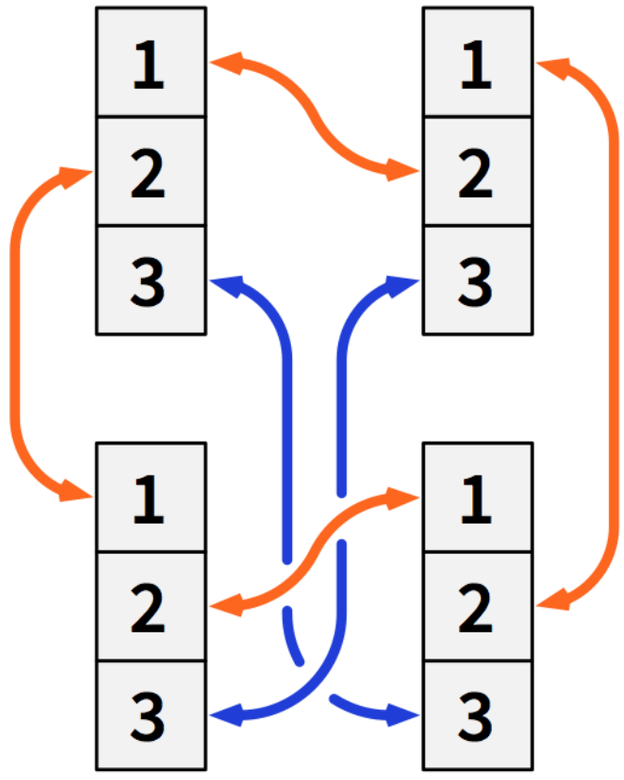
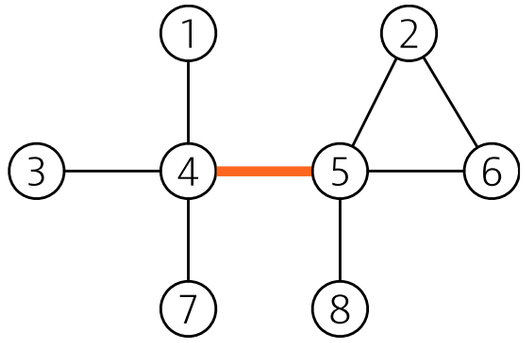
Port Numbering Model

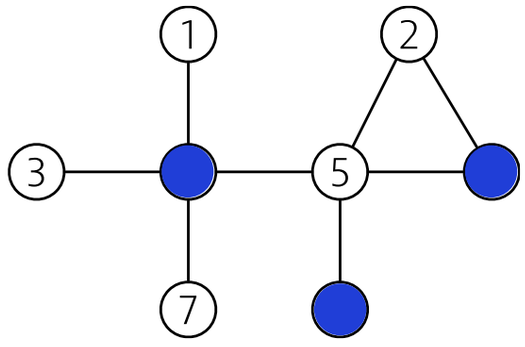


Graph problems

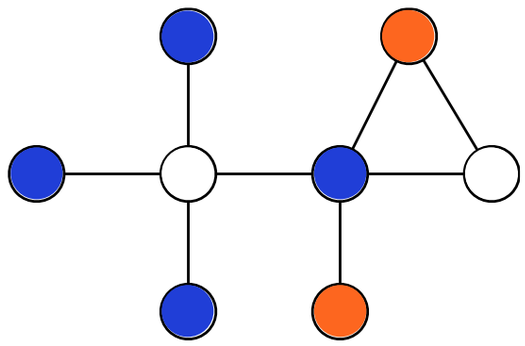
Examples



Matching



Vertex Cover



Coloring

Please do not confuse

- **Maximal**
  - not a subset of another solution
  - very easy to find: add greedily
- **Maximum**
  - largest possible solution
  - often hard to find

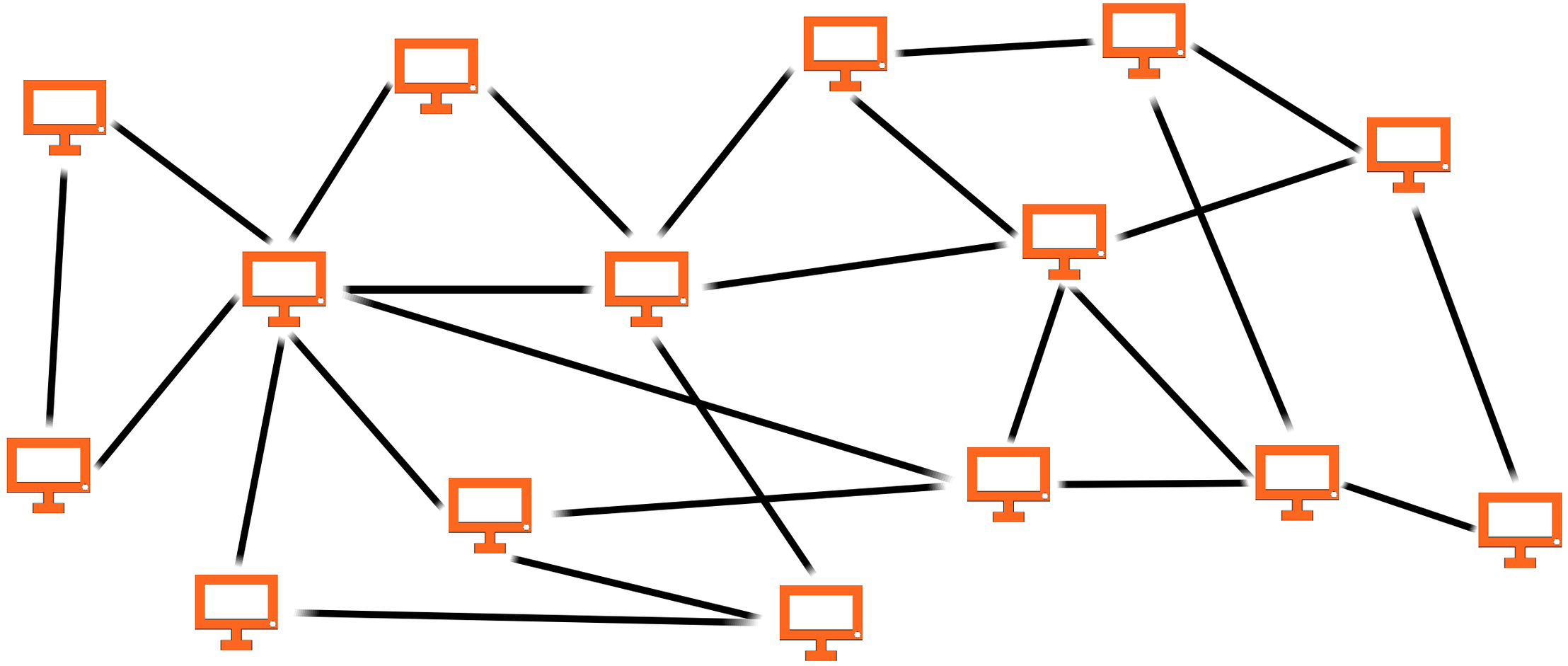
Please do not confuse

- **Minimal**
  - not a superset of another solution
  - very easy to find: remove greedily
- **Minimum**
  - smallest possible solution
  - often hard to find

Distributed algorithms

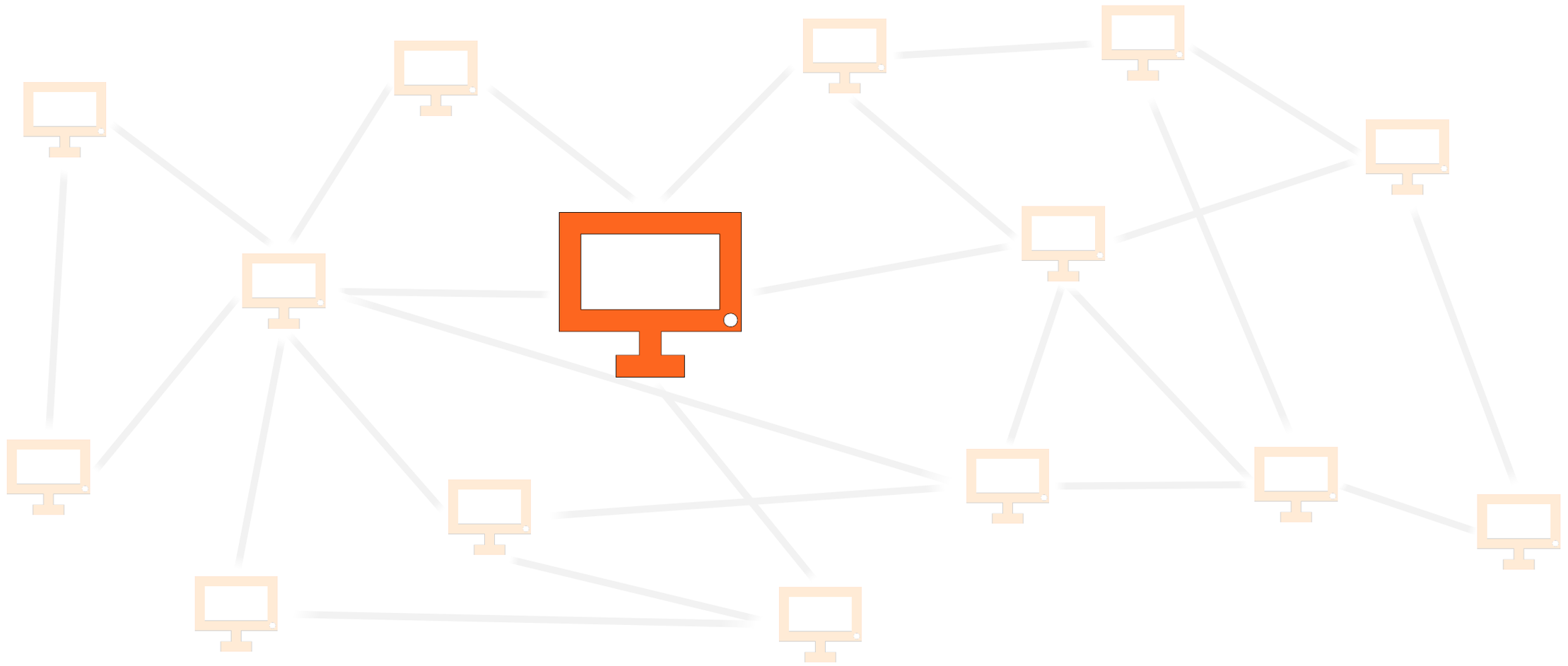
Distributed algorithms

- **Algorithms for computer networks**



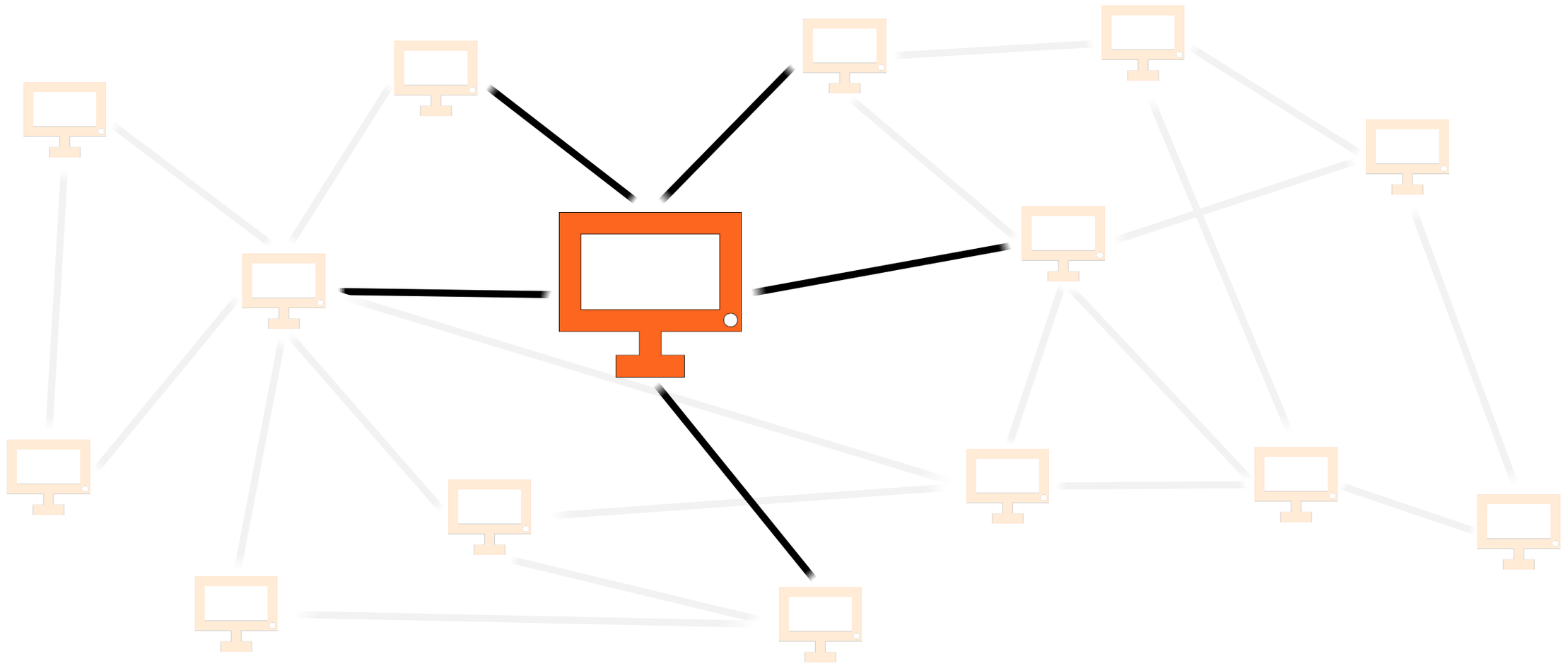
Distributed algorithms

- Algorithms for computer networks



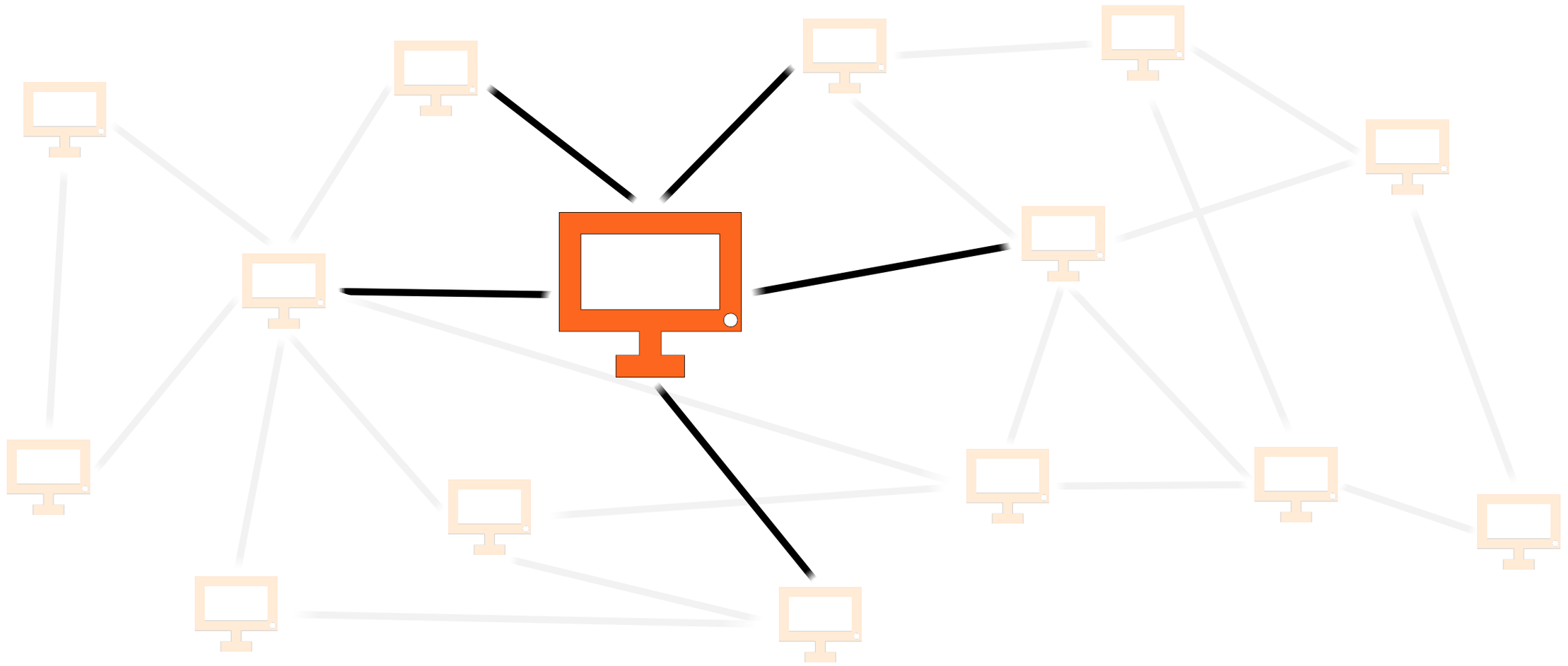
Distributed algorithms

- Algorithms for computer networks



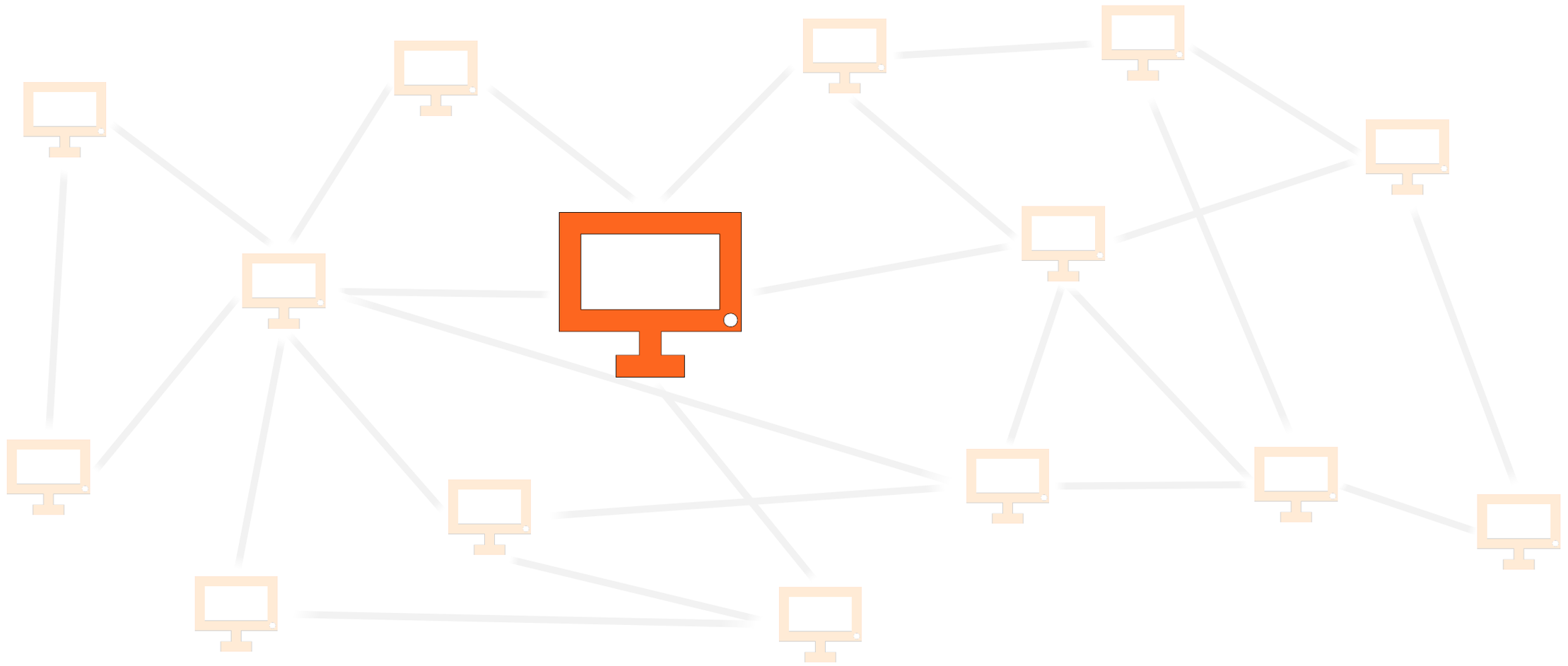
Distributed algorithms

- Algorithms for computer networks
- Synchronous communication model



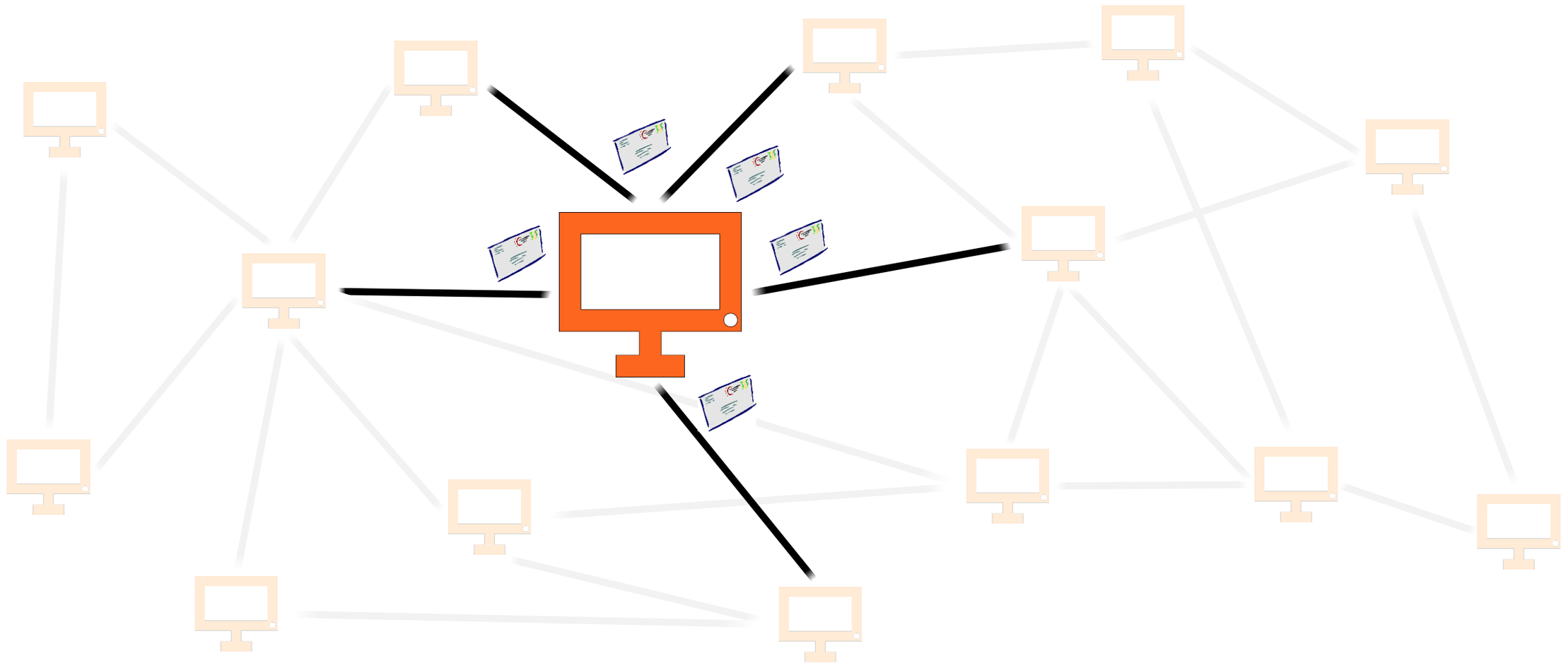
Synchronous distributed algorithm

- **Initially:** each node only aware of *itself*



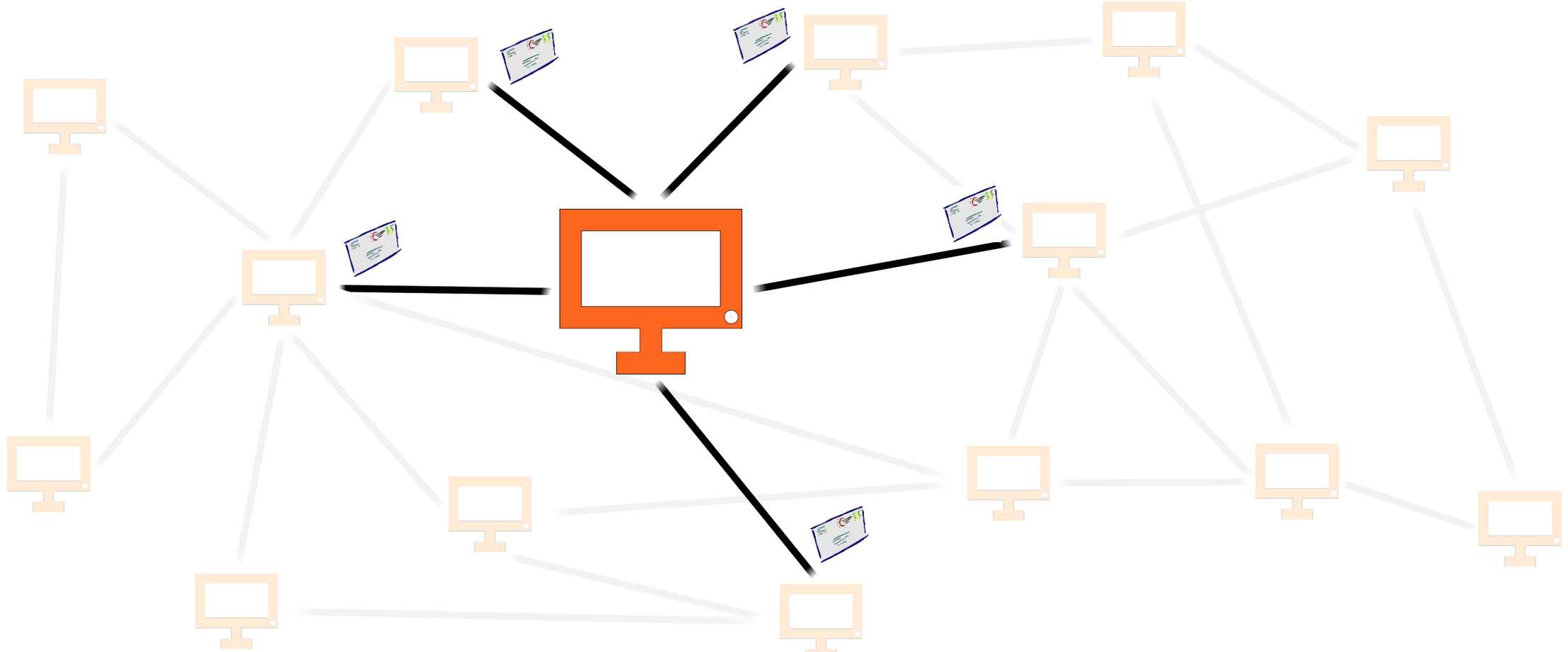
Synchronous distributed algorithm

- **One communication round:**
  - each node *sends* messages to its own neighbors



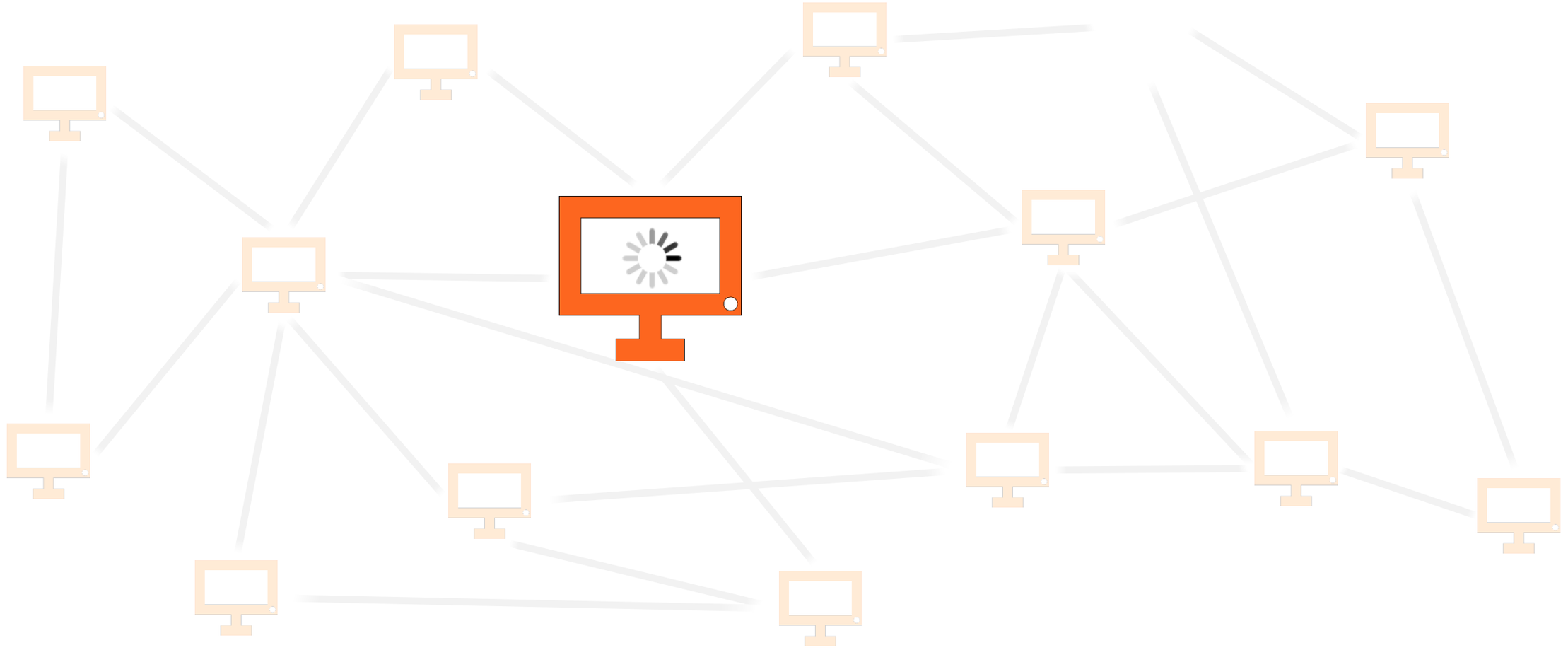
Synchronous distributed algorithm

- **One communication round:**
  - each node *receives* messages from its neighbors



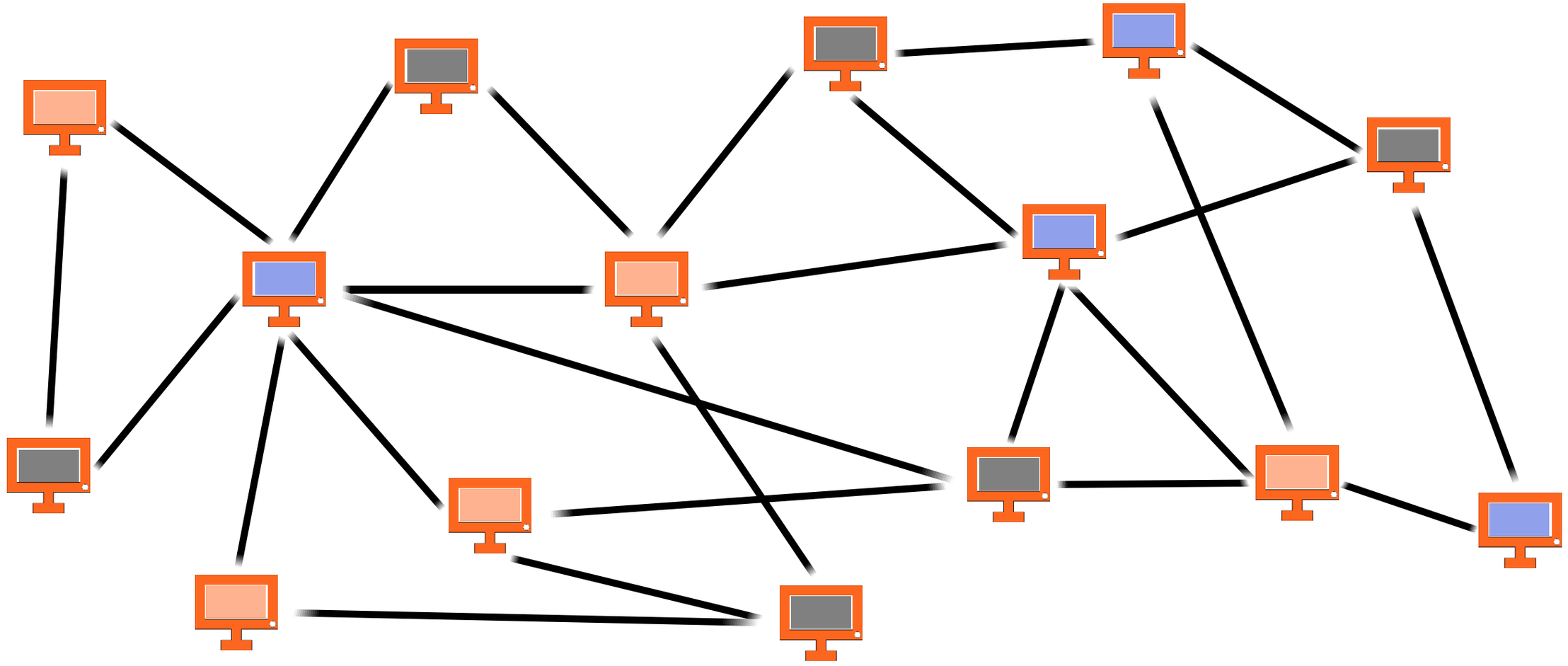
Synchronous distributed algorithm

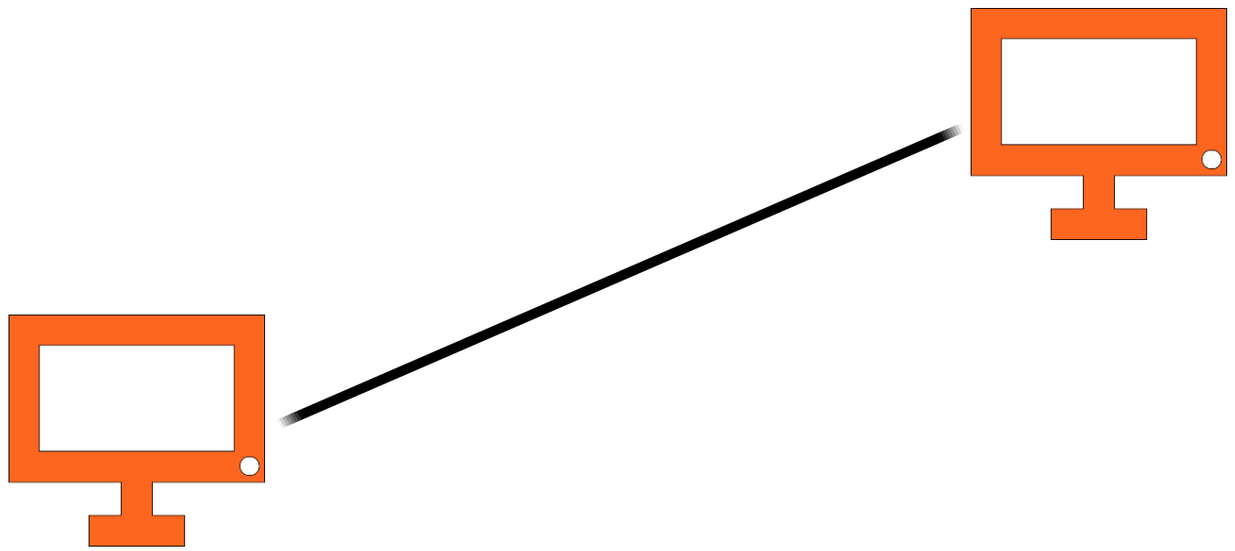
- **One communication round:**
  - each node does some (deterministic) *local computation*



Synchronous distributed algorithm

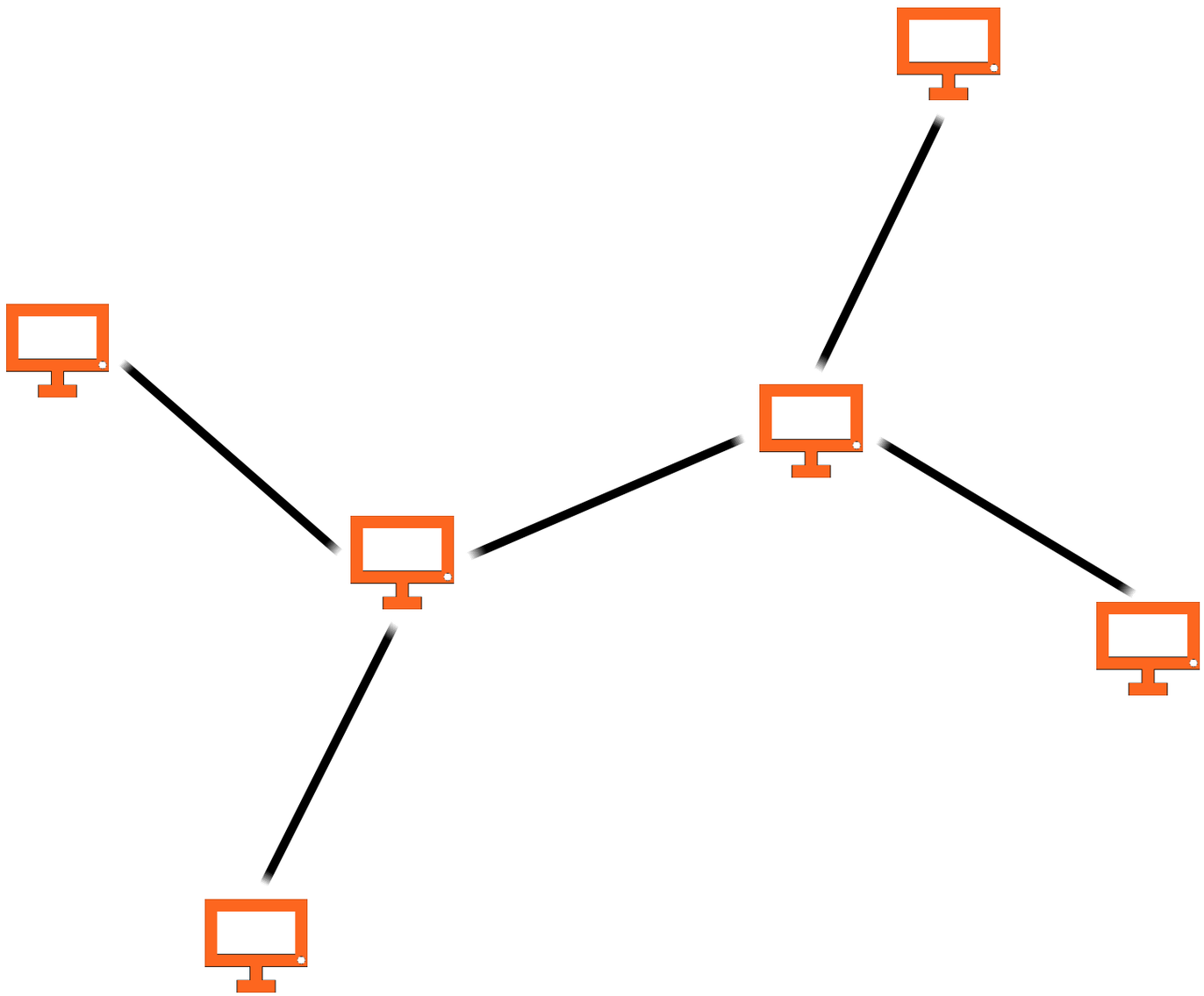
- **Finally:** each node knows *its own part of the solution*

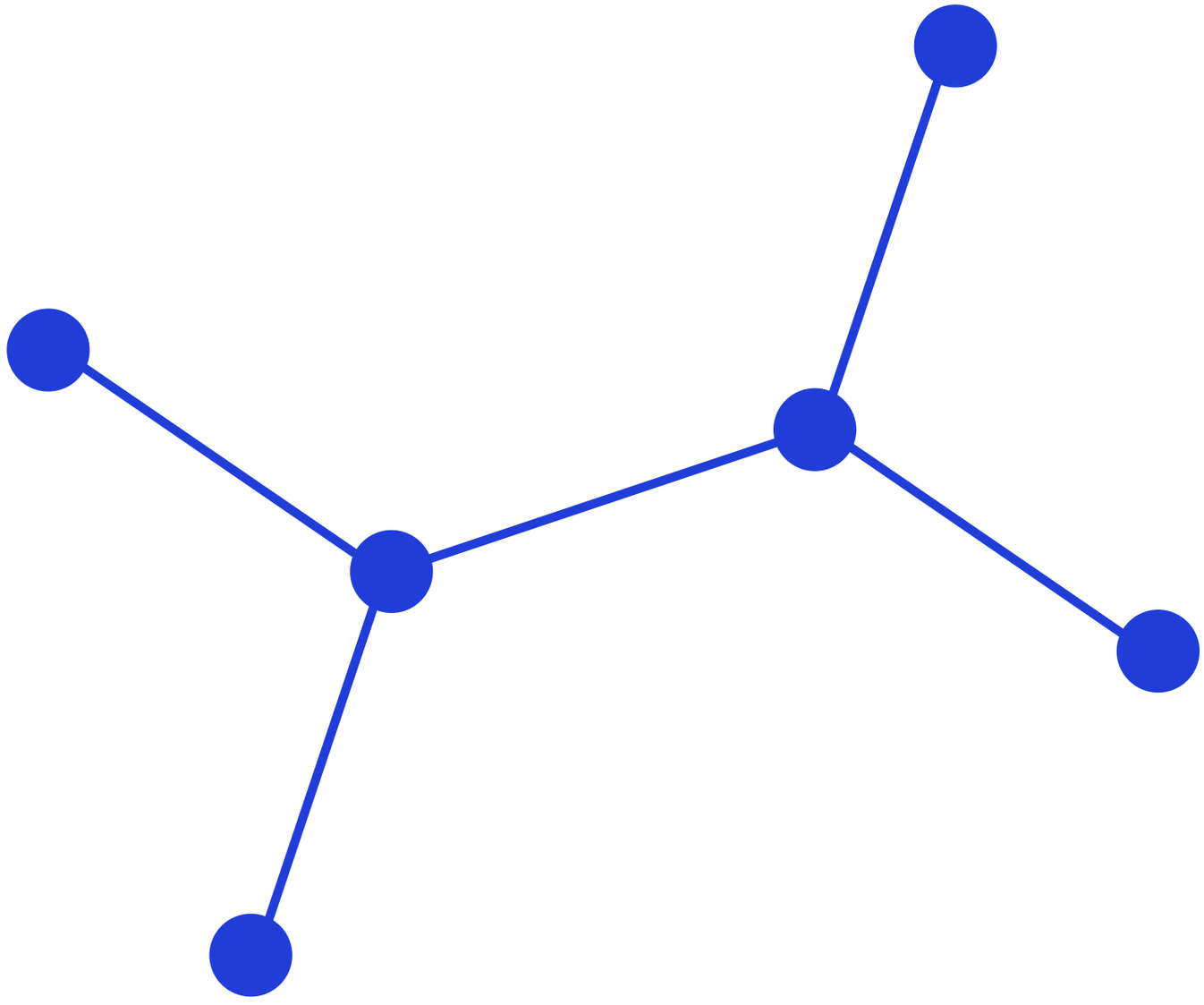


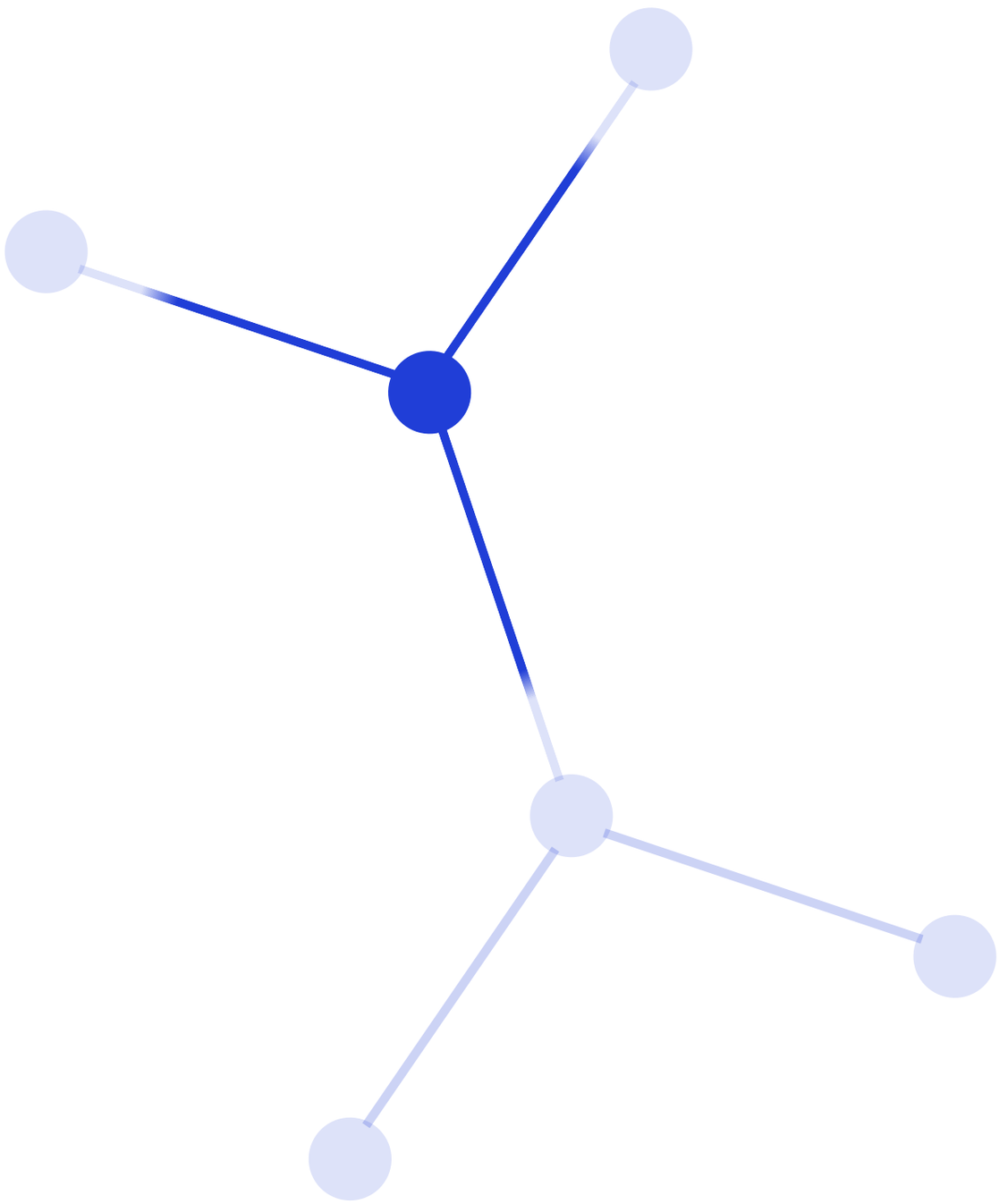


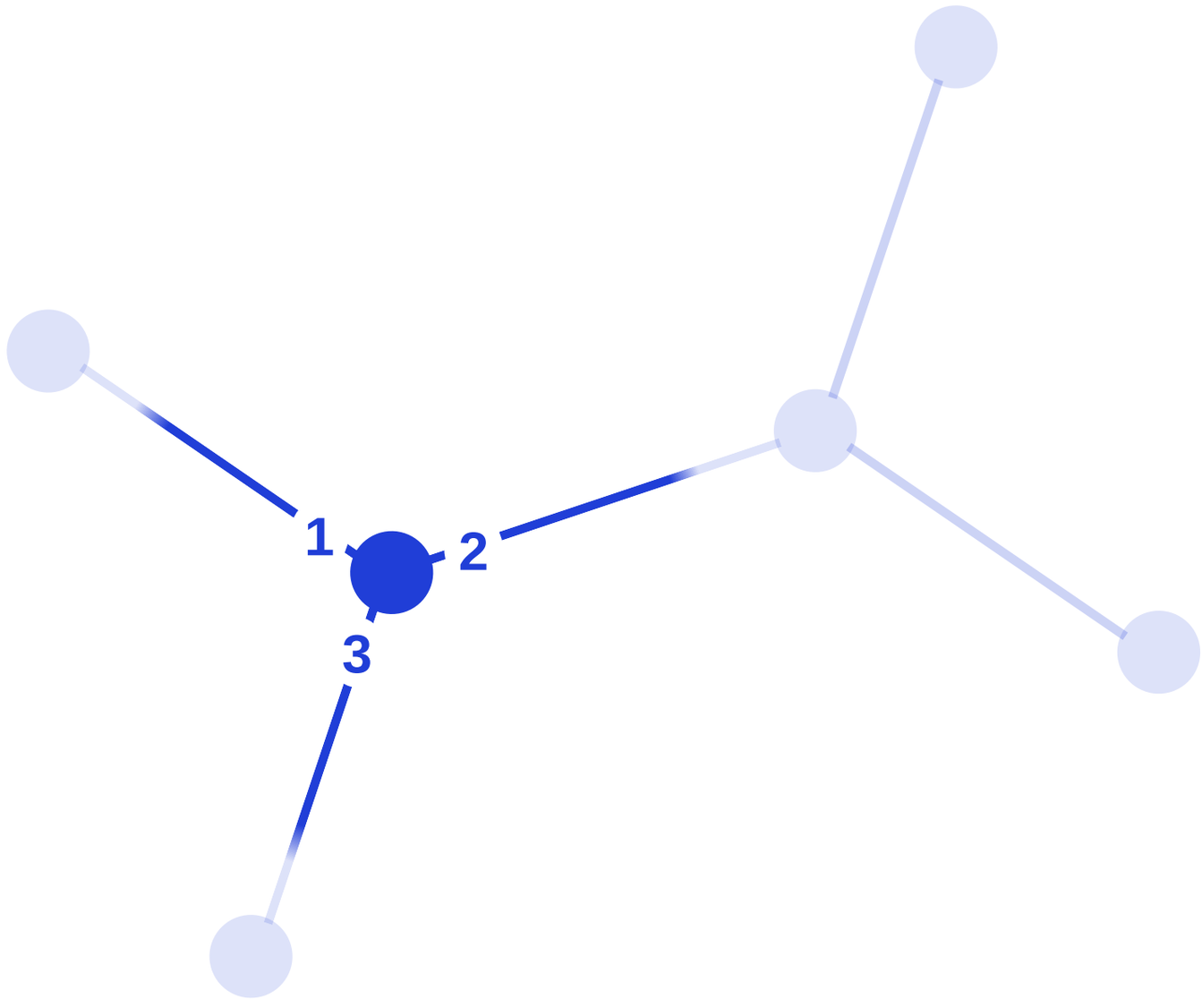
Examples

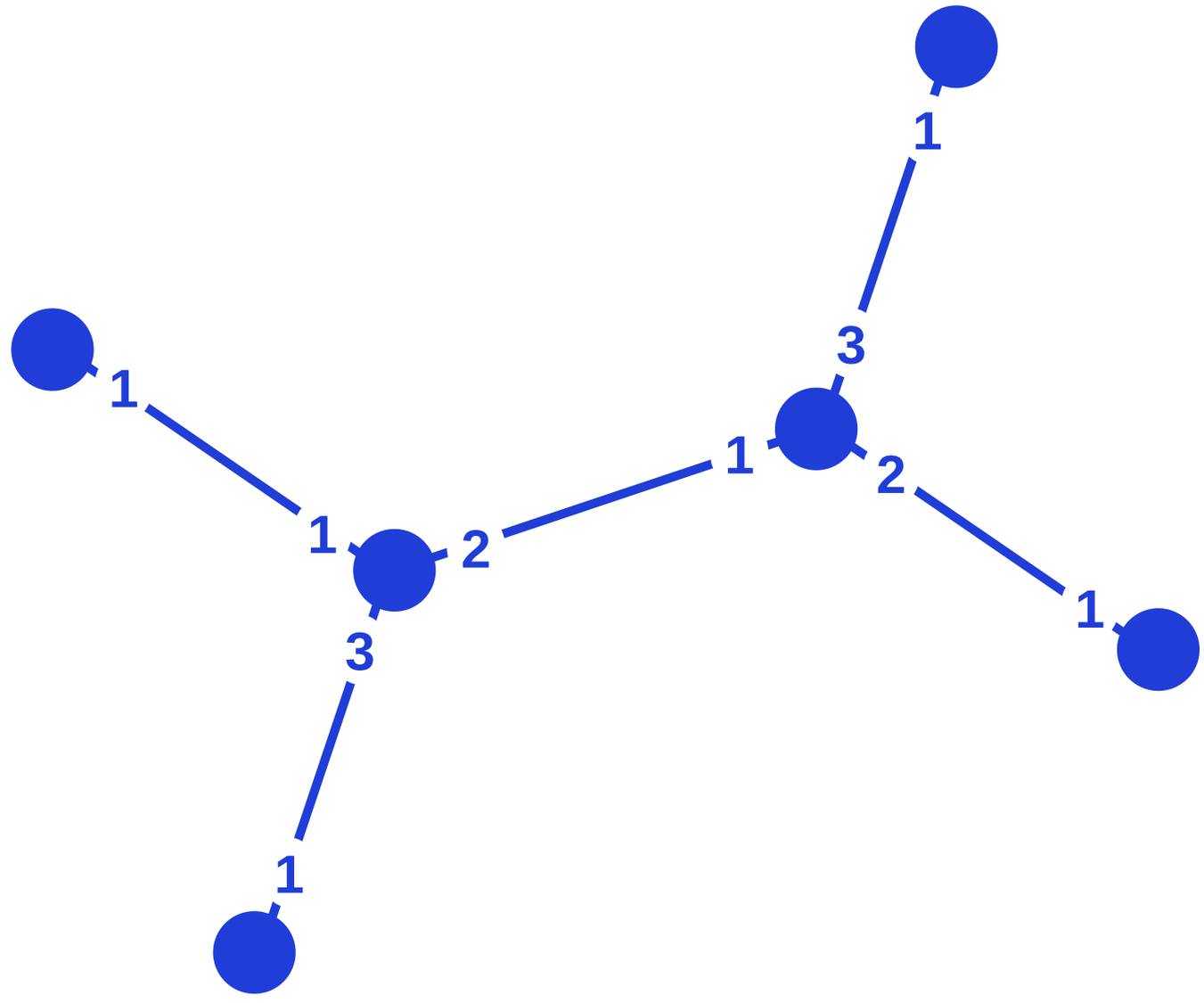
Port-numbered networks

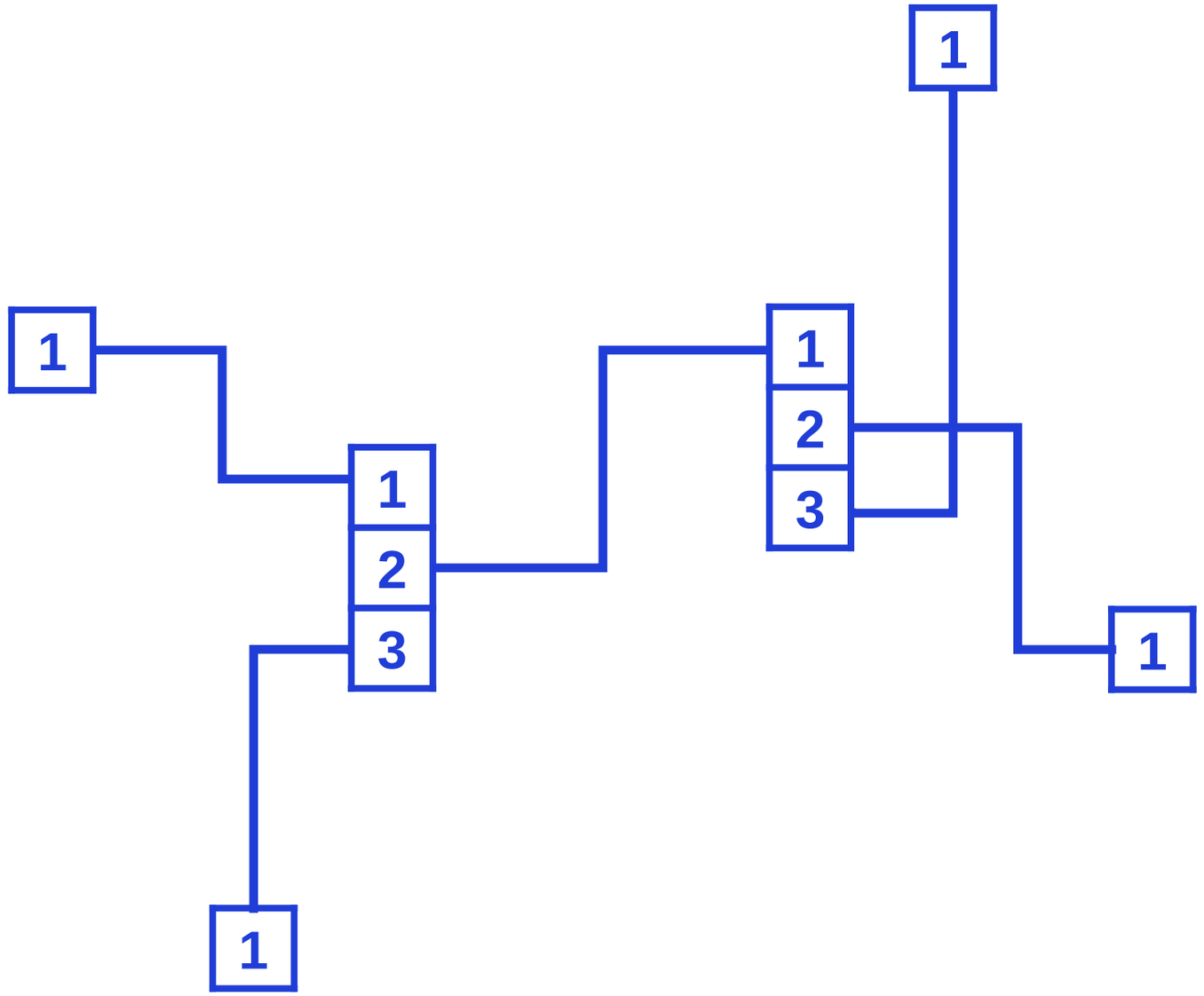


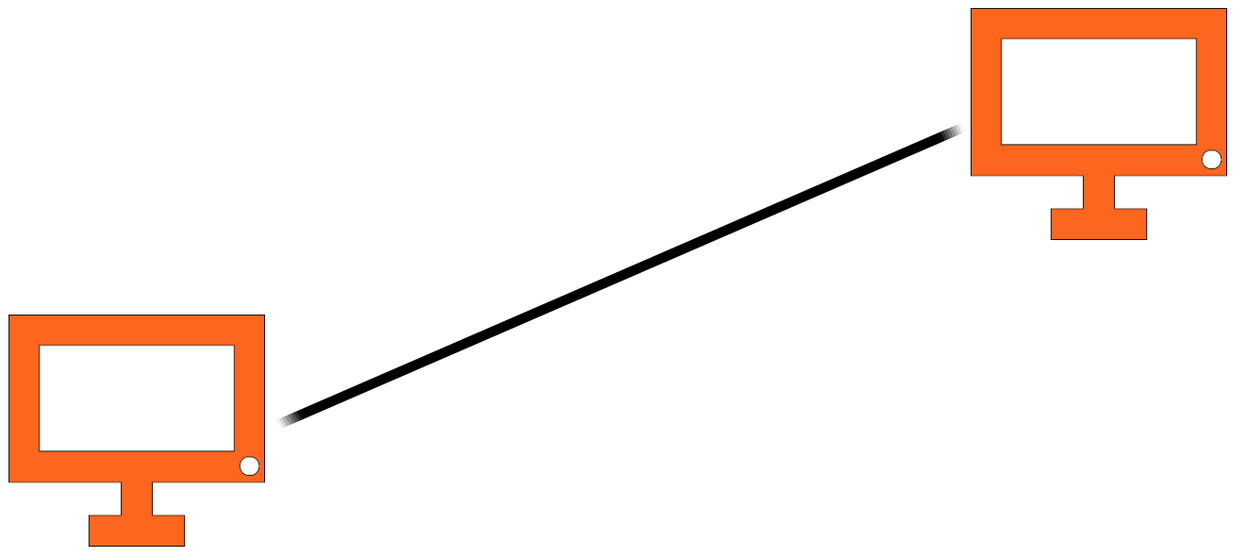





1
2
3


1
1
1
3
1
2
3
2
1
1
1

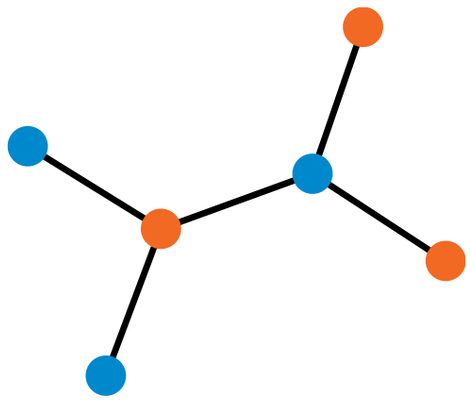

1
1
1
2
3
1
2
3
1
1



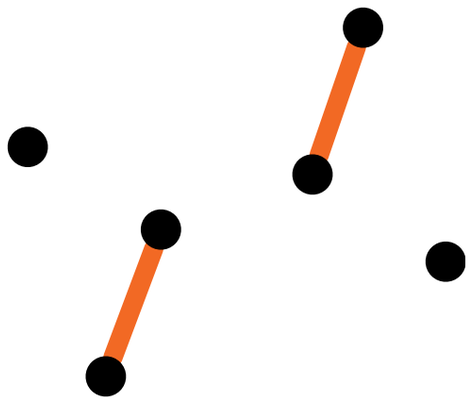
Non-empty matching

Bipartite maximal matching

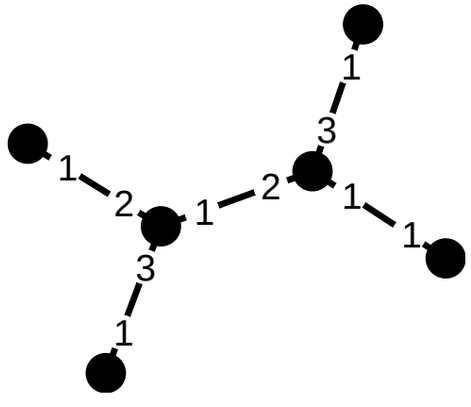
Bipartite graphs are 2-colorable graphs



Input:
proper 2-coloring



Output:
maximal matching



Model of computing:
PN model

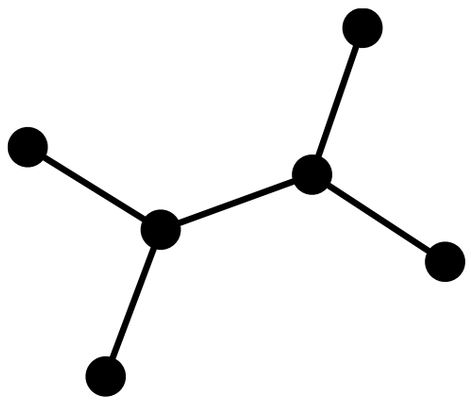
Algorithm

- *Orange* nodes send *proposals* to their neighbors, one by one
  - order by port numbers
- *Blue* nodes *accept* the first proposal they get, reject everything else
  - break ties by port numbers

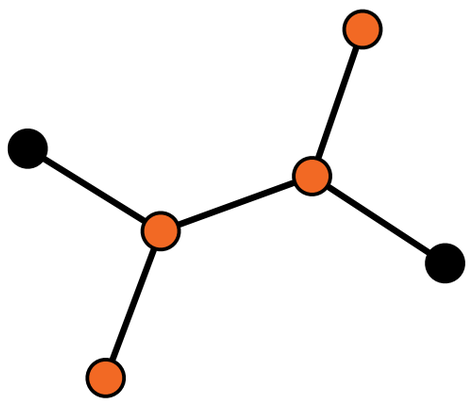
Examples

Minimum vertex cover approximation

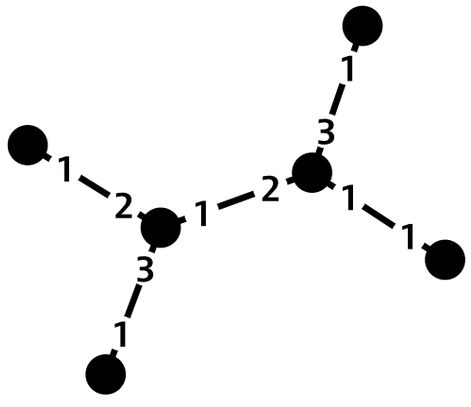
α-approximation of a minimum vertex cover = vertex cover that is at most α times as large as the smallest vertex cover



Input:
nothing



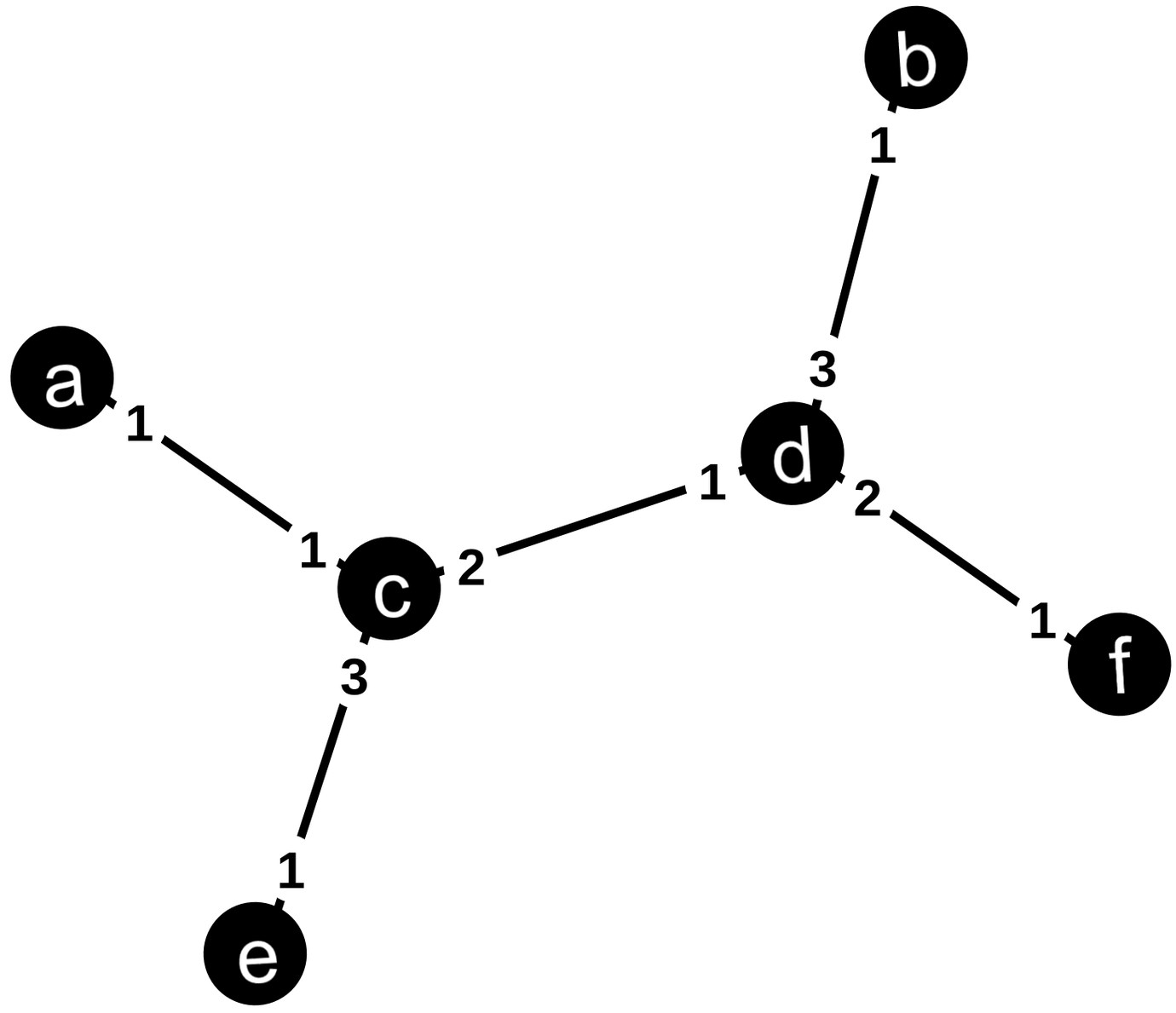
Output: 4-approximation of minimum vertex cover



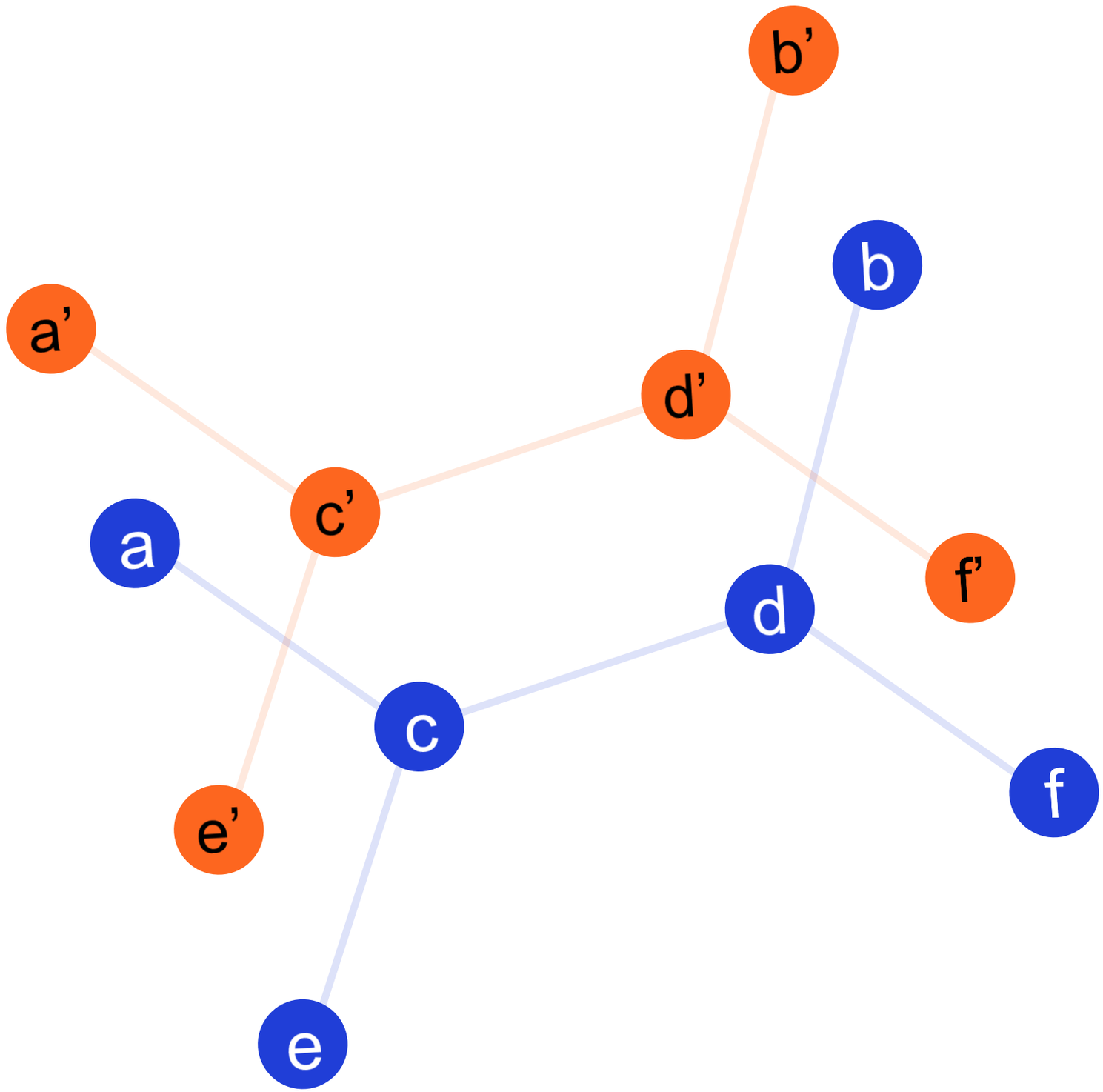
Model of computing:
PN model

Bipartite double cover

Graph *G*

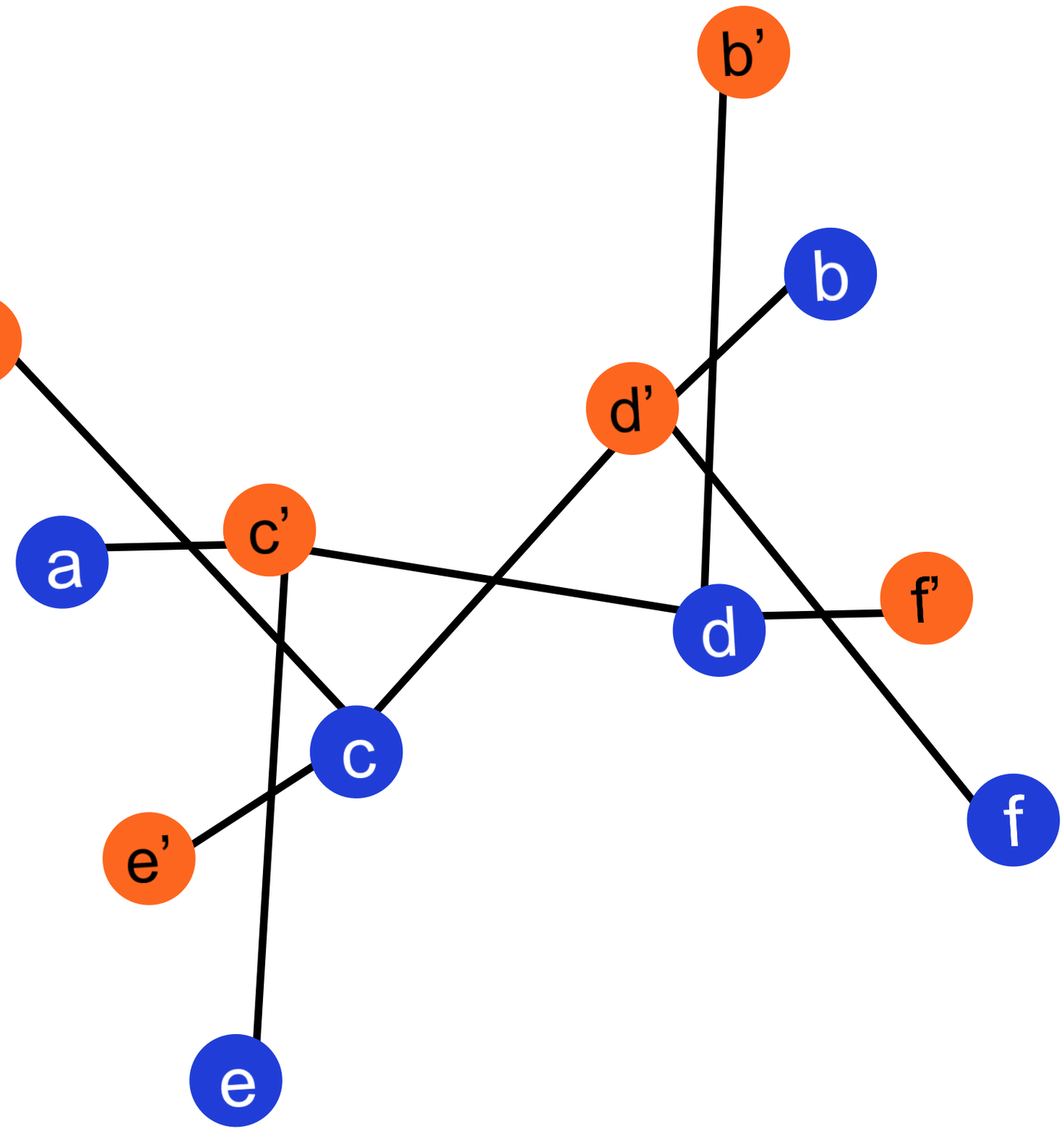


Bipartite double cover



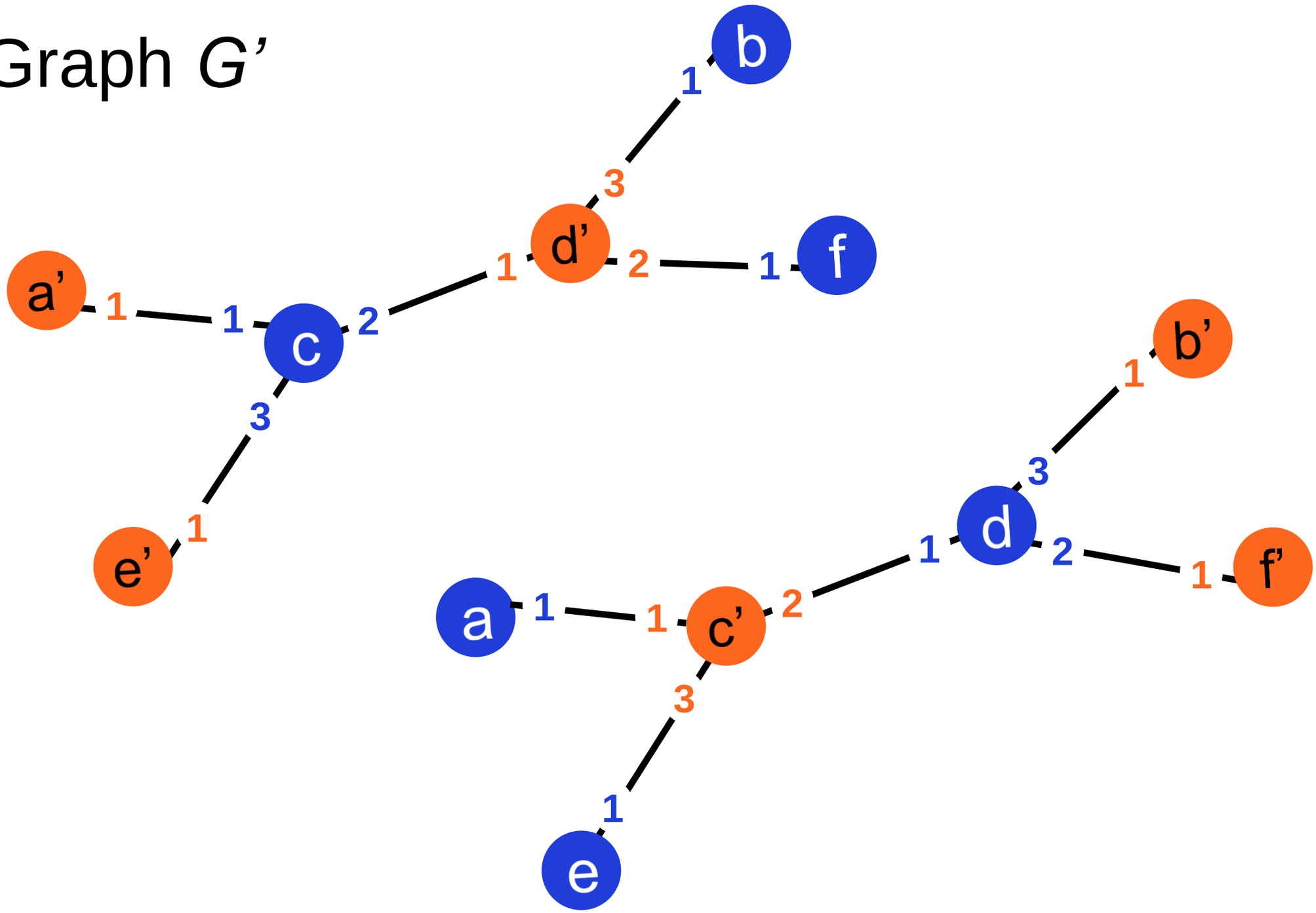
Bipartite double cover

Graph *G’*

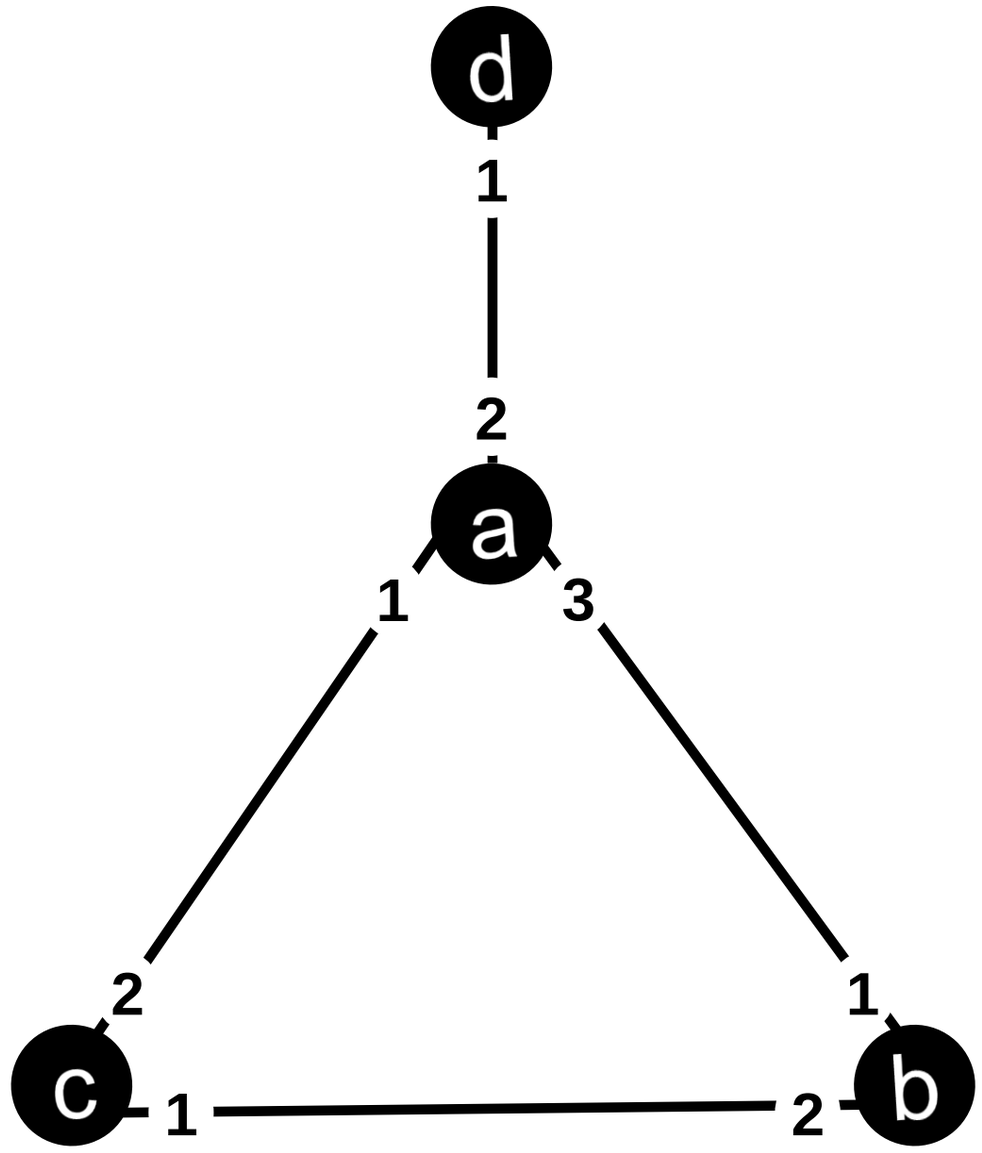


Bipartite double cover

Graph *G’*



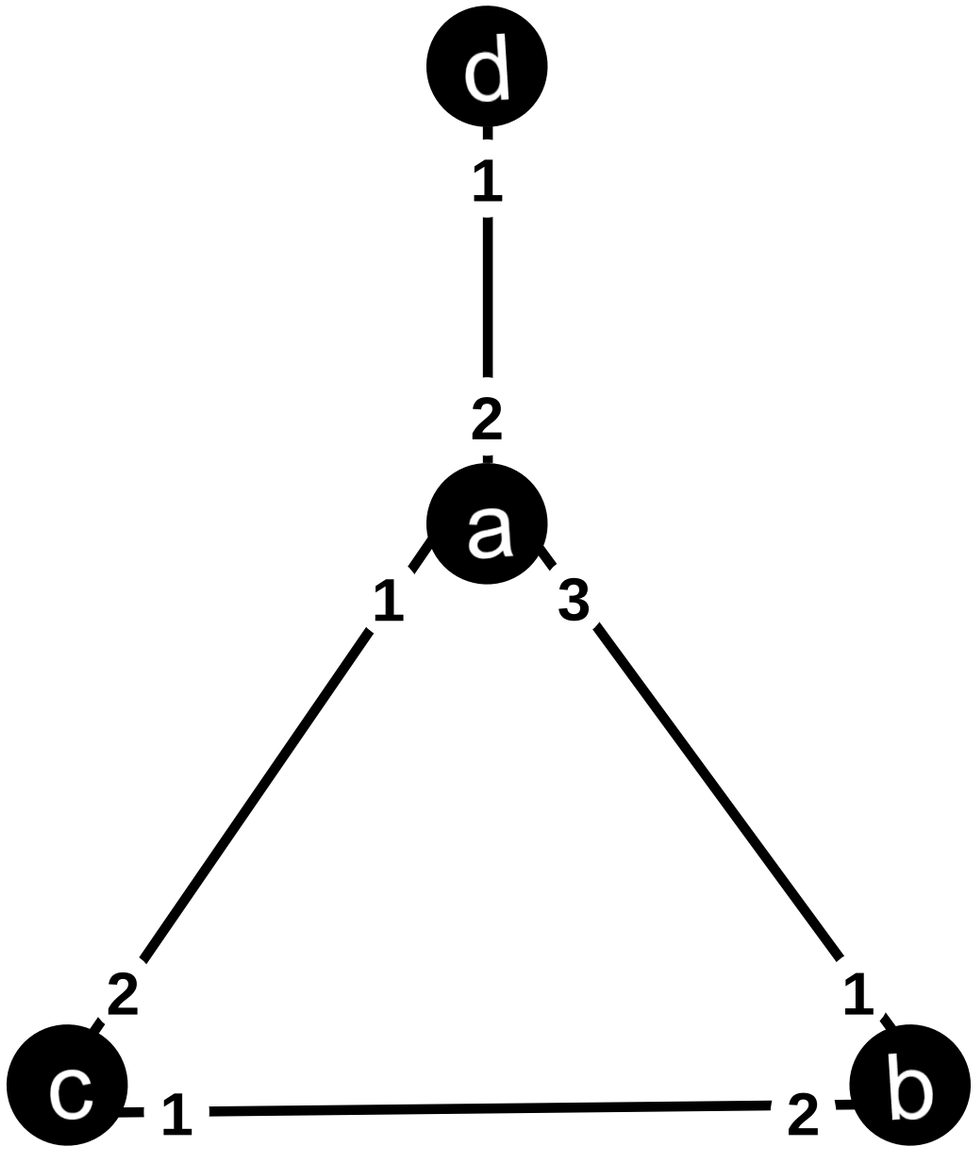
Examples



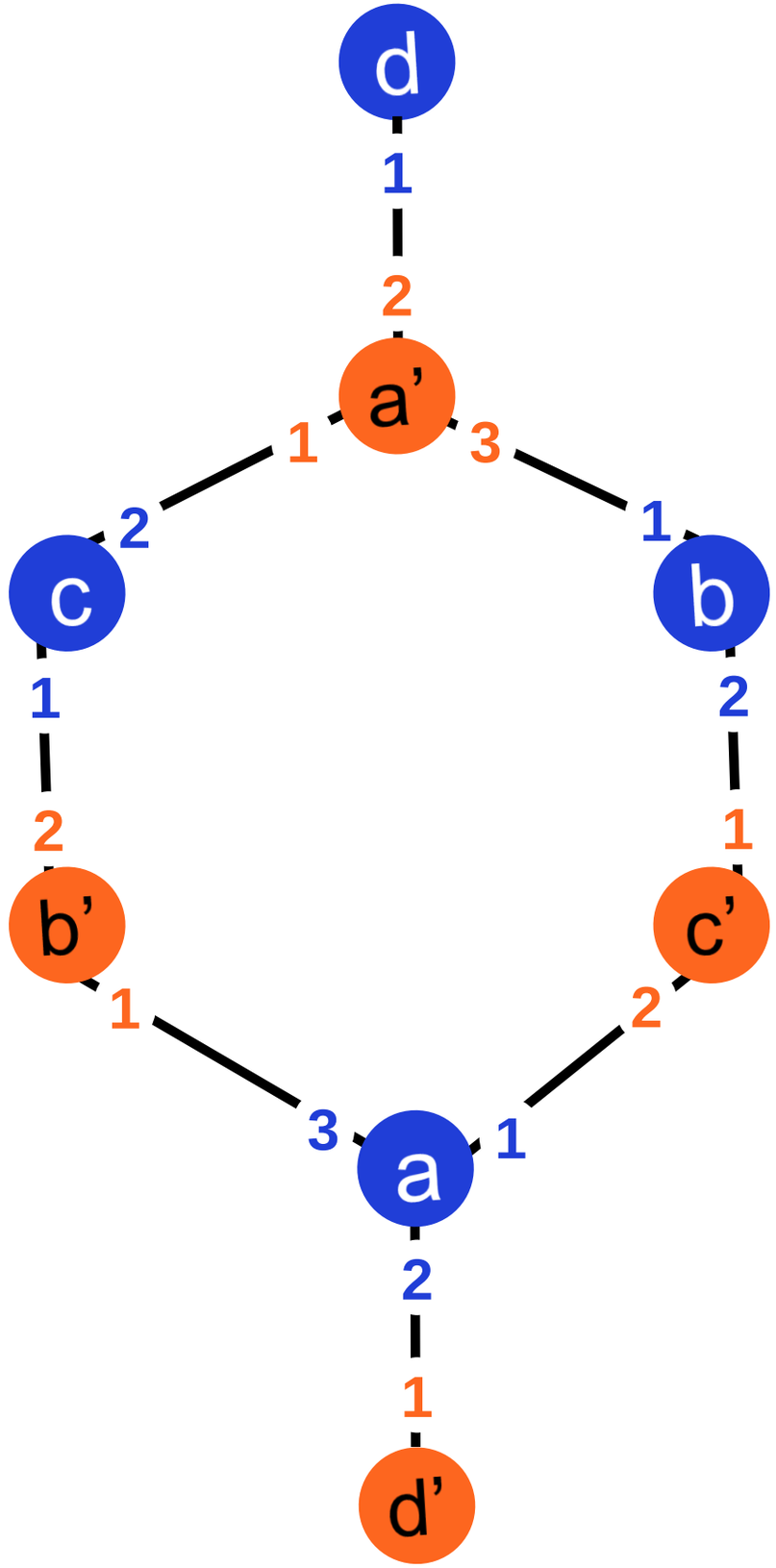
Algorithm

- Construct bipartite double cover *G'*
  - one node in *G*: two virtual copies in *G'*
  - one edge in *G*: two virtual copies in *G'*
- Find a maximal matching *M'* in *G'*
- Take all original nodes of *G* whose virtual copies are matched in *M'*

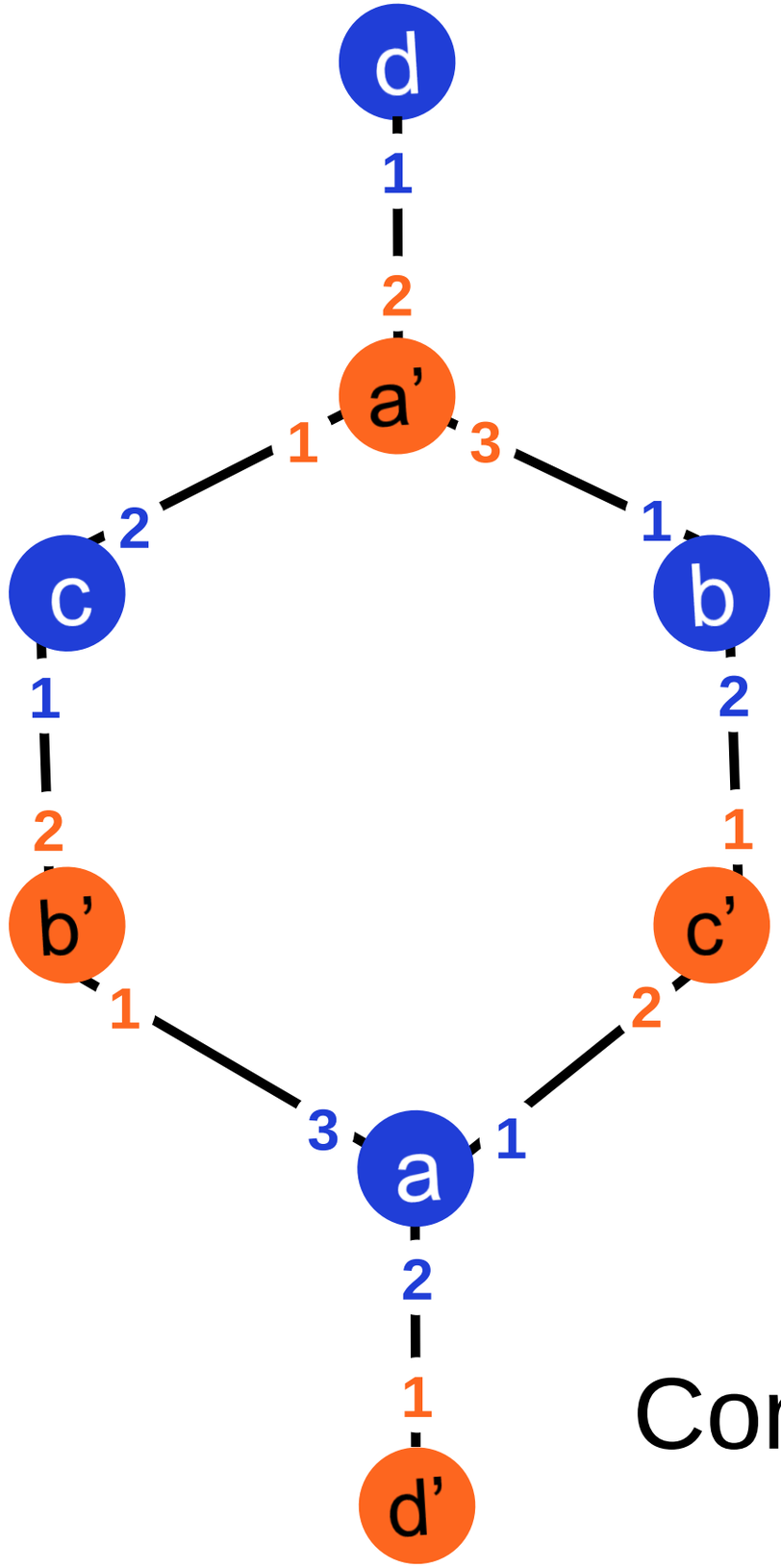
Graph *G*



Graph *G’*

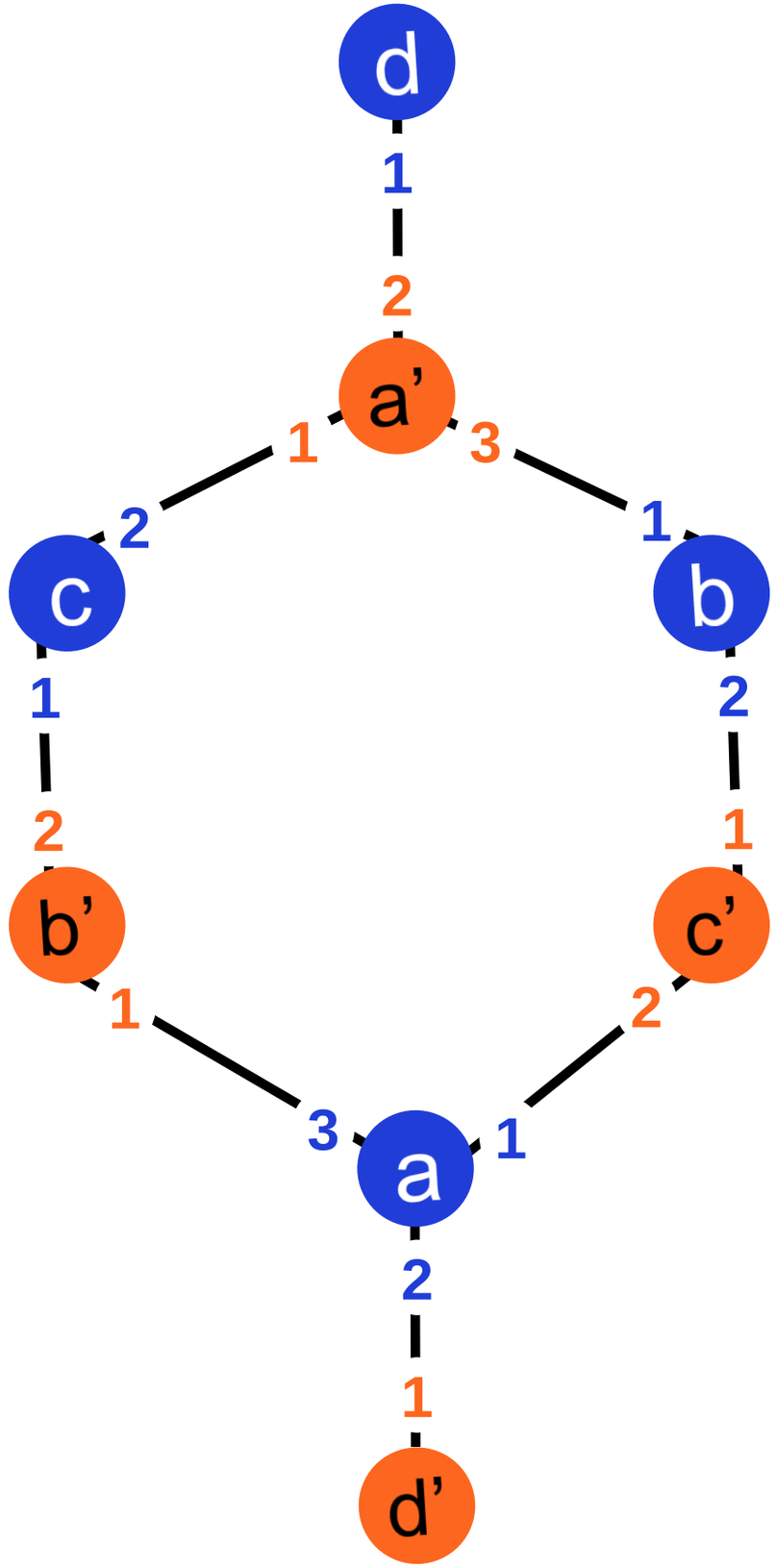


Graph *G’*

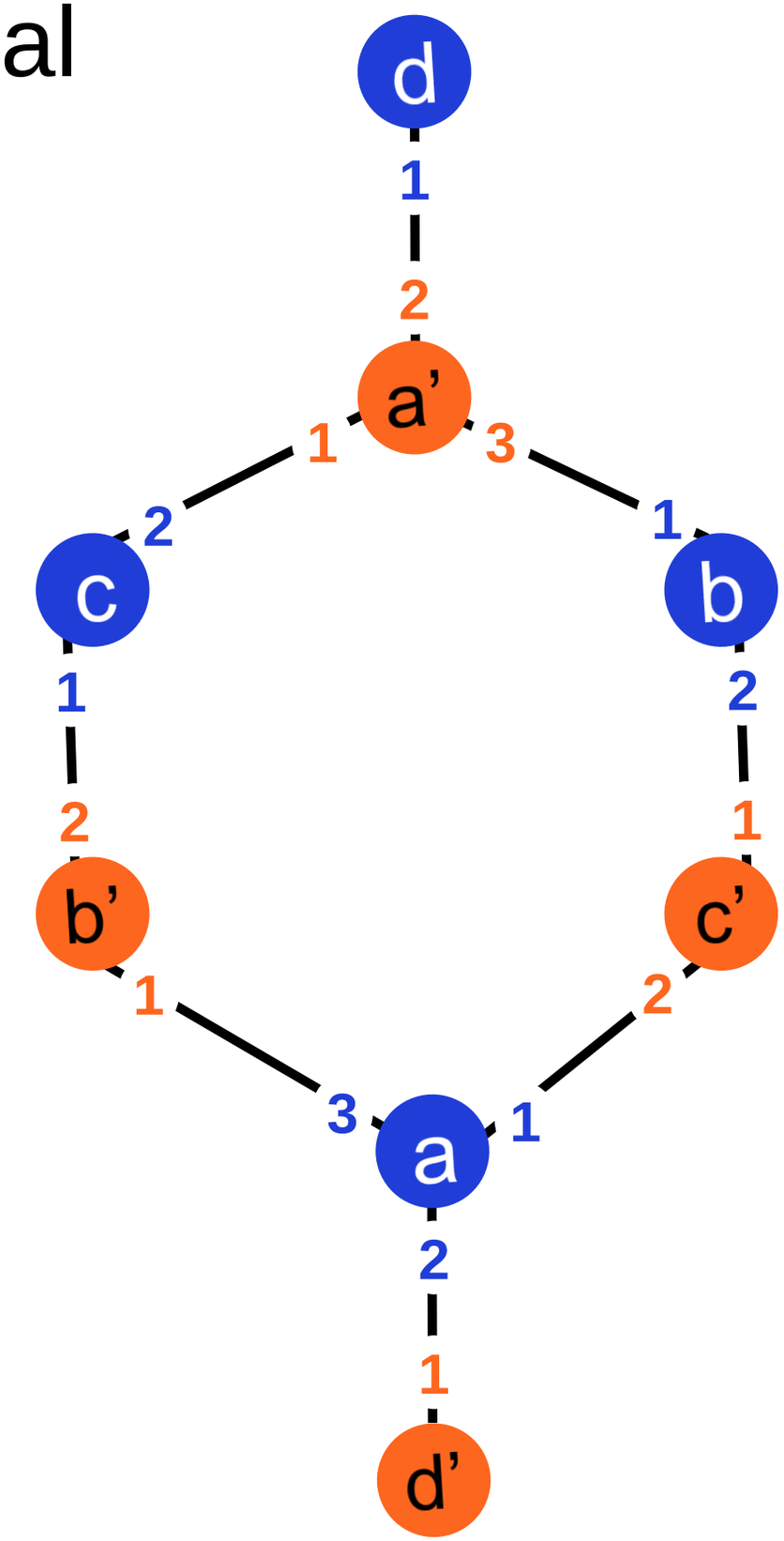


Compute a matching!

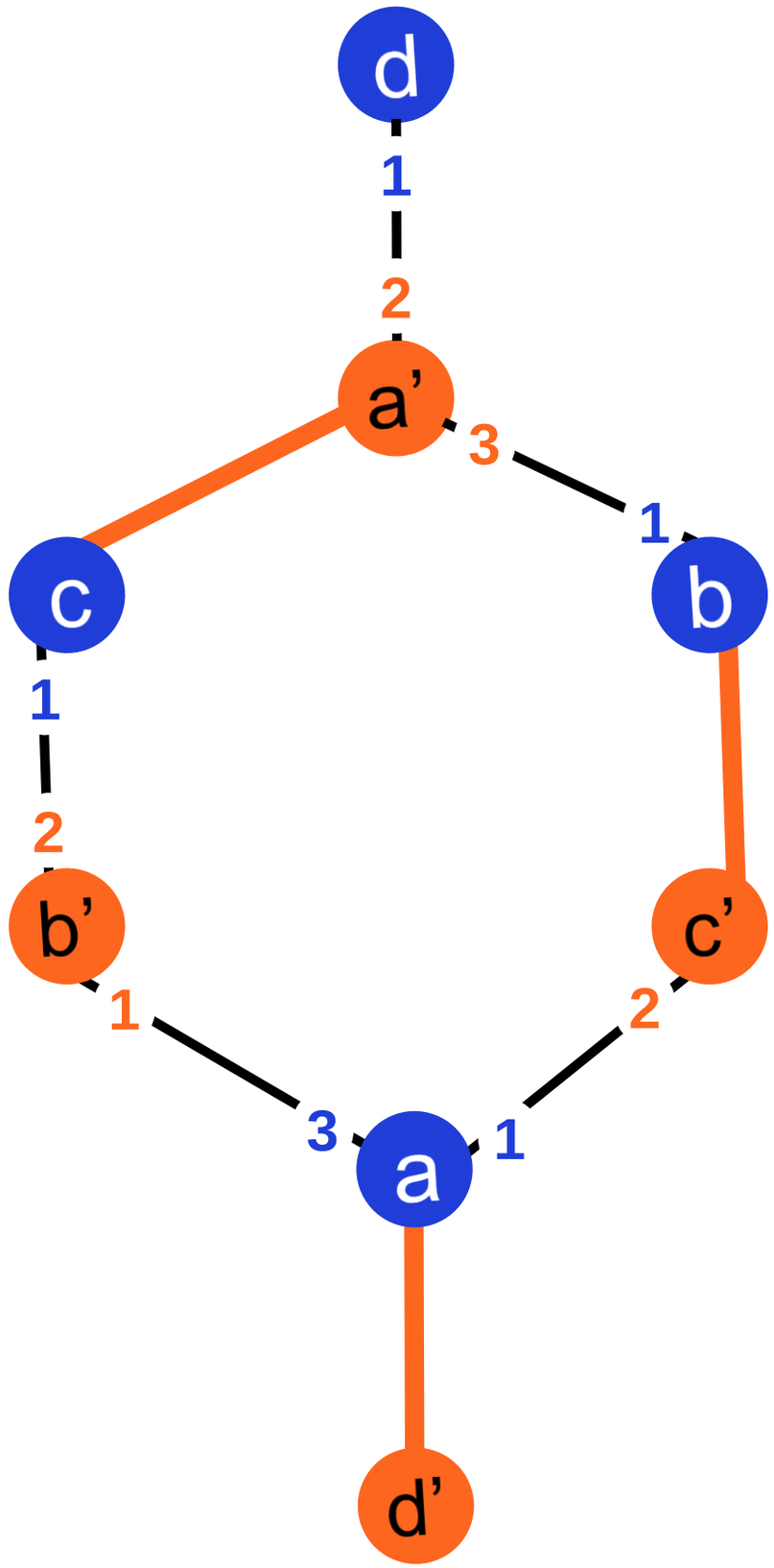
Orange nodes send proposals



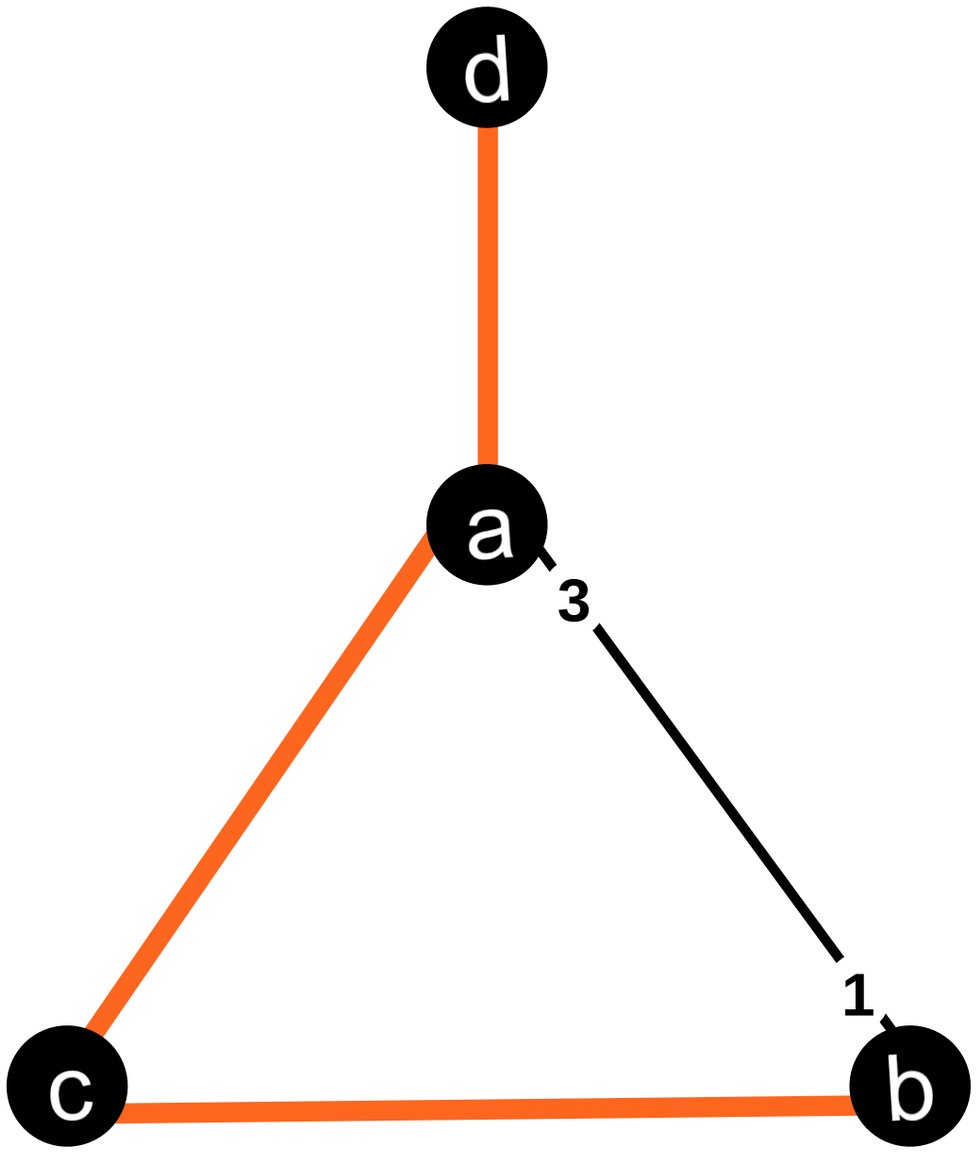
Blue nodes accept/
reject proposal



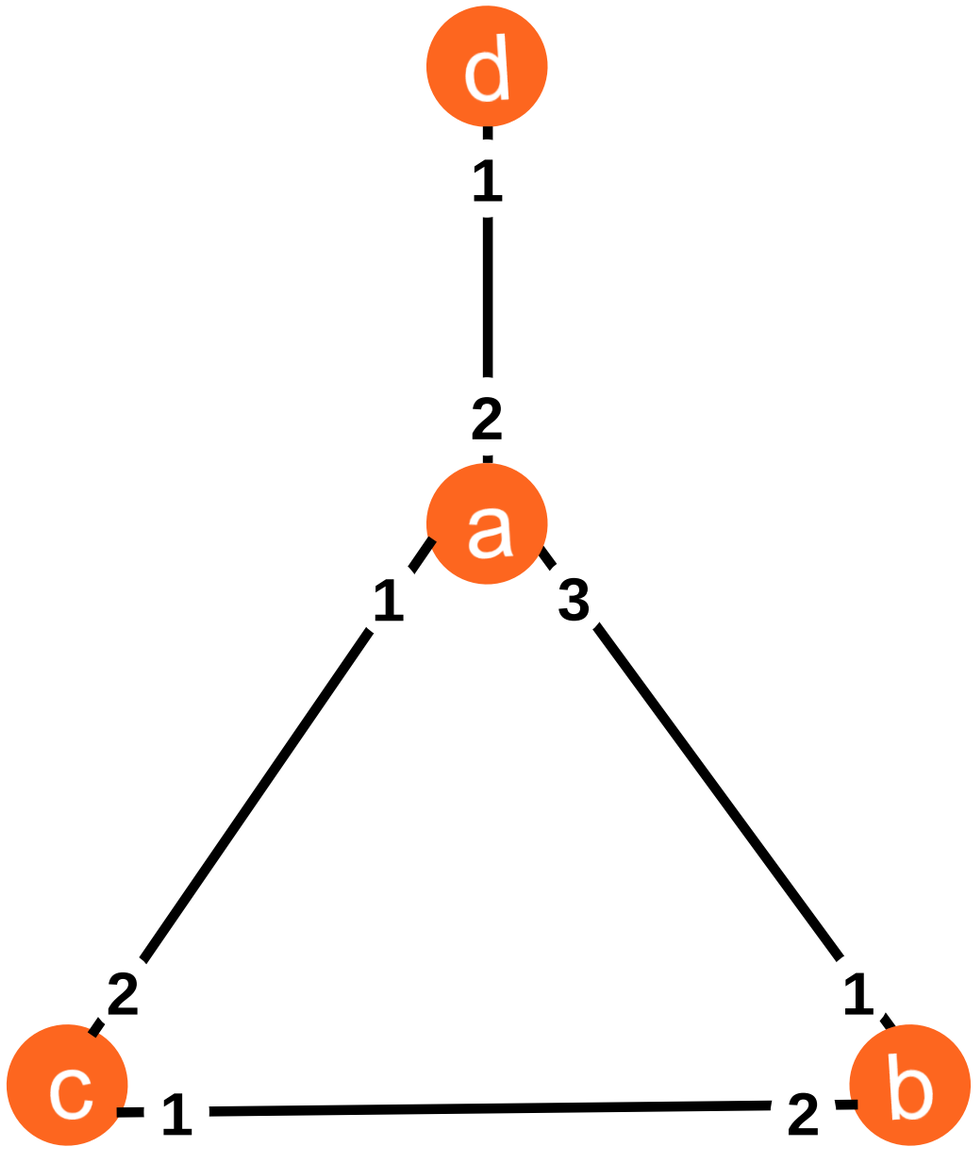
Maximal Matching on *G'*



Virtual matching edges on *G*



Vertex cover on *G*



This algorithm

- terminates…
  - *because the maximal matching algorithm terminates in 2Δ rounds.*
- computes a vertex cover.
  - *Idea: endpoints of any maximal matching form a vertex cover*
- returns at most a 4-approximation of the minimum vertex cover.
  - *Idea: any maximal matching is a 2-approximation of the minimum vertex cover*

Learning goals

- **Graph problems:**
  - (bipartite) matching, vertex cover, coloring
- **Distributed models:**
  - Synchronous communication model
  - Port numbering model
- **Algorithms:**
  - Bipartite maximal matching
  - 4-Approximation of the minimum vertex cover