Lower Bounds for Subgraph Detection in the CONGEST Model

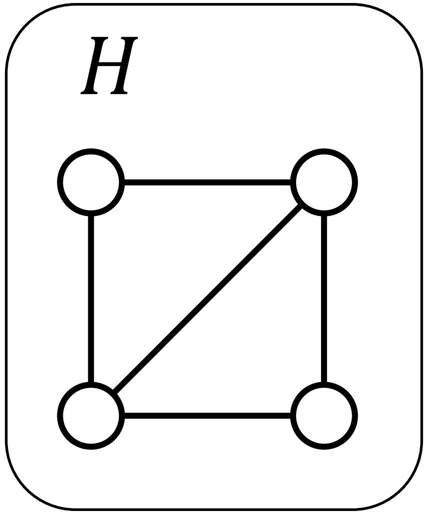
TZLIL GONEN

JOINT WORK WITH ROTEM OSHMAN

Tel Aviv University

Introduction: Subgraph Freeness

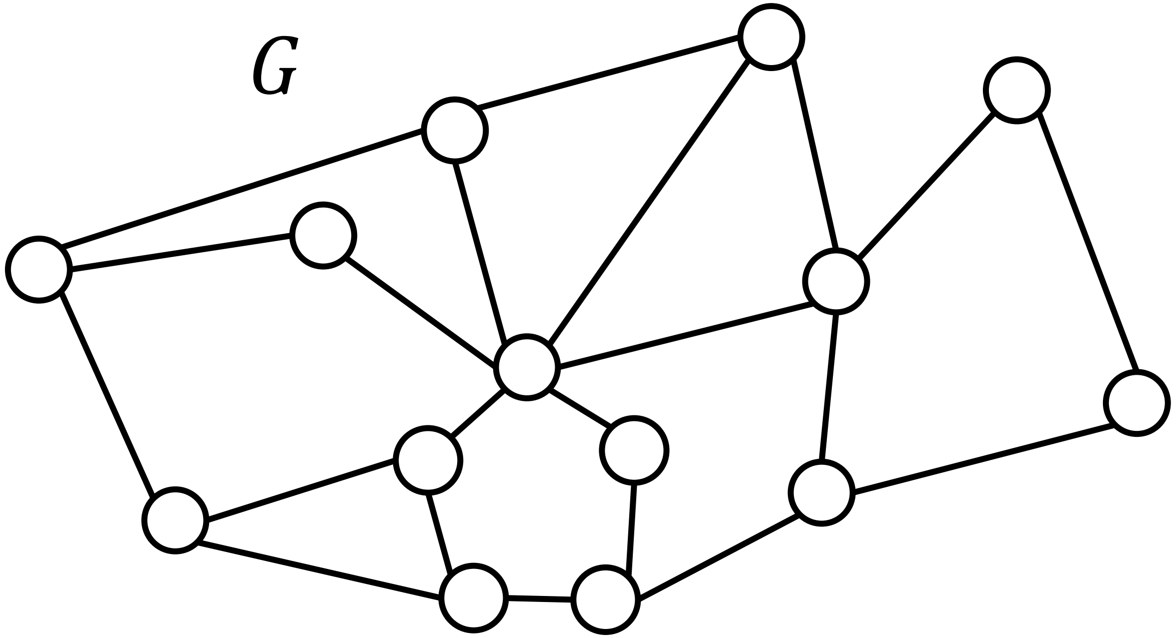
Introduction: Subgraph Freeness



<u>Input:</u>

1. Network graph G
2. Fixed graph H

<u>Output:</u> does G contain H as a subgraph?

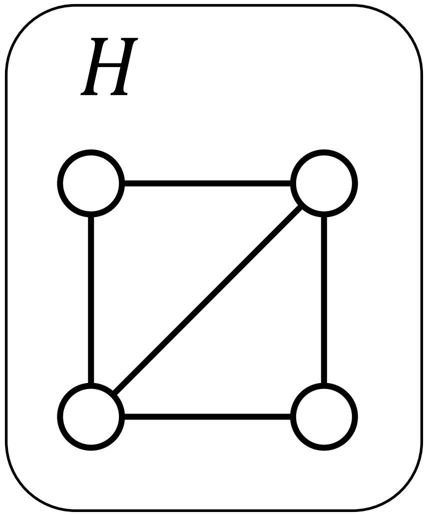


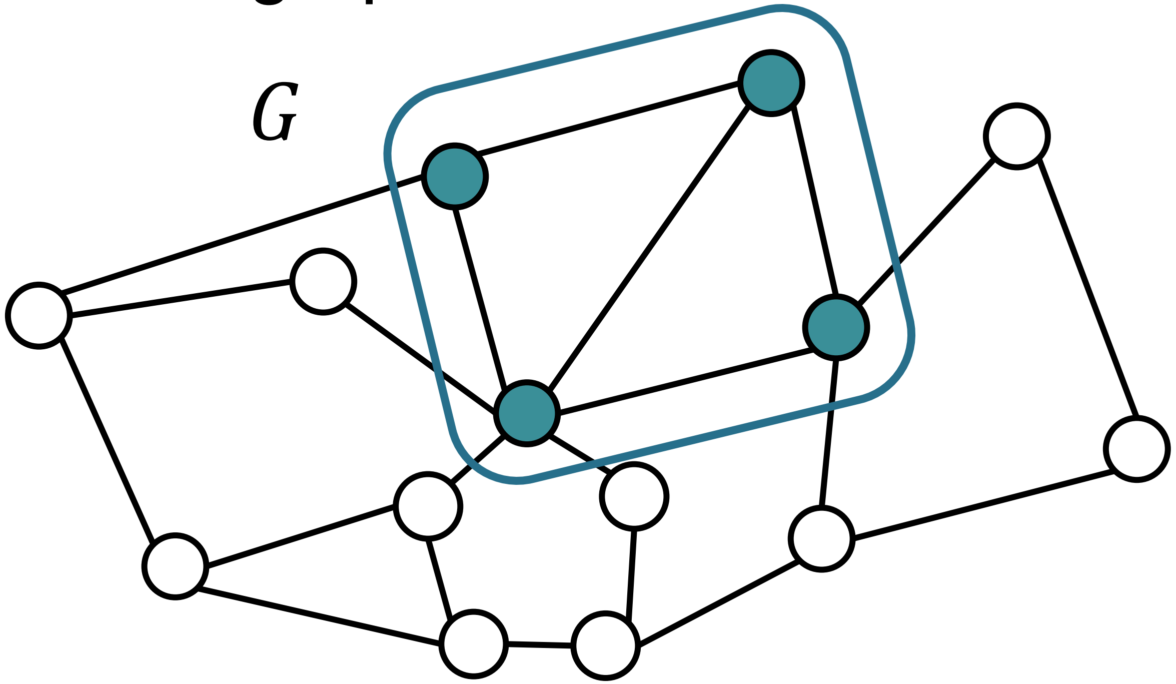
Introduction: Subgraph Freeness

Input:

1. Network graph G
2. Fixed graph H

Output: does G contain H as a subgraph?





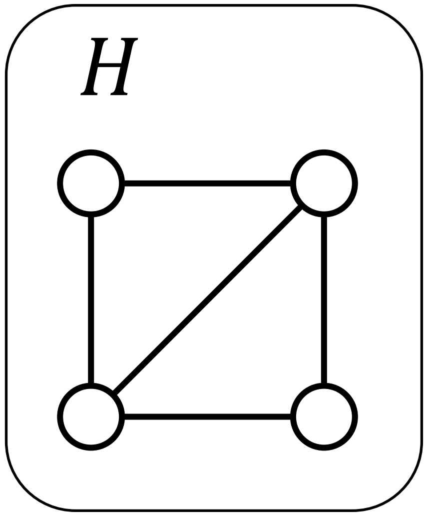
Introduction: Subgraph Freeness

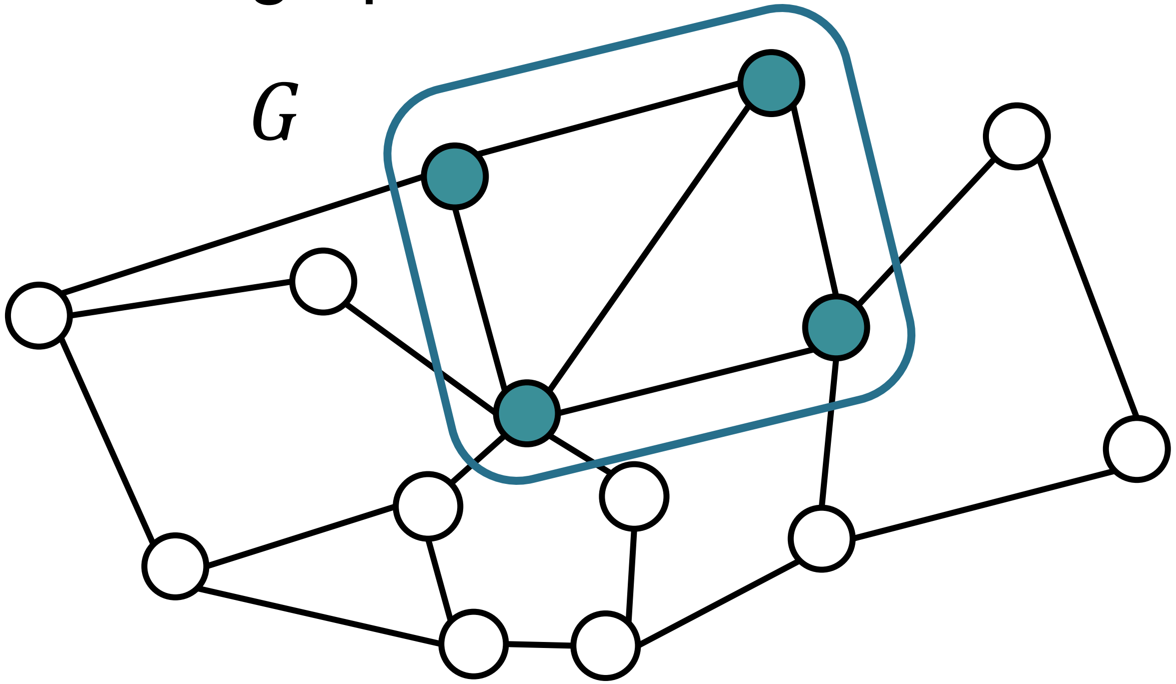
<u>Input:</u>

1. Network graph G
2. Fixed graph H

<u>Output:</u> does G contain H as a subgraph?

Model: CONGEST





What We Know So Far…

What We Know So Far…

- Lower bounds for cycles

What We Know So Far…

- Lower bounds for cycles
  - Even cycles: $\tilde{\Omega}(\sqrt{n})$ rounds [Korhonen, Rybicki '17]
  - Odd cycles: $\tilde{\Omega}(n)$ rounds [Drucker, Kuhn, Oshman '14]

What We Know So Far…

- Lower bounds for cycles
  - Even cycles: $\tilde{\Omega}(\sqrt{n})$ rounds [Korhonen, Rybicki ’17]
  - Odd cycles: $\tilde{\Omega}(n)$ rounds [Drucker, Kuhn, Oshman ’14]
- Trees require $\Theta(1)$ rounds [Fraigniaud, Montealegre, Olivetti, Rapaport, Todinca 17’] and [Fischer, Gonen, Oshman ’17]

What We Know So Far…

- Lower bounds for cycles
  - Even cycles: $\widetilde{\Omega}(\sqrt{n})$ rounds [Korhonen, Rybicki ’17]
  - Odd cycles: $\widetilde{\Omega}(n)$ rounds [Drucker, Kuhn, Oshman ’14]
- Trees require $\Theta(1)$ rounds [Fraigniaud, Montealegre, Olivetti, Rapaport, Todinca 17’] and [Fischer, Gonen, Oshman ’17]
- Some graphs of size k require $\widetilde{\Omega}\left(n^{2-1/k}\right)$ rounds [Fischer, Gonen, Oshman ’17]

What We Know So Far…

- Lower bounds for cycles
  - Even cycles: $\tilde{\Omega}(\sqrt{n})$ rounds [Korhonen, Rybicki ’17]
  - Odd cycles: $\tilde{\Omega}(n)$ rounds [Drucker, Kuhn, Oshman ’14]
- Trees require $\Theta(1)$ rounds [Fraigniaud, Montealegre, Olivetti, Rapaport, Todinca 17’] and [Fischer, Gonen, Oshman ’17]
- Some graphs of size k require $\tilde{\Omega}\left(n^{2-1/k}\right)$ rounds [Fischer, Gonen, Oshman ’17]
- Our question: what else?

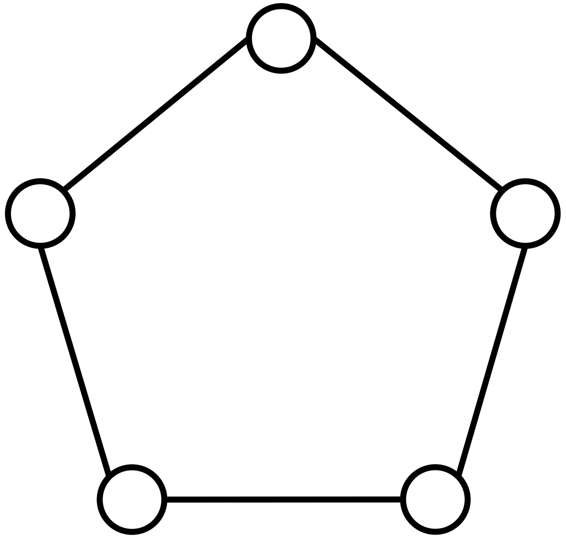
Reduction #1: Vertex Replacement

Reduction #1: Vertex Replacement

- Take any 2-connected graph for which a lower bound is known…

Reduction #1: Vertex Replacement

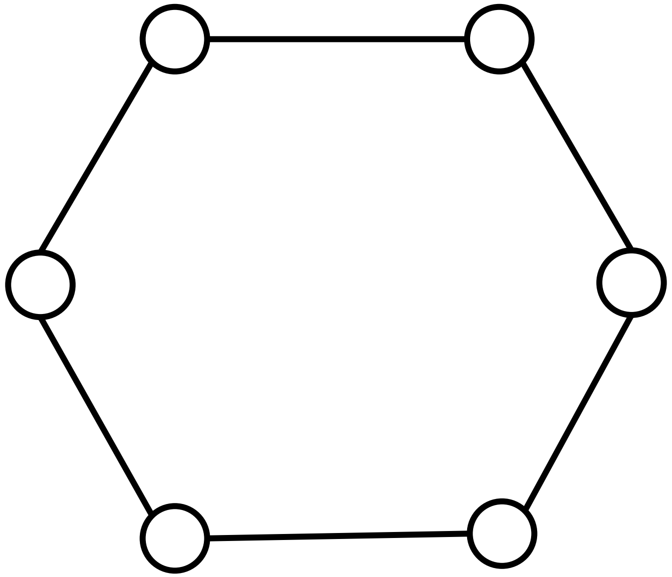
- Take any 2-connected graph for which a lower bound is known…
- Like this graph:



$\tilde{\Omega}(n)$ rounds
[Drucker, Kuhn, Oshman '14]

Reduction #1: Vertex Replacement

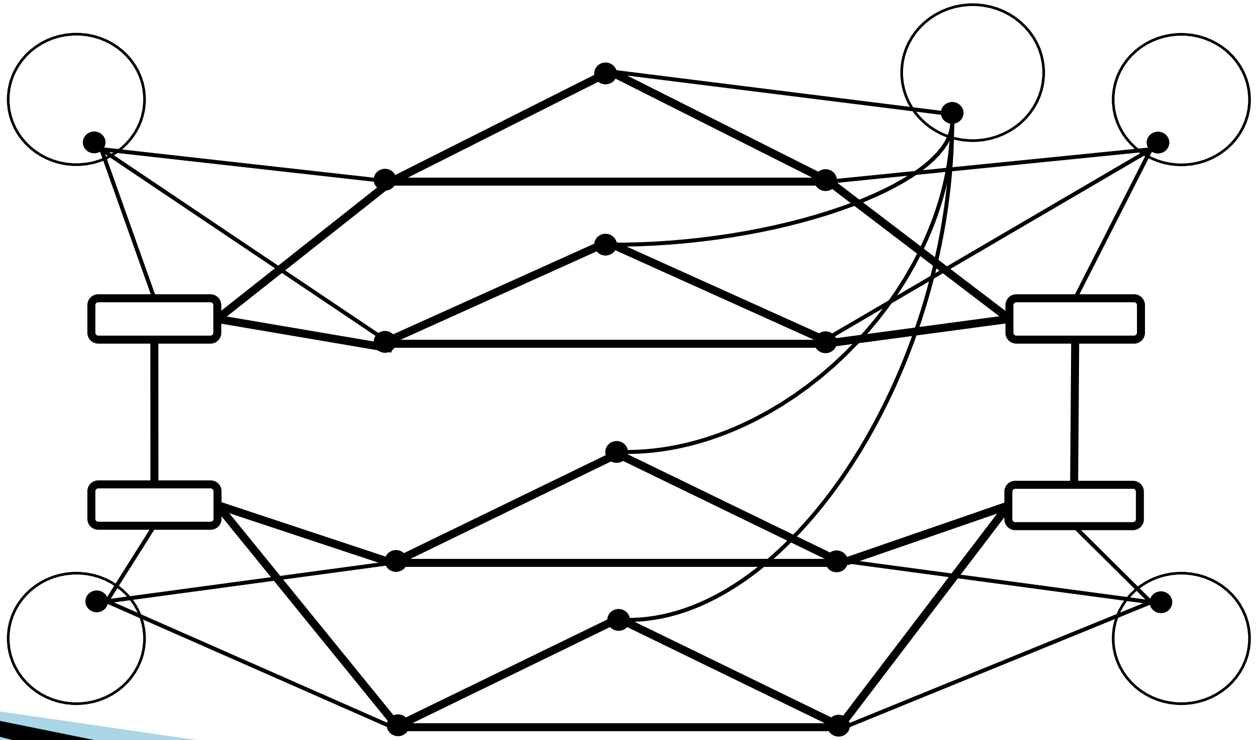
- Take any 2-connected graph for which a lower bound is known…
- … or this graph:



$\tilde{\Omega}(\sqrt{n})$ rounds
[Korhonen, Rybicki '17]

Reduction #1: Vertex Replacement

- Take any 2-connected graph for which a lower bound is known…
- … or even this graph:

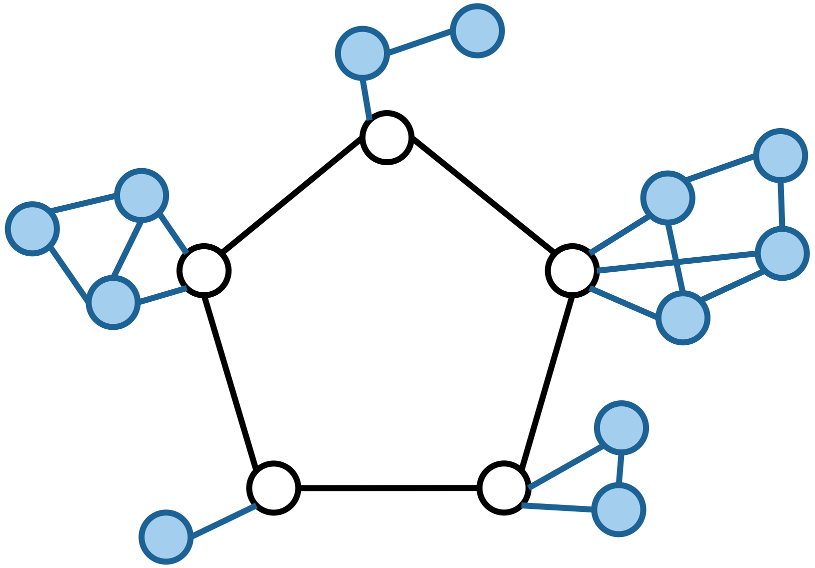


$\tilde{\Omega}\left(n^{2-1/k}\right)$ rounds for size $\Theta(k)$ graph

[Fischer, Gonen, Oshman '17]

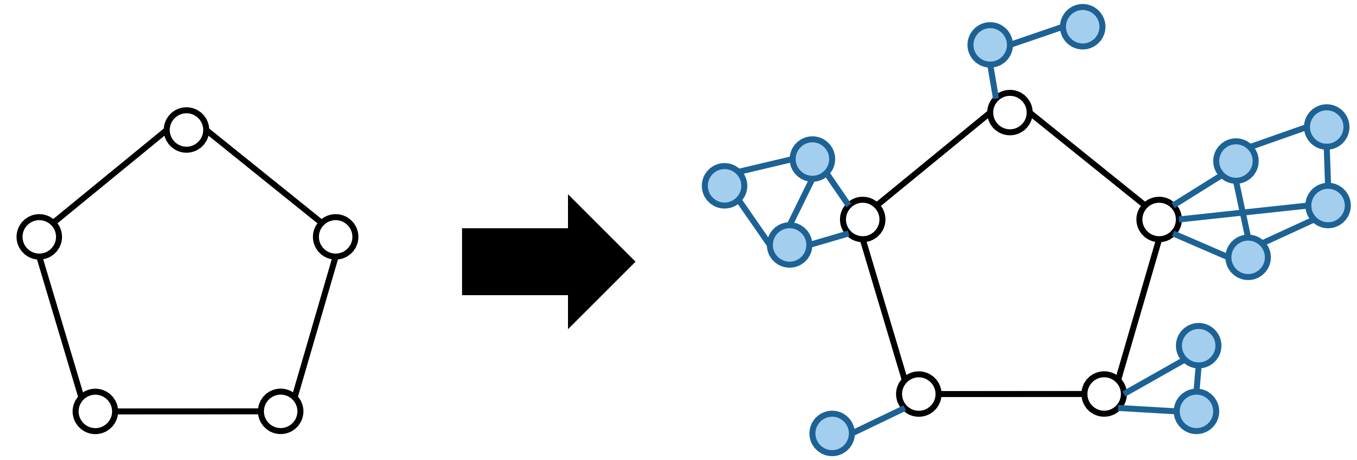
Reduction #1: Vertex Replacement

- Take any 2-connected graph for which a lower bound is known…
- Attach arbitrary connected graph to each vertex…



Reduction #1: Vertex Replacement

- Take any 2-connected graph for which a lower bound is known…
- Attach arbitrary connected graph to each vertex…
- Obtain nearly the same lower bound

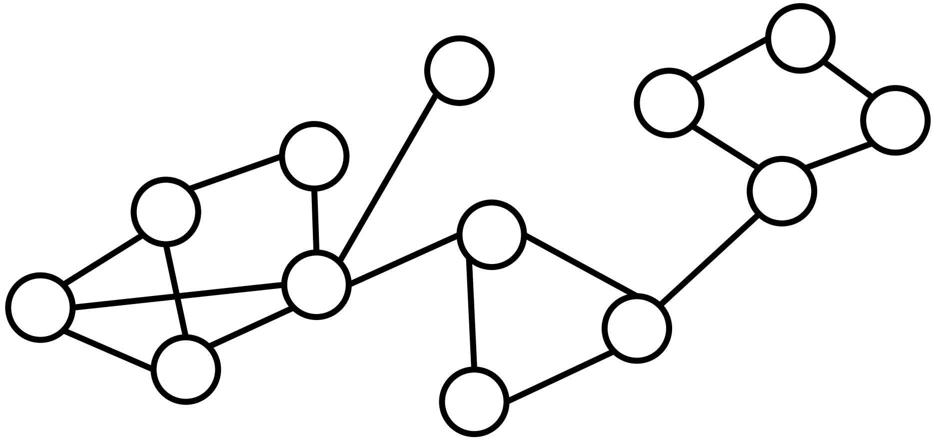


$\tilde{\Omega}(n)$ rounds $\tilde{\Omega}(n)$ rounds

Corollary from Reduction #1

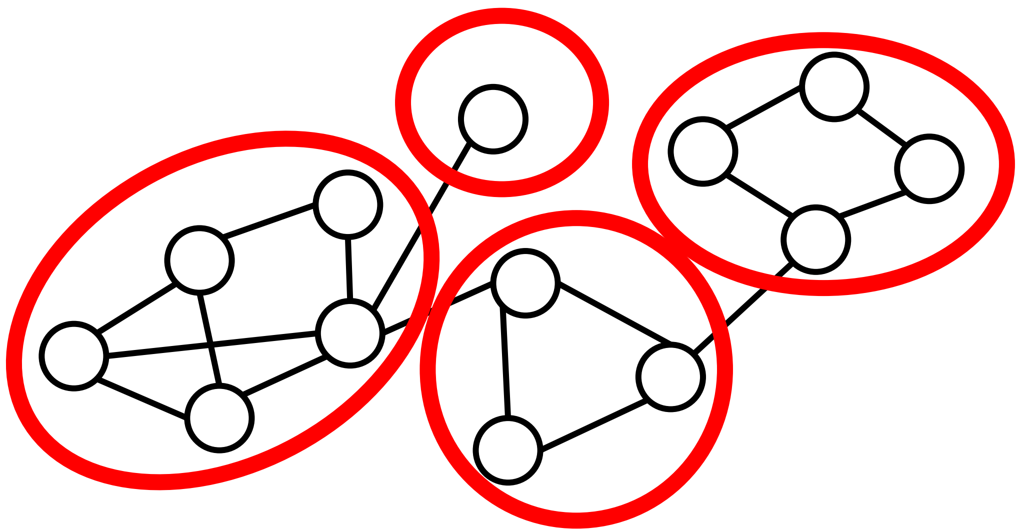
Corollary from Reduction #1

- Consider any connected graph H



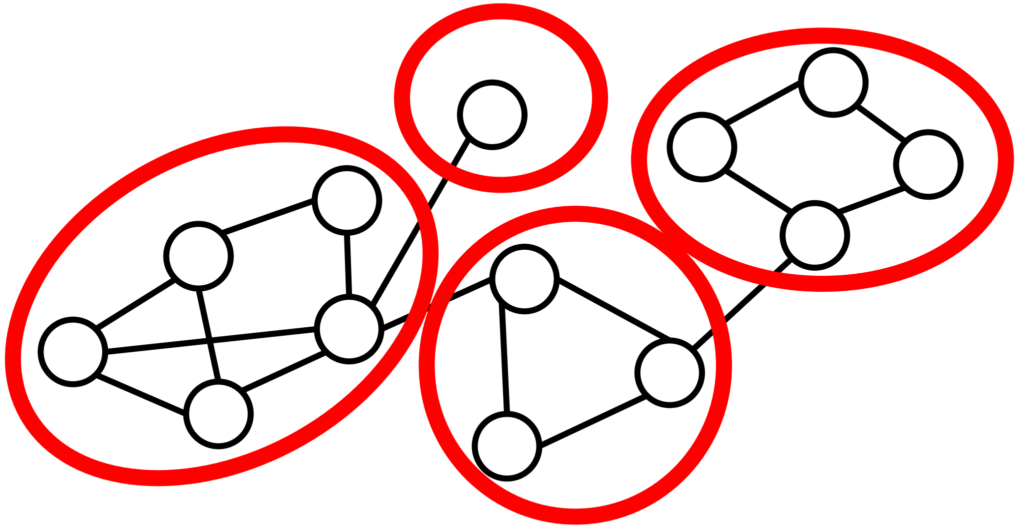
Corollary from Reduction #1

- Consider any connected graph H
- Decompose H into a tree of 2-connected components $C_1, \dots, C_k$



Corollary from Reduction #1

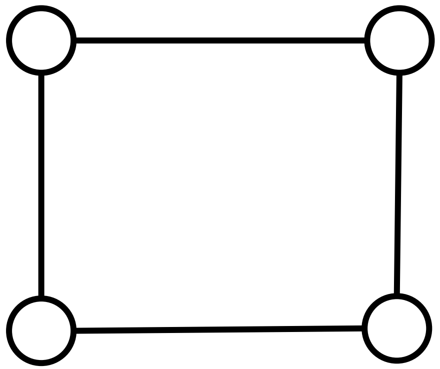
- Consider any connected graph H
- Decompose H into a tree of 2-connected components $C_1, \ldots, C_k$
- Hardness of H-freeness $\geq \max_i$ (hardness of C_i-freeness)



Reduction #2: Edge Replacement

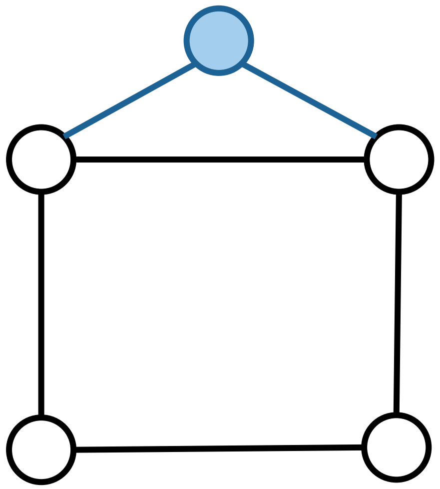
Reduction #2: Edge Replacement

- Take a 4-cycle (or larger cycle)
- Replace each edge by an arbitrary graph s.t. the result is 2-connected



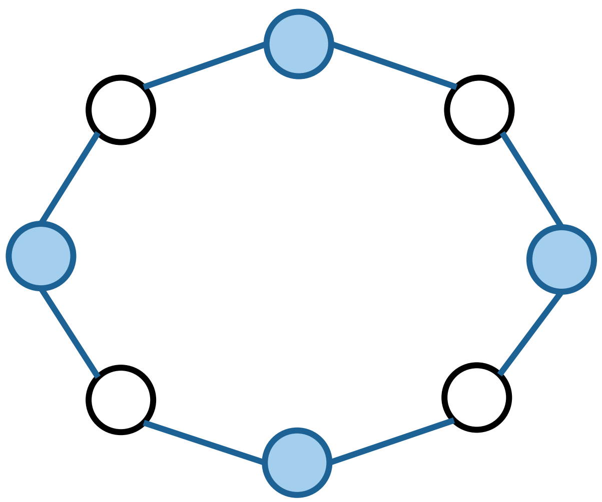
Reduction #2: Edge Replacement

- Take a 4-cycle (or larger cycle)
- Replace each edge by an arbitrary graph s.t. the result is 2-connected
- Like this graph:



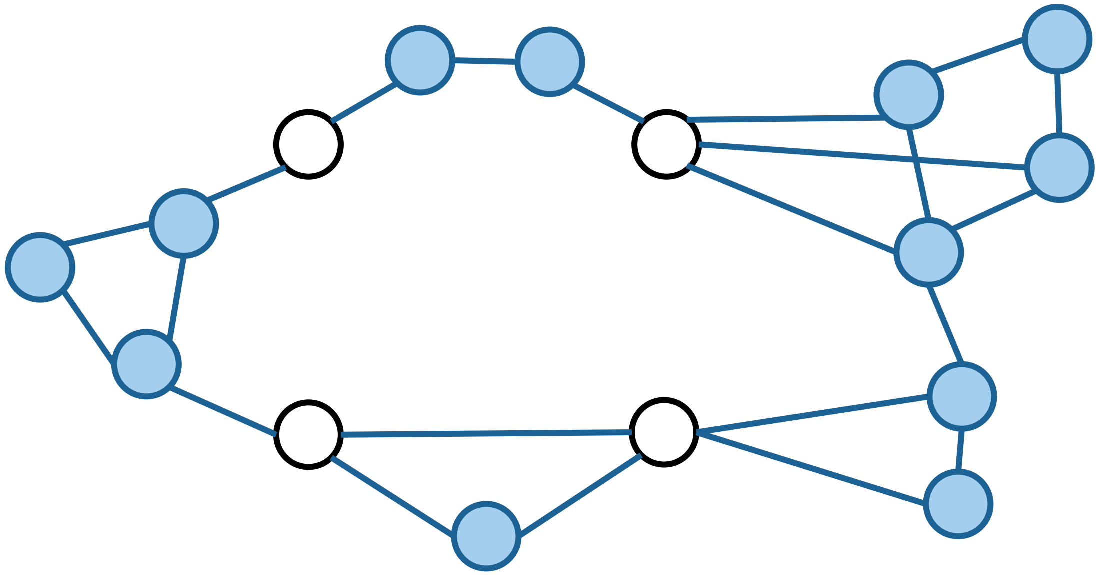
Reduction #2: Edge Replacement

- Take a 4-cycle (or larger cycle)
- Replace each edge by an arbitrary graph s.t. the result is 2-connected
- … or this graph:



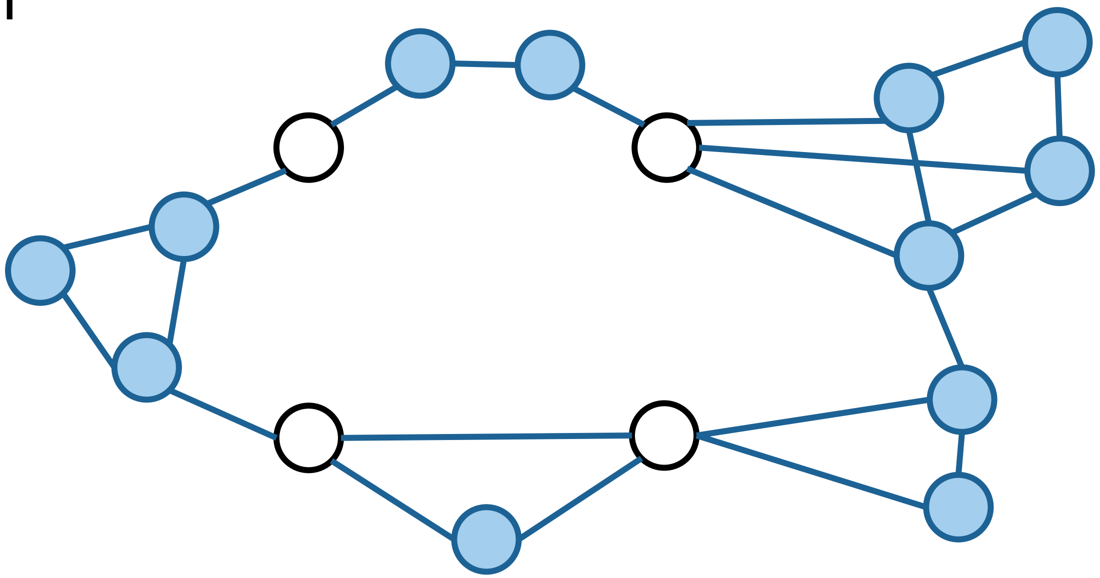
Reduction #2: Edge Replacement

- Take a 4-cycle (or larger cycle)
- Replace each edge by an arbitrary graph s.t. the result is 2-connected
- … or even this graph:



Reduction #2: Edge Replacement

- Take a 4-cycle (or larger cycle)
- Replace each edge by an arbitrary graph s.t. the result is 2-connected
- Obtain lower bound of $\Omega(n^{\delta})$ rounds
- $\delta \in (0, 1/2]$ depends on $|H|$



This Talk:

- Vertex replacement reduction:

This Talk:

- Vertex replacement reduction:
  - The reduction
  - Proof outline

This Talk:

- Vertex replacement reduction:
  - The reduction
  - Proof outline
- Outline of edge replacement reduction

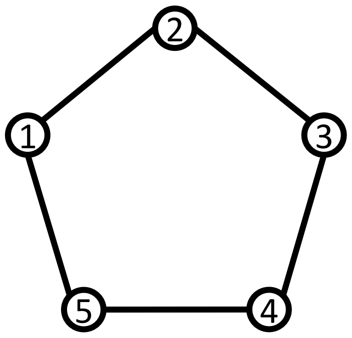
This Talk:

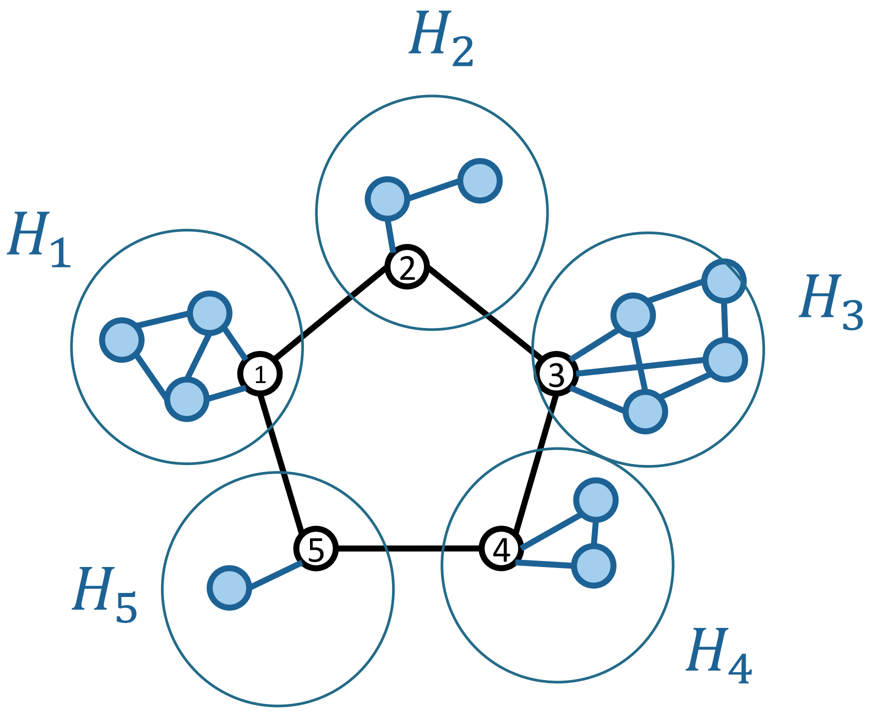
- Vertex replacement reduction:
  - The reduction
  - Proof outline
- Outline of edge replacement reduction
- … the rest is in the paper

Vertex Replacement Reduction

Vertex Replacement Reduction

- Notation:



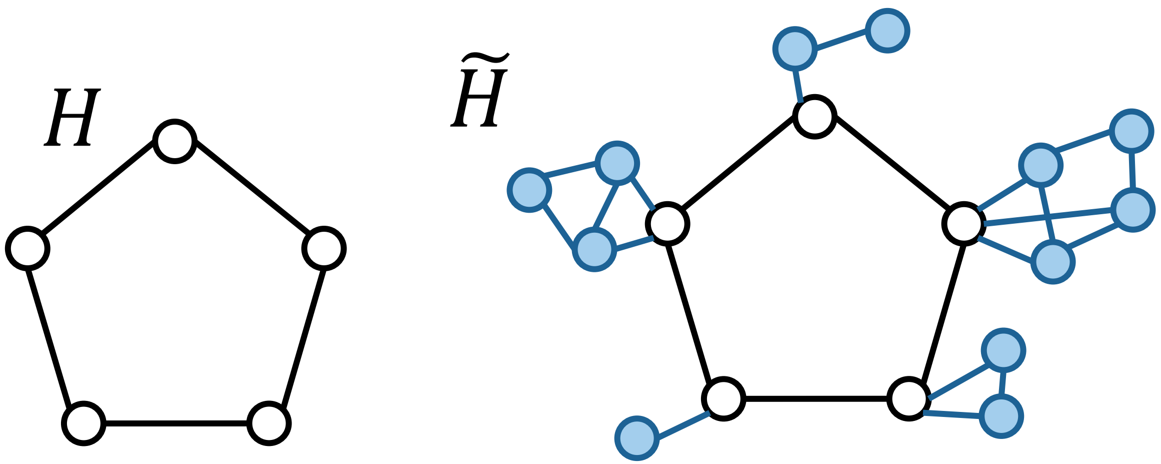


H: Graph we know a lower bound for

$\widetilde{H}$: Graph we want a lower bound for

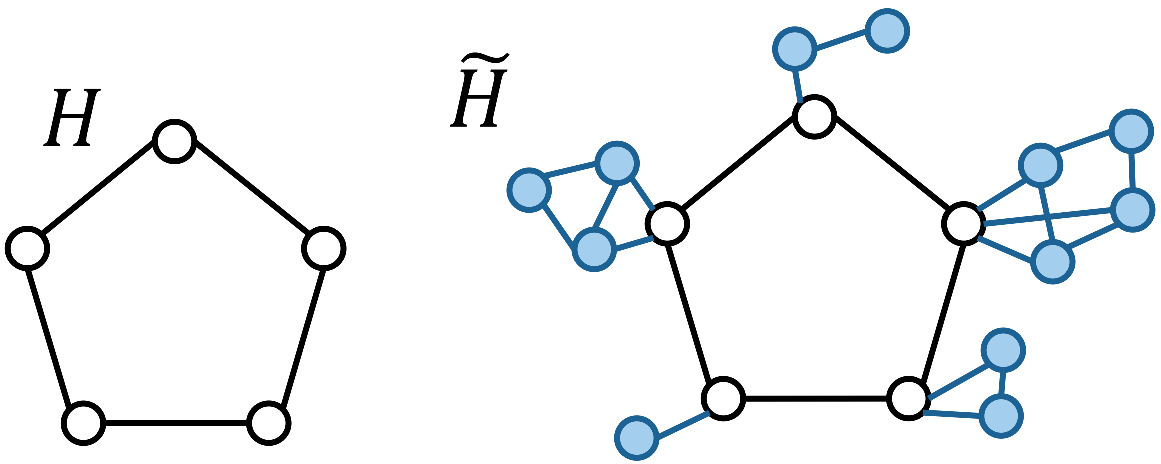
Vertex Replacement Reduction

- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness



Vertex Replacement Reduction

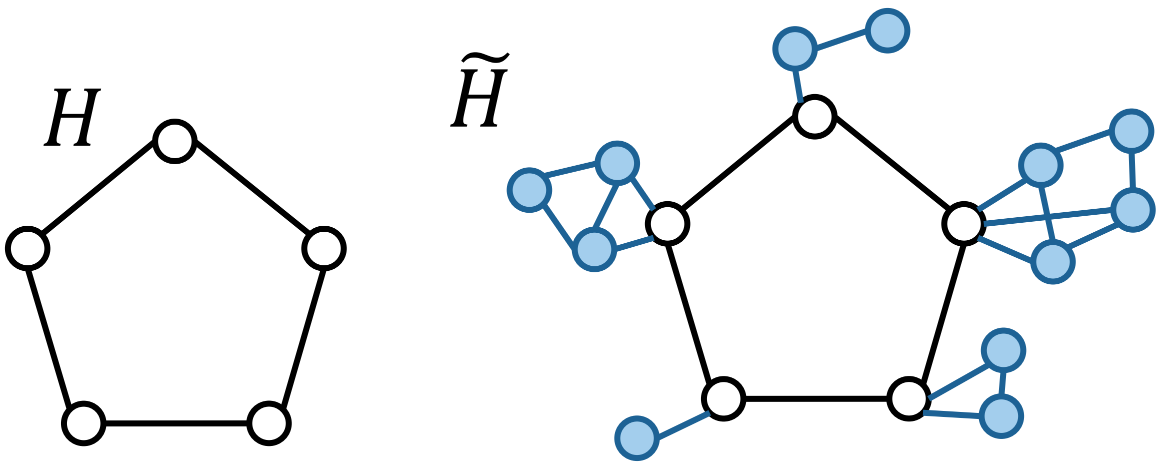
- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness
- Construction for testing H-freeness:



Vertex Replacement Reduction

- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness
- Construction for testing H-freeness:

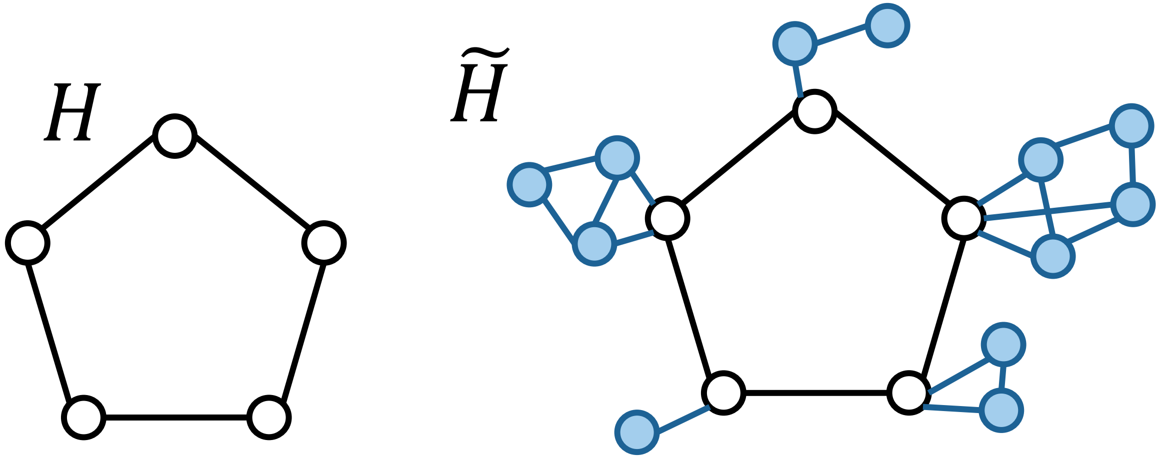
1. Each $v \in V$ chooses a "role in H", $c(v)$, and *"imagines"* it is connected to $H_{c(v)}$



Vertex Replacement Reduction

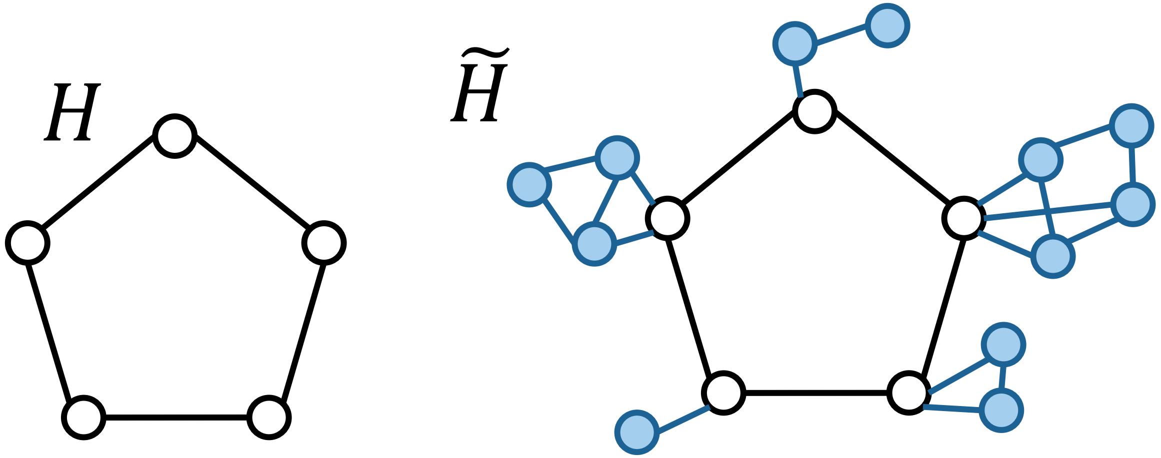
- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness
- Construction for testing H-freeness:

1. Each $v \in V$ chooses a "role in H", $c(v)$, and *"imagines"* it is connected to $H_{c(v)}$
2. Simulate $\tilde{A}$ on resulting network (real + imaginary)



Vertex Replacement Reduction

- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness
- Construction for testing H-freeness:
  1. Each $v \in V$ chooses a "role in H", $c(v)$, and *"imagines"* it is connected to $H_{c(v)}$
  2. Simulate $\tilde{A}$ on resulting network (real + imaginary)
- How to choose a role?

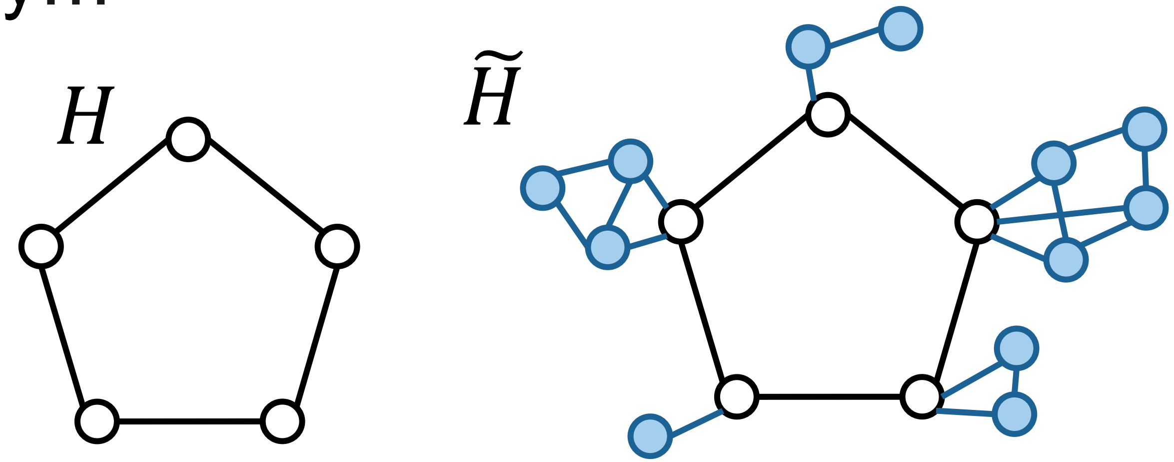


Vertex Replacement Reduction

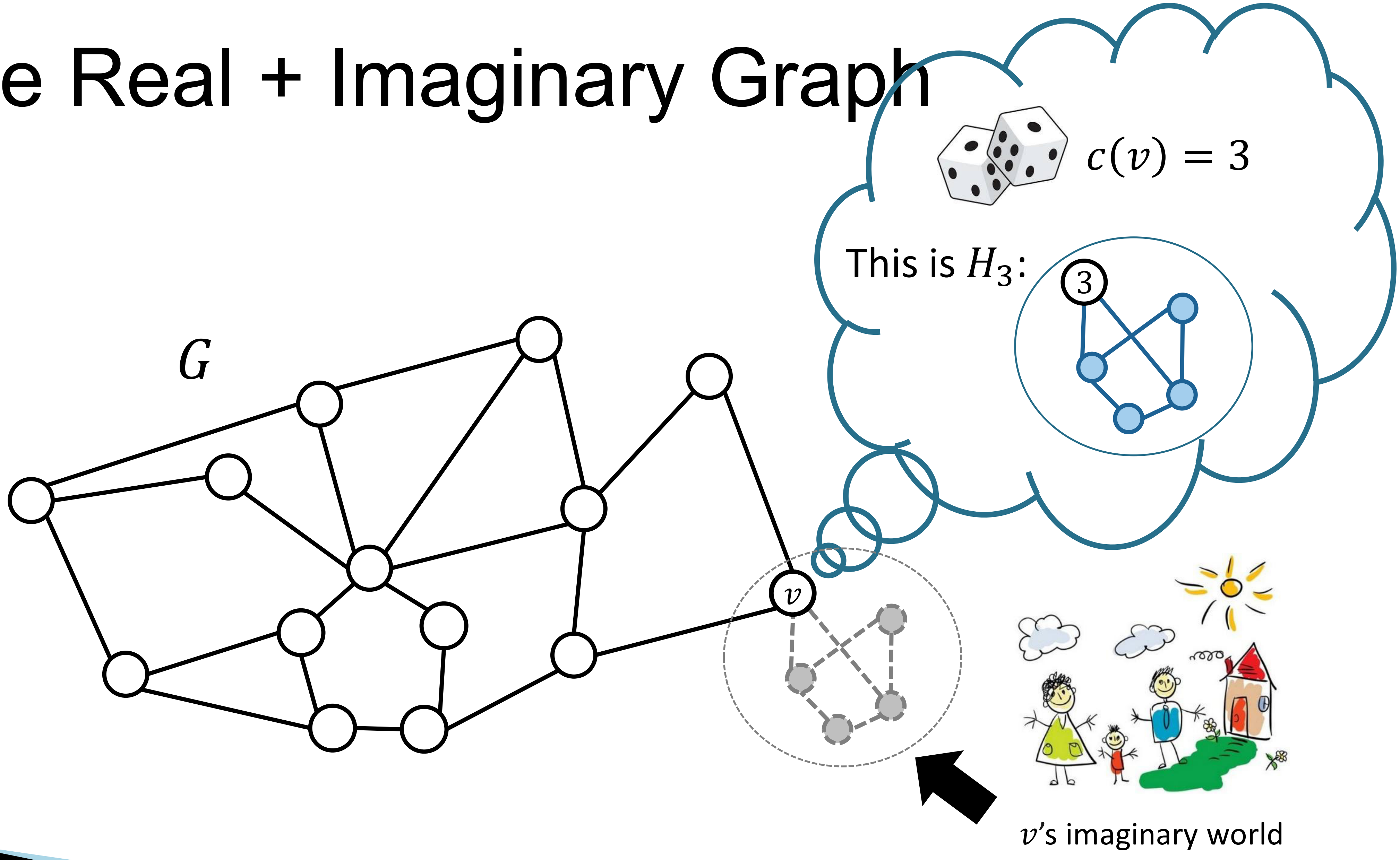
- Fix an algorithm $\tilde{A}$ for testing $\widetilde{H}$-freeness
- Construction for testing H-freeness:

1. Each $v \in V$ chooses a "role in H", $c(v)$, and *"imagines"* it is connected to $H_{c(v)}$
2. Simulate $\tilde{A}$ on resulting network (real + imaginary)

- How to choose a role? Randomly…



The Real + Imaginary Graph



Vertex Replacement Reduction

- Let $k = |V(H)|$

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph
- Claim:
    1. If G contains H, then $\tilde{G}$ contains $\widetilde{H}$ w.p. $\geq 1/k^k$

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph
- Claim:
  1. If G contains H, then $\tilde{G}$ contains $\widetilde{H}$ w.p. $\geq 1/k^k$

So $\tilde{A}$ must reject

…with high probability

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph
- Claim:
  1. If G contains H, then $\tilde{G}$ contains $\widetilde{H}$ w.p. $\geq 1/k^k$
  2. If G does not contain H, then $\tilde{G}$ never contains $\widetilde{H}$

So $\tilde{A}$ must reject

…with high probability

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph
- Claim:
  1. If G contains H, then $\tilde{G}$ contains $\widetilde{H}$ w.p. $\geq 1/k^k$
  2. If G does not contain H, then $\tilde{G}$ never contains $\widetilde{H}$

So $\tilde{A}$ must reject

…with high probability

So $\tilde{A}$ must accept

…with high probability

Vertex Replacement Reduction

- Let $k = |V(H)|$
- Let $\tilde{G}$ be the real + imaginary network graph
- Claim:
  1. If G contains H, then $\tilde{G}$ contains $\widetilde{H}$ w.p. $\geq 1/k^k$
  2. If G does not contain H, then $\tilde{G}$ never contains $\widetilde{H}$
- Reduction can be derandomized
  - Additional $O(\log n)$ factor

So $\tilde{A}$ must reject

…with high probability

So $\tilde{A}$ must accept

…with high probability

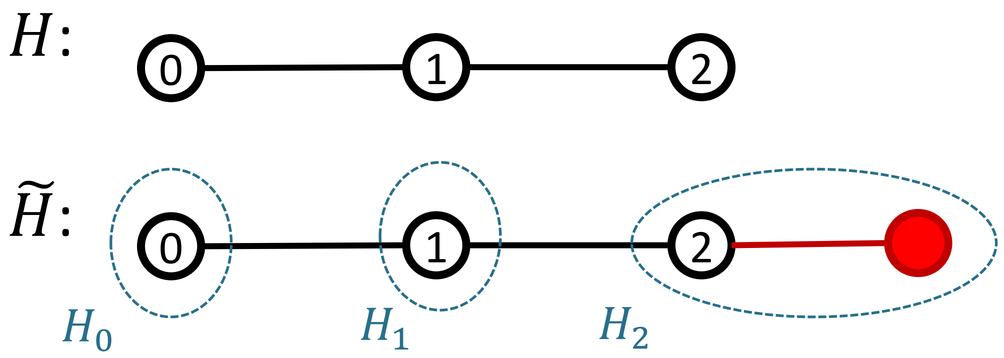
What Could Go Wrong?

What Could Go Wrong?

- If H is not 2-connected:

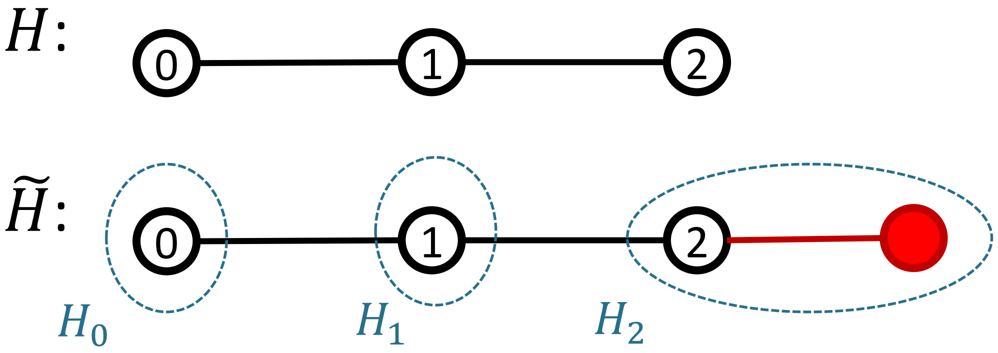
What Could Go Wrong?

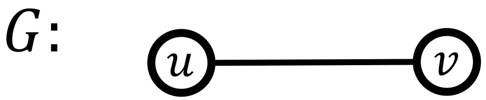
- If H is not 2-connected:



What Could Go Wrong?

- If H is not 2-connected:

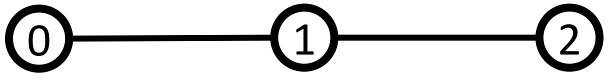




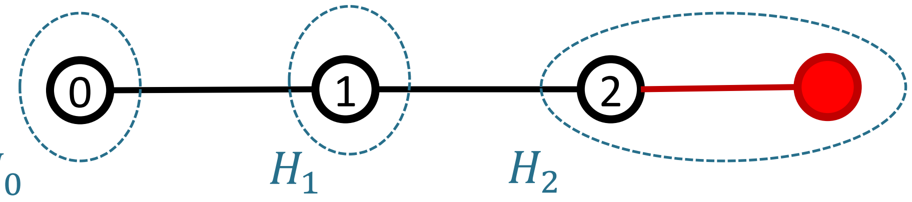
What Could Go Wrong?

- If H is not 2-connected:

H:



$\widetilde{H}$:



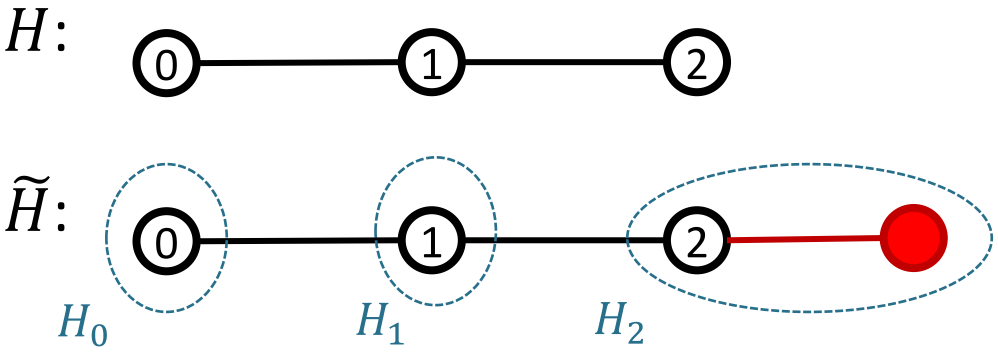
G:

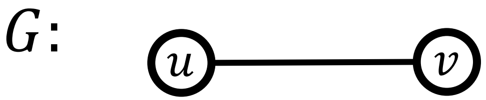
G does not contain H! But…

If $c(u) = c(v) = 2$:

What Could Go Wrong?

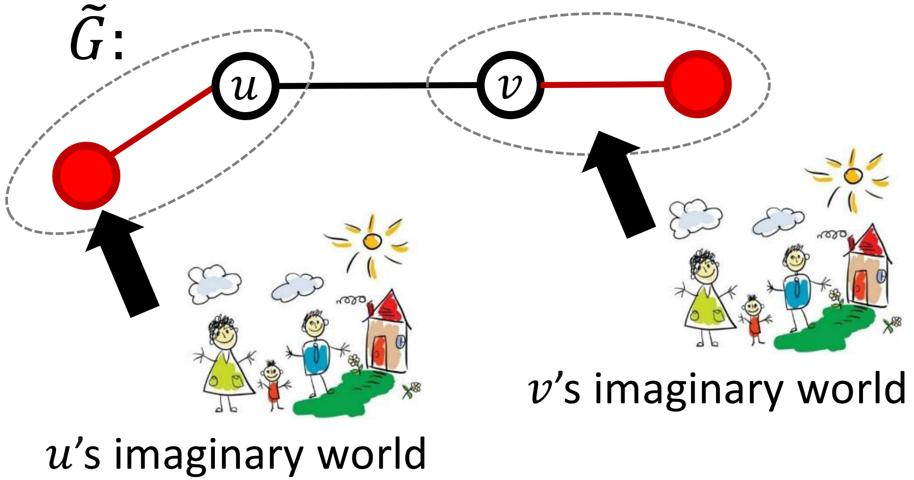
- If H is not 2-connected:





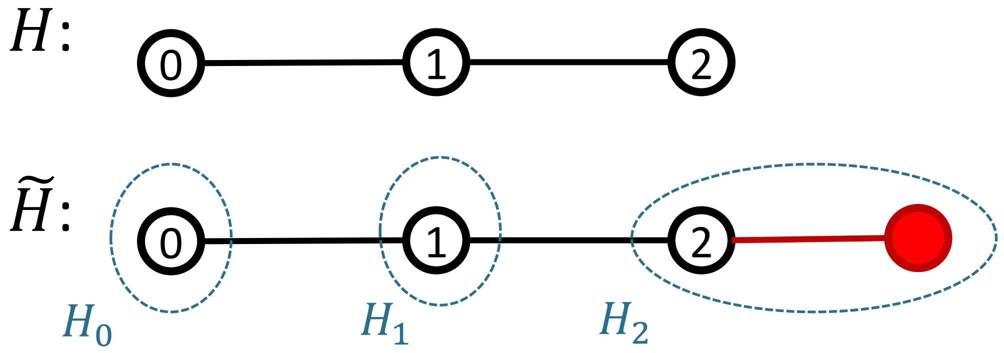
G does not contain H! But…

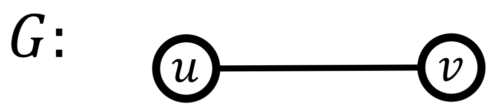
If $c(u) = c(v) = 2$:



What Could Go Wrong?

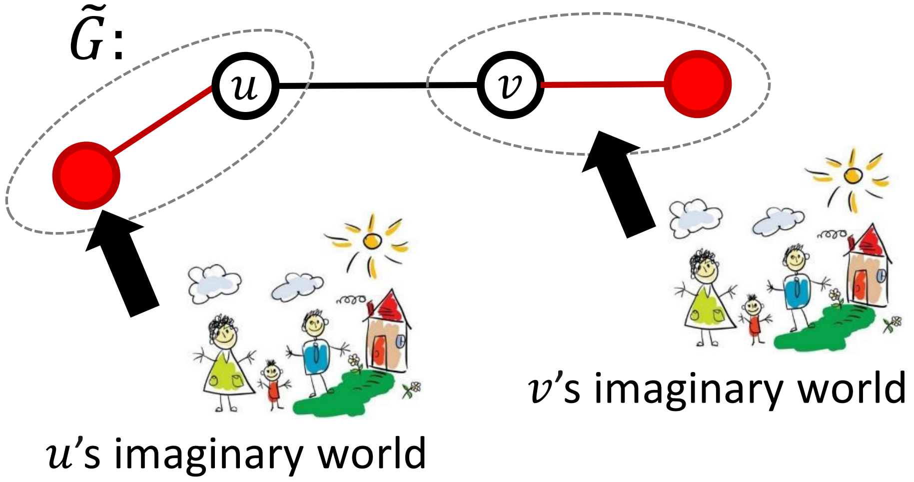
- If H is not 2-connected:





G does not contain H! But…

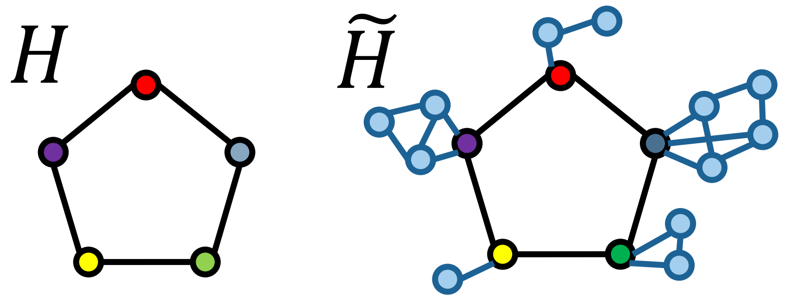
If $c(u) = c(v) = 2$:



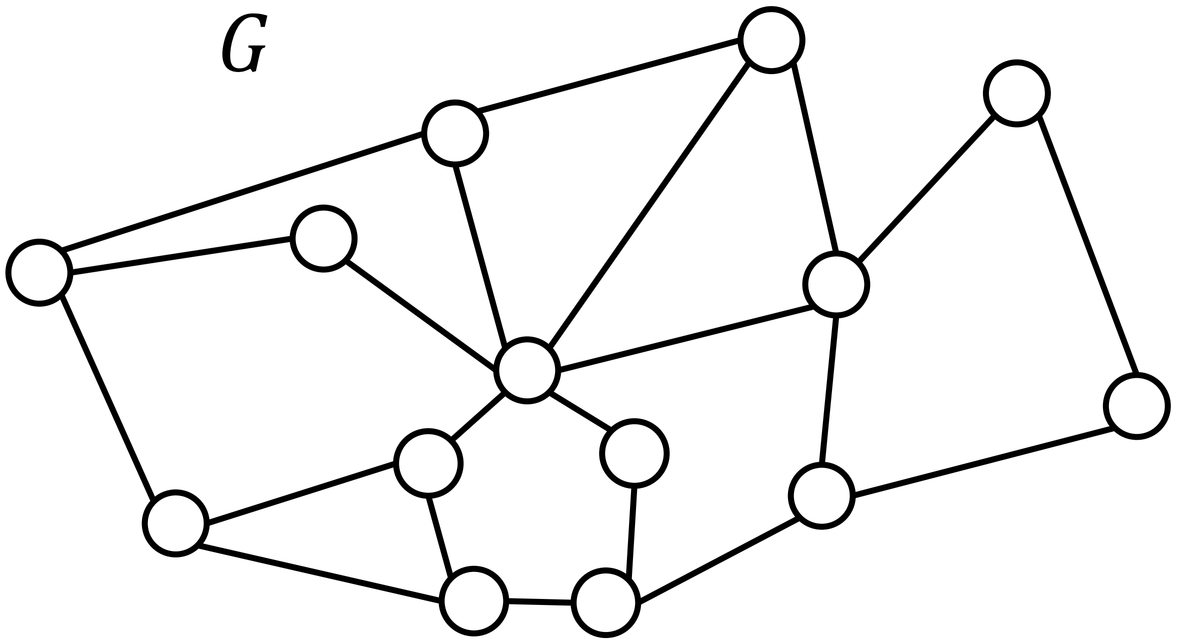
$\tilde{G}$ contains $\widetilde{H}$…

Correctness Proof

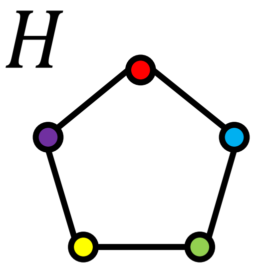
Correctness Proof

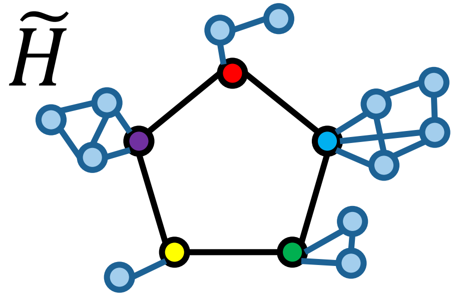


- Suppose G contains H as a subgraph

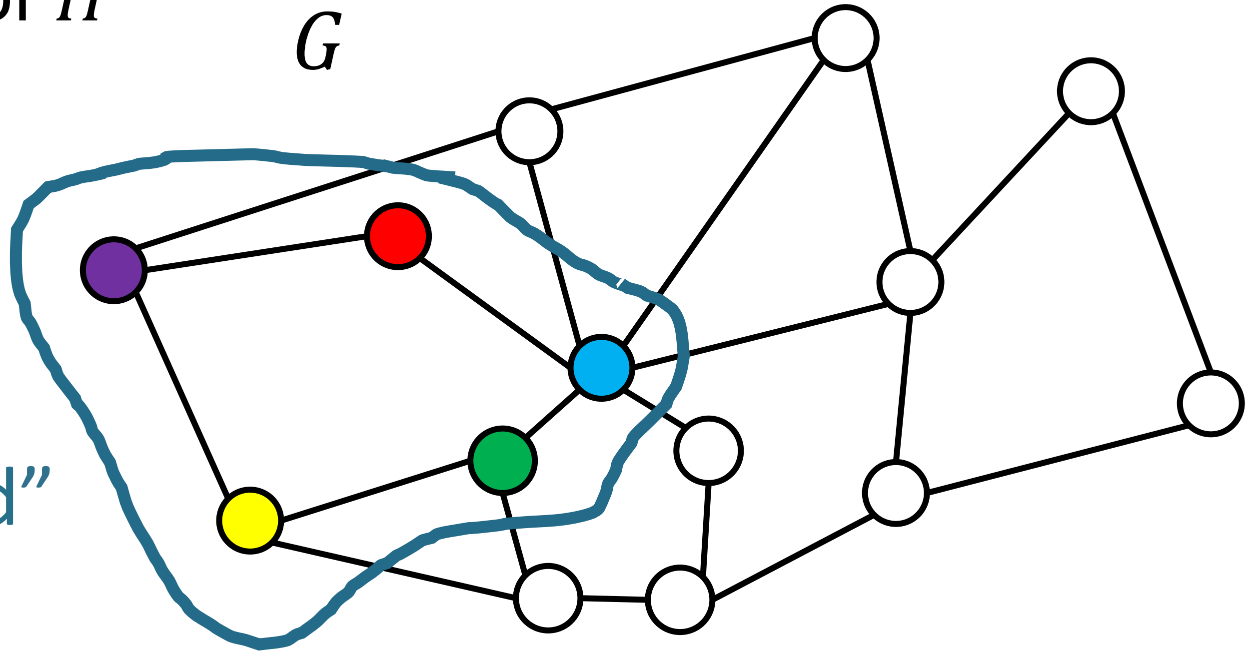


Correctness Proof



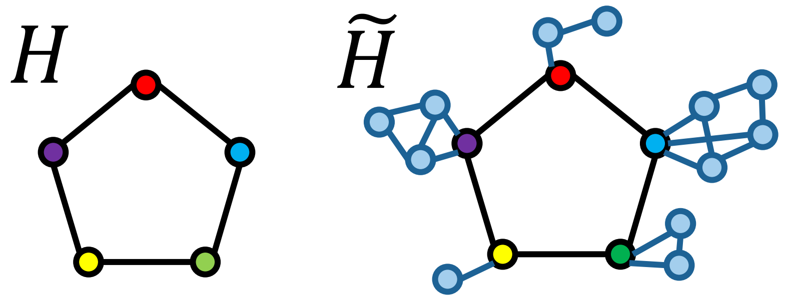


- Suppose G contains H as a subgraph
- W.p. $\geq 1/k^k$, G contains a properly colored copy of H

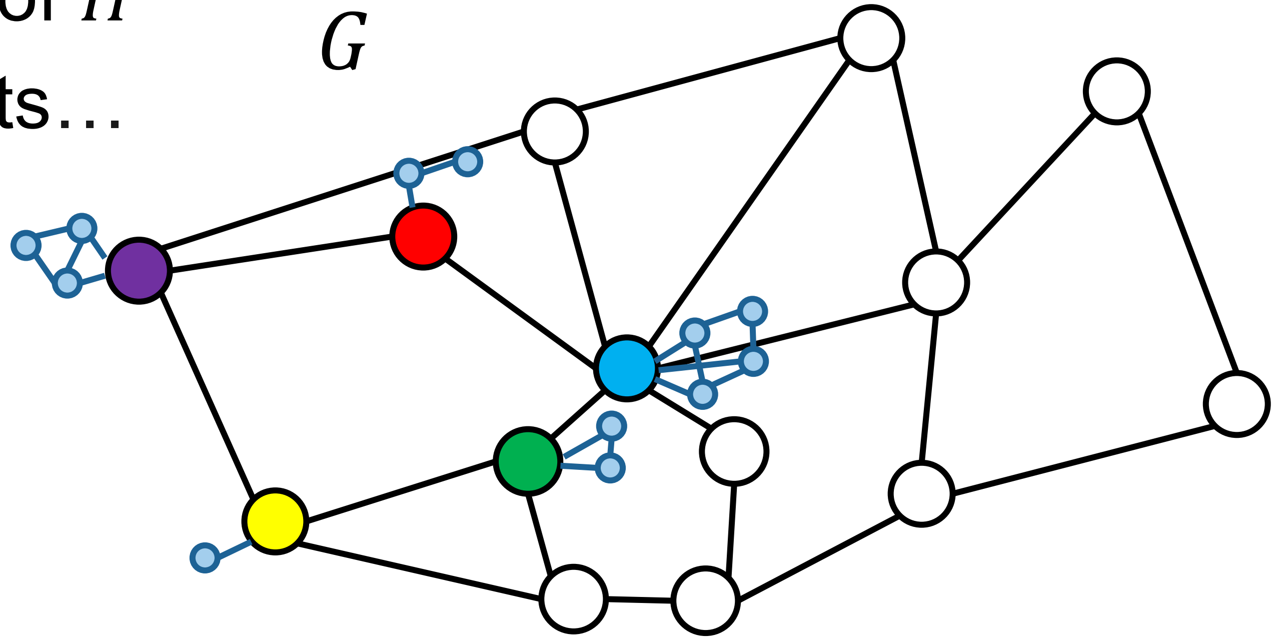


"Properly colored"

Correctness Proof



- Suppose G contains H as a subgraph
- W.p. $\geq 1/k^k$, G contains a properly colored copy of H
- After adding imaginary parts…
- A copy of $\widetilde{H}$ appears



Correctness Proof

- Now suppose G does not contain H

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…
  - Let $\sigma: \widetilde{H} \rightarrow \tilde{G}$ be an isomorphism from $\widetilde{H}$ into $\tilde{G}$

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…
  - Let $\sigma: \widetilde{H} \rightarrow \tilde{G}$ be an isomorphism from $\widetilde{H}$ into $\tilde{G}$
  - Let $\sigma(\widetilde{H})$ be the image of $\widetilde{H}$ inside $\tilde{G}$

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…
  - Let $\sigma: \widetilde{H} \rightarrow \tilde{G}$ be an isomorphism from $\widetilde{H}$ into $\tilde{G}$
  - Let $\sigma(\widetilde{H})$ be the image of $\widetilde{H}$ inside $\tilde{G}$
  - Let $\sigma(H)$ be the image of H inside $\sigma(\widetilde{H})$

Correctness Proof

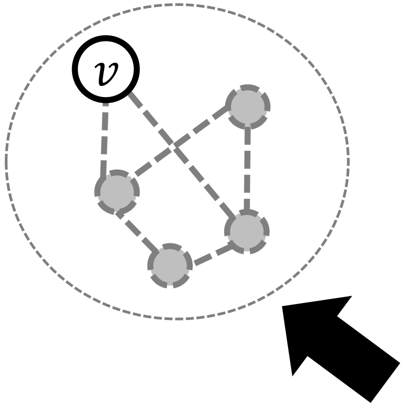
- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…
  - Let $\sigma : \widetilde{H} \to \tilde{G}$ be an isomorphism from $\widetilde{H}$ into $\tilde{G}$
  - Let $\sigma\left(\widetilde{H}\right)$ be the image of $\widetilde{H}$ inside $\tilde{G}$
  - Let $\sigma(H)$ be the image of H inside $\sigma\left(\widetilde{H}\right)$
    - G is H-free $\Rightarrow \sigma(H)$ must contain some imaginary nodes!

Correctness Proof

- Now suppose G does not contain H
- We show that for any coloring of G, resulting $\tilde{G}$ does not contain $\widetilde{H}$
- Suppose that it does…
  - Let $\sigma: \widetilde{H} \rightarrow \tilde{G}$ be an isomorphism from $\widetilde{H}$ into $\tilde{G}$
  - Let $\sigma(\widetilde{H})$ be the image of $\widetilde{H}$ inside $\tilde{G}$
  - Let $\sigma(H)$ be the image of H inside $\sigma(\widetilde{H})$
    - G is H-free $\Rightarrow \sigma(H)$ must contain some imaginary nodes!
- Proof strategy: show that $\tilde{G}$ contains infinitely many copies of H…

Correctness Proof

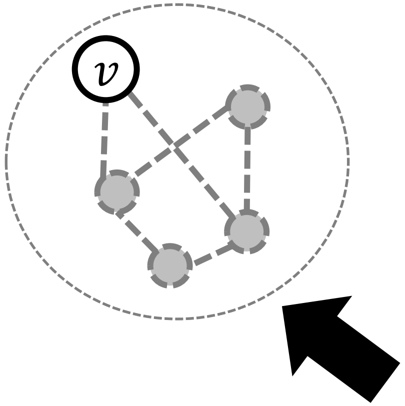
- Lemma 1: $\sigma(H)$ is inside the imaginary world of some vertex v



v's imaginary world

Correctness Proof

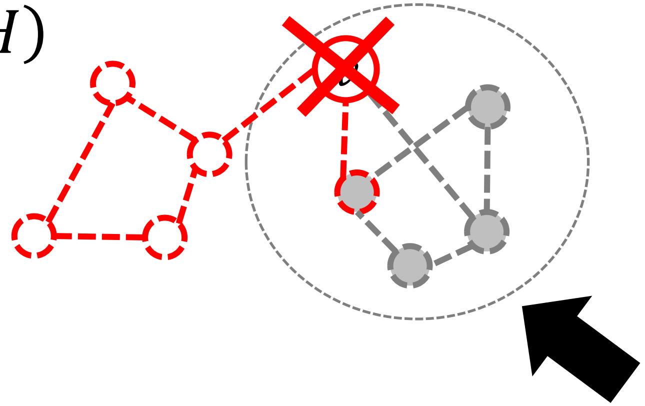
- Lemma 1: $\sigma(H)$ is inside the imaginary world of some vertex v
- Proof:
  - We said: $\sigma(H)$ intersects some vertex v's imaginary world
  - Can't reach rest of the world except through v



v's imaginary world

Correctness Proof

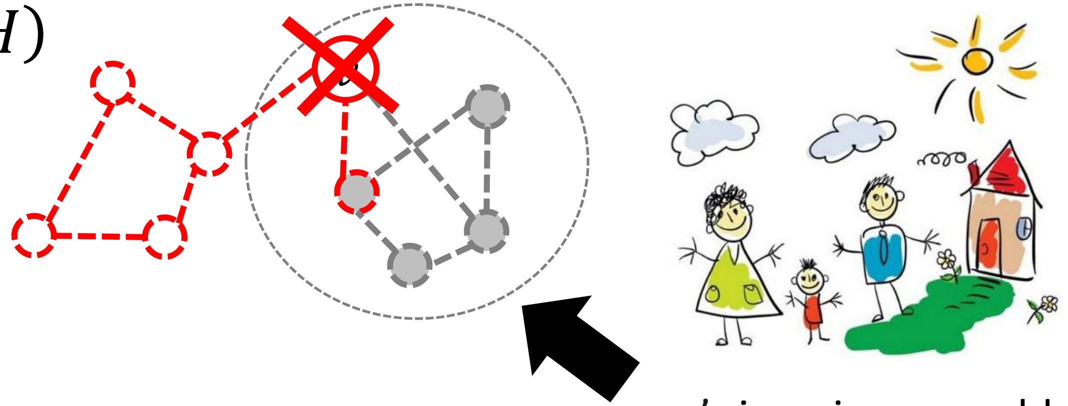
- Lemma 1: $\sigma(H)$ is inside the imaginary world of some vertex v
- Proof:
  - We said: $\sigma(H)$ intersects some vertex v's imaginary world
  - Can't reach rest of the world except through v
  - If $\sigma(H)$ "spills outside"…
  - Removing v disconnects $\sigma(H)$



v's imaginary world

Correctness Proof

- Lemma 1: $\sigma(H)$ is inside the imaginary world of some vertex v
- Proof:
  - We said: $\sigma(H)$ intersects some vertex v's imaginary world
  - Can't reach rest of the world except through v
  - If $\sigma(H)$ "spills outside"…
  - Removing v disconnects $\sigma(H)$
  - … but H is 2-connected

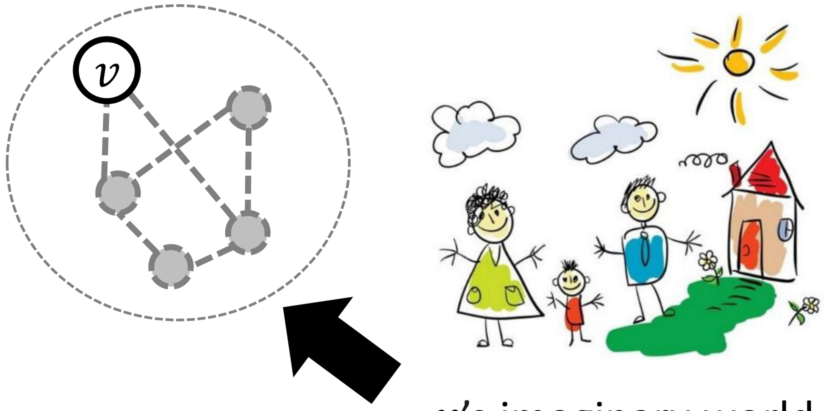


v's imaginary world

Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

- <u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \not\subseteq v$'s imaginary world
- <u>Proof:</u> not enough room in there…



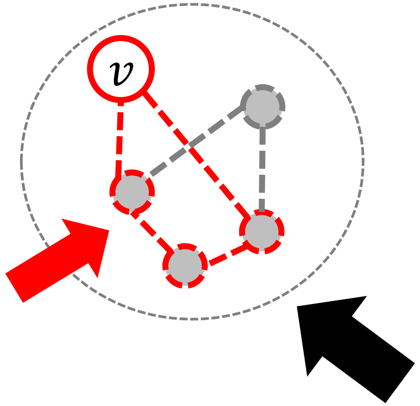
v's imaginary world

Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

- <u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \nsubseteq v$'s imaginary world
- <u>Proof:</u> not enough room in there…

imaginary world $\sim H_{c(v)}$

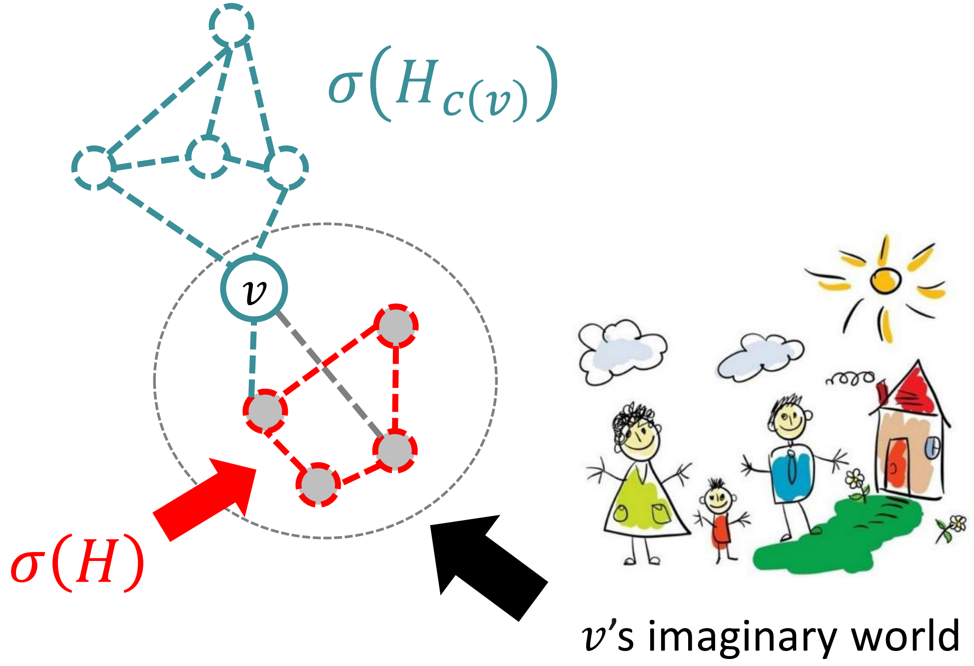


$\sigma(H)$ takes up space

v's imaginary world

Correctness Proof

- ✓ <u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v
- ✓ <u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \not\subseteq v$'s imaginary world
- <u>Lemma 3:</u> $v \in \sigma\left(H_{c(v)}\right)$

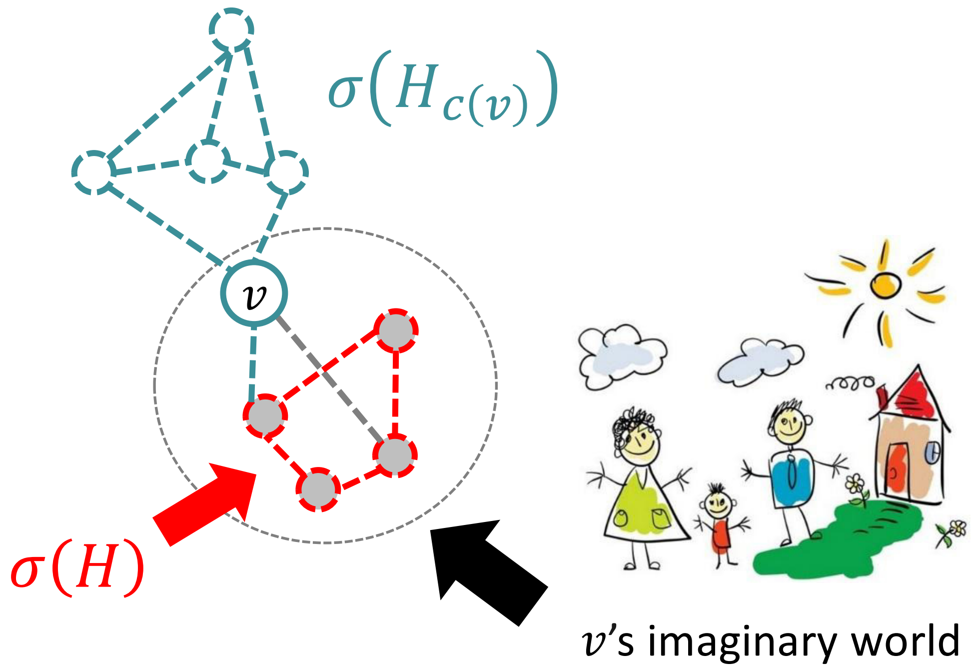


Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma(H_{c(v)}) \nsubseteq v$'s imaginary world

- <u>Lemma 3:</u> $v \in \sigma(H_{c(v)})$
- <u>Proof:</u>
  - $\sigma(H) \subseteq v$'s imaginary world

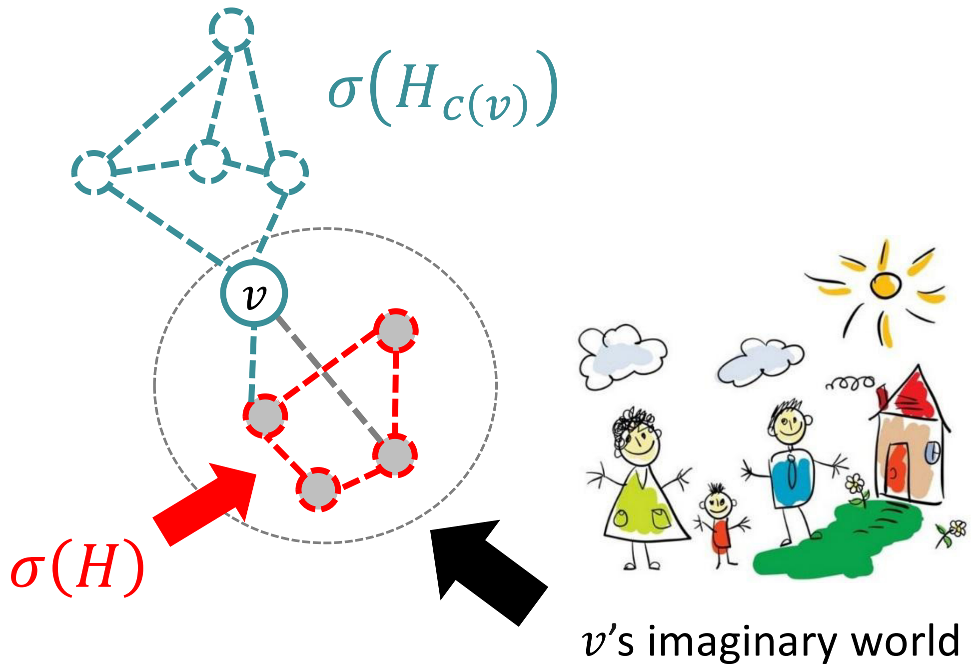


Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

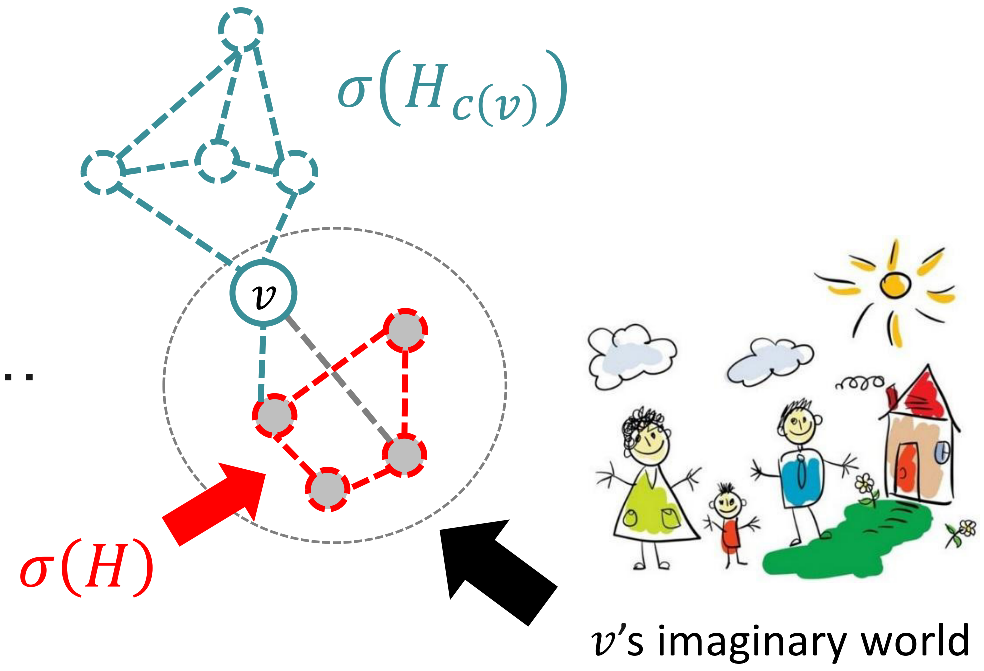
✓<u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \nsubseteq v$'s imaginary world

- <u>Lemma 3:</u> $v \in \sigma\left(H_{c(v)}\right)$
- <u>Proof:</u>
  - $\sigma(H) \subseteq v$'s imaginary world
  - $\sigma\left(H_{c(v)}\right)$ spills out



Correctness Proof

- ✓ <u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v
- ✓ <u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \not\subseteq v$'s imaginary world
- <u>Lemma 3:</u> $v \in \sigma\left(H_{c(v)}\right)$
- <u>Proof:</u>
  - $\sigma(H) \subseteq v$'s imaginary world
  - $\sigma\left(H_{c(v)}\right)$ spills out
  - Can't get to $\sigma\left(H_{c(v)}\right)$ except through v…

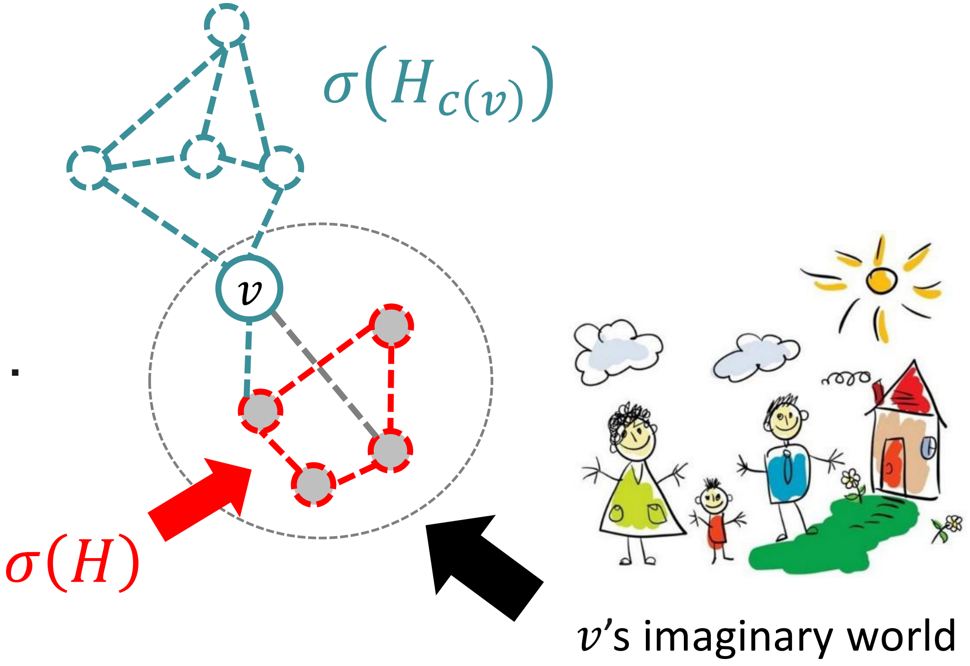


Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma(H_{c(v)}) \not\subseteq v$'s imaginary world

- <u>Lemma 3:</u> $v \in \sigma(H_{c(v)})$
- <u>Proof:</u>
  - $\sigma(H) \subseteq v$'s imaginary world
  - $\sigma(H_{c(v)})$ spills out
  - Can't get to $\sigma(H_{c(v)})$ except through v…
  - So $v \in \sigma(H_{c(v)})$



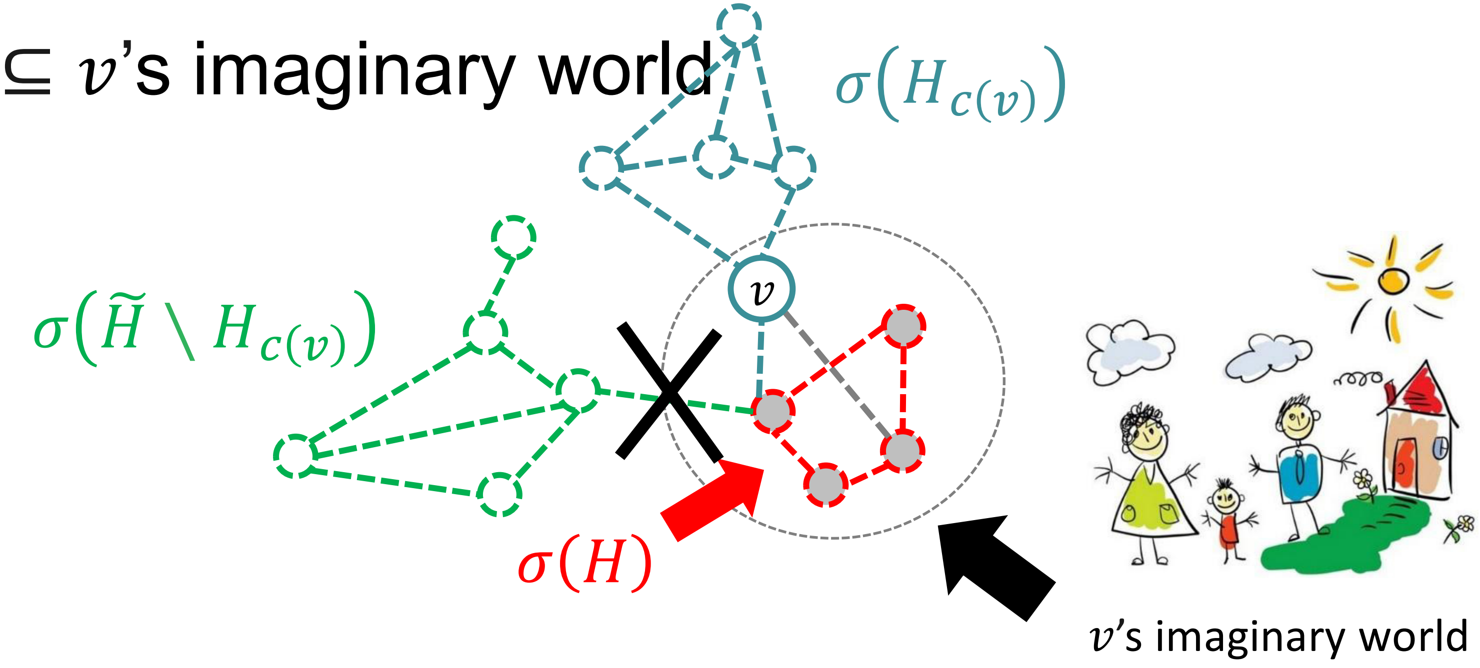
Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma(H_{c(v)}) \nsubseteq v$'s imaginary world

✓<u>Lemma 3:</u> $v \in \sigma(H_{c(v)})$

- <u>Lemma 4:</u> $\sigma(\widetilde{H} \setminus H_{c(v)}) \subseteq v$'s imaginary world



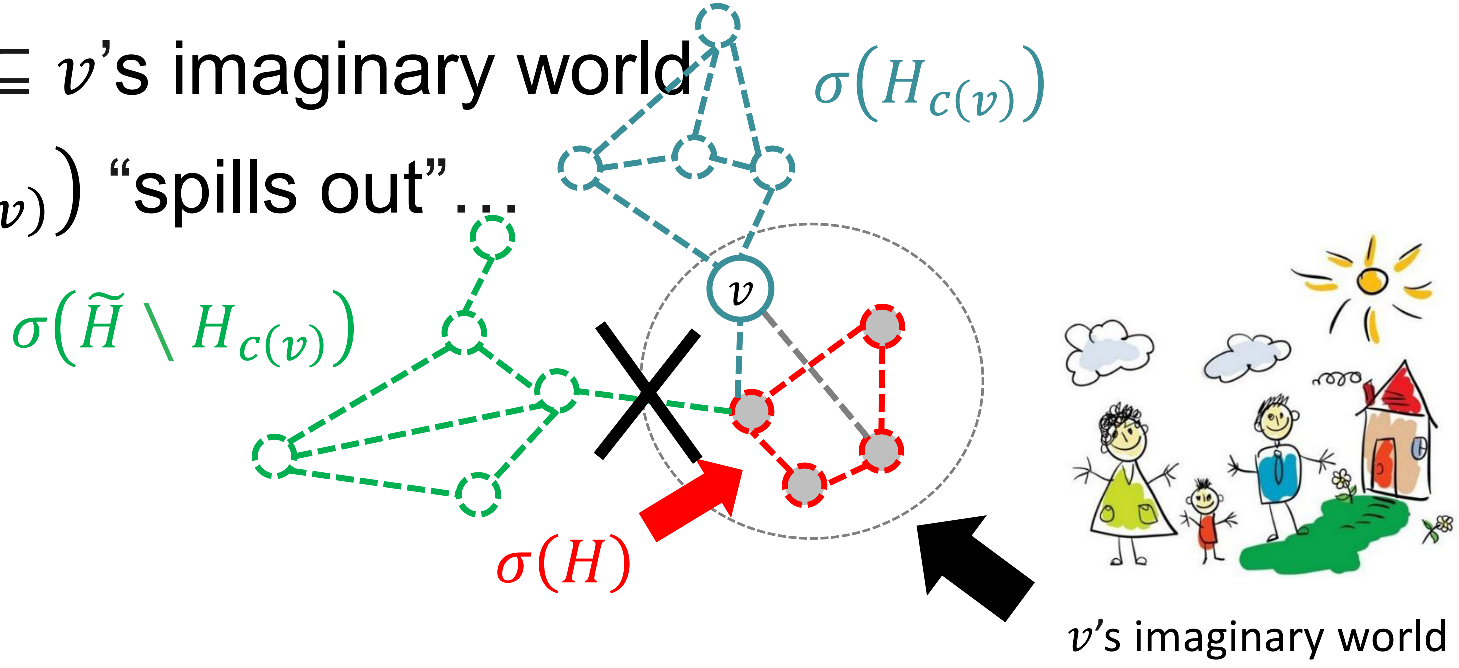
Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma(H_{c(v)}) \not\subseteq v$'s imaginary world

✓<u>Lemma 3:</u> $v \in \sigma(H_{c(v)})$

- <u>Lemma 4:</u> $\sigma(\widetilde{H} \setminus H_{c(v)}) \subseteq v$'s imaginary world
- <u>Proof:</u> suppose $\sigma(\widetilde{H} \setminus H_{c(v)})$ "spills out"…



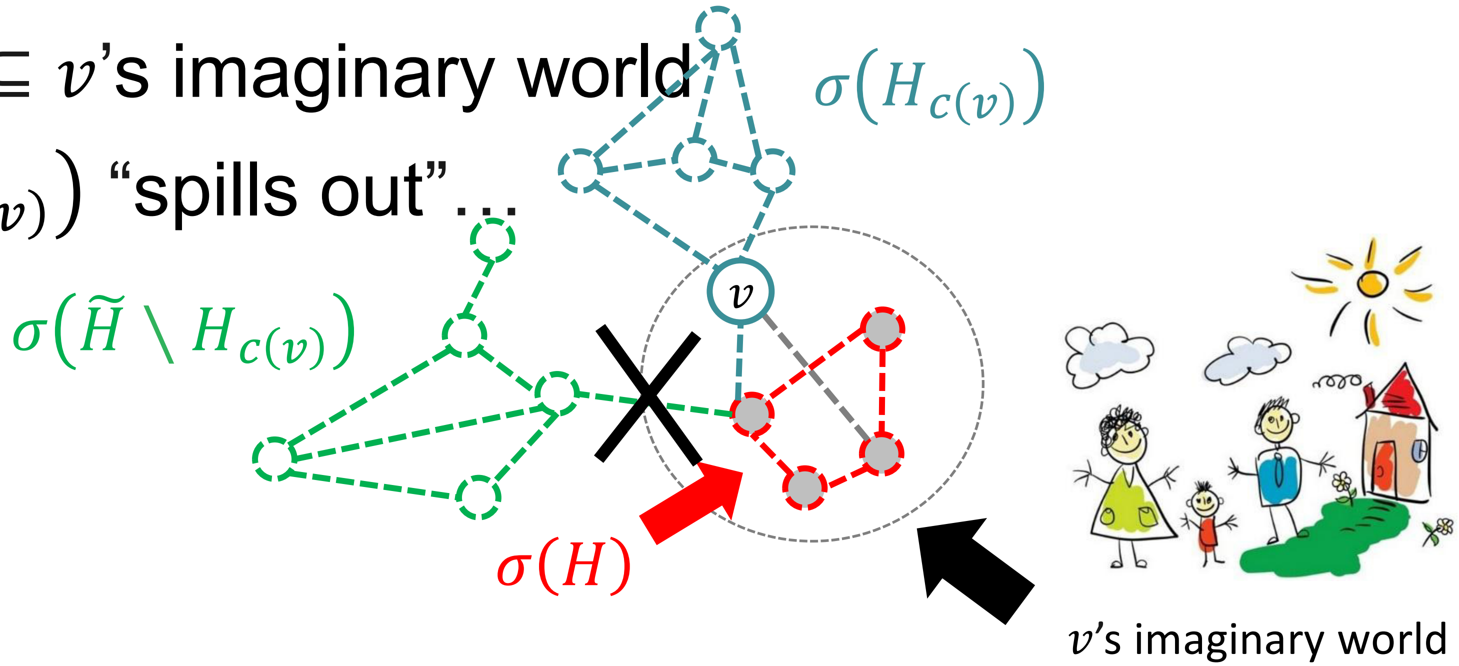
Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma(H_{c(v)}) \nsubseteq v$'s imaginary world

✓<u>Lemma 3:</u> $v \in \sigma(H_{c(v)})$

- <u>Lemma 4:</u> $\sigma(\widetilde{H} \setminus H_{c(v)}) \subseteq v$'s imaginary world
- <u>Proof:</u> suppose $\sigma(\widetilde{H} \setminus H_{c(v)})$ "spills out"…
  - But v is the only exit!



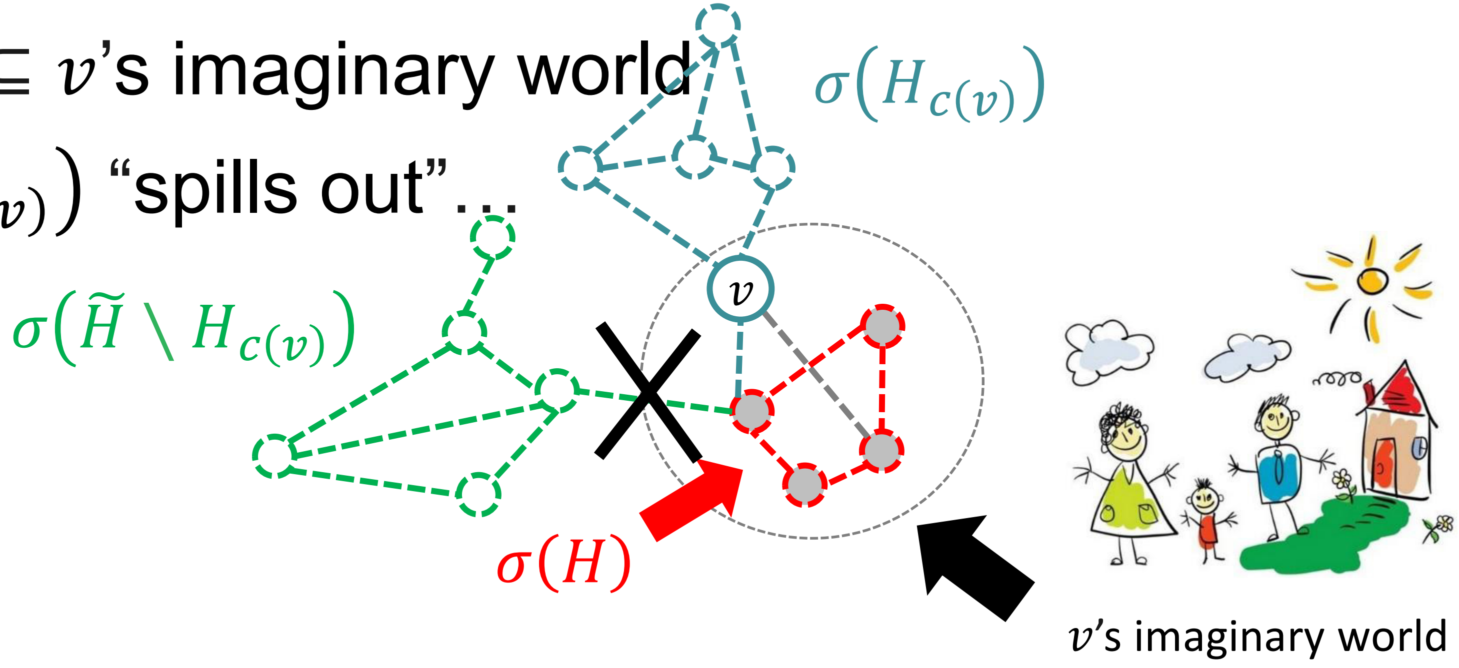
Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \not\subseteq v$'s imaginary world

✓<u>Lemma 3:</u> $v \in \sigma\left(H_{c(v)}\right)$

- <u>Lemma 4:</u> $\sigma\left(\widetilde{H} \setminus H_{c(v)}\right) \subseteq v$'s imaginary world
- <u>Proof:</u> suppose $\sigma\left(\widetilde{H} \setminus H_{c(v)}\right)$ "spills out"…
  - But v is the only exit!
  - And v is already taken…
    since $v \in \sigma\left(H_{c(v)}\right)$



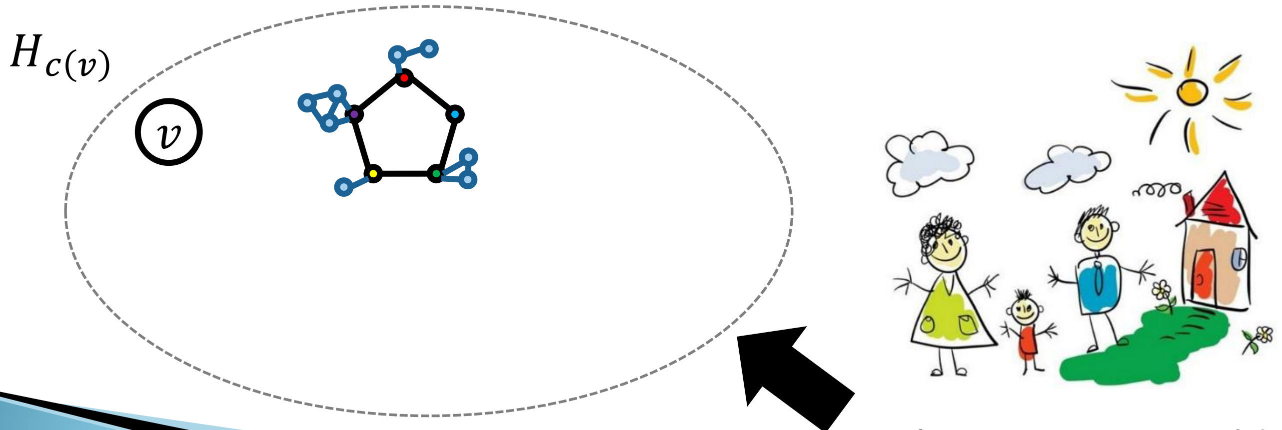
Correctness Proof

✓<u>Lemma 1:</u> $\sigma(H)$ is inside the imaginary world of some vertex v

✓<u>Lemma 2:</u> $\sigma\left(H_{c(v)}\right) \not\subseteq v$'s imaginary world

✓<u>Lemma 3:</u> $v \in \sigma\left(H_{c(v)}\right)$

✓<u>Lemma 4:</u> $\sigma\left(\widetilde{H} \setminus H_{c(v)}\right) \subseteq v$'s imaginary world $\sim H_{c(v)}$

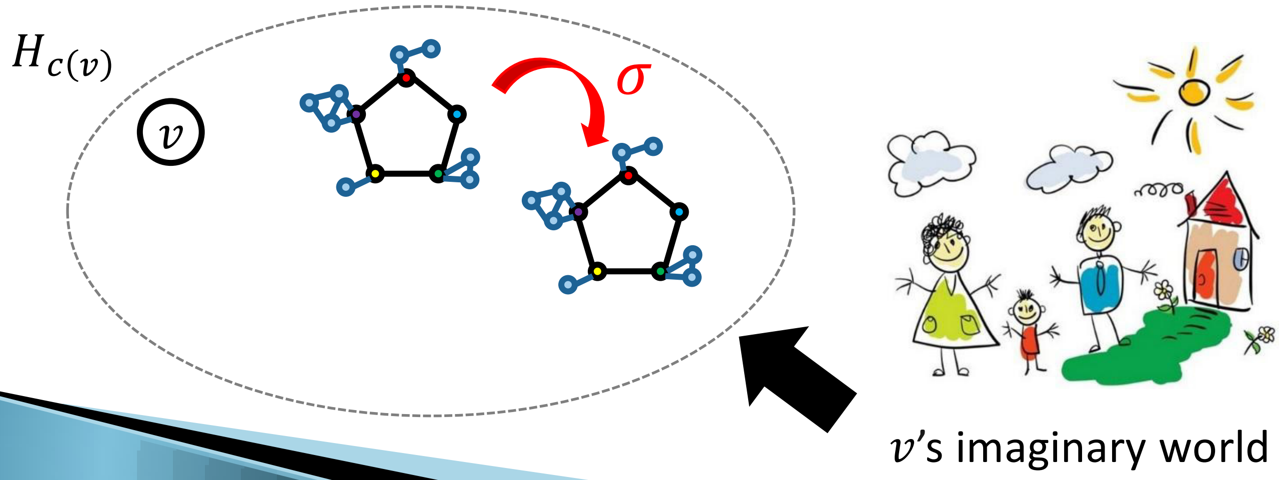


v's imaginary world

- So… $H_{c(v)}$ contains a copy of $\widetilde{H} \setminus H_{c(v)}$???
- What does σ do to this copy???

Correctness Proof

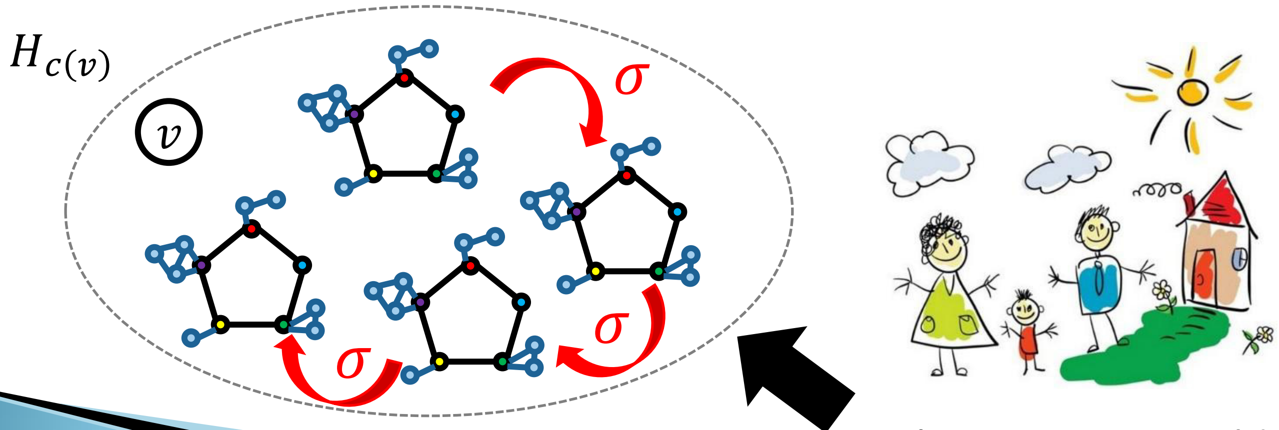
- <u>Claim:</u> σ maps the copy of $\widetilde{H} \setminus H_{c(v)}$ inside $H_{c(v)}$ to a new copy of $\widetilde{H} \setminus H_{c(v)}$ inside v's imaginary world
  - Can't re-use the old copy
  - Can't spill out



- So… $H_{c(v)}$ contains a copy of $\widetilde{H} \setminus H_{c(v)}$???
- What does σ do to this copy???

Correctness Proof

- <u>Claim:</u> σ maps the copy of $\widetilde{H} \setminus H_{c(v)}$ inside $H_{c(v)}$ to a new copy of $\widetilde{H} \setminus H_{c(v)}$ inside v's imaginary world
  - Can't re-use the old copy
  - Can't spill out
- By induction…

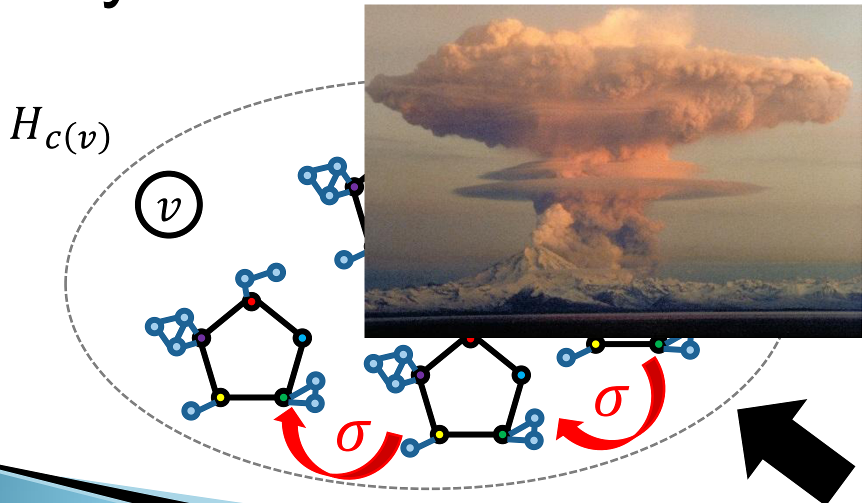


v's imaginary world

- So… $H_{c(v)}$ contains a copy of $\widetilde{H} \setminus H_{c(v)}$???
- What does σ do to this copy???

Correctness Proof

- <u>Claim:</u> σ maps the copy of $\widetilde{H} \setminus H_{c(v)}$ inside $H_{c(v)}$ to a new copy of $\widetilde{H} \setminus H_{c(v)}$ inside v's imaginary world
  - Can't re-use the old copy
  - Can't spill out
- By induction…

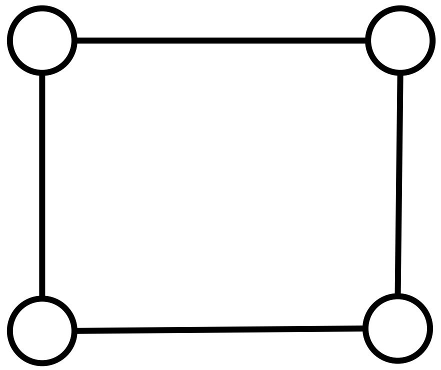


v's imaginary world

- So… $H_{c(v)}$ contains a copy of $\widetilde{H} \setminus H_{c(v)}$???
- What does σ do to this copy???

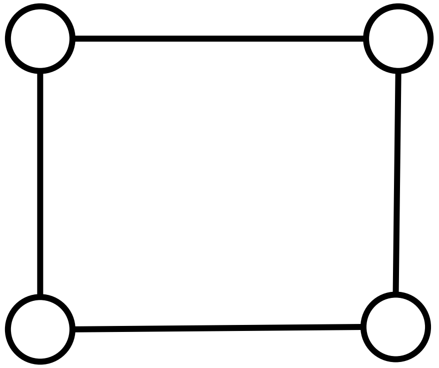
Reduction #2: Edge Replacement - Reminder

- Take a 4-cycle graph



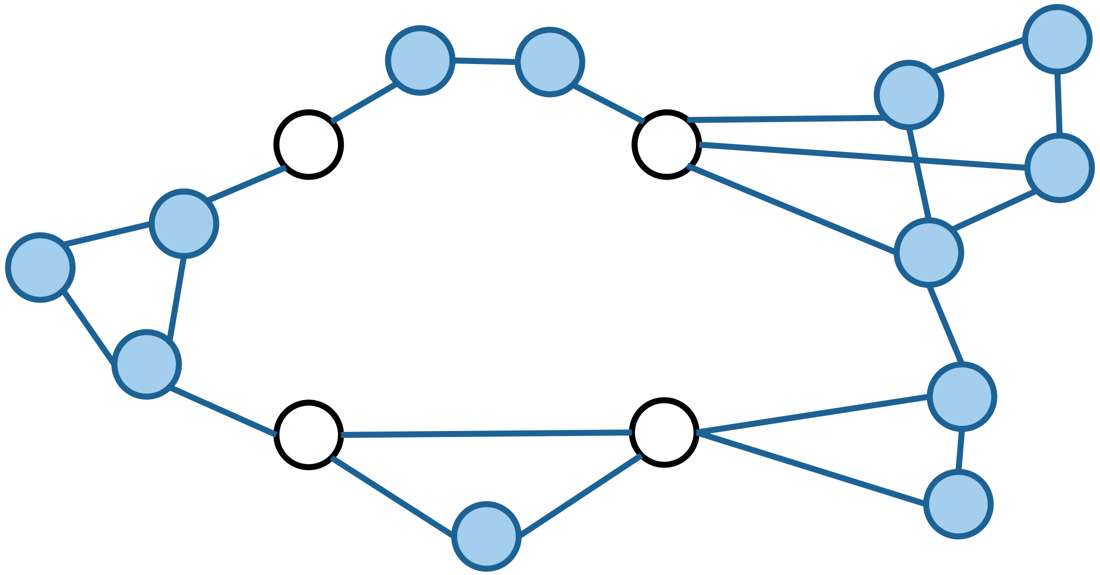
Reduction #2: Edge Replacement - Reminder

- Take a 4-cycle graph
- Replace each edge by an arbitrary graph such that the whole graph is a 2-connected graph



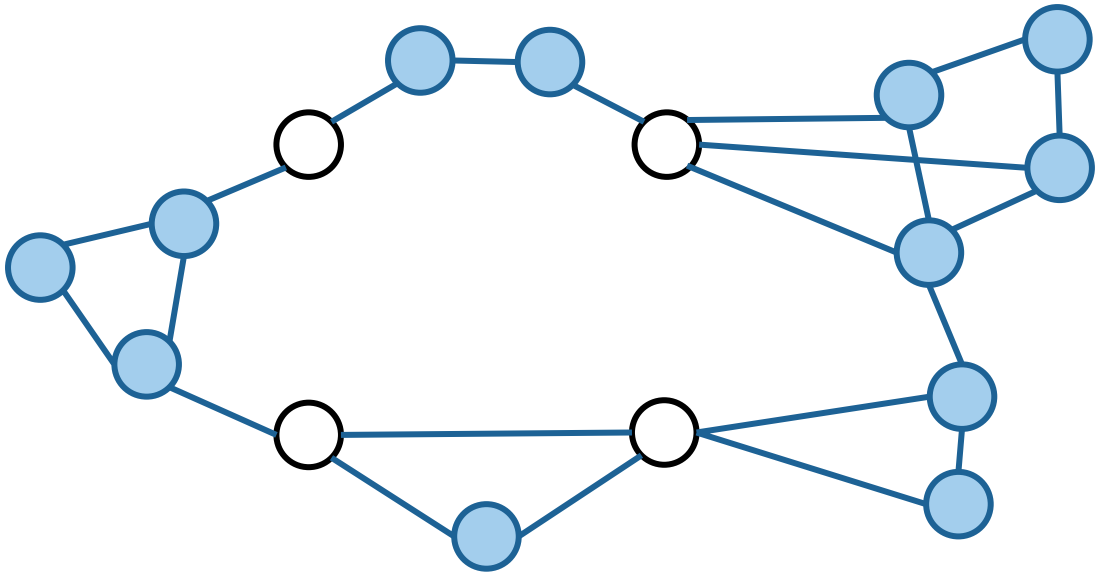
Reduction #2: Edge Replacement - Reminder

- Take a 4-cycle graph
- Replace each edge by an arbitrary graph such that the whole graph is a 2-connected graph



Reduction #2: Edge Replacement - Reminder

- Take a 4-cycle graph
- Replace each edge by an arbitrary graph such that the whole graph is a 2-connected graph
- We call this class of graphs class B

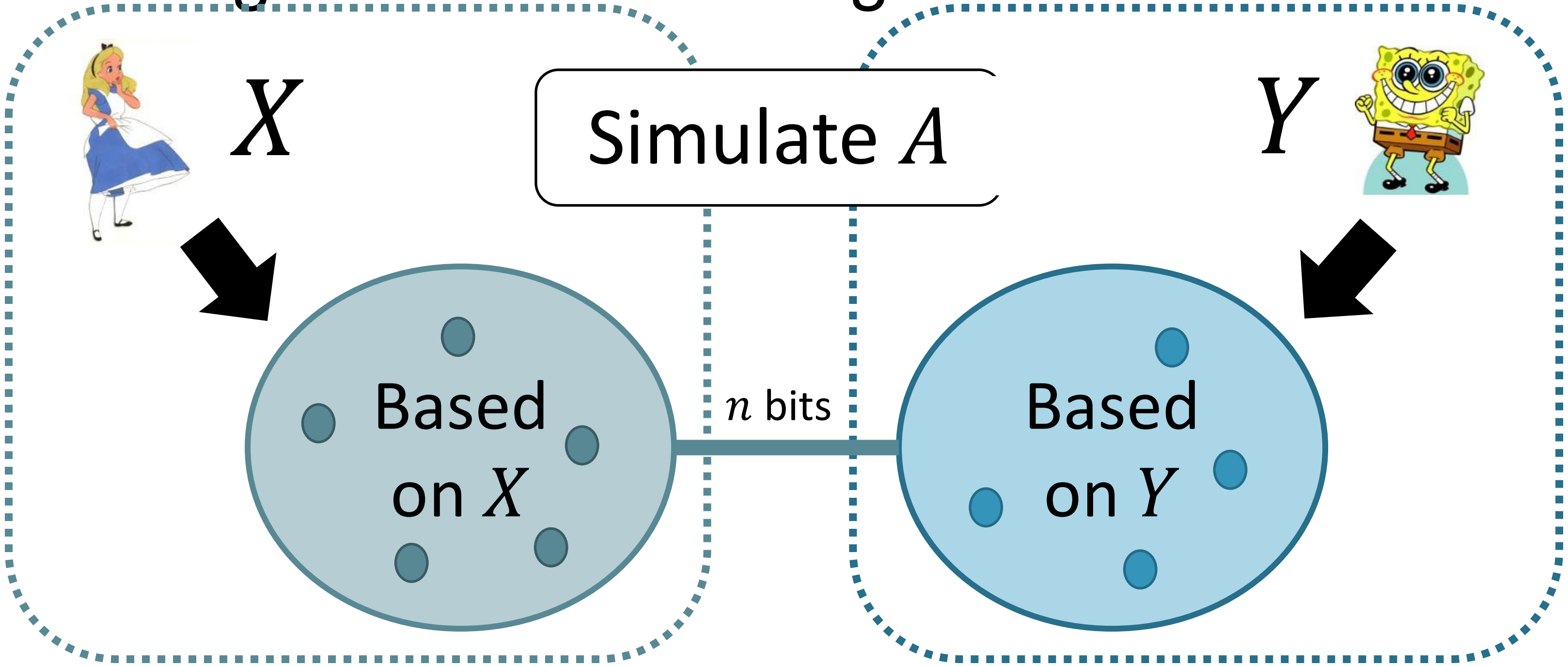


2-Party Communication Complexity

Applying Two-Player Communication Complexity Lower Bounds

Textbook reduction [Kushilevitz-Nisan]:

Given algorithm A for solving task T…



Solution for T ⇨ answer for DISJOINTNESS