# Kernelization III

Every block of stone has a statue inside it and it is the task of the sculptor to discover it.

— Michelangelo Buonarroti

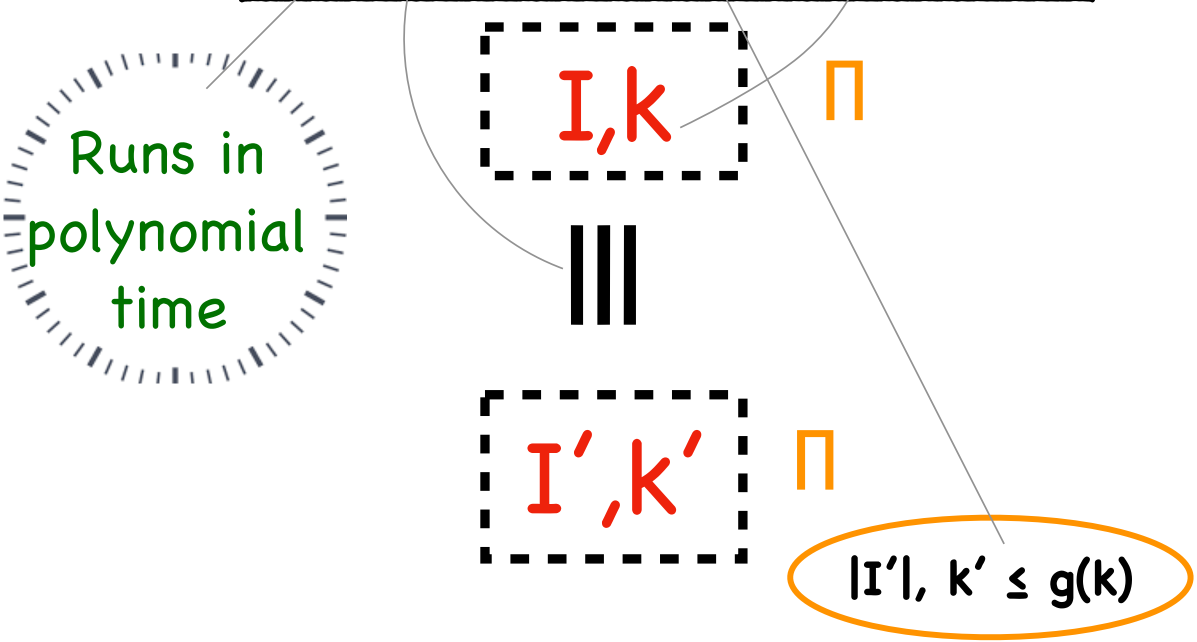
## Lecture #7

Roohani Sharma

November 30, 2021

# Kernelization

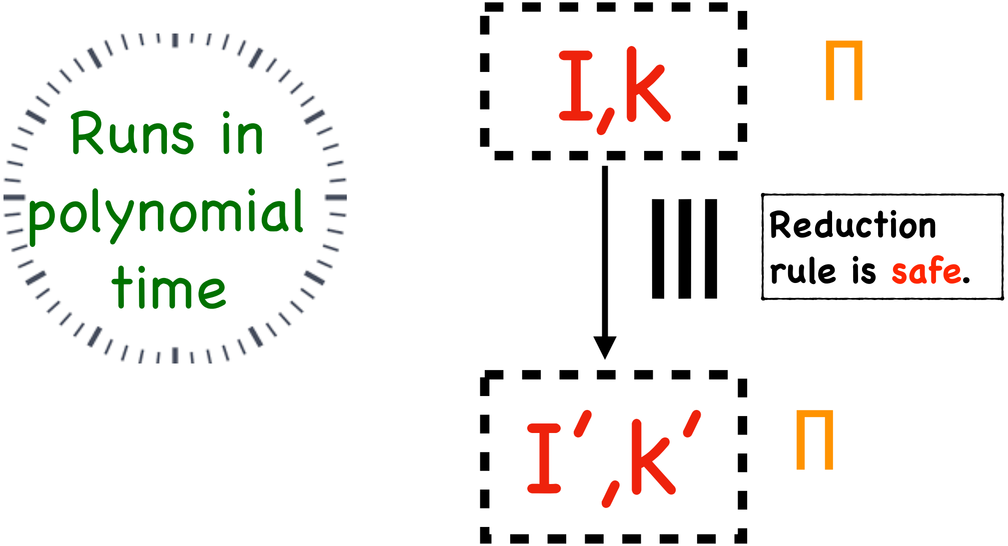
- Efficient pre-processing with guarantees for parameterized problems




Π admits a kernel of size g(k).

If g(k) is a polynomial/exponential function, then Π admits a polynomial/exponential kernel.

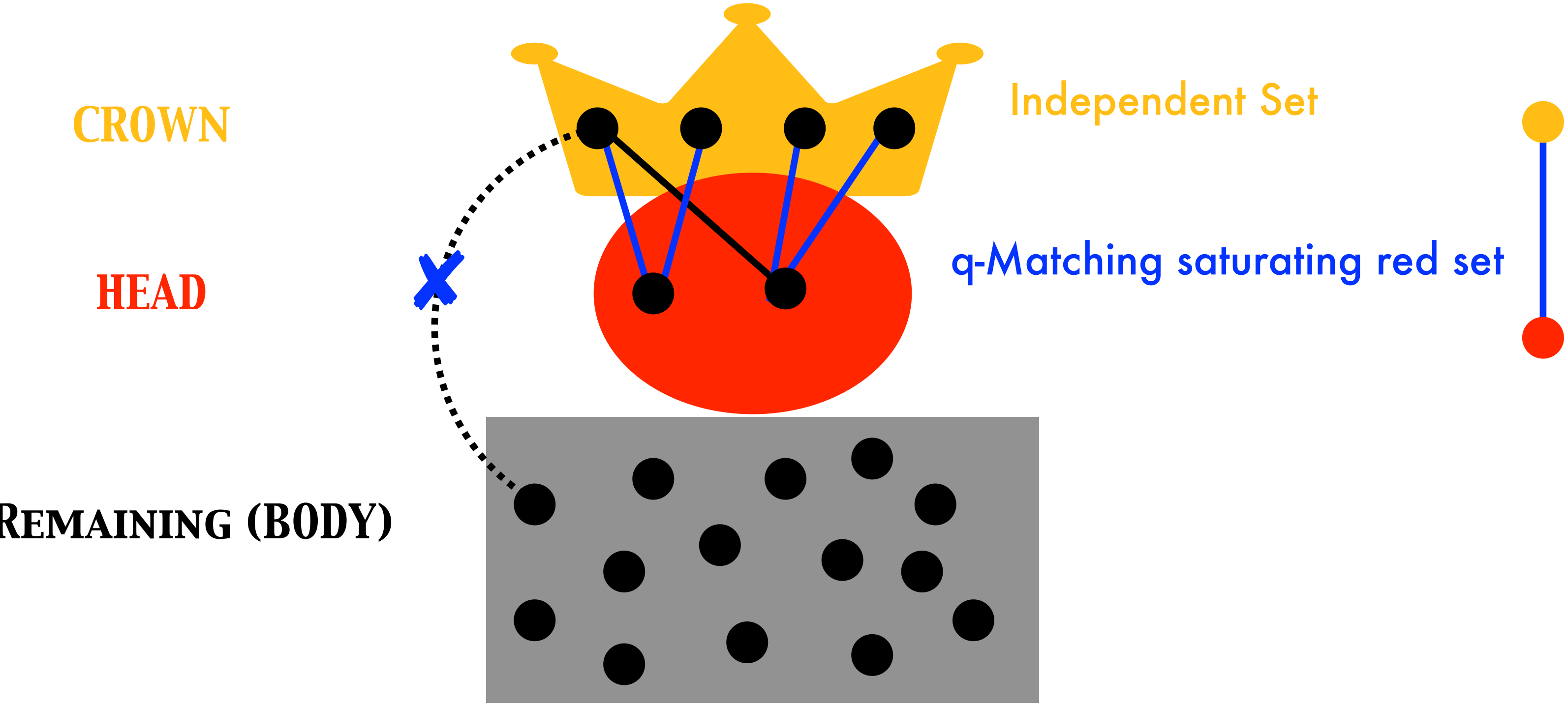
# Reduction rules (RR)




- A kernelization algorithm comprises of possibly several (polynomially many) safe reduction rules.
- Each reduction rules should be applicable only a polynomial number of times.
- This is followed by an analysis showing that if none of the designed reduction rules are applicable, then the size of the instance is bounded by some $g(k)$.

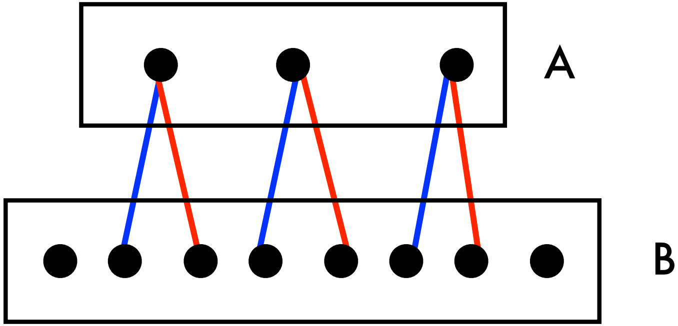
# Expansion Lemma

# q-CROWN DECOMPOSITION



# q-Expansion

Bipartite graph G=(A,B),
a set of edges M is a q-expansion of A into B if
M is a disjoint union of |A| q-stars each with its centre vertex in A
and the leaves in B.



# Expansion Lemma

Bipartite graph $G=(A,B)$, integer $q$ such that
$|B| \geq q\,|A|$, no isolated vertices in $B$:
then $\exists\, H \subseteq A$ and $C \subseteq B$ such that there is a
$q$-expansion from $A$ into $B$ and $N(C) \subseteq H$.

Can be proved using arguments similar to the proof of Hall's theorem.

# Feedback Vertex Set

# Problem Definition

FEEDBACK VERTEX SET **Parameter:** $k$
**Input:** An undirected graph $G$ and a positive integer $k$.
**Question:** Does there exists a subset $X$ of size at most $k$ such that $G - X$ is acyclic?

$X$ is called feedback-vertex set (fvs) of $G$.
Goal is to obtain a polynomial kernel for FEEDBACK VERTEX SET.

# What reduction rules we already know?

## Reduction.FVS

*If there is a loop at a vertex* $v$*, delete* $v$ *from the graph and decrease* $k$ *by one.*

# What reduction rules we already know?

Multiplicity of a multiple edge does not influence the set of feasible solutions to the instance $(G, k)$.

Reduction.FVS

*If there is an edge of multiplicity larger than* $2$*, reduce its multiplicity to* $2$*.*

# What reduction rules we already know?

Any vertex of degree at most $1$ does not participate in any cycle in $G$, so it can be deleted.

Reduction.FVS

*If there is a vertex* $v$ *of degree at most* $1$*, delete* $v$*.*

# What reduction rules we already know?

Concerning vertices of degree 2, observe that, instead of including into the solution any such vertex, we may as well include one of its neighbors.

## Reduction.FVS

*If there is a vertex $v$ of degree 2, delete $v$ and connect its two neighbors by a new edge.*

# What do we achieve after all these?

After exhaustively applying these four reduction rules, the resulting graph $\mathsf{G}$

(P1) contains no loops,

(P2) has only single and double edges, and

(P3) has minimum vertex degree at least 3.

# Stopping rule.

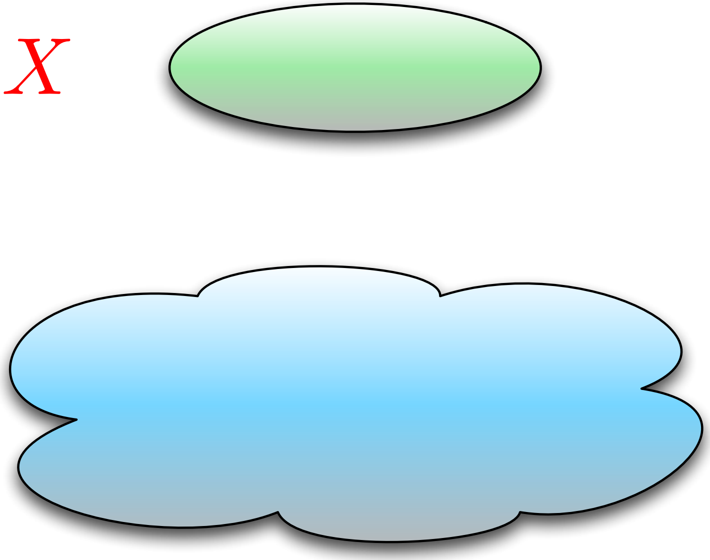
A rule that stops the algorithm if we already exceeded our budget.

**Reduction.FVS**

*If* $\mathsf{k} < 0$, *terminate the algorithm and conclude that* $(\mathsf{G}, \mathsf{k})$ *is a no-instance.*

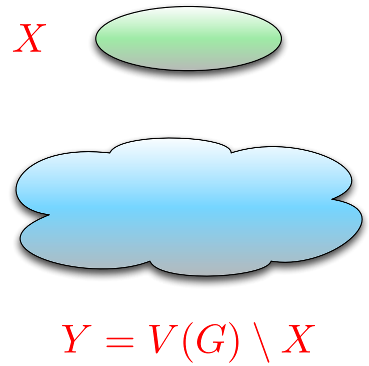
# A picture :)

$X$

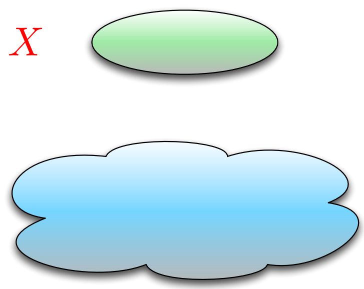


$$Y = V(G) \setminus X$$

# Maximum degree is $\mathbf{d}$.



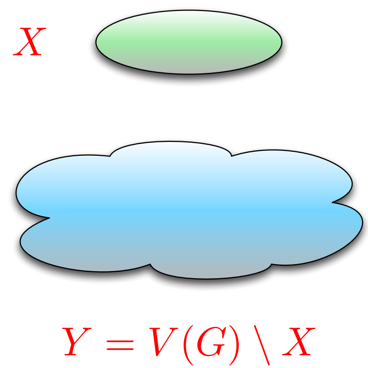
# Maximum degree is **d**.




$Y = V(G) \setminus X$

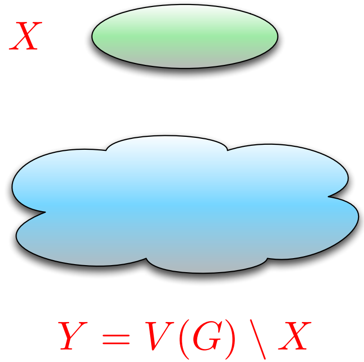
$$\sum_{v \in V(G)} \text{degree}(v) = 2|E(G)|$$

# Maximum degree is **d**.



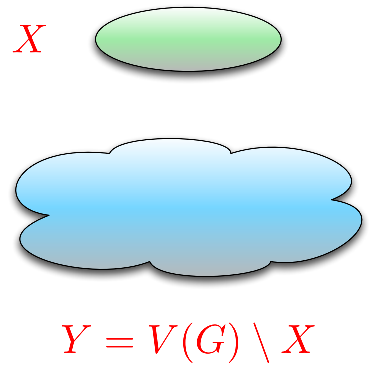

$$3|V(G)| \leqslant \sum_{v \in V(G)} \text{degree}(v) = 2|E(G)|$$

# Maximum degree is $\mathbf{d}$.



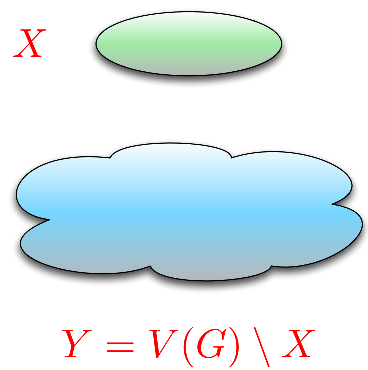

$$1.5|V(G)| \leqslant |E(G)|$$

# Maximum degree is $\mathbf{d}$.



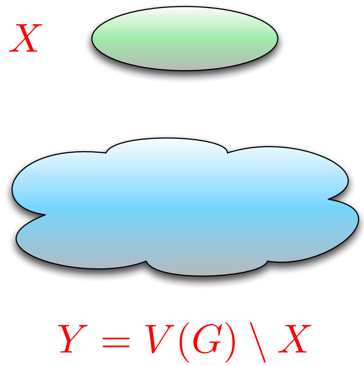

$$|E(G)| \leqslant d|X| + (|V(G)| - |X| - 1)$$

# Maximum degree is $\mathbf{d}$.




$$1.5|V(G)| \leqslant |E(G)| \leqslant d|X| + (|V(G)| - |X|)$$

# Maximum degree is **d**.




$$1.5|V(G)| \leqslant |E(G)| \leqslant d|X| + (|V(G)| - |X|)$$

$$\implies |V(G)| \leqslant 2(d-1)|X| \leqslant 2(d-1)k.$$

# Summarizing:

**Lemma**

*If a graph* $G$ *has minimum degree at least* $3$*, maximum degree at most* $d$*, and feedback vertex set of size at most* $k$*, then it has less than* $2(d-1)k$ *vertices and less than* $2(d-1)dk$ *edges.*

So what do we need to get the polynomial kernel?

Bound the maximum degree of the graph by a polynomial in $\mathbf{k}$.

# So what do we need to get the polynomial kernel?

## Bound the maximum degree of the graph by a polynomial in $k$.

Fix a vertex v such that deg(v)>=10k.

The goal henceforth is to bound the degree of v.

- For Vertex Cover – if a vertex has degree $k + 1$ then we must have it in the solution.

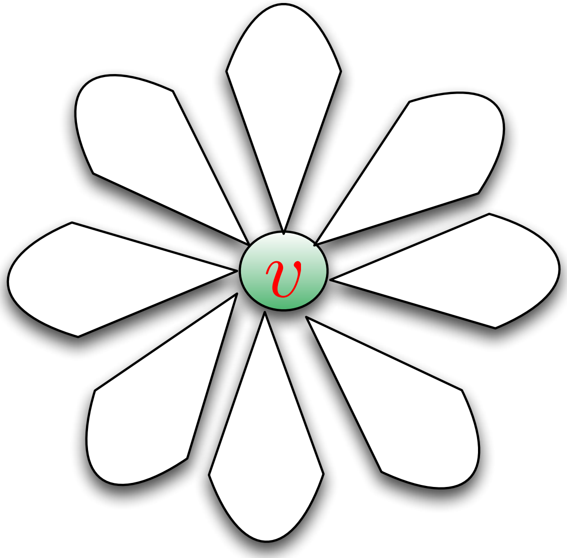
What would be the analogous rule for Feedback vertex Set.

- For VERTEX COVER – if a vertex has degree $k + 1$ then we must have it in the solution.

What would be the analogous rule for FEEDBACK VERTEX SET.

For VERTEX COVER – wanted to hit edges and for FEEDBACK VERTEX SET – want to hit cycles..

# k+1– FLOWER




$k+1-$ vertex disjoint cycles passing through it

# Flower Rule.

Reduction.FVS

*If there is a* $k+1$-*flower passing through a vertex* $v$ *then* $(G \setminus \{v\}, k-1)$.

# Flower Rule.

**Reduction.FVS**

*If there is a* $k+1$*-flower passing through a vertex* $v$ *then* $(G \setminus \{v\}, k-1)$.

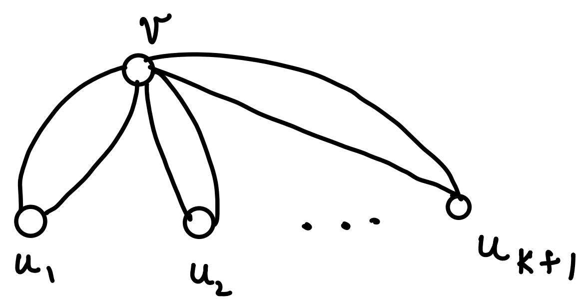
How to find a k+1-flower through v in polynomial time?

# Flower Rule.

**Reduction.FVS**
*If there is a* $k+1$*-flower passing through a vertex* $v$ *then* $(G \setminus \{v\}, k-1)$.

How to find a k+1-flower through v in polynomial time?

Special k+1-flower through v:



# Finding flowers at v?

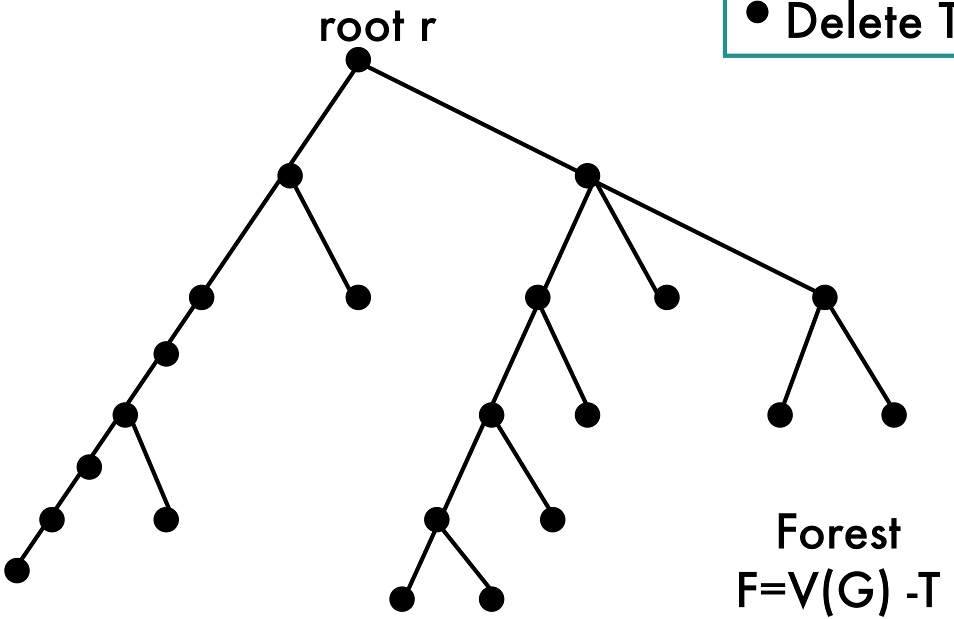
Given a vertex v and integer k,

either find a k+1-flower at v,

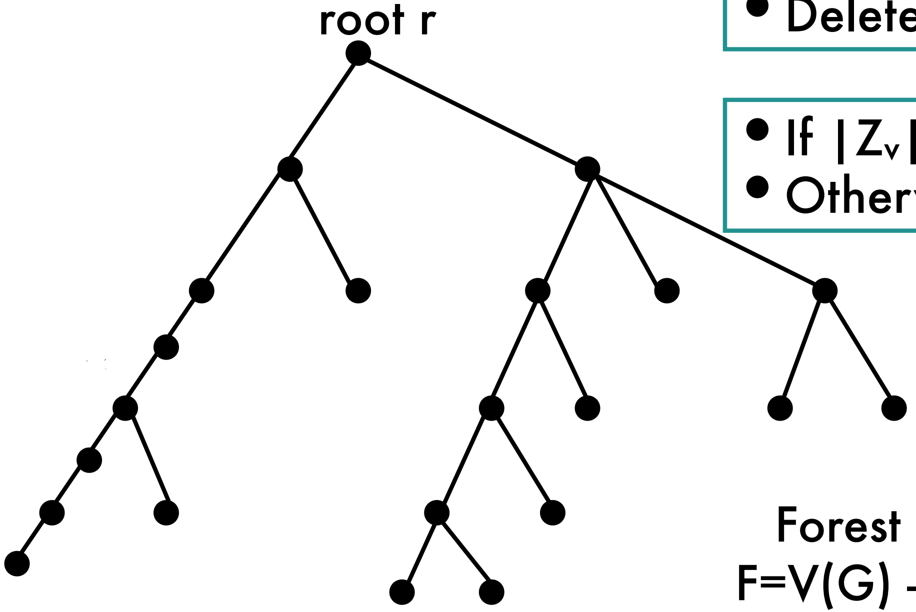
or find a solution $Z_v$ of size at most 3k that do not include v.

- Find an approximate solution T of size at most 2k (Classical result).
- If $v \notin T$, then $Z_v = T$.
- Otherwise, let F=V(G)-T be the forest…


- Initialize $Z_v = T \setminus \{v\}$.
- Let u be a `lowest' vertex in F such that there is a cycle in $G[v \cup T_u]$.
- Pick u in $Z_v$, that is $Z_v = u \cup Z_v$.
- Delete $T_u$ from F and repeat.

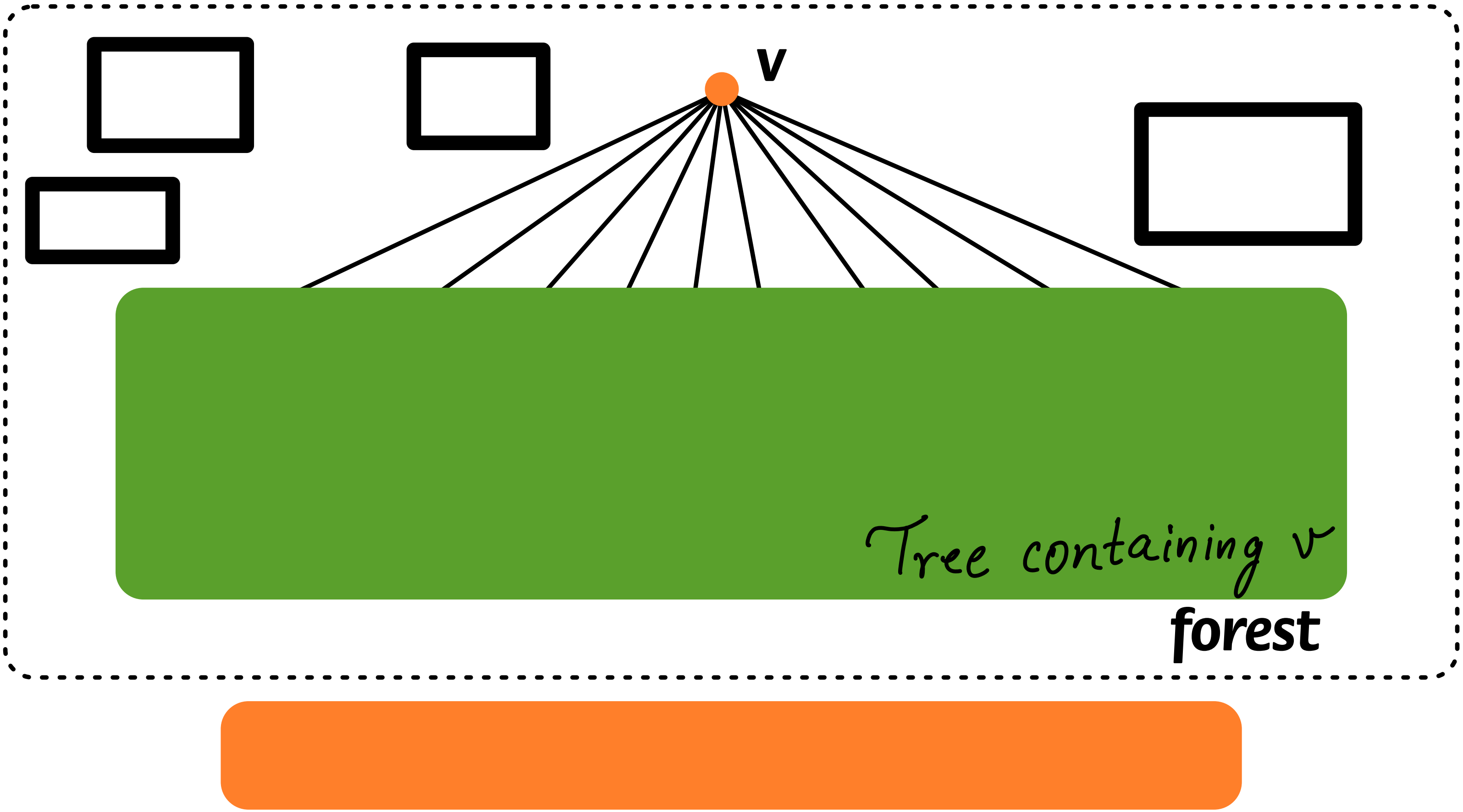



- Initialize $Z_v = T$.
- Let u be a `lowest' vertex in F such that there is a cycle in $G[v \cup T_u]$.
- Pick u in $Z_v$, that is $Z_v = u \cup Z_v$.
- Delete $T_u$ from F and repeat.




- If $|Z_v| \leq |T|+k$, then output $S_v$.
- Otherwise, there is k+1-flower at v.

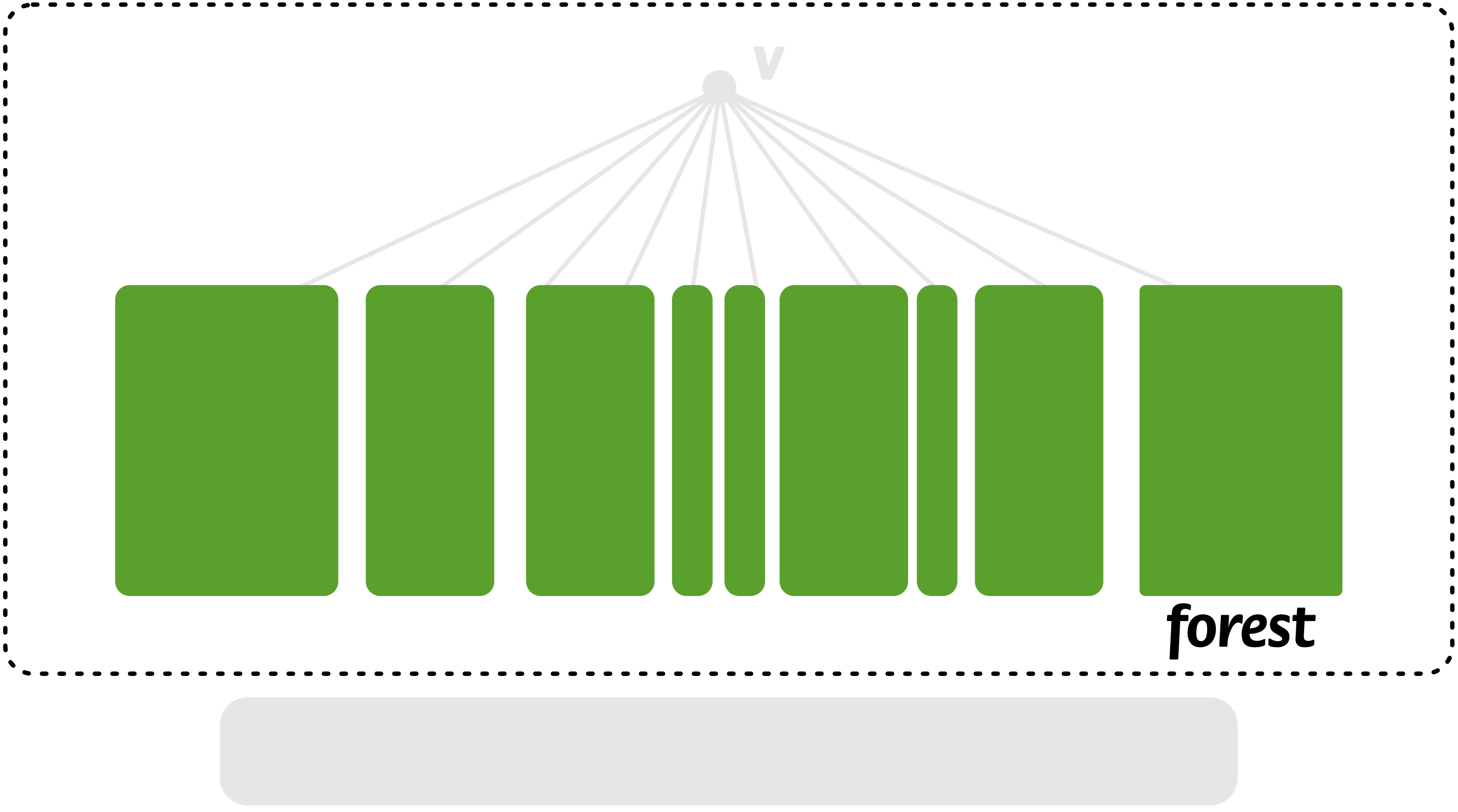
Forest
F=V(G) -T




**hitting set that excludes v** : $Z_v$
(solution)

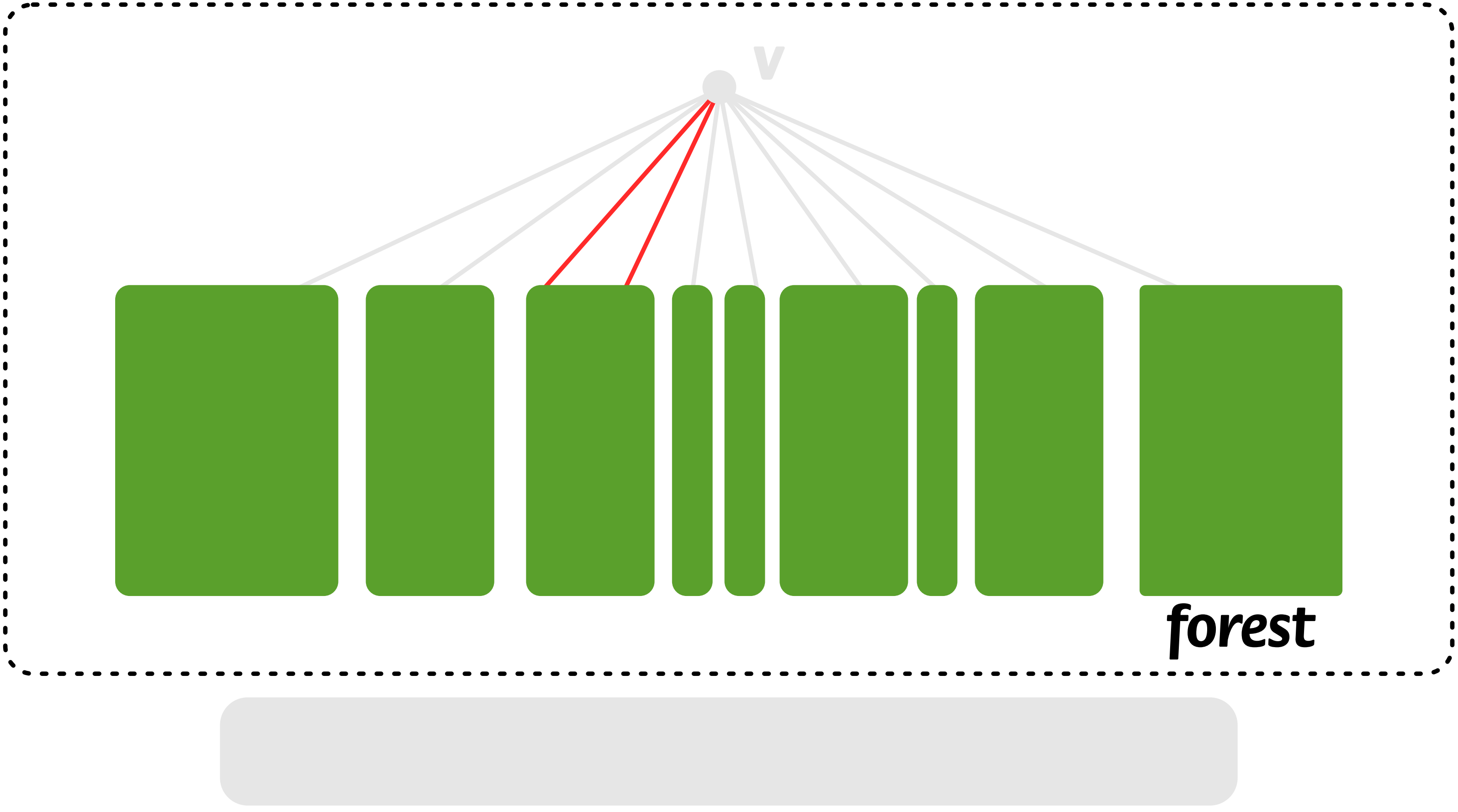
# Focussing on the green Part

Consider the connected components of $\mathsf{V}(\mathsf{G}) \setminus (\mathsf{Z}_v \cup \{v\})$.

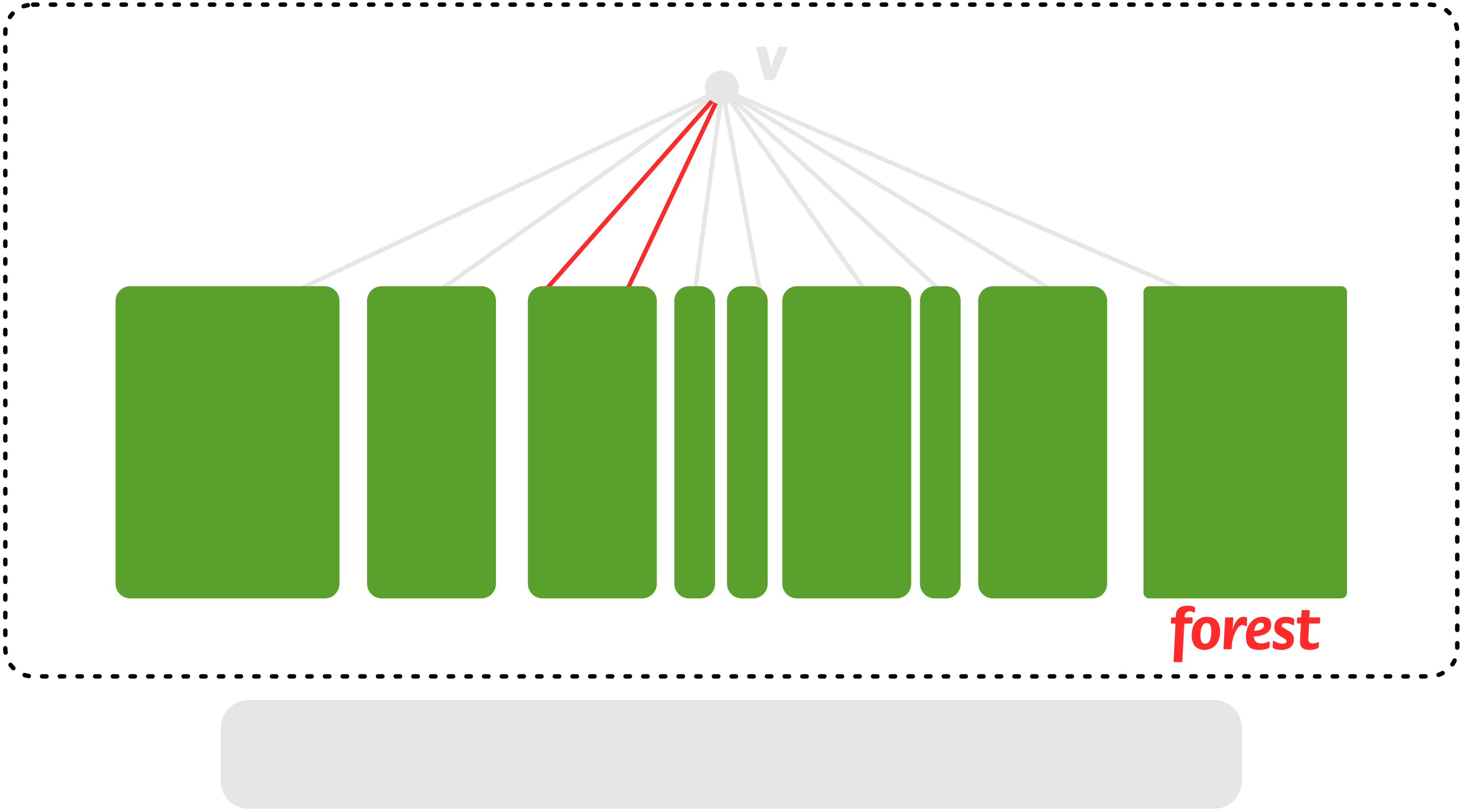



**hitting set that excludes v** : $Z_v$ ; $|Z_v| \leq 3k$

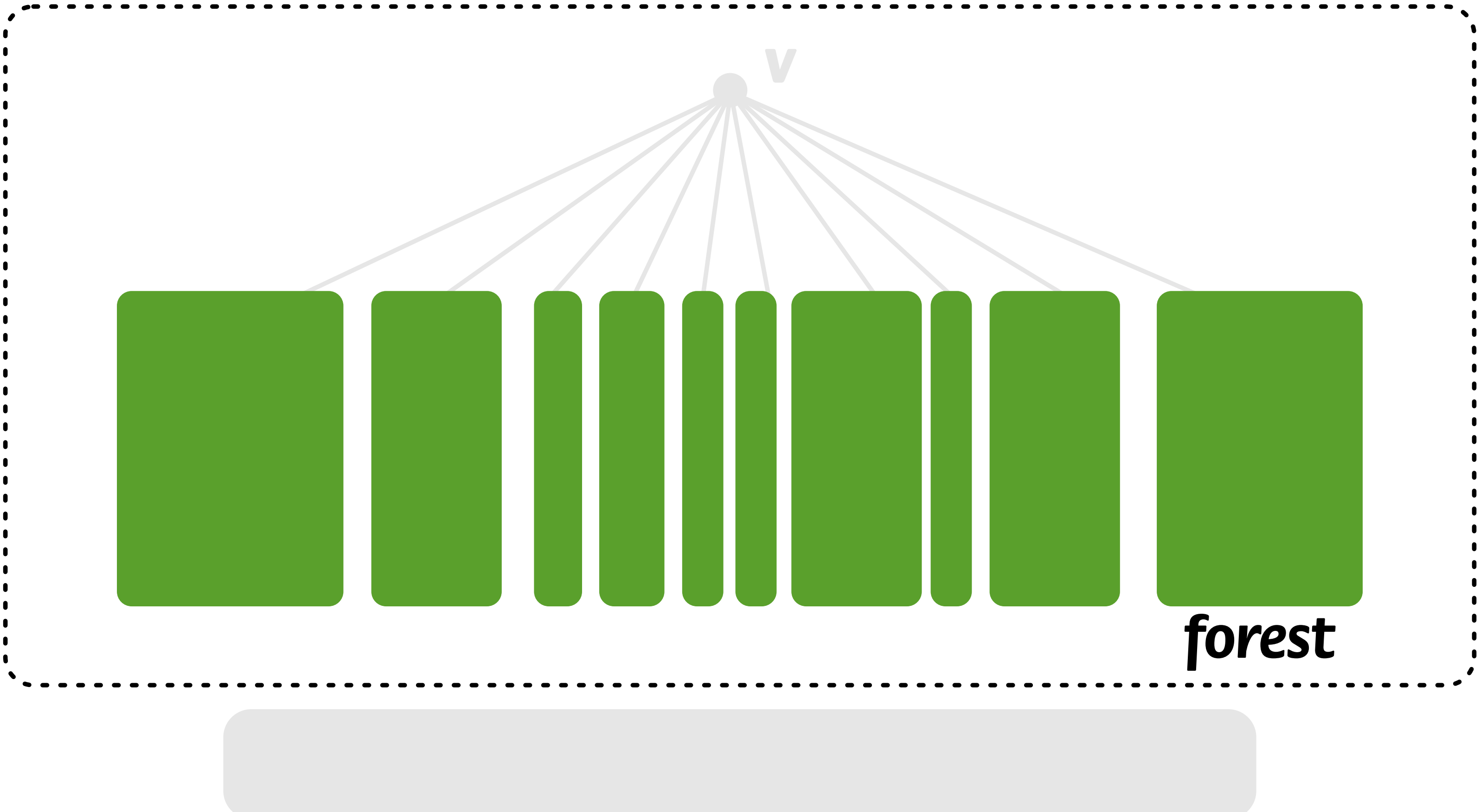
Could $v$ have two neighbor in a connected components of $\mathsf{V}(\mathsf{G}) \setminus (\mathsf{Z}_v \cup \{v\})$?




**hitting set that excludes v** : $Z_v$

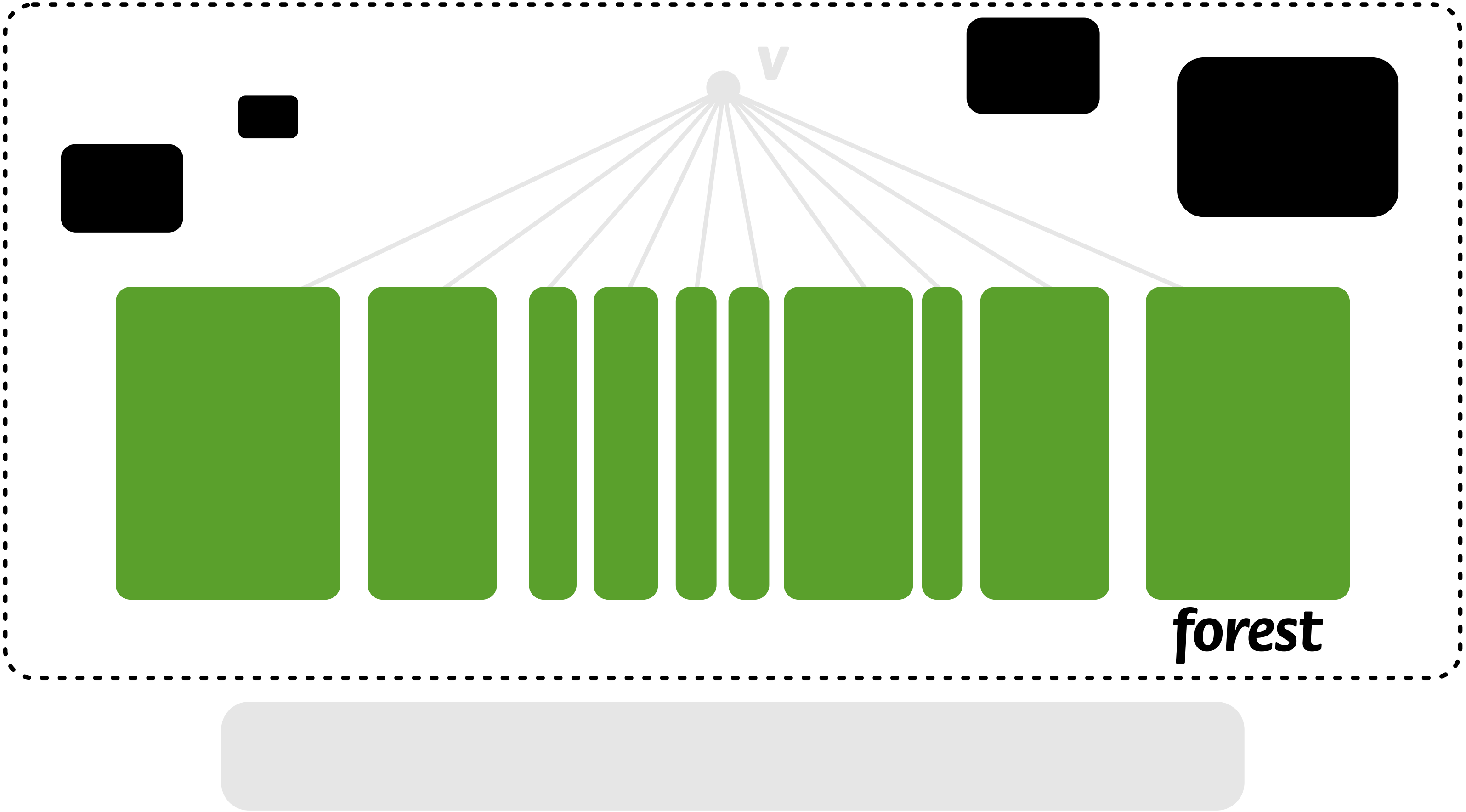



**hitting set that excludes v** : $Z_v$



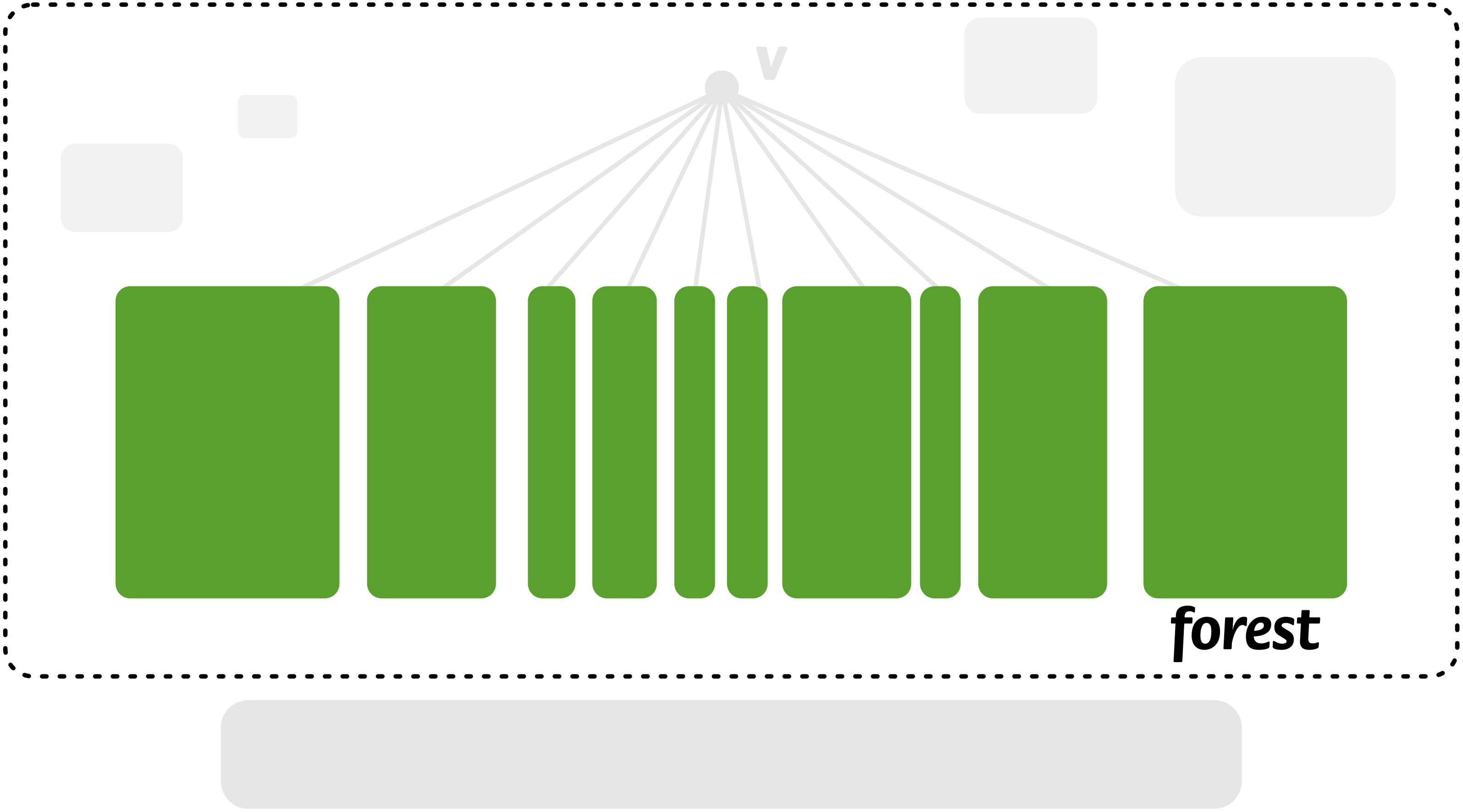

hitting set that excludes v

There could be components in $\mathsf{V}(\mathsf{G}) \setminus (\mathsf{Z}_v \cup \{v\})$ that do not see any neighbor of $v$. Important, for us is that any component contains at most one neighbor of $v$ and we will focus on them.



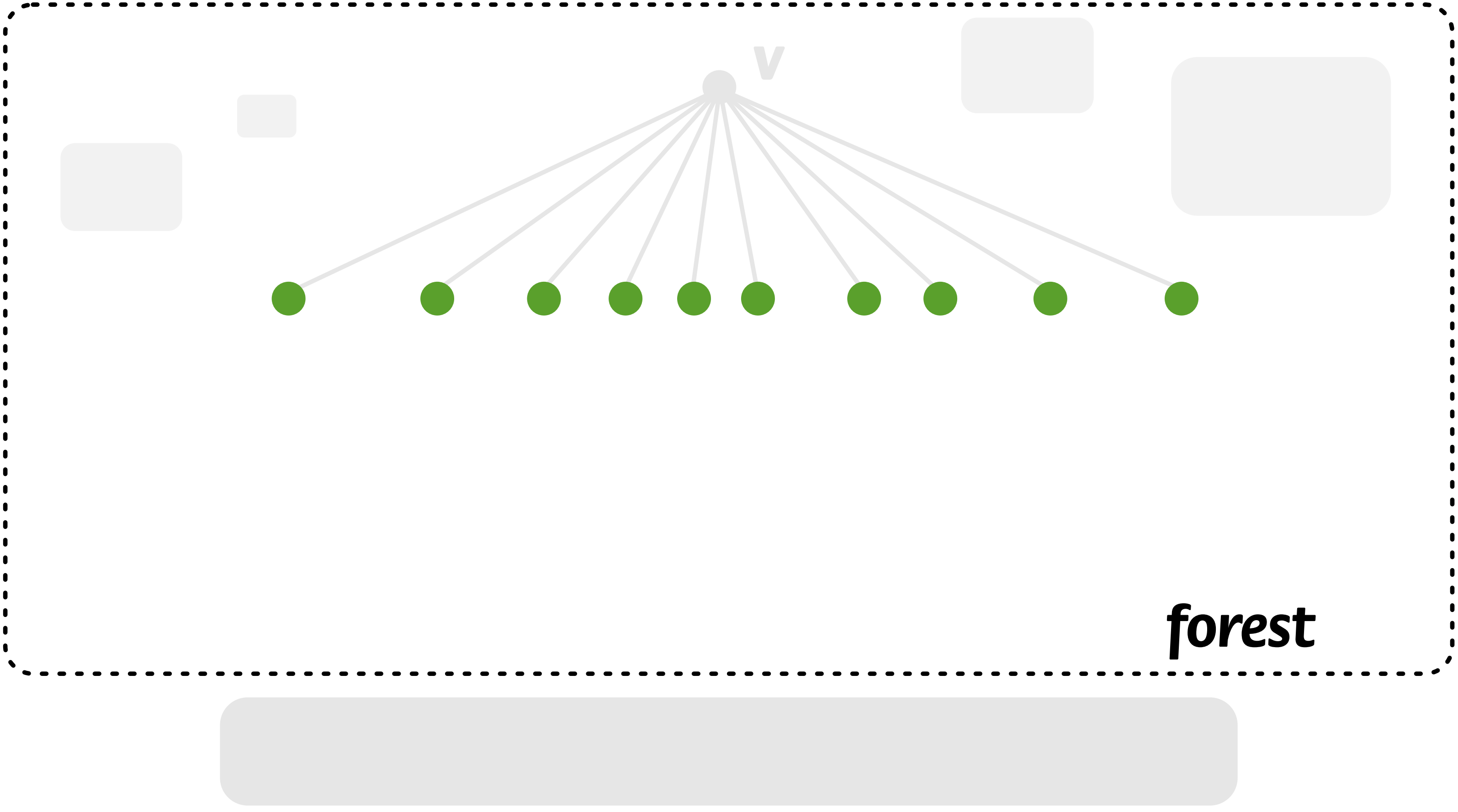

**hitting set that excludes v** : $Z_v$

To bound the degree of $v$ or to delete an edge incident to $v$ we only focus on those components that contain some (exactly one) neighbor of $v$.



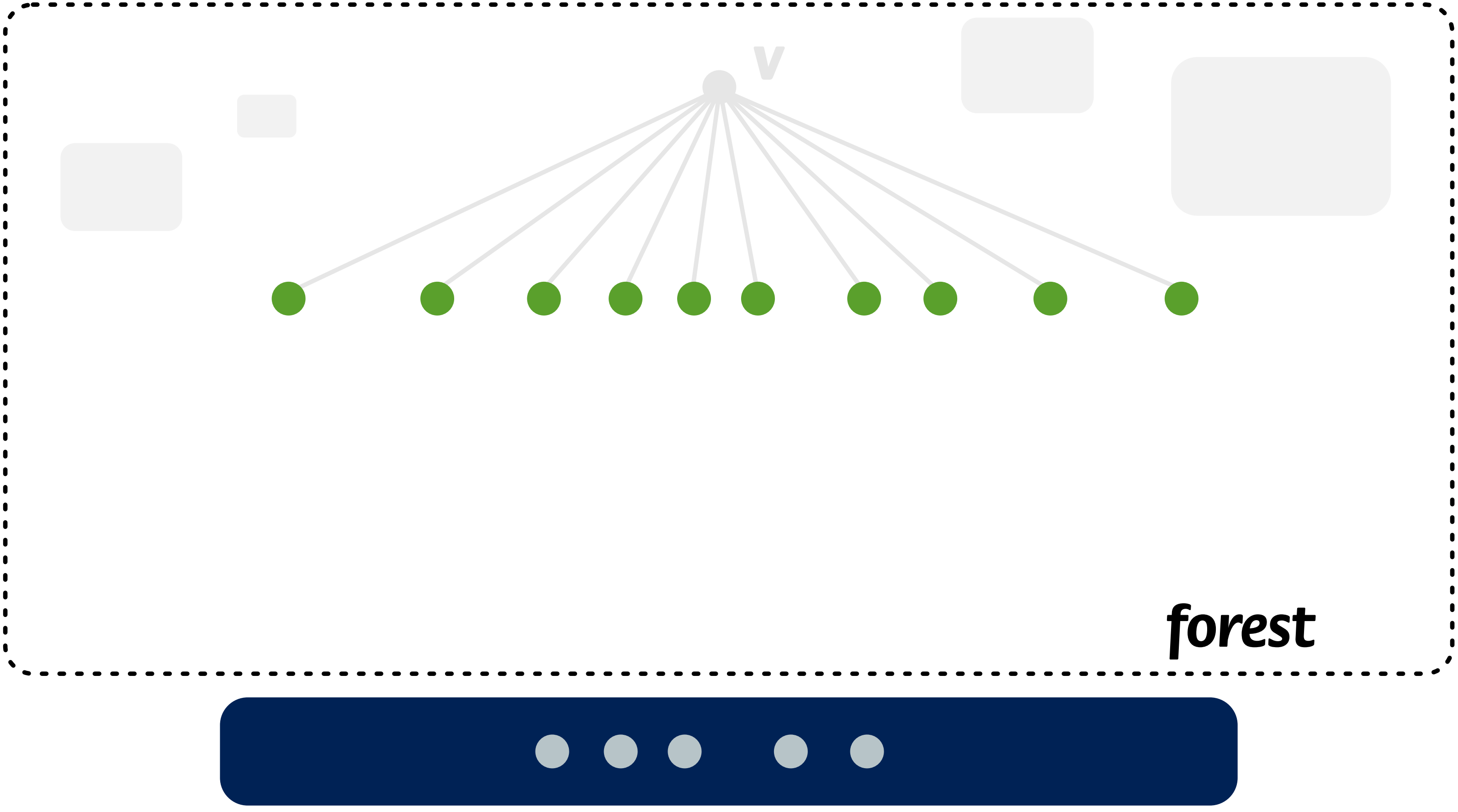

**hitting set that excludes v** : $Z_v$

To apply $2$-expansion lemma we need a bipartite graph. In one part (say $B$) we will have a vertex for every component in $V(G) \setminus (Z_v \cup \{v\})$ that contains a neighbor of $v$.




**hitting set that excludes v** $Z_v$

To apply $2$-expansion lemma we need a bipartite graph. In one part (say $B$) we will have a vertex for every component in $V(G) \setminus (Z_v \cup \{v\})$ that contains a neighbor of $v$. The other part $A$ will be $Z_v$.




hitting set that excludes v : $Z_v$

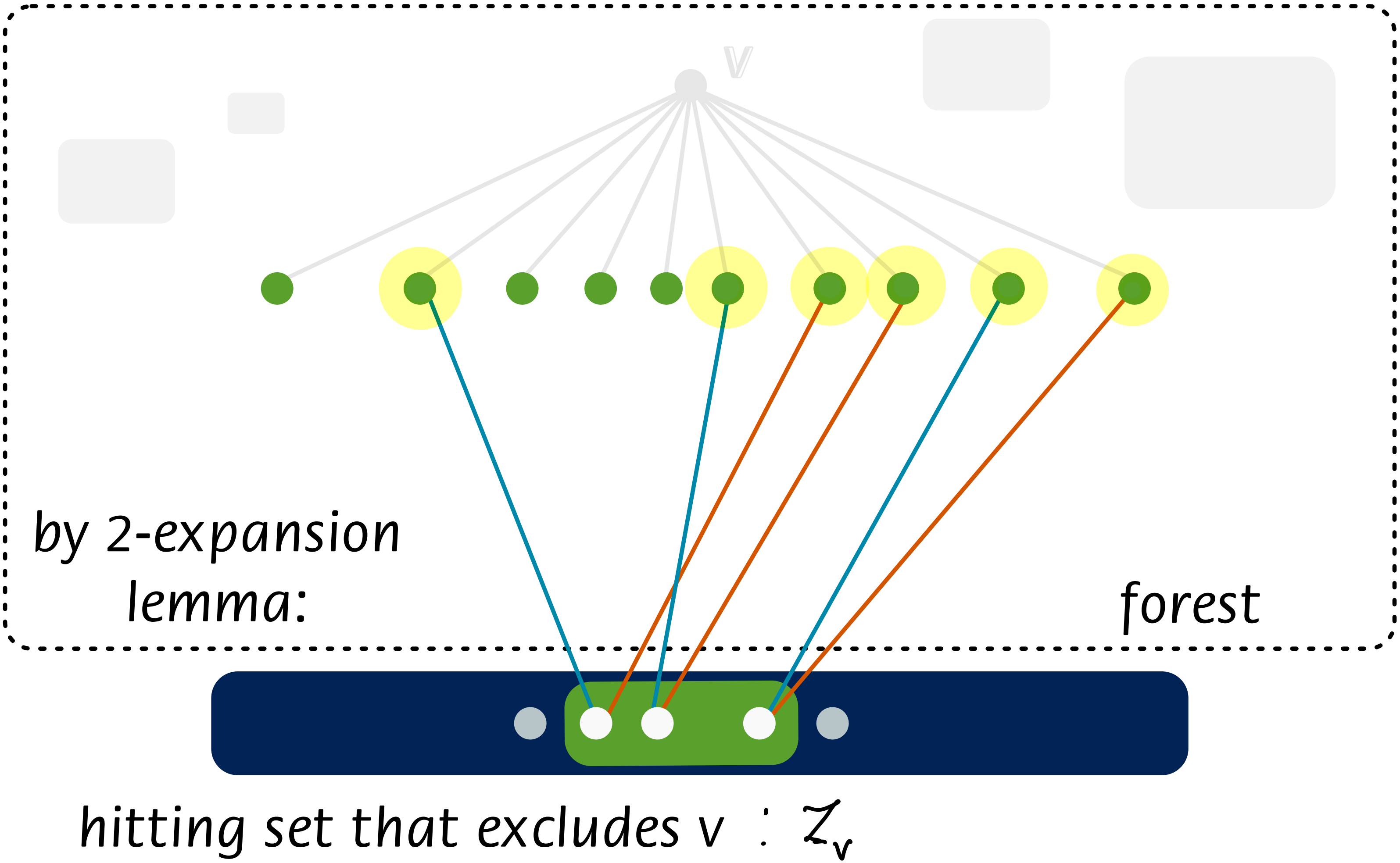
- So we have $A$ and B. We put an edge between a vertex $x$ in $A$ and a vertex $w$ in B, if $x$ is adjacent to some vertex in the component represented by the vertex $w$. Essentially, we have obtained this bipartite graph by contracting the components.
-

- So we have $A$ and B. We put an edge between a vertex $x$ in $A$ and a vertex $w$ in B, if $x$ is adjacent to some vertex in the component represented by the vertex $w$. Essentially, we have obtained this bipartite graph by contracting the components.
- If $|B| < 2|A| \leqslant 6k$ then $v$ already has its degree bounded by $6k+3k+k$. So assume that

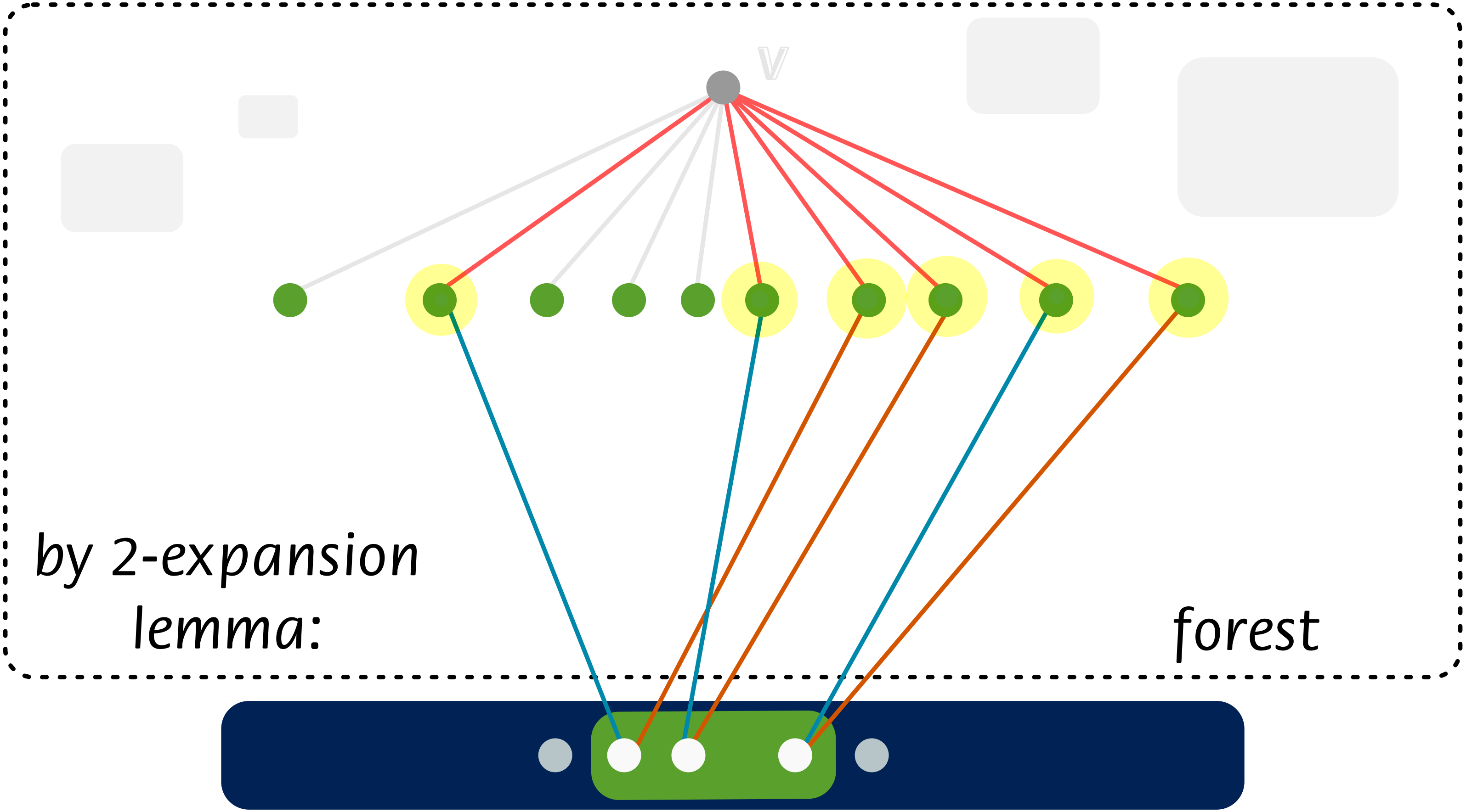
$$|B| > 2|A|$$

Now by expansion lemma (applied with $q = 2$ ) we have that there exist nonempty vertex sets $X \subseteq A$ and $Y \subseteq B$ such that

- there is a 2-expansion of $X$ into $Y$, and
- no vertex in $Y$ has a neighbor outside $X$, that is, $N(Y) \subseteq X$.

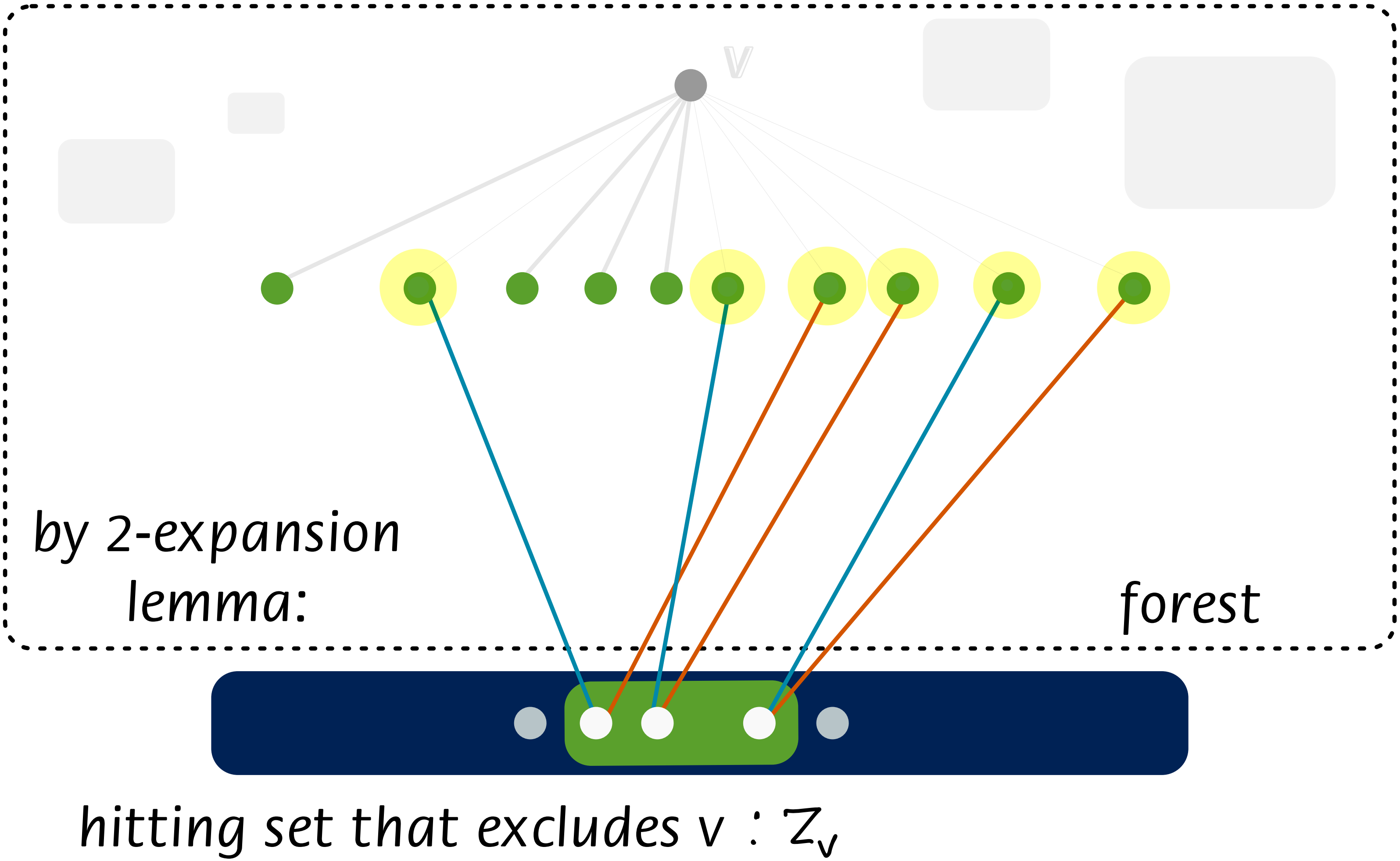


v
by 2-expansion
lemma:
forest
hitting set that excludes v : $\mathcal{Z}_v$

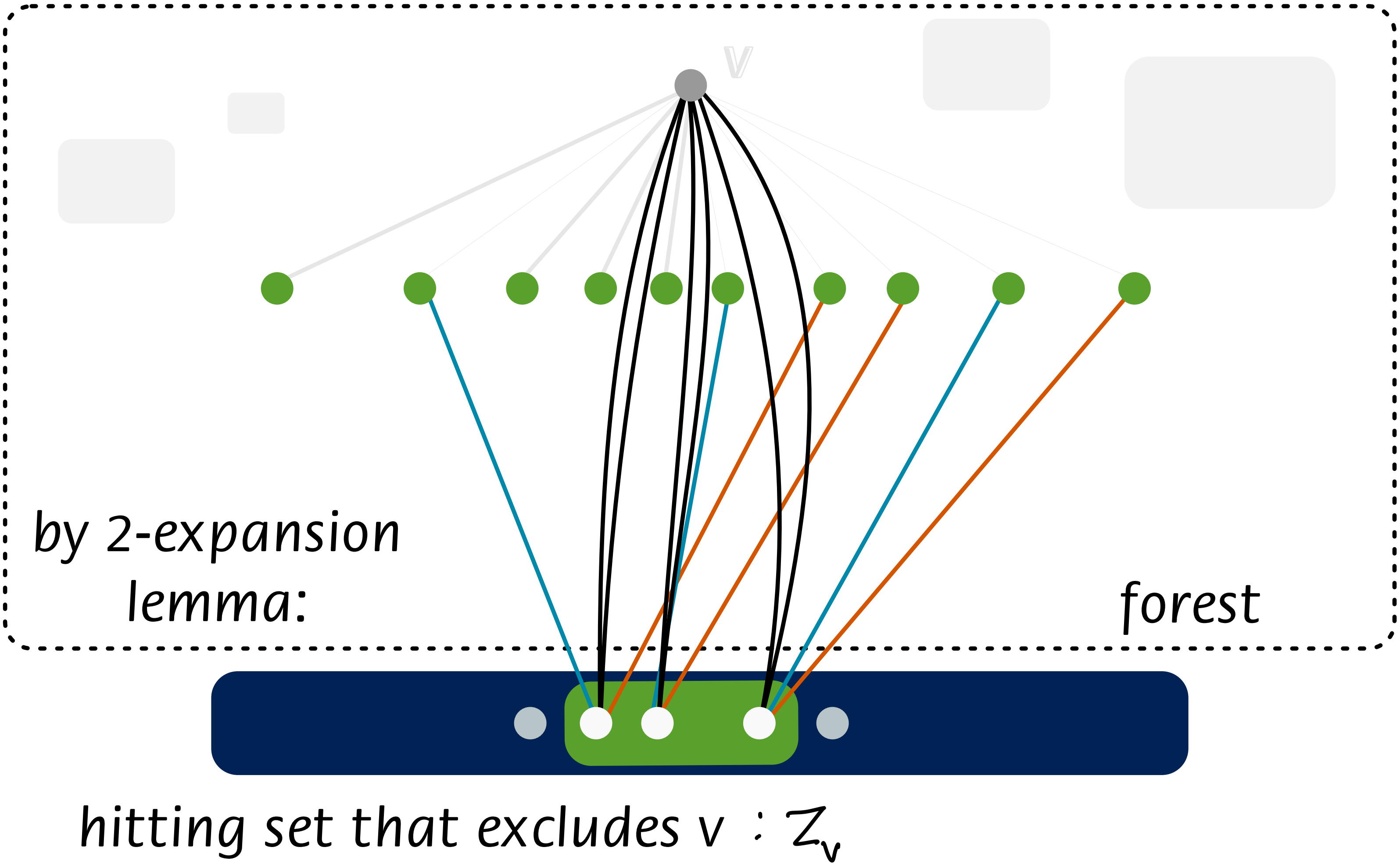



hitting set that excludes v : $Z_v$

So the reduction rule is:


v
by 2-expansion
lemma:
forest
hitting set that excludes v : $Z_v$

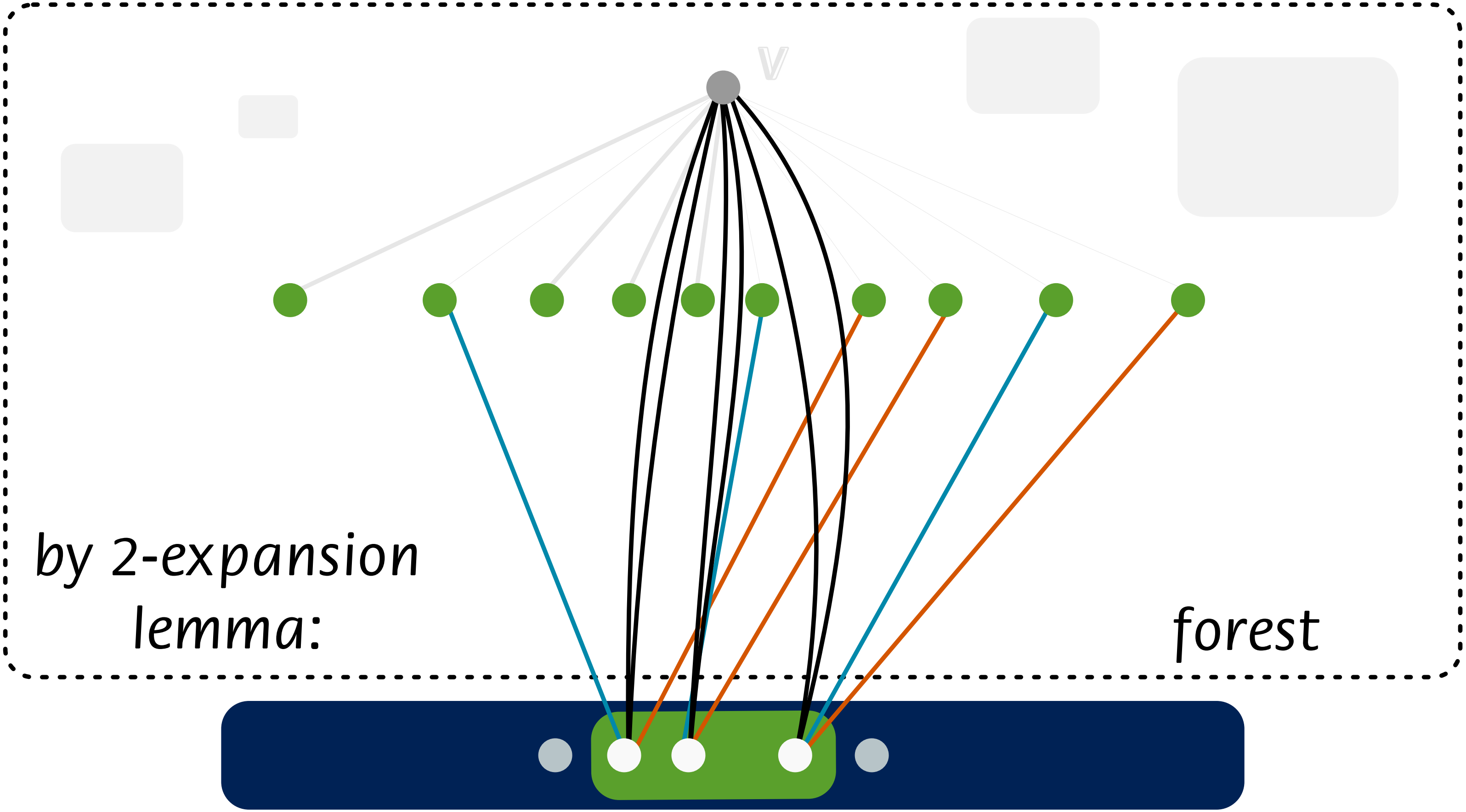
… and add the following edges if already not present.


v
by 2-expansion lemma:
forest
hitting set that excludes v : $Z_v$

Let us argue correctness!

# The Forward Direction

$$\text{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{G} \Rightarrow \text{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{H}$$

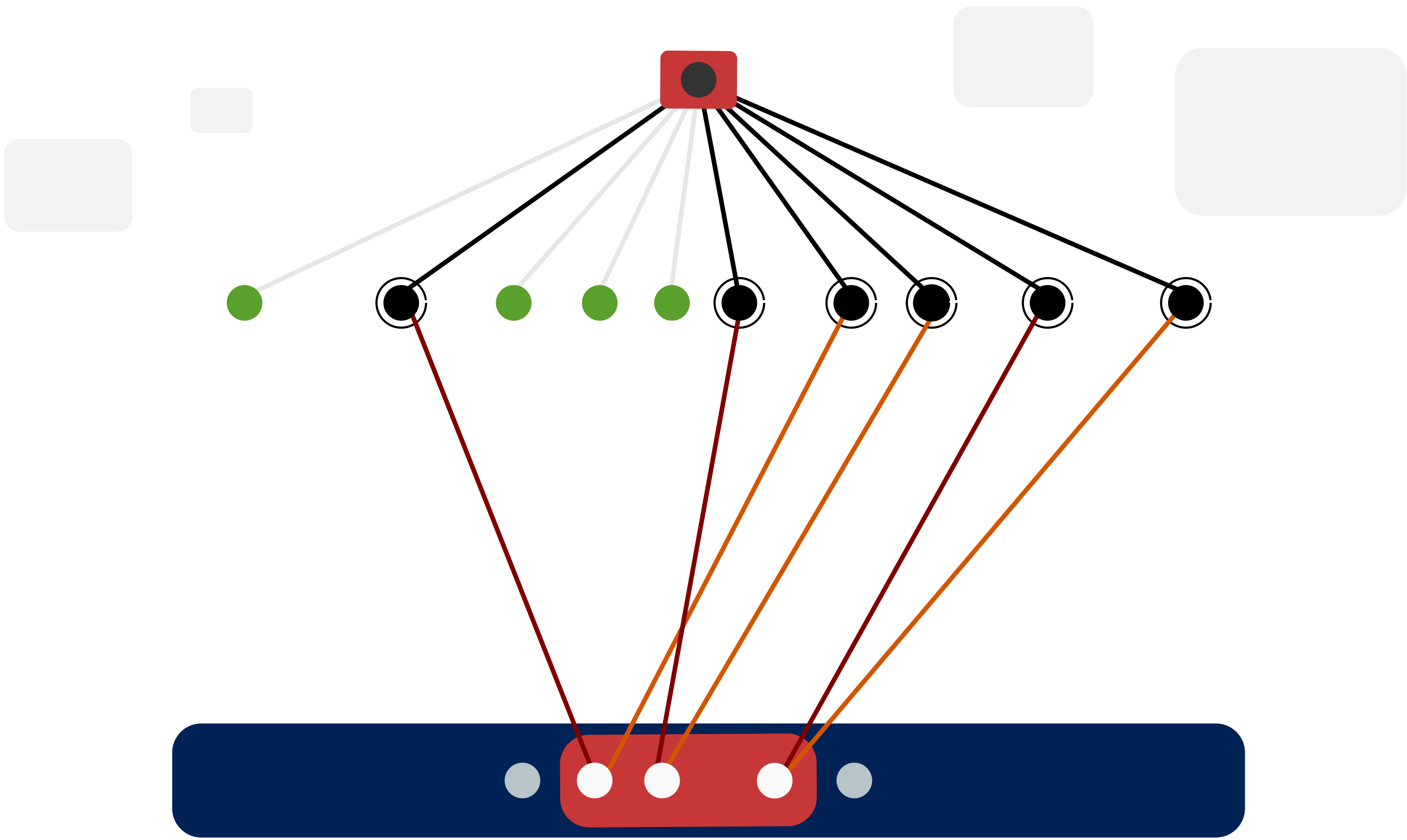


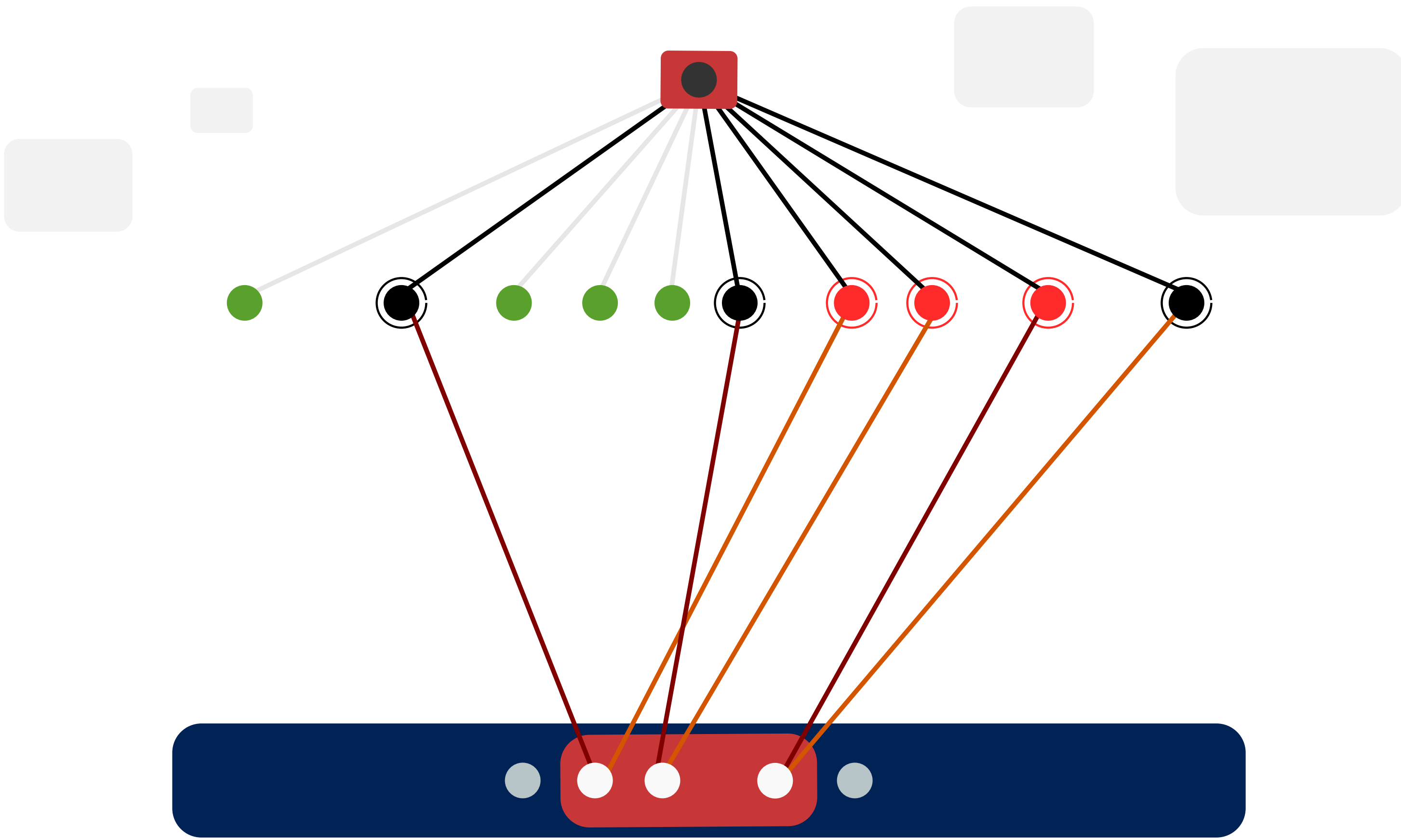

hitting set that excludes v : $Z_v$

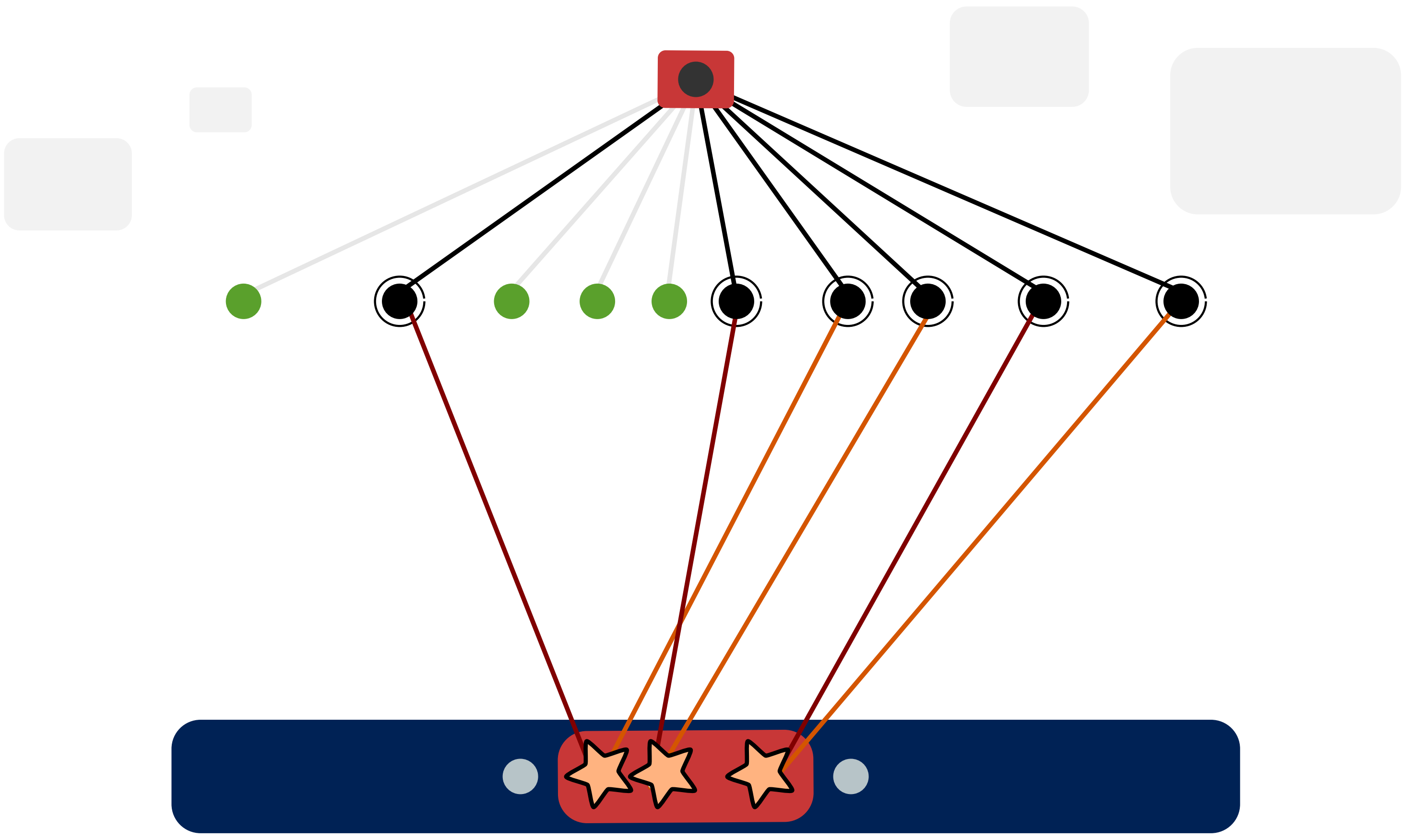
If $G$ has a FVS that either contains $v$ or all of $X$, we are in good shape.

Consider now a FVS that:

- Does not contain $v$,
- and omits at least one vertex of X.







Notice that this does not lead to a larger FVS:

For every vertex $v$ in $X$ that a FVS of $G$ leaves out,

it must pick a vertex $u$ that kills no more than all of $X$.

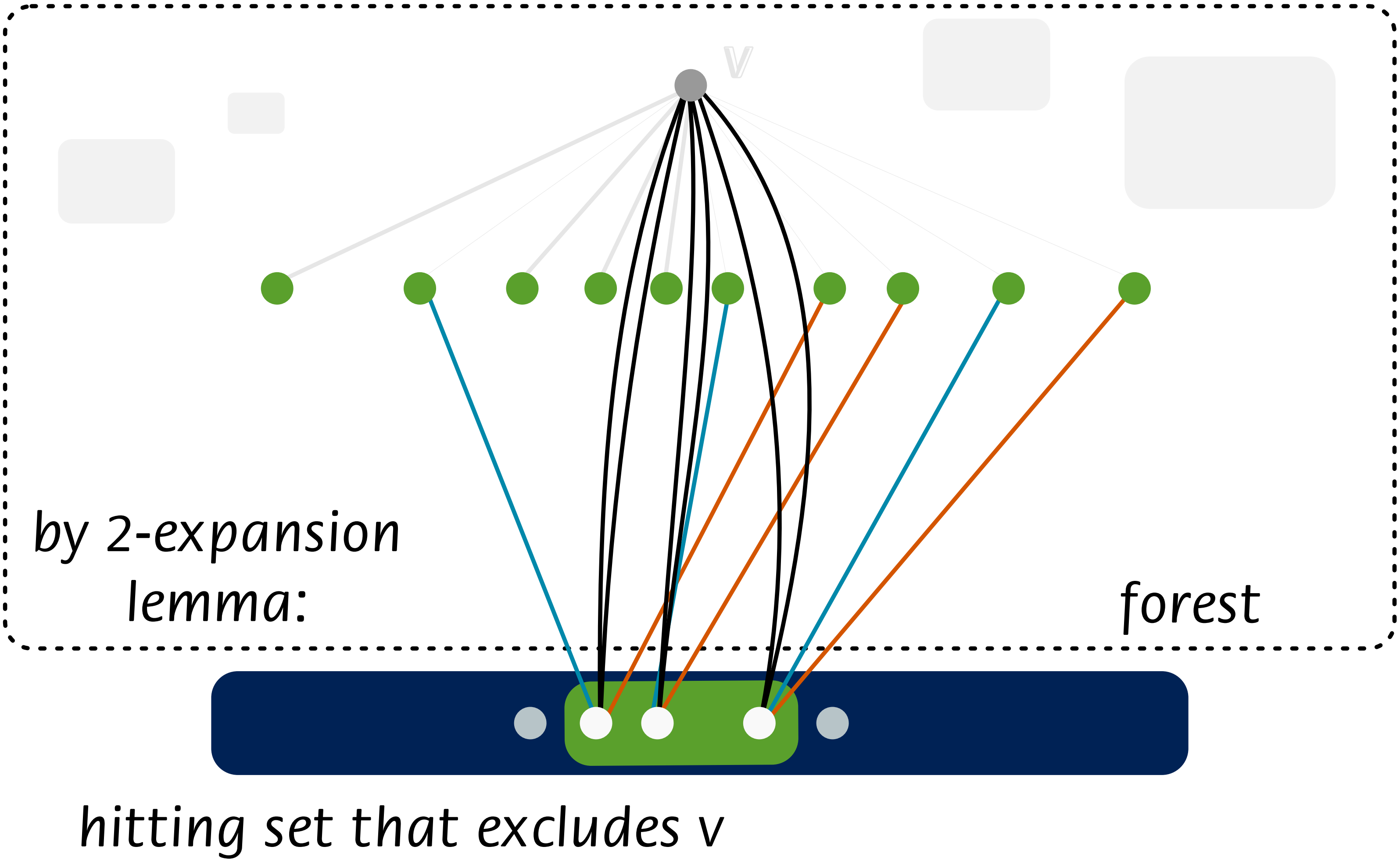
The Reverse Direction

$$\text{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{G} \Leftarrow \text{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{H}$$

# The Reverse Direction

$$\mathsf{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{G} \Leftarrow \mathsf{FVS} \leqslant \mathsf{k} \text{ in } \mathsf{H}$$

If FVS in H contains $v$ then the same works for G also as $\mathsf{G} \setminus \{v\}$ is isomorphic to $\mathsf{H} \setminus \{v\}$. So assume that FVS in H does not contain $v$.

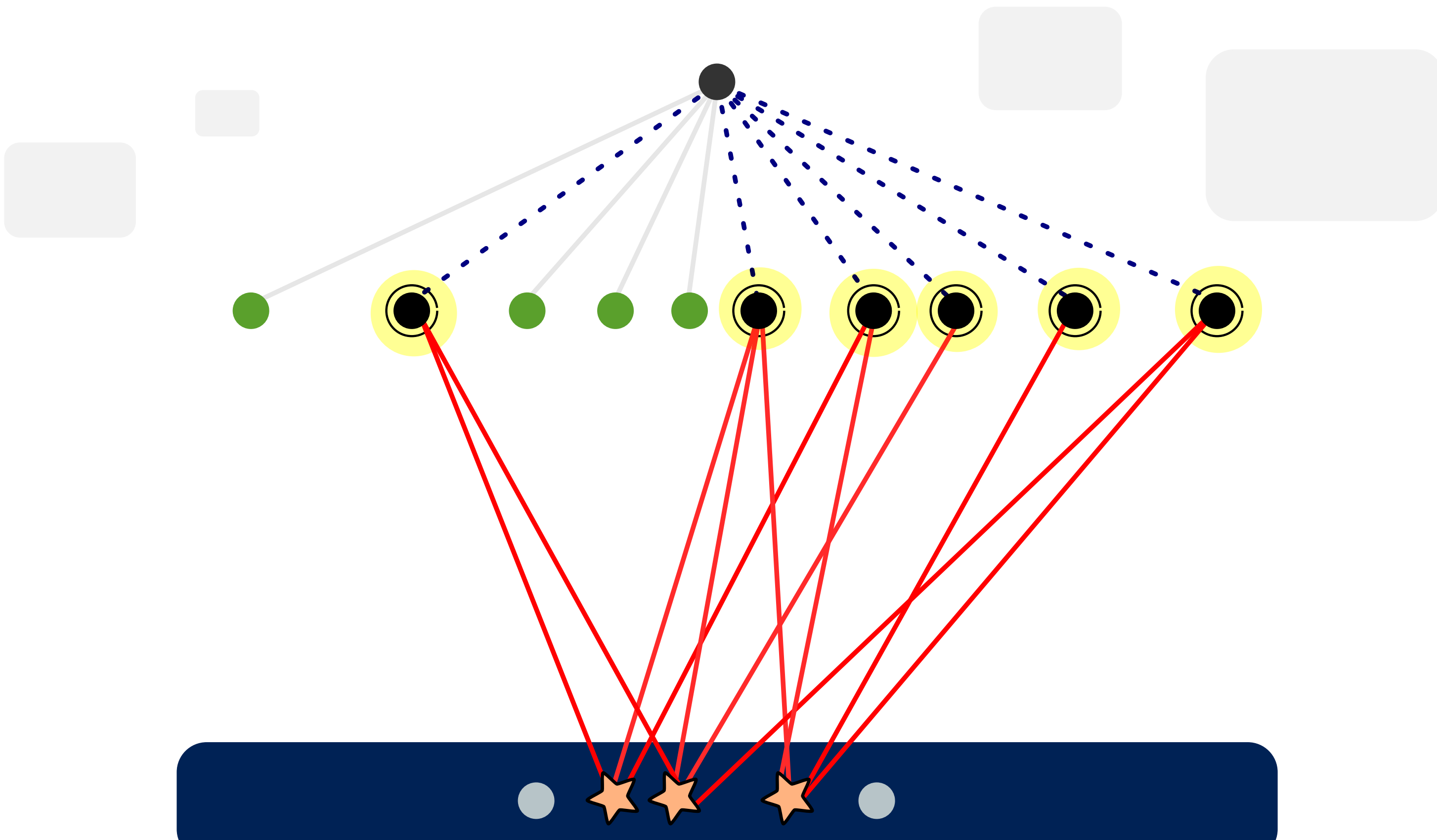


v
by 2-expansion
lemma:
forest
hitting set that excludes v

Let $W$ be a FVS of H, the Only Danger for $W$ to be a FVS of G:

Cycles that:

- pass through $v$,
- non-neighbors of $v$ in H (neighbors in G, however)
- and do not pass through X.

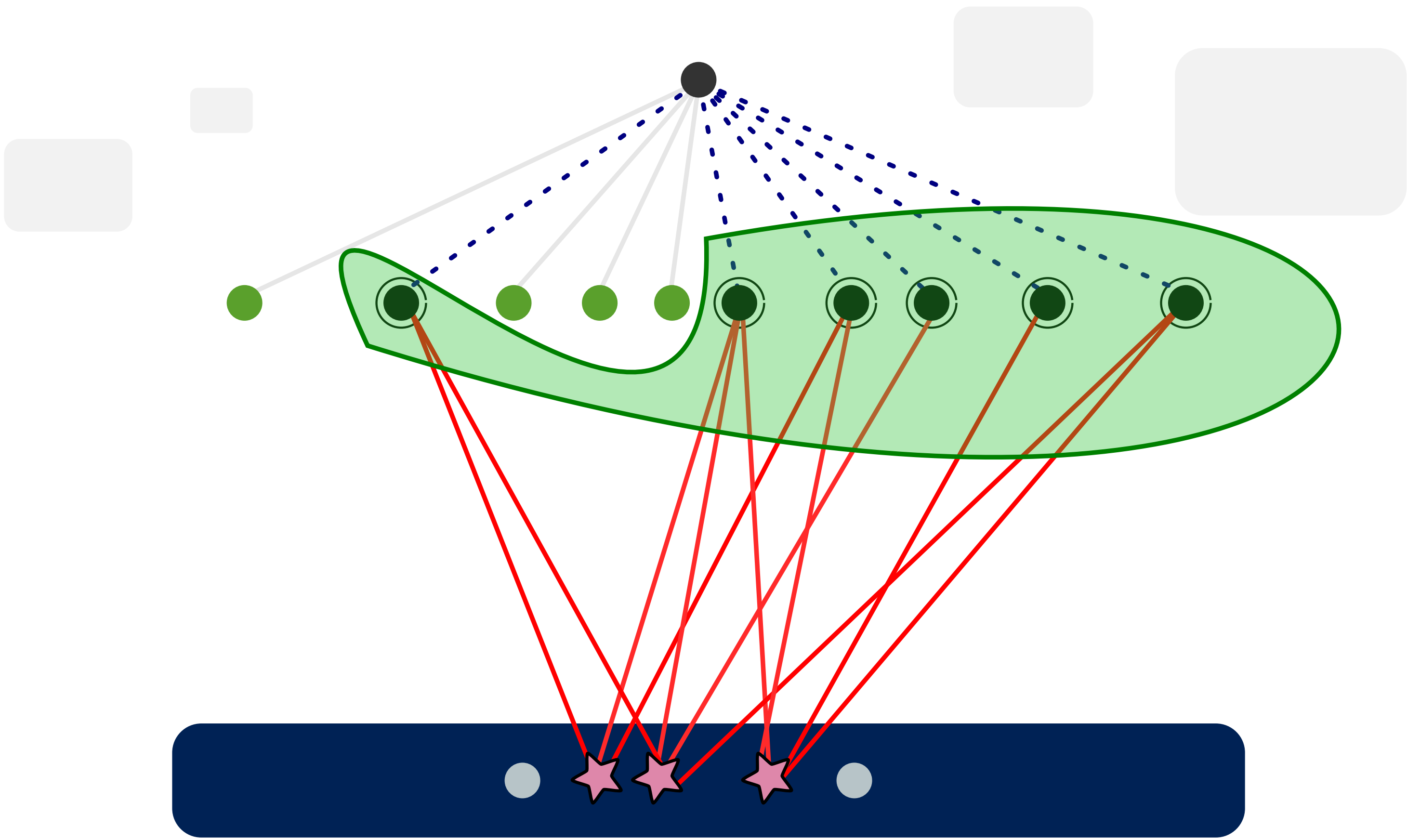


Let $W$ be a FVS of H, the Only Danger for $W$ to be a FVS of G:

Cycles that:

- pass through $v$,
- non-neighbors of $v$ in H (neighbors in G, however)
- and do not pass through X.

However recall that $N(Y) \subseteq X$.



# Wrapping Up

- A priori it is not obvious that previous Reduction Rule actually makes some simplification of the graph, since it *substitutes some set of edges with some other set of double edges!*
-

# Wrapping Up

- A priori it is not obvious that previous Reduction Rule actually makes some simplification of the graph, since it *substitutes some set of edges with some other set of double edges!*
- We need to formally prove that the reduction rules cannot be applied infinitely, or superpolynomially many times.

# Final Result

**Theorem**

FEEDBACK VERTEX SET *admits a kernel with at most* $O(k^2)$ *vertices and* $O(k^2)$ *edges.*