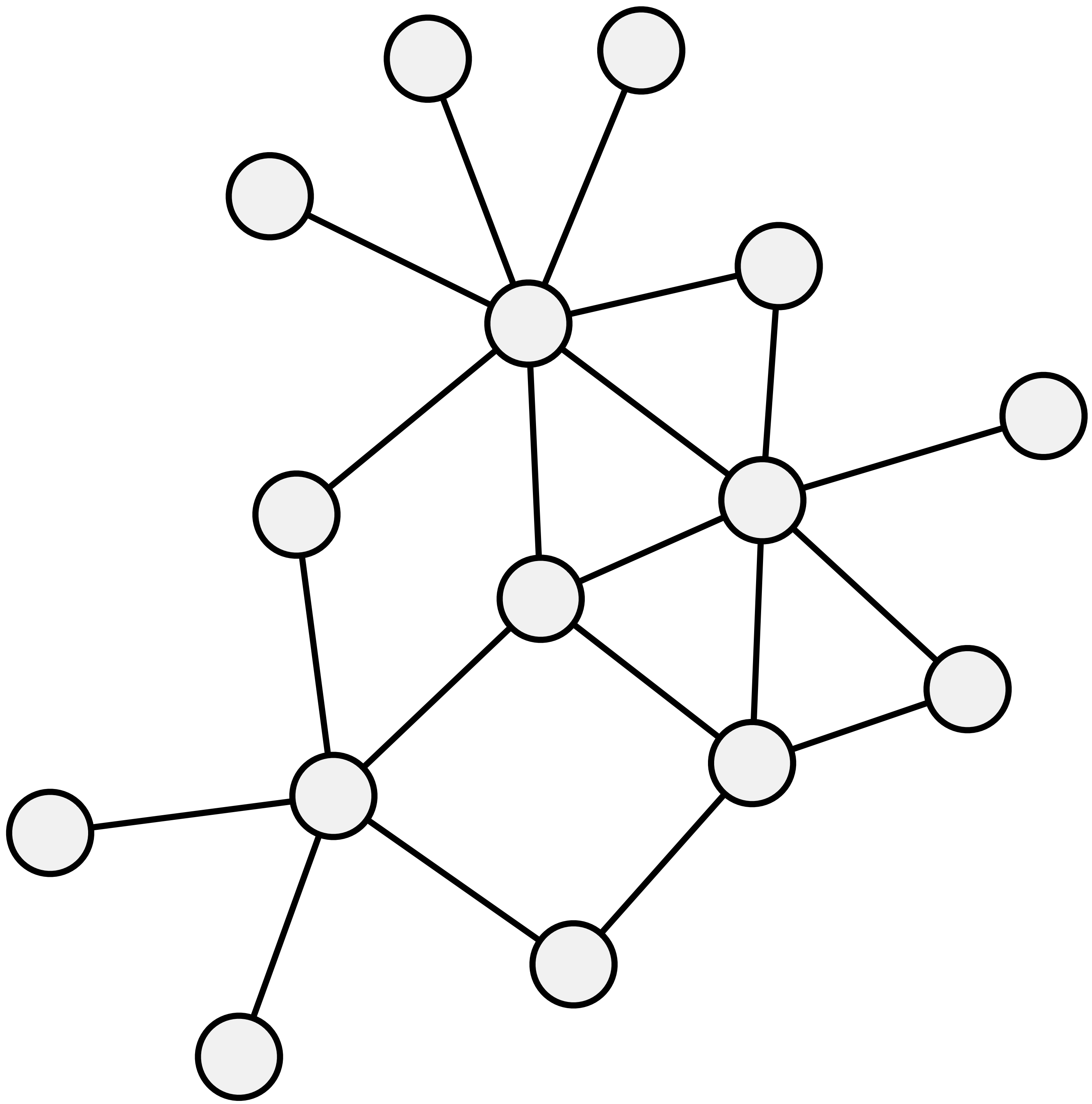


Jukka Suomela

graph

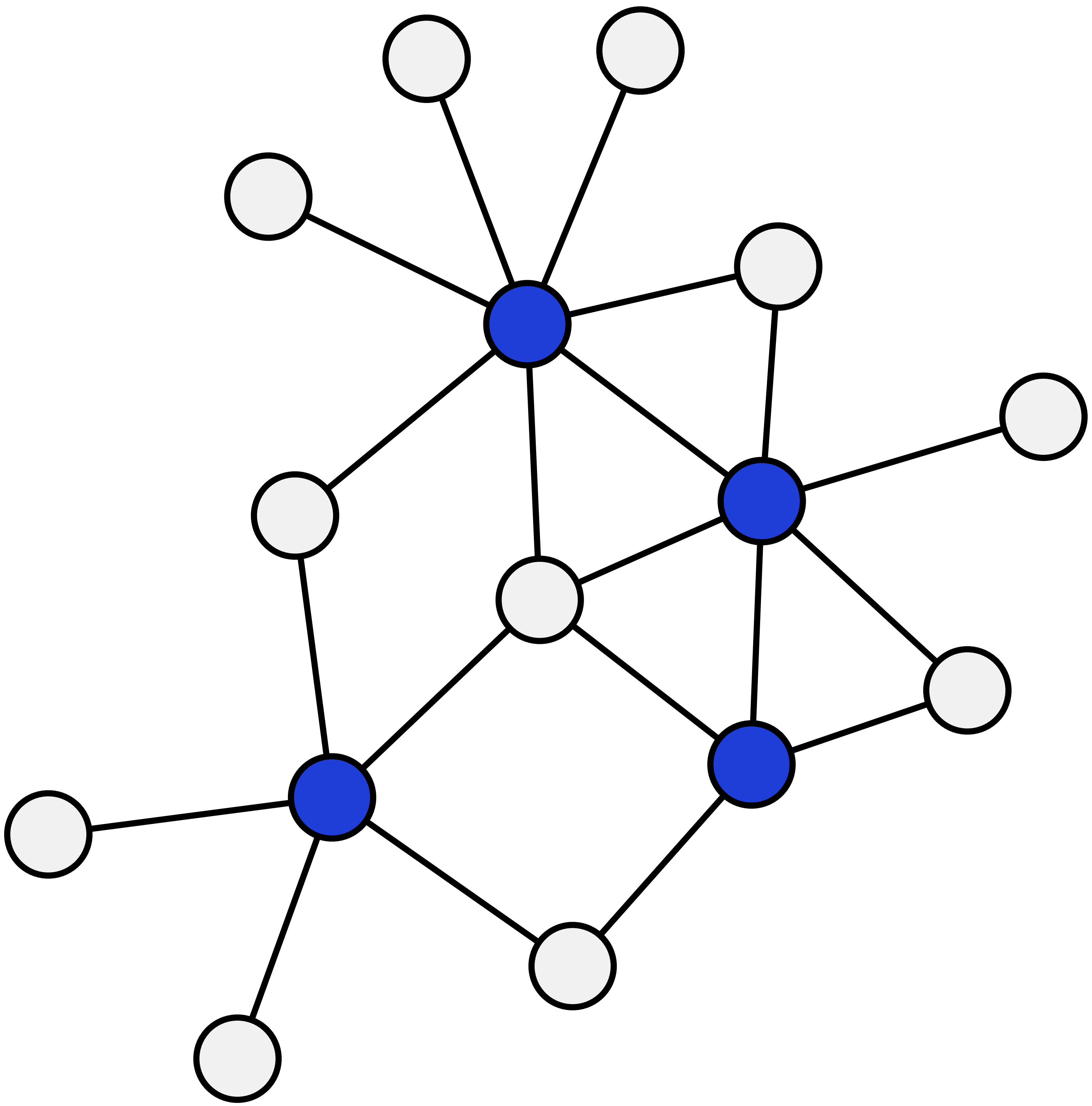


graph



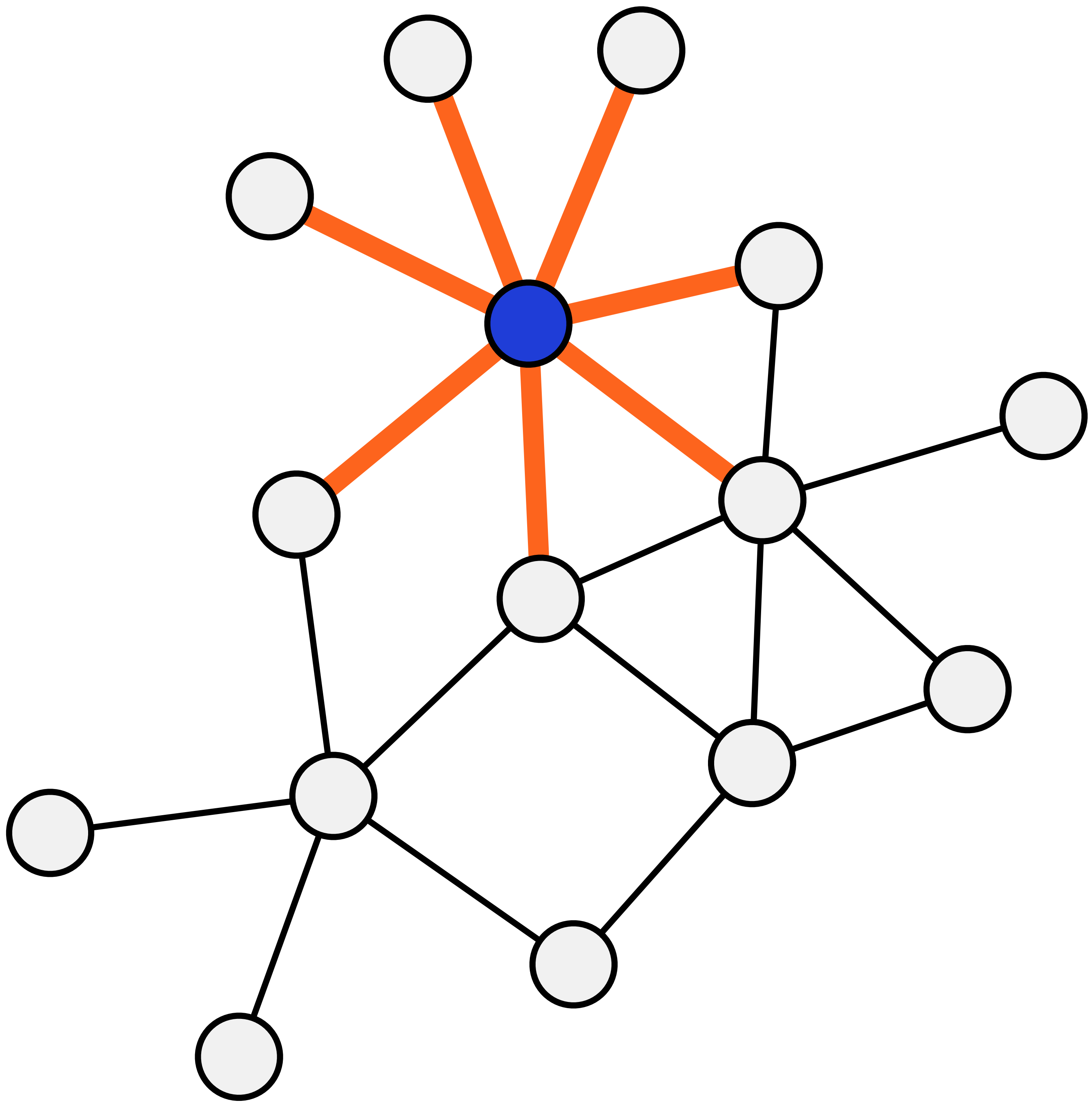
nodes
+
edges

graph problems



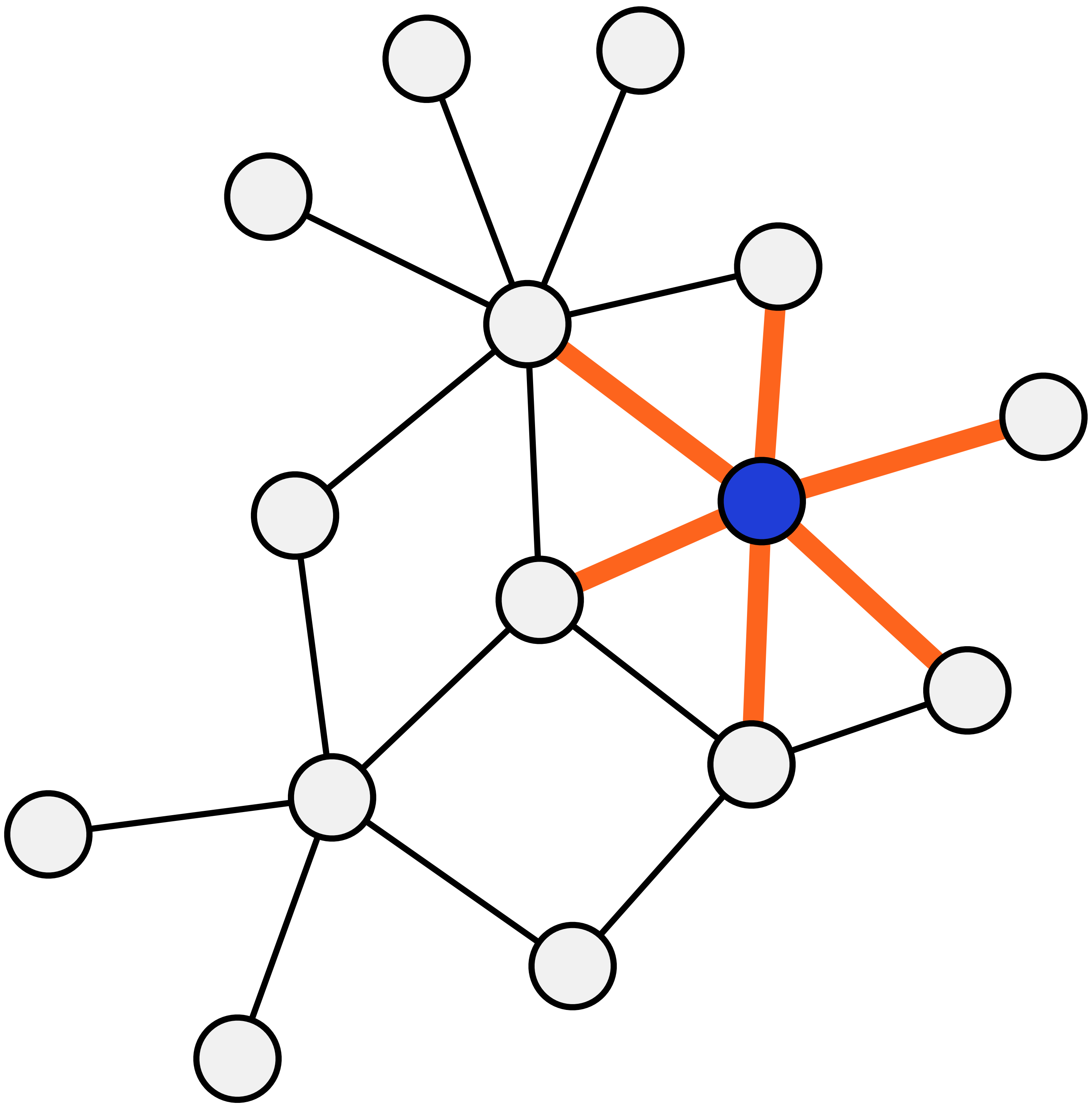
vertex cover

graph problems



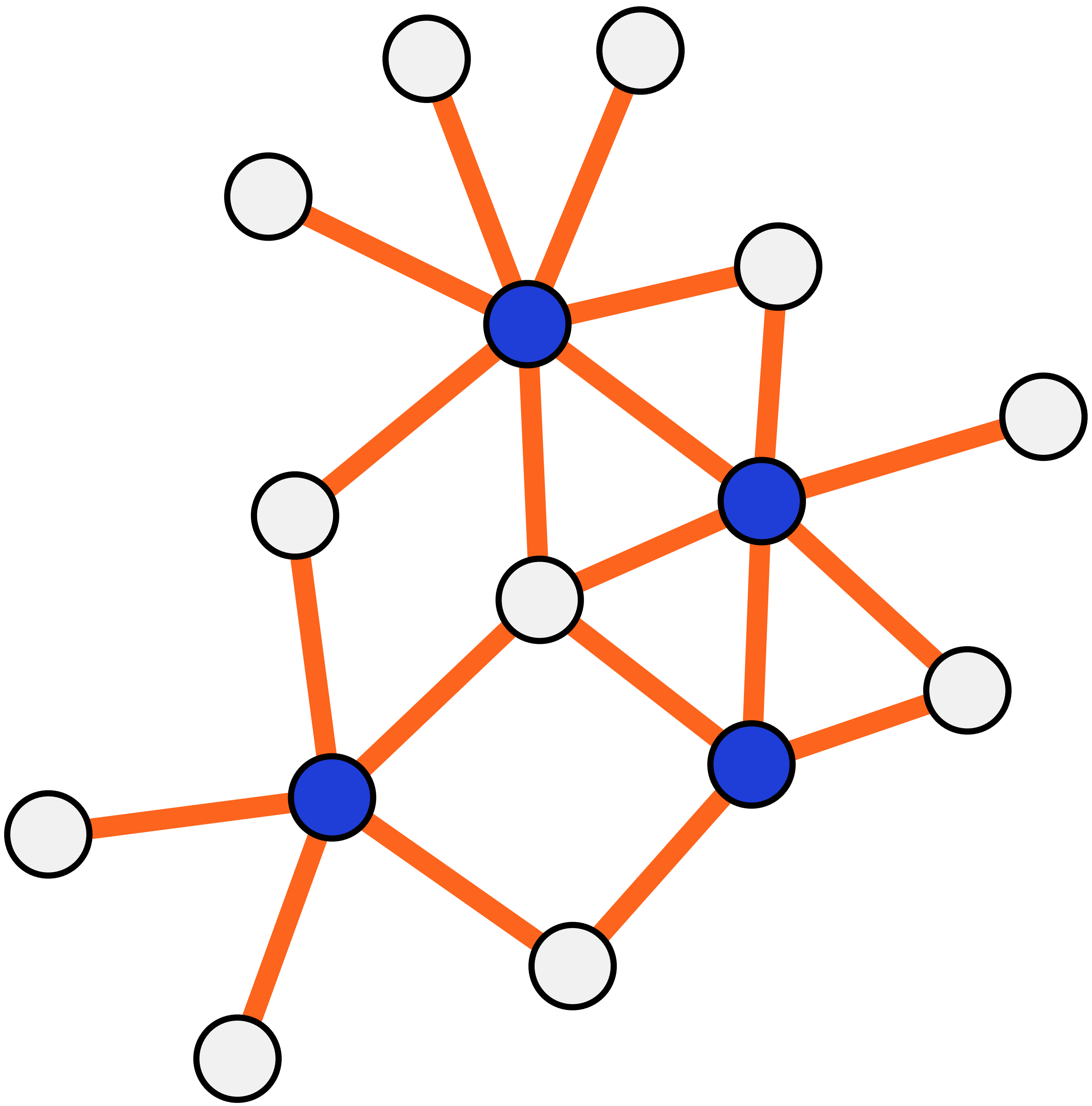
vertex cover

graph problems



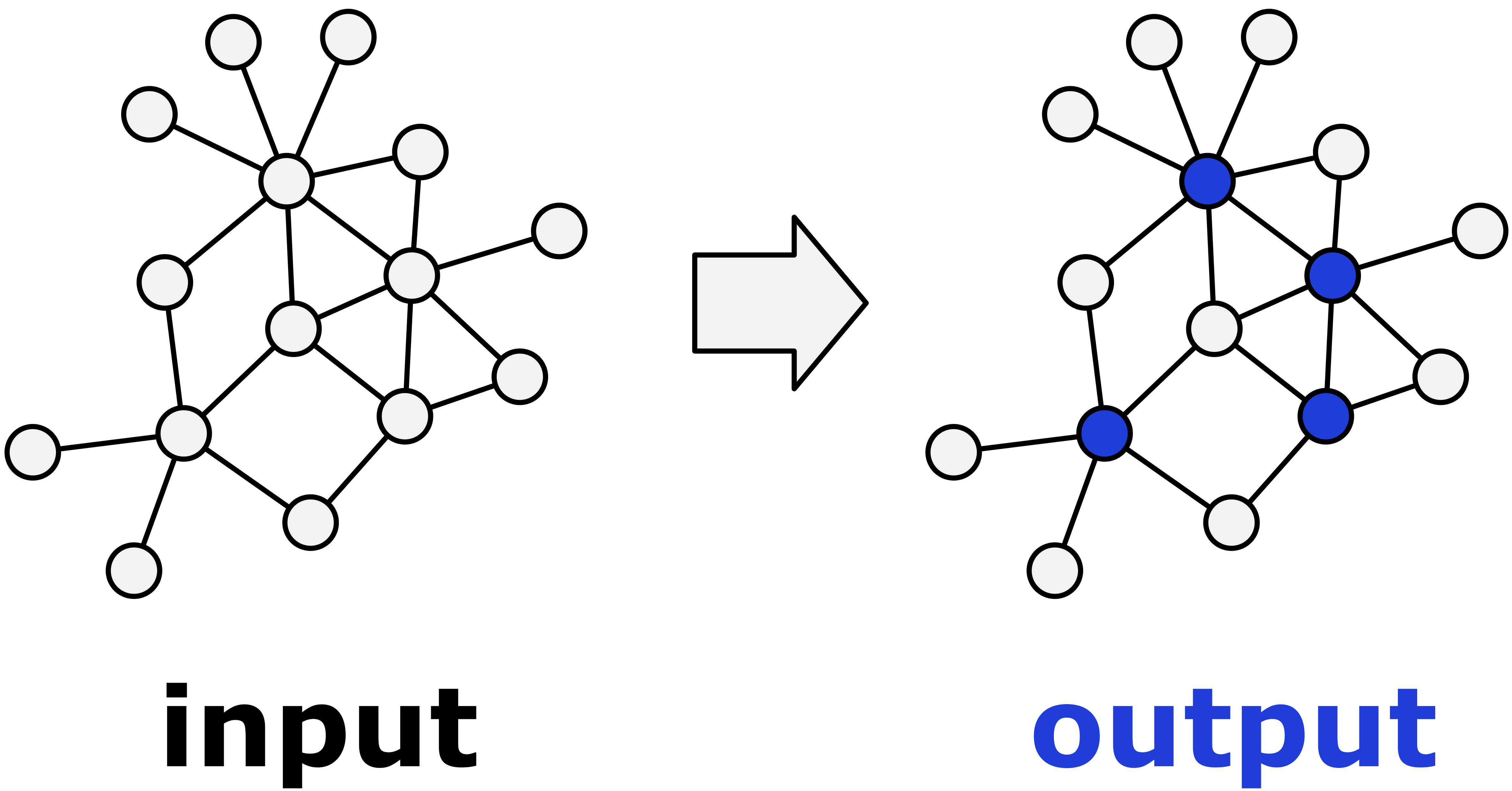
vertex cover

graph problems



vertex cover

graph algorithms

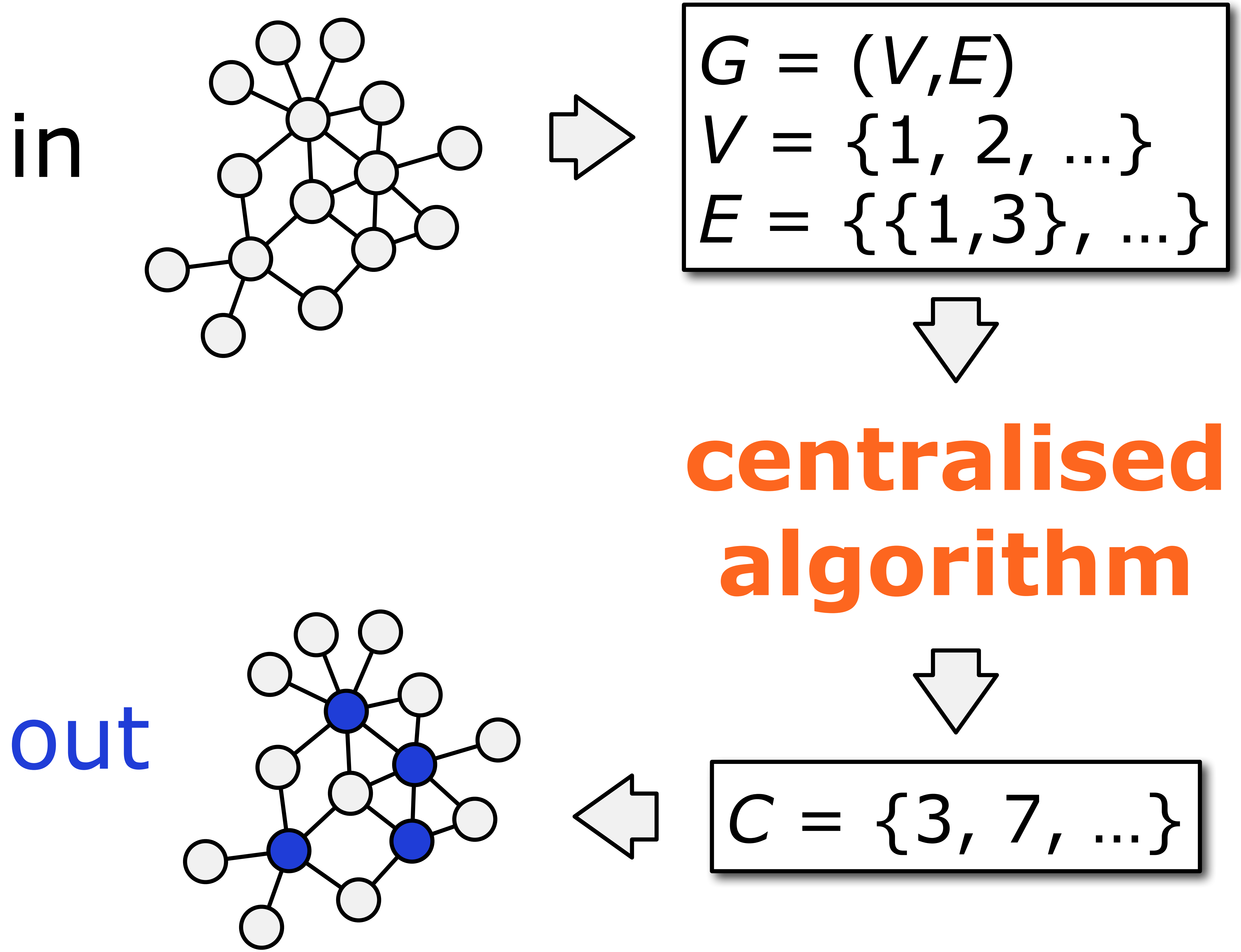


graph algorithms

“distributed”

vs.

“centralised”


in
G = (V,E)
V = {1, 2, …}
E = {{1,3}, …}
centralised algorithm
out
C = {3, 7, …}

all input in
one location

$G = (V, E)$
$V = \{1, 2, \ldots\}$
$E = \{\{1,3\}, \ldots\}$

centralised algorithm

all output in
one location

$C = \{3, 7, \ldots\}$

all input in
one location

time unit ≈
one step of
computation

all output in
one location

$G = (V,E)$
$V = \{1, 2, \ldots\}$
$E = \{\{1,3\}, \ldots\}$

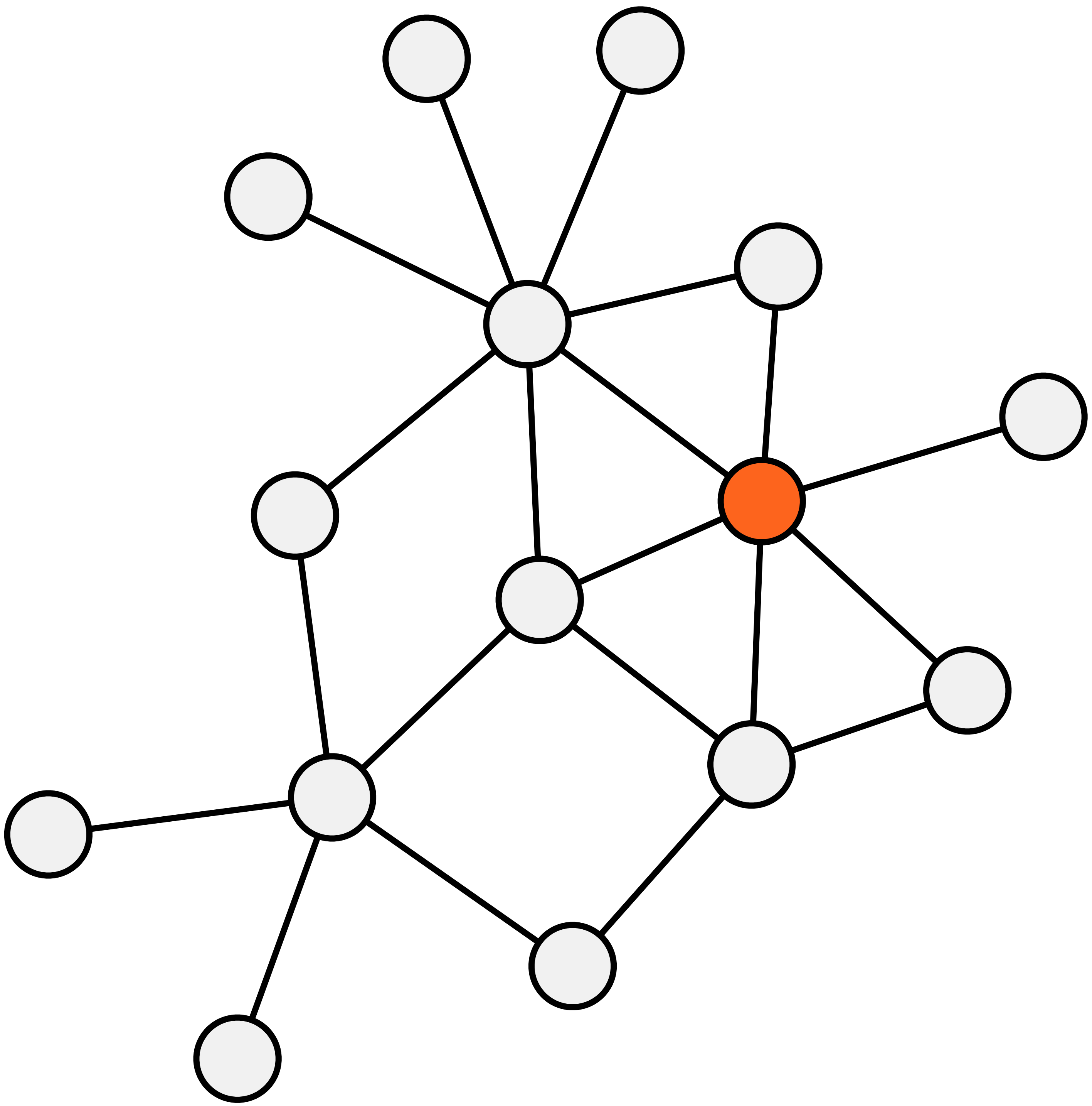
centralised algorithm

$C = \{3, 7, \ldots\}$

distributed graph algorithms

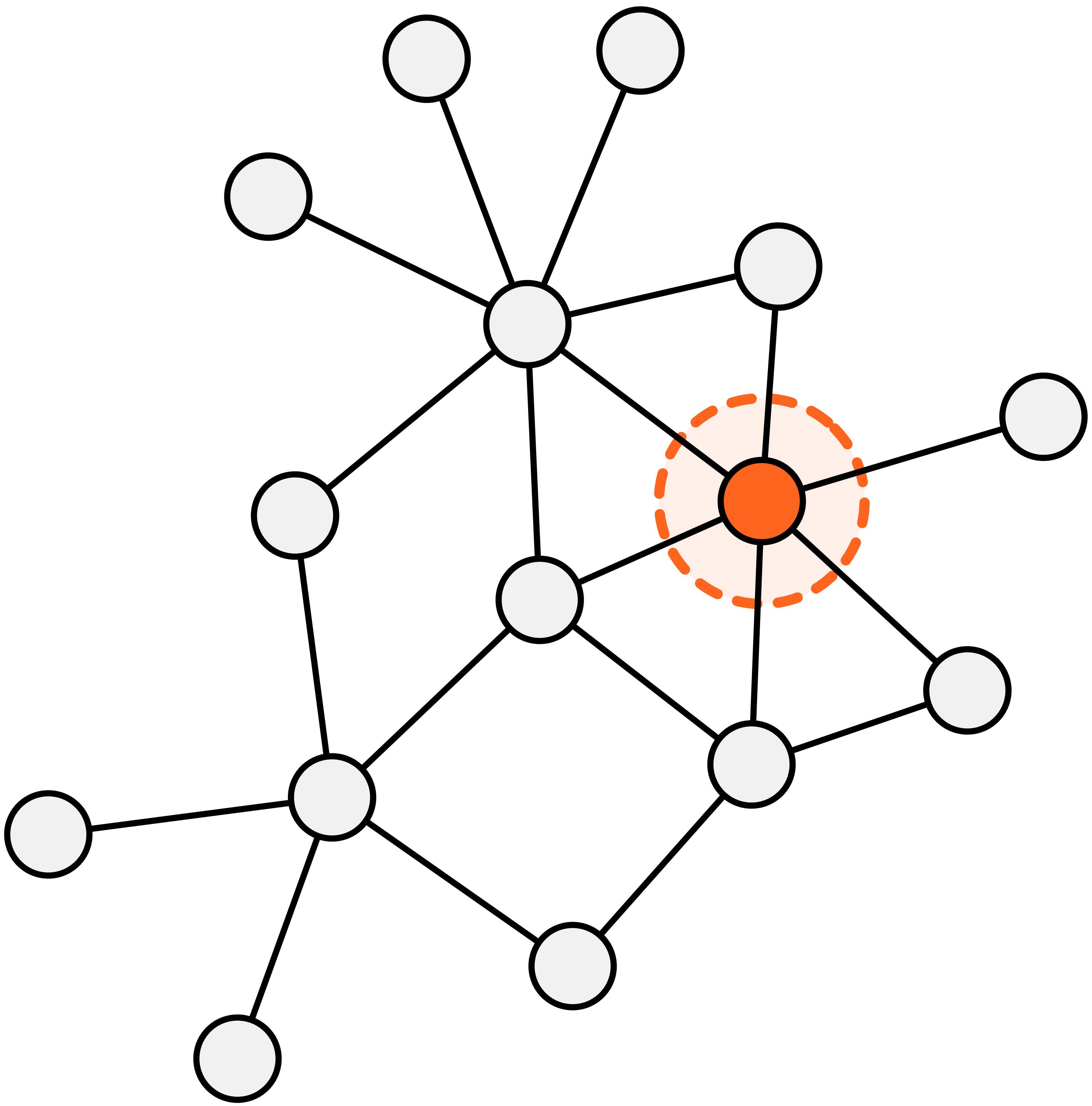
graph = computer network
node = computer
edge = communication link
time = communication steps

graph: computer network



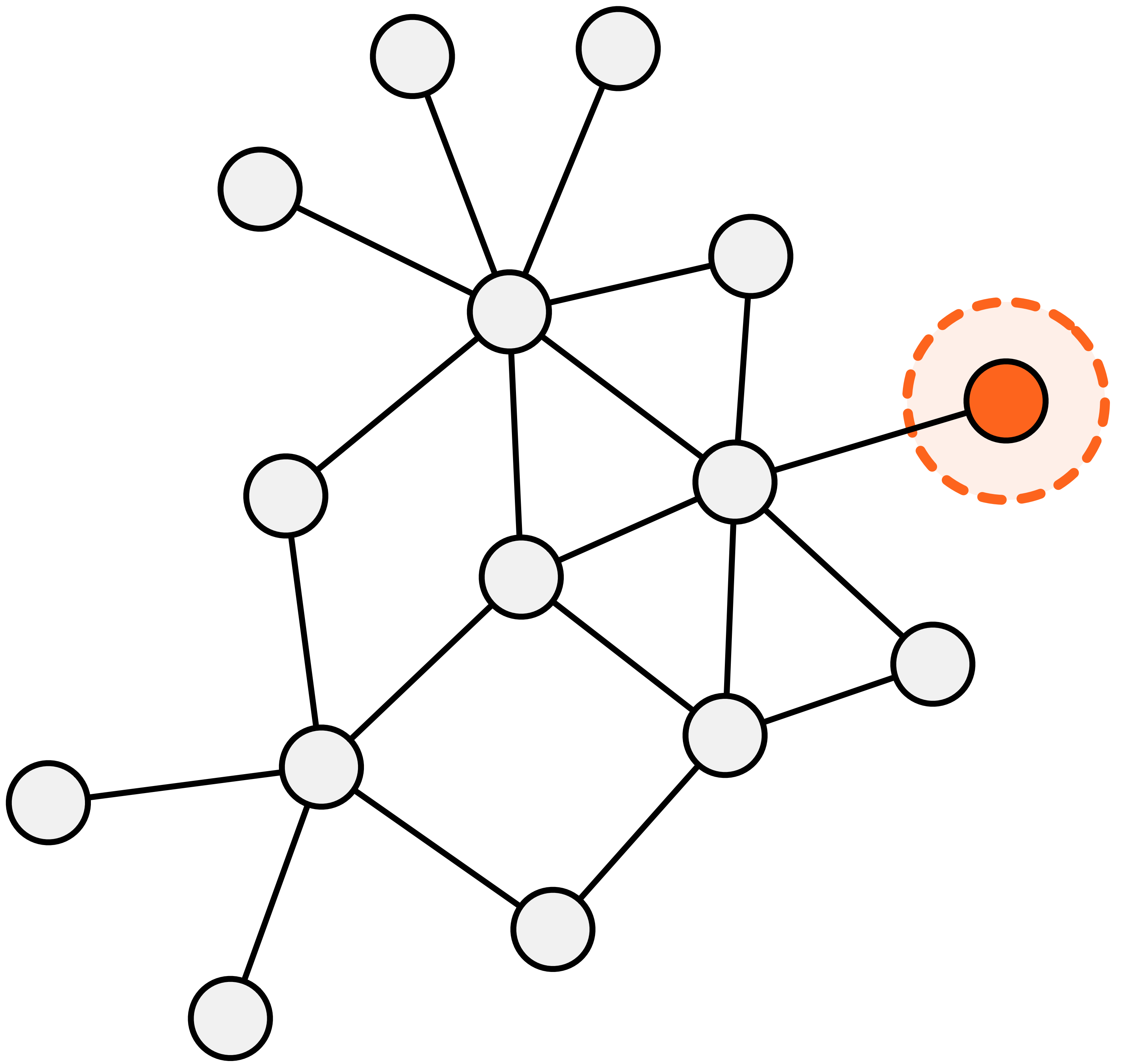
node:
computer

initial information



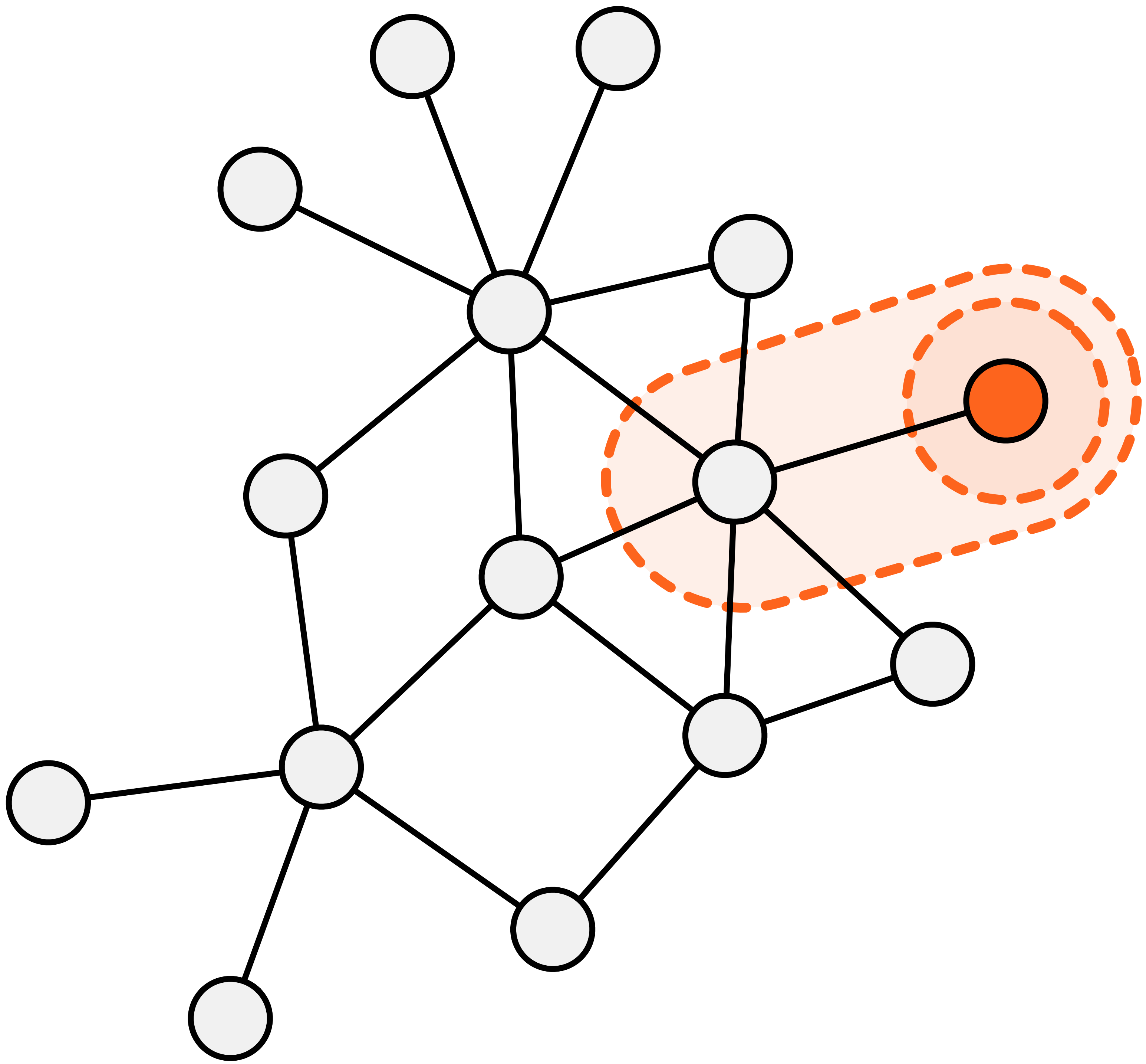
$t = 0$

initial information



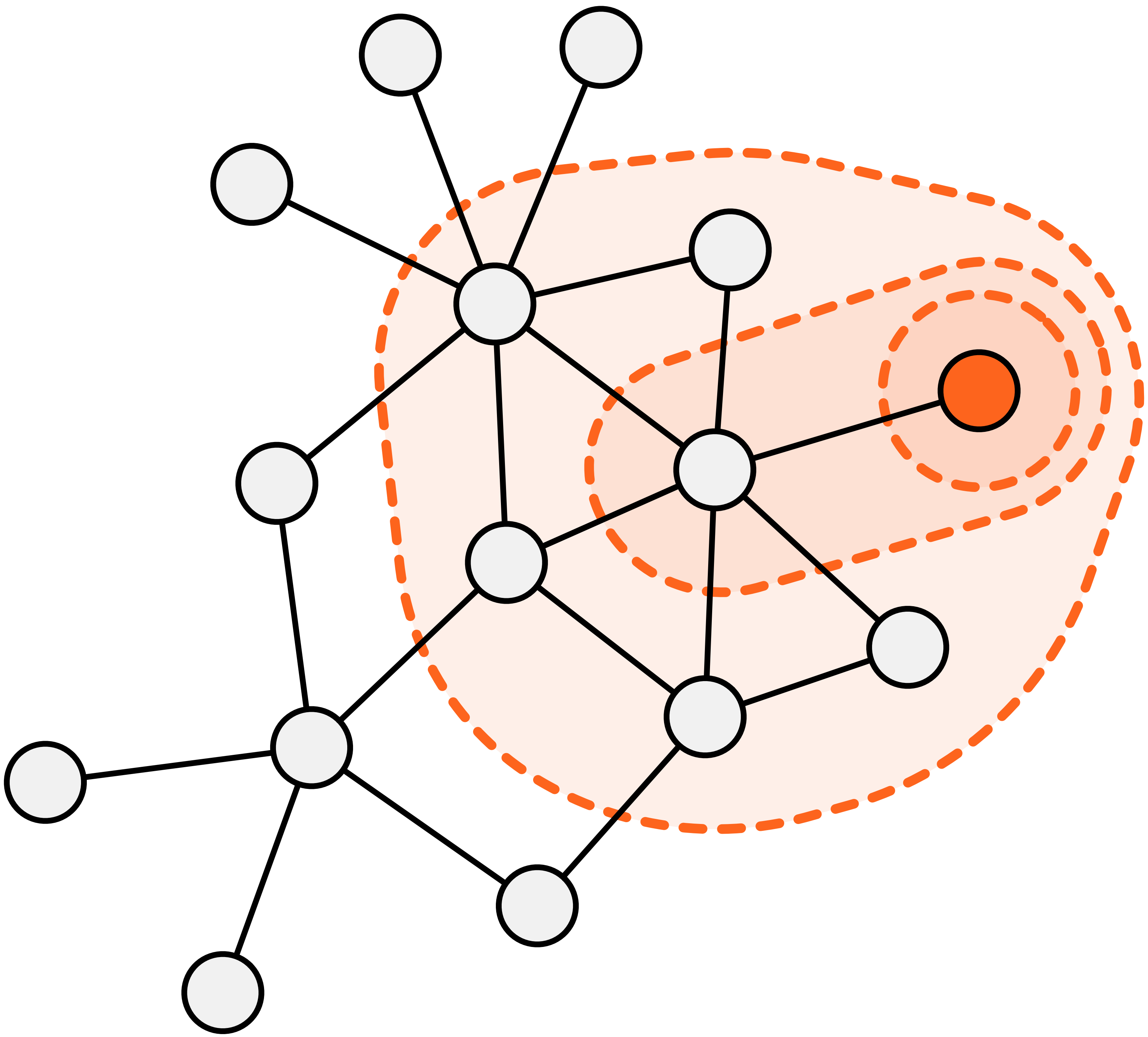
$t = 0$

time step: communication



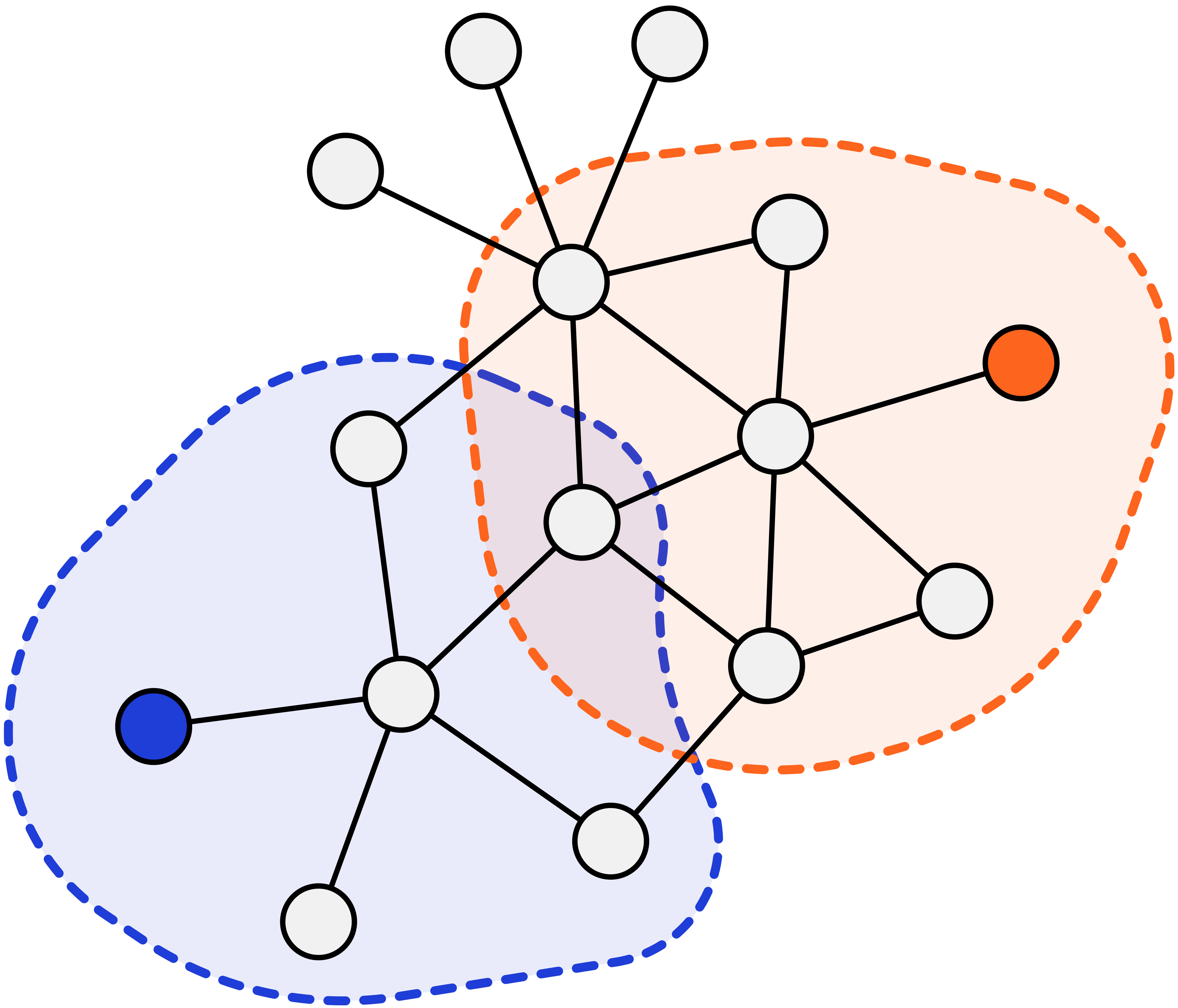
$t = 1$

time step: communication



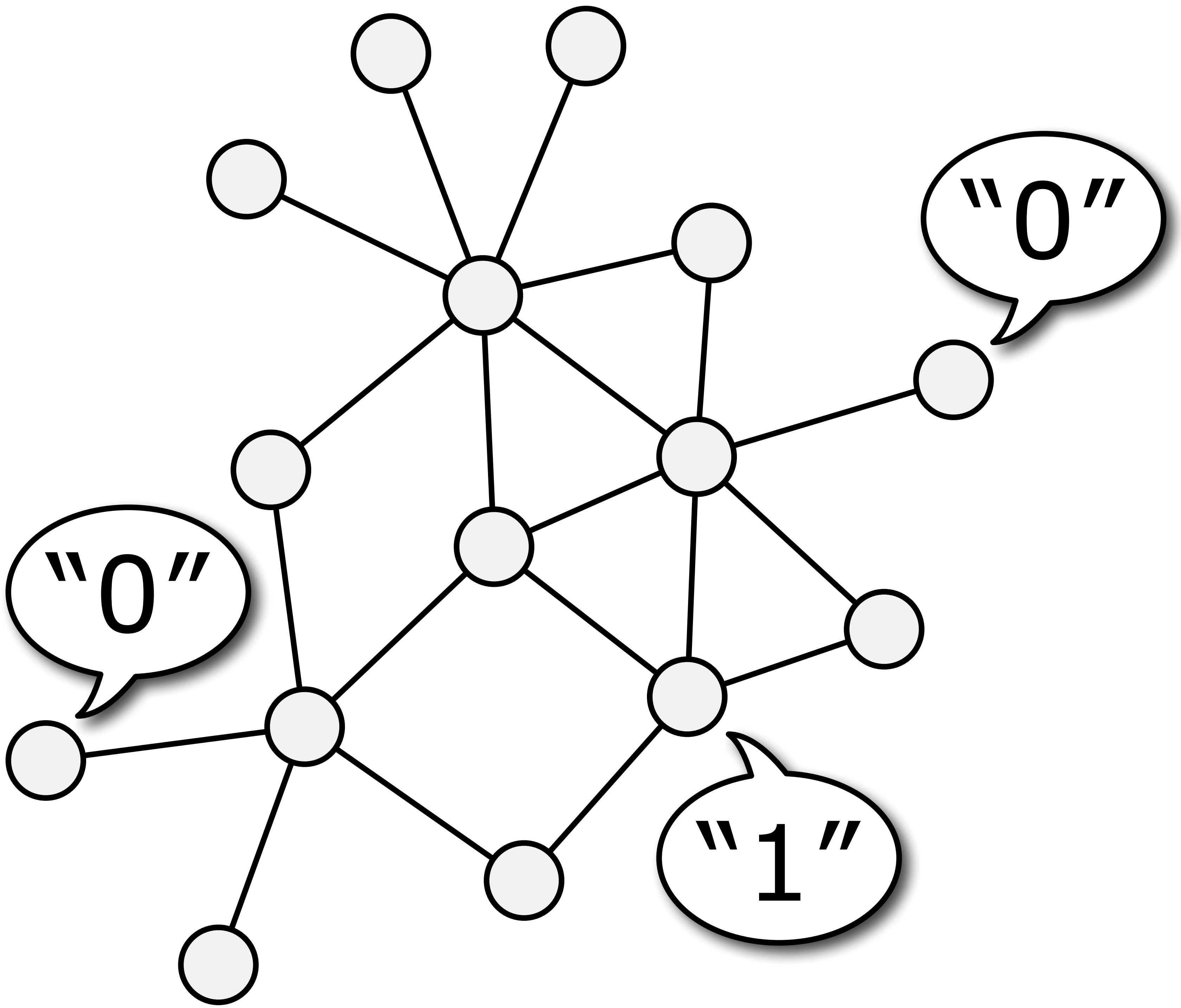
$t = 2$

all nodes in parallel



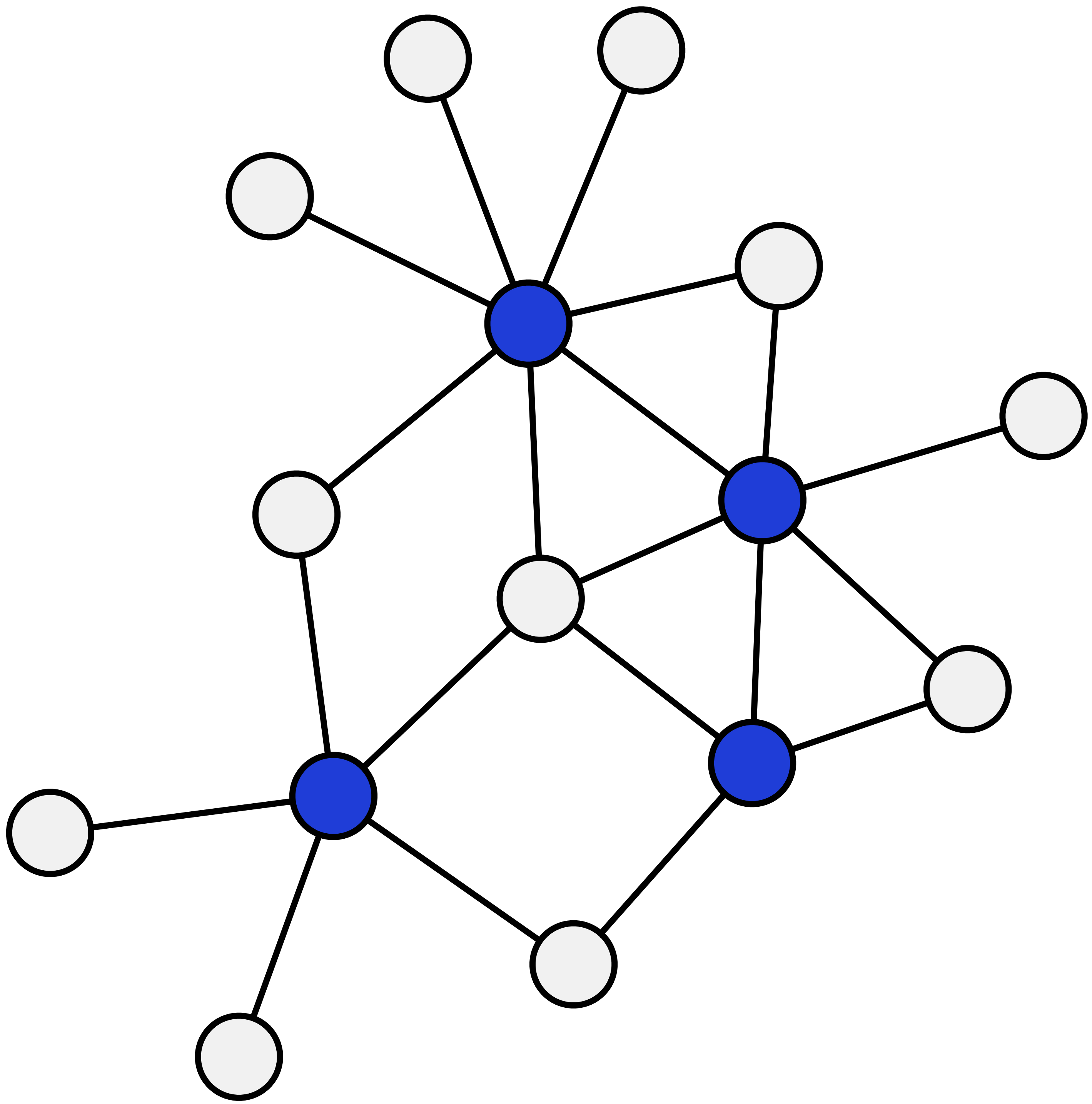
$t = 2$

local outputs



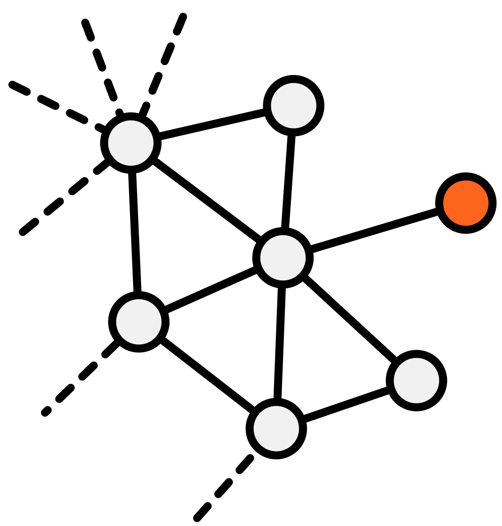
$t = 2$

nodes that output “1”

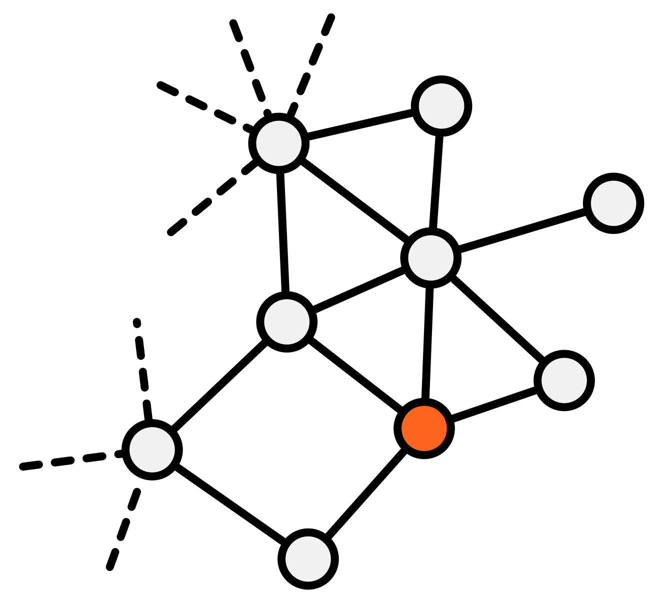


vertex cover

distributed algorithm



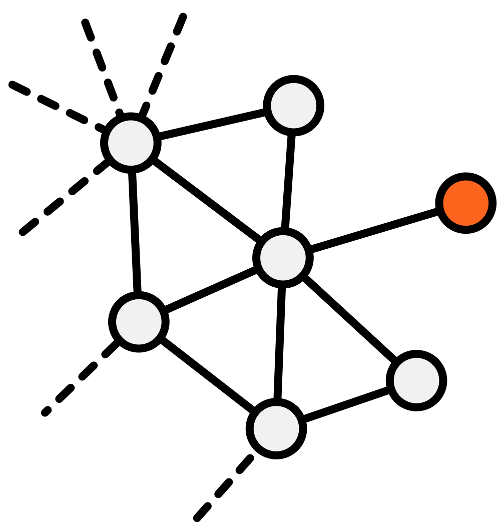
$\mapsto 0$



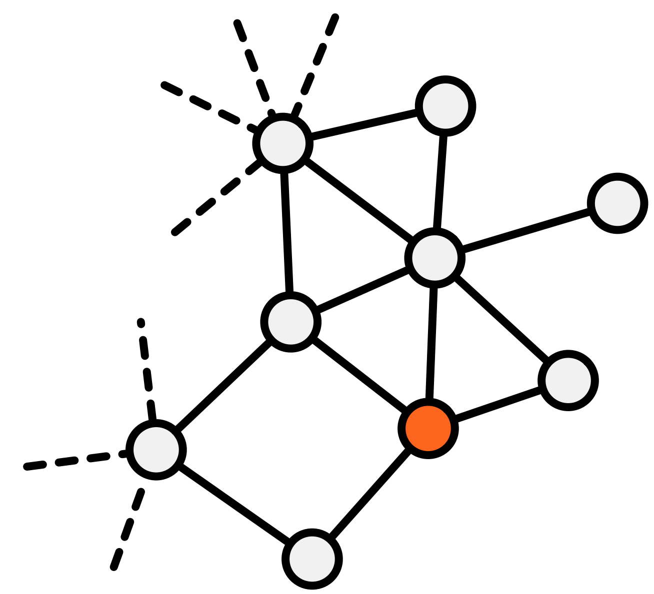
$\mapsto 1$

map from radius-t
neighbourhoods to
local outputs

distributed algorithm



$\mapsto 0$

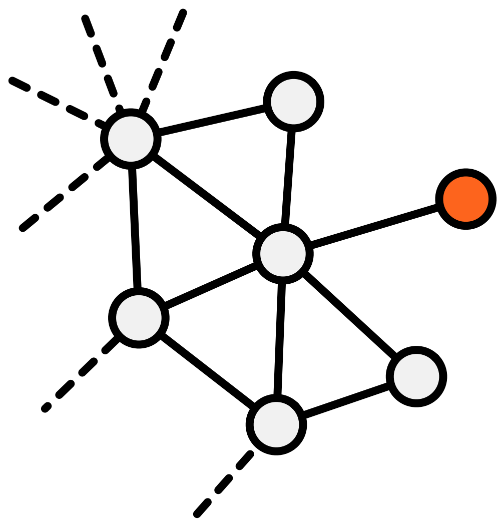


$\mapsto 1$

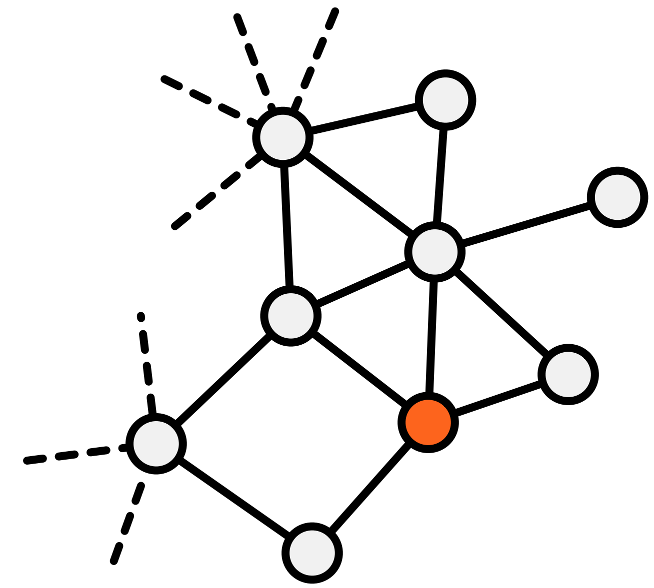
trivial: $t \geq$ diameter

focus: small *t*

distributed algorithm



$\mapsto 0$



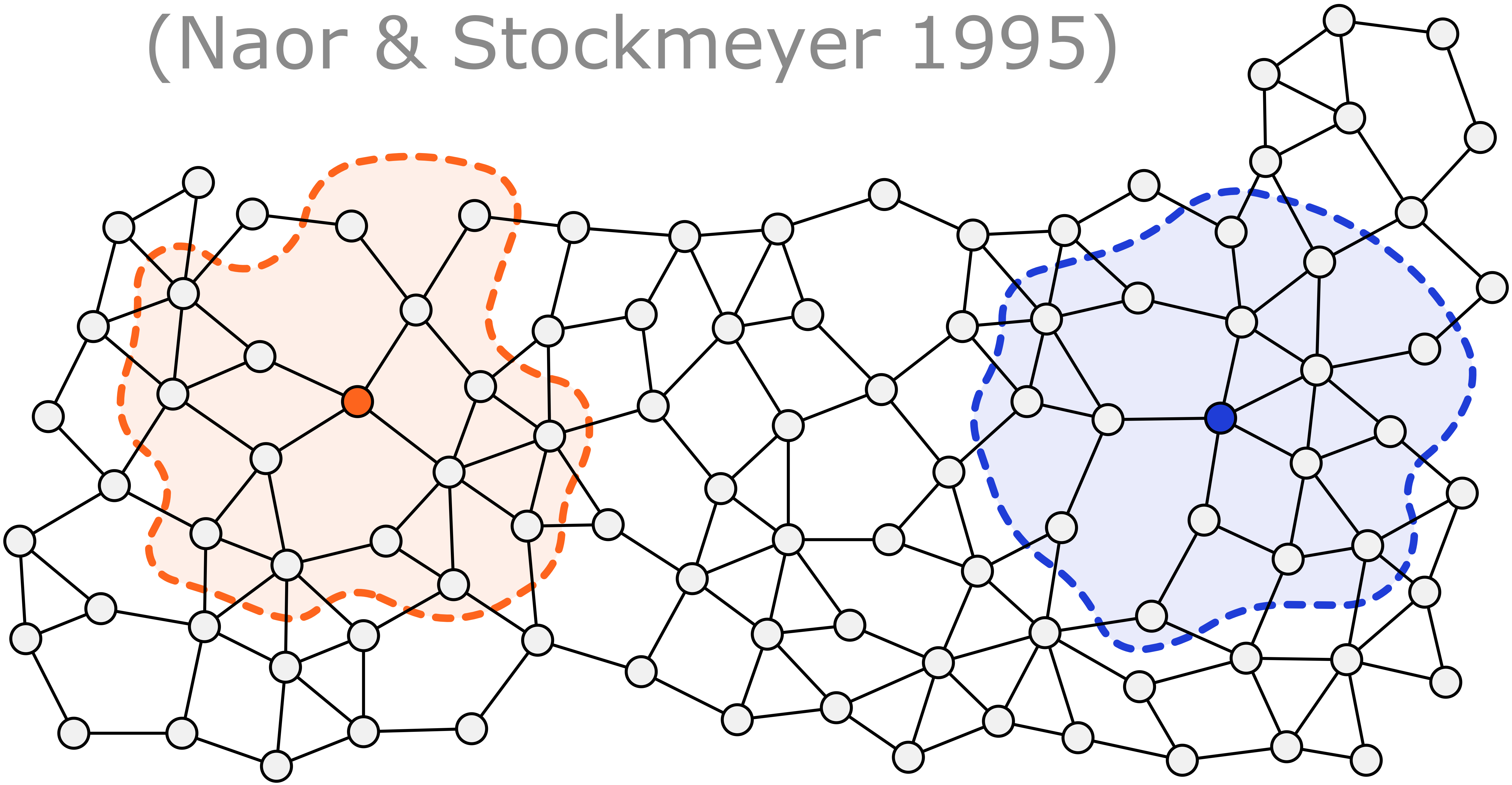
$\mapsto 1$

our research:

local algorithms, $t = O(1)$

“what can be computed locally?”

(Naor & Stockmeyer 1995)



“what can be computed locally?”

- **fast** and **fault-tolerant** distributed algorithms
- understanding social networks, markets, biological systems, …

local algorithms

- **vertex cover:** 2-approx.
- edge dominating sets
- almost stable matchings
- linear programming…

(bounded-degree graphs)

local algorithms

matching lower bounds!

general proof techniques

e.g.: *unique identifiers do not help with local approximation*

decision problems…

distributed graph algorithms

local algorithms: $O(1)$ time

Thanks!