Quantum query complexity of minor-closed graph properties

Andrew Childs
C&O and IQC

Robin Kothari
CS and IQC

University of Waterloo

arXiv:1011.1443 STACS 2011

Query complexity

Problem: Compute a function $f\colon \{0,1\}^m \to \{0,1\}$

Input $x \in \{0,1\}^m$ is given by a black box

$i \in \{1,\ldots,m\}$

$x_i \in \{0,1\}$

Minimum number of queries needed to compute f is called its *query complexity*

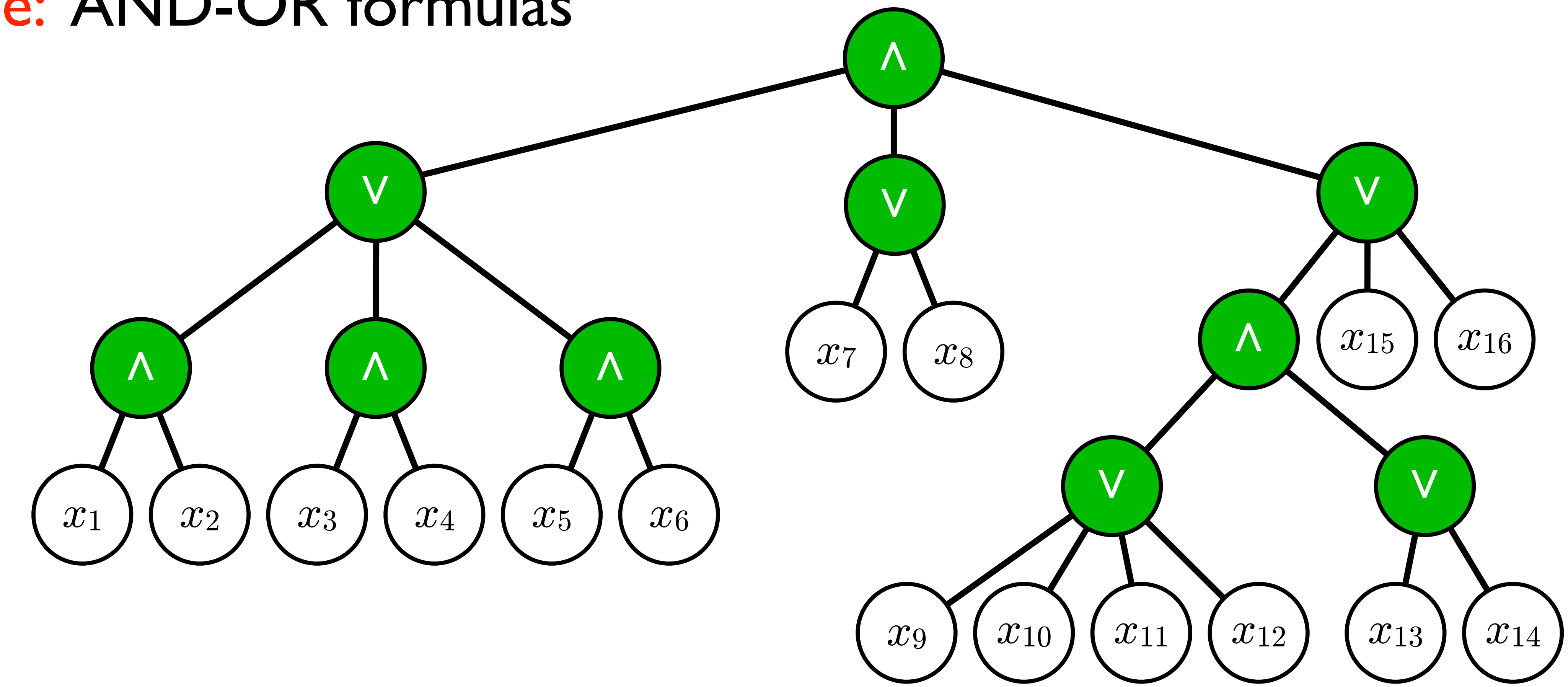
Models

- Deterministic classical algorithm: $D(f)$
- Randomized classical algorithm, bounded error: $R(f)$
- Quantum algorithm, bounded error: $Q(f)$

$$Q(f) \le R(f) \le D(f) \le m$$

Classical vs. quantum query complexity

Example: AND-OR formulas



Classical query complexity depends on the kind of formula; it can take many values between $R(f) = \Theta(m^{0.753\ldots})$ and $R(f) = \Theta(m)$

Quantum query complexity is $\tilde{\Theta}(m^{0.5})$ independent of the formula!

[Farhi, Goldstone, Gutmann 07], [ACRŠZ 07], [Reichardt 09]

Classical vs. quantum query complexity

Example: Graph properties

$m = \binom{n}{2}$

bits $x_1, \ldots, x_{\binom{n}{2}}$ represent the edges of an n-vertex graph G

f is a graph property if it doesn't depend on the vertex labels

Classical: $R(f) = \Theta(n^2)$ for most graph properties (including all the properties in this talk)

Quantum: Can have $Q(f) = \Theta(n^\alpha)$ for any $\alpha \in [1, 2]$ [BBCMW 98]

Subgraph detection

H is a subgraph of G if H can be obtained from G by
(1) deleting edges and
(2) deleting isolated vertices

Triangle problem: Does G contain a triangle as a subgraph?

Quantum query complexity: $\Omega(n),\ O(n^{1.3})$ [Magniez, Santha, Szegedy 05]

More generally, does G contain a d-vertex graph H as a subgraph?

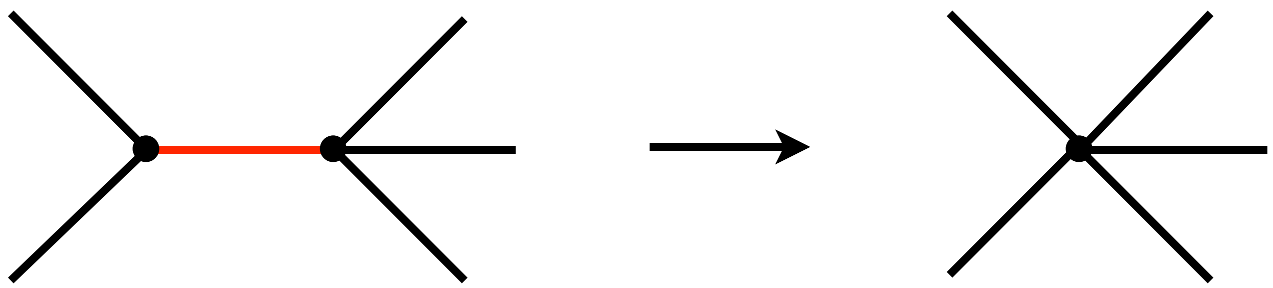
$\Omega(n),\ \tilde{O}(n^{2-2/d})$ [Magniez, Santha, Szegedy 05]

We call the property of *not* containing any of the subgraphs $\{H_1, \ldots, H_k\}$ a *forbidden subgraph property* (FSP)

Minor-closed graph properties

H is a minor of G if H can be obtained from G by
(1) deleting edges,
(2) deleting isolated vertices, and
(3) contracting edges

Edge contraction:



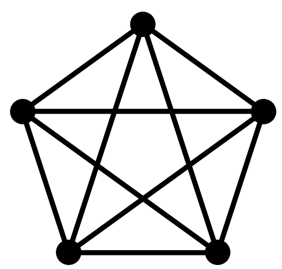
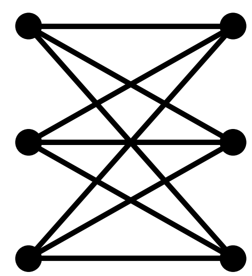
A graph property is *minor closed* if all minors of a graph with the property also have the property

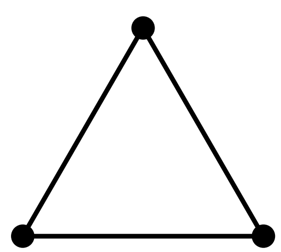
Examples: planarity, acyclicity, not containing a path of length ℓ

Graph minor theorem

Theorem: Any minor-closed property is characterized by a finite list of forbidden minors [Robertson, Seymour 83-04]

Examples

Planarity:  

Acyclicity: 

No path of length 3: 

So deciding minor-closed graph properties can be viewed as the problem of detecting forbidden minors.

Sparse graph properties

Call a graph property *sparse* if graphs $G = (V, E)$ with the property have $|E| = O(|V|)$

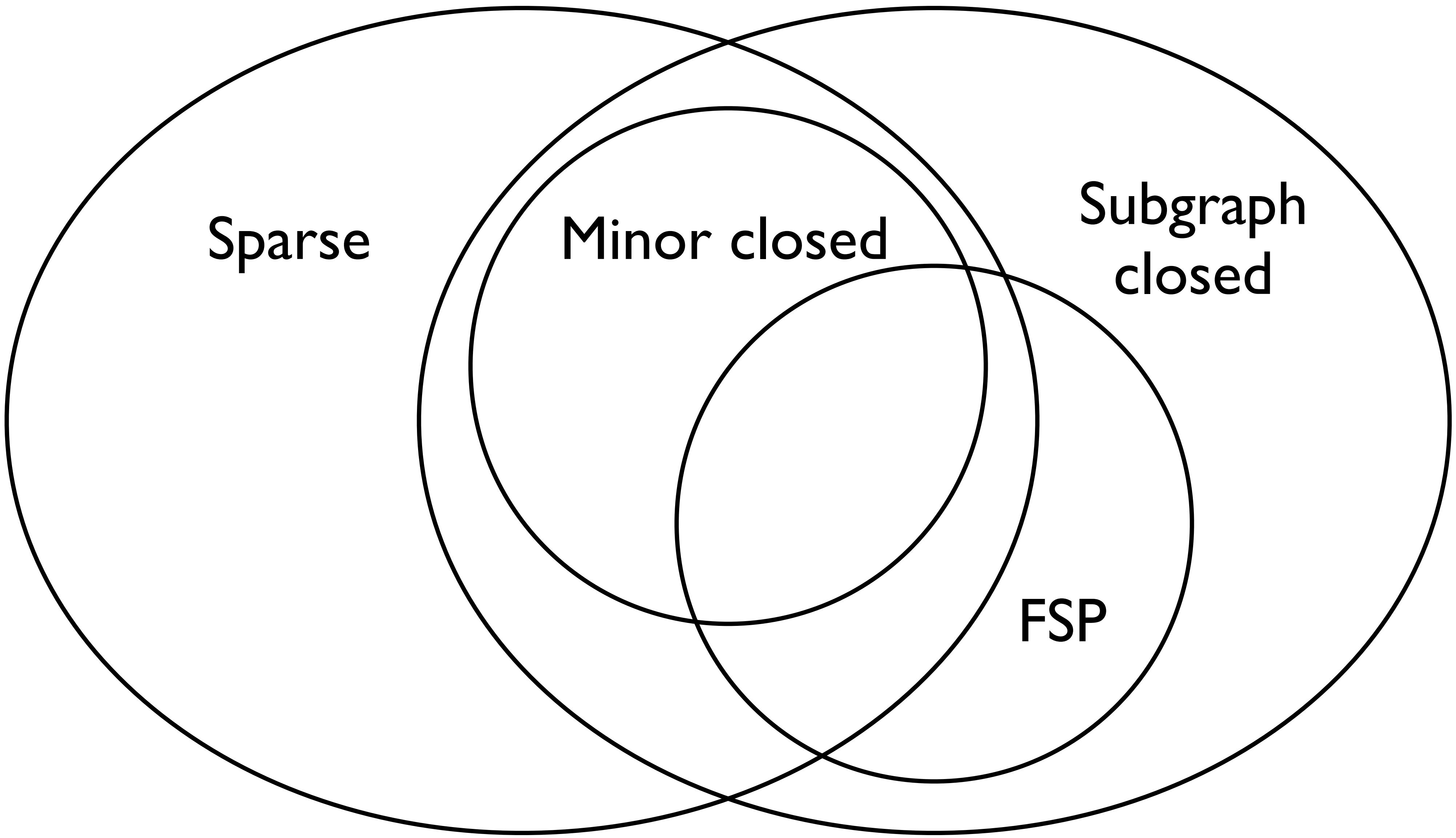
Examples

If G is acyclic (a forest) then $|E| \leq |V| - 1$

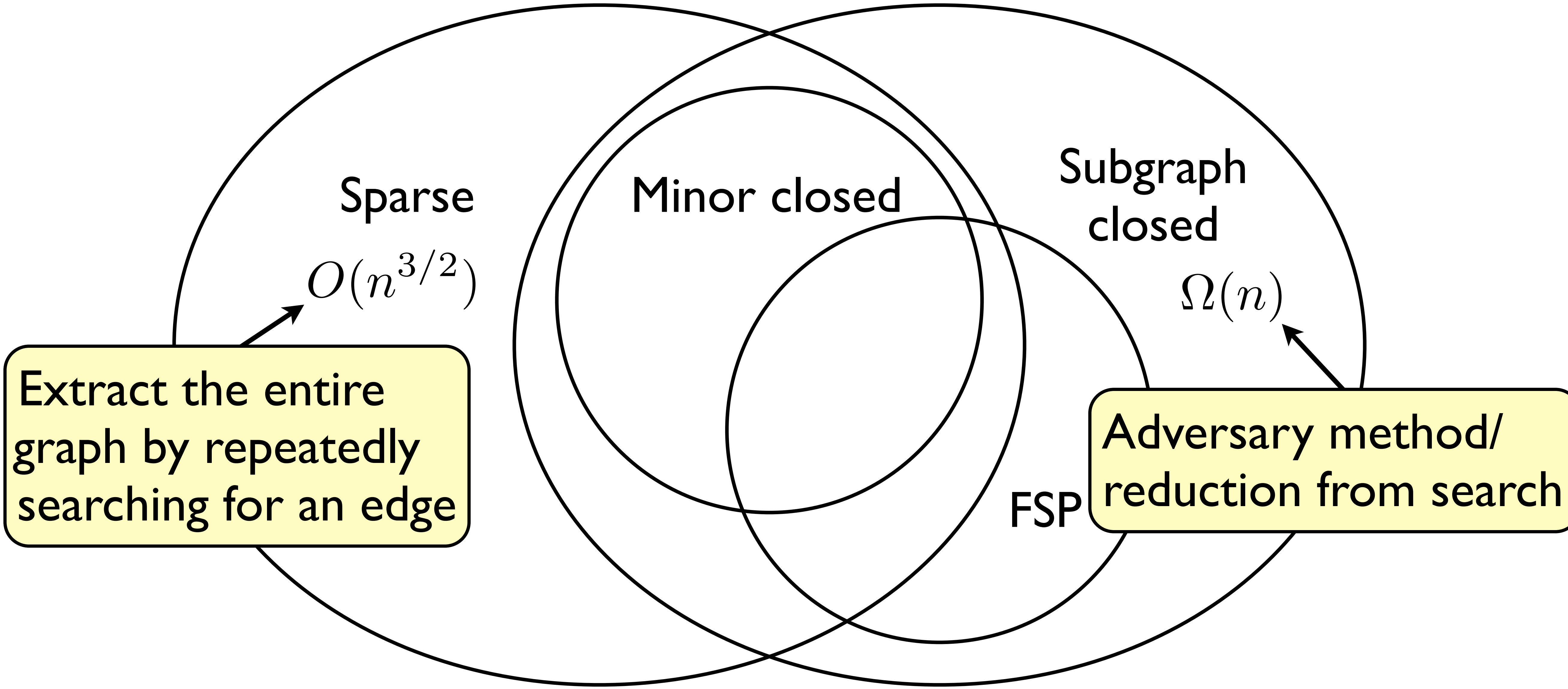
If G is planar (and $|V| > 2$) then $|E| \leq 3|V| - 6$

Theorem: Minor-closed properties are sparse [Mader 67]

Relationships between graph properties



Query complexity of minor-closed properties

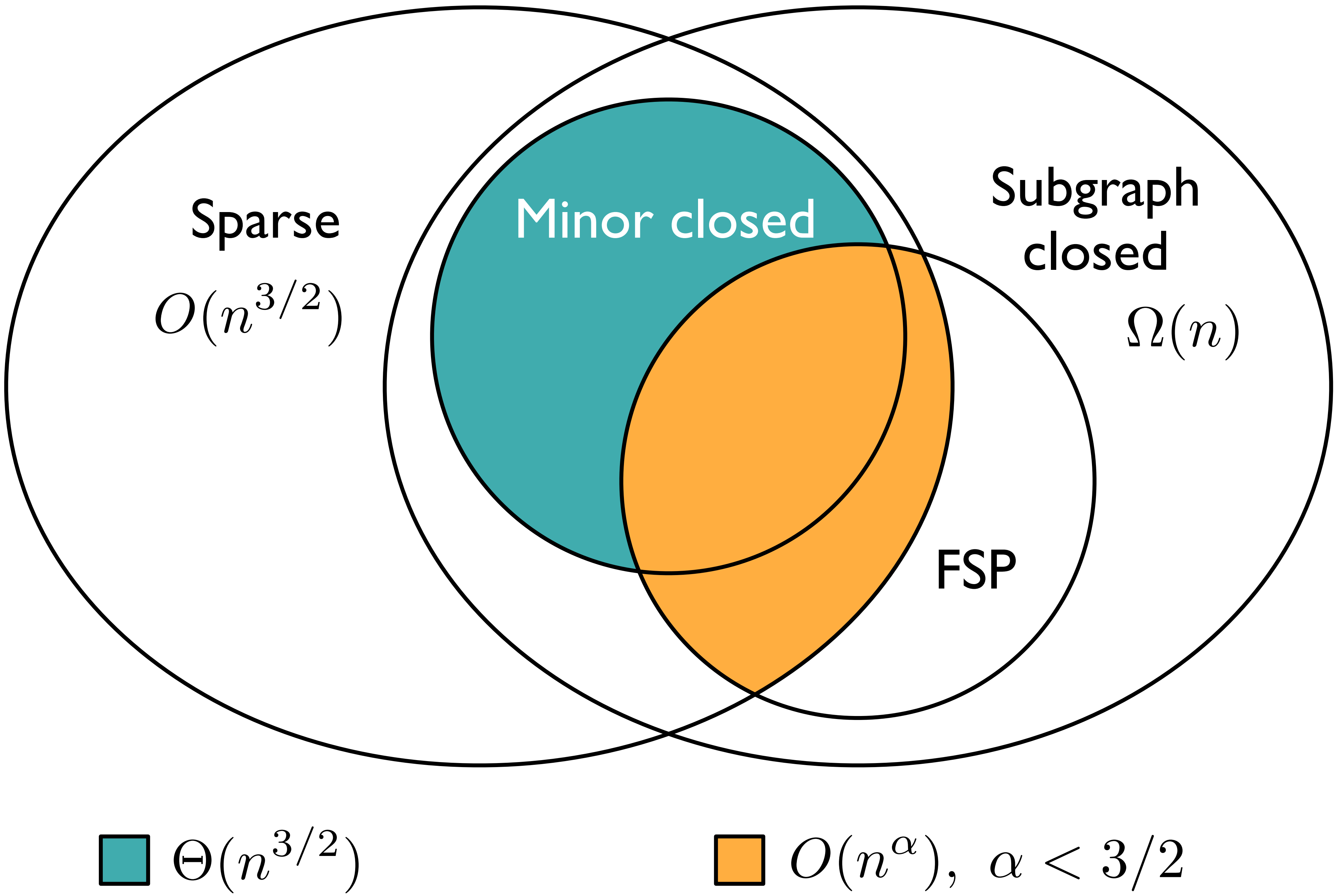


Examples

Planarity: $\Theta(n^{3/2})$ [AINNRTY 08]

Emptiness: $\Theta(n)$ [Grover 96], [BBBV 97]

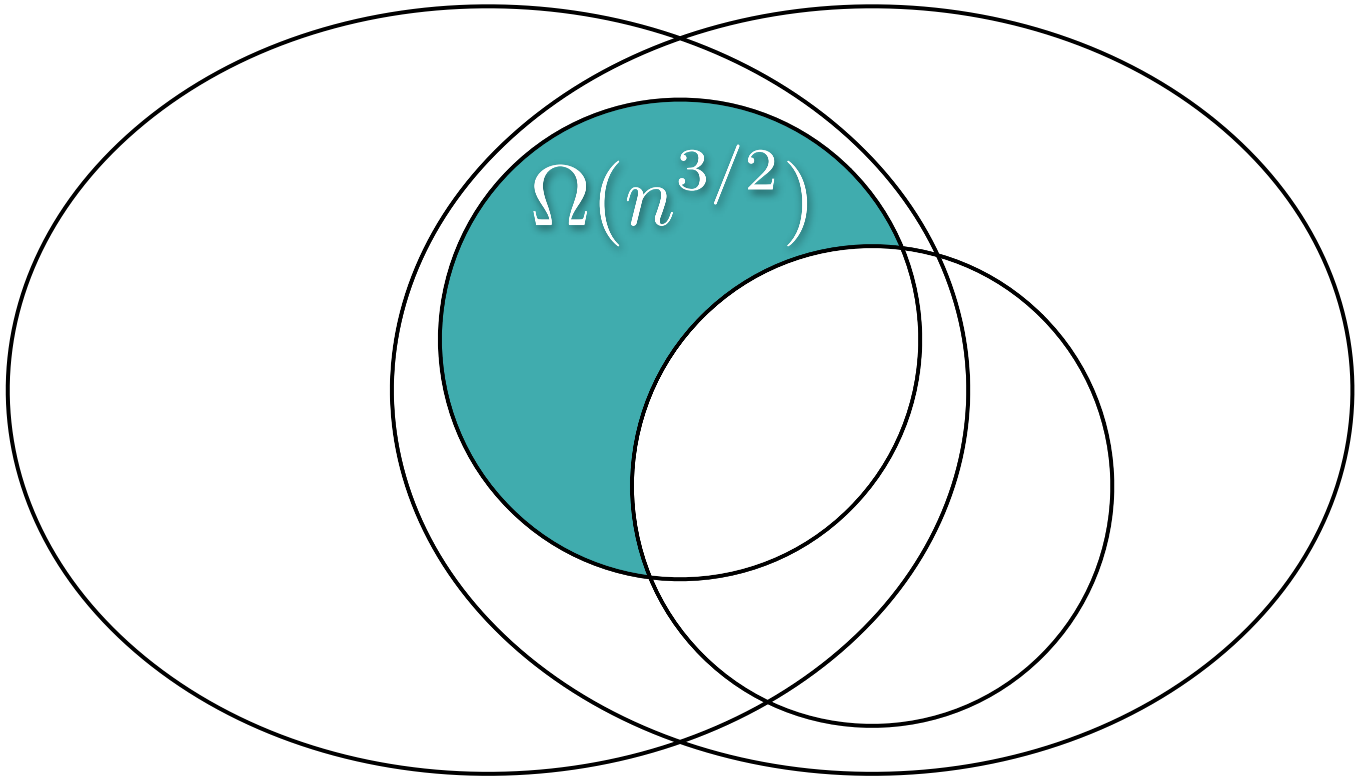
Our results



Plus improved algorithms for other subgraph-finding problems, e.g.,

bipartite H, d vertices: $\tilde{O}(n^{2-\frac{3d+2}{d(d+2)}})$ C_4: $\tilde{O}(n^{5/4})$

1. Lower bound



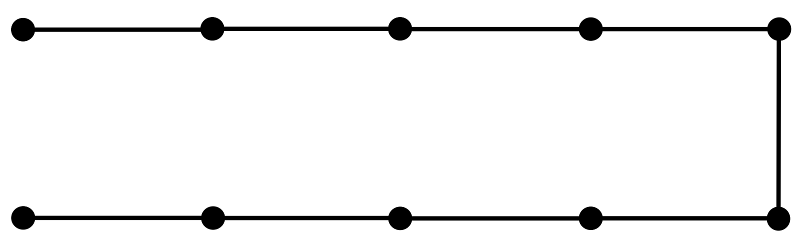
Lower bound for acyclicity

Claim: The quantum query complexity of acyclicity is $\Omega(n^{3/2})$

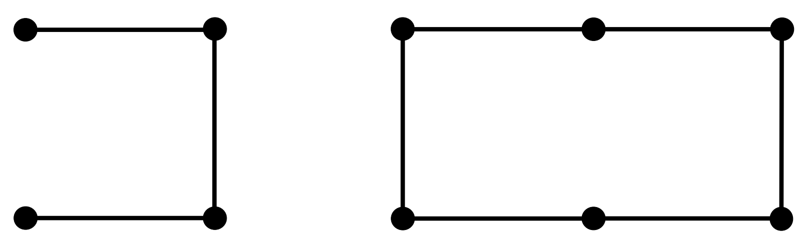
Proof uses the quantum adversary method [Ambainis 00]

Very roughly: If there are a lot of similar "yes" and "no" instances, then many queries are required to distinguish them

Hard instances:



path

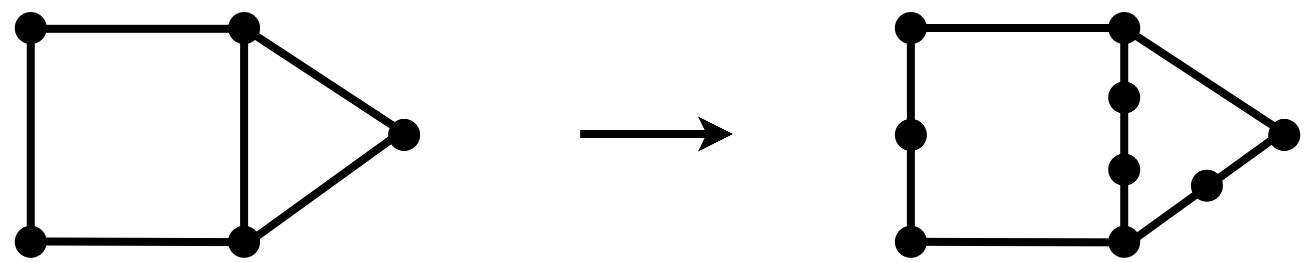


long path + long cycle

(Similar to lower bound for connectivity [DHHM 06])

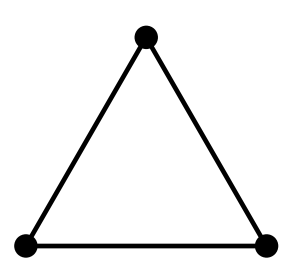
Topological minors

A subdivision of H is a graph obtained by inserting extra vertices along some of the edges:

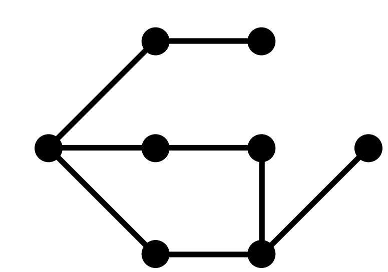


Definition: H is a topological minor of G if some subdivision of H is a subgraph of G

(Can only contract edges for which one vertex has degree 2)



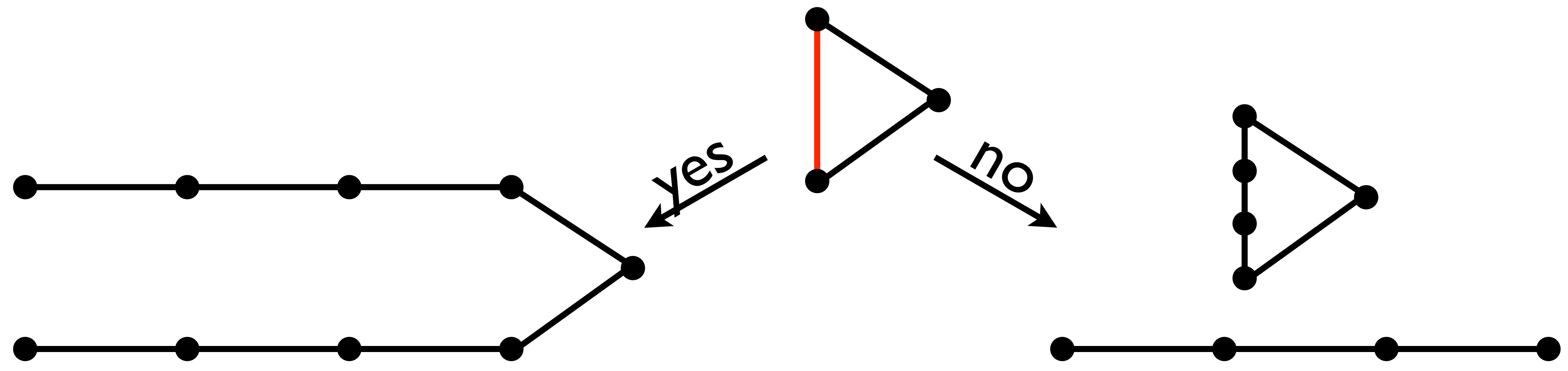
is a topological minor of



Theorem: Any minor-closed property can be characterized by a finite list of forbidden *topological* minors [Robertson, Seymour 90]

Generalization

How to construct hard instances for acyclicity:



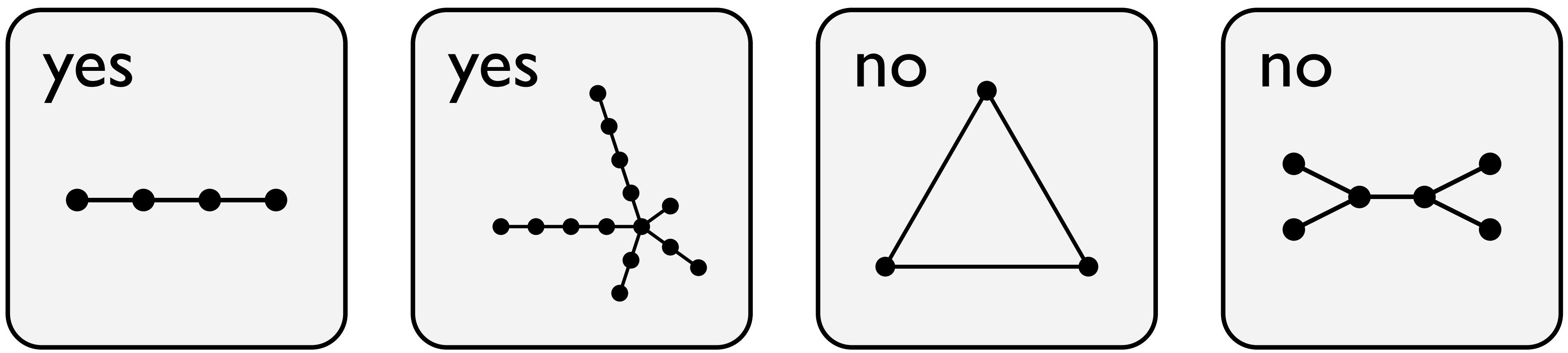
Lemma: Consider some graph property that is closed under topological minors. Let H be a graph that does not have the property. If H has an edge such that deleting the edge and adding paths of any length to its two vertices gives a graph with the property, then $\Omega(n^{3/2})$ quantum queries are needed to decide the property.

Topological minor containment

Problem: Does G contain H as a topological minor?

When can this be characterized by a forbidden subgraph?

Examples:



Lemma: Containing H as a topological minor is the same as containing it as a subgraph if and only H does not contain a cycle or a path joining two vertices of degree 3 or more

Theorem: Topological minor containment problems that are not FSP have quantum query complexity $\Omega(n^{3/2})$

Lower bound for minor-closed properties

General minor-closed properties may be characterized by many forbidden (topological) minors

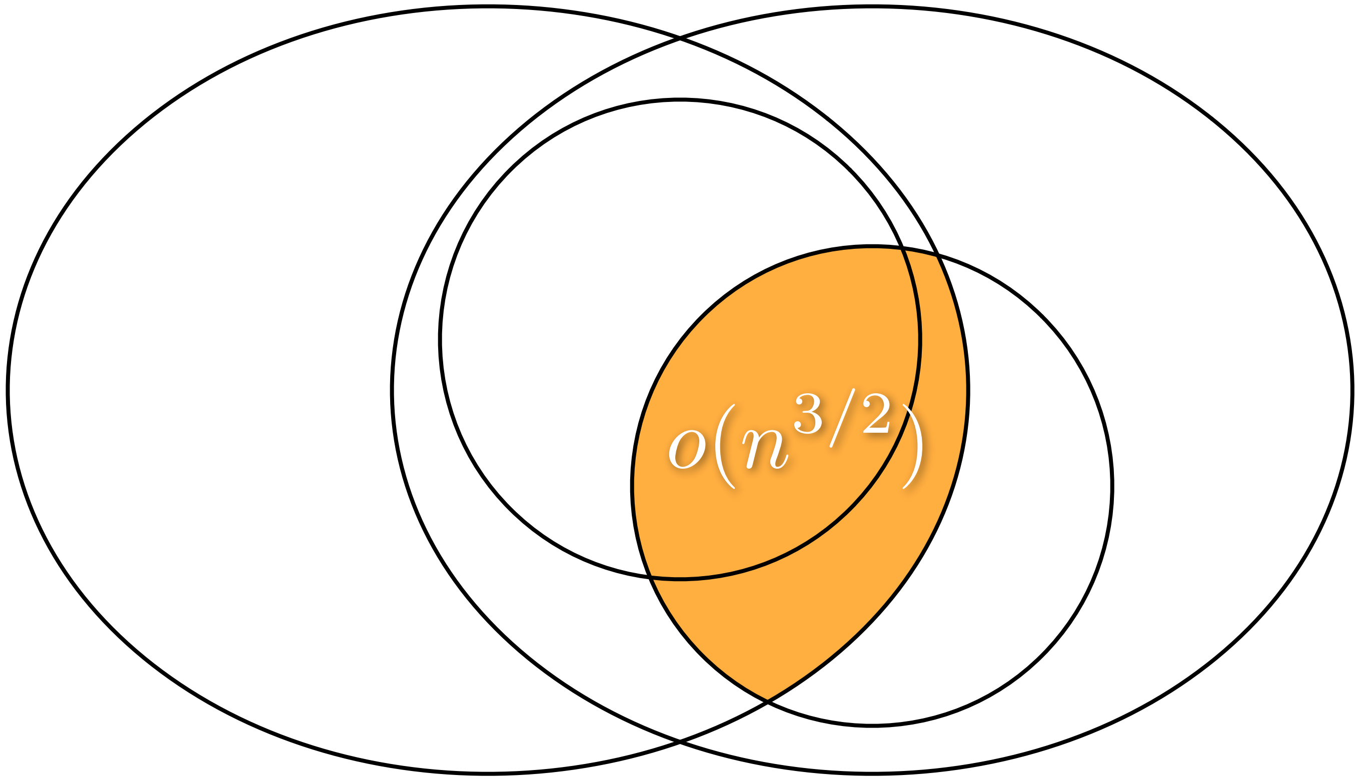
If the property is not FSP then at least one (but not necessarily all) of its forbidden topological minors cannot equivalently be viewed as a forbidden subgraph

But replacing the edge by a pair of paths may introduce one of the other forbidden topological minors!

Main idea: We identify a forbidden topological minor that is minimal in a certain sense and argue by contradiction that it must contain a suitable edge

Theorem: Minor-closed graph properties that are not FSP have quantum query complexity $\Omega(n^{3/2})$

2. Upper bounds



Random walk search

Create a Markov chain whose states represent possible solutions

Designate some of the states as "marked"

Create a new Markov chain with steps defined as follows:
If the current state is marked, stay there
Otherwise, run the original Markov chain for t steps

Costs:
- $S =$ number of queries to set up the initial state
- $U =$ number of queries to move to an adjacent state
- $C =$ number of queries to check if a state is marked

Parameters:
- $\epsilon =$ fraction of marked vertices
- $\delta =$ spectral gap of the Markov chain

Such a process can find a marked item using $S + \frac{1}{\epsilon}(\frac{1}{\delta}U + C)$ queries

Quantum walk search

Idea: Replace the classical Markov chain with a quantum walk

Theorem: The quantum walk can be used to detect a marked state with $S + \frac{1}{\sqrt{\epsilon}}(\frac{1}{\sqrt{\delta}}U + C)$ queries [Ambainis 04], [Szegedy 04], [Magniez, Nayak, Roland, Santha 07]

Quantum walk provides a powerful framework:

- Grover's algorithm [Grover 96]: search on the complete graph
- Spatial search [CG 04], [AKR 05]: search on a d-dimensional lattice
- Element distinctness [Ambainis 04],
  triangle finding [MSS 05], etc.: search on a Johnson graph

Quantum counting

Problem: Given a black box for an input $x \in \{0,1\}^N$, how many bits of x are 1?

Theorem: We can estimate the number of 1s to within a constant multiplicative factor using $O(\sqrt{N})$ quantum queries [BHMT 02]

Applications to black-box graphs:

- Detect graphs with $\Omega(n)$ edges using $O(n)$ queries (so when deciding sparse graph properties, we can assume that the input graph is sparse)
- Approximate the degree of a vertex using $O(\sqrt{n})$ queries
- Approximately count the number of vertices of degree approximately q using $\tilde{O}(n)$ queries

Detecting subgraphs of sparse G: Setup

To take advantage of sparsity, we treat vertices differently depending on their approximate degree

Assume vertices $1, \ldots, d$ of H have approximate degree $q_1, \ldots, q_d$ in G

There are $O(\log n)$ ranges for the degrees, so we can iterate over the possible q_is with overhead only $\mathrm{poly}(\log n)$

Approximately count the number of vertices of degree near $q_1, \ldots, q_d$; call the number of vertices $t_1, \ldots, t_d$

Search space:

- Subsets of size $k_1, \ldots, k_d$ (to be optimized) of the vertices of G of approximate degree $q_1, \ldots, q_d$
- For each vertex, store its complete neighbor list

Detecting subgraphs of sparse G: The walk

Start from a uniform superposition over $k_1, ..., k_d$-tuples of vertices of degree near $q_1, ..., q_d$, along with their neighbor lists

Quantize this Markov chain:

For $i = 1, ..., d$
Repeat α_i times
Replace one of the vertices of degree near q_i (randomly selected among the k_i possibilities) with another vertex of degree near q_i (randomly selected among the t_i possibilites)
End repeat
End for

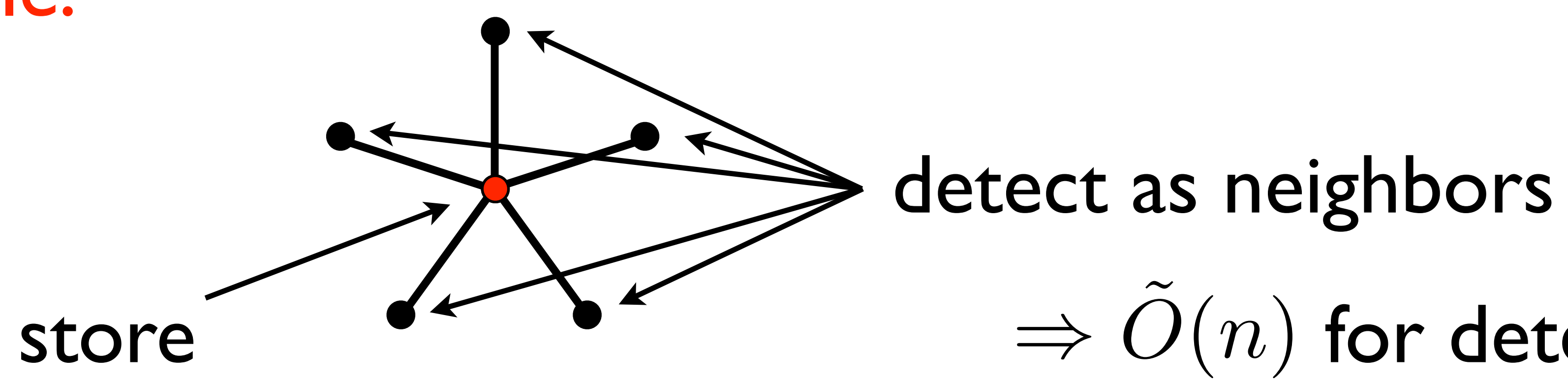
Optimize the choices of $k_1, ..., k_d$ and $\alpha_1, ..., \alpha_d$

Optimized running time is $\tilde{O}\left(n^{\frac{3}{2} - \frac{1}{d+1}}\right)$

Improvements

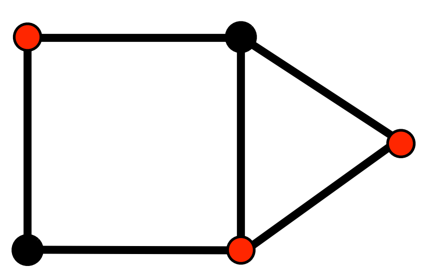
Since we store the complete neighbor list for each vertex, we don't need to explicitly search for every vertex of H

Example:



$\Rightarrow \tilde{O}(n)$ for detecting stars

In general, it suffices to look for a *vertex cover* of H, a subset of vertices such that every edge is incident to at least one of them:



Query complexity $\tilde{O}\left(n^{\frac{3}{2}-\frac{1}{\mathrm{vc}(H)+1}}\right)$

In fact we can also delete degree-one vertices of H (that do not belong to an isolated edge)

Relaxing sparsity: Bipartite subgraphs

The bound $|E| = O(|V|)$ is not essential; we can do something similar given any bound $|E| \leq \bar{m}$

Query complexity $\tilde{O}\big(\sqrt{\bar{m}}\, n^{1-\frac{1}{\mathrm{vc}(H)+1}}\big)$

Theorem: If G does not contain $K_{s,t}$ as a subgraph, where $1 \leq s \leq t$, then $|E(G)| \leq c_{s,t}|V(G)|^{2-1/s}$ [Kővári, Sós, Turán 54]

$\Rightarrow$ If H is a d-vertex bipartite graph, we can detect H using $\tilde{O}(n^{2-\frac{3d+2}{d(d+2)}})$ queries

Theorem: If G does not contain $C_{2\ell}$ as a subgraph, then $|E(G)| \leq 100\ell\,|V(G)|^{1+1/\ell}$ [Bondy, Simonovits 74]

$\Rightarrow$ We can detect $C_{2\ell}$ using $\tilde{O}\big(n^{\frac{3}{2}-\frac{\ell-1}{2\ell(\ell+1)}}\big)$ queries

Relaxing sparsity: C_4

Sometimes we can do even better with a nontrivial checking step

Theorem: The quantum query complexity of deciding whether G contains C_4 as a subgraph is $\tilde{O}(n^{1.25})$

Contrast with C_3: Best known upper bound is $O(n^{1.3})$ [MSS 05]!

Why?

- Graphs with $\Omega(n^{3/2})$ edges must contain a 4-cycle
- Graphs with $\Omega(n^2)$ edges might or might not contain a triangle

Open problems

Close the gap between $\Omega(n)$ and $o(n^{3/2})$ for minor-closed FSPs

Algorithms:

- Use other properties of minor-closed graphs (e.g., bounded degeneracy)
- Focus on particular cases (e.g., paths)

Lower bounds:

- Cannot do better using the positive adversary method (certificate complexity barrier)
- Major challenge: Prove a superlinear lower bound for *any* FSP (Might be easier for non-sparse properties, but still seems hard in that case, e.g., for triangle)