Techniques for Counting Problems, Lecture 8
Limitations of Counting Dichotomies

Philip Wellnitz
Graph Homomorphism

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \to G)$ for the set of all graph hom’s from $H$ to $G$. 

$\Phi = $ bipartite $H$ $|V(H)| = 4$ $G$
Graph Homomorphism

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \to G)$ for the set of all graph hom’s from $H$ to $G$. 

$\Phi = \text{bipartite}$ $H$ $|V(H)| = 4$ $G$
Graph Homomorphism

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \rightarrow G)$ for the set of all graph hom’s from $H$ to $G$. 

$\Phi = \text{bipartite}$

$|V(H)| = 4$

$G$

$H$
Graph Homomorphism
Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \to G)$ for the set of all graph hom’s from $H$ to $G$.

$\Phi =$ bipartite $H$ \mid $|V(H)| = 4$ $G$

$\#\text{Hom}(H \to G) = 16$
Graph Homomorphism

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \rightarrow G)$ for the set of all graph hom’s from $H$ to $G$. 

$\Phi = \text{bipartite}$ $|V(H)| = 4$ $G$
Graph Homomorphism
Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \rightarrow G)$ for the set of all graph hom’s from $H$ to $G$.

No homomorphisms from $H$ to $G$. 
**Graph Homomorphism**

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \rightarrow G)$ for the set of all graph hom’s from $H$ to $G$.

Finding (counting) homomorphisms is important for finding patterns in graphs.
**Graph Homomorphism**

Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \to G)$ for the set of all graph hom's from $H$ to $G$. 

$\Phi = \text{bipartite } H (G\text{-colored}) \quad |V(H)| = 4$
Summary: Counting Graph Homomorphisms

**Graph Homomorphism**
Mapping from graph $H$ to $G$ that preserves edges;
Write $\text{Hom}(H \to G)$ for the set of all graph hom’s from $H$ to $G$.

Finding (counting) homomorphisms generalizes graph coloring problems
\[ \text{Hom}(H \rightarrow G) \]

Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$. 
\textbf{Hom}(H \rightarrow G)

Given graphs $H \in \mathcal{H}$ and $G \in \mathcal{G}$, check if there is a graph homomorphism from $H$ to $G$. 

\textbf{Graph classes}
Summary: Counting Graph Homomorphisms

**Hom**(\(H \rightarrow G\))

Given graphs \(H \in \mathcal{H}\) and \(G \in \mathcal{G}\), check if there is a graph hom from \(H\) to \(G\).
**Summary: Counting Graph Homomorphisms**

\[ \text{Hom}(H \rightarrow G) \]

Given graphs \( H \in H \) and \( G \in G \), check if there is a graph hom from \( H \) to \( G \).

**Graph classes**
- All Graphs (\( \top \))
- All Bipartite Graphs
- All Cliques

---

**Techniques for Counting Problems, Lecture 8**
**Summary: Known Results**

\[ \text{Hom}(H \rightarrow G) \]

Given graphs \( H \in H \) and \( G \in G \), check if there is a graph hom from \( H \) to \( G \).
**Summary: Known Results**

**$\text{Hom}(H \to G)$**

Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$.

$\text{Hom}(\top \to \{K_3\})$

3-colorable

$\text{Hom}(\top \to \top)$

NP-complete

$\text{Hom}(T \to T)$
\textbf{Hom}(H \rightarrow G)

Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$.

\textbf{NP-complete}

\textbf{Hom}(\top \rightarrow \top)

3-COLORABLE
Given graphs $H \in \mathcal{H}$ and $G \in \mathcal{G}$, check if there is a graph homomorphism from $H$ to $G$.

The problem is NP-complete.

Diagram:

- $\text{Hom}(\top \rightarrow \{K_3\})$
- $\text{Hom}(\top \rightarrow \{\} )$
- $3$-COLORABLE

Summary: Known Results
Summary: Known Results

**Hom(H → G)**

Given graphs $H \in H$ and $G \in G$, check if there is a graph homomorphism from $H$ to $G$.

Are there fast algorithms for special cases of $\text{Hom}(T \rightarrow T)$?
Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$.

What makes $\text{Hom}(T \to T)$ hard?
**Summary: Known Results**

**$\text{Hom}(H \to G)$**

Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$.

<table>
<thead>
<tr>
<th>$\text{Hom}(T \to G)$</th>
<th>$G$ contains only bipartite graphs</th>
<th>$G$ contains a non-bipartite graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>poly-time solvable</td>
<td>[Hell, Nešetřil ’90]</td>
<td>[Hell, Nešetřil ’90]</td>
</tr>
<tr>
<td>NP-complete</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Summary: Known Results

## $\#\text{Hom}(H \to G)$

Given graphs $H \in \mathbb{H}$ and $G \in \mathbb{G}$, count all graph homomorphisms from $H$ to $G$.

<table>
<thead>
<tr>
<th>$#\text{Hom}(T \to G)$</th>
<th>poly-time solvable</th>
<th>$#\text{P}$-complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>(explicit criterion exists)</td>
<td>[Dyer, Greenhill ’00]</td>
<td>(explicit criterion exists)</td>
</tr>
</tbody>
</table>
Summary: Known Results

**Hom\( (H \rightarrow G) \)**

Given graphs \( H \in H \) and \( G \in G \), check if there is a graph hom from \( H \) to \( G \).

What about the other side, \( \text{Hom}(H \rightarrow T) \)?
$\text{Hom}(H \to G)$

Given graphs $H \in H$ and $G \in G$, check if there is a graph hom from $H$ to $G$.

When is $\text{Hom}(H \to T)$ easy?
Summary: Known Results

**\text{Hom}(H \rightarrow G)**

Given graphs \( H \in H \) and \( G \in G \), check if there is a graph hom from \( H \) to \( G \).

**When is \text{Hom}(H \rightarrow T) easy?**

Always in time \( O(|V(G)|^{|V(H)|}) \) (brute-force)

(fast if \(|V(H)|\) bounded for all \( H \in H \), this is the boring case)
Given graphs $H \in \mathbb{H}$ and $G \in \mathbb{G}$, check if there is a graph homomorphism from $H$ to $G$.

When is $\text{Hom}(H \rightarrow T)$ fixed-parameter tractable?

(in $O(f(|V(H)|) \cdot poly(|V(G)|)$ time)
**Summary: Known Results**

**Hom\((H \rightarrow G)\)**

Parameter: \(|V(H)|\)

Given graphs \(H \in H\) and \(G \in G\), check if there is a graph hom from \(H\) to \(G\).

<table>
<thead>
<tr>
<th>FPT</th>
<th>W[1]-hard</th>
</tr>
</thead>
<tbody>
<tr>
<td>((f(</td>
<td>V(H)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hom((H \rightarrow T))</th>
<th>“(H) contains only graphs with small treewidth”</th>
<th>“(H) contains graphs with arbitrary large tw”</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Grohe ’03]</td>
<td></td>
<td>[Grohe ’03]</td>
</tr>
</tbody>
</table>
### Known Results

| $\#\text{Hom}(H \to G)$ | Parameter: $|V(H)|$ |
|--------------------------|---------------------|
| **Given graphs** $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$. | |

<table>
<thead>
<tr>
<th>$#\text{Hom}(H \to T)$</th>
<th>FPT</th>
<th>$#W[1]$-hard</th>
</tr>
</thead>
<tbody>
<tr>
<td>“$H$ contains only graphs with small treewidth”</td>
<td>$(f(</td>
<td>V(H)</td>
</tr>
<tr>
<td>[Dalmau, Jonsson '04]</td>
<td>“$H$ contains a graph with large treewidth”</td>
<td>[Dalmau, Jonsson '04]</td>
</tr>
</tbody>
</table>
Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

Complexity dichotomies when restricting either $G$ or $H$. 
#Hom(H → G) Parameter: |V(H)|
Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

Complexity dichotomies when restricting either $G$ or $H$.

What if we restrict both sides?
Summary: Known Results

$\#\text{Hom}(H \to G)$  
Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

Parameter: $|V(H)|$

Complexity dichotomies when restricting either $G$ or $H$.

What if we restrict both sides?

This lecture.
Main Result

**Main Result**

Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

**Parameter: $|V(H)|$**

**Theorem**

For any problem $P$ in $\#W[1]$ (or $w[1]$), there are graph classes $H_P$ and $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$ (or $\text{Hom}(H_P \to G_P)$).
Main Result

**Theorem**

For any problem $P$ in $\#W[1]$ (or $W[1]$), there are graph classes $H_P$ and $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$ (or $\text{Hom}(H_P \rightarrow G_P)$).

- Cannot hope for clear categorization into $\text{FPT}/W[1]$-hard for all pairs $(H, G)$

  (recall Ladner’s Theorem: If $P \neq \text{NP}$, there are NP-intermediate problems; similar results by Downey and Fellows for $\text{FPT}/W[1]$)
**Proof Ideas**

\[ \# \text{Hom}(H \to G) \]

**Parameter:** $|V(H)|$

Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

**Theorem**

For any problem $P$ in $\#W[1]$ (or $W[1]$), there are graph classes $H_P$ and $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$ (or $\text{Hom}(H_P \to G_P)$).
# Homomorphic Counting Problems

**Proof Ideas**

**#Hom(H → G)**

Given graphs $H \in H$ and $G \in G$, count all graph homomorphisms from $H$ to $G$.

**Theorem**

For any problem $P$ in #W[1] (or W[1]), there are graph classes $H_P$ and $G_P$ such that $P$ is equivalent to #Hom($H_P → G_P$) (or Hom($H_P → G_P$)).

**Recall:** #Hom($H → T$) is #W[1]-hard if $H$ has “unbounded treewidth” [DalJon’04]
Proof Ideas

#Hom(H → G)

Given graphs $H \in H$ and $G \in G$, count the number of graph hom's from $H$ to $G$.

Parameter: $|V(H)|$

Theorem

For any $P$ in $\#W[1]$, there are $H_p, G_p$ such that $P$ is equivalent to $\#Hom(H_p \rightarrow G_p)$.

Problem $P$ \hspace{10pt} #W[1]

$\#k$-CLIQUE \hspace{10pt} #W[1]$-hard

$H$ w/ unbounded treewidth \hspace{10pt} #Hom(H \rightarrow T)$
# Hom (H → G)

Given graphs \( H \in \mathcal{H} \) and \( G \in \mathcal{G} \), count the number of graph homomorphisms from \( H \) to \( G \).

**Theorem**

For any \( P \) in \#W[1], there are \( H_p, G_p \) such that \( P \) is equivalent to \#Hom(\( H_p \rightarrow G_p \)).
Proof Ideas

\#Hom(H → G)

Parameter: |V(H)|

Given graphs \( H \in H \) and \( G \in G \), count the number of graph hom's from \( H \) to \( G \).

Theorem

For any \( P \) in \#W[1], there are \( H_p, G_p \) such that \( P \) is equivalent to \#Hom(H_p → G_p).

Problem

\#P

\#W[1]-hard

\#K-CLIQUE

H w/ unbounded treewidth

\#Hom(H → T)
Theorem

For any $P$ in #$W[1]$, there are $H_p, G_p$ such that $P$ is equivalent to $\#\text{Hom}(H_p \rightarrow G_p)$.
**Theorem**

For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.
Theorem

For any $P$ in \#W[1], there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

Problem $P$

Graphs $H_J, G_J$

Instance $J$ of $P$
**Theorem**

For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

**Proof Ideas**

**Problem $P$**

$$\#\text{Hom}(H \to T)$$

**Instance $J$ of $P$**

**Graphs $H_J$, $G_J$**

**Approach:**

$$H_P := \{H_J \mid \text{instance } J \text{ of } P\}$$

$$G_P := \{G_J \mid \text{instance } J \text{ of } P\}$$
Proof Ideas

**Theorem**
For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

Instance $J$ of $P$ \[\xrightarrow{\text{Problem } P} \#\text{Hom}(H \to T)\] \[\xrightarrow{\text{Graphs } H_J, G_J}\]

Approach:

- $H_P := \{H_J \mid \text{instance } J \text{ of } P\}$
- $G_P := \{G_J \mid \text{instance } J \text{ of } P\}$

$P \preceq \#\text{Hom}(H_P \to G_P)$
Theorem

For any $P$ in $\mathbb{W}[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

Instance $J$ of $P$ ↔ Problem $P$ ↔ $\#\text{Hom}(H \rightarrow T)$ ↔ Graphs $H_J$, $G_J$

Approach:

$H_P := \{H_J \mid \text{instance } J \text{ of } P\}$

$G_P := \{G_J \mid \text{instance } J \text{ of } P\}$

$P \leq \#\text{Hom}(H_P \rightarrow G_P) \checkmark$

$\#\text{Hom}(H_P \rightarrow G_P) \leq P$
Proof Ideas

**Theorem**
For any $P$ in $\#W[1]$, there are $H_P, G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

**Approach:**
- $H_P := \{H_J \mid \text{instance } J \text{ of } P\}$
- $G_P := \{G_J \mid \text{instance } J \text{ of } P\}$

$P \leq \#\text{Hom}(H_P \rightarrow G_P)$

$\#\text{Hom}(H_P \rightarrow G_P) \not\leq P$

How do we obtain instance $J$ from $(H_J, G_J)$?
**Theorem**

For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

**Proof Ideas**

- **Instance $J$ of $P$**
- **Problem $P$**
- **Encode**
- **Graphs $H_J, G_J$**
- **Graph $\langle J \rangle$**

**Question:** How do we ensure $\#\text{Hom}(H_J \to G_L \cup \langle L \rangle) = 0$?

(ensure $\#\text{Hom}(H_J \to \langle J \rangle) = 0$)

**Instance $J$ of $P$**

- Graphs $H_J, G_J$
**Theorem**

For any $P$ in $\#W[1]$, there are $H_P, G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

**Proof Ideas**

```plaintext
Instance $J$ of $P$ → Problem $P$ → $\#\text{Hom}(H \rightarrow T)$ → Graphs $H_J, G_J$
```

Approach:

- $H_P := \{H_J \mid \text{instance } J \text{ of } P\}$
- $G_P := \{G_J \cup \langle J \rangle \mid \text{instance } J \text{ of } P\}$
**Theorem**

For any $P$ in $\mathbb{W}[1]$, there are $H_P, G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

Approach:

$$H_P := \{H_J \mid \text{instance } J \text{ of } P\}$$

$$G_P := \{G_J \cup \langle J \rangle \mid \text{instance } J \text{ of } P\}$$

$$P \preceq \#\text{Hom}(H_P \to G_P)$$

(ensure $\#\text{Hom}(H_J \to \langle J \rangle) = 0$)
**Theorem**

For any $P$ in $\text{#W}[1]$, there are $H_P, G_P$ such that $P$ is equivalent to $\text{#Hom}(H_P \rightarrow G_P)$.

**Instance $J$ of $P$**

**Problem $P$**

$\text{#Hom}(H \rightarrow T)$

**Graphs $H_J, G_J$**

**Approach:**

$H_P := \{H_J \mid \text{instance } J \text{ of } P\}$

$G_P := \{G_J \cup \langle J \rangle \mid \text{instance } J \text{ of } P\}$

$P \leq \text{#Hom}(H_P \rightarrow G_P)$

(ensure $\text{#Hom}(H_J \rightarrow \langle J \rangle) = 0$)

$\text{#Hom}(H_P \rightarrow G_P) \leq P$
**Theorem**
For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

**Approach:**
- $H_P := \{H_J \mid \text{instance } J \text{ of } P\}$
- $G_P := \{G_J \cup \langle J \rangle \mid \text{instance } J \text{ of } P\}$

$P \leq \#\text{Hom}(H_P \to G_P)$ (ensure $\text{Hom}(H_J \to \langle J \rangle) = 0$)

$\#\text{Hom}(H_P \to G_P) \leq P$

How do we handle malformed input $(H_J, G_L)$?
Theorem

For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \to G_P)$.

Approach:

$H_P := \{H_J \mid \text{instance } J \text{ of } P\}$

$G_P := \{G_J \cup \langle J \rangle \mid \text{instance } J \text{ of } P\}$

$P \leq \#\text{Hom}(H_P \to G_P)$

(ensure $\#\text{Hom}(H_J \to \langle J \rangle) = 0$)

$\#\text{Hom}(H_P \to G_P) \leq P$

How do we ensure $\#\text{Hom}(H_J \to G_L \cup \langle L \rangle) = 0$?
Theorem

For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

$P \leq \#\text{Hom}(H_P \rightarrow G_P)$

Can solve instance $J$ with $(H_J, G_J \cup \langle J \rangle)$ by computing $\#\text{Hom}(H_J \rightarrow G_J \cup \langle J \rangle)$

(ensuring $\#\text{Hom}(H_J \rightarrow \langle J \rangle) = 0$)

$\#\text{Hom}(H_P \rightarrow G_P) \leq P$

Can extract instance $J$ from pair $(H_J, G_J \cup \langle J \rangle)$

Can extract instance $J$ from pair $(H_J, G_J \cup \langle J \rangle)$

How do we ensure $\#\text{Hom}(H_J \rightarrow G_L \cup \langle L \rangle) = 0$?
Theorem

For any $P$ in $\#W[1]$, there are $H_p, G_p$ such that $P$ is equivalent to $\#\text{Hom}(H_p \rightarrow G_p)$.

- $P \leq \#\text{Hom}(H_p \rightarrow G_p)$
- $\#\text{Hom}(H_p \rightarrow G_p) \leq P$

Can solve instance $J$ with $(H_J, G_J \cup \langle J \rangle)$ by computing $\#\text{Hom}(H_J \rightarrow G_J \cup \langle J \rangle)$

(ensuring $\#\text{Hom}(H_J \rightarrow \langle J \rangle) = 0$)

Can extract instance $J$ from pair $(H_J, G_J \cup \langle J \rangle)$

How do we ensure $\#\text{Hom}(H_J \rightarrow G_L \cup \langle L \rangle) = 0$?

Instance $J$ of $P$ → Problem $P$ → $\#\text{Hom}(H \rightarrow T)$ → Graphs $H_J, G_J$
Theorem
For any $P$ in $\#W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to $\#\text{Hom}(H_P \rightarrow G_P)$.

$P \leq \#\text{Hom}(H_P \rightarrow G_P)$
Can solve instance $J$ with $(H_J, G_J \cup \langle J \rangle)$ by computing $\#\text{Hom}(H_J \rightarrow G_J \cup \langle J \rangle)$
(ensuring $\#\text{Hom}(H_J \rightarrow \langle J \rangle) = 0$)

$\#\text{Hom}(H_P \rightarrow G_P) \leq P$
Can extract instance $J$ from pair $(H_J, G_J \cup \langle J \rangle)$

How do we ensure $\#\text{Hom}(H_J \rightarrow G_L \cup \langle L \rangle) = 0$?

Philip Wellnitz
Techniques for Counting Problems, Lecture 8
Theorem

For any $P$ in #$W[1]$, there are $H_P$, $G_P$ such that $P$ is equivalent to #$\text{Hom}(H_P \to G_P)$.

\[
P \leq #\text{Hom}(H_P \to G_P)
\]

Can solve instance $J$ with $(H_J, G_J \cup \langle J \rangle)$ by computing $#\text{Hom}(H_J \to G_J \cup \langle J \rangle)$ (ensuring $#\text{Hom}(H_J \to \langle J \rangle) = 0$)

\[
#\text{Hom}(H_P \to G_P) \leq P
\]

Can extract instance $J$ from pair $(H_J, G_J \cup \langle J \rangle)$

How do we ensure $#\text{Hom}(H_J \to G_L \cup \langle L \rangle) = 0$?

No homomorphisms
$\text{Hom}(\tilde{H}_j \to G_j)$

$\text{Aut}(K(2\kappa(J) + 3))$

$\langle J, H_j \rangle$

$K(2\kappa(J) + 3)$
### Main Result

| **#Hom(\(H \to G\))** | Parameter: \(|V(H)|\) |
|------------------------|------------------|
| Given graphs \(H \in H\) and \(G \in G\), count all graph homomorphisms from \(H\) to \(G\). |

### Theorem

For any problem \(P\) in \#W[1] (or \(W[1]\)), there are graph classes \(H_P\) and \(G_P\) such that \(P\) is equivalent to \#Hom(\(H_P \to G_P\)) (or Hom(\(H_P \to G_P\))).

- Cannot hope for clear categorization into FPT/\#W[1]-hard for all pairs \((H, G)\)
- Need to look at specific pairs of graph classes
### Main Result

**#Hom** \((H \to G)\)

| Parameter: | \(|V(H)|\) |
|-------------|-------------|

Given graphs \(H \in \mathbb{H}\) and \(G \in \mathbb{G}\), count all graph homomorphisms from \(H\) to \(G\).

**Theorem**

For any problem \(P\) in \#W[1] (or \(W[1]\)), there are graph classes \(H_P\) and \(G_P\) such that \(P\) is equivalent to \#Hom\((H_P \to G_P)\) (or Hom\((H_P \to G_P)\)).

- Cannot hope for clear categorization into FPT/W[1]-hard for all pairs \((H, G)\)
- Need to look at specific pairs of graph classes
TikZ code for Kneser graphs available on GitHub

github.com/PH111P/tikz-kneser

Thank you!