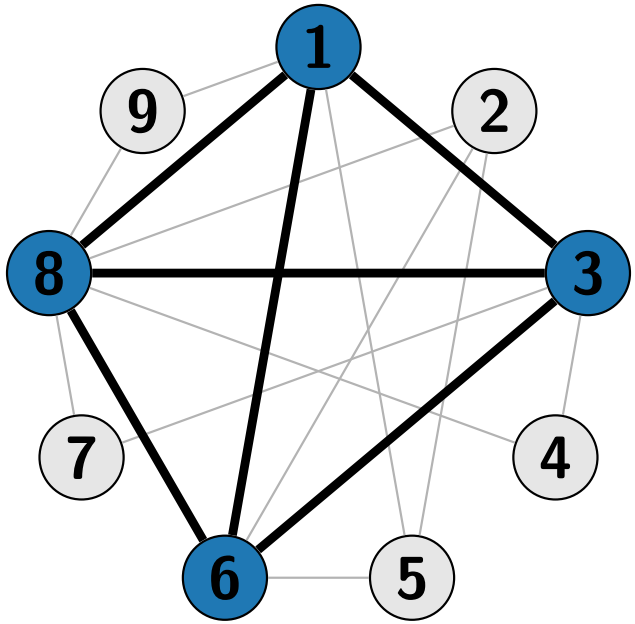
Cliques, Bicliques, Clubs and Colours

Ciaran McCreesh Patrick Prosser

November 19, 2013

Cliques and Colours

The Maximum Clique Problem

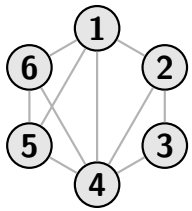


- Determining the size of a maximum clique, denoted ω, is NP-hard.

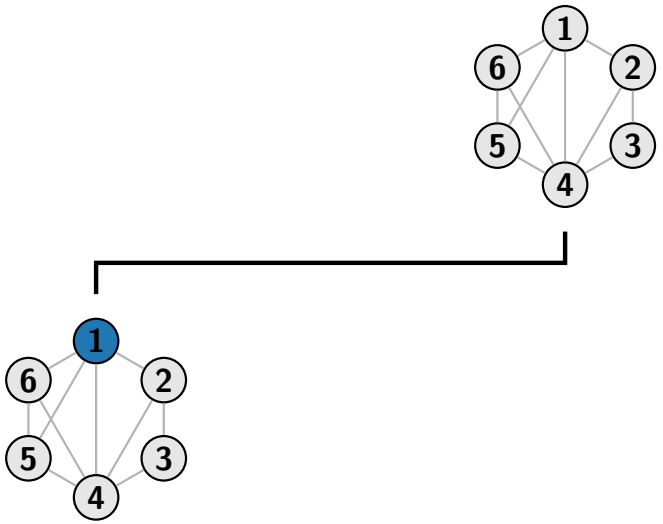
Branch and Bound for Maximum Clique

- Pick a vertex v. Now we *branch*: either v is in a maximum clique...
  - Add v to a growing candidate clique.
  - Discard undecided vertices not adjacent to v.
  - Recurse.
- ...or it is not. Exclude v, and try again until we're out of vertices.
- Keep track of the largest clique found so far (the *incumbent*), and avoid recursing if we cannot beat the incumbent. So we need a *bound* on the size of a maximum clique.

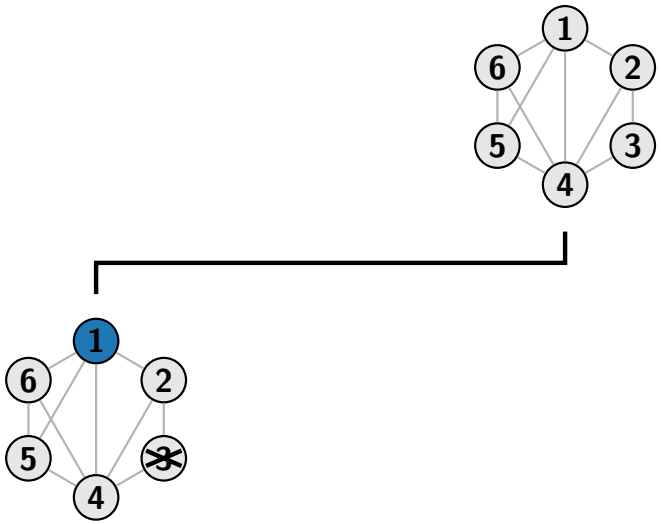
Branch and Bound for Maximum Clique



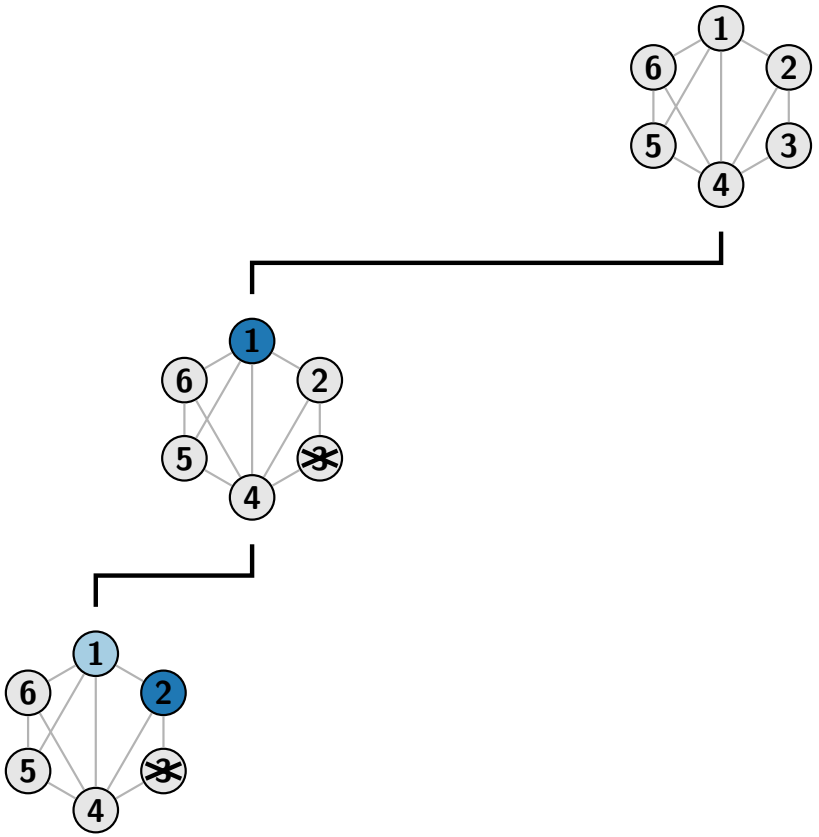
Branch and Bound for Maximum Clique



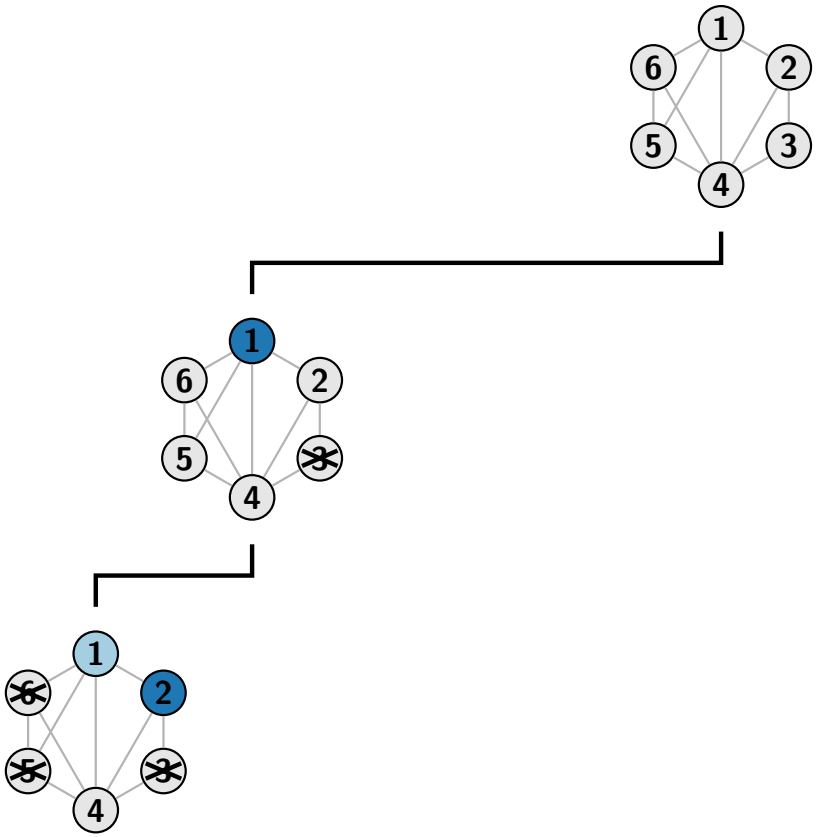
Branch and Bound for Maximum Clique



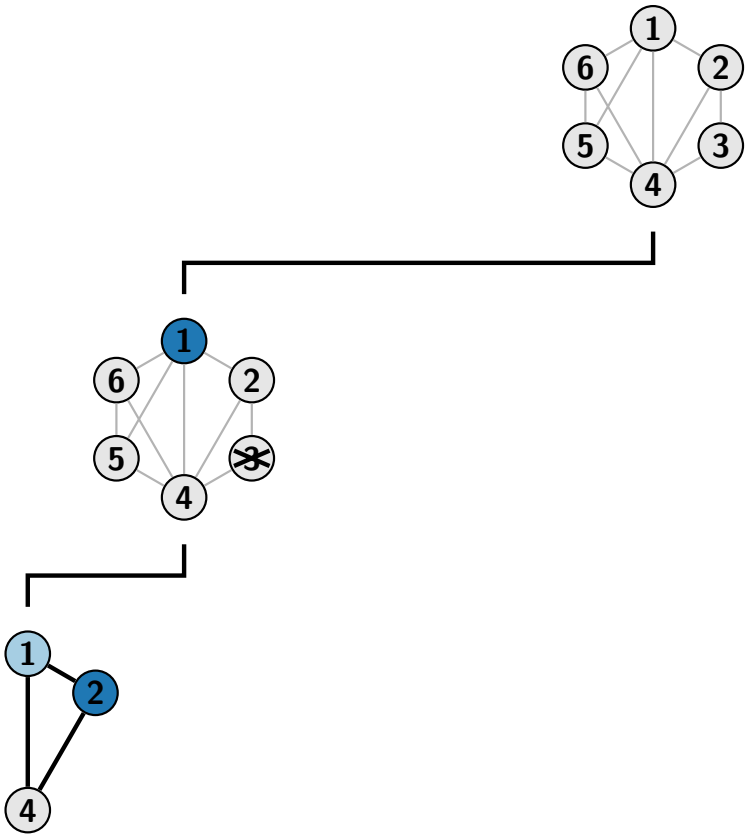
Branch and Bound for Maximum Clique



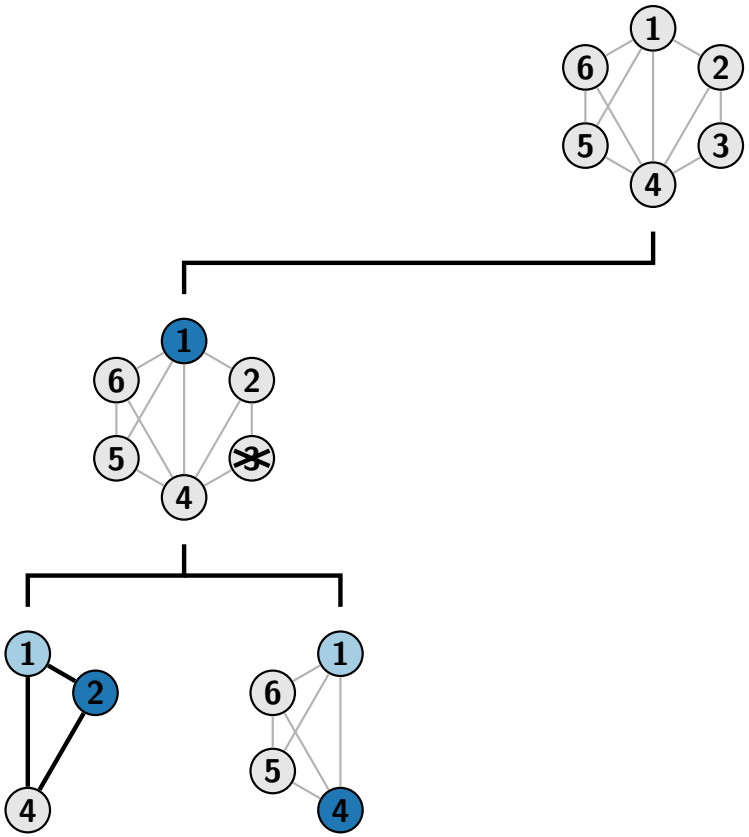
Branch and Bound for Maximum Clique



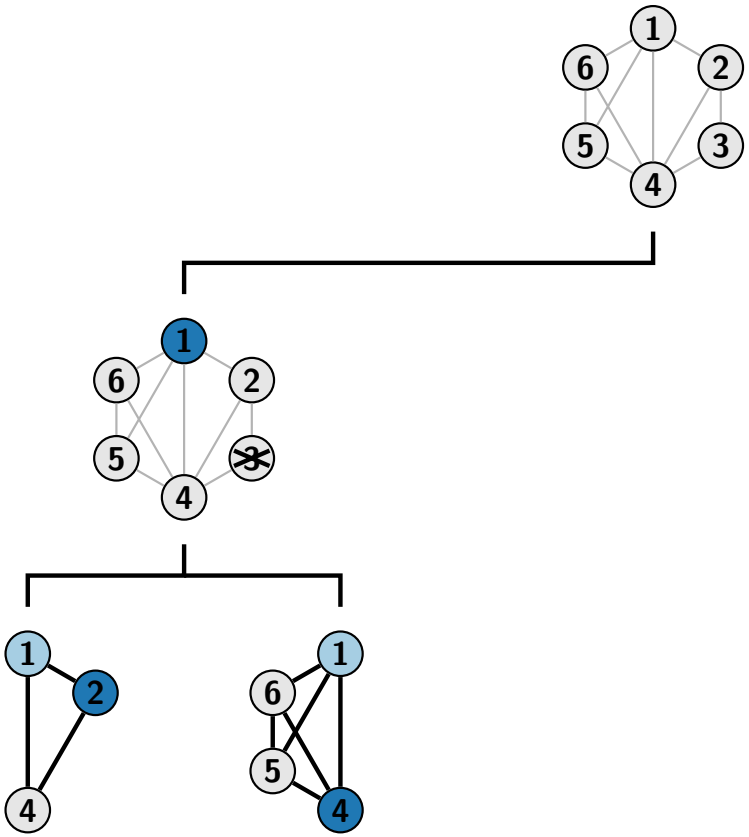
Branch and Bound for Maximum Clique



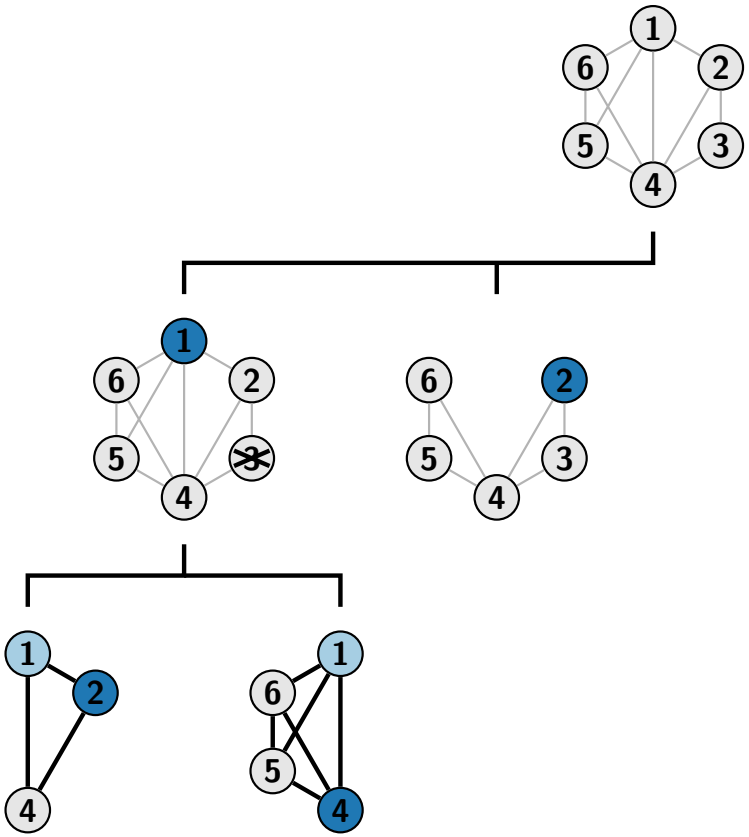
Branch and Bound for Maximum Clique



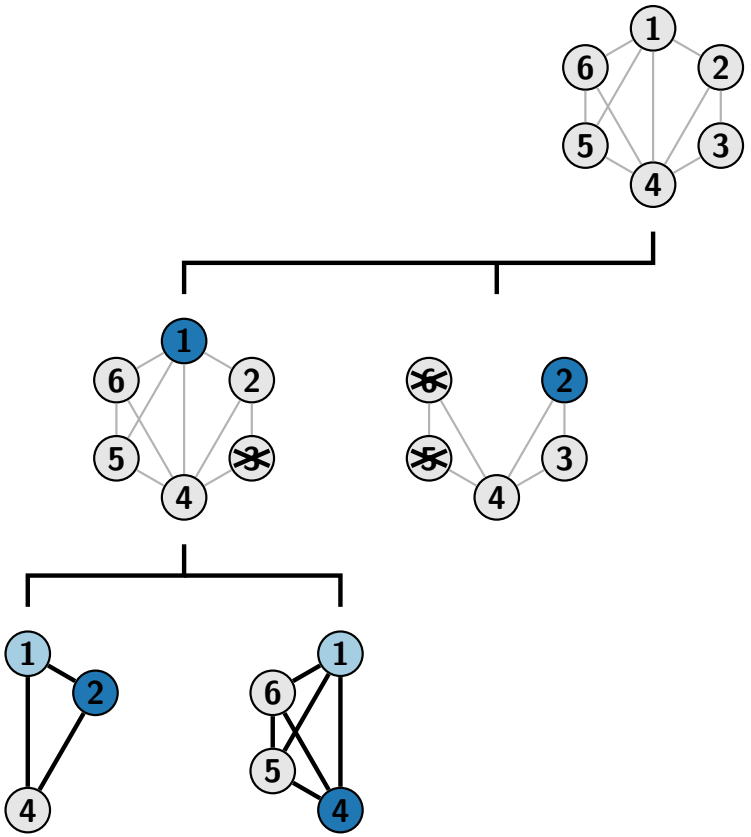
Branch and Bound for Maximum Clique



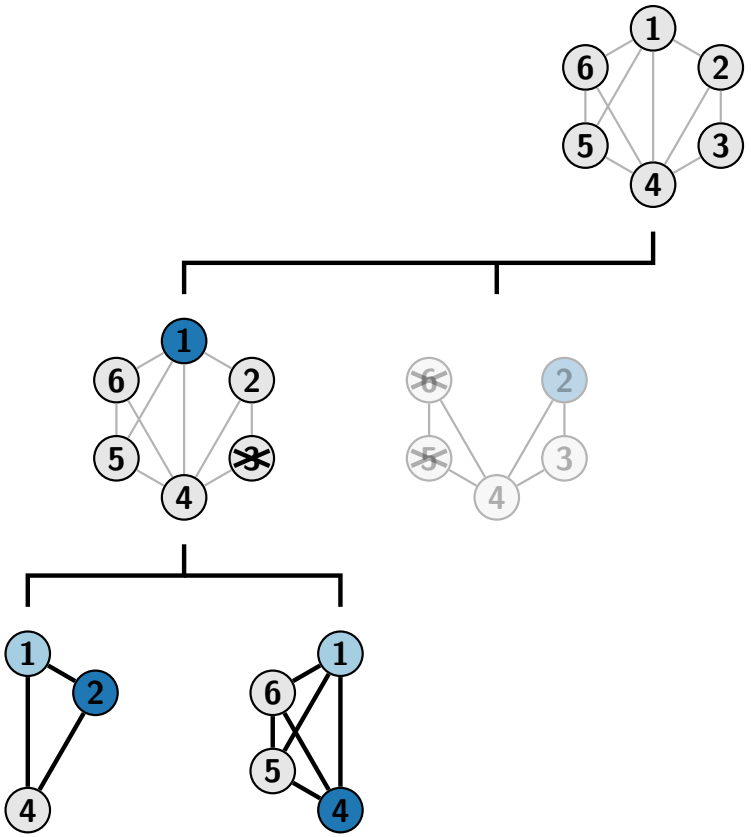
Branch and Bound for Maximum Clique



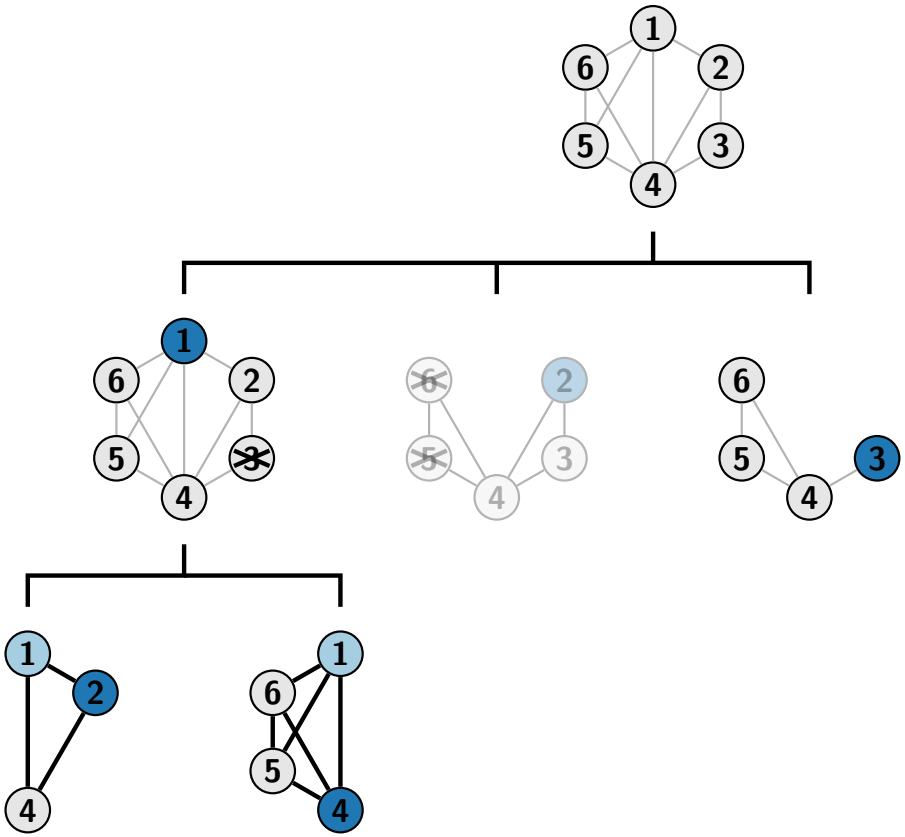
Branch and Bound for Maximum Clique



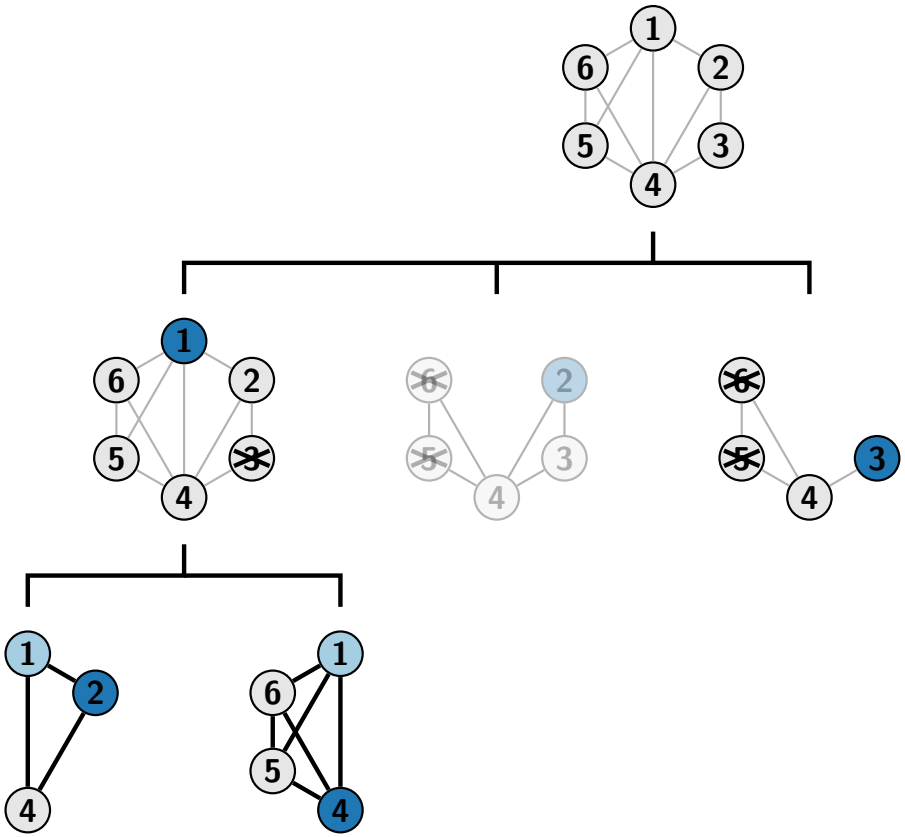
Branch and Bound for Maximum Clique



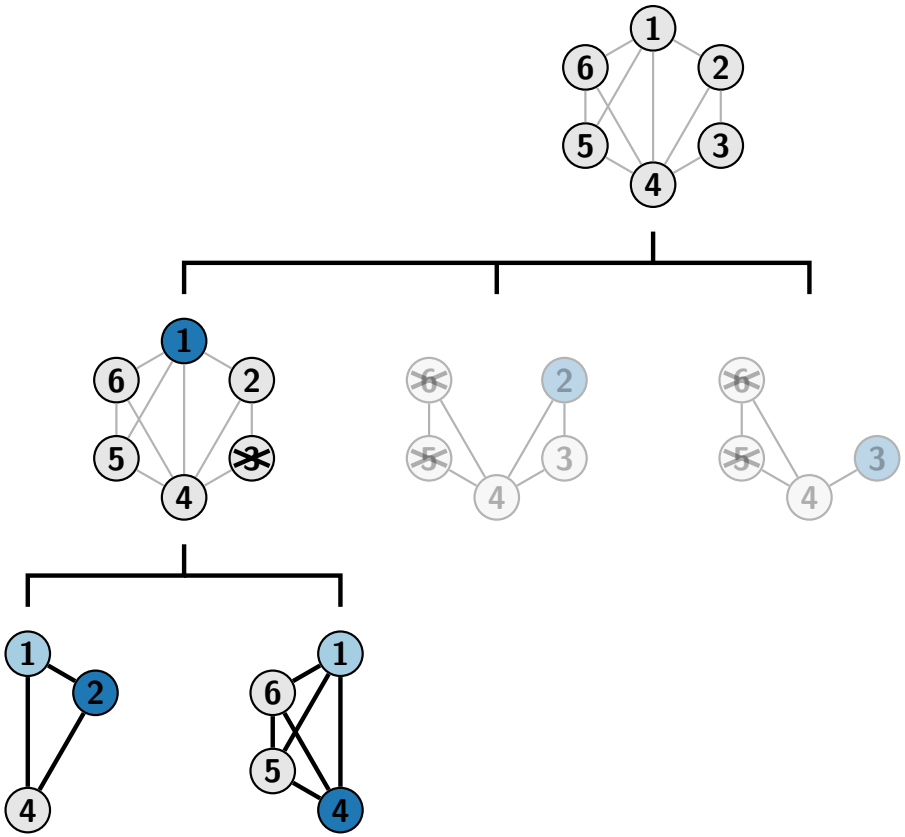
Branch and Bound for Maximum Clique



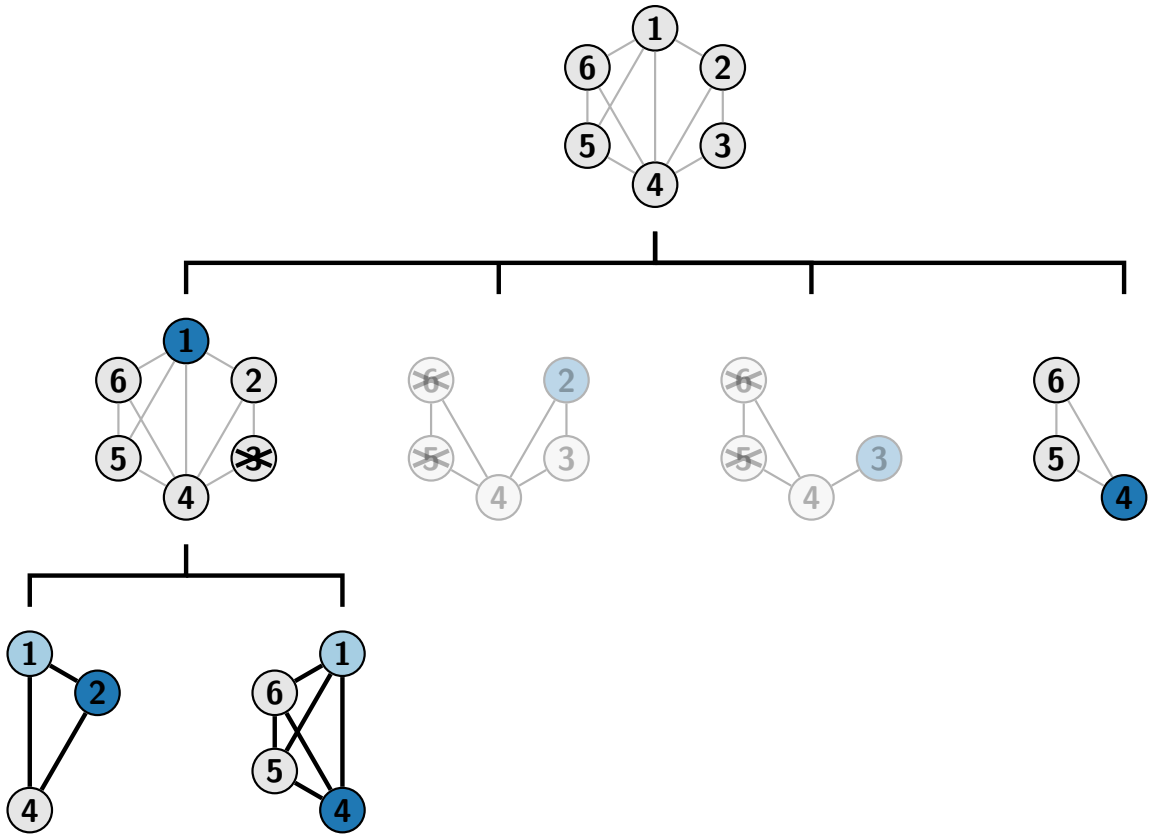
Branch and Bound for Maximum Clique



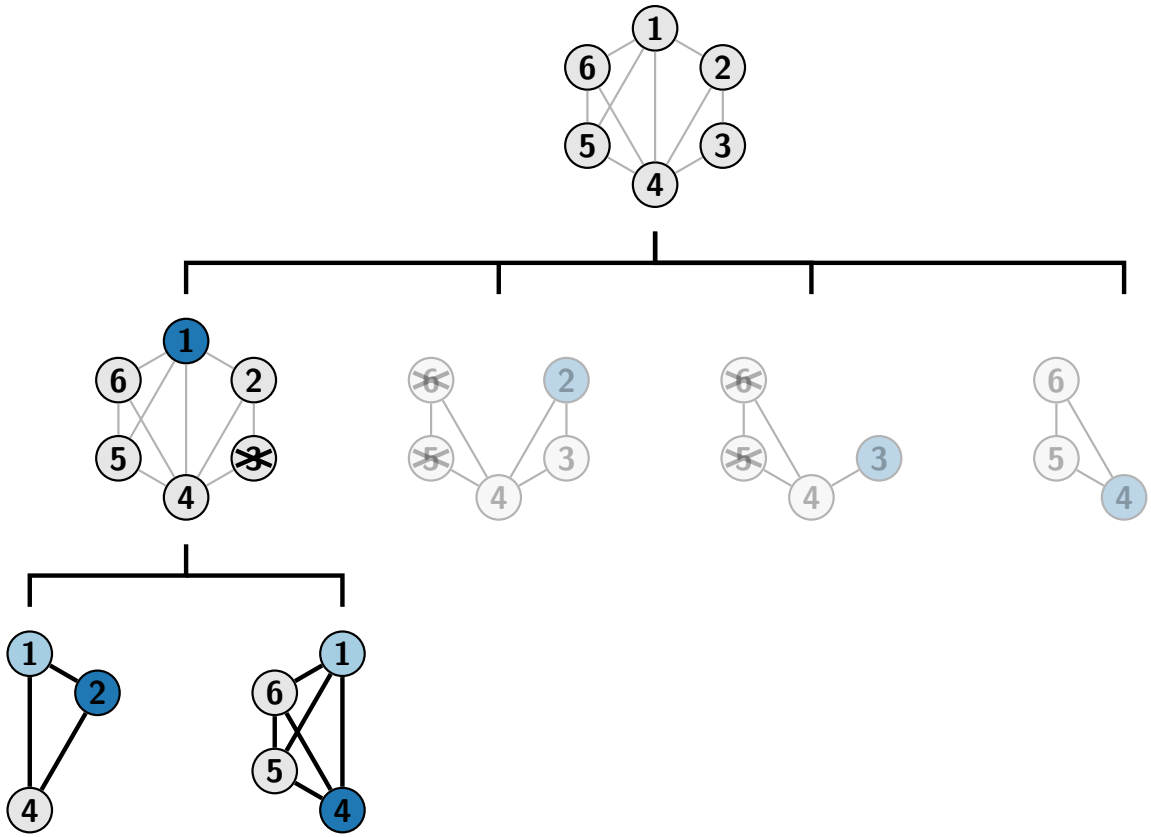
Branch and Bound for Maximum Clique



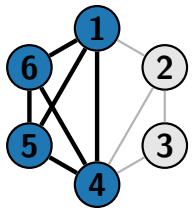
Branch and Bound for Maximum Clique



Branch and Bound for Maximum Clique



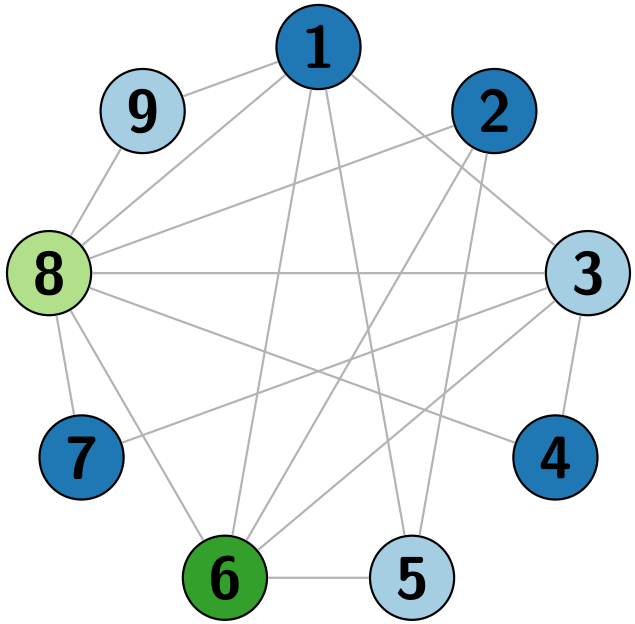
Branch and Bound for Maximum Clique



Bound?

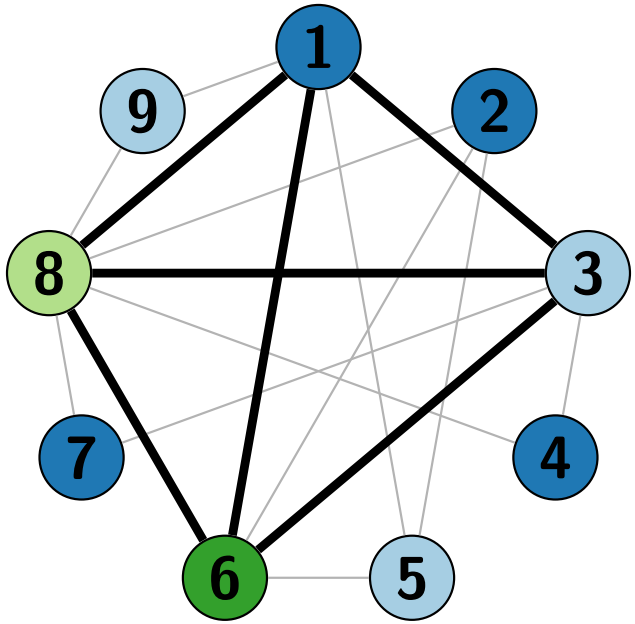
- $\omega \leq |V|$. But this is not very good.

Graph Colouring



- Determining the smallest number of colours needed, denoted χ, is NP-hard.

Graph Colouring

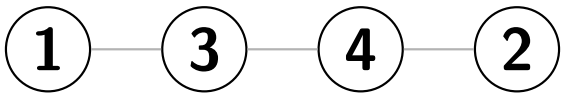


- Determining the smallest number of colours needed, denoted χ, is NP-hard.
- $\omega \leq \chi$. In this case we also know that any clique of size three must contain a green vertex, and any clique of size four must contain both green vertices.

Greedy Colouring

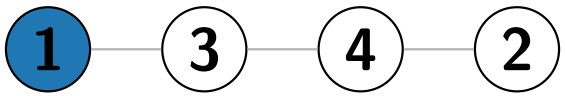
- We can construct a non-optimal colouring in polynomial time by colouring greedily:
  - Give each vertex in turn the first available colour
  - . . . or while there are vertices uncoloured, start a new colour, and assign each remaining vertex in turn that colour if possible.

Greedy Colouring is Bad and Misleading



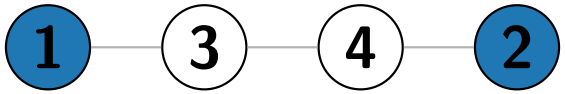
- Greedy colourings can be arbitrarily bad.

Greedy Colouring is Bad and Misleading



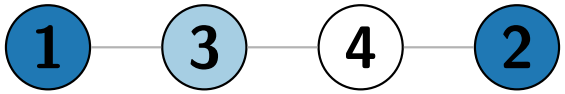
- Greedy colourings can be arbitrarily bad.

Greedy Colouring is Bad and Misleading



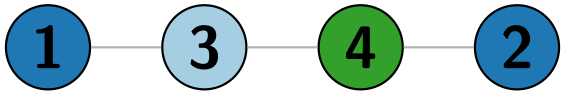
- Greedy colourings can be arbitrarily bad.

Greedy Colouring is Bad and Misleading



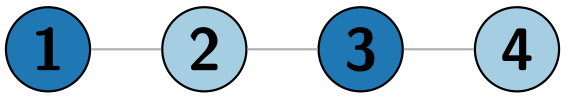
- Greedy colourings can be arbitrarily bad.

Greedy Colouring is Bad and Misleading



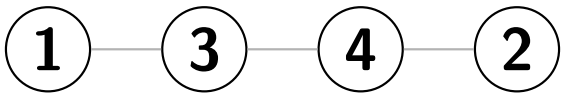
- Greedy colourings can be arbitrarily bad.

Greedy Colouring is Bad and Misleading



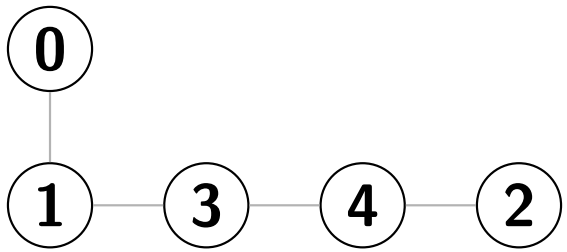
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.

Greedy Colouring is Bad and Misleading



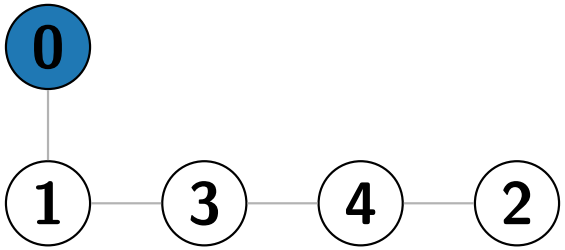
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



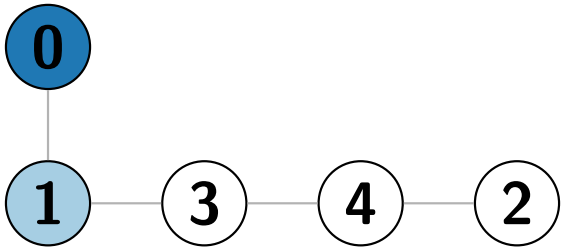
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



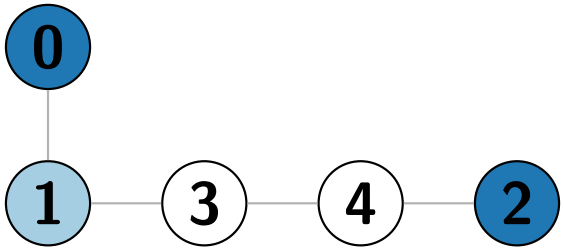
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



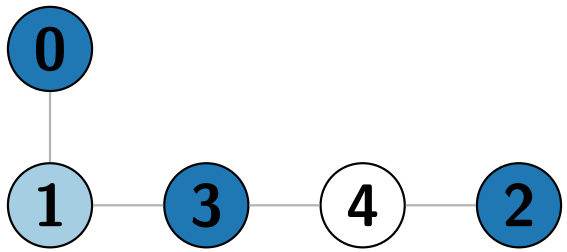
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



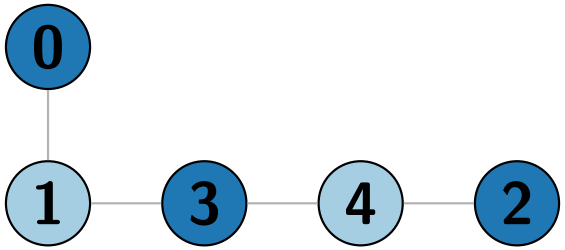
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



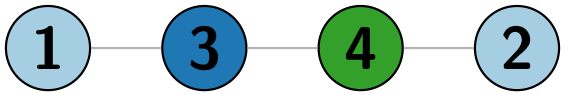
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Greedy Colouring is Bad and Misleading



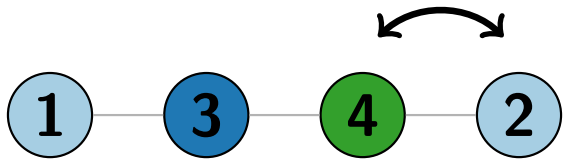
- Greedy colourings can be arbitrarily bad.
- Although there is always an order which will produce an optimal colouring.
- Greedy colourings can get worse on induced subgraphs.

Fixing a Greedy Colouring?



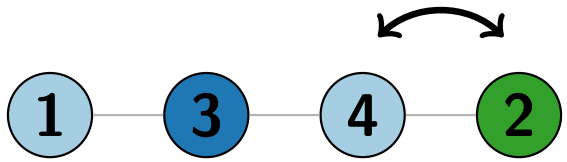
- If we fiddle around with a greedy colouring slightly, we can sometimes improve it.

Fixing a Greedy Colouring?



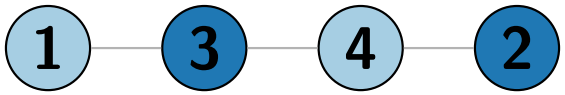
- If we fiddle around with a greedy colouring slightly, we can sometimes improve it.

Fixing a Greedy Colouring?



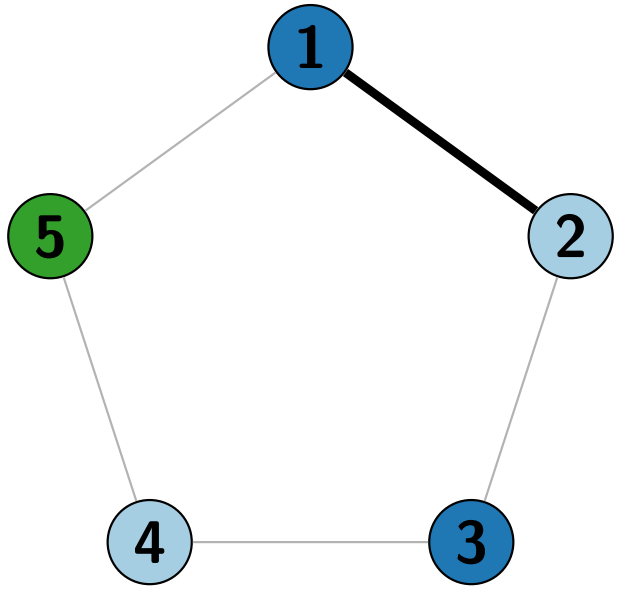
- If we fiddle around with a greedy colouring slightly, we can sometimes improve it.

Fixing a Greedy Colouring?



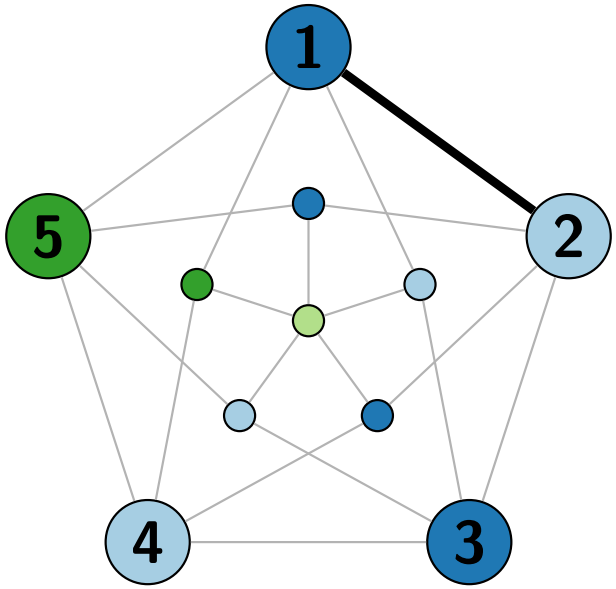
- If we fiddle around with a greedy colouring slightly, we can sometimes improve it.

$\chi - \omega$ Can Be Large



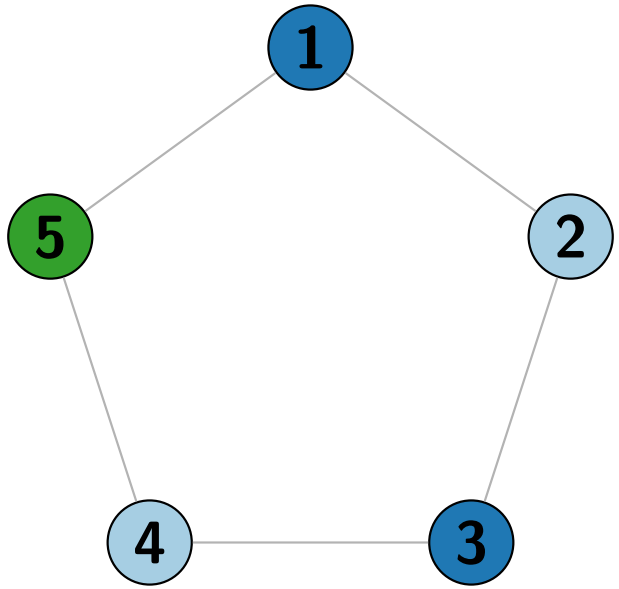
- There are graphs where $\omega < \chi$.

$\chi - \omega$ Can Be Large



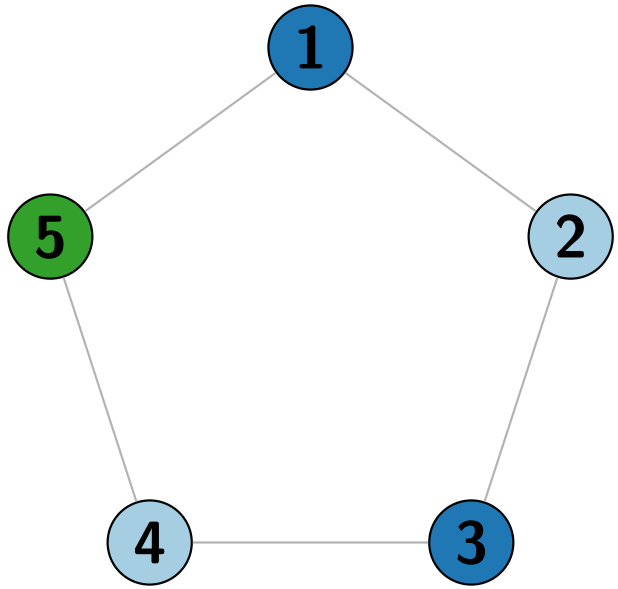
- There are graphs where $\omega < \chi$.
- There are graphs where $\chi - \omega$ is large. Testing whether $\chi - \omega = 1$ is NP-complete.

Measuring the Gap?



- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.

Measuring the Gap?



- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

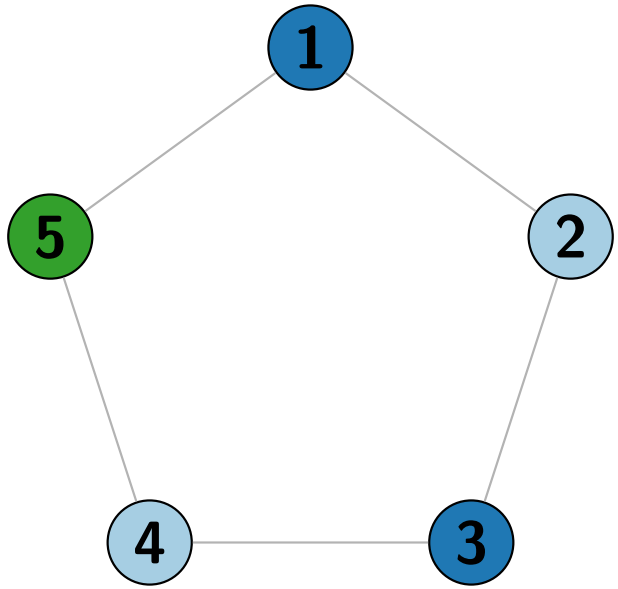
$(\overline{x_1} \vee \overline{x_3})$

$(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_5})$

$(\overline{x_3} \vee \overline{x_5})$



- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

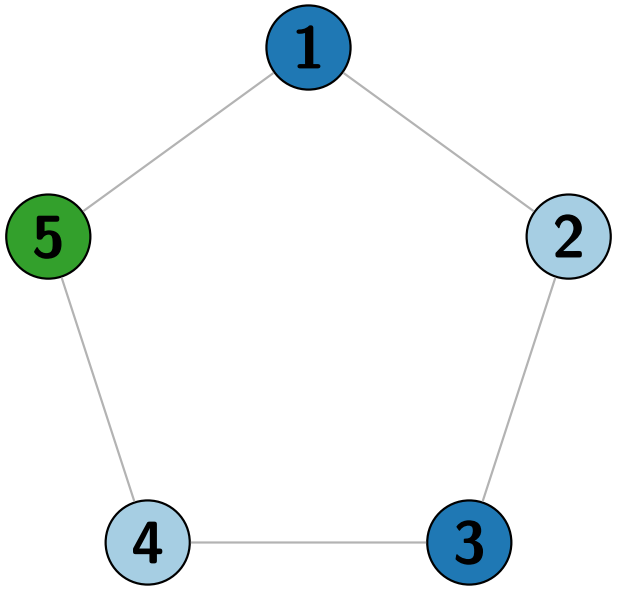
$(\overline{x_1} \vee \overline{x_3})$

$(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_5})$

$(\overline{x_3} \vee \overline{x_5})$



Soft:

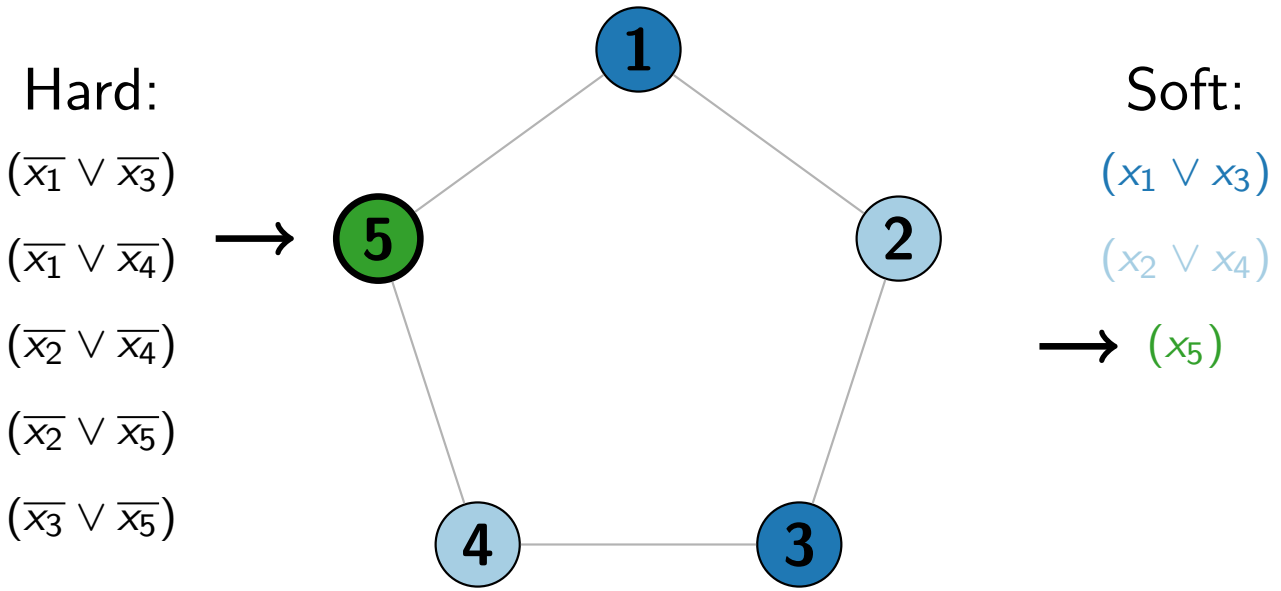
$(x_1 \vee x_3)$

$(x_2 \vee x_4)$

(x_5)

- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?



- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

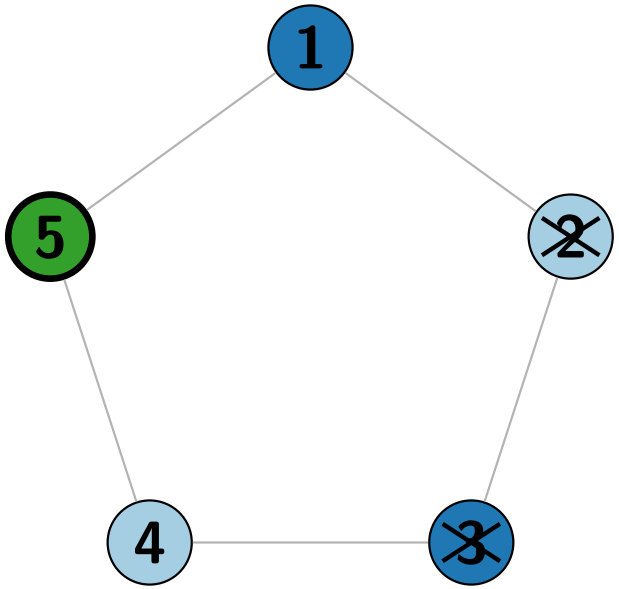
$(\overline{x_1} \vee \overline{x_3})$

$(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$\longrightarrow$ $(\overline{x_2} \vee \overline{x_5})$

$\longrightarrow$ $(\overline{x_3} \vee \overline{x_5})$



Soft:

$(x_1 \vee x_3)$

$(x_2 \vee x_4)$

(x_5)

- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

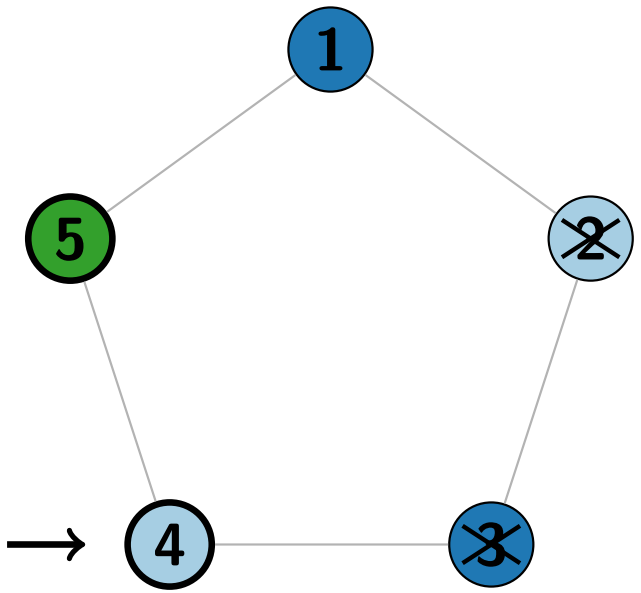
$(\overline{x_1} \vee \overline{x_3})$

$(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_5})$

$(\overline{x_3} \vee \overline{x_5})$



Soft:

$(x_1 \vee x_3)$

$(x_2 \vee x_4)$

(x_5)

- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

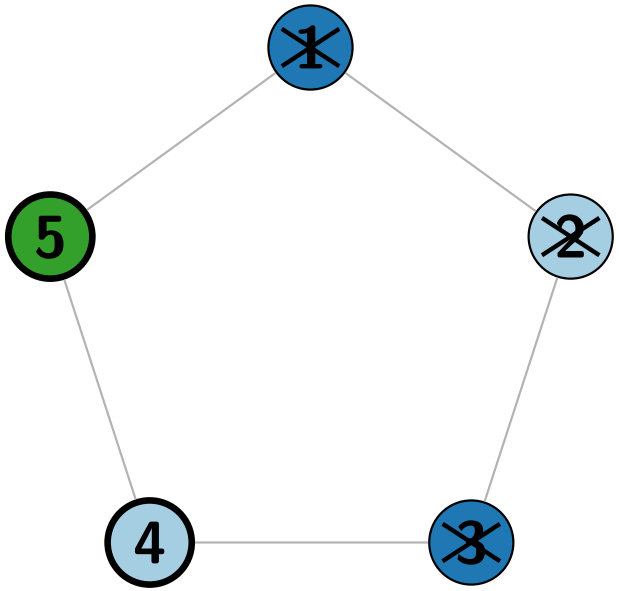
$(\overline{x_1} \vee \overline{x_3})$

$\longrightarrow$ $(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_5})$

$(\overline{x_3} \vee \overline{x_5})$



Soft:

$(x_1 \vee x_3)$

$(x_2 \vee x_4)$

(x_5)

- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Measuring the Gap?

Hard:

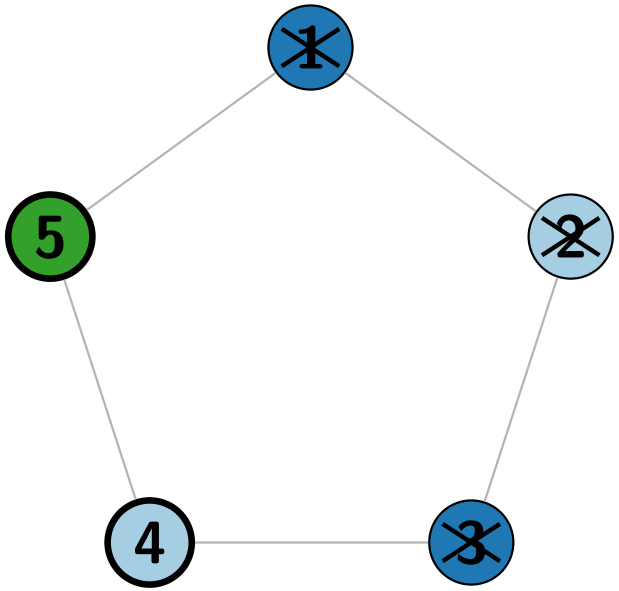
$(\overline{x_1} \vee \overline{x_3})$

$(\overline{x_1} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_4})$

$(\overline{x_2} \vee \overline{x_5})$

$(\overline{x_3} \vee \overline{x_5})$



Soft:

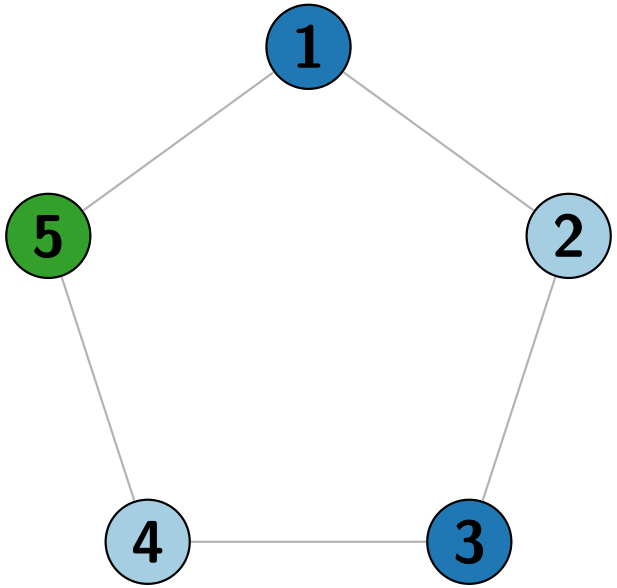
$(x_1 \vee x_3)$ #

$(x_2 \vee x_4)$

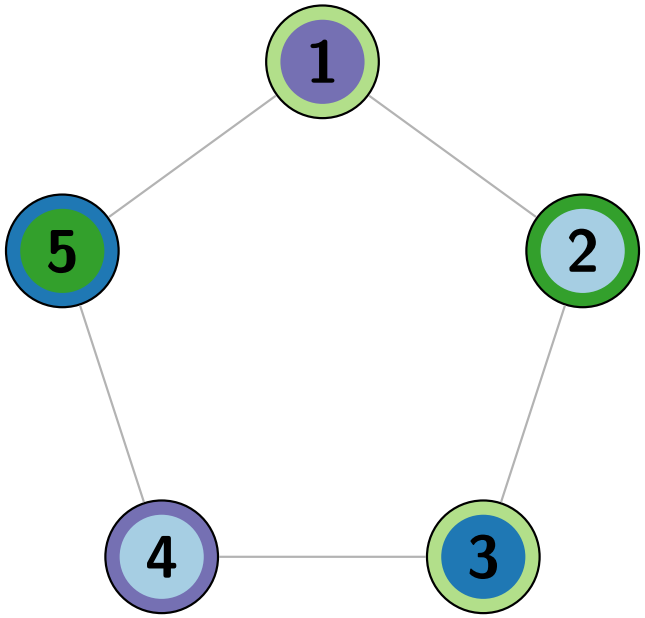
(x_5)

- If we pick the green vertex, we can't also pick both a light blue and a dark blue vertex.
- MaxSAT unit propagation can prove this, giving a lower bound on $\chi - \omega$.

Fractional Colouring?

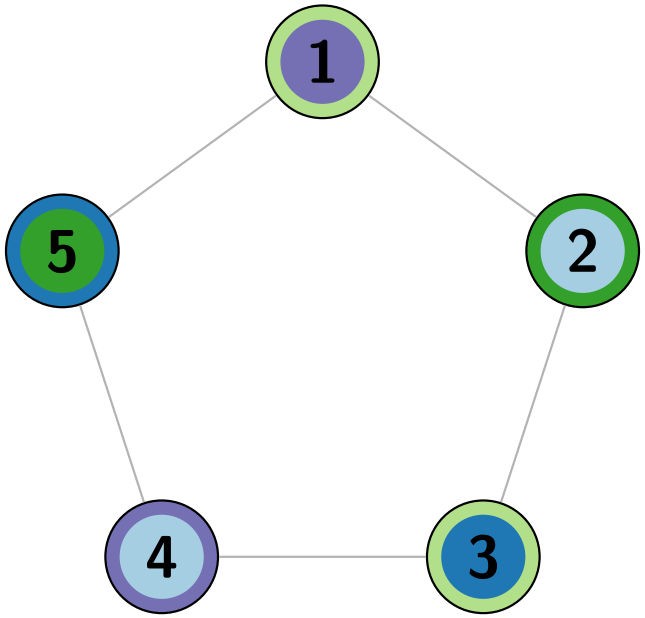


Fractional Colouring?



- If we change the rules, we only need $\frac{5}{2}$ colours.

Fractional Colouring?



- If we change the rules, we only need $\frac{5}{2}$ colours.
- $\omega \leq \chi_f \leq \chi$
- Finding χ_f is NP-hard.

The Lovász Number?

- $\omega \leq \overline{\vartheta} \leq \chi$

The Lovász Number?

- $\omega \leq \overline{\vartheta} \leq \chi$
- And $\overline{\vartheta}$ can be found in polynomial time...

The Lovász Number?

- $\omega \leq \overline{\vartheta} \leq \chi$
- And $\overline{\vartheta}$ can be found in polynomial time...
- ...in theory.

Selected Exact Maximum Clique Algorithms

- Branch and bound, using constructive greedy colouring as a bound and for ordering:
  - Tomita et. al.: MCQ (2003), MCR (2007), MCS (2010)

Selected Exact Maximum Clique Algorithms

- Branch and bound, using constructive greedy colouring as a bound and for ordering:
  - Tomita et. al.: MCQ (2003), MCR (2007), MCS (2010)
- MCR and MCS, using bitset encodings:
  - San Segundo et. al.: BBMC (2011), BBMCI (2011), BBMCW (2013), BBMCL (2013)

Selected Exact Maximum Clique Algorithms

- Branch and bound, using constructive greedy colouring as a bound and for ordering:
  - Tomita et. al.: MCQ (2003), MCR (2007), MCS (2010)
- MCR and MCS, using bitset encodings:
  - San Segundo et. al.: BBMC (2011), BBMCI (2011), BBMCW (2013), BBMCL (2013)
- Computational experiments:
  - Prosser (2012)

Selected Exact Maximum Clique Algorithms

- Branch and bound, using constructive greedy colouring as a bound and for ordering:
  - Tomita et. al.: MCQ (2003), MCR (2007), MCS (2010)
- MCR and MCS, using bitset encodings:
  - San Segundo et. al.: BBMC (2011), BBMCI (2011), BBMCW (2013), BBMCL (2013)
- Computational experiments:
  - Prosser (2012)
- Multi-threaded BBMC variants:
  - McCreesh and Prosser (2013)
  - Depolli et. al. (2013)

Selected Exact Maximum Clique Algorithms

- Branch and bound, using constructive greedy colouring as a bound and for ordering:
  - Tomita et. al.: MCQ (2003), MCR (2007), MCS (2010)
- MCR and MCS, using bitset encodings:
  - San Segundo et. al.: BBMC (2011), BBMCI (2011), BBMCW (2013), BBMCL (2013)
- Computational experiments:
  - Prosser (2012)
- Multi-threaded BBMC variants:
  - McCreesh and Prosser (2013)
  - Depolli et. al. (2013)
- Integrating lower bounds:
  - Batsyn et. al. (2013)

More Exact Maximum Clique Algorithms

- Branch and bound, using fractional colouring:
  - Wood (1997)

More Exact Maximum Clique Algorithms

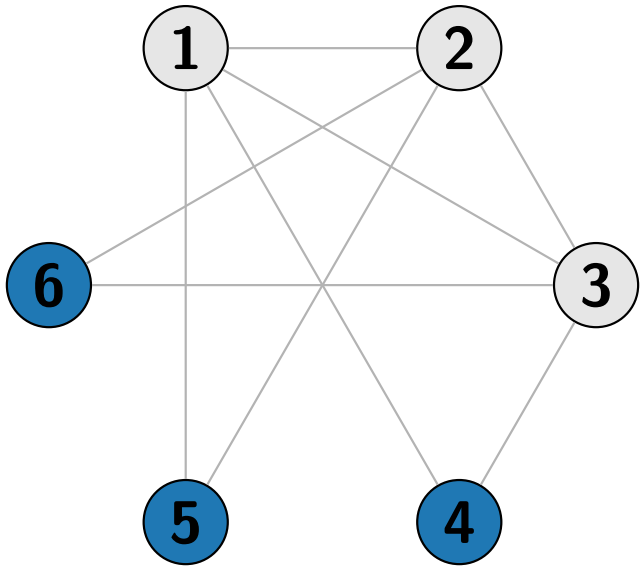
- Branch and bound, using fractional colouring:
  - Wood (1997)
- Constraint programming:
  - Régin (2003)

More Exact Maximum Clique Algorithms

- Branch and bound, using fractional colouring:
  - Wood (1997)
- Constraint programming:
  - Régin (2003)
- Branch and bound, using MaxSAT:
  - Li and Quan: MaxCLQ (2010)

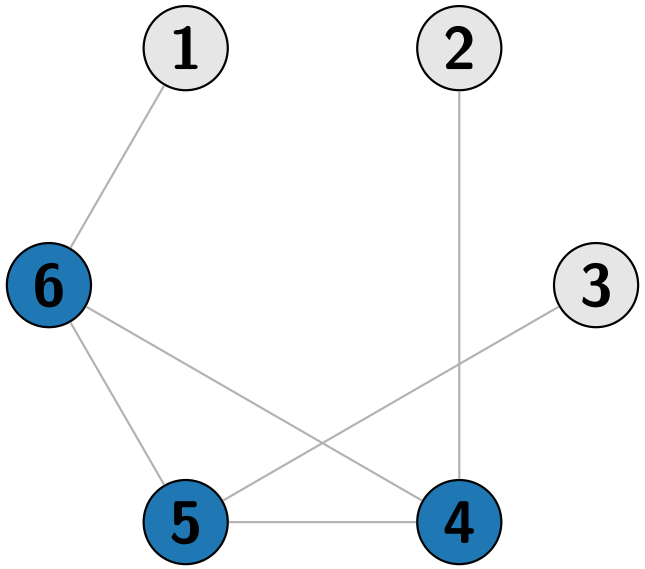
Independent Sets

Maximum Independent Set Algorithms



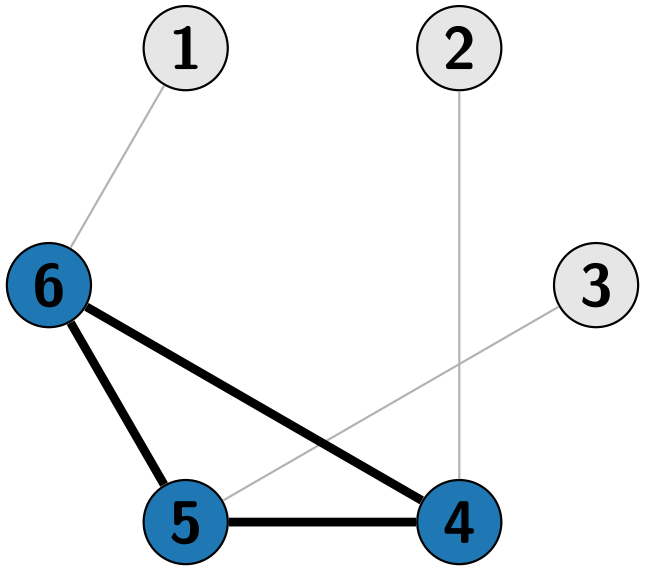
- The size of a maximum independent set is denoted α. Finding α is NP-hard.

Maximum Independent Set Algorithms



- The size of a maximum independent set is denoted α. Finding α is NP-hard.
- A clique in $\overline{G}$ is an independent set in G, so we may reuse our maximum clique algorithms.

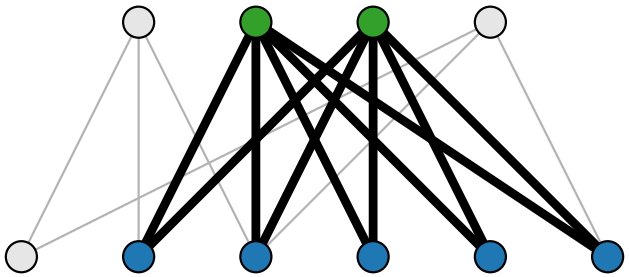
Maximum Independent Set Algorithms



- The size of a maximum independent set is denoted α. Finding α is NP-hard.
- A clique in $\overline{G}$ is an independent set in G, so we may reuse our maximum clique algorithms.
- A colouring of $\overline{G}$ is a *clique cover* of G.

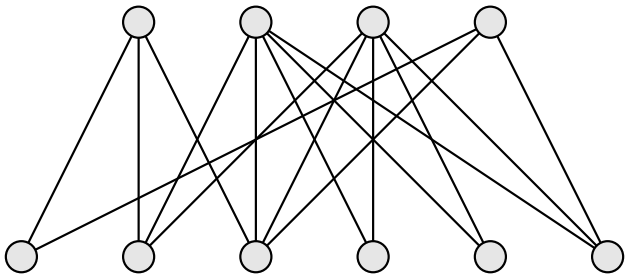
Bicliques

Bicliques in Bipartite Graphs



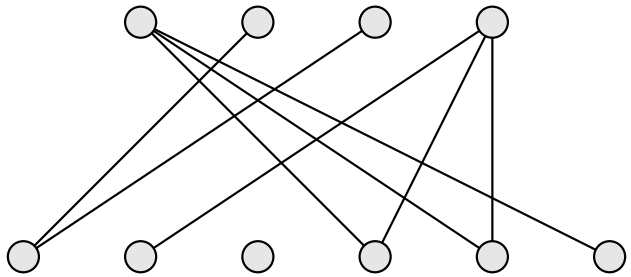
- An *induced biclique* in a graph is two independent sets A and B, and with every vertex in A adjacent to every vertex in B and vice-versa.

Bicliques in Bipartite Graphs



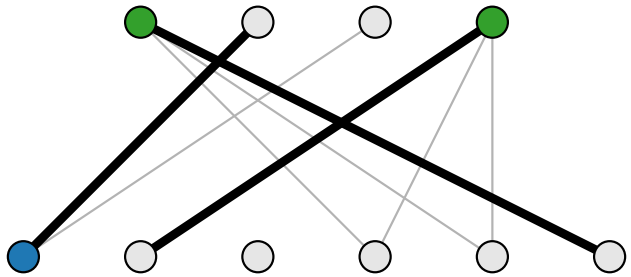
- We can find a maximum vertex biclique in a bipartite graph in polynomial time.

Bicliques in Bipartite Graphs



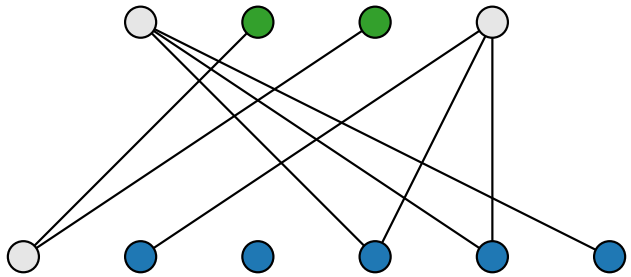
- We start by taking the bipartite complement of the graph, to get a new bipartite graph.

Bicliques in Bipartite Graphs



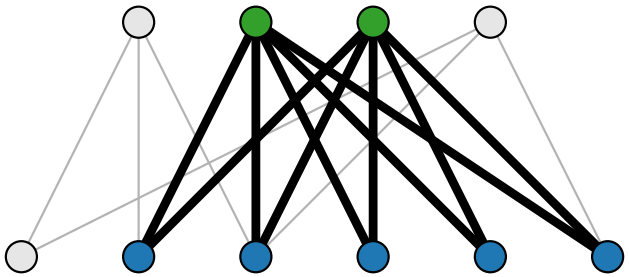
- Then we find a maximum cardinality matching, and use König's Theorem to get a minimum vertex cover.

Bicliques in Bipartite Graphs



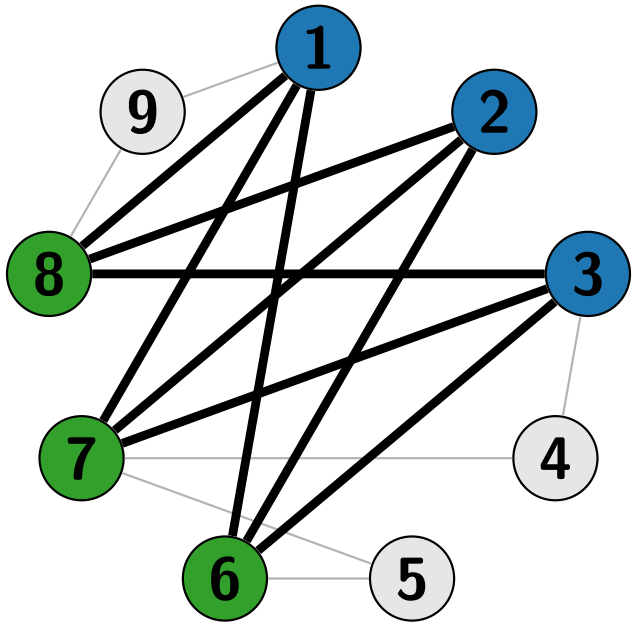
- The vertex set complement of a minimum vertex cover is a maximum independent set.

Bicliques in Bipartite Graphs



- And a maximum independent set in the bipartite complement gives us a maximum biclique in the original graph.

Balanced Induced Bicliques



- An induced biclique is *balanced* if $|A| = |B|$.
- Determining the size of a maximum balanced induced biclique, which we denote $\ddot{\omega}$, is NP-hard.

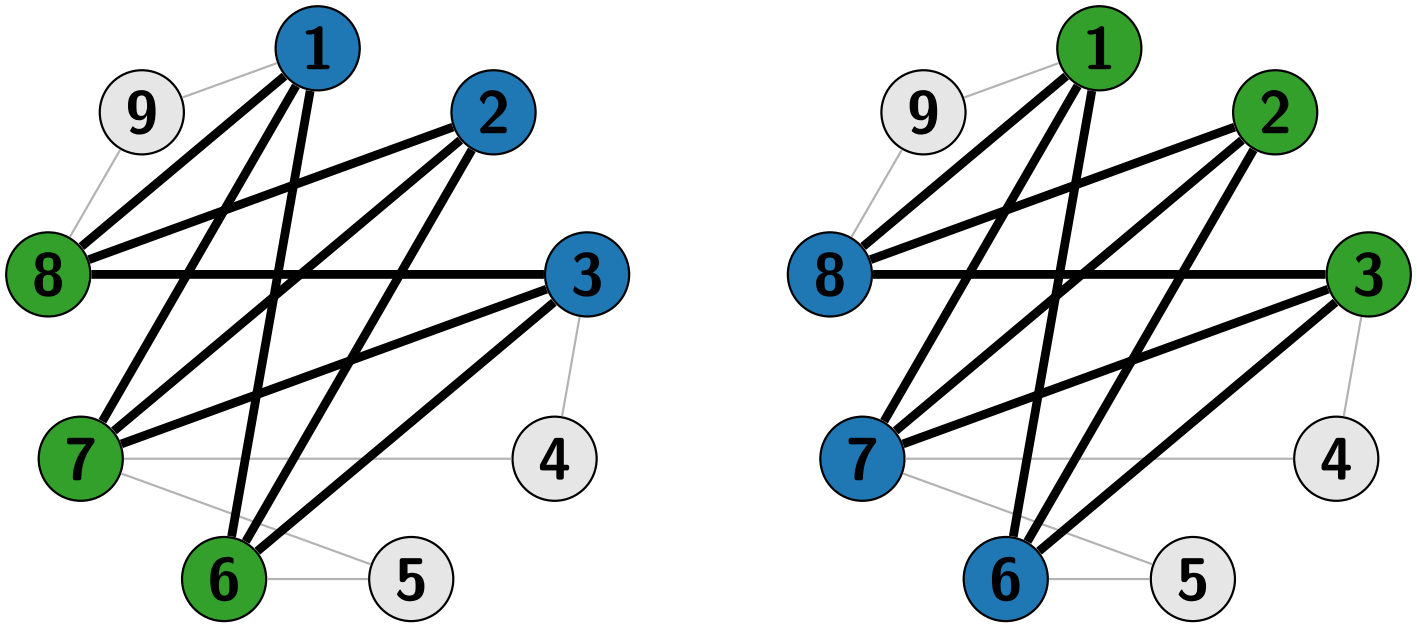
Branch and Bound for Maximum Balanced Biclique

- Grow two independent sets, adding vertices to alternating sides.
- When picking a vertex, throw away adjacent vertices on the same side, and non-adjacent vertices on the other side.

Branch and Bound for Maximum Balanced Biclique

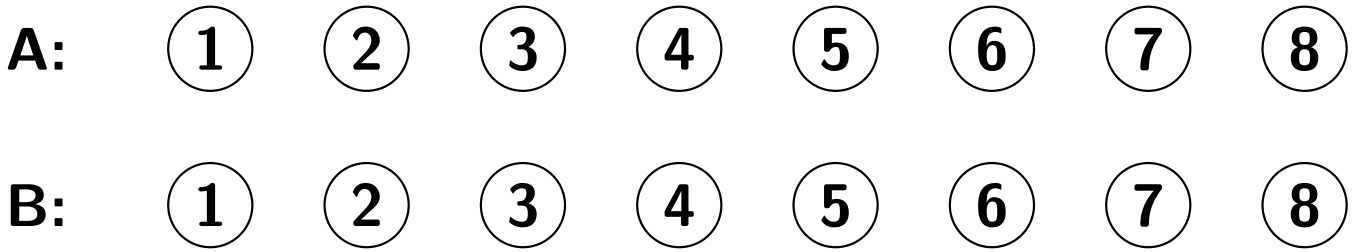
- Grow two independent sets, adding vertices to alternating sides.
- When picking a vertex, throw away adjacent vertices on the same side, and non-adjacent vertices on the other side.
- Be careful with the balance condition: the size of the two independent sets can differ by 1.

Symmetry



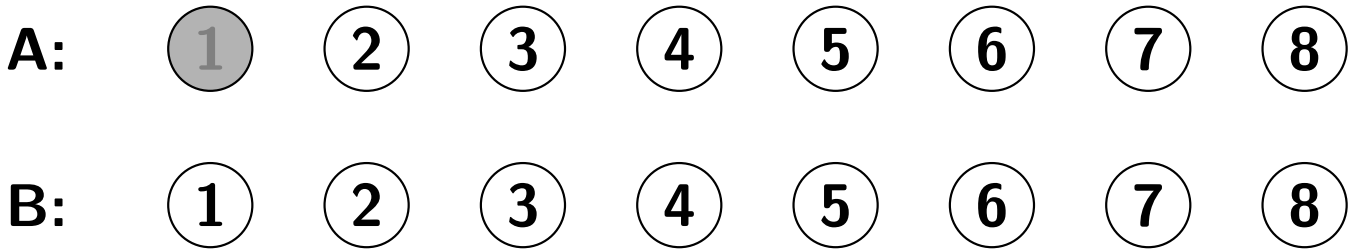
- We are growing an *ordered* pair of sets.
- But (A, B) is a balanced induced biclique iff (B, A) is.

Excluding Symmetries



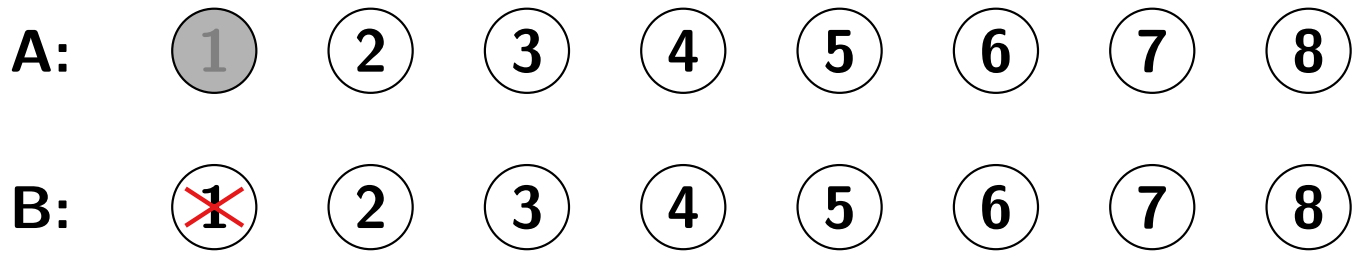
- Use a 0-1 encoding for A and B.

Excluding Symmetries



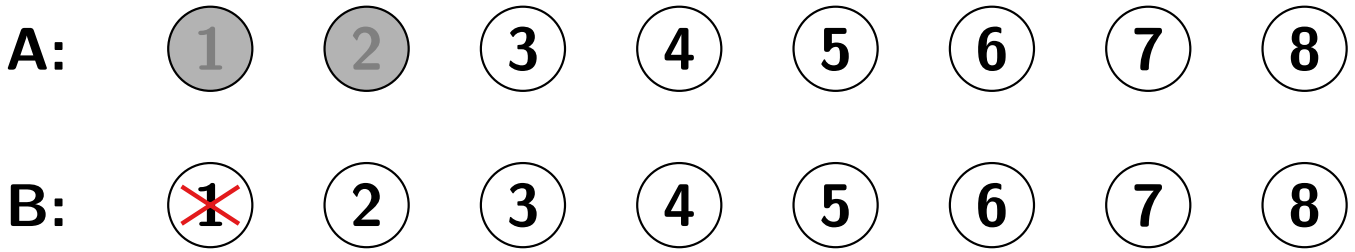
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



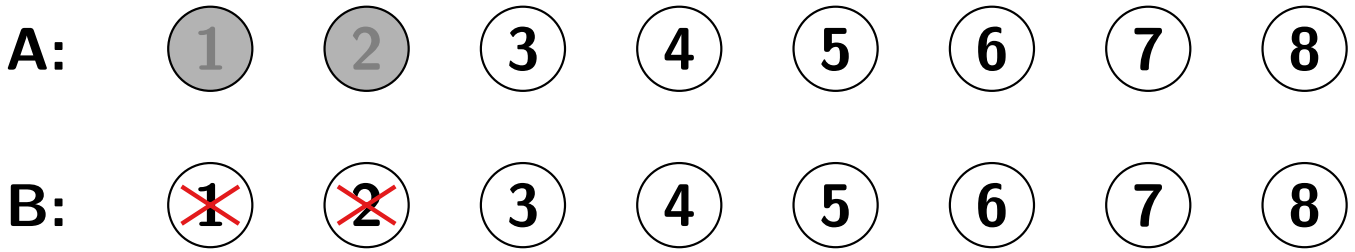
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



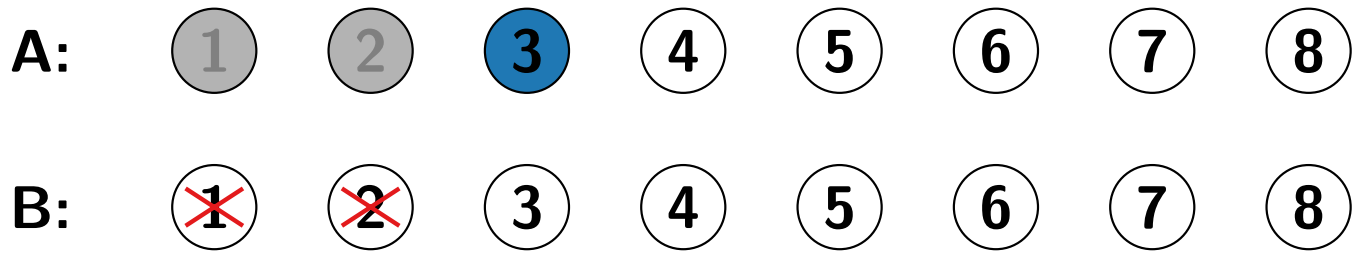
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



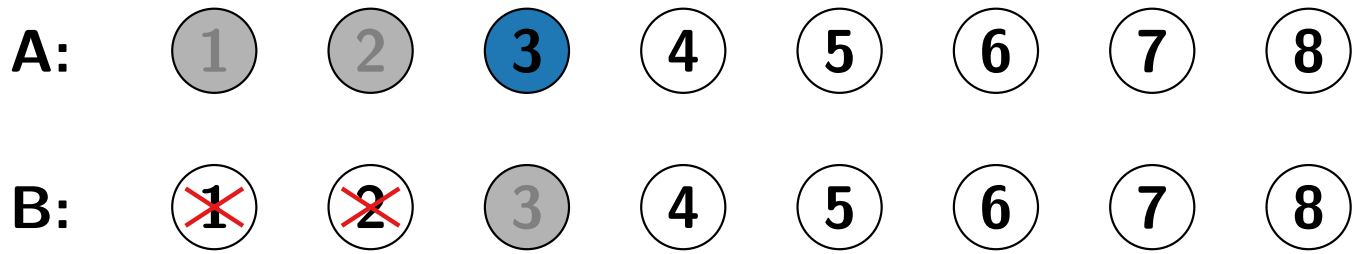
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



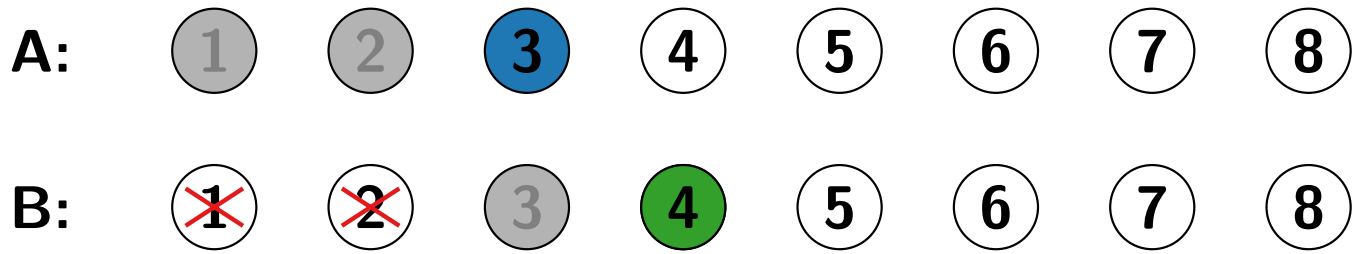
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



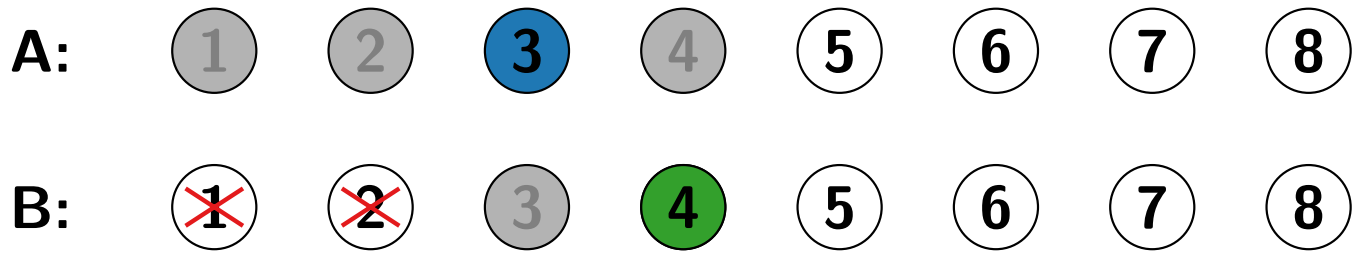
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



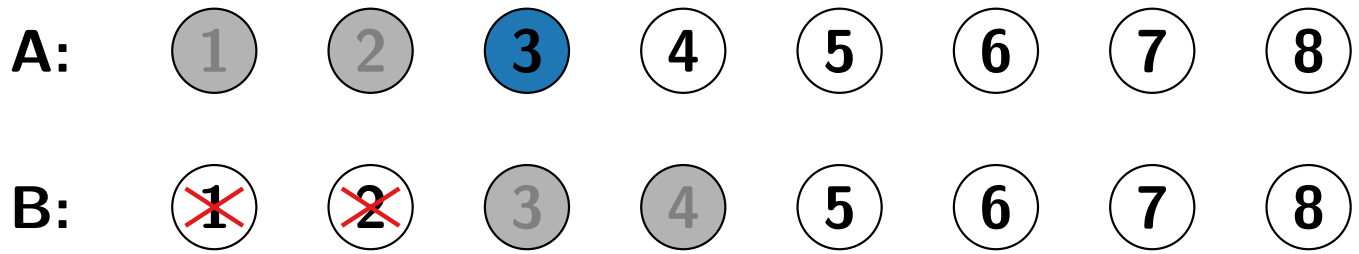
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



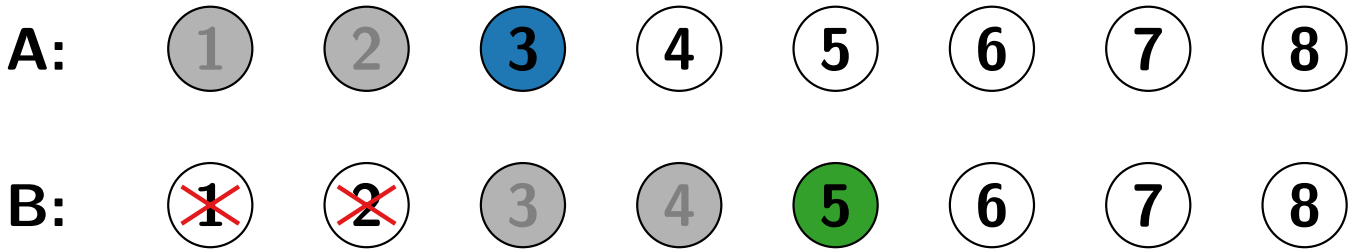
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



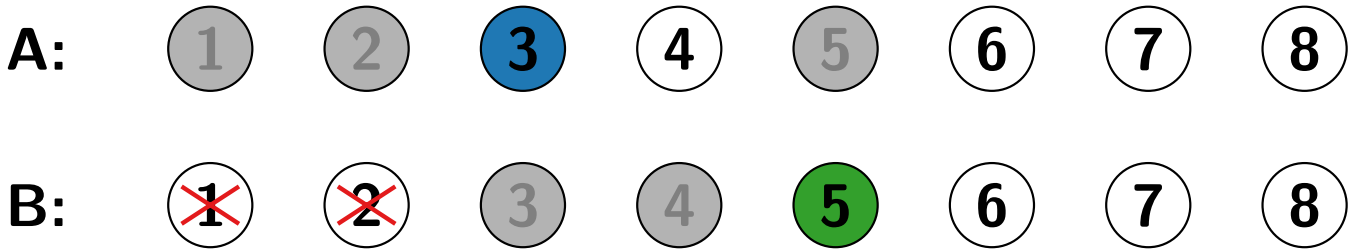
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



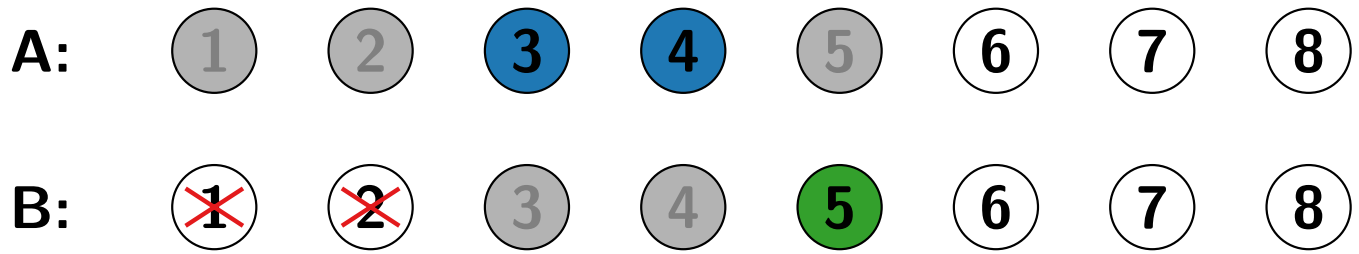
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



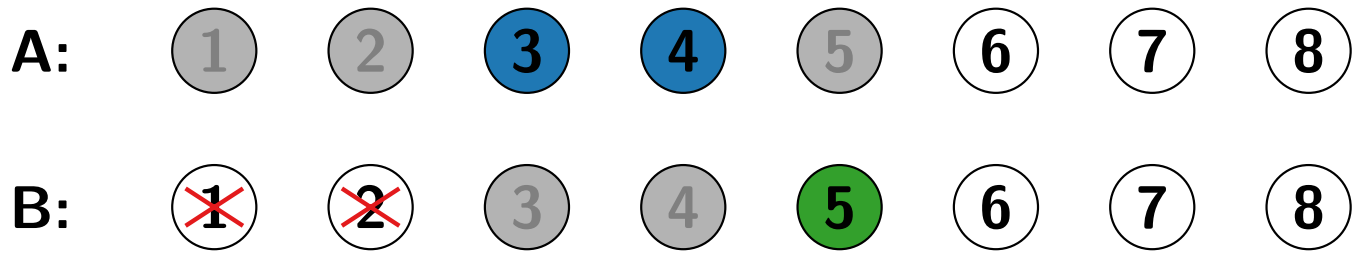
- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.

Excluding Symmetries



- Use a 0-1 encoding for A and B.
- If we exclude v from A, and if B is currently empty, then do not allow $v \in B$.
- So we force $A > B$ lexicographically (and we can do this in decision order).

Bipartite Graphs?

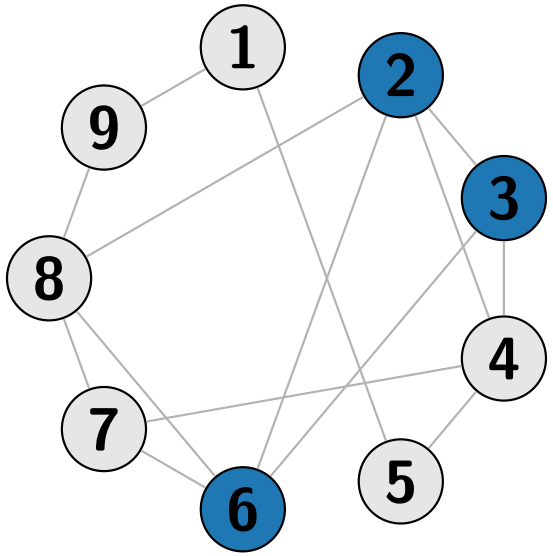
- Our independent set bound is useless if the remaining vertices on each side are independent sets. This happens immediately if the graph happens to be bipartite (this is still NP-hard).

Bipartite Graphs?

- Our independent set bound is useless if the remaining vertices on each side are independent sets. This happens immediately if the graph happens to be bipartite (this is still NP-hard).
- We can detect this: a greedy clique cover of n vertices uses n cliques iff we have an independent set. We can then do something different for a bound.
  - The size of a balanced biclique is at most the size of a biclique. But the cost of calculation seems to outweigh the pruning.
  - Something involving relative degrees?
  - Indirect colouring?

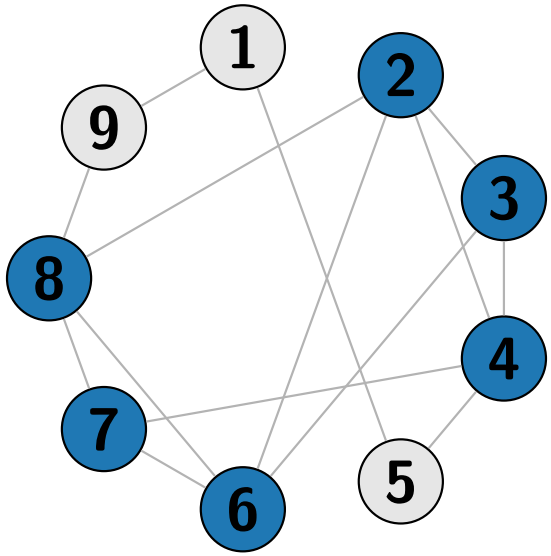
k-Cliques

k-Clique



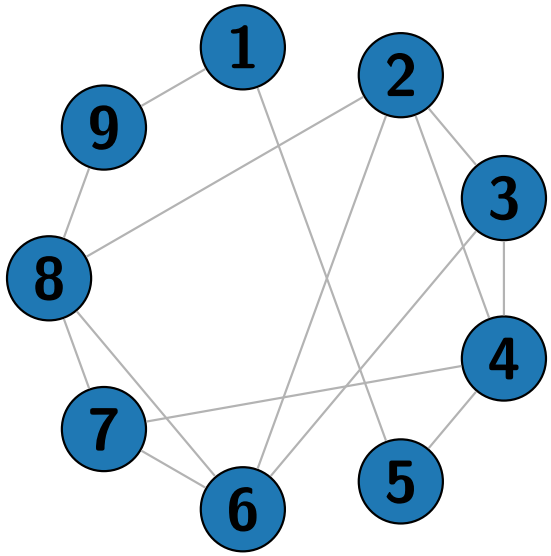
- A clique is "a set of people who all know each other". This may be too strict for real world requirements.

k-Clique



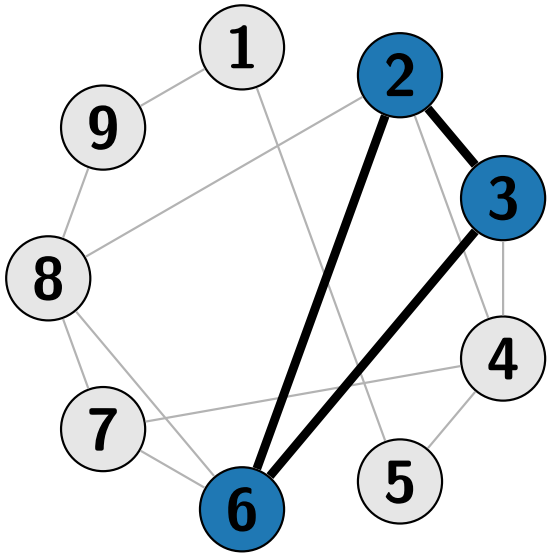
- A clique is "a set of people who all know each other". This may be too strict for real world requirements.
- A 2-clique is "a set of people who all either know each other, or have a mutual acquaintance".

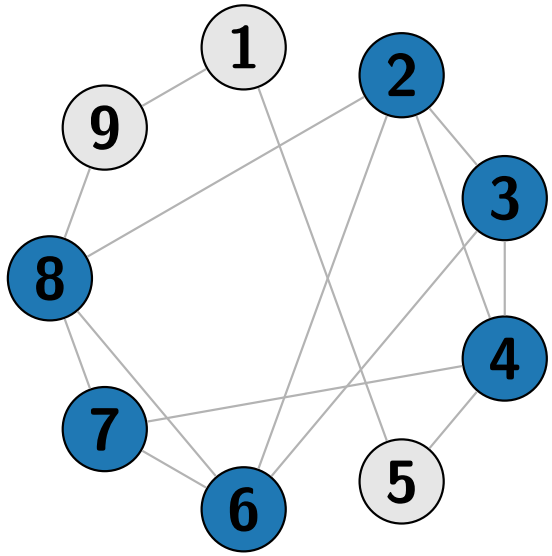
k-Clique



- A clique is "a set of people who all know each other". This may be too strict for real world requirements.
- A 2-clique is "a set of people who all either know each other, or have a mutual acquaintance".
- A k-clique is a set of vertices, each pair of which is connected by a path of length at most k.

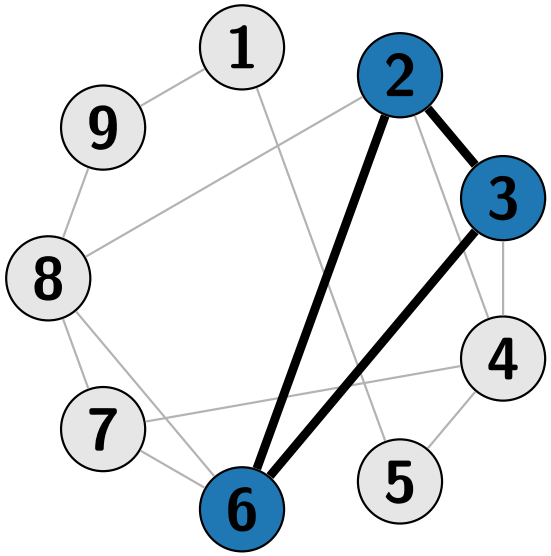
k-Clique Algorithms

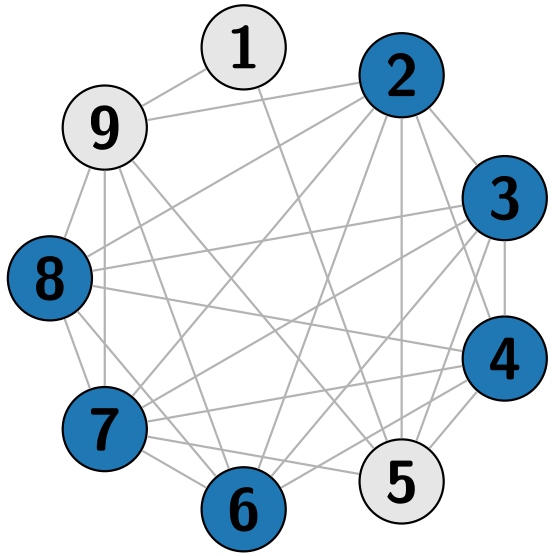




- The size of a maximum k-clique is denoted $\tilde{\omega}_k$. Finding $\tilde{\omega}_k$ is NP-hard.

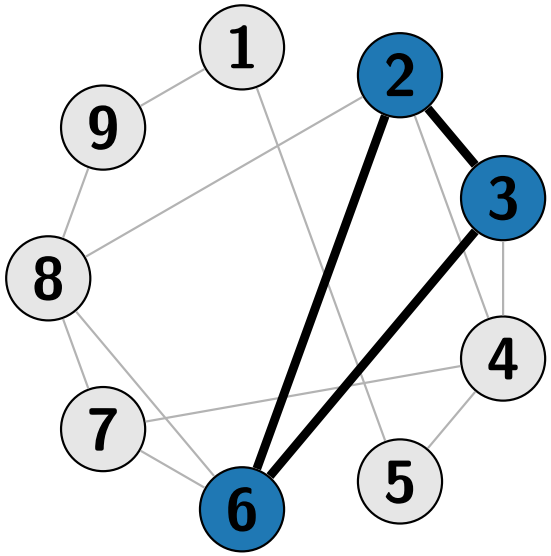
k-Clique Algorithms

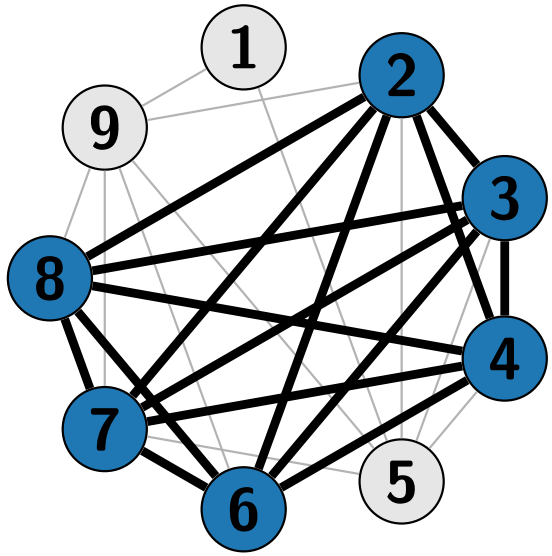




- The size of a maximum k-clique is denoted $\tilde{\omega}_k$. Finding $\tilde{\omega}_k$ is NP-hard.
- A clique in G^k is a k-clique in G, so we may reuse our maximum clique algorithms.

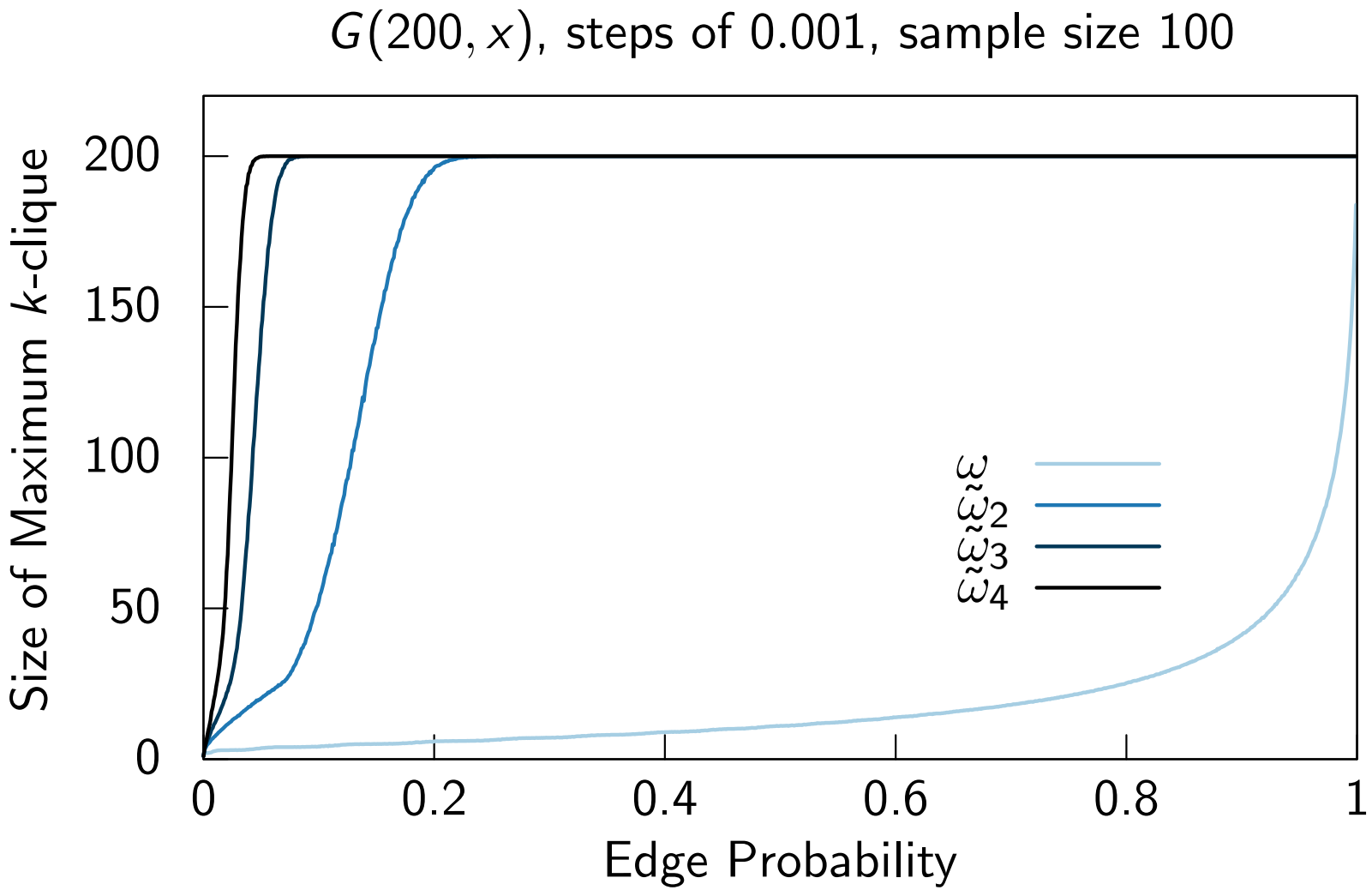
k-Clique Algorithms



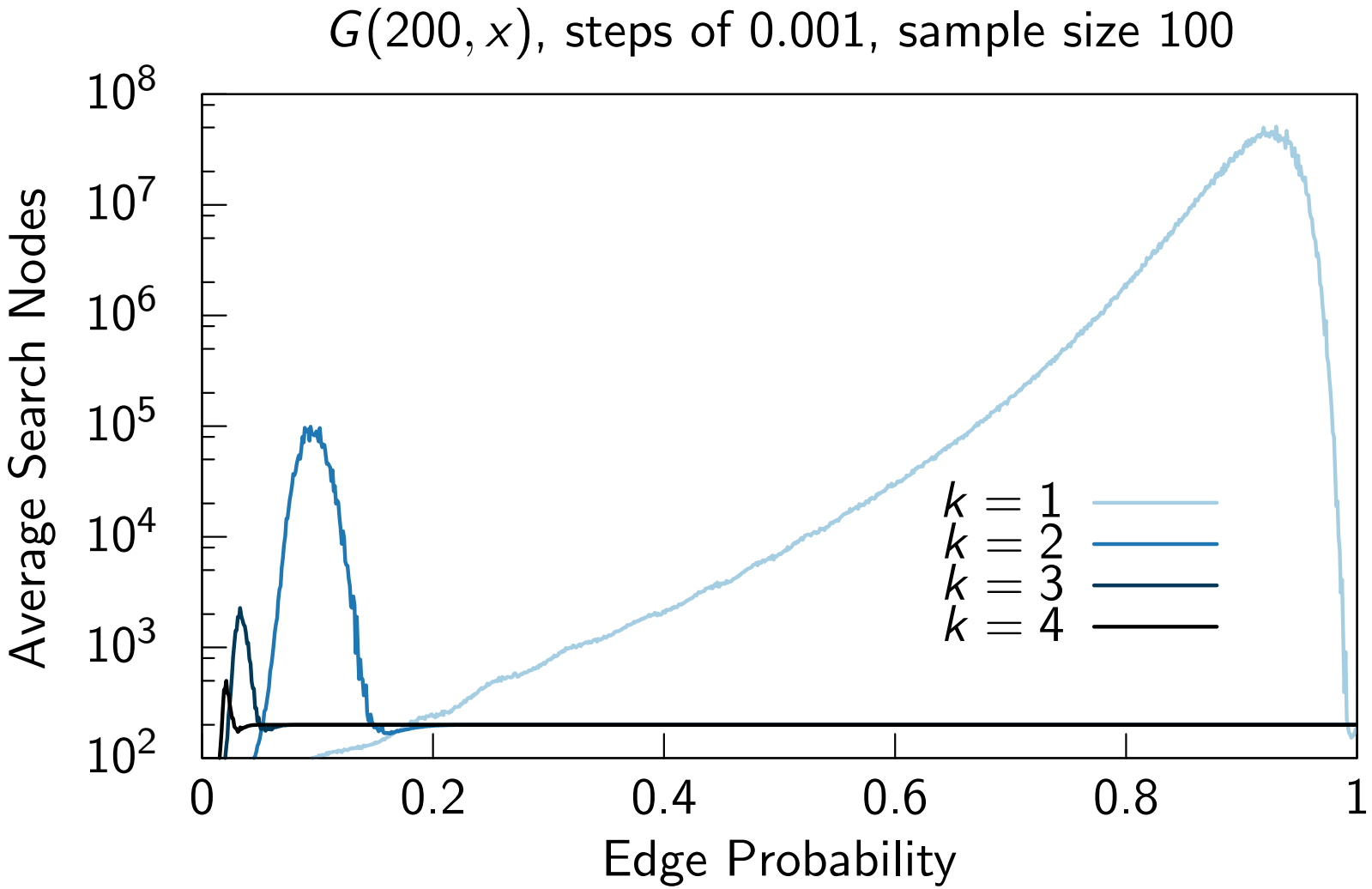


- The size of a maximum k-clique is denoted $\tilde{\omega}_k$. Finding $\tilde{\omega}_k$ is NP-hard.
- A clique in G^k is a k-clique in G, so we may reuse our maximum clique algorithms.

How Big is $\tilde{\omega}_k$?

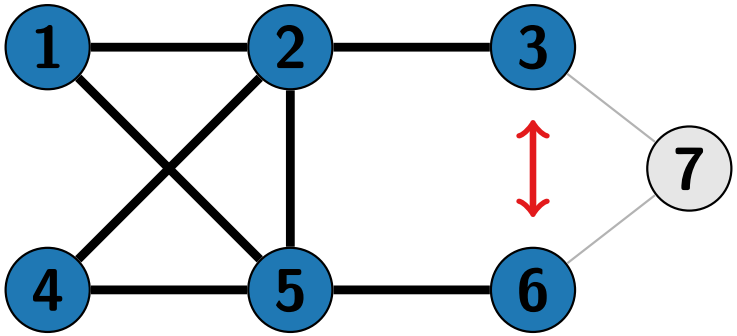


How Hard is Solving $\tilde{\omega}_k$?



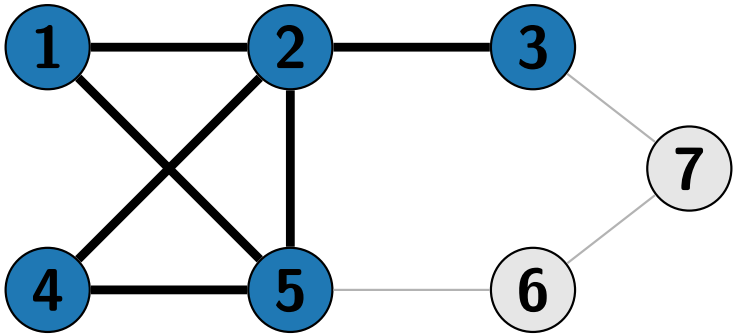
k-Clubs

k-Club



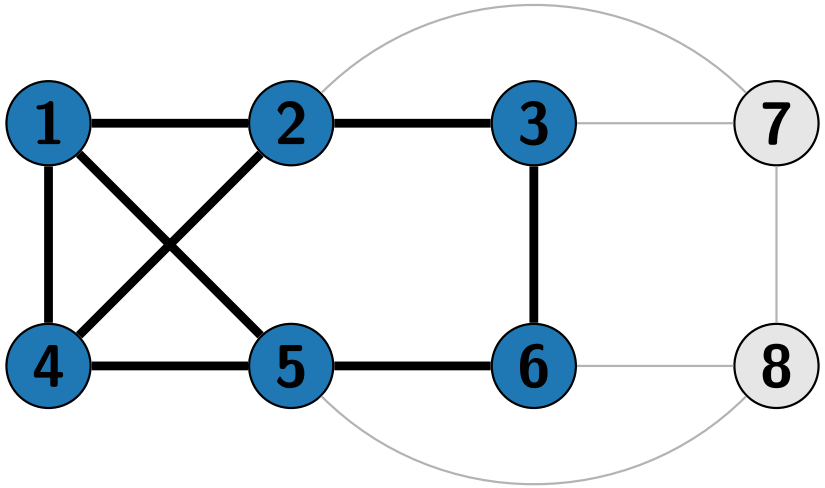
- A k-clique can use paths "that go outside the set".

k-Club

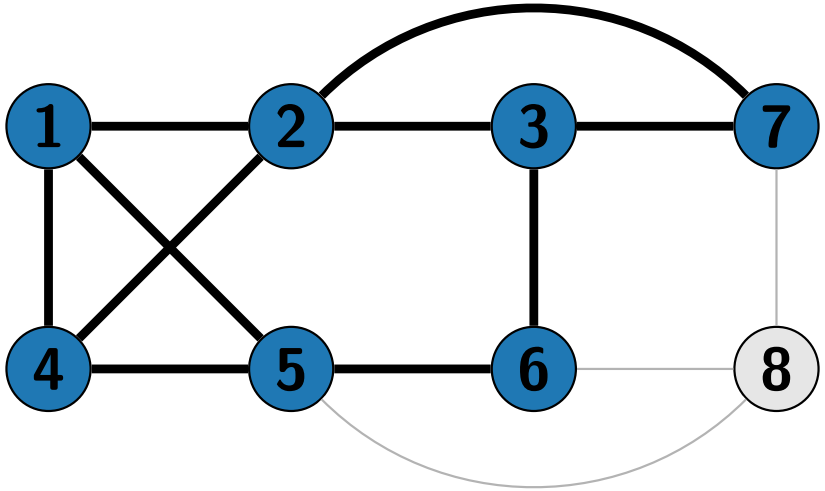


- A k-clique can use paths "that go outside the set".
- A k-club is a set of vertices, each pair of which is connected by a path of length at most k, using only vertices in this set.
- The size of a maximum k-club is denoted $\bar{\omega}_k$. Finding $\bar{\omega}_k$ is NP-hard.

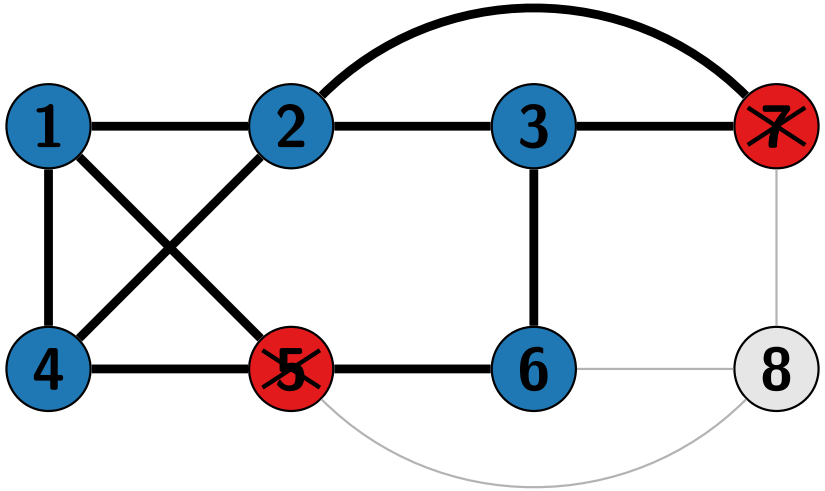
k-Clubs Don't Grow on Trees



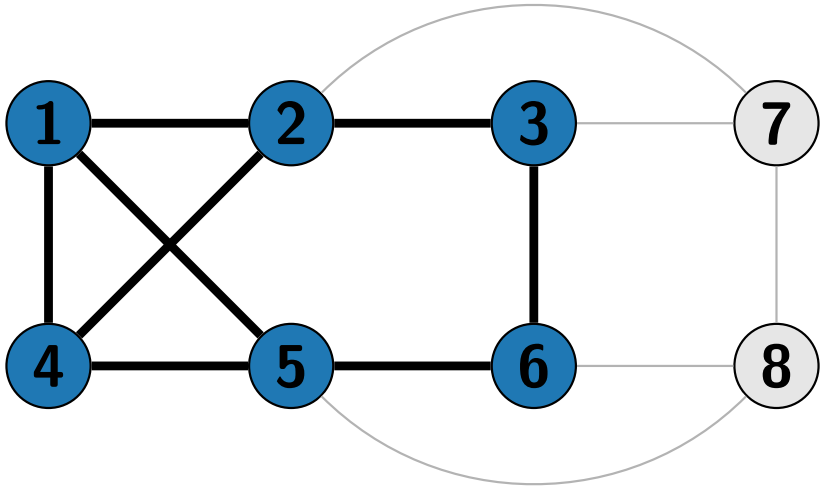
k-Clubs Don't Grow on Trees



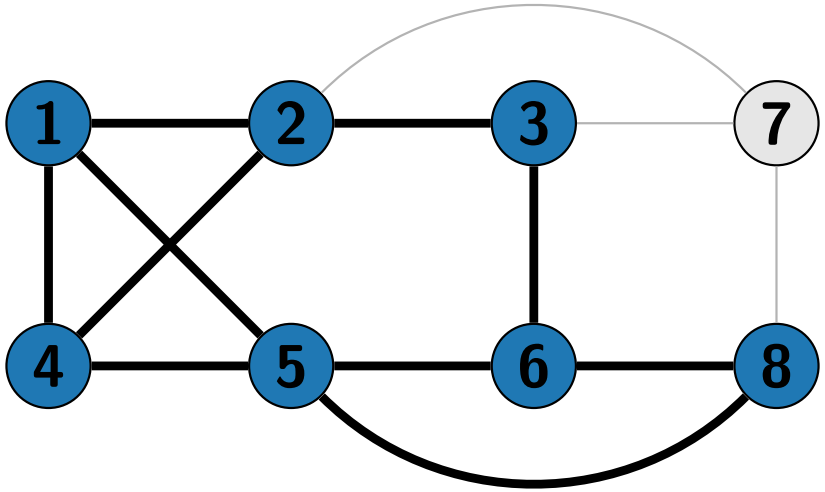
k-Clubs Don't Grow on Trees



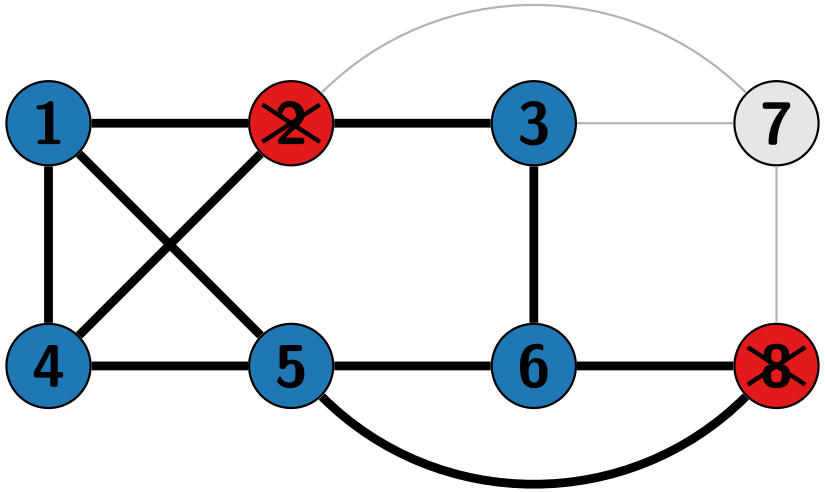
k-Clubs Don't Grow on Trees



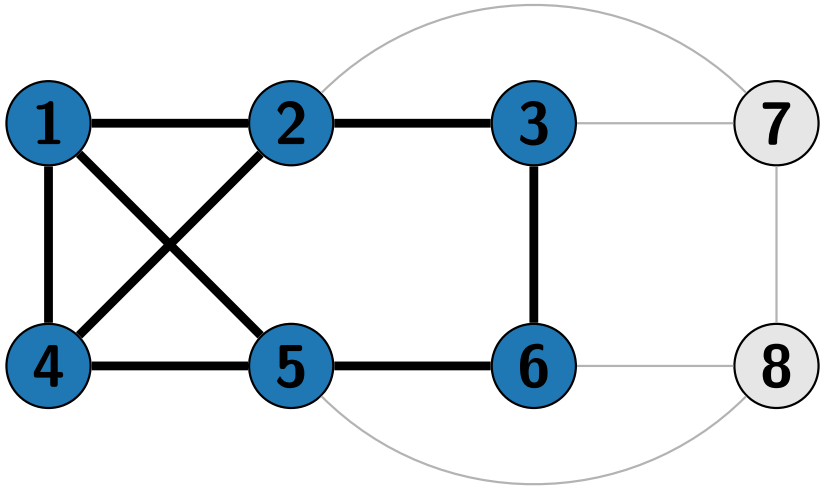
k-Clubs Don't Grow on Trees



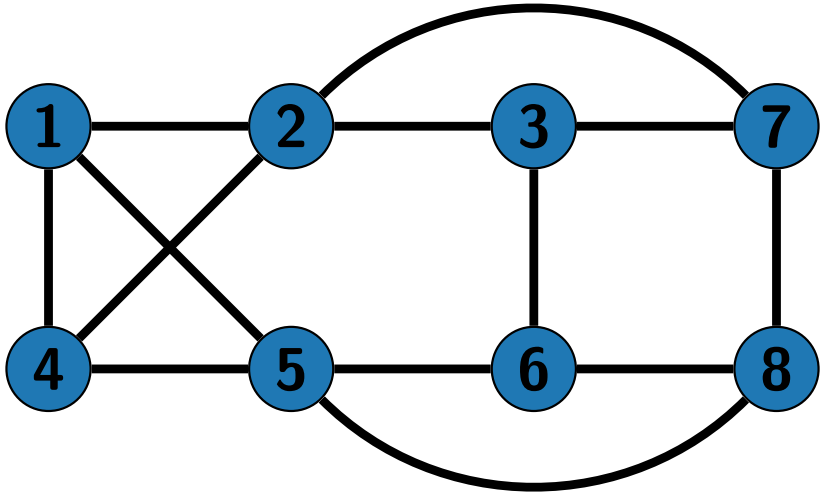
k-Clubs Don't Grow on Trees



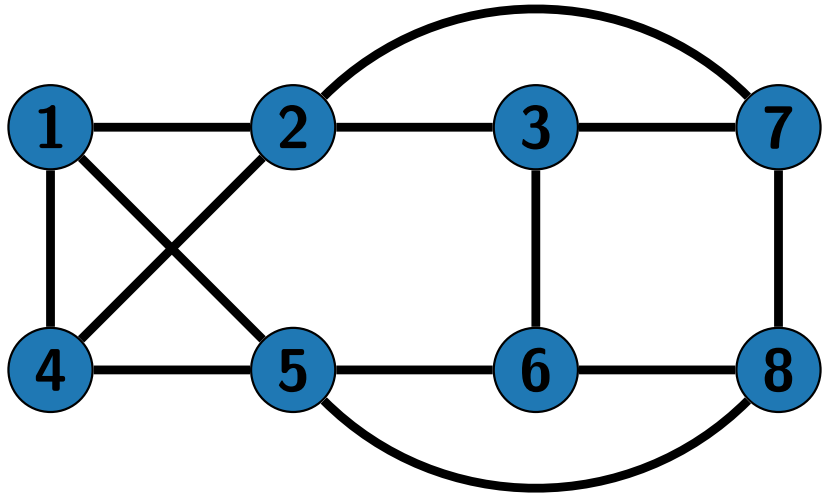
k-Clubs Don't Grow on Trees



k-Clubs Don't Grow on Trees



k-Clubs Don't Grow on Trees

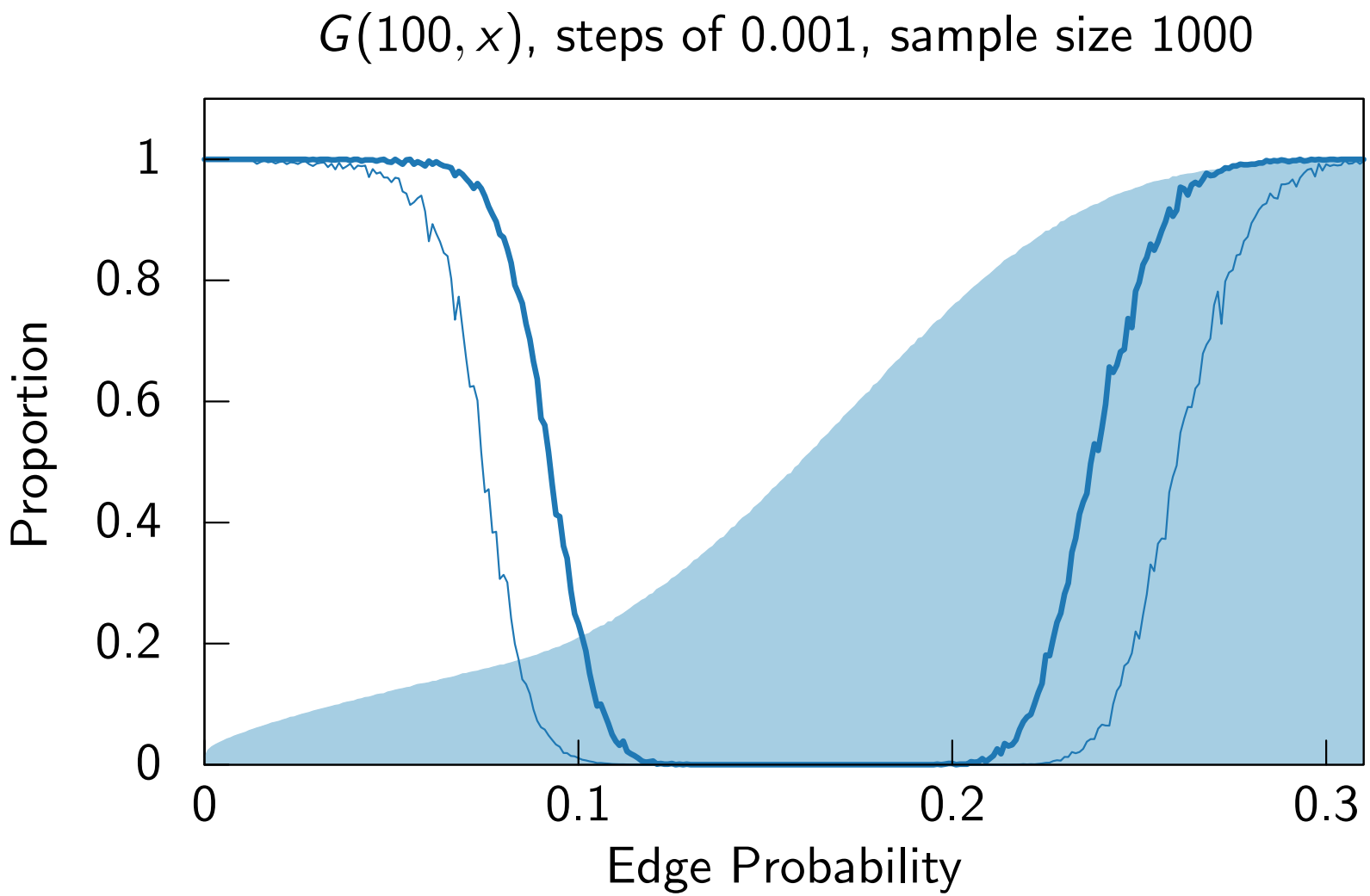


- We can't grow k-clubs recursively, one vertex at a time (being a k-club is not *hereditary on induced subgraphs*).
- Determining whether a k-club is maximal is NP-complete.
- But we can at least check whether a given k-clique is a k-club.

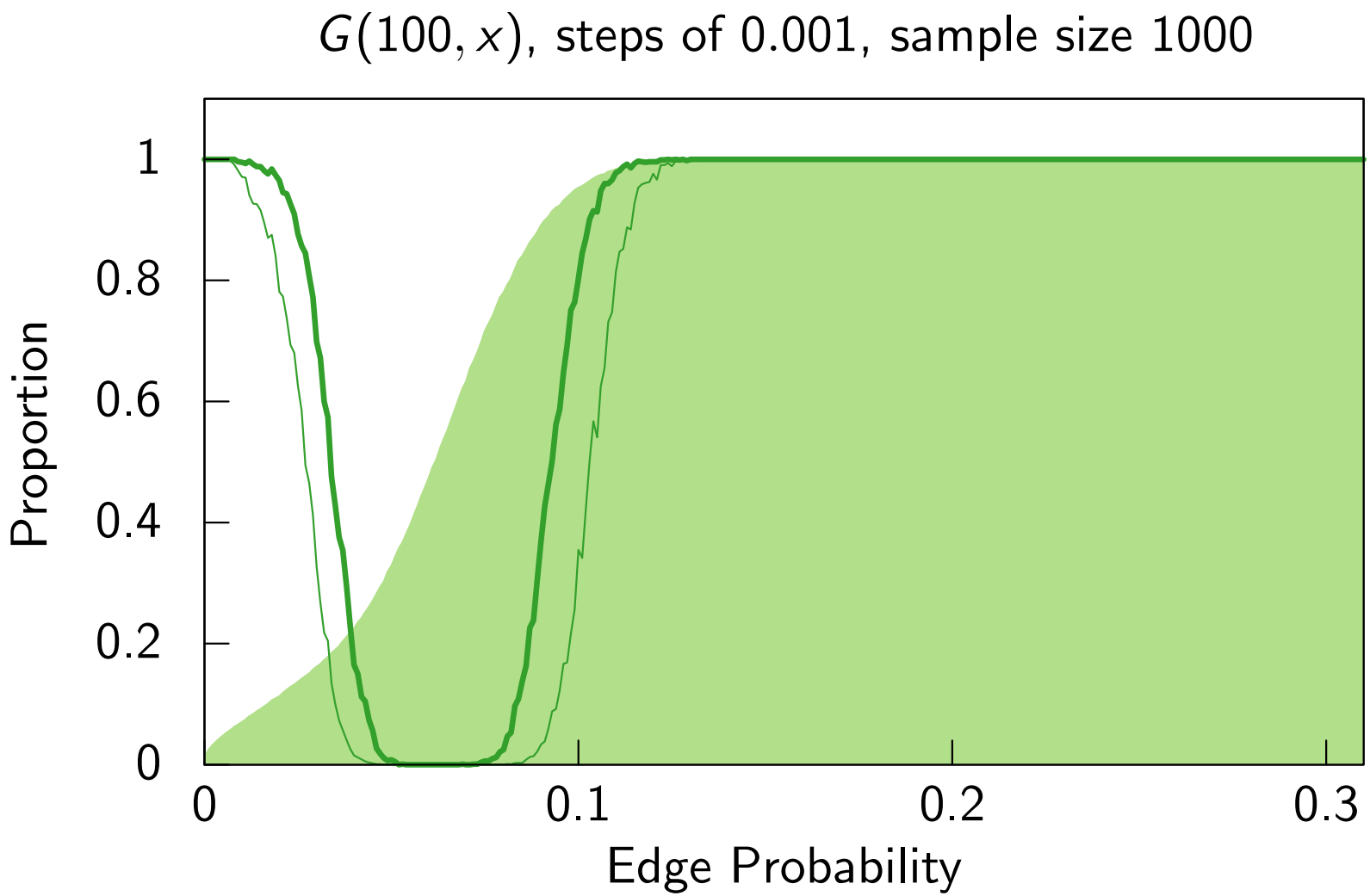
Maximum Clique Enumeration

- Find ω.
- Re-run with the incumbent fixed to $\omega - 1$.

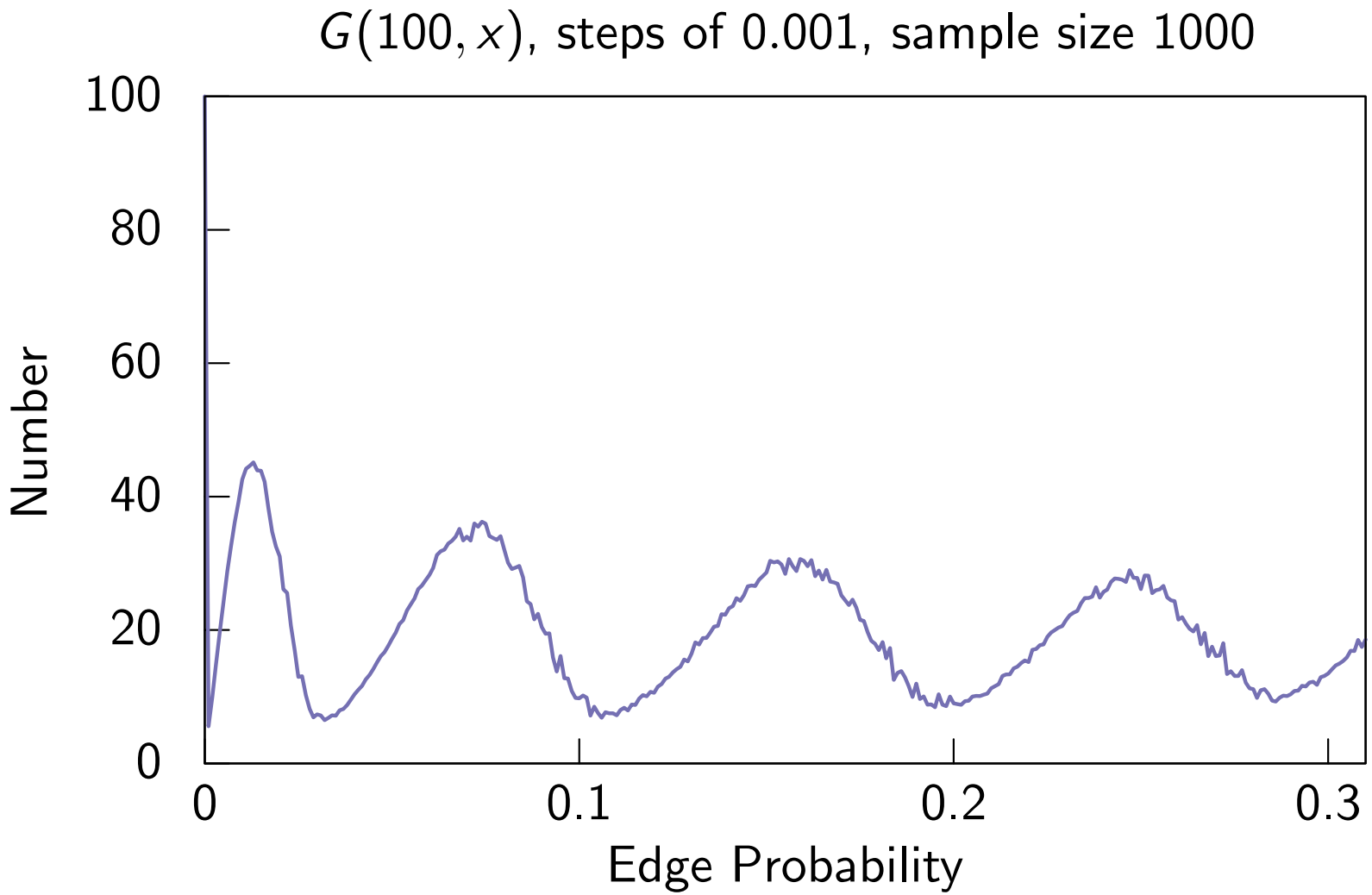
Are $\tilde{\omega}_2$ and $\bar{\omega}_2$ Different?



Are $\tilde{\omega}_3$ and $\bar{\omega}_3$ Different?



Incidentally, How Many Maximum Cliques Are There?



c.mccreesh.1@research.gla.ac.uk