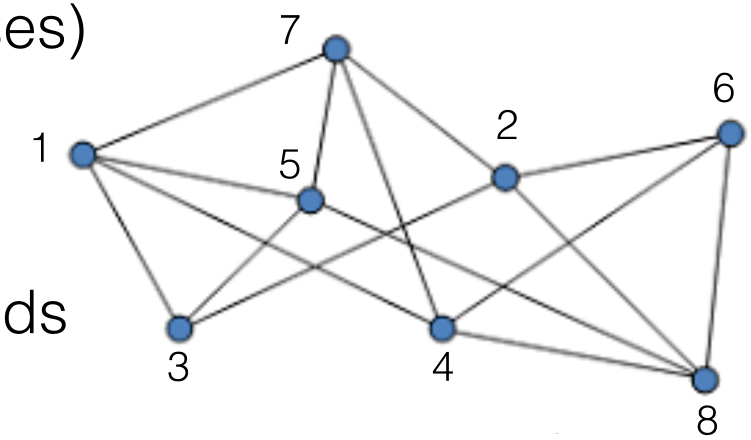
CONGEST Model

- Definition
- Local and global problems
- Solving local problems
- Lower bounds

CONGEST Model

- Each process is located at a node of a network modeled as an n-node graph (n = #processes)



- Each process has a unique ID in $\{1,\ldots,n\}$
- Computation proceeds in synchronous rounds during which every process:

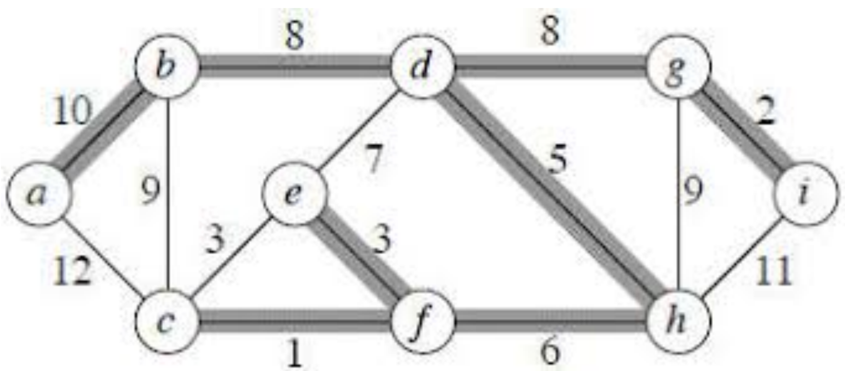
  1. Sends a message to each neighbor
  2. Receives a message from each neighbor
  3. Performs individual computation (same algorithm for all nodes)

Typically, $B = O(\log n)$

Non Local Problems

- In LOCAL, all (Turing constructible) problems can be solved in $O(D)$ rounds in graphs with maximum diameter D.

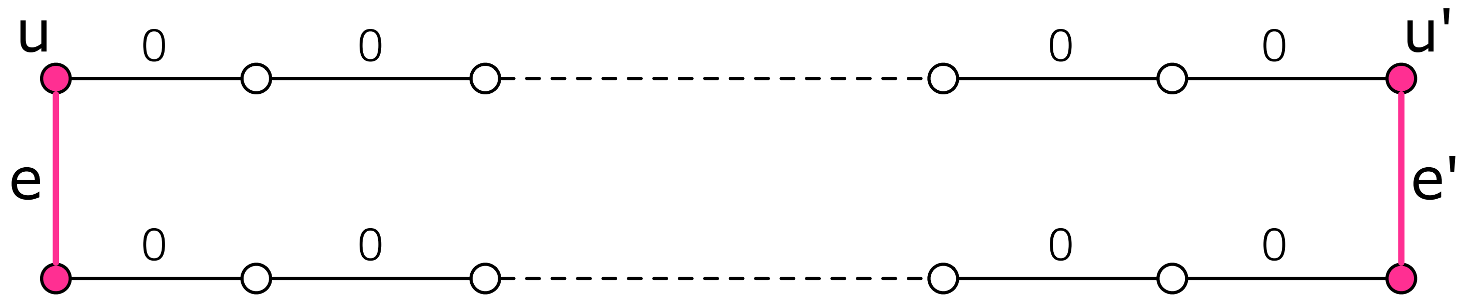
- Computing a Minimum-Weight Spanning Tree (MST) requires $\Omega(D)$ rounds in the LOCAL model.



Input of node u : ID(u), w(e) for every e ∊ E(u)

Output of node u : list of edges e ∊ E(u) belonging to MST

MST is a non-local problem



input configuration $I = (w(e), w(e'))$

diameter$(C_{2n}) = n$

Assume performing less than n rounds

Then consider the three configurations:

$I_1 = (1, 3) \quad I_2 = (3, 2) \quad I_3 = (1, 2)$

Local Problems

Problems solvable in $g(n)$ rounds in LOCAL, typically $g(n) = \text{polylog } n$ rounds, or $g(n) = O(n^{\epsilon})$ rounds, with $\epsilon < 1$.

Objective

In CONGEST, we aim at the following:

- Local problems

$$\#\text{rounds} = g(n)$$

$$\text{goal} = \text{minimizing } g$$

- Non-local problems

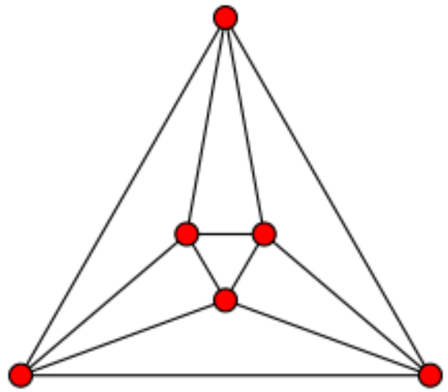
$$\#\text{rounds} = D + f(n)$$

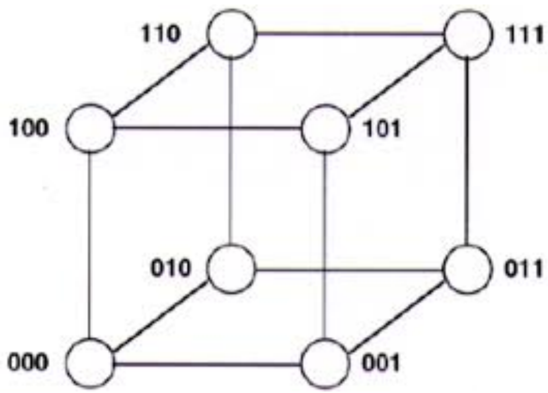
$$\text{goal} = \text{minimizing } f$$

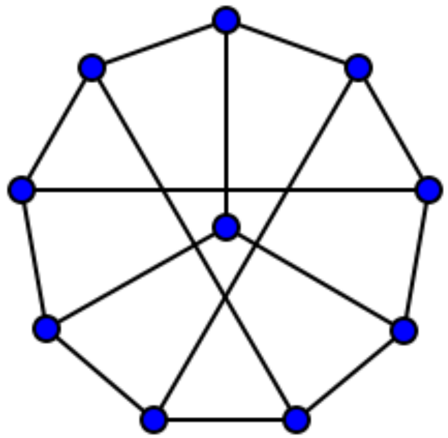
Detecting subgraphs

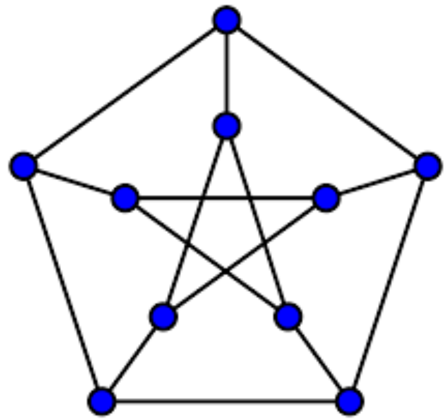
H is a subgraph of G if V(H) ⊆ V(G) and E(H) ⊆ E(G)

G is H-free if G does not contain H as a subgraph.









Distributed decision

A distributed algorithm A *decides* ϕ if and only if:

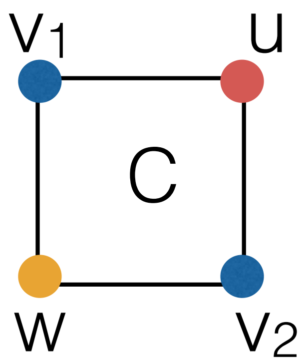
- $G \models \phi$ ⇒ all nodes output *accept*
- $G \not\models \phi$ ⇒ at least one node output *reject*

Theorem Deciding C_4-freeness can be done in $O(\sqrt{n})$ rounds.

Algorithm

Algorithm 3 C_4-detection executed by node u.

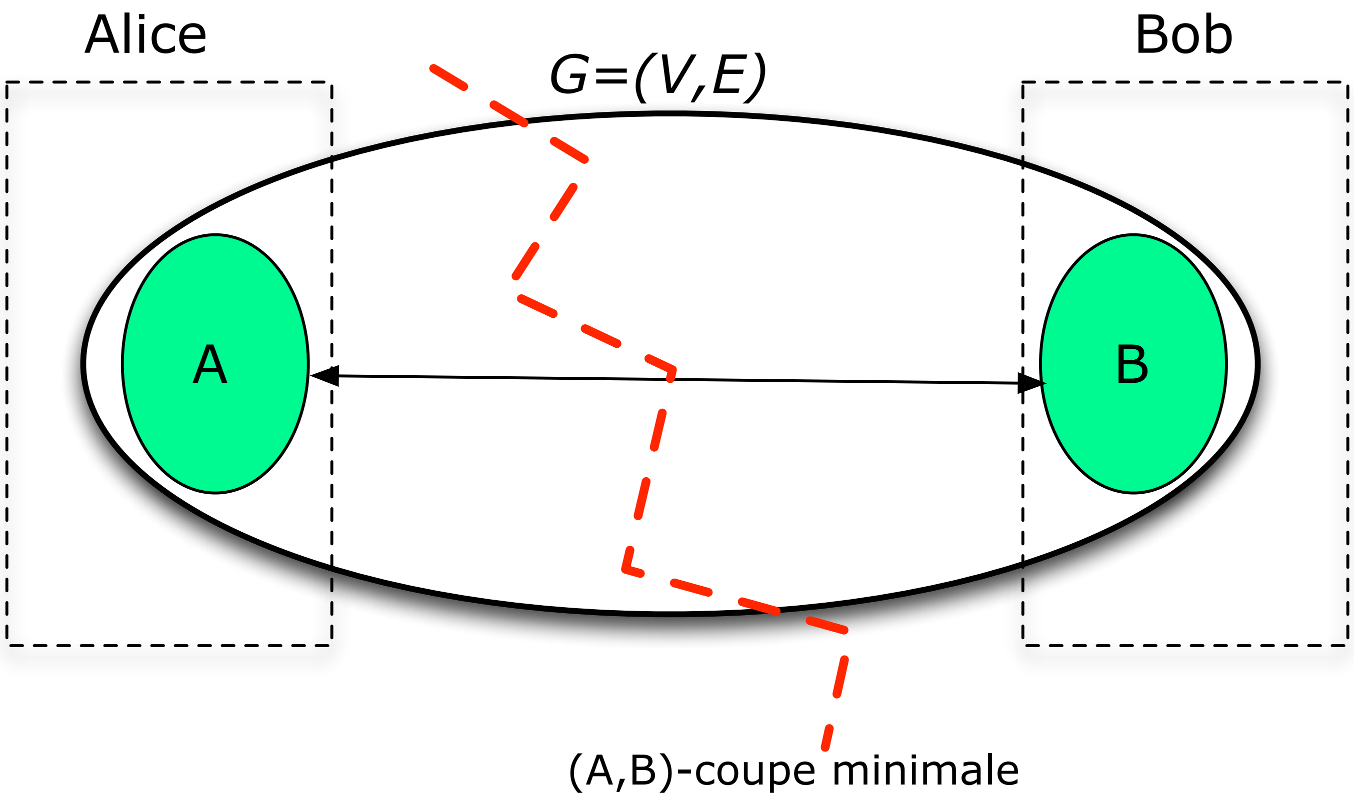
send $\text{ID}(u)$ to all neighbors, and receive $\text{ID}(v)$ from every neighbor v
send $\deg(u)$ to all neighbors, and receive $\deg(v)$ from every neighbor v
$S(u) \leftarrow \{\text{IDs of the } \min\{\sqrt{2n}, \deg(u)\} \text{ neighbors with largest degrees}\}$
send $S(u)$ to all neighbors, and receive $S(v)$ from every neighbor v
if $\sum_{v \in N(u)} \deg(v) \geq 2n + 1$ **then**
    output reject
else
    if $\exists v_1, v_2 \in N(u), \exists w \in S(v_1) \cap S(v_2) : w \neq u$ and $v_1 \neq v_2$ **then**
        output reject
    else
        output accept
    end if
end if



Case 1: there exists a 'large' node w in C
Case 2: all nodes of C are 'small'

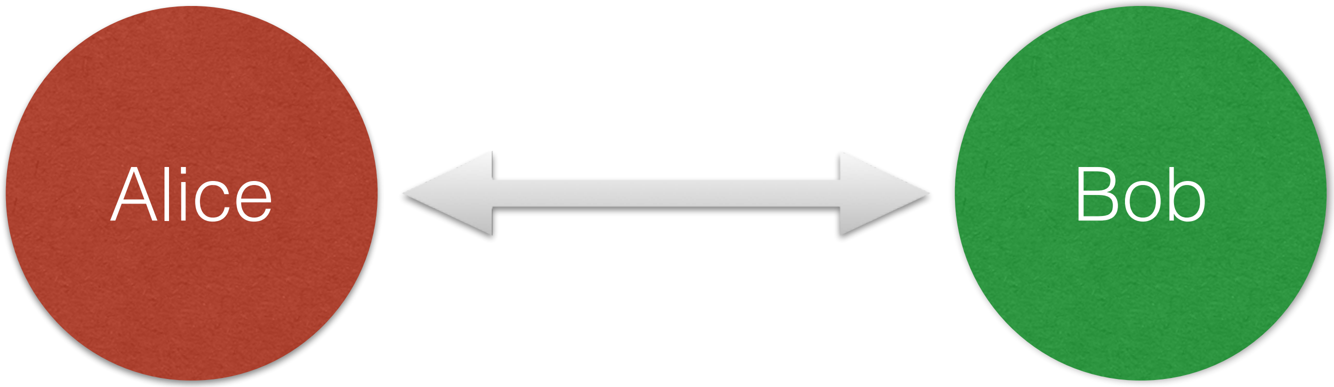
Lower bound techniques

Reduction to communication complexity



Communication complexity

$f : \{0,1\}^N \times \{0,1\}^N \rightarrow \{0,1\}$



$a \in \{0,1\}^N$ $b \in \{0,1\}^N$

Alice & Bob must compute $f(a,b)$

How many bits need to be exchanged between them?

Equality

- Alice gets $a \in \{0,1\}^N$, and Bob gets $b \in \{0,1\}^N$

$$f(a,b) = 1 \iff a = b$$

Theorem $CC(EQ) = \Omega(N)$.

Set-disjointness

- Ground set S of size N
- Alice gets $A \subseteq S$, and Bob gets $B \subseteq S$

$$f(A,B) = 1 \iff A \cap B = \emptyset$$

Theorem $CC(DISJ) = \Omega(N)$, even using randomization (i.e., even if Alice and Bob have access to sources of random bits).

Application

Deciding C_4-freeness

Theorem (Drucker, Kuhn & Oshman, 2014)
Deciding C_4-freeness required $\Omega(\sqrt{n}/\log n)$ rounds.

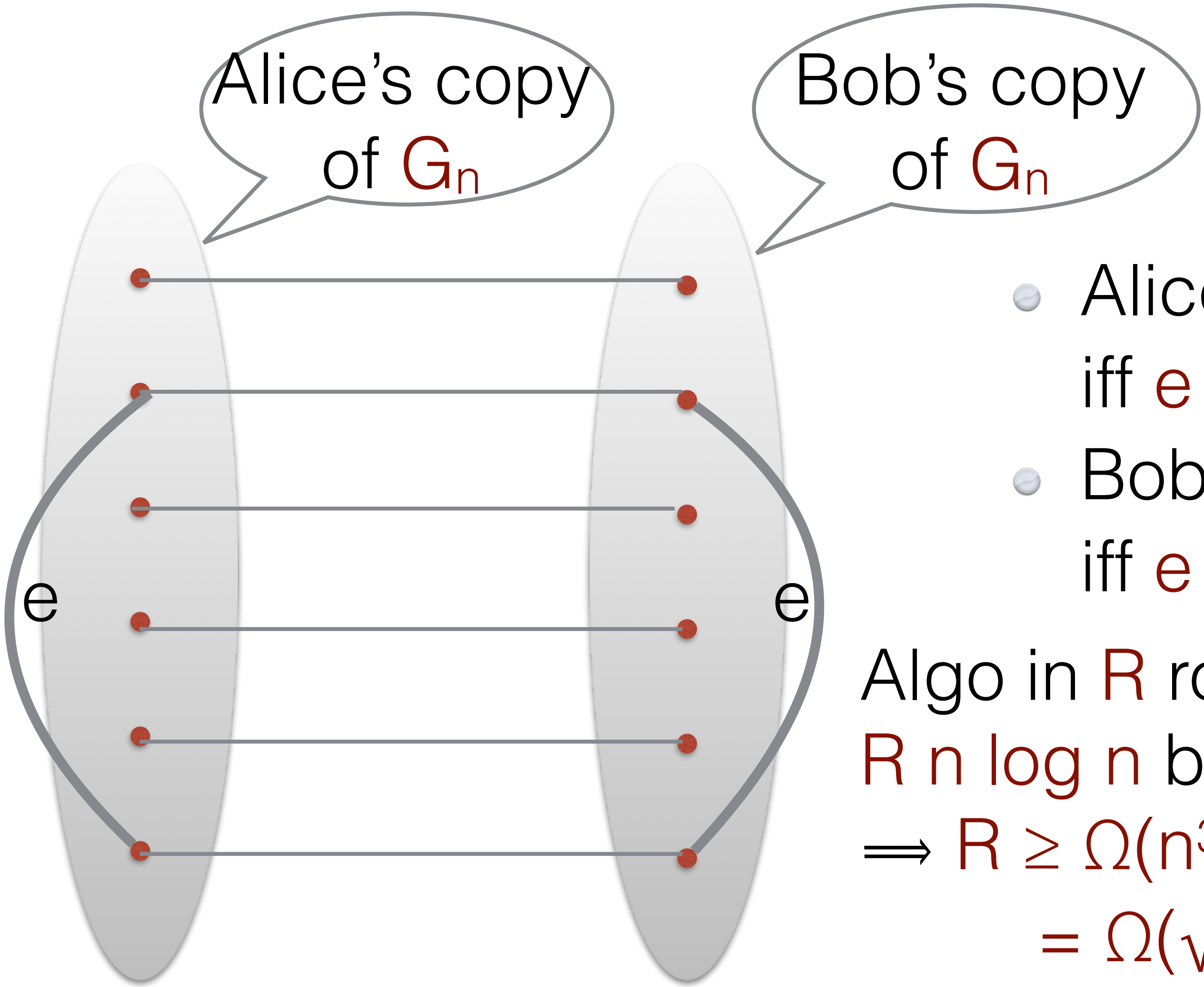
Reduction from Set-Disjointness.

We use the following result:

Lemma There is an infinite family of C_4-free graphs $\{G_n : n \geq 1\}$ such that, for every $n \geq 1$, G_n has n nodes and $m = \Omega(n^{3/2})$ edges.

Reduction

Let A and B as in set-disjointness with $N = m = \Omega(n^{3/2})$



- Alice keeps $e \in E(G_n)$ iff $e \in A$
- Bob keeps $e \in E(G_n)$ iff $e \in B$

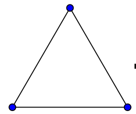
Algo in R rounds exchanges R n log n bits

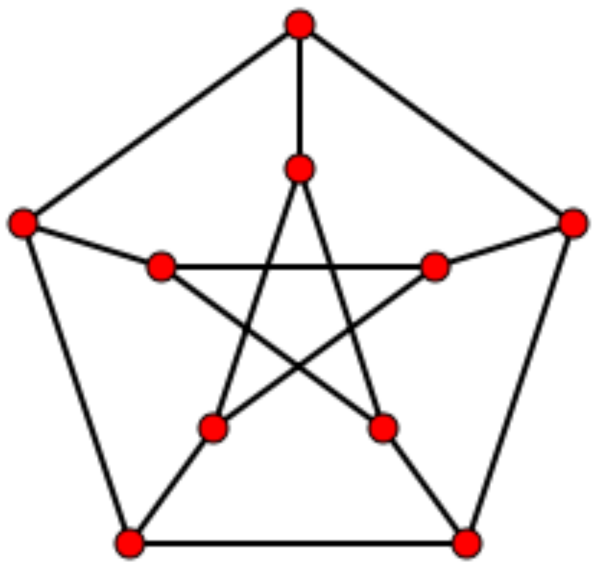
$\Longrightarrow R \geq \Omega(n^{3/2})/(n \log n)$

$= \Omega(\sqrt{n}/\log n)$

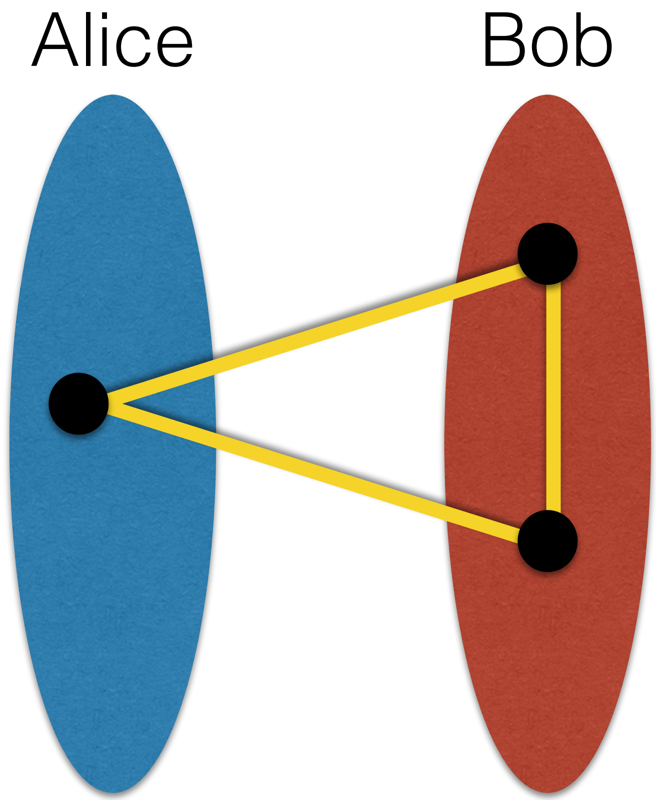
❑

Open problem

Complexity of deciding  -freeness



C_3-free graph



2-party communication complexity fails

Detecting Induced Subgraphs

A graph H is an induced subgraph of a graph G if

1. $V(H) \subseteq V(G)$

2. For every $(u, v) \in V(H) \times V(H)$, we have

$$\{u, v\} \in E(H) \iff \{u, v\} \in E(G)$$

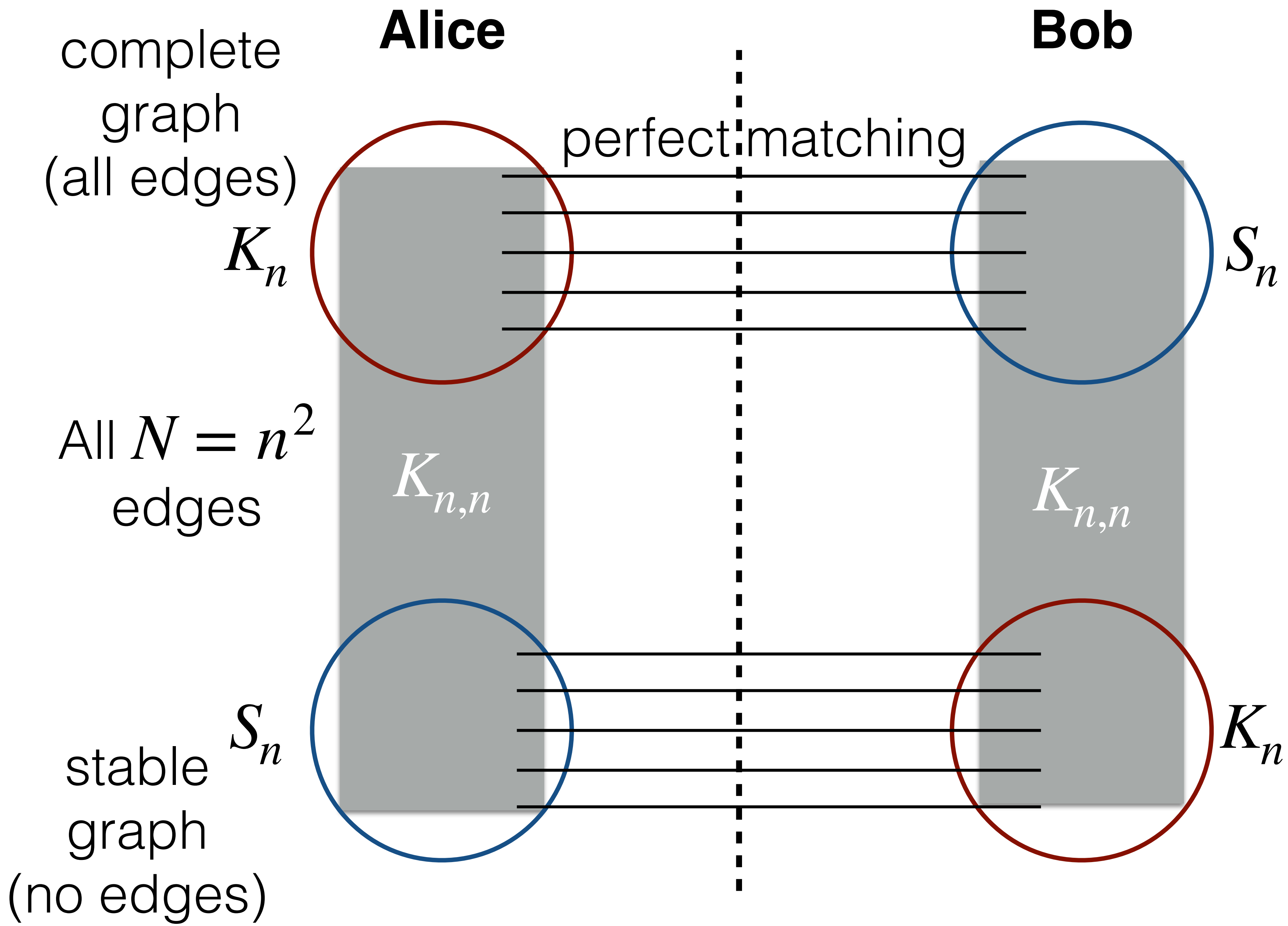
Detecting induced subgraphs is hard

Theorem Detecting induced C_4-freeness requires $\tilde{\Theta}(n)$ rounds in the CONGEST model.

Upper bound: Every node send the IDs of all its neighbors to each of its neighbors.

Each nodes send $O(n)$ IDs, each on $O(\log n)$ bits.

Lower Bound



Alice
Bob
complete graph (all edges)
K_n
perfect matching
S_n
All $N = n^2$ edges
$K_{n,n}$
$K_{n,n}$
S_n
stable graph (no edges)
K_n

Proof

Reduction from set-disjointness: Let $N = n^2$

- Alice and Bob agree on an order $e_1, e_2, \ldots, e_N$ of the edges in $K_{n,n}$
- Alice receives input $x \in \{0,1\}^N$ and keeps only edges e_i for which $x_i = 1$
- Bob receives input $y \in \{0,1\}^N$ and keeps only edges e_i for which $y_i = 1$

Claim There is an induced C_4 in G if and only if $\exists i : x_i = y_i = 1$

- Algorithme in R rounds exchanges $O(Rn \log n)$ bits between Alice and Bob.
- Since $\mathrm{CC(DISJ)} = \Omega(n^2)$, we get $R = \Omega(n / \log n)$.

Exercice

Show that deciding between $D = 2$ and $D = 3$ requires $\tilde{\Theta}(n)$ rounds in the CONGEST model.

End Lecture 5