Approximation Algorithms

Lecture 6:
k-Center via Parametric Pruning

Part I:
Metric-k-Center

Philipp Kindermann

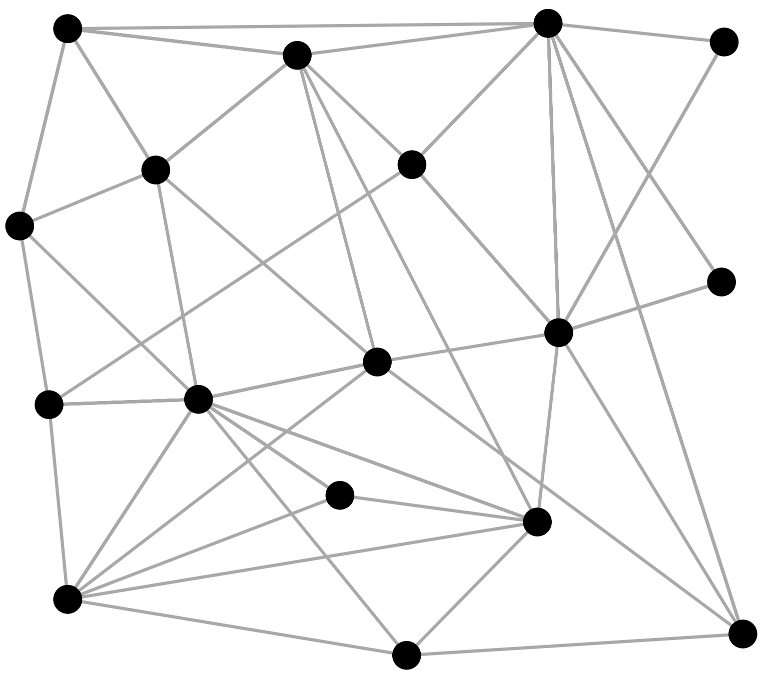
Summer Semester 2020

Metric-k-Center

Given: A graph $G = (V, E)$

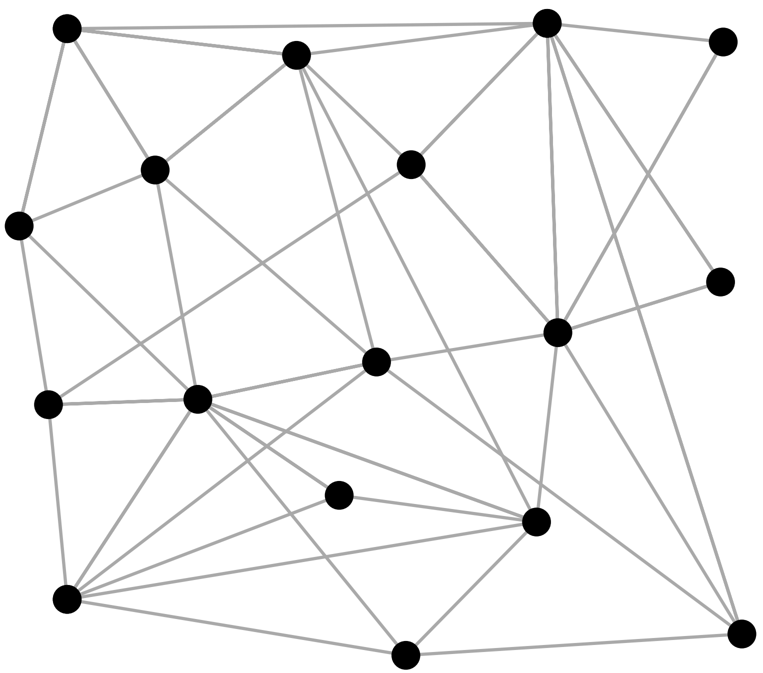
Metric-k-Center

Given: A graph $G = (V, E)$



Metric-k-Center

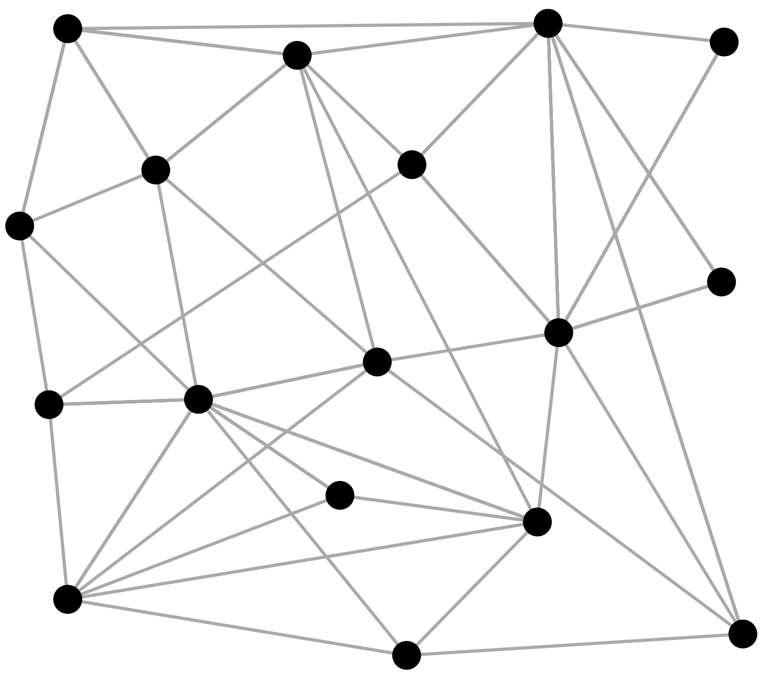
Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality



METRIC-k-CENTER

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

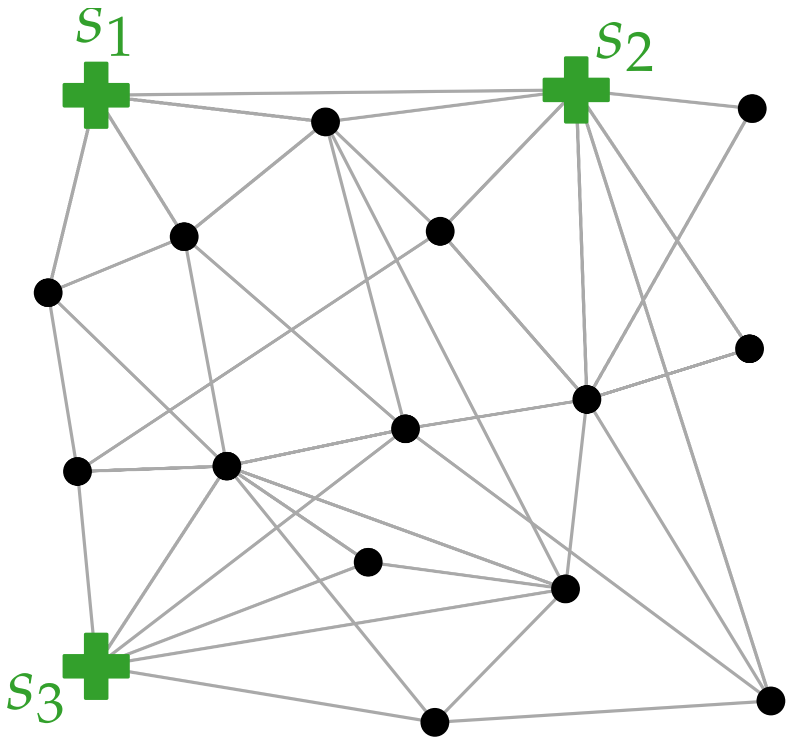
vertex set $S \subseteq V$



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

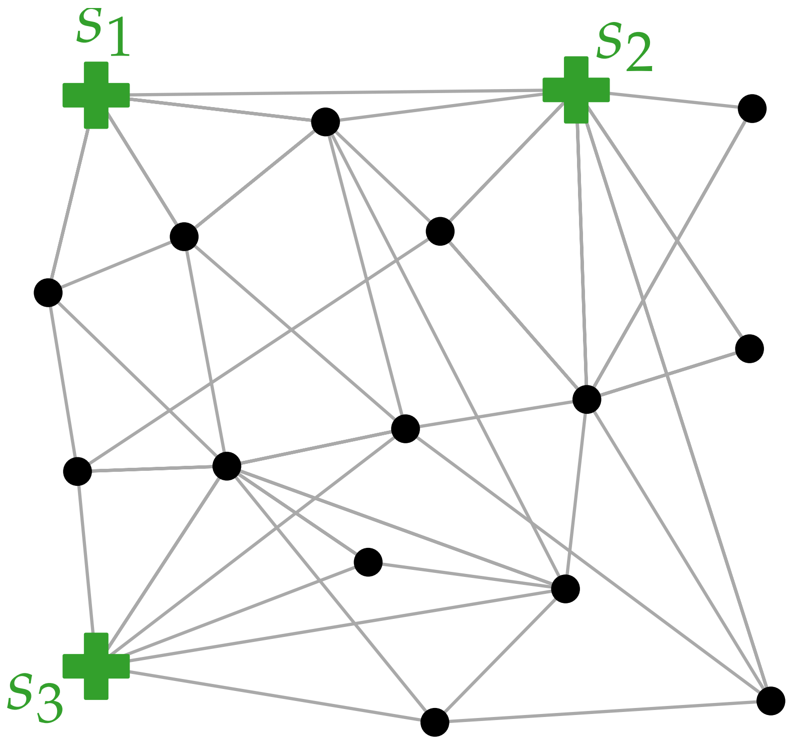
vertex set $S \subseteq V$



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

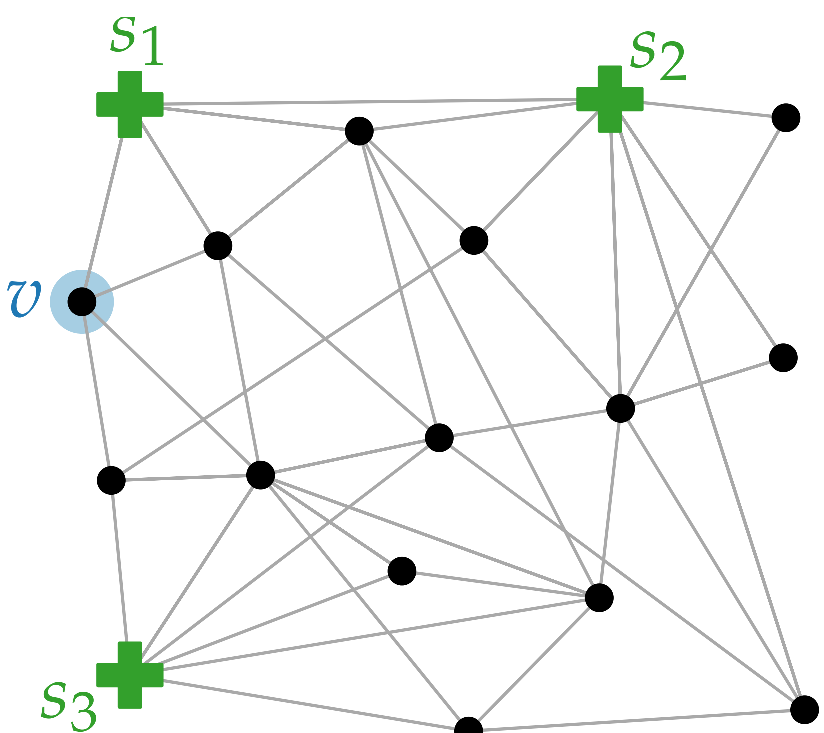
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

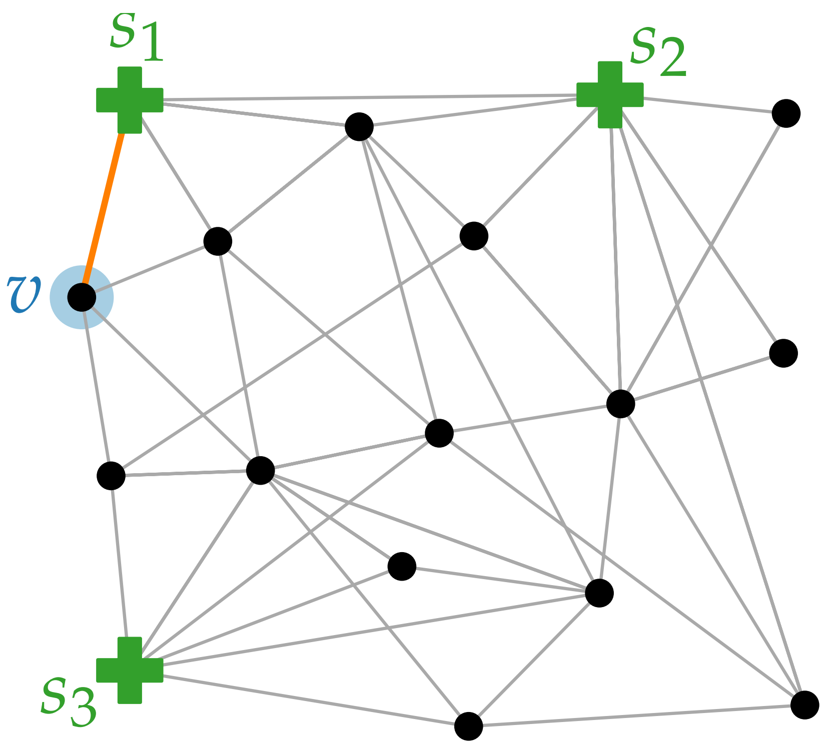
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

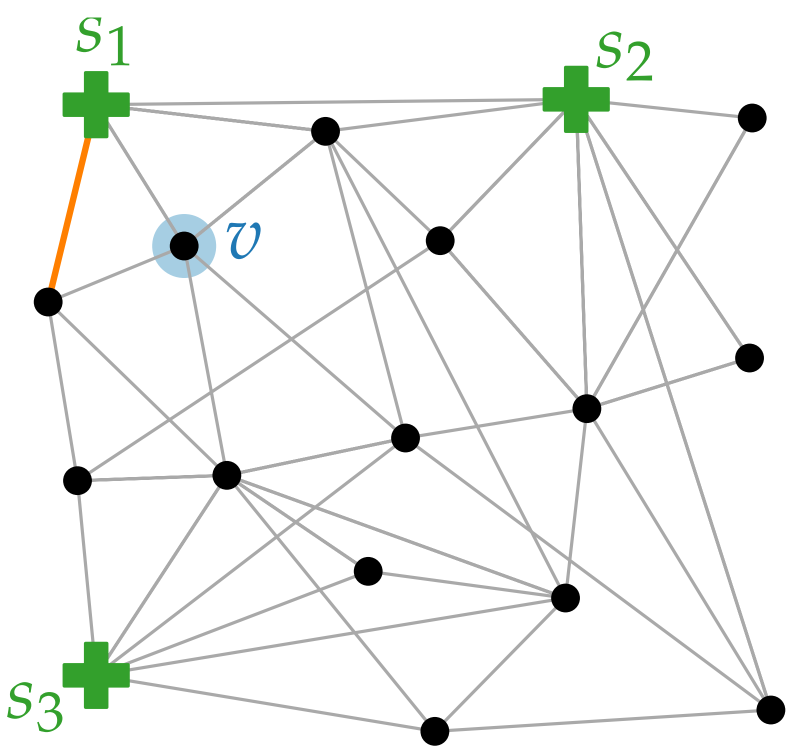
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

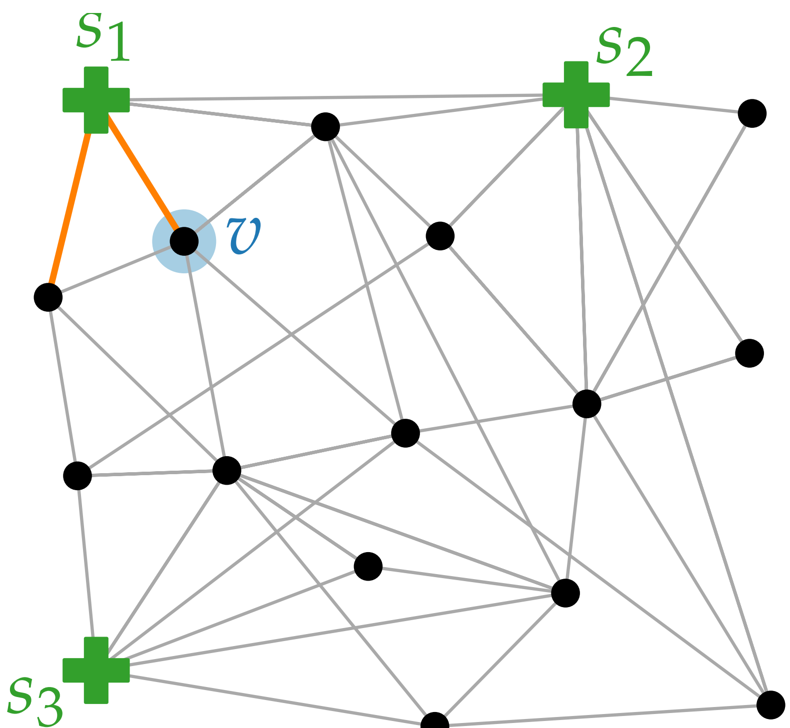
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

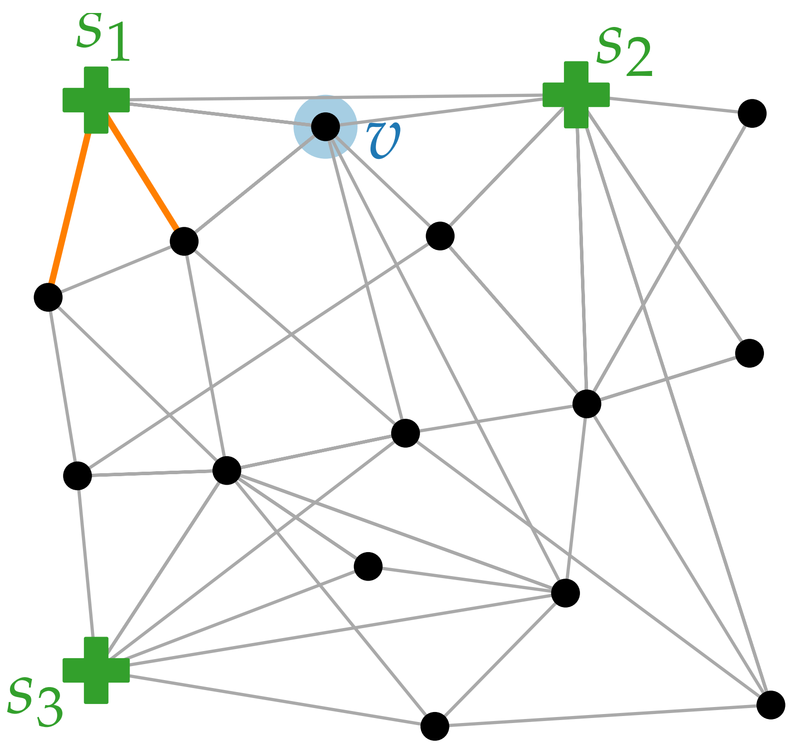
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

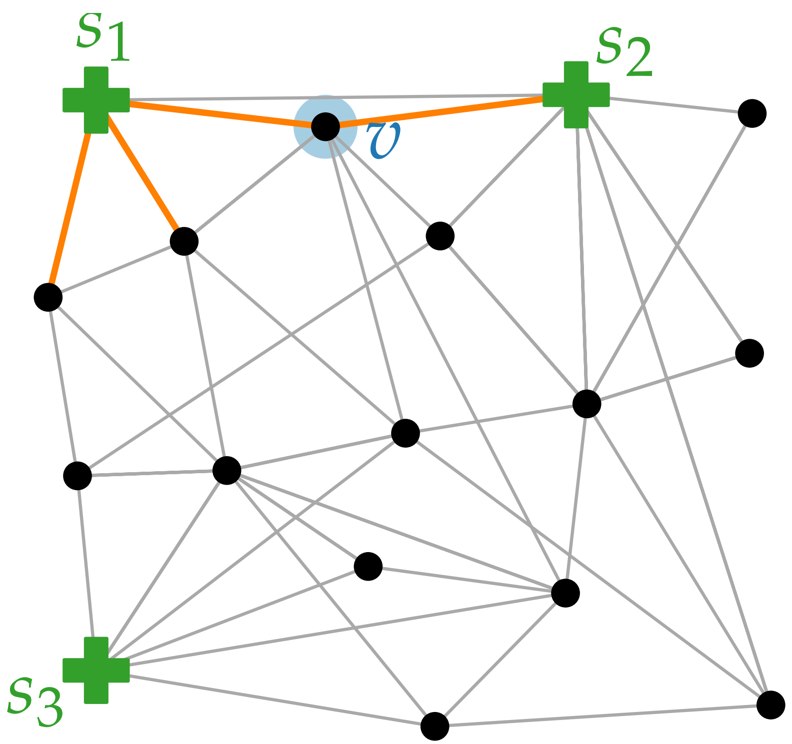
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

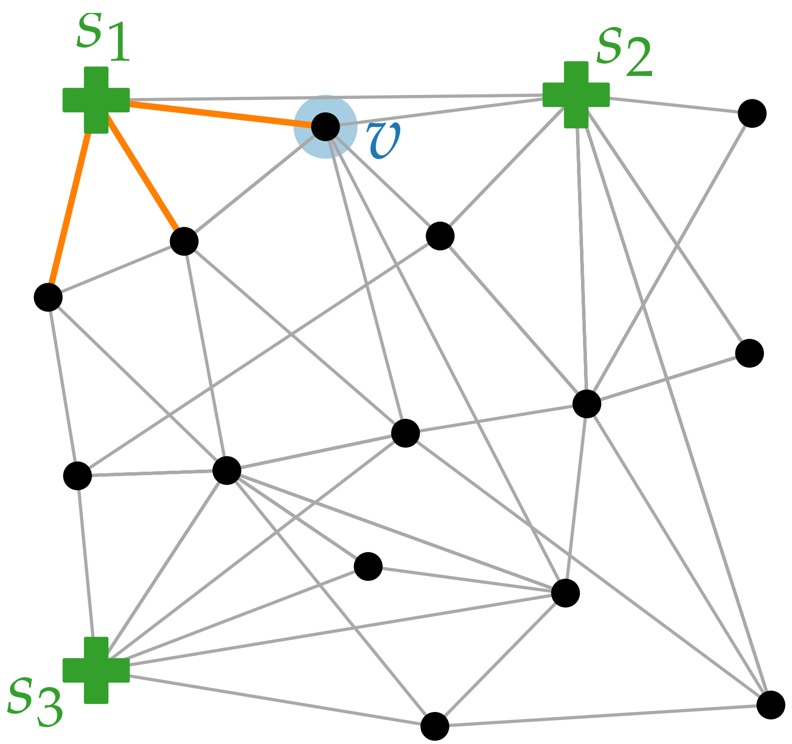
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

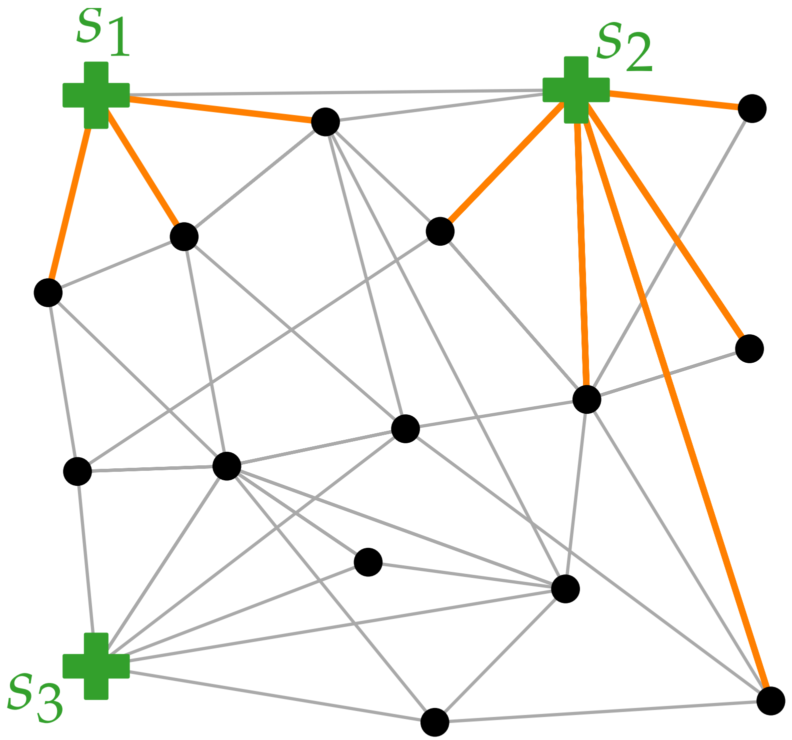
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

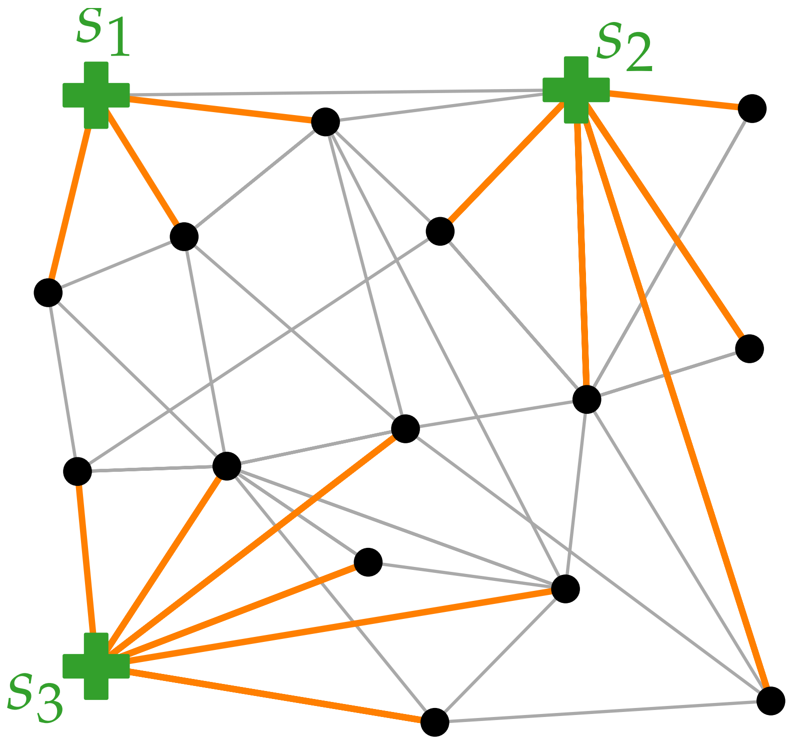
For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.



Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

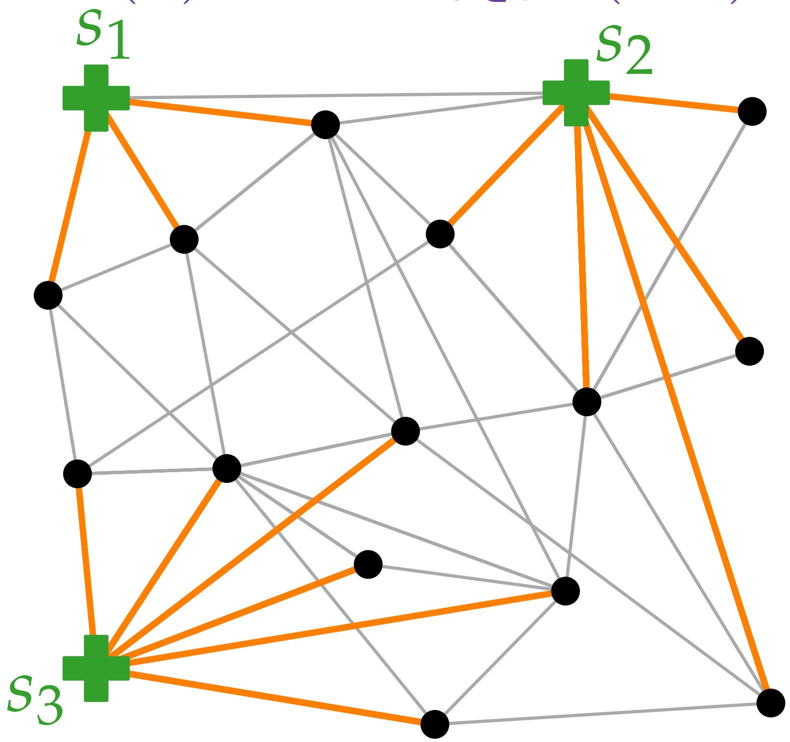


Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

$\text{cost}(S) := \max_{v \in V} c(v, S)$

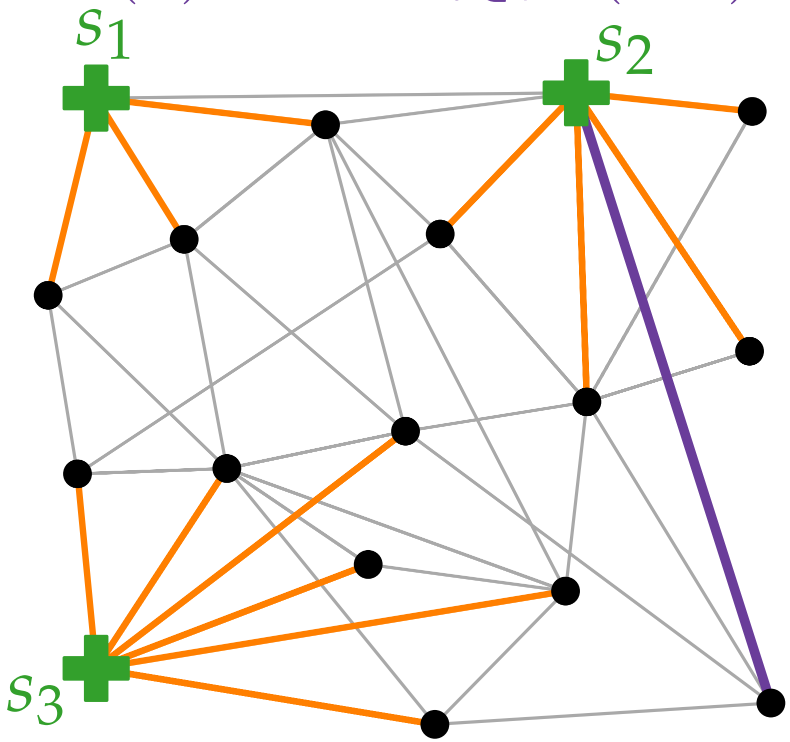


Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

$\text{cost}(S) := \max_{v \in V} c(v, S)$

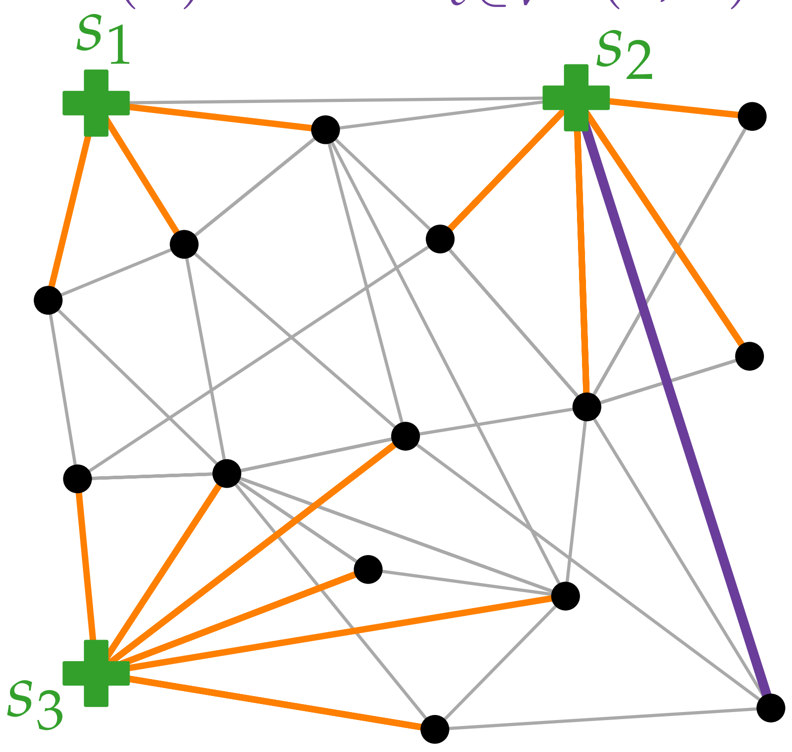


Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A vertex set S, such that $\text{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

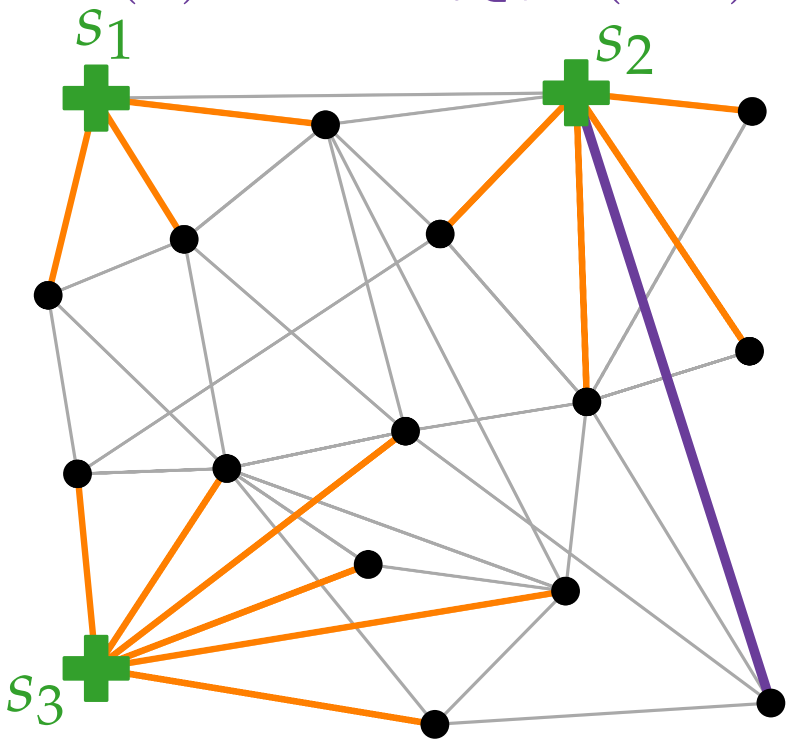


Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A   vertex set S, such that $\mathrm{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

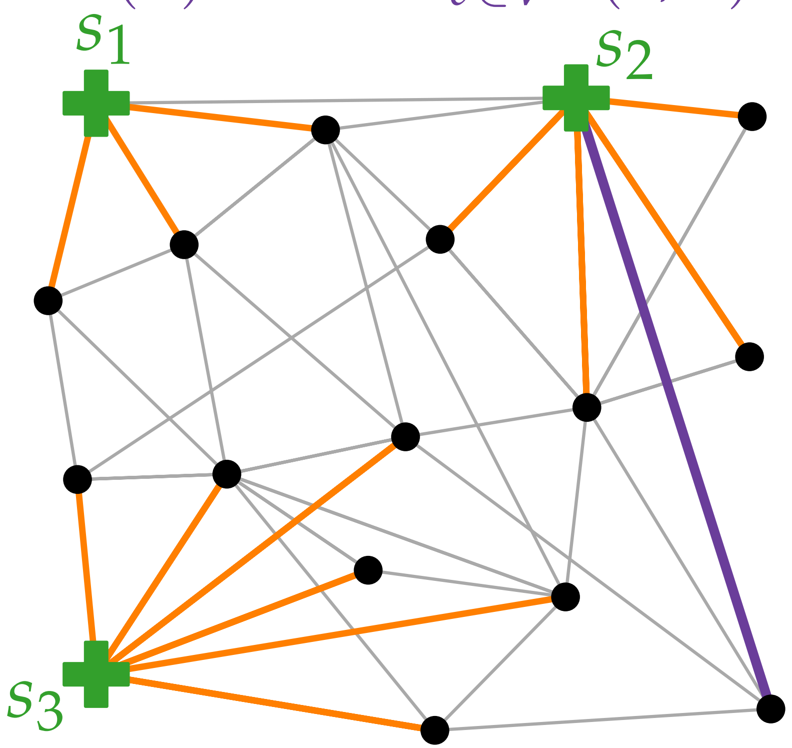


Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A k-element vertex set S, such that $\text{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

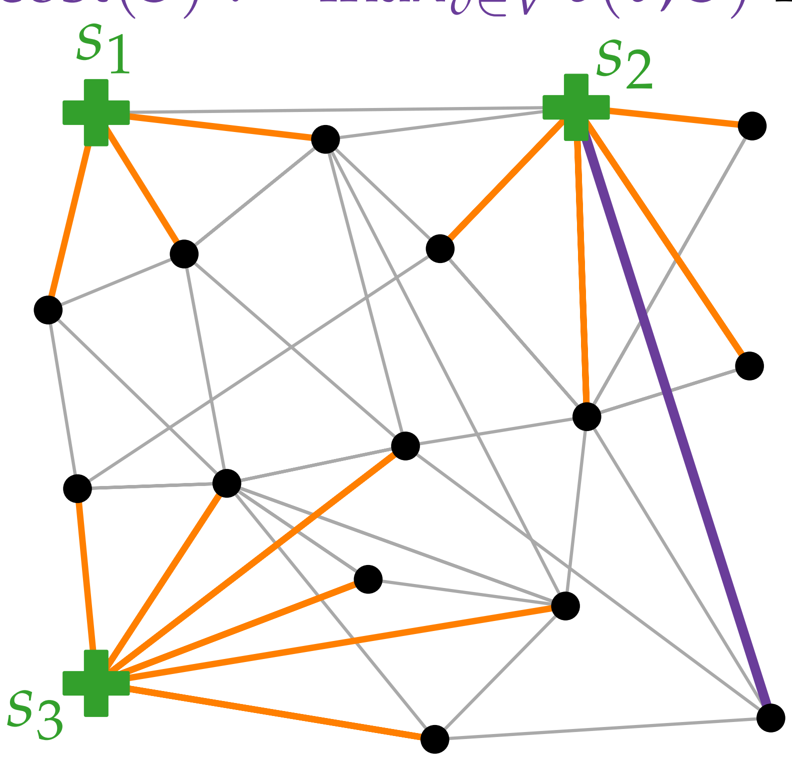


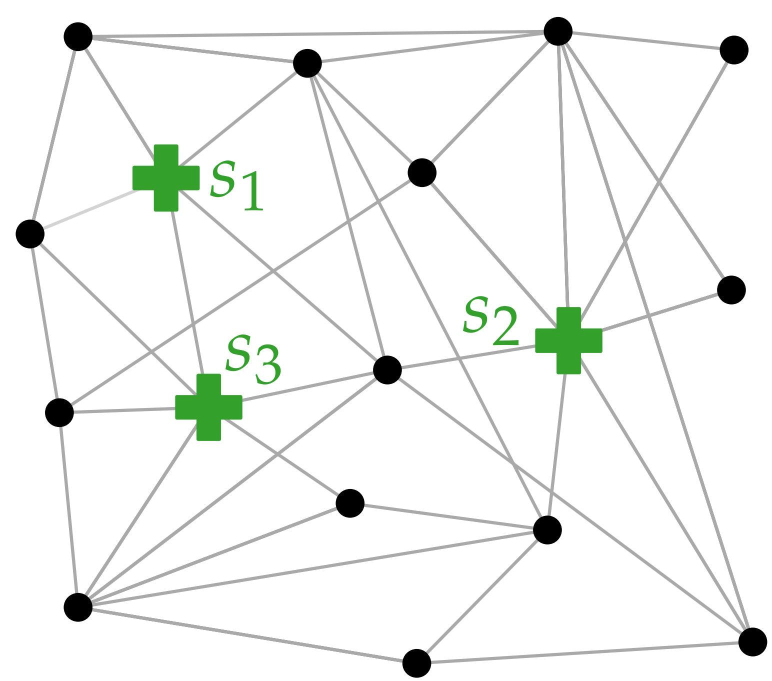
Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A k-element vertex set S, such that $\operatorname{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.



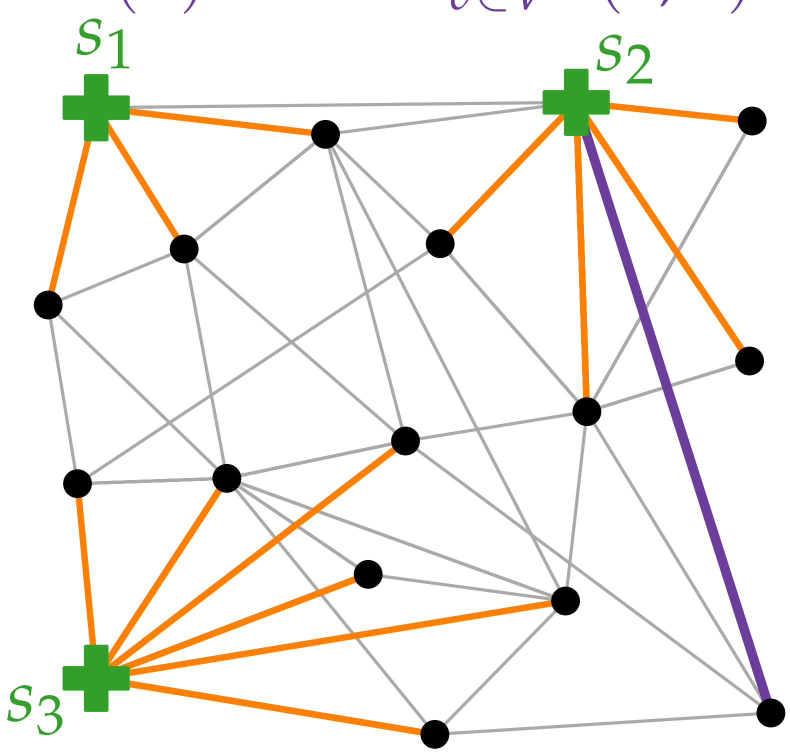


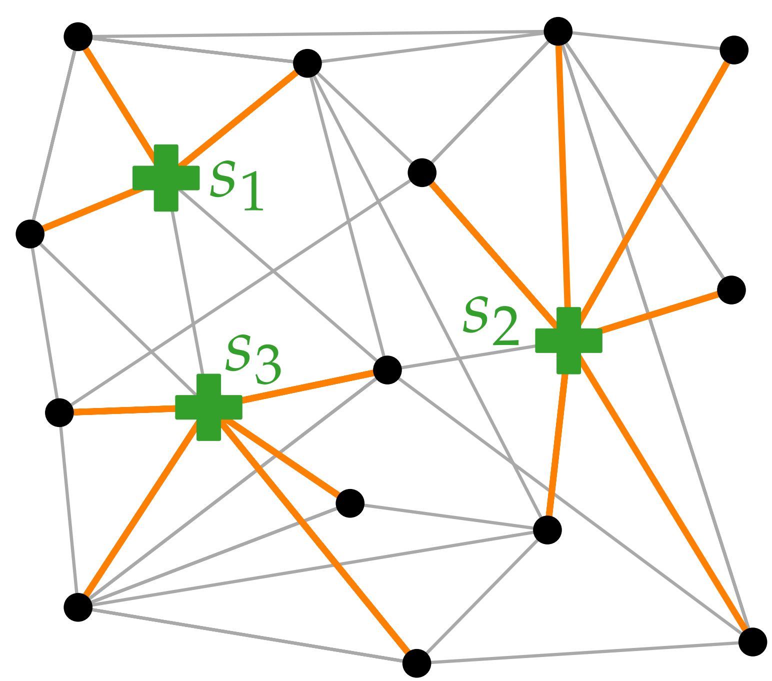
Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A k-element vertex set S, such that $\mathrm{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.



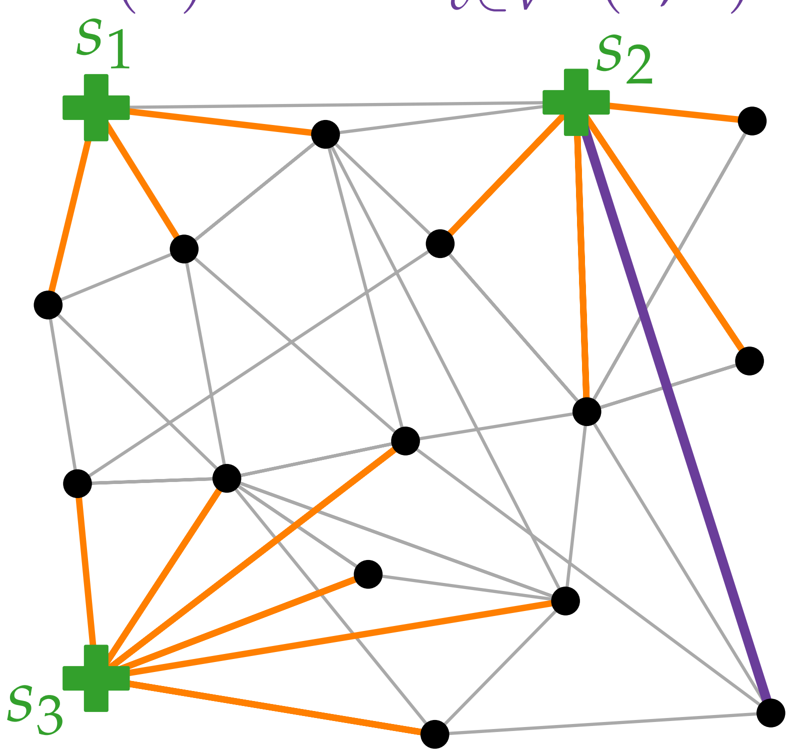


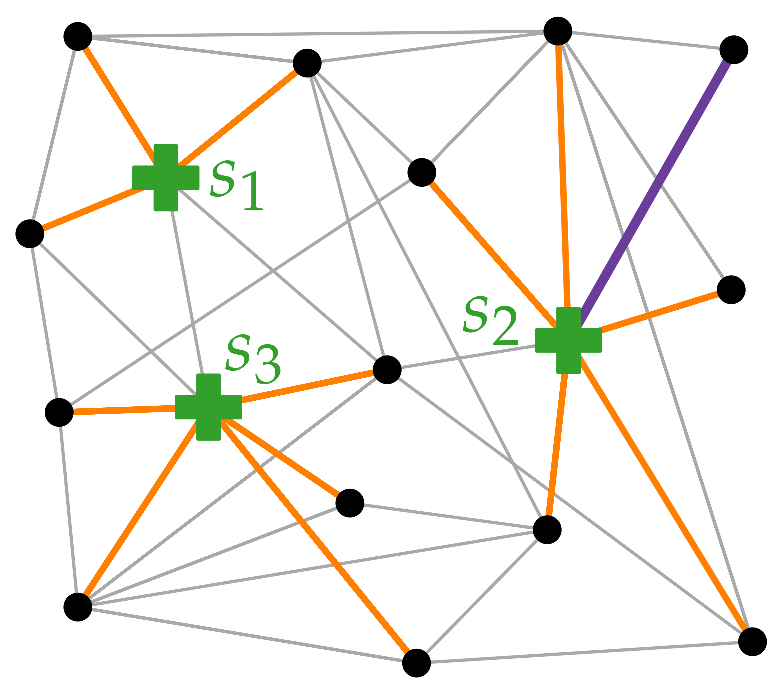
Metric-k-Center

Given: A complete graph $G = (V, E)$ with edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ satisfying the triangle inequality and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to a vertex in S.

Find: A k-element vertex set S, such that $\mathrm{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.





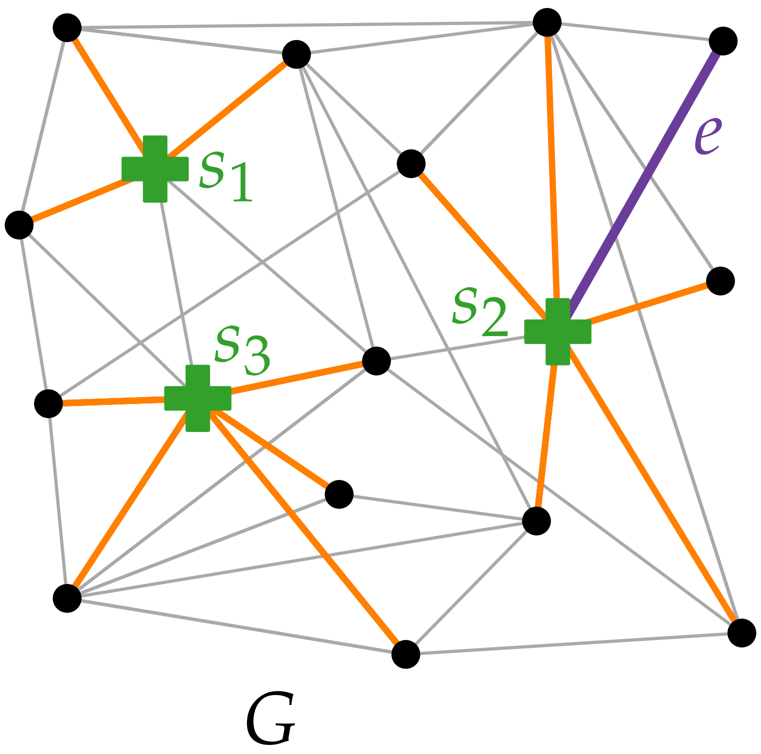
Approximation Algorithms

Lecture 6:
k-Center via Parametric Pruning

Part II:
Parametric Pruning

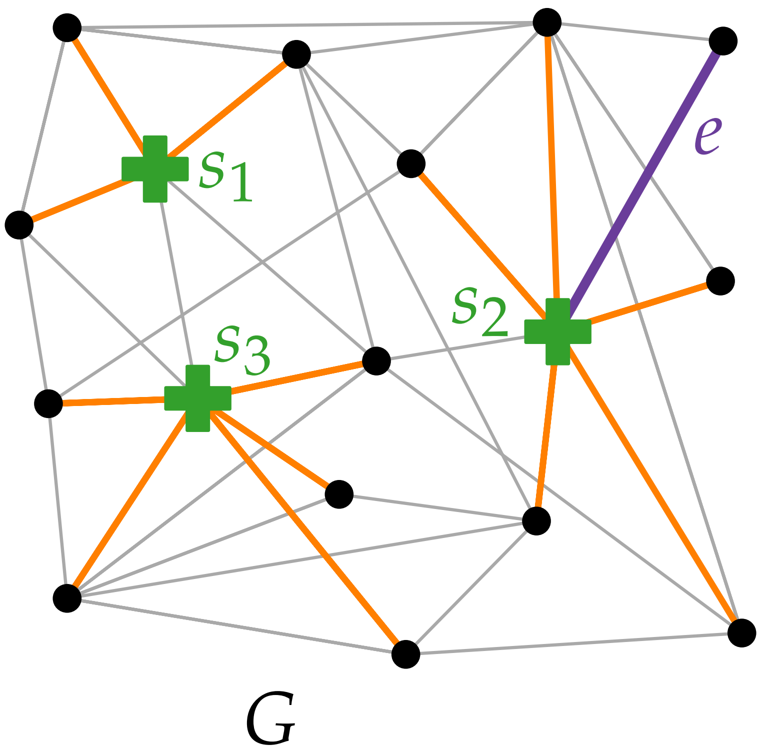
Philipp Kindermann Summer Semester 2020

Parametric Pruning



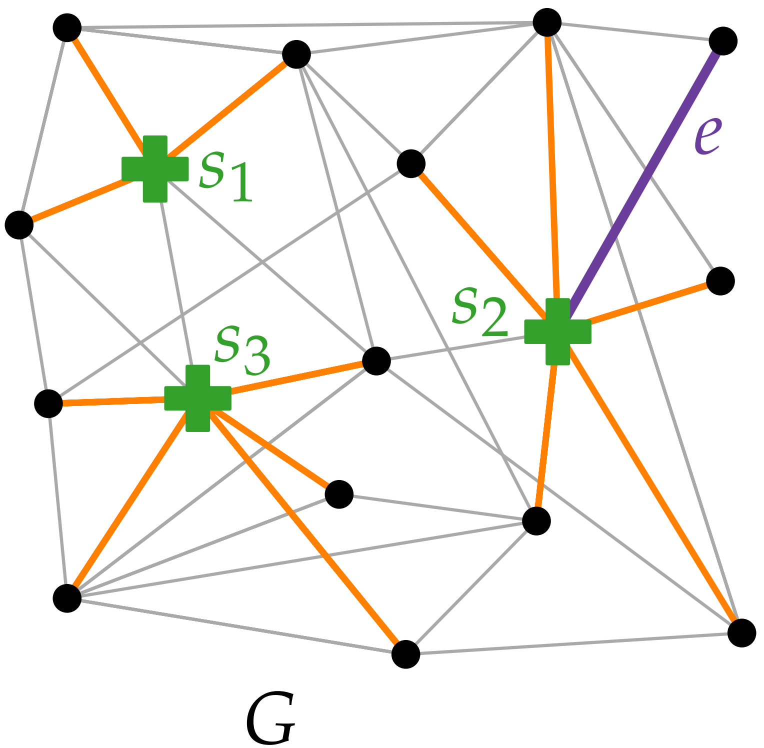
Parametric Pruning

Let $E = \{e_1, \ldots, e_m\}$ with $c(e_1) \leq \ldots \leq c(e_m)$.



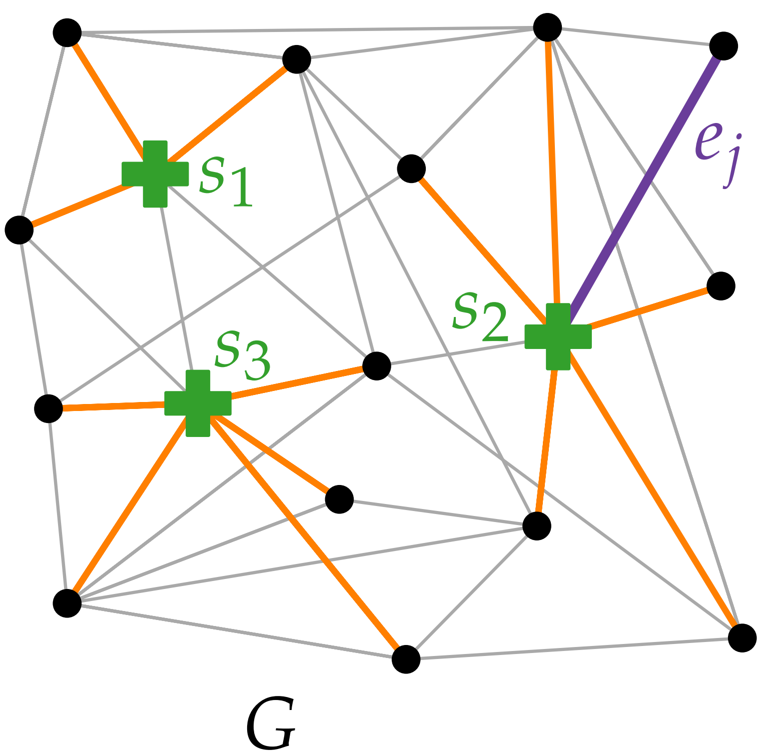
Parametric Pruning

Let $E = \{e_1, \ldots, e_m\}$ with $c(e_1) \leq \ldots \leq c(e_m)$.
Suppose we know that $\text{OPT} = c(e_j)$.



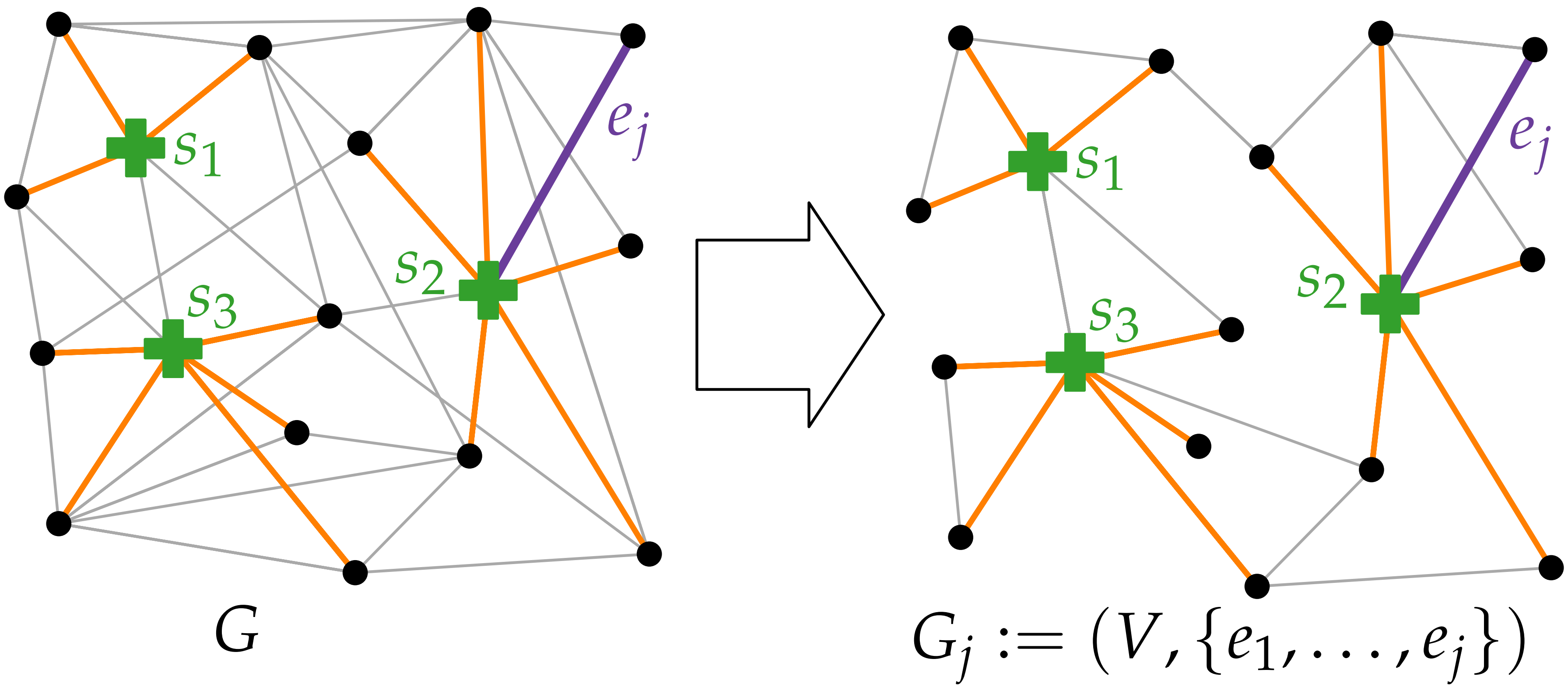
Parametric Pruning

Let $E = \{e_1, \ldots, e_m\}$ with $c(e_1) \leq \ldots \leq c(e_m)$.
Suppose we know that $\mathrm{OPT} = c(e_j)$.



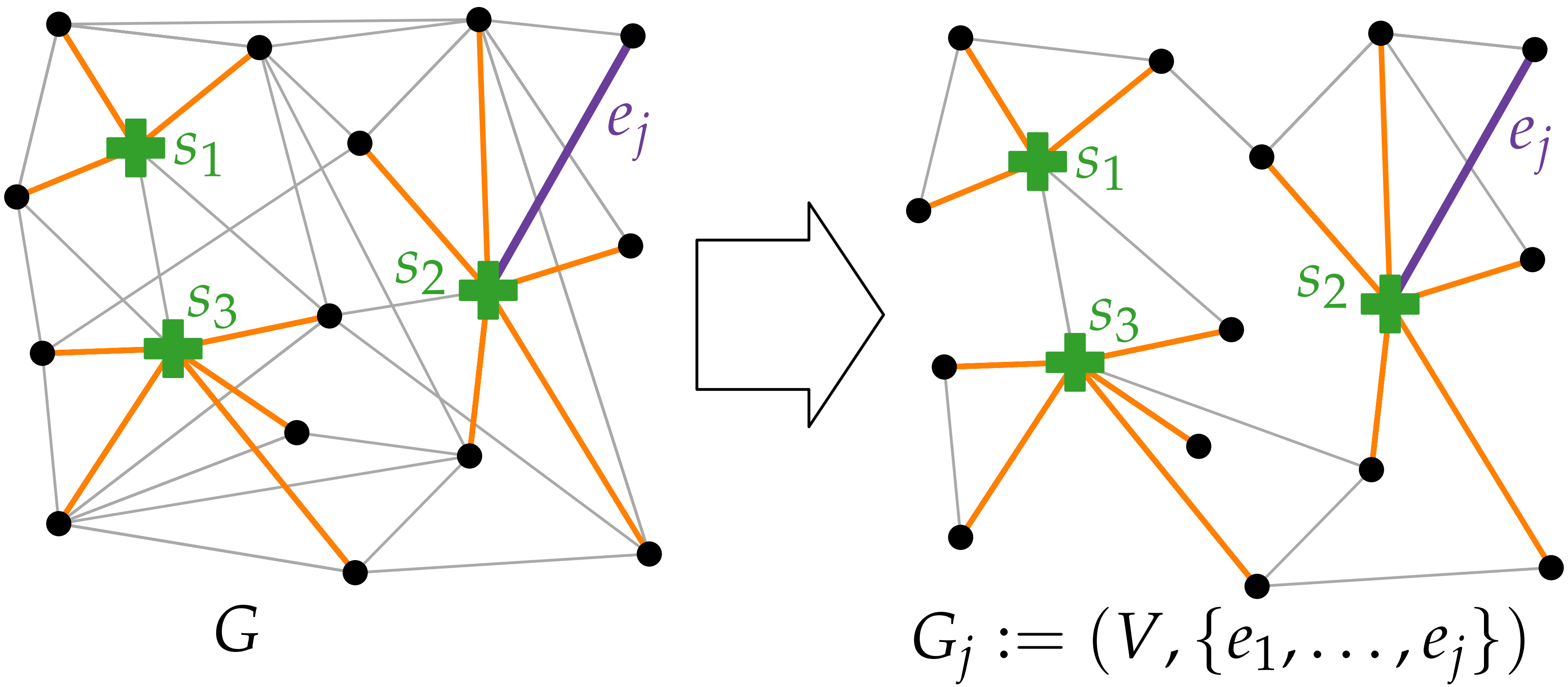
Parametric Pruning

Let $E = \{e_1, \ldots, e_m\}$ with $c(e_1) \leq \ldots \leq c(e_m)$.
Suppose we know that $\mathrm{OPT} = c(e_j)$.



Parametric Pruning

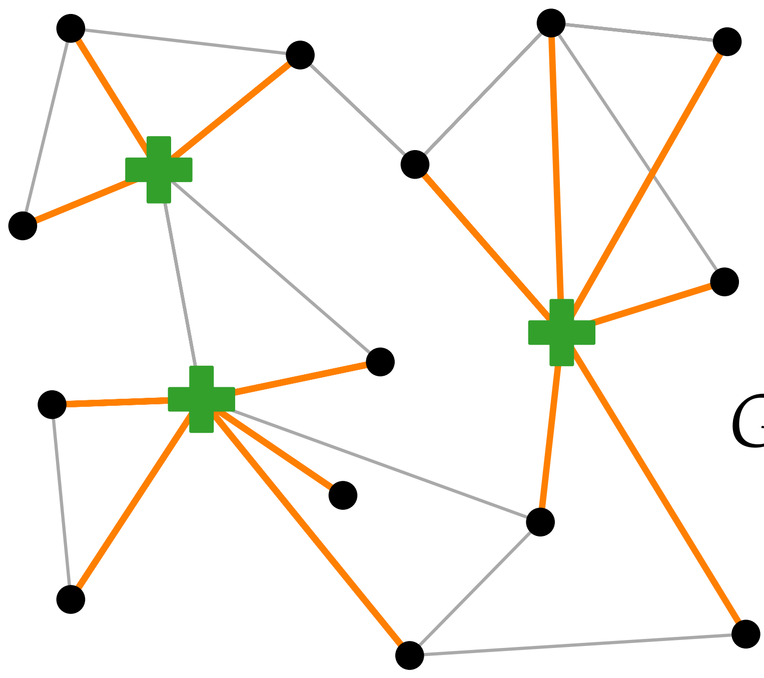
Let $E = \{e_1, \ldots, e_m\}$ with $c(e_1) \leq \ldots \leq c(e_m)$.
Suppose we know that $\text{OPT} = c(e_j)$.



...try each G_j.

. . . try each G_j.

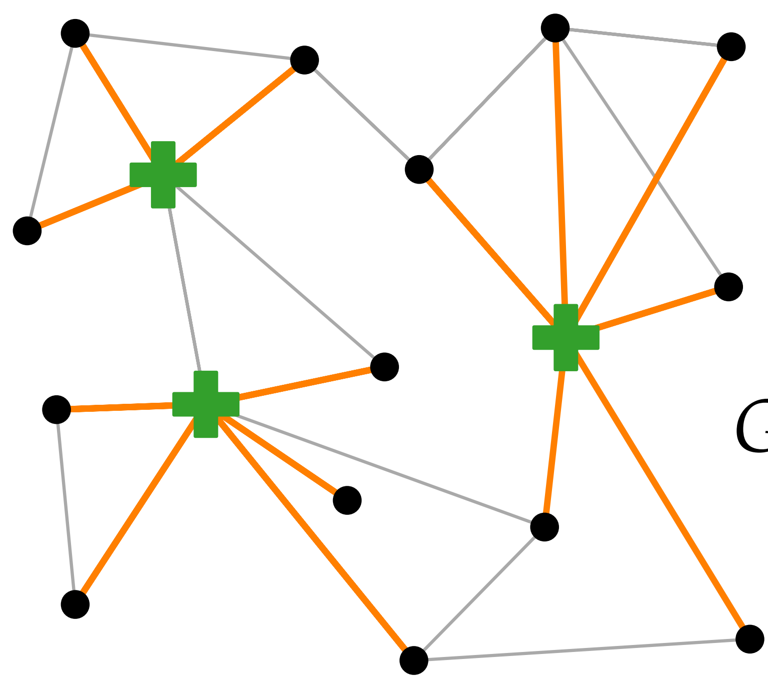
Def.



$$G_j := (V, \{e_1, \ldots, e_j\})$$

. . . try each G_j.

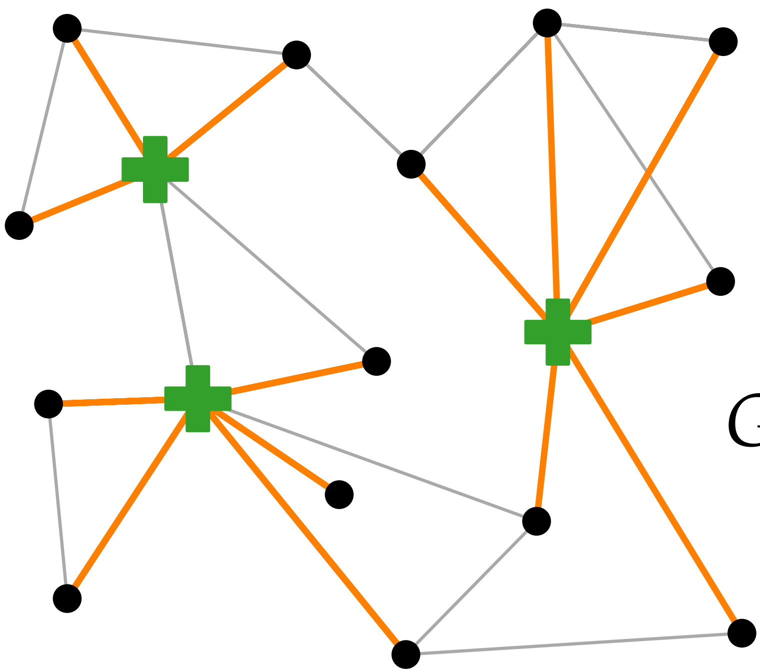
Def. A vertex set D of a graph H is **dominating** if each vertex is either in D or adjacent to a vertex in D.



$G_j := (V, \{e_1, \ldots, e_j\})$

. . . try each G_j.

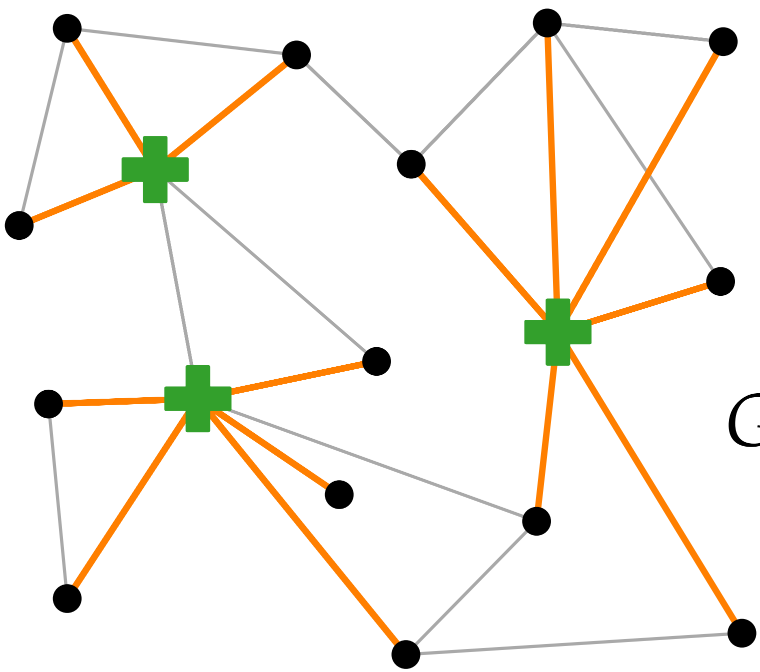
Def. A vertex set D of a graph H is **dominating** if each vertex is either in D or adjacent to a vertex in D. The cardinality of a smallest dominating set in H is denoted by $\text{dom}(H)$.



$$G_j := (V, \{e_1, \dots, e_j\})$$

. . . try each G_j.

Def. A vertex set D of a graph H is **dominating** if each vertex is either in D or adjacent to a vertex in D. The cardinality of a smallest dominating set in H is denoted by $\mathrm{dom}(H)$.

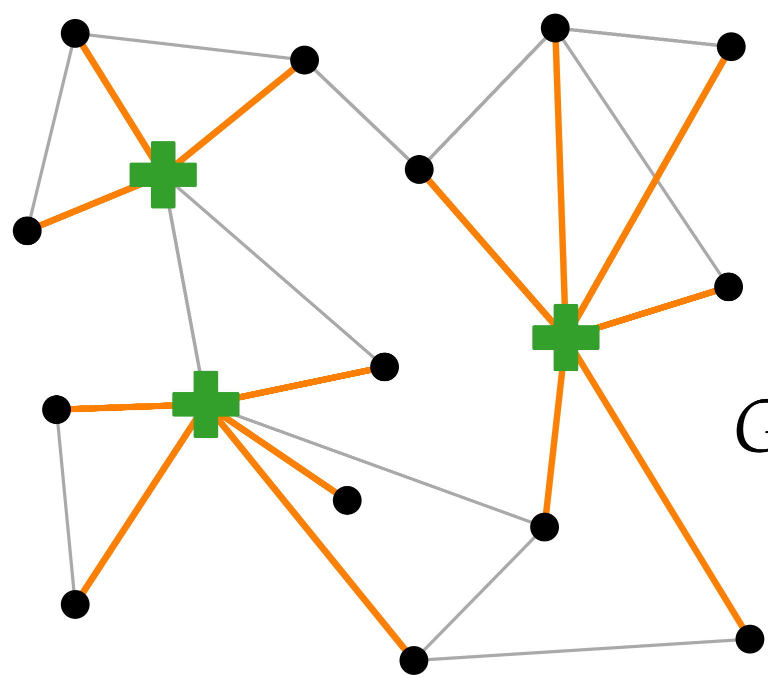


$$\mathrm{dom}(G_j) \leq k$$

$$G_j := (V, \{e_1, \ldots, e_j\})$$

. . . try each G_j.

Def. A vertex set D of a graph H is **dominating** if each vertex is either in D or adjacent to a vertex in D. The cardinality of a smallest dominating set in H is denoted by $\mathrm{dom}(H)$.



$$\mathrm{dom}(G_j) \leq k$$

$$G_j := (V, \{e_1, \ldots, e_j\})$$

. . . but computing $\mathrm{dom}(H)$ is NP-hard.

Approximation Algorithms

Lecture 6:
k-Center via Parametric Pruning

Part III:
Square of a Graph

Philipp Kindermann Summer Semester 2020

Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

Square of a Graph

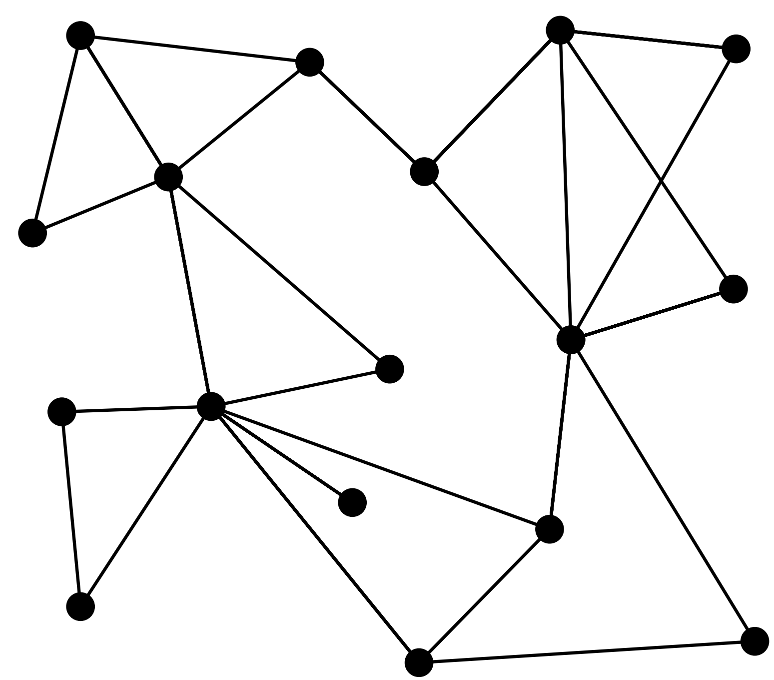
Idea: Find a small dominating set in a "coarsened" G_j

Def. The **square** H^2 of a graph H has the same vertex set as H.

Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

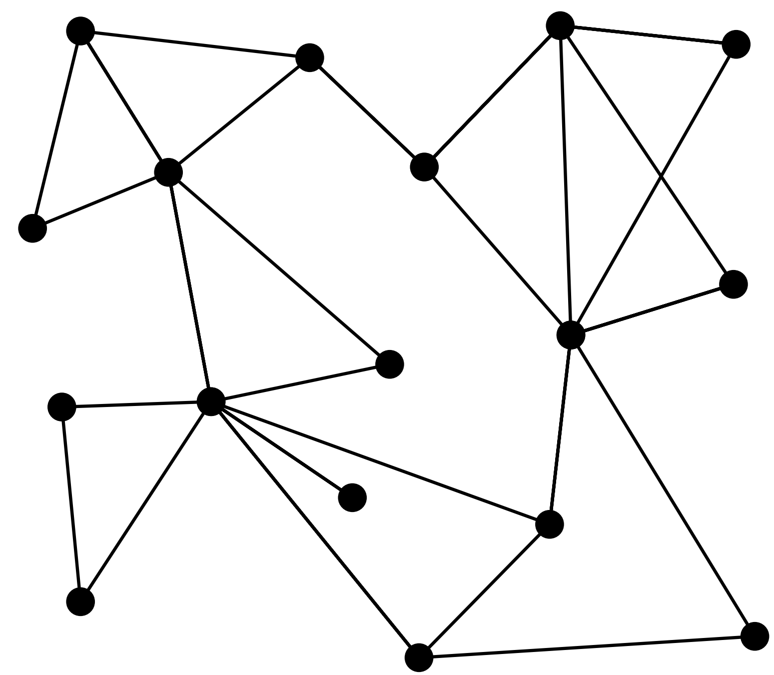
Def. The **square** H^2 of a graph H has the same vertex set as H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

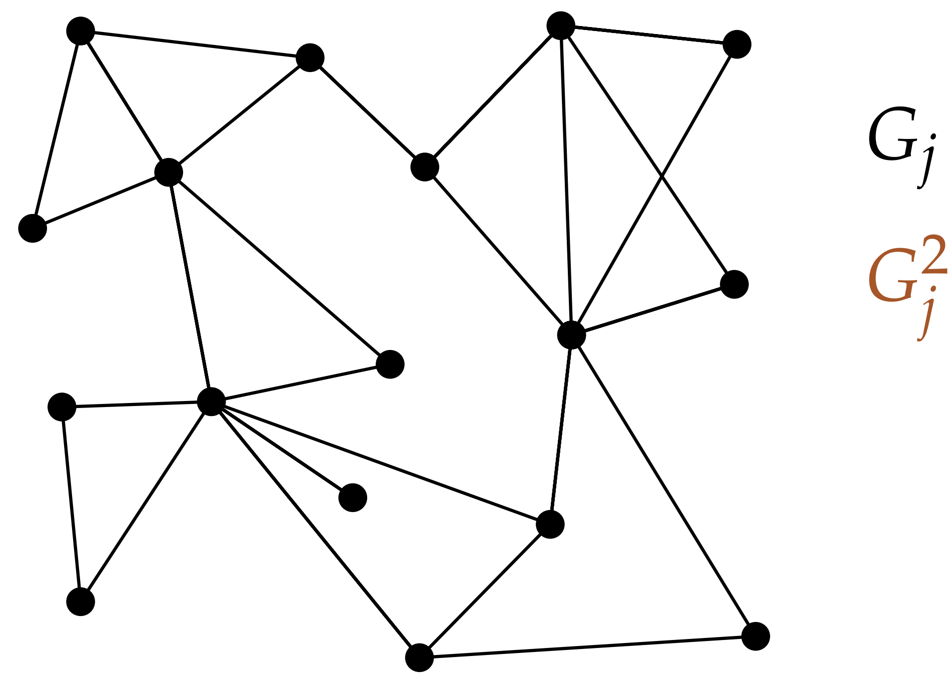
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

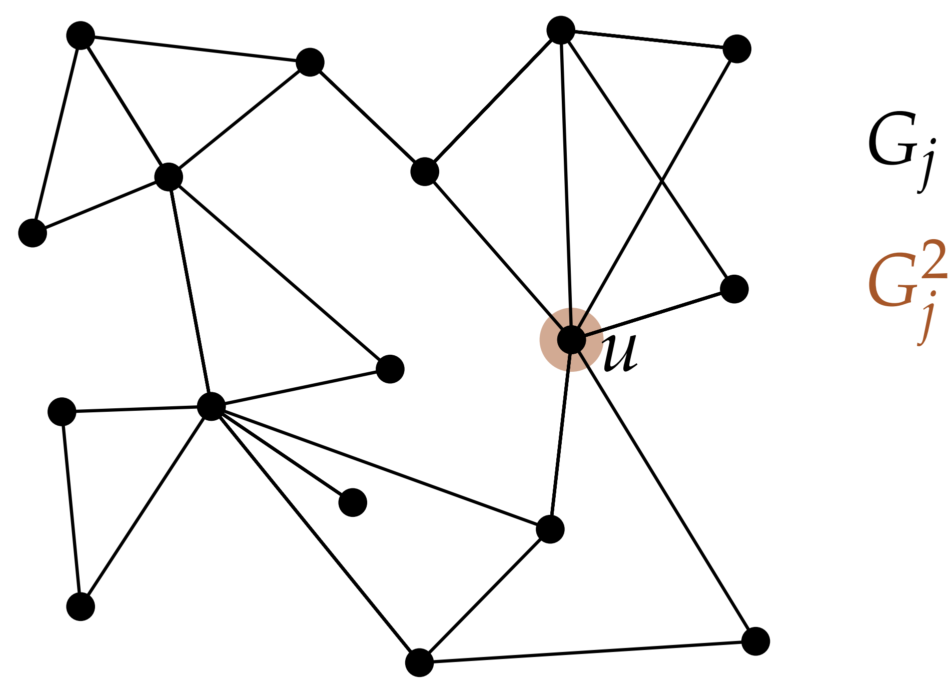
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

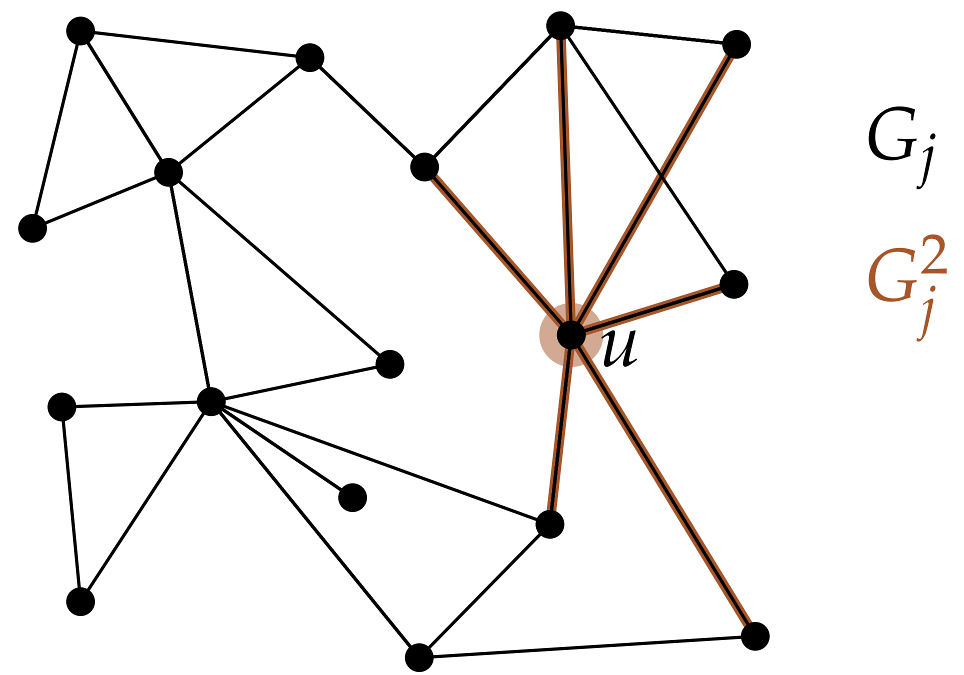
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

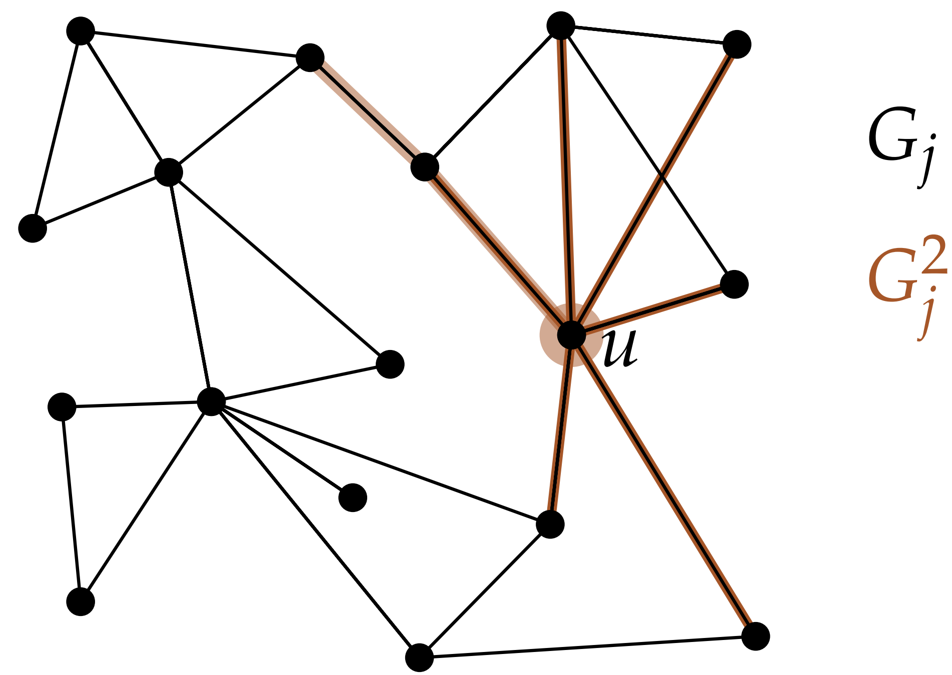
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

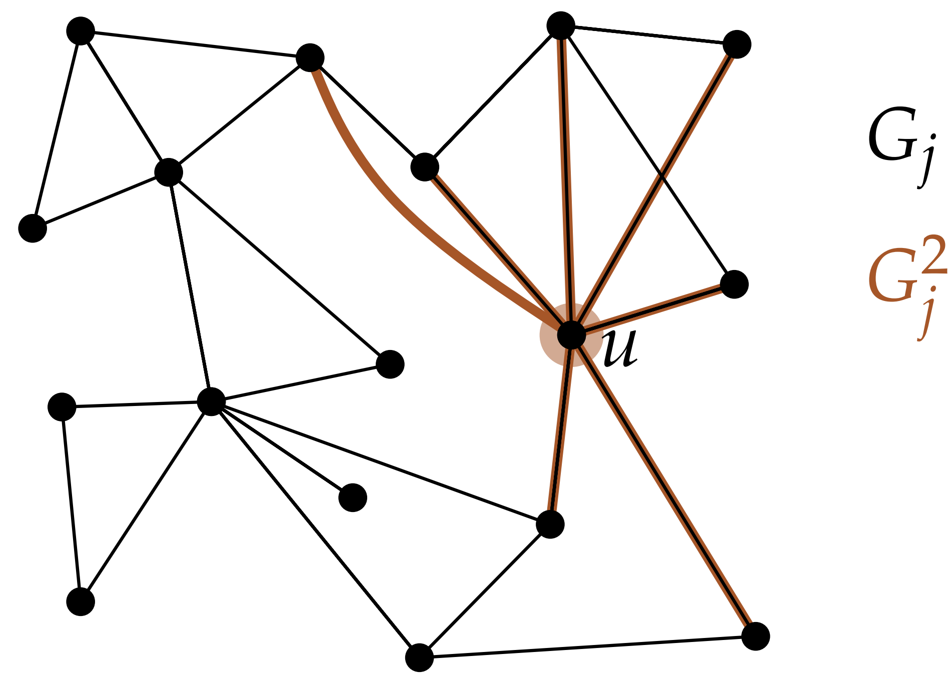
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

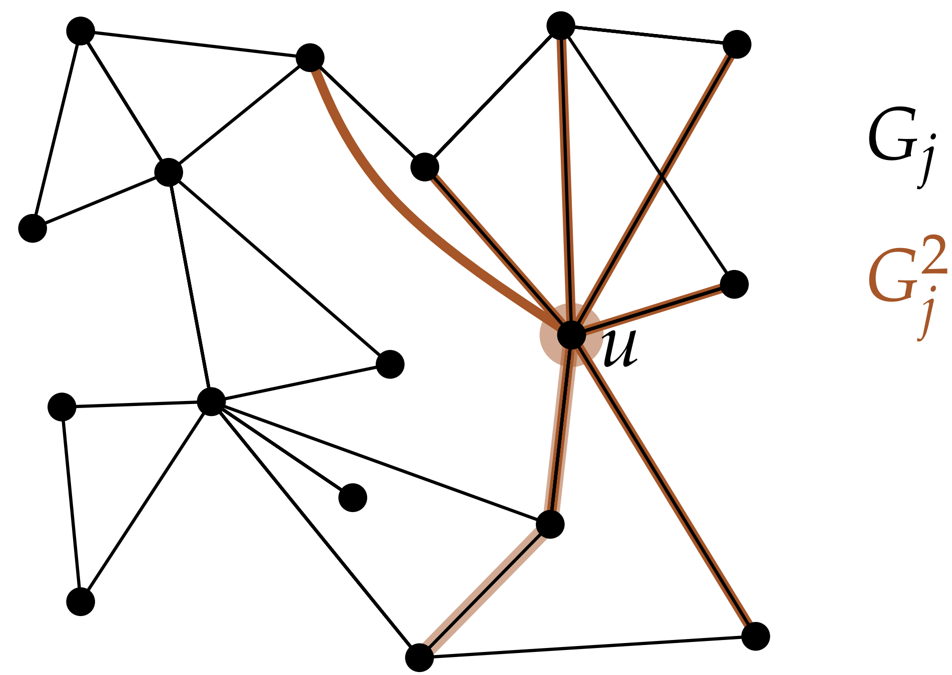
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

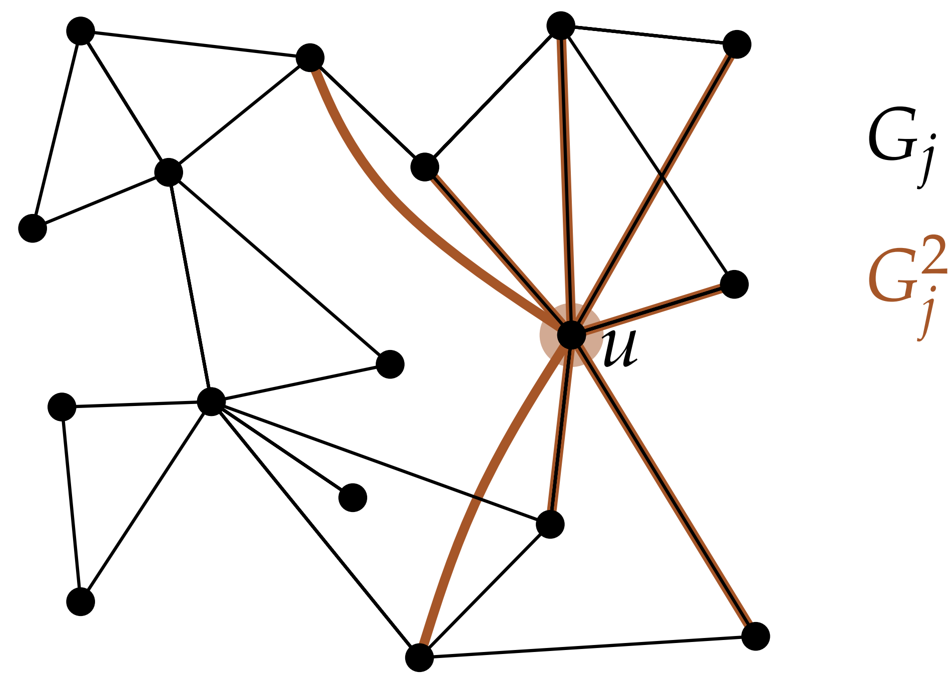
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

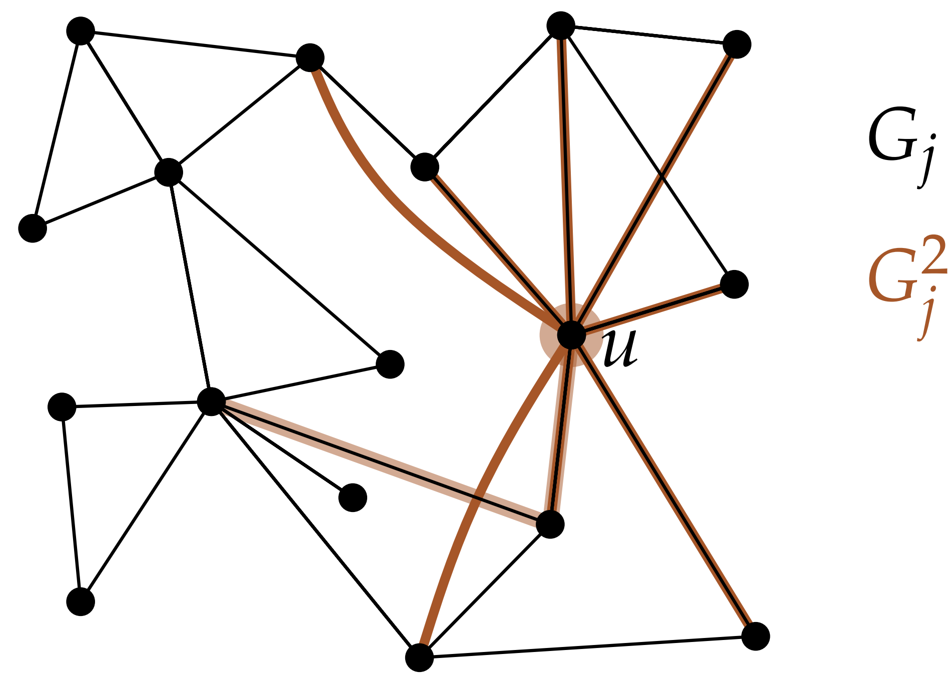
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

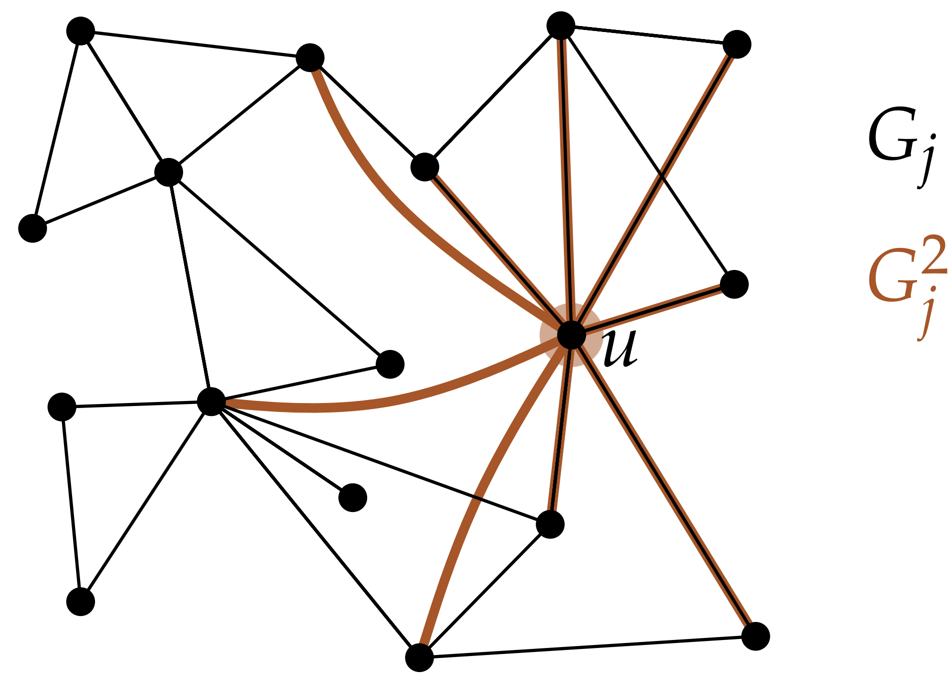
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

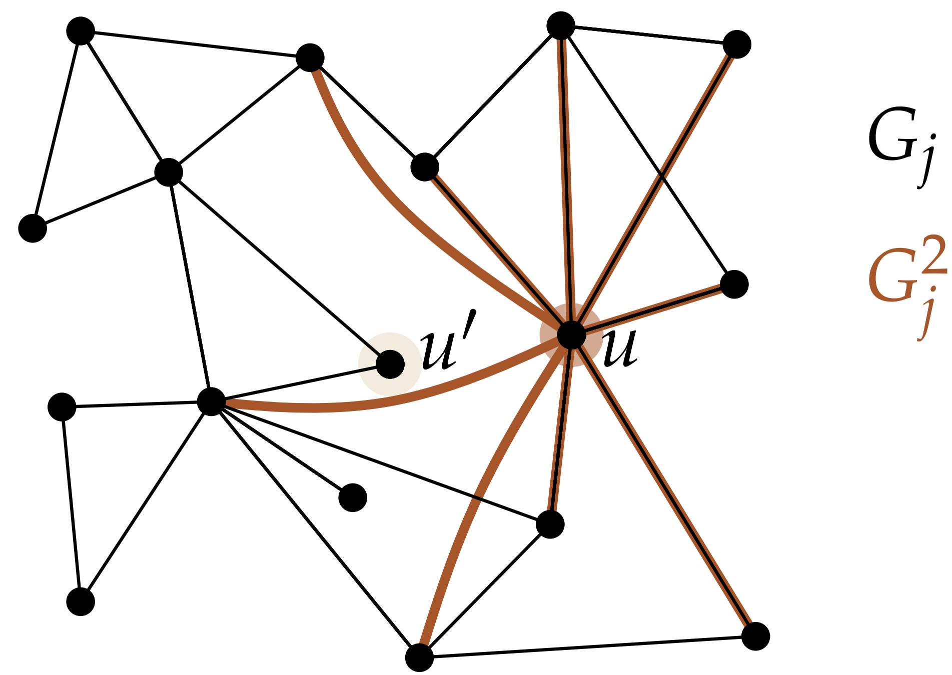
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

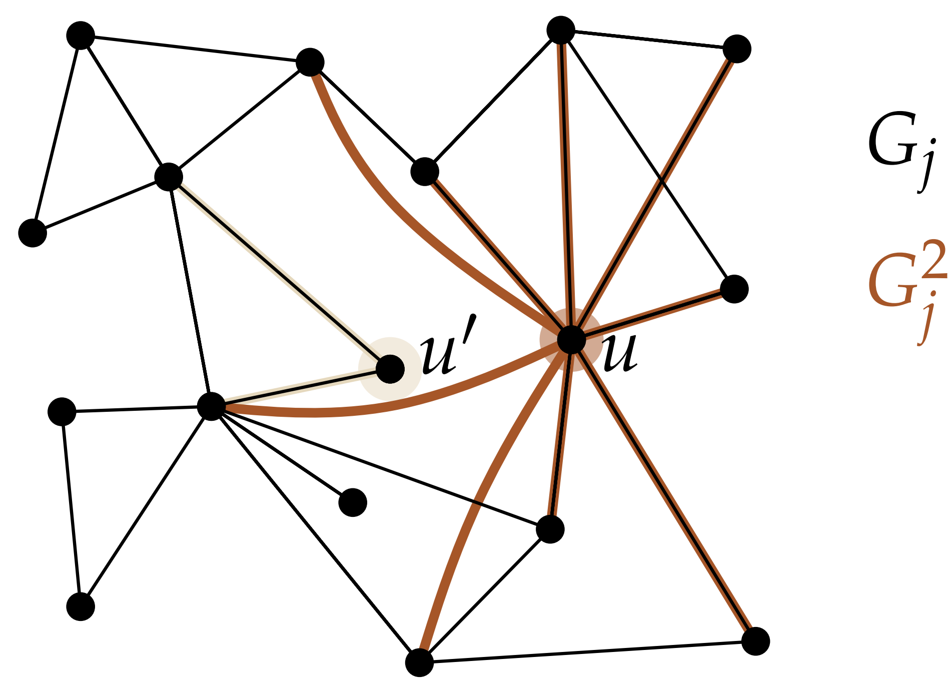
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

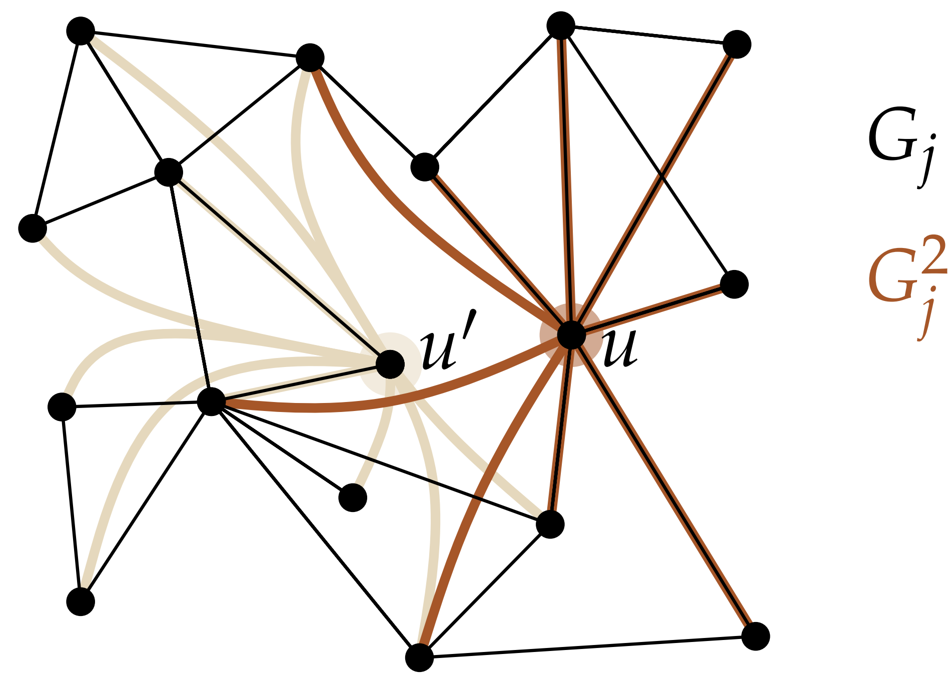
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.

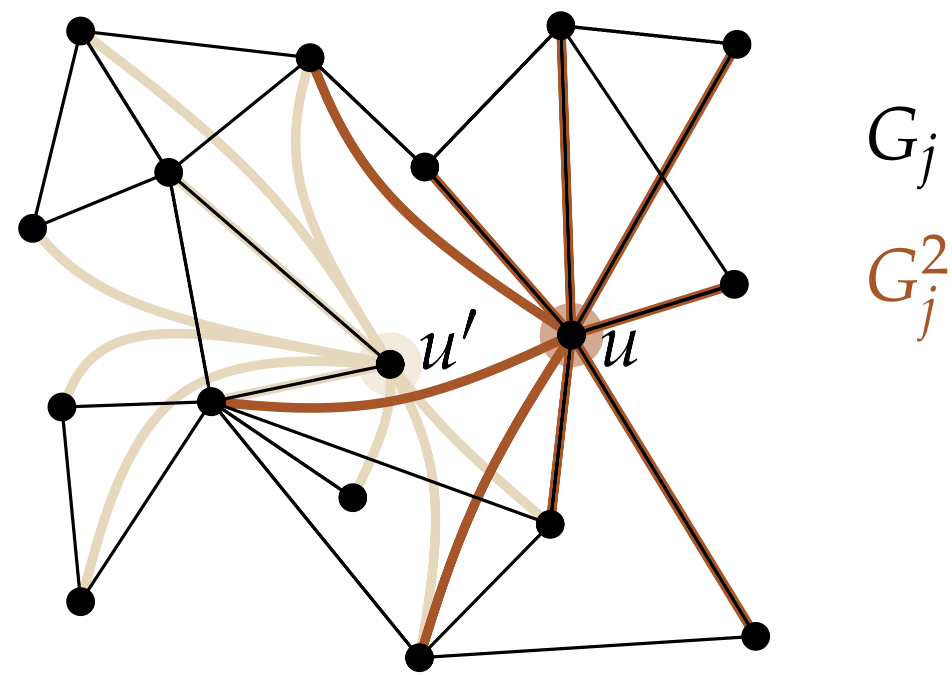


Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.

Obs. A dominating set in G_j^2 with $\leq k$ elements is already a 2-approximation.



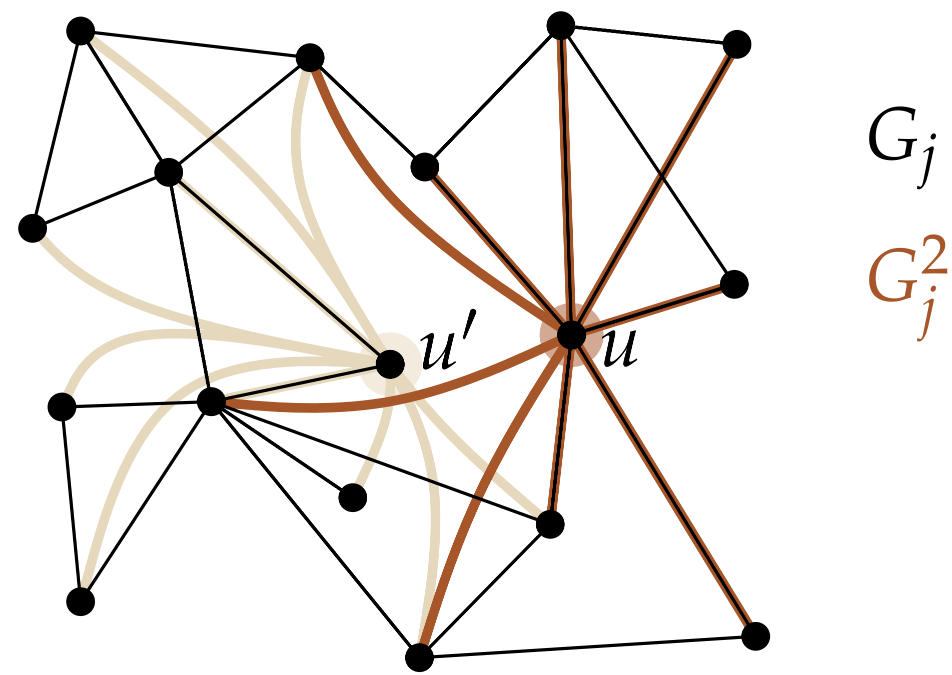
Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.

Obs. A dominating set in G_j^2 with $\leq k$ elements is already a 2-approximation.

Why?



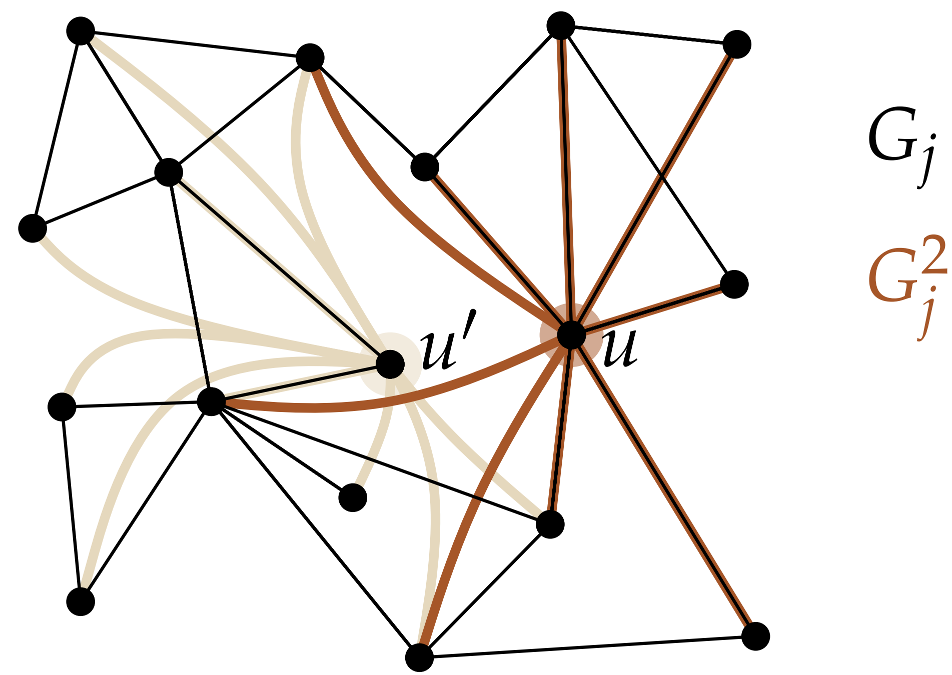
Square of a Graph

Idea: Find a small dominating set in a "coarsened" G_j

Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.

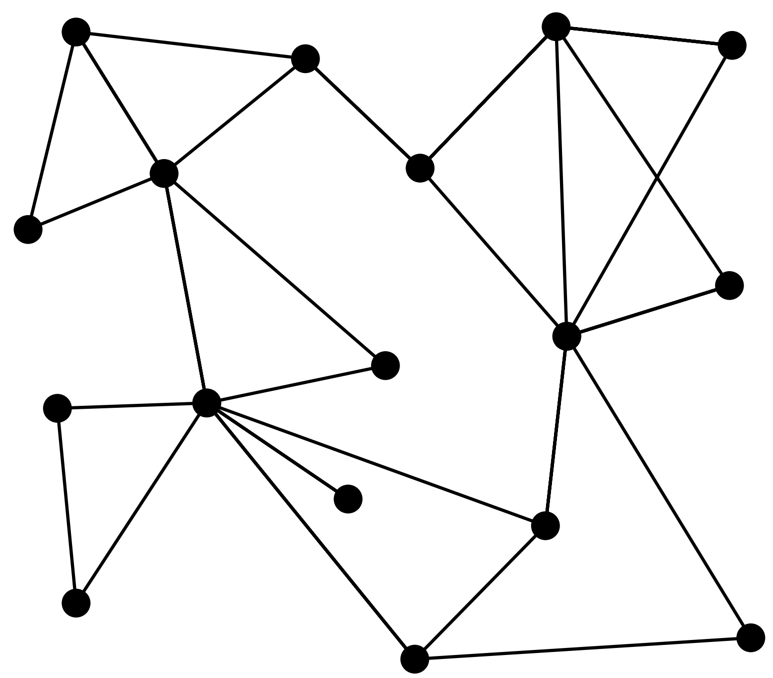
Obs. A dominating set in G_j^2 with $\leq k$ elements is already a 2-approximation.

Why? $\max_{e \in E(G_j)} = e_j$!



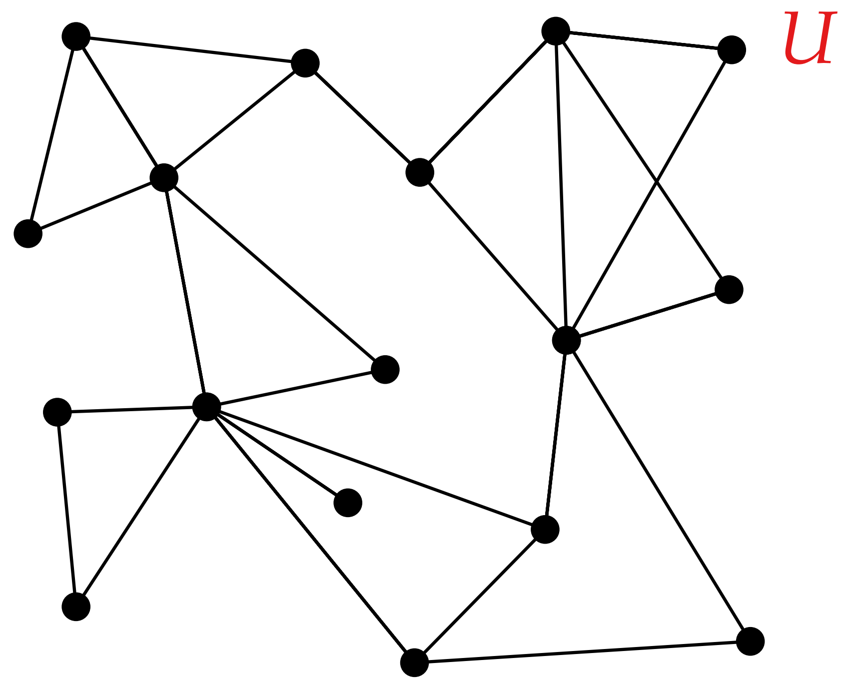
Independent Sets

Def. A vertex set *U* in a graph is called **independent** (or **stable**), if no pair of vertices in *U* form an edge.



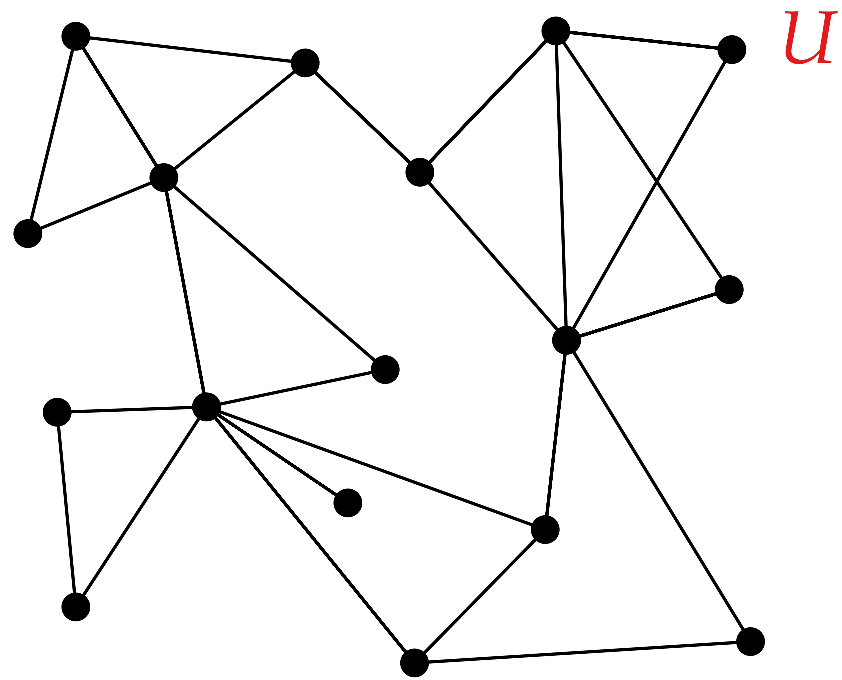
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge.



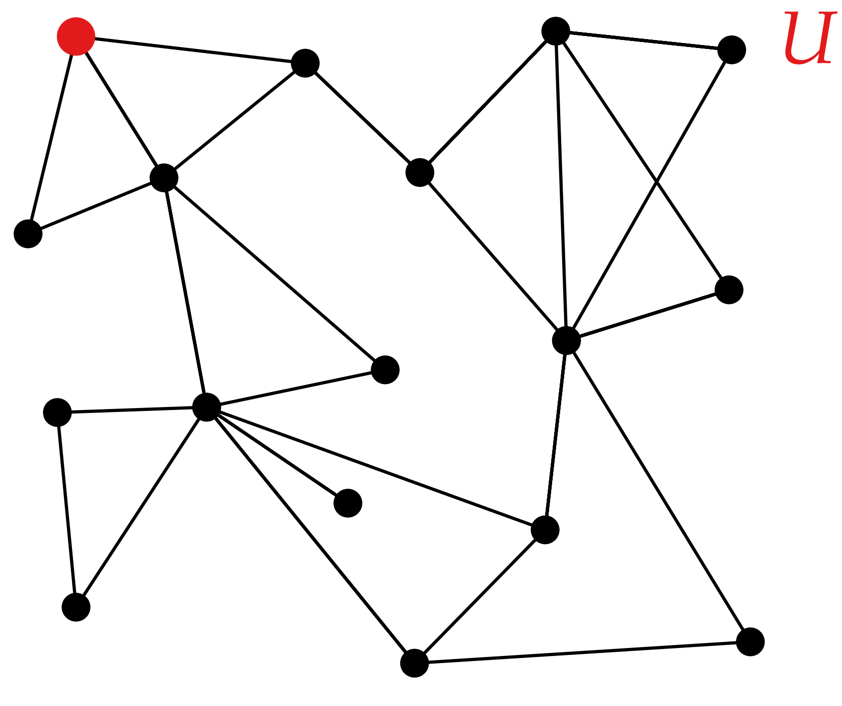
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



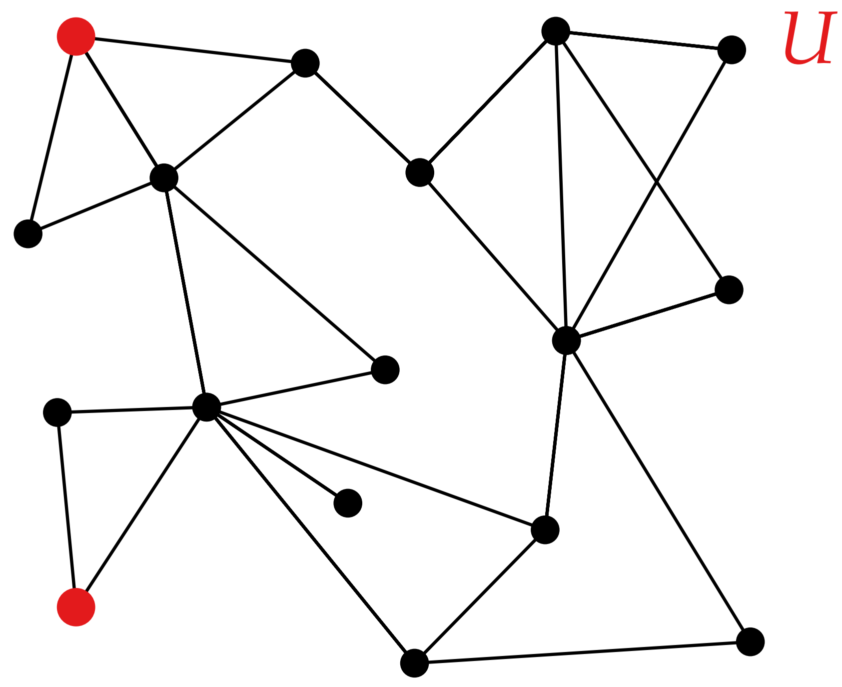
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



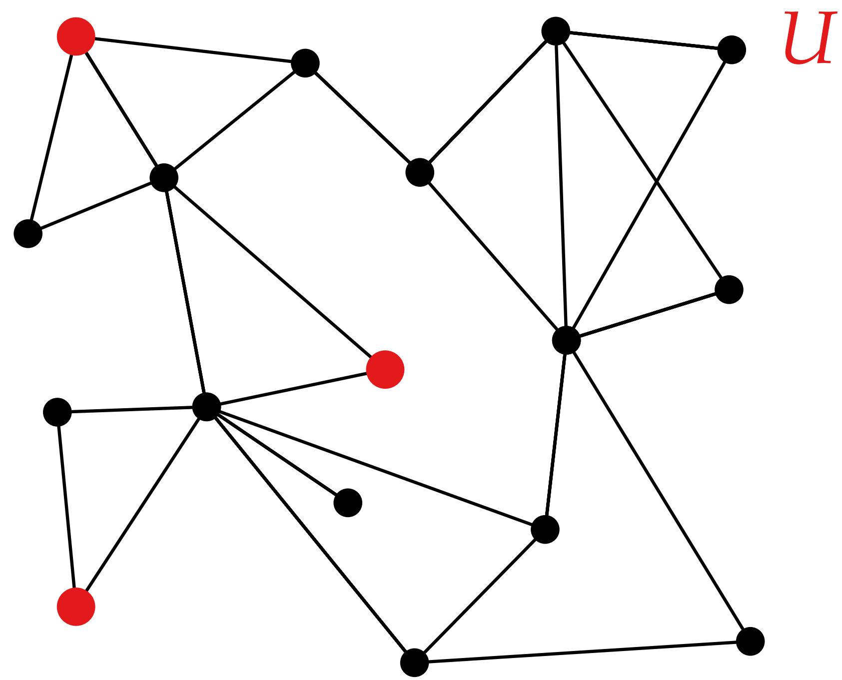
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



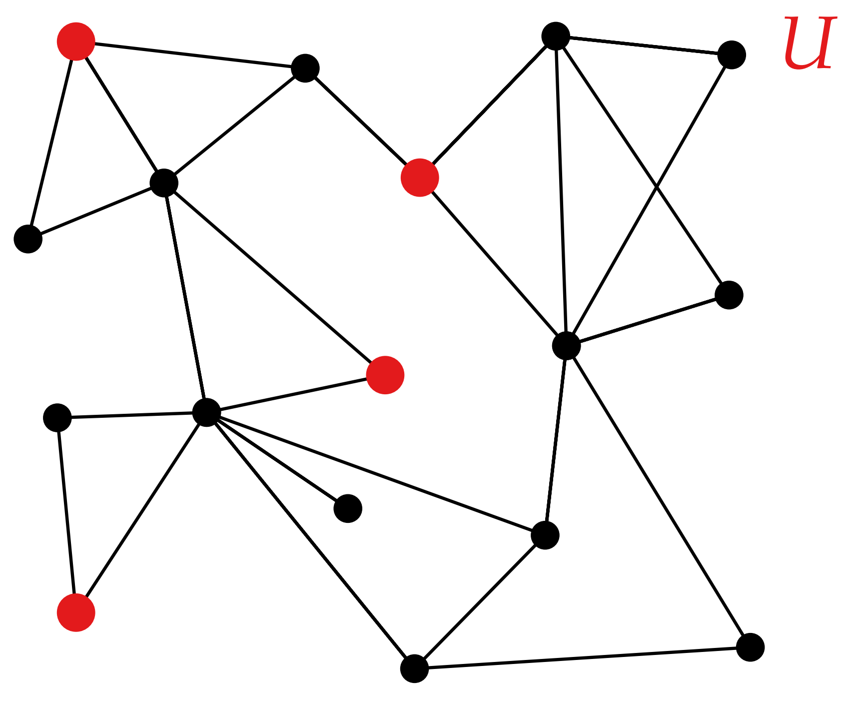
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



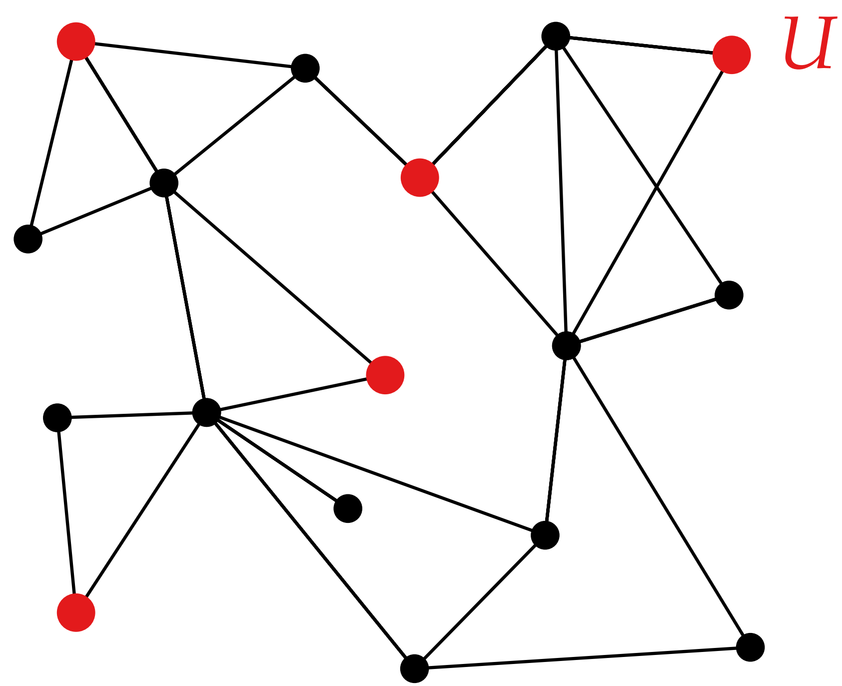
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



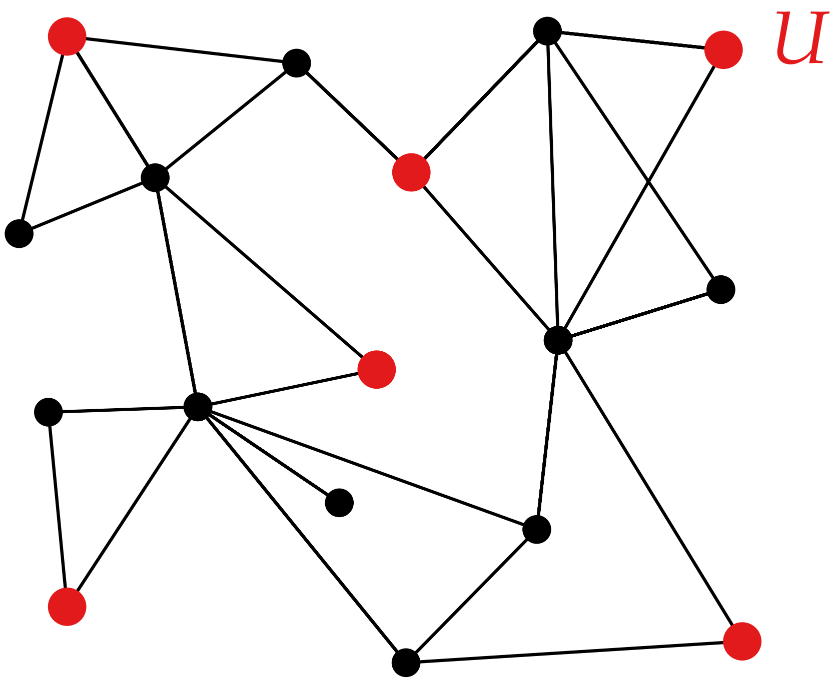
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



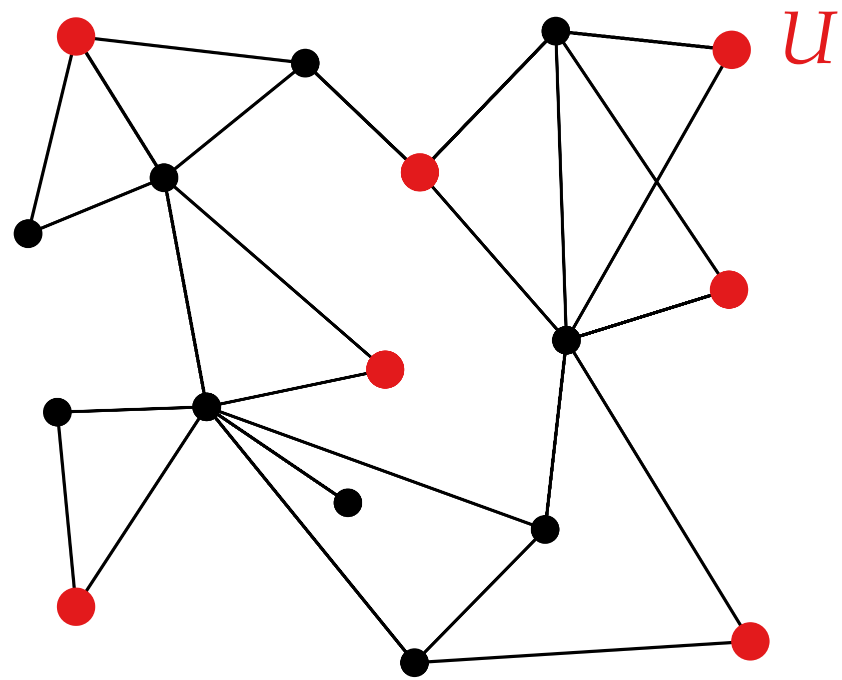
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



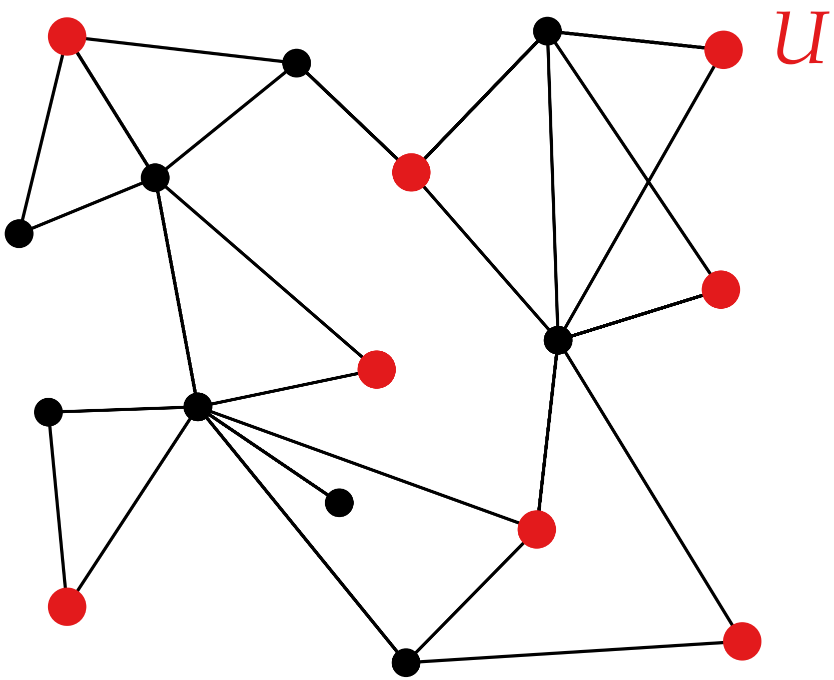
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



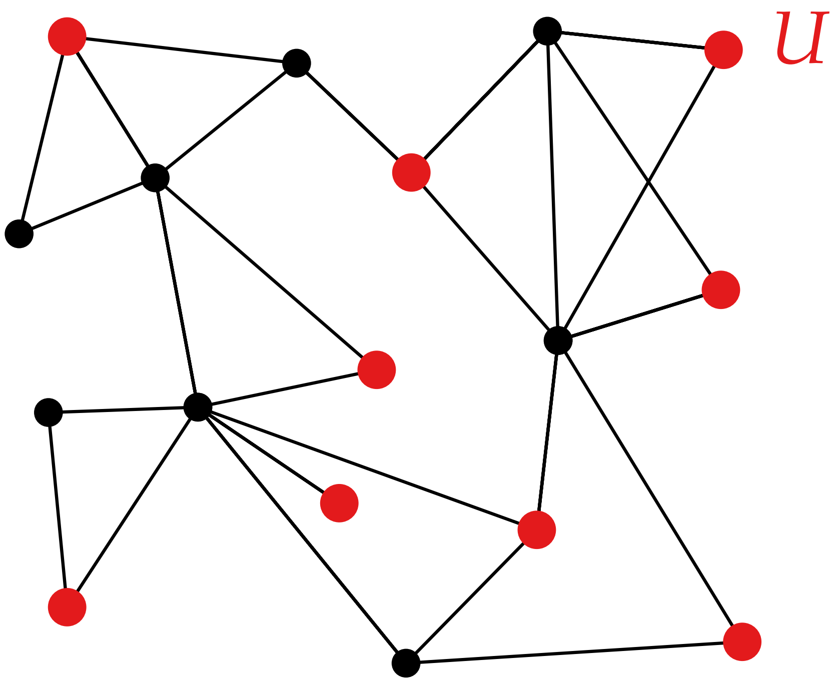
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



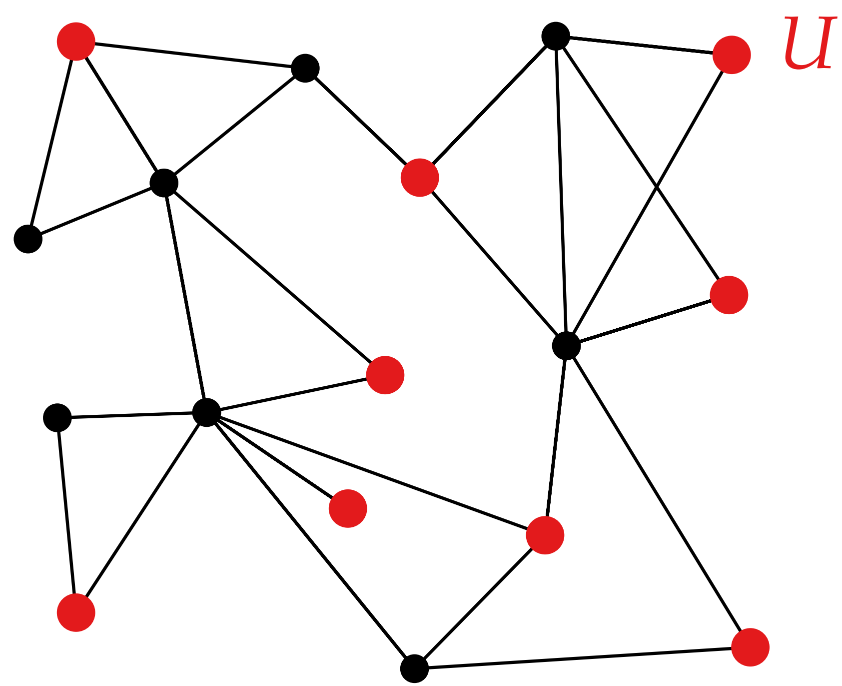
Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



Independent Sets

Def. A vertex set U in a graph is called **independent** (or **stable**), if no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



Obs. Maximal independent sets are dominating sets :-)

Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

Independent Sets in H^2

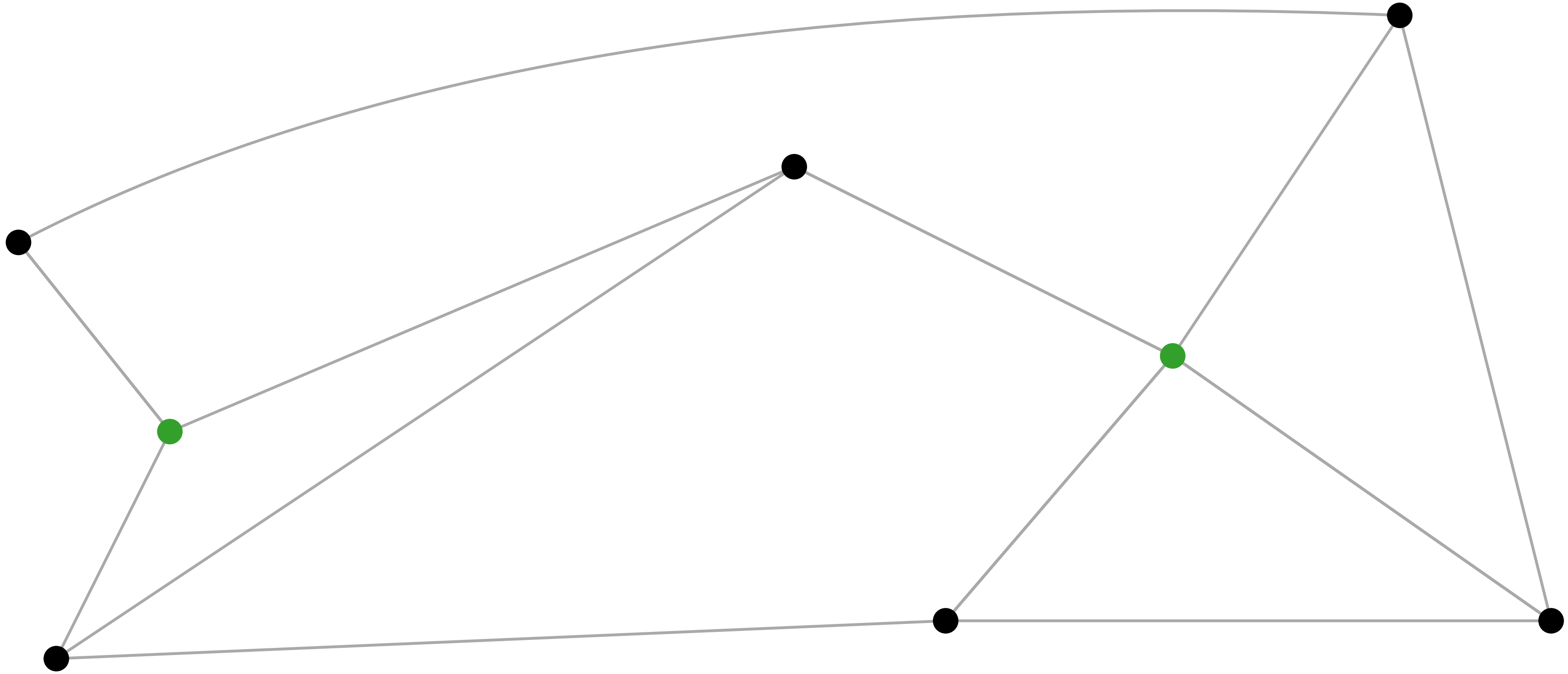
Lemma. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$.

What does a dominating set of H look like in H^2?

Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$.

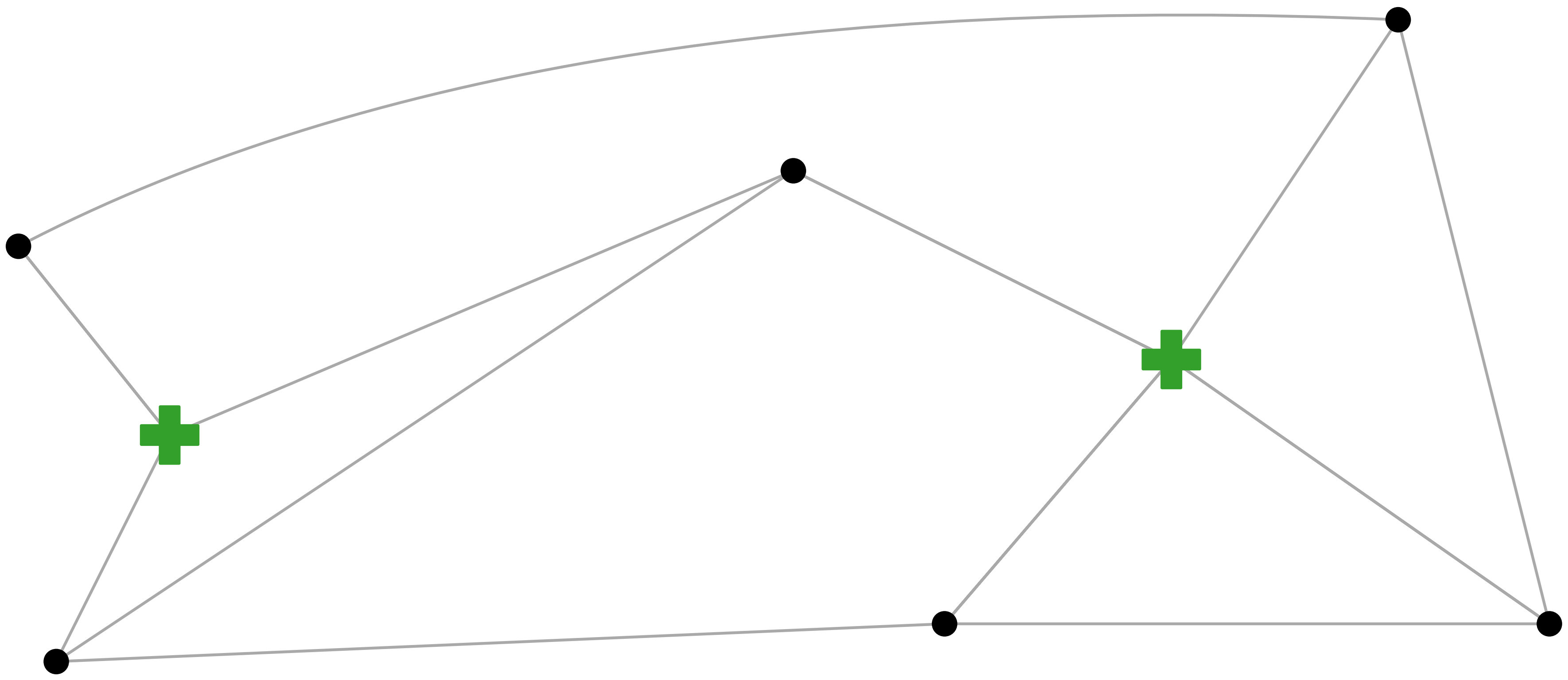
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$.

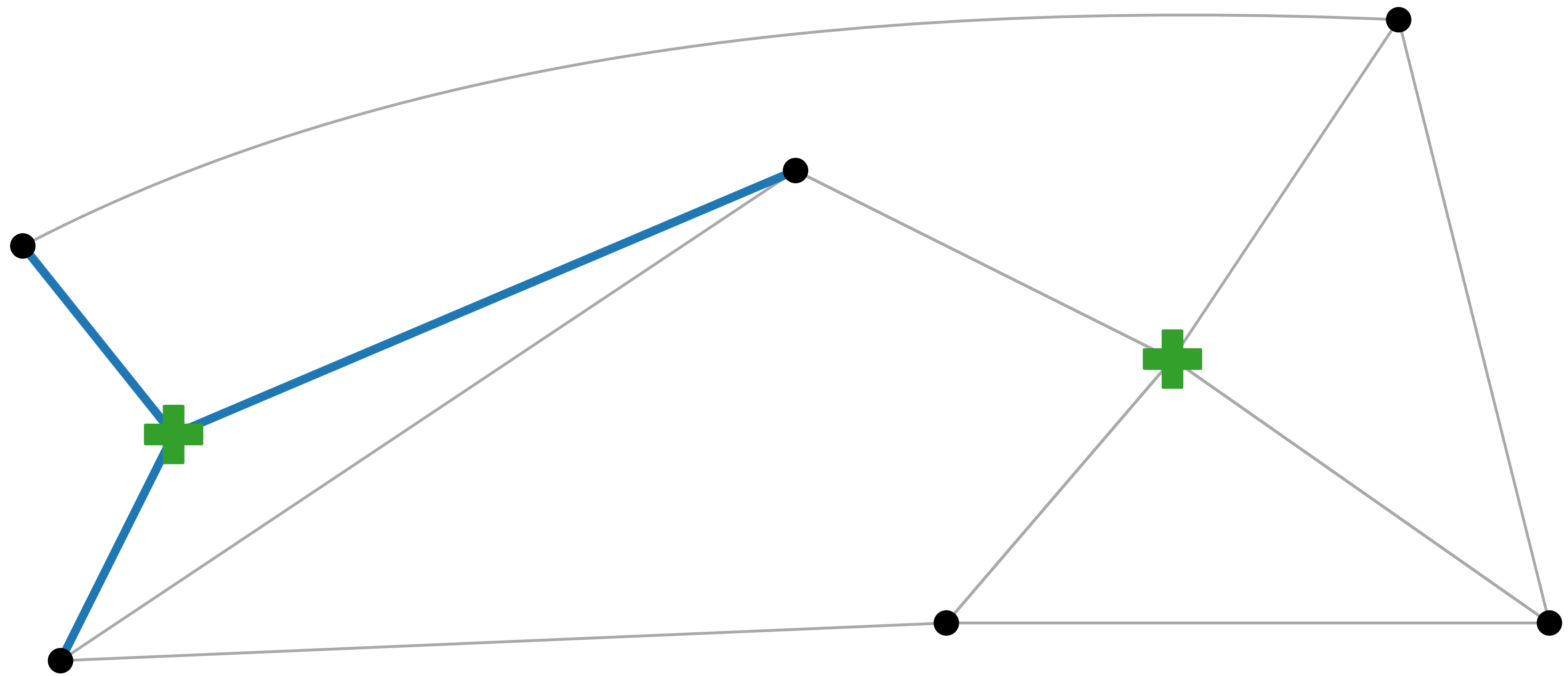
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

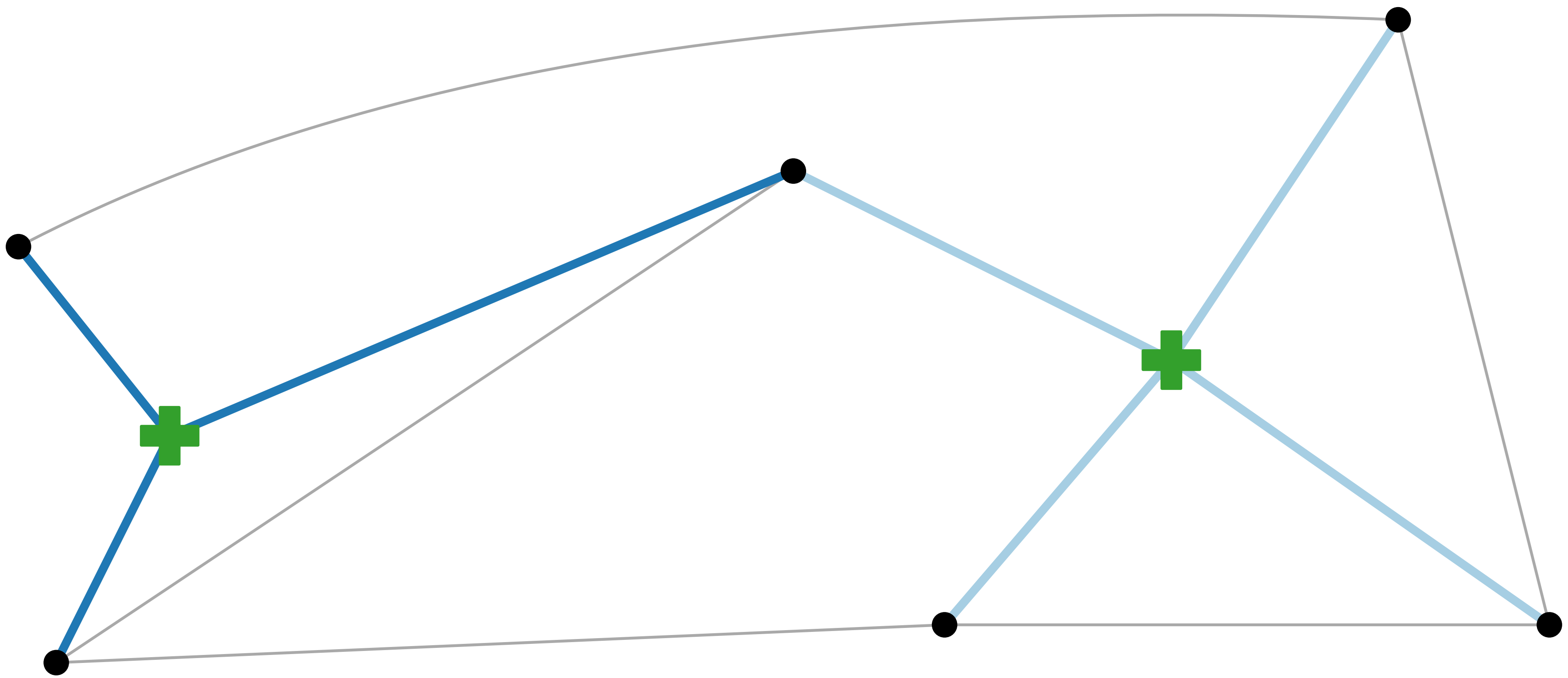
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$.

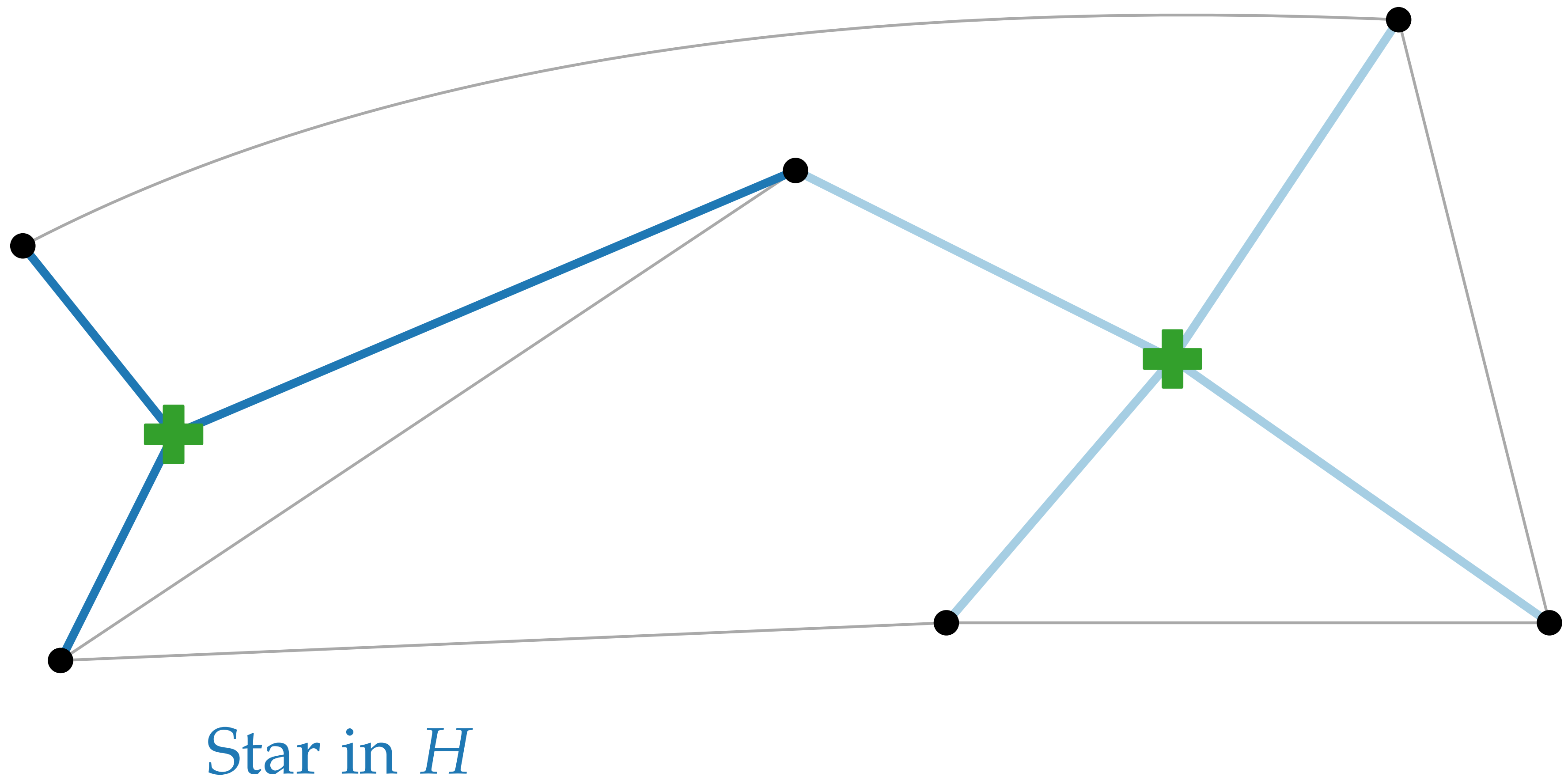
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

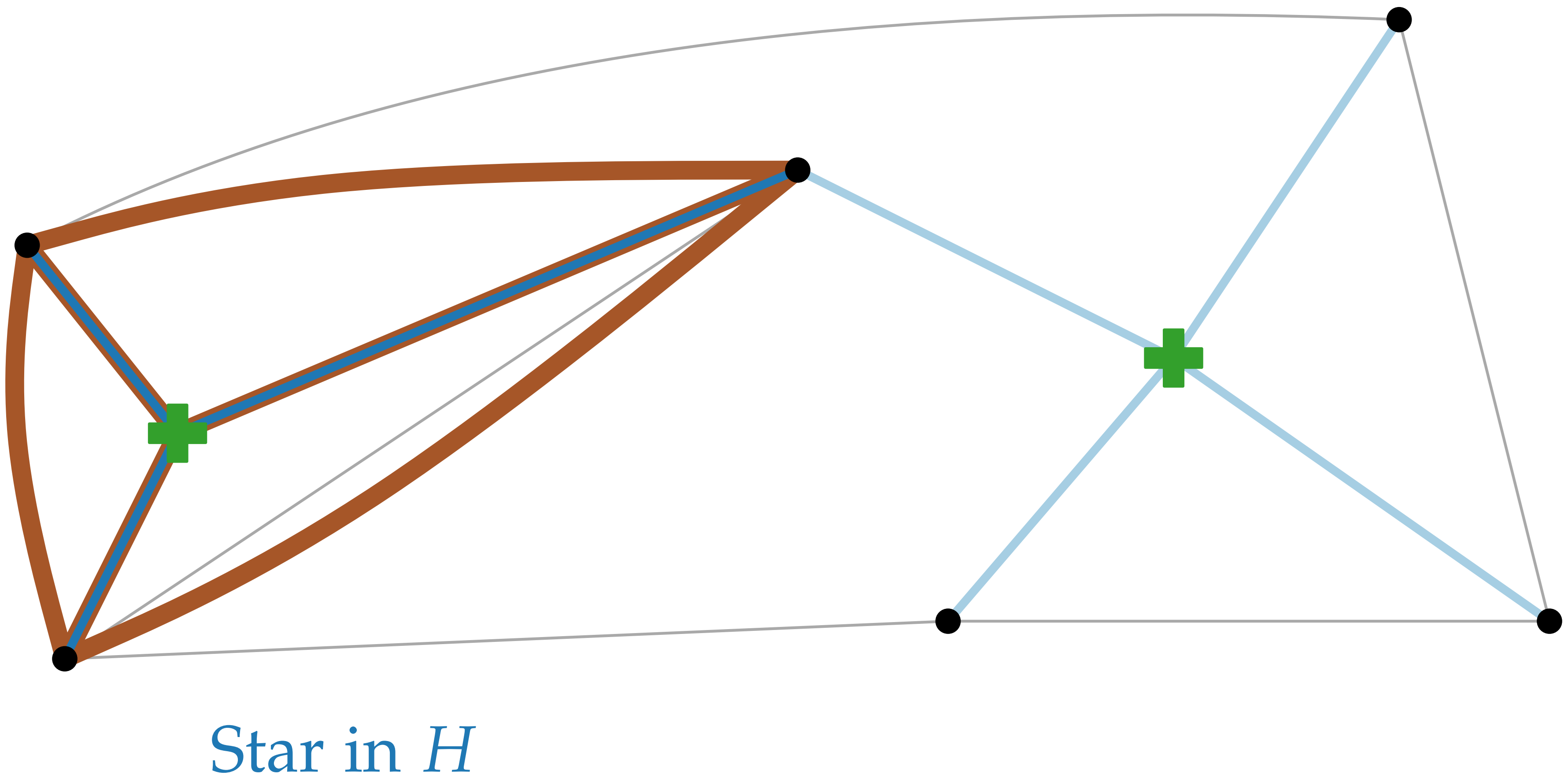
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$.

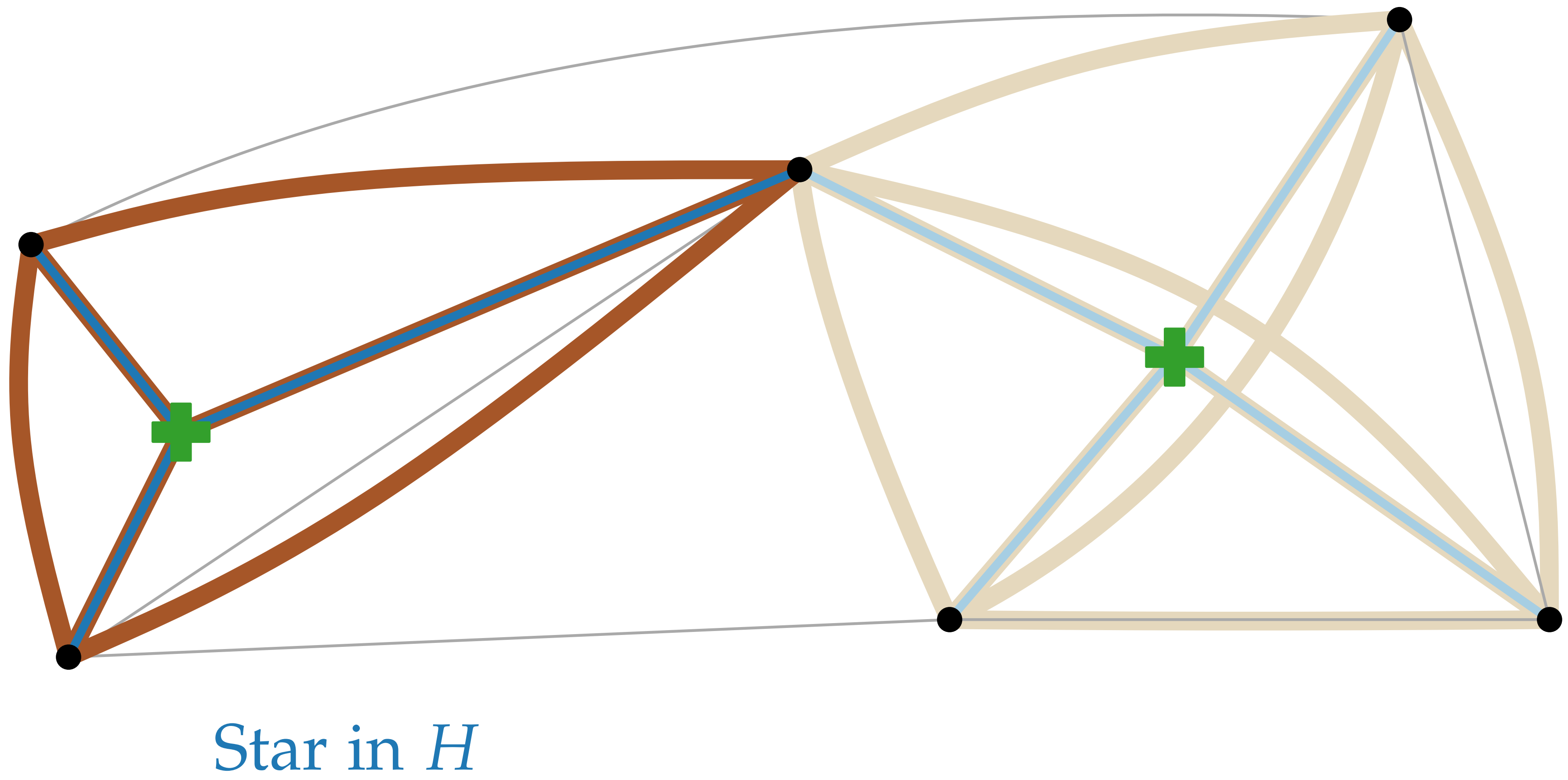
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

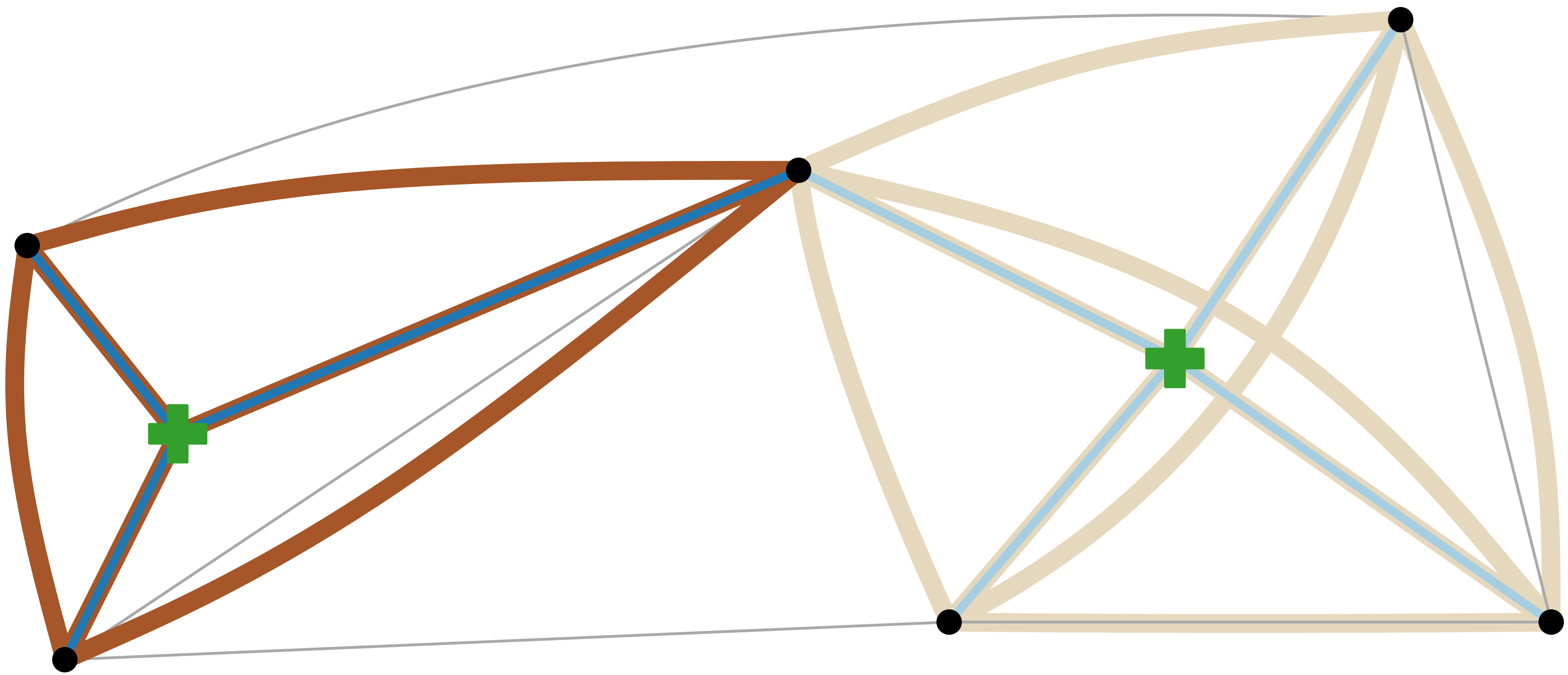
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

What does a dominating set of H look like in H^2?



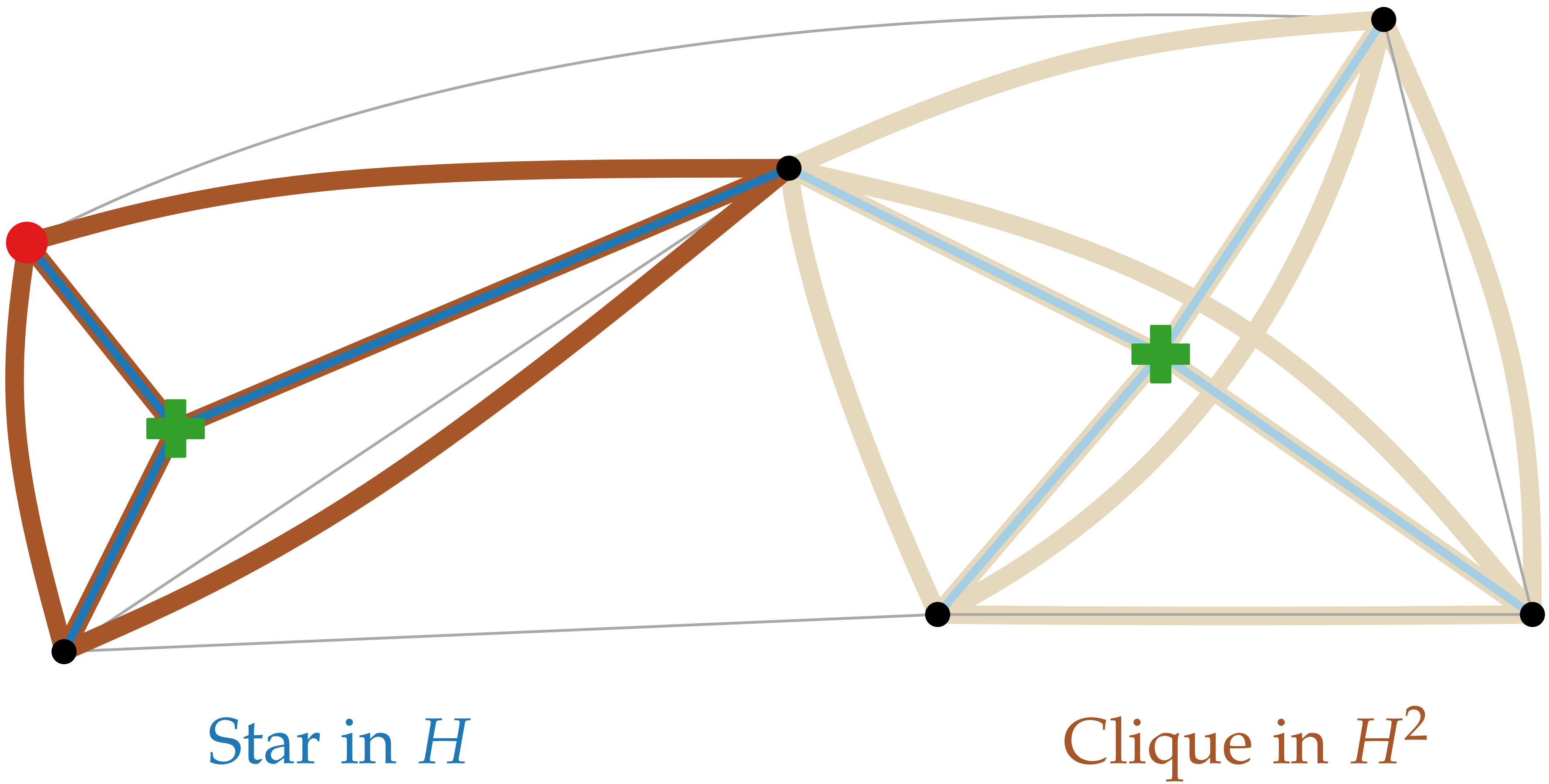
Star in H

Clique in H^2

Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

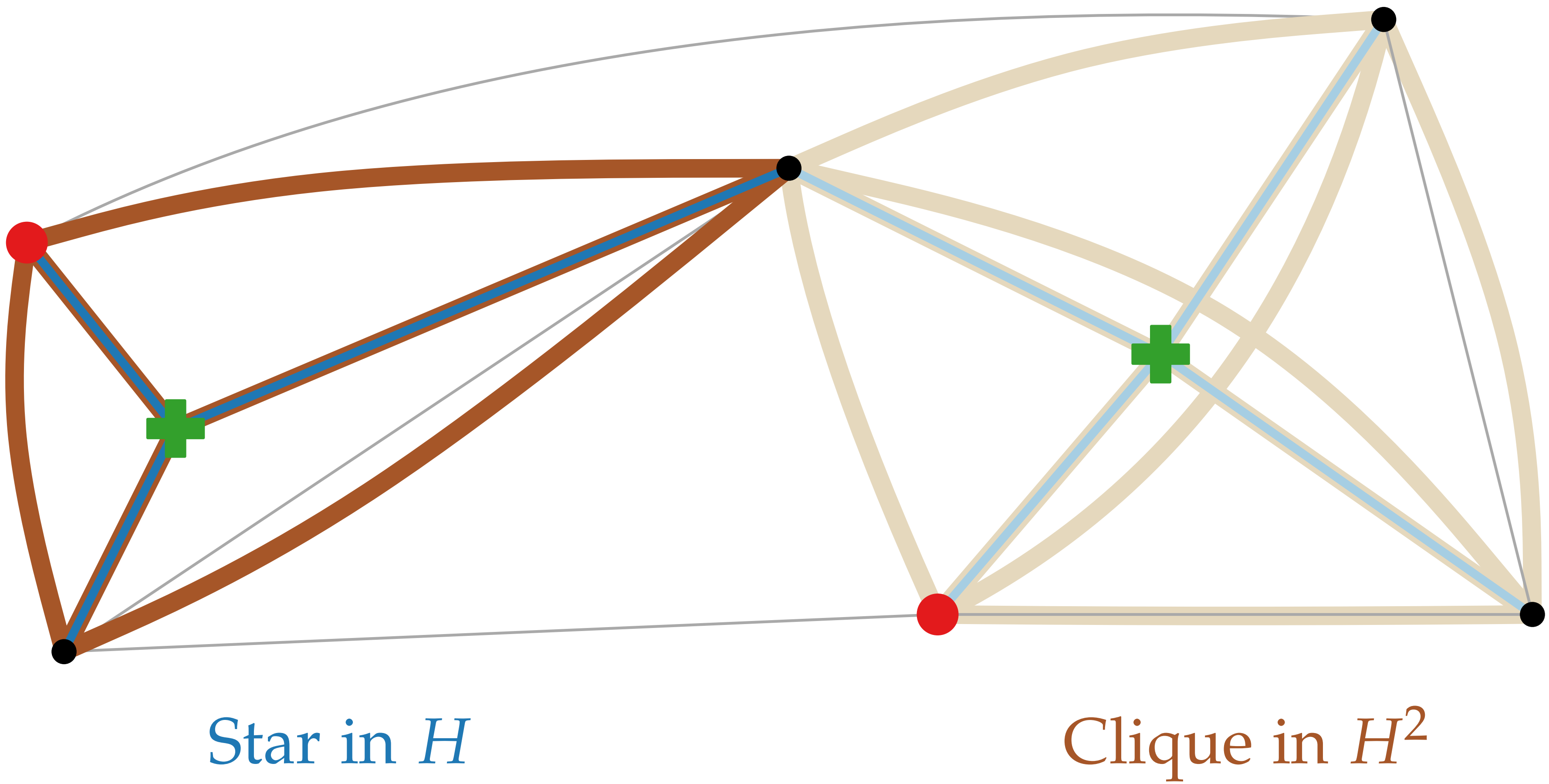
What does a dominating set of H look like in H^2?



Independent Sets in H^2

Lemma. For a graph H and an independent set U in H^2, $|U| \leq \mathrm{dom}(H)$.

What does a dominating set of H look like in H^2?



Approximation Algorithms

Lecture 6:
k-Center via Parametric Pruning

Part IV:
Factor-2-Approximation for Metric-k-Center

Philipp Kindermann

Summer Semester 2020

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Construct G_j^2

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Construct G_j^2

Find a maximal independent set U_j in G_j^2

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Construct G_j^2

Find a maximal independent set U_j in G_j^2

if $|U_j| \leq k$ **then**

return U_j

Factor-2-Approx for Metric-k-Center

Metric-k-Center($G = (V, E; c), k$)

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Construct G_j^2

Find a maximal independent set U_j in G_j^2

if $|U_j| \leq k$ **then**

return U_j

Lemma. For j provided by the algorithm, we have $c(e_j) \leq \text{OPT}$.

Factor-2-Approx for Metric-k-Center

Metric-k-Center$(G = (V, E; c), k)$

Sort the edges of G by cost: $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

  if $|U_j| \leq k$ **then**

    return U_j

Lemma. For j provided by the algorithm, we have $c(e_j) \leq \mathrm{OPT}$.

Theorem. The above algorithm is a factor-2-approximation algorithm for Metric-k-Center problem.

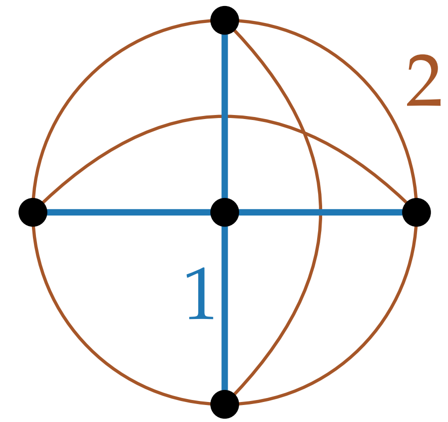
Can we do better . . . ?

Can we do better . . . ?

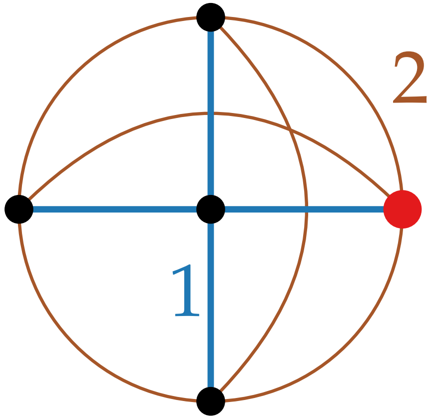
What about a tight example?

Can we do better . . . ?

What about a tight example?

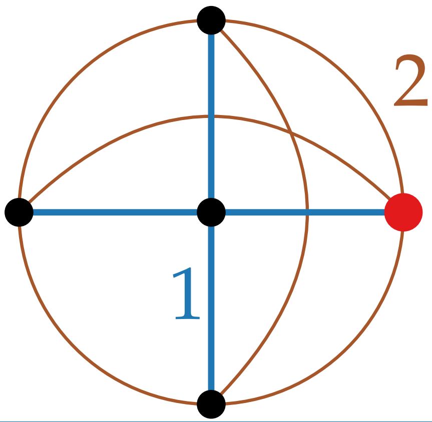


Can we do better . . . ?



What about a tight example?

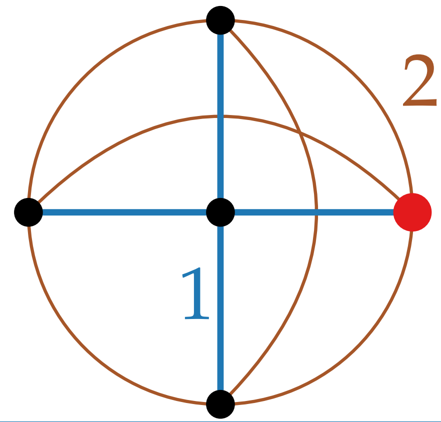
Can we do better . . . ?



What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2 - \varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Can we do better . . . ?

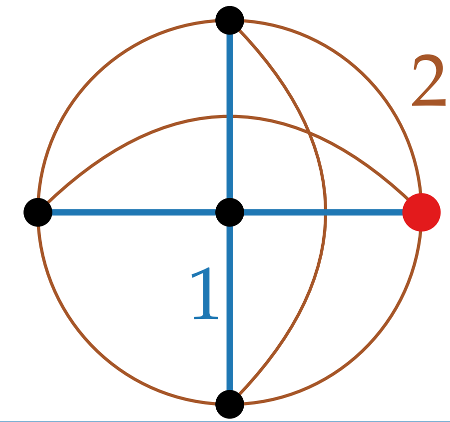


What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2 - \varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-Center.

Can we do better . . . ?

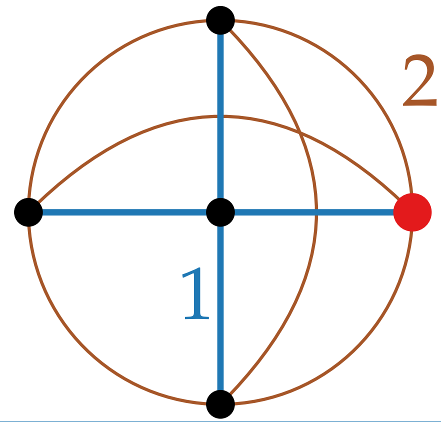


What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E)$, k

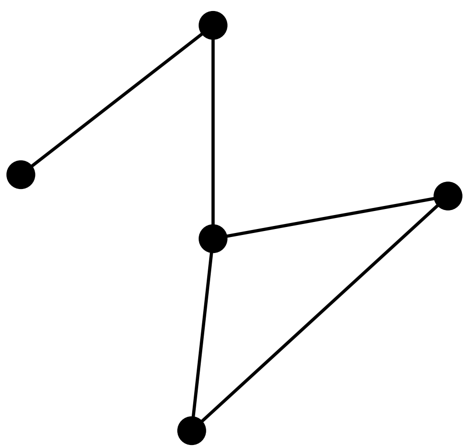
Can we do better . . . ?



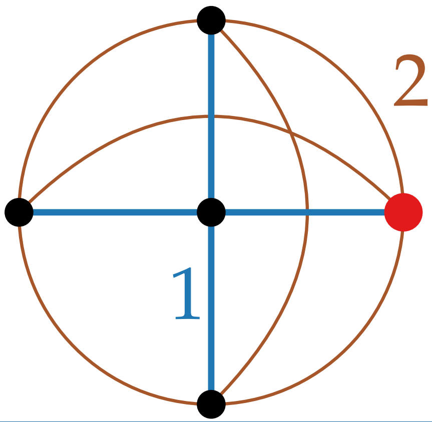
What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E)$, k



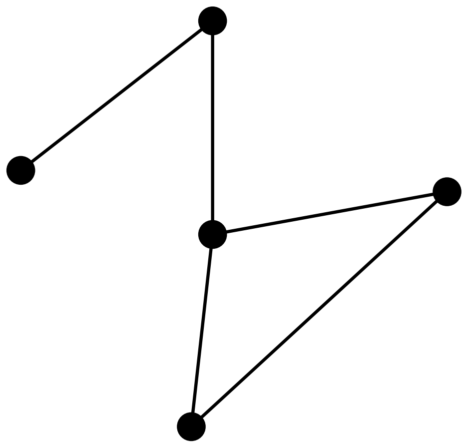
Can we do better . . . ?



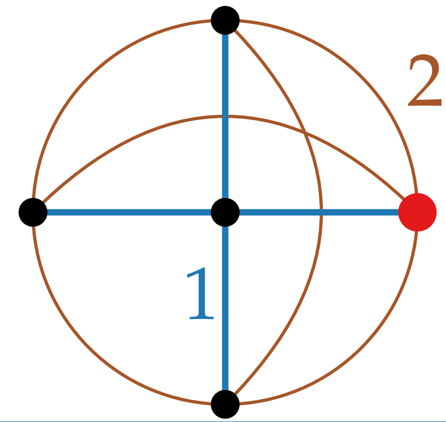
What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-Center.
Given.: $G = (V, E)$, k
Constr. complete graph $G' = (V, E \cup E')$



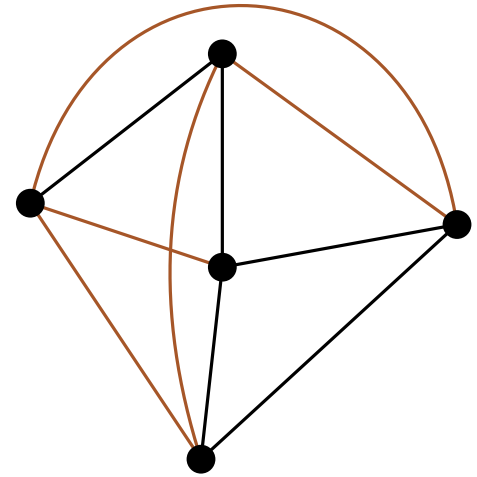
Can we do better . . . ?



What about a tight example?

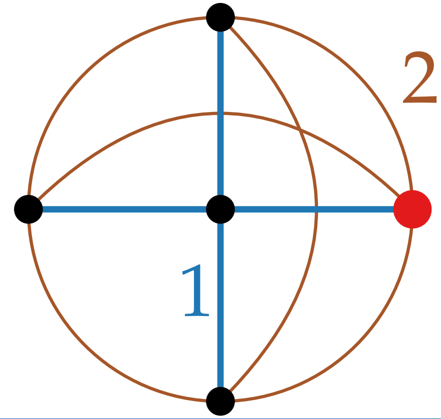
Theorem. Assuming P $\neq$ NP, there is no factor-$(2 - \varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-Center.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$



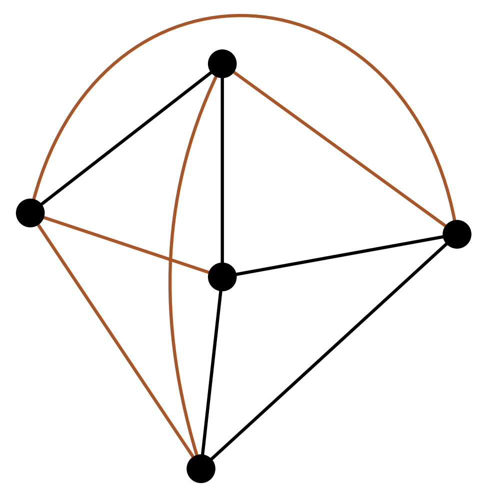
Can we do better . . . ?

What about a tight example?



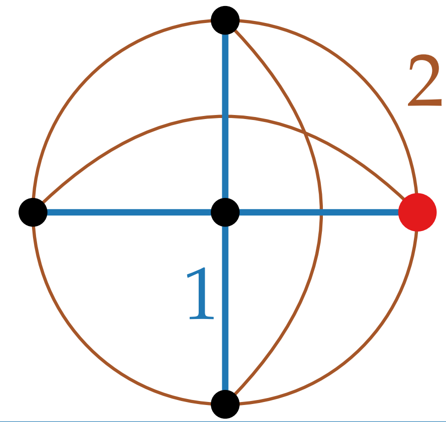
Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-Center.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



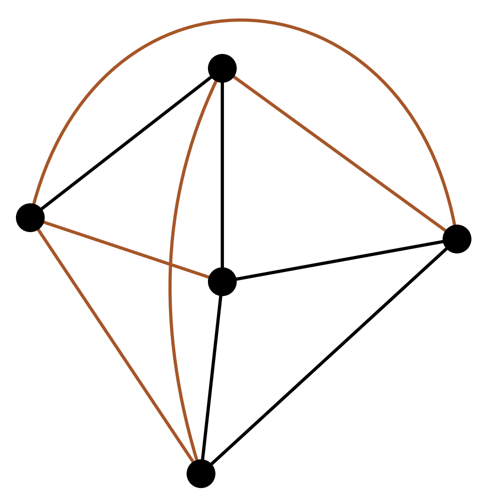
Can we do better . . . ?

What about a tight example?



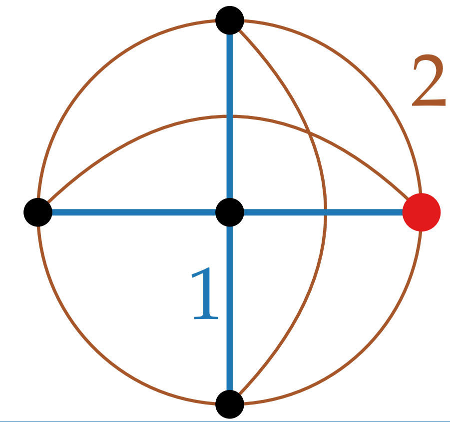
Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center

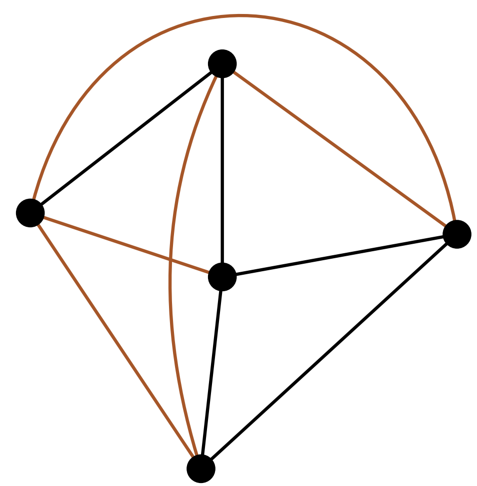
Can we do better . . . ?



What about a tight example?

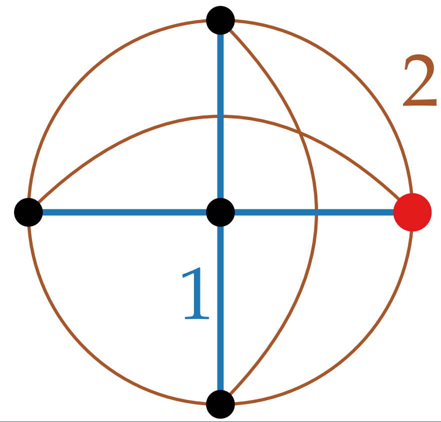
Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center
If $\text{dom}(G) \leq k$, then $\text{cost}(S) = 1$

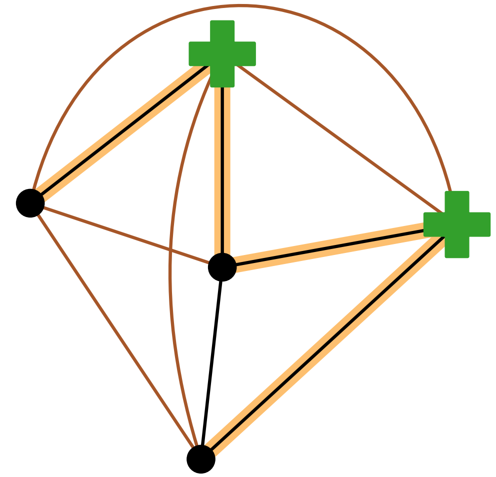
Can we do better . . . ?



What about a tight example?

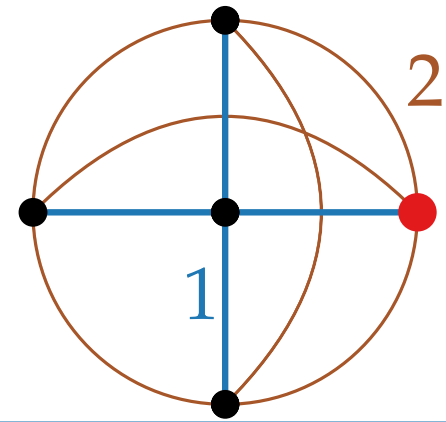
Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center
If $\text{dom}(G) \leq k$, then $\text{cost}(S) = 1$

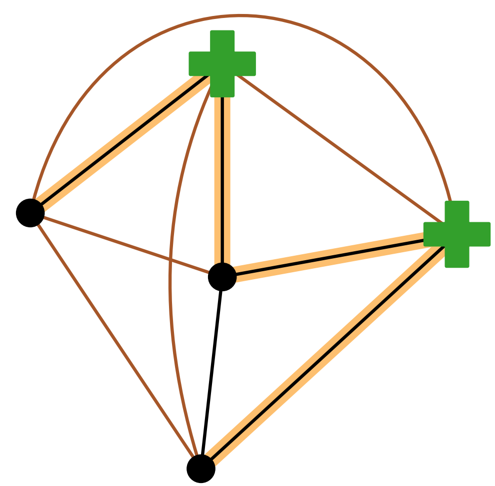
Can we do better . . . ?



What about a tight example?

Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

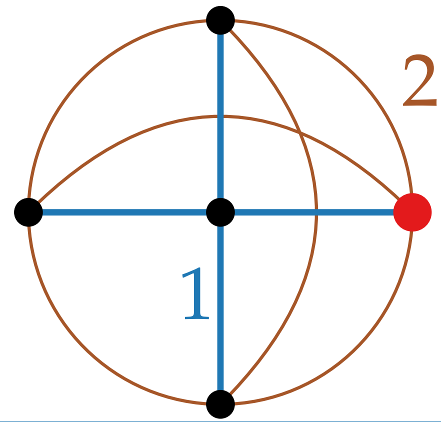
Proof. Reduce from dominating set to metric k-Center.
Given.: $G = (V, E)$, k
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center
If $\text{dom}(G) \leq k$, then $\text{cost}(S) = 1$
If $\text{dom}(G) > k$, then $\text{cost}(S) = 2$

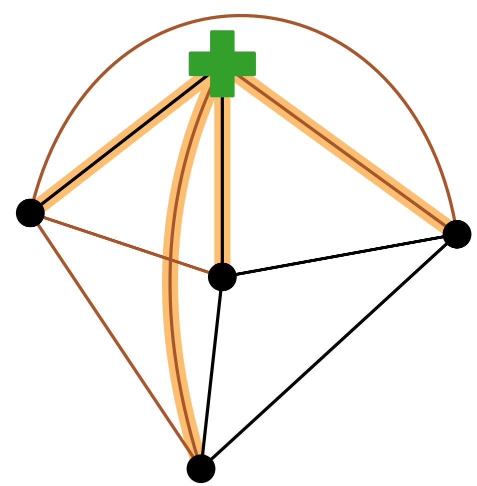
Can we do better . . . ?

What about a tight example?



Theorem. Assuming $P \neq NP$, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

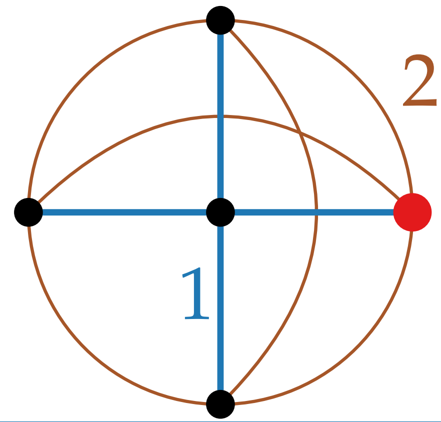
Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center
If $\mathrm{dom}(G) \leq k$, then $\mathrm{cost}(S) = 1$
If $\mathrm{dom}(G) > k$, then $\mathrm{cost}(S) = 2$

Can we do better . . . ?

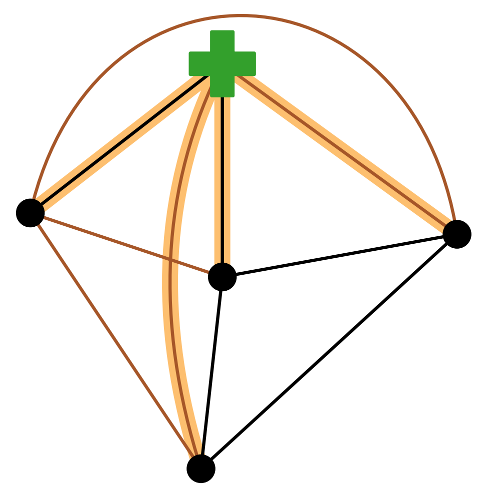
What about a tight example?



Theorem. Assuming P $\neq$ NP, there is no factor-$(2-\varepsilon)$ approximation algorithm for the metric k-CENTER problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-CENTER.
Given.: $G = (V, E)$, k
Constr. complete graph $G' = (V, E \cup E')$

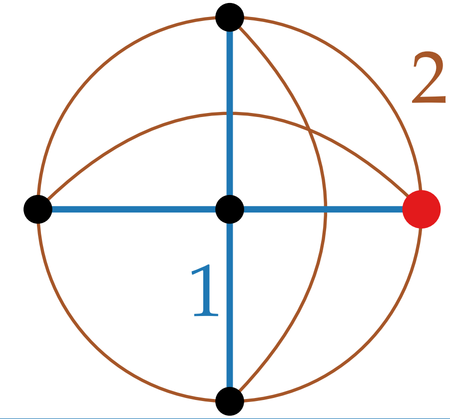
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$



S: metric k-Center
If $\mathrm{dom}(G) \leq k$, then $\mathrm{cost}(S) = 1$
If $\mathrm{dom}(G) > k$, then $\mathrm{cost}(S) = 2$

Can we do better . . . ?

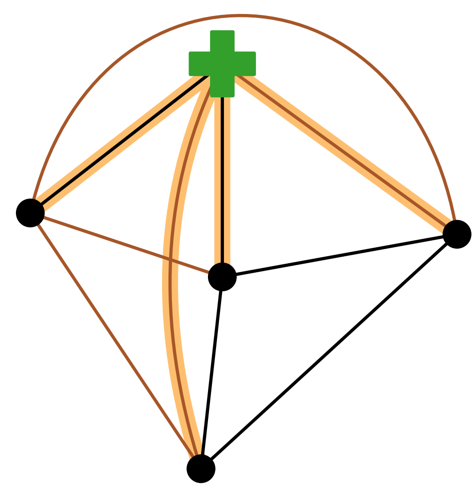
What about a tight example?



Theorem. Assuming P $\neq$ NP, there is no factor-$(2 - \varepsilon)$ approximation algorithm for the metric k-Center problem, for any $\varepsilon > 0$.

Proof. Reduce from dominating set to metric k-Center.
Given.: $G = (V, E), k$
Constr. complete graph $G' = (V, E \cup E')$
with $c(e) = \begin{cases} 1, & \text{if } e \in E \\ 2, & \text{if } e \in E' \end{cases}$

$\triangle$-inequality holds



S: metric k-Center
If $\text{dom}(G) \leq k$, then $\text{cost}(S) = 1$
If $\text{dom}(G) > k$, then $\text{cost}(S) = 2$

Approximation Algorithms

Lecture 6: k-Center via Parametric Pruning

Part V: Metric-Weighted-Center

Philipp Kindermann

Summer Semester 2020

Metric-k-Center

Given: A complete graph $G = (V, E)$ with metric edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to the a vertex in S.

Find: A k-element vertex set S, such that $\mathrm{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

METRIC-k-CENTER

WEIGHTED

Given: A complete graph $G = (V, E)$ with metric edge costs $c \colon E \to \mathbb{Q}_{\geq 0}$ and a natural number $k \leq |V|$.

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to the a vertex in S.

Find: A k-element vertex set S, such that $\text{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

METRIC-~~k~~-CENTER

WEIGHTED

Given: A complete graph $G = (V, E)$ with metric edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ and ~~a natural number $k \leq |V|$.~~ , vertex weights $w\colon V \to \mathbb{Q}_{\geq 0}$ and a weight limit $W \in \mathbb{Q}_+$

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to the a vertex in S.

Find: A k-element vertex set S, such that $\operatorname{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

Metric-~~k~~-Center

Weighted

Given: A complete graph $G = (V, E)$ with metric edge costs $c\colon E \to \mathbb{Q}_{\geq 0}$ and ~~a natural number $k \leq |V|$.~~ , vertex weights $w\colon V \to \mathbb{Q}_{\geq 0}$ and a weight limit $W \in \mathbb{Q}_{+}$

For each vertex set $S \subseteq V$, $c(v, S)$ is the cost of the cheapest edge from v to the a vertex in S.

vertex set S of weight at most W

Find: A ~~k-element vertex set S~~, such that $\mathrm{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.

Algorithm for the Weighted Version

Algorithm Metric- -CENTER

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

  if $|U_j| \leq k$ **then**

    return U_j

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

- Construct G_j^2
- Find a maximal independent set U_j in G_j^2
- **if** $|U_j| \leq k$ **then**
  - **return** U_j

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

- Construct G_j^2
- Find a maximal independent set U_j in G_j^2
  - what about the weights?
- **if** $|U_j| \leq k$ **then**
  - **return** U_j

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

  Construct G_j^2

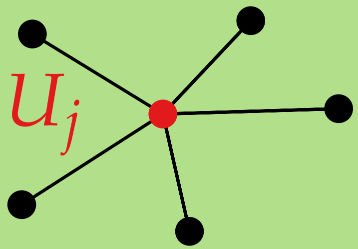
  Find a maximal independent set U_j in G_j^2

  what about the weights?

  if $|U_j| \leq k$ **then**

    return U_j

$u \in U_j$



Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

Construct G_j^2

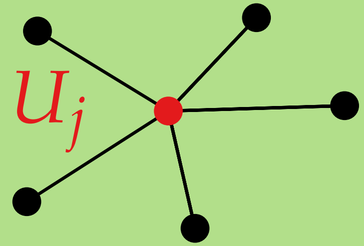
Find a maximal independent set U_j in G_j^2

what about the weights?

if $|U_j| \leq k$ **then**

return U_j

$u \in U_j$



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

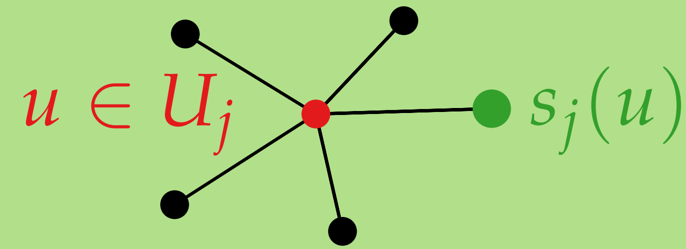
Construct G_j^2

Find a maximal independent set U_j in G_j^2

if $|U_j| \leq k$ **then**

return U_j

what about the weights?



$s_j(u)$:= lightest node in $N_{G_j}(u) \cup \{u\}$

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

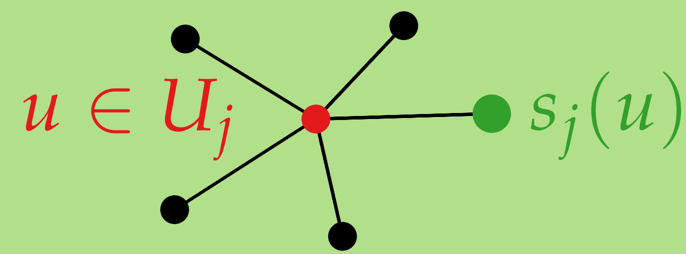
  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

  Compute $S_j := \{ s_j(u) \mid u \in U_j \}$

  if $|U_j| \leq k$ **then**

    return U_j



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

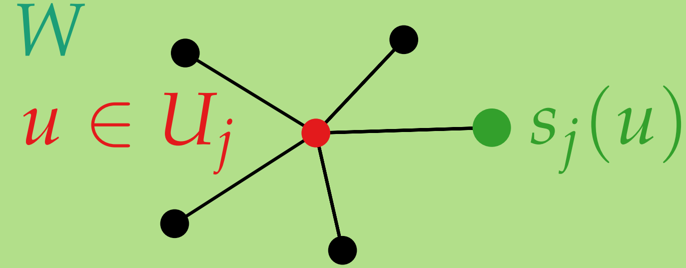
  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

  Compute $S_j := \{ s_j(u) \mid u \in U_j \}$

  if ~~$|U_j| \leq k$~~ $w(S_j) \leq W$ **then**

    return U_j



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

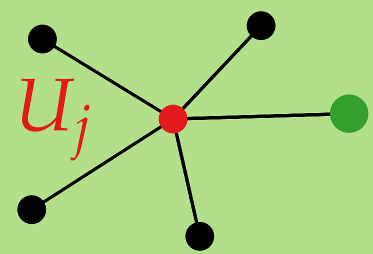
Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

- Construct G_j^2
- Find a maximal independent set U_j in G_j^2
- Compute $S_j := \{ s_j(u) \mid u \in U_j \}$
- **if** ~~$|U_j| \leq k$~~ $w(S_j) \leq W$ **then**
  - **return** ~~U_j~~ S_j



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

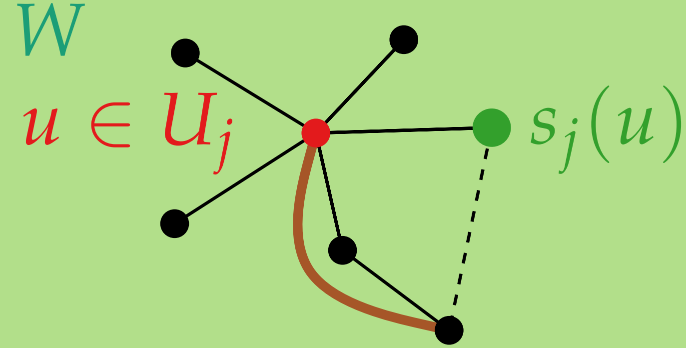
Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

- Construct G_j^2
- Find a maximal independent set U_j in G_j^2
- Compute $S_j := \{ s_j(u) \mid u \in U_j \}$
- **if** ~~$|U_j| \leq k$~~ $w(S_j) \leq W$ **then**
  - **return** ~~U_j~~ S_j

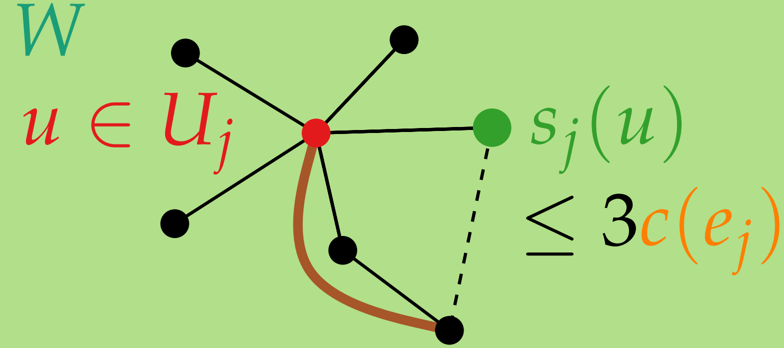


$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$
for $j = 1, \ldots, m$ **do**
  Construct G_j^2
  Find a maximal independent set U_j in G_j^2
  Compute $S_j := \{ s_j(u) \mid u \in U_j \}$
  if ~~$|U_j| \leq k$~~ $w(S_j) \leq W$ **then**
    return ~~U_j~~ S_j



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

Algorithm for the Weighted Version

Algorithm Metric-Weighted-Center

Sort the edges of G by cost : $c(e_1) \leq \ldots \leq c(e_m)$

for $j = 1, \ldots, m$ **do**

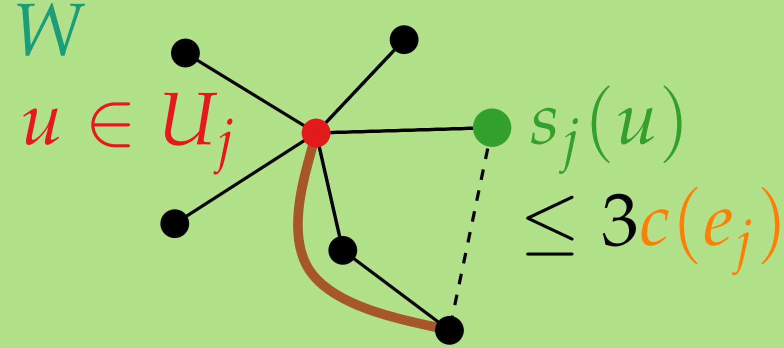
  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

  Compute $S_j := \{ s_j(u) \mid u \in U_j \}$

  if ~~$|U_j| \leq k$~~ $w(S_j) \leq W$ **then**

    return ~~U_j~~ S_j



$s_j(u) :=$ lightest node in $N_{G_j}(u) \cup \{u\}$

Theorem. The above is a factor-3-approximation algorithm for Metric-Weighted-Center.

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Tight Example... ?

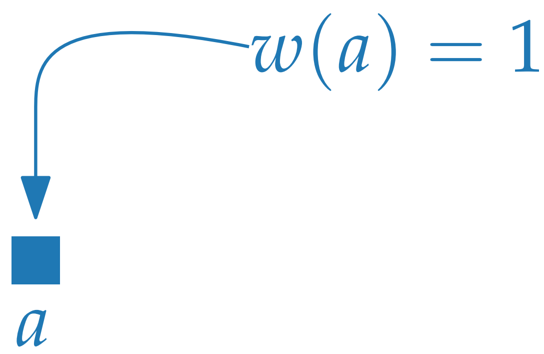
Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

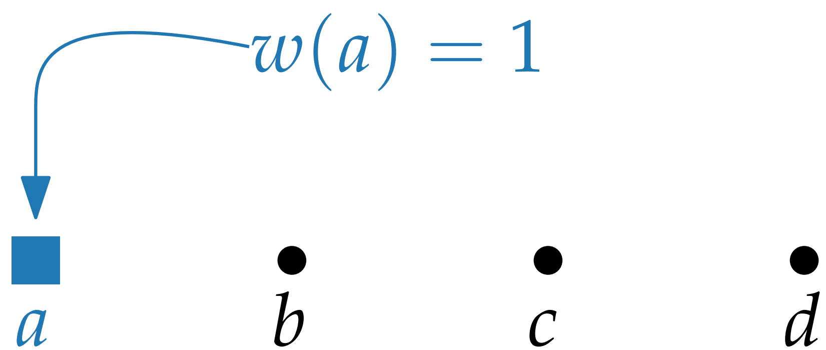
Consider $W = 3$



Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

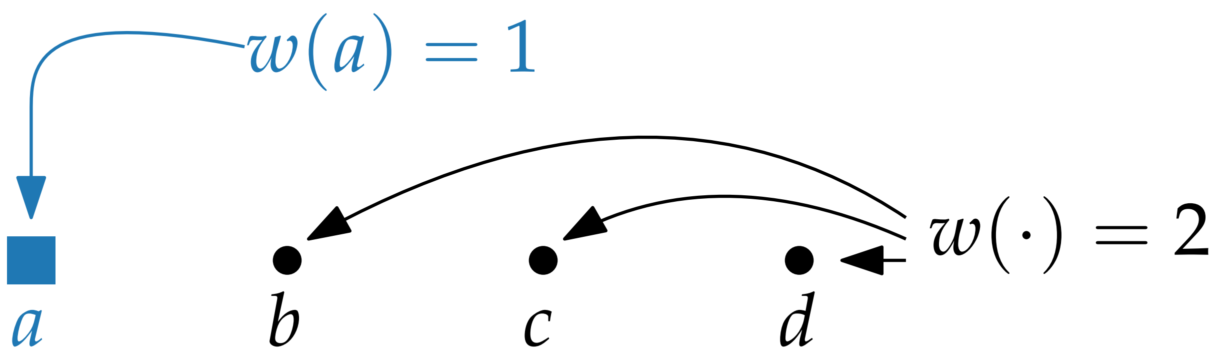
Consider $W = 3$



Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

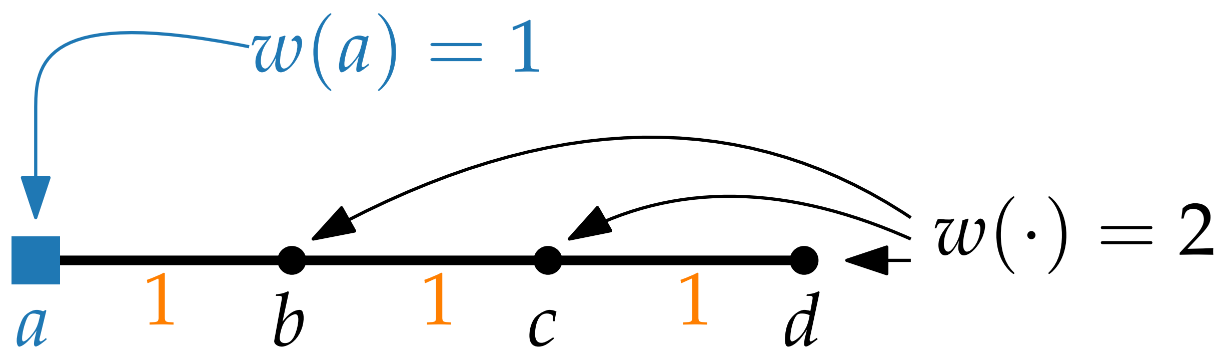
Consider $W = 3$



Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

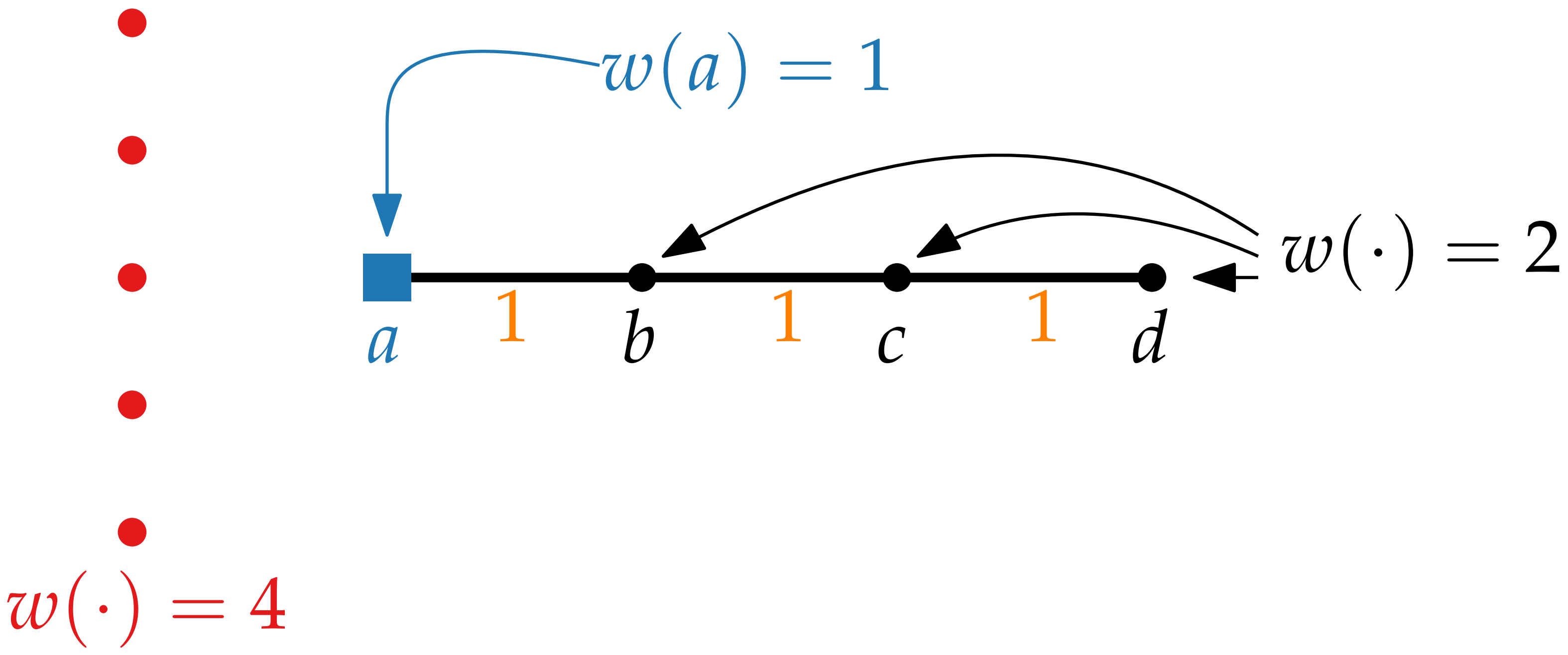
Consider $W = 3$



Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

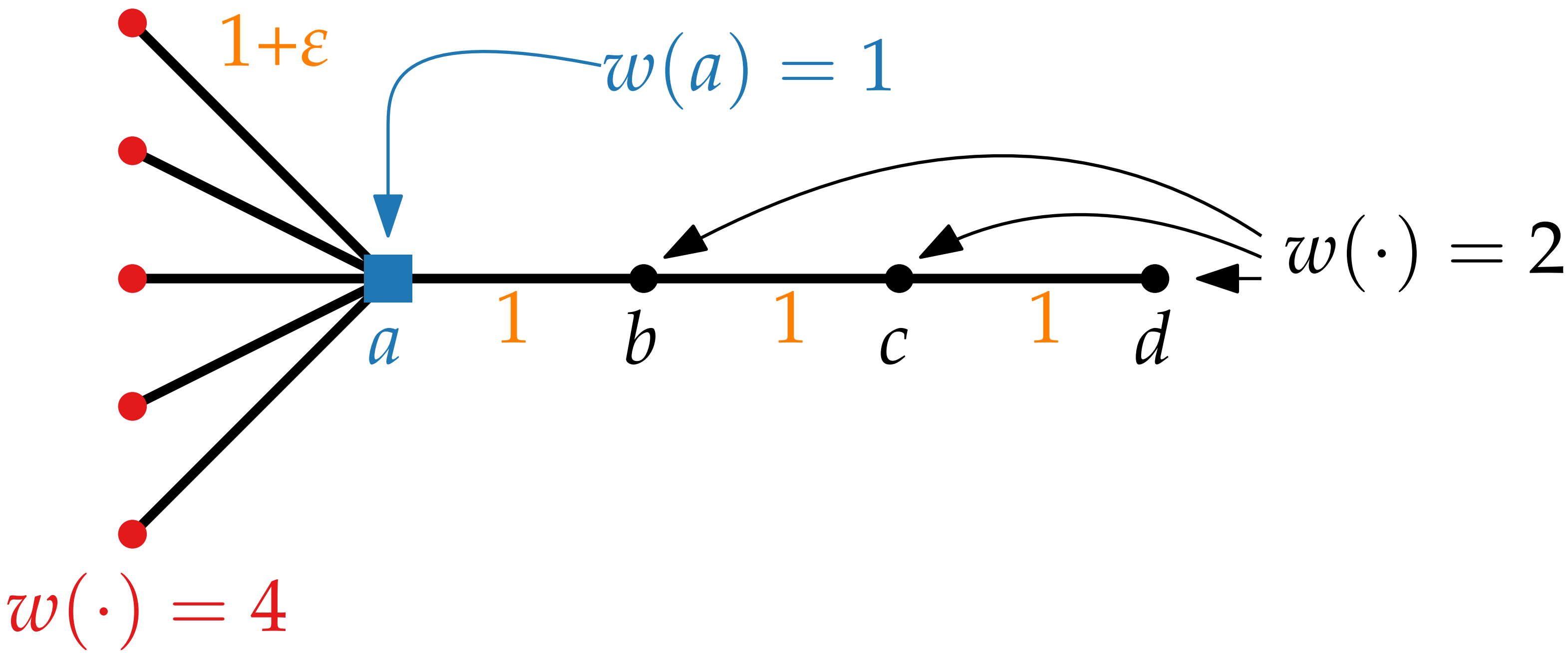
Consider $W = 3$



Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

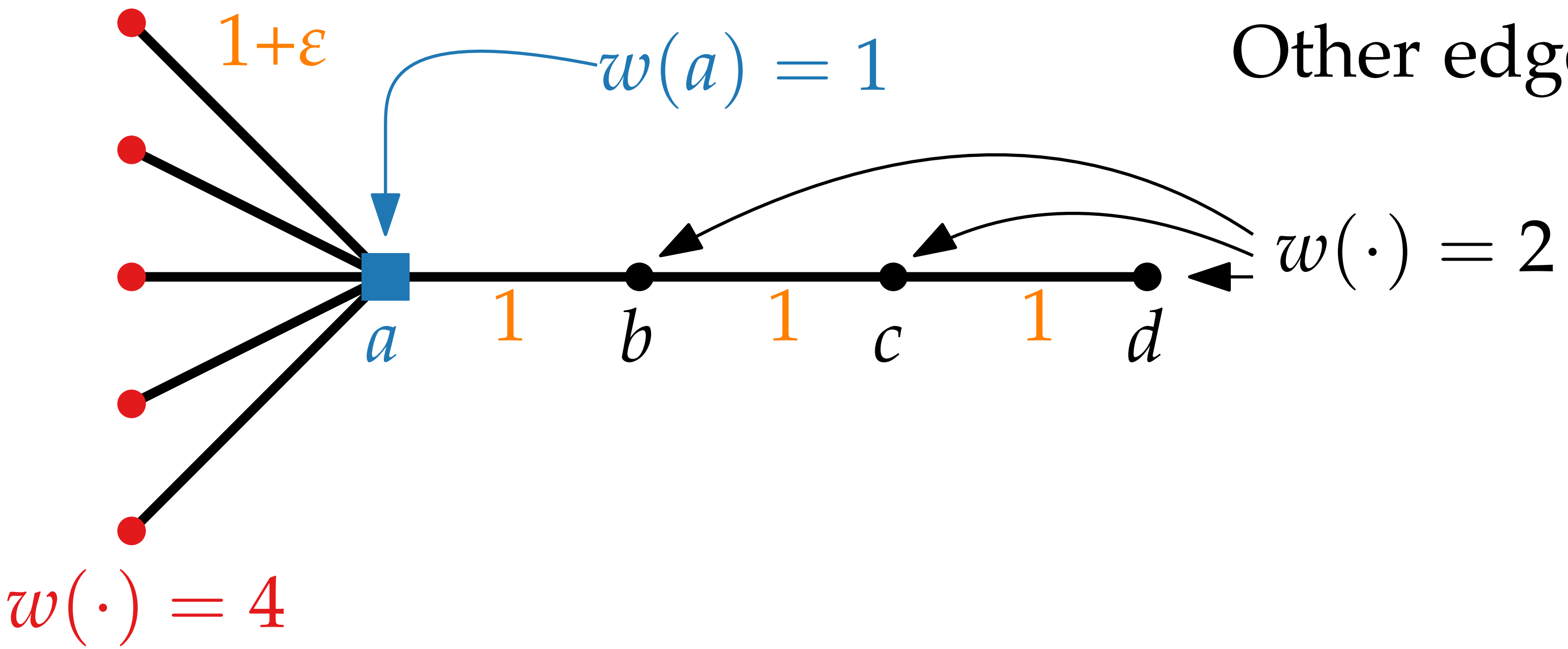


Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs?

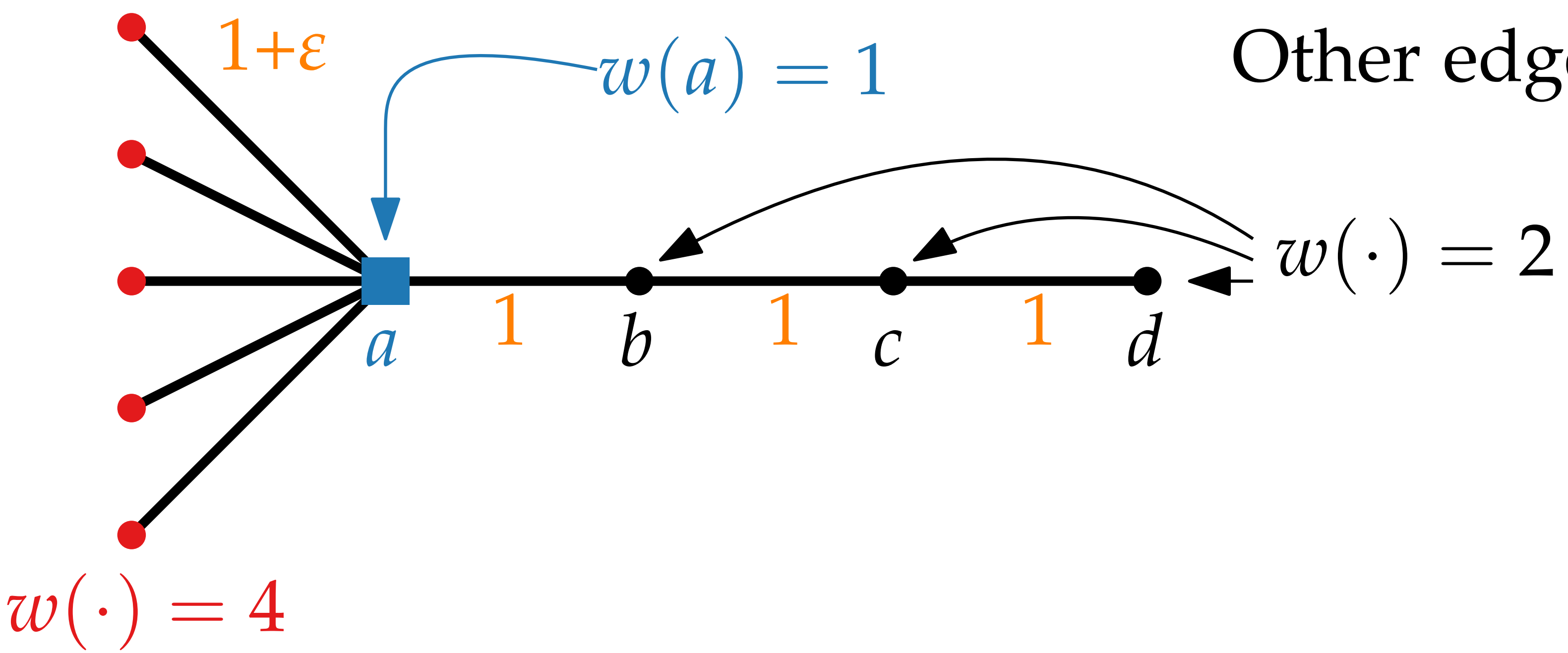


Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs? 2

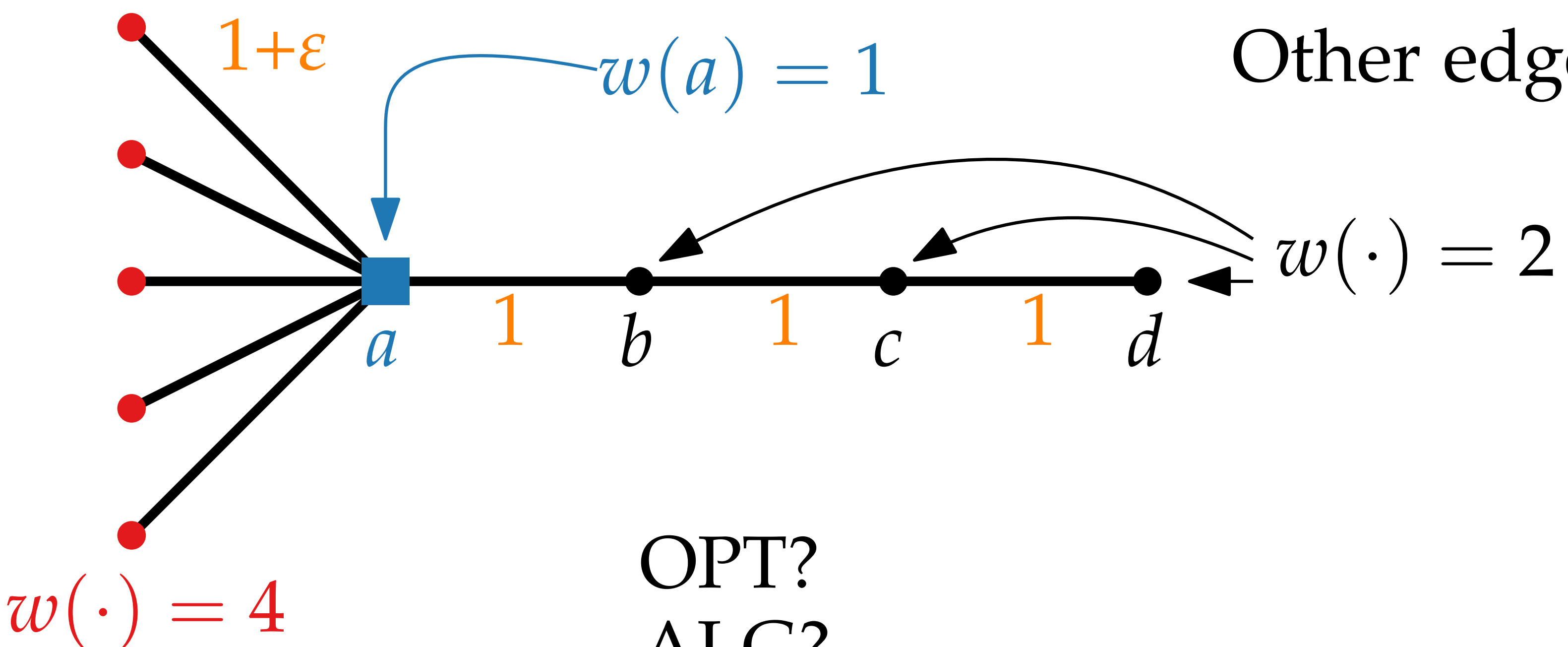


Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs? 2



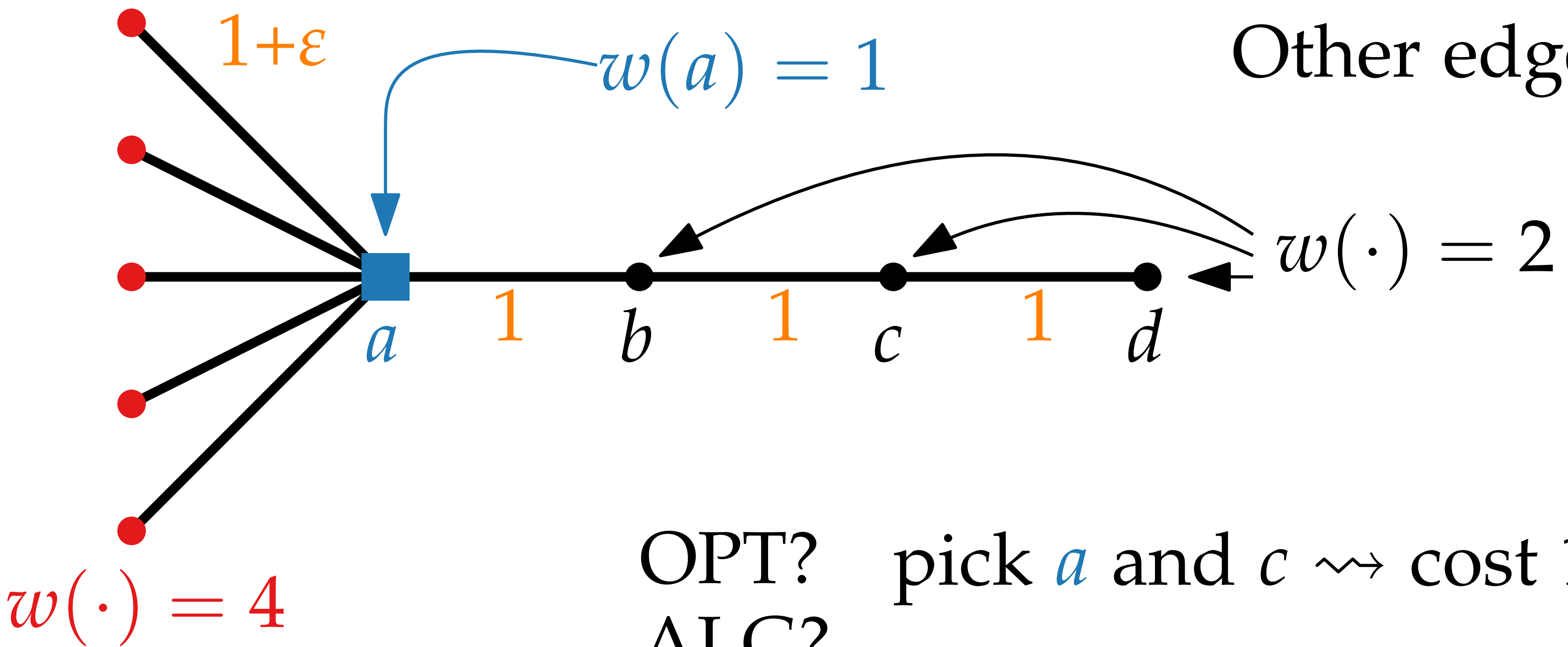
OPT?
ALG?

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs? 2



OPT? pick a and $c \rightsquigarrow$ cost $1 + \varepsilon$

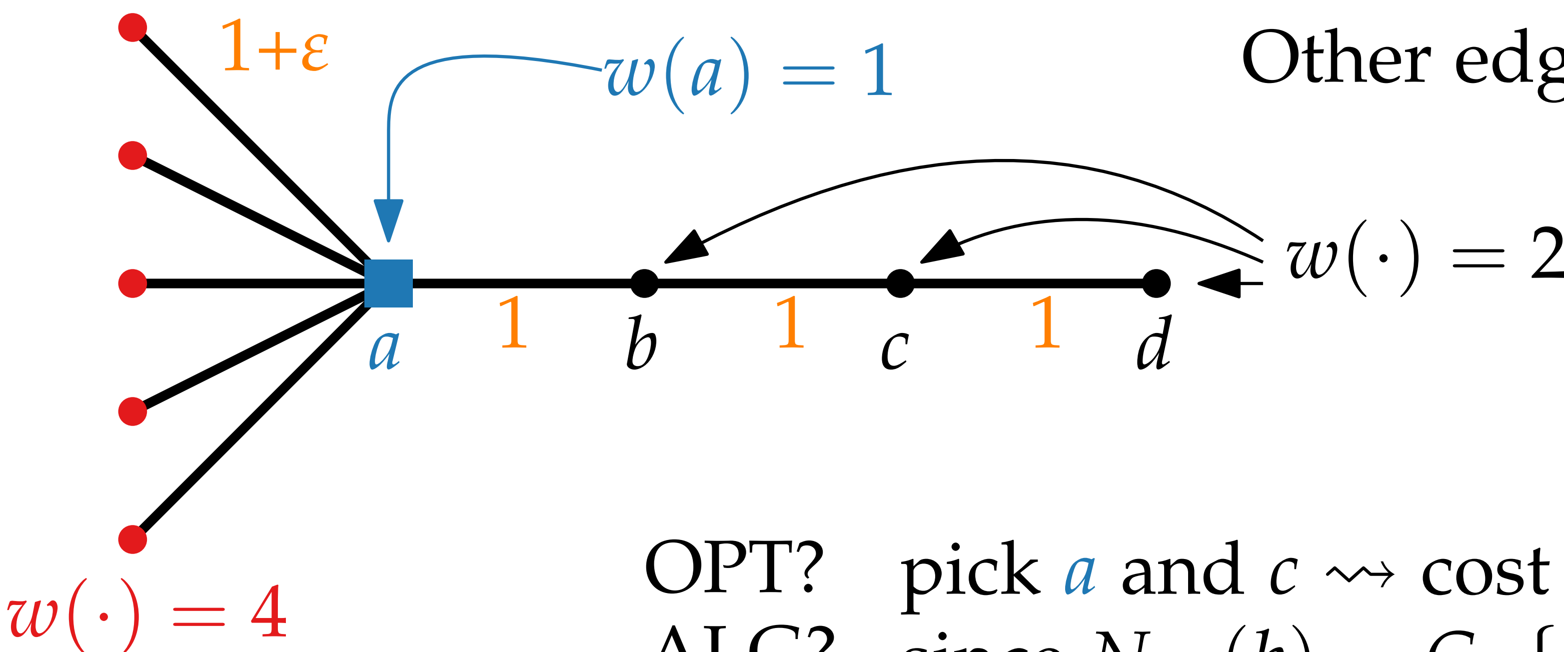
ALG?

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs? 2



OPT? pick a and c $\rightsquigarrow$ cost $1 + \varepsilon$

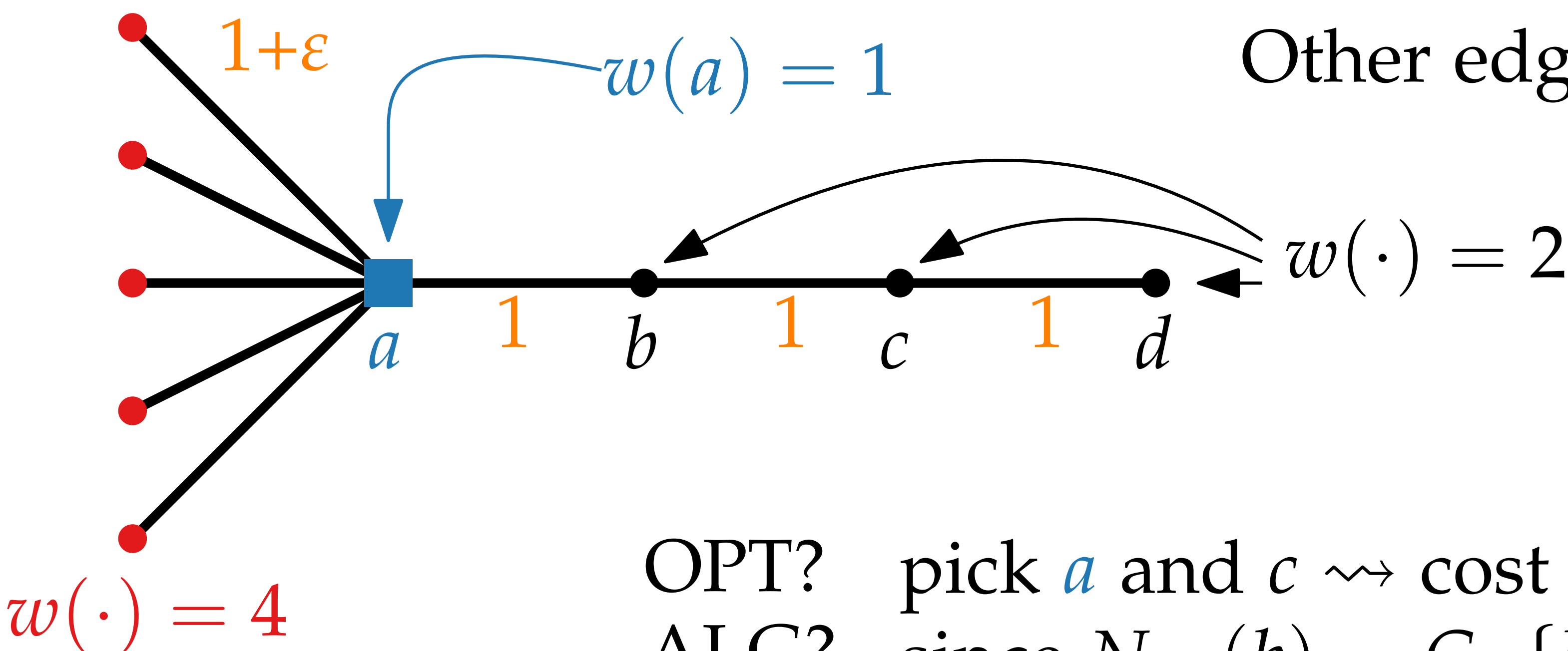
ALG? since $N_{G^2}(b) = G$, $\{b\}$ is a maximal independent set in G^2

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

Consider $W = 3$

Other edge costs? 2



OPT? pick a and c $\rightsquigarrow$ cost $1 + \varepsilon$

ALG? since $N_{G^2}(b) = G$, $\{b\}$ is a maximal independent set in G^2

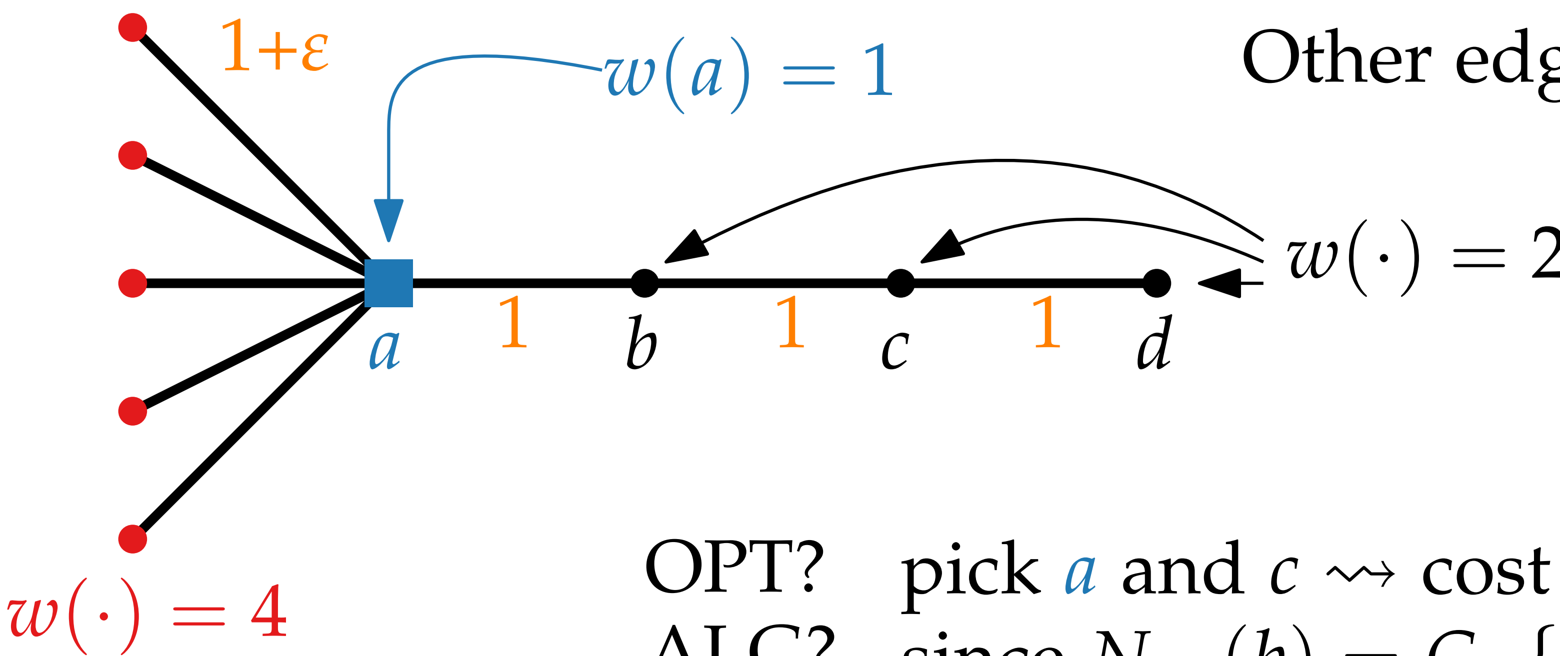
Thus, alg. picks a $\rightsquigarrow$ cost 3

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

~~Consider $W = 3$~~

Other edge costs? 2



OPT? pick a and $c \rightsquigarrow$ cost $1 + \varepsilon$

ALG? since $N_{G^2}(b) = G$, $\{b\}$ is a maximal independent set in G^2

Thus, alg. picks $a \rightsquigarrow$ cost 3

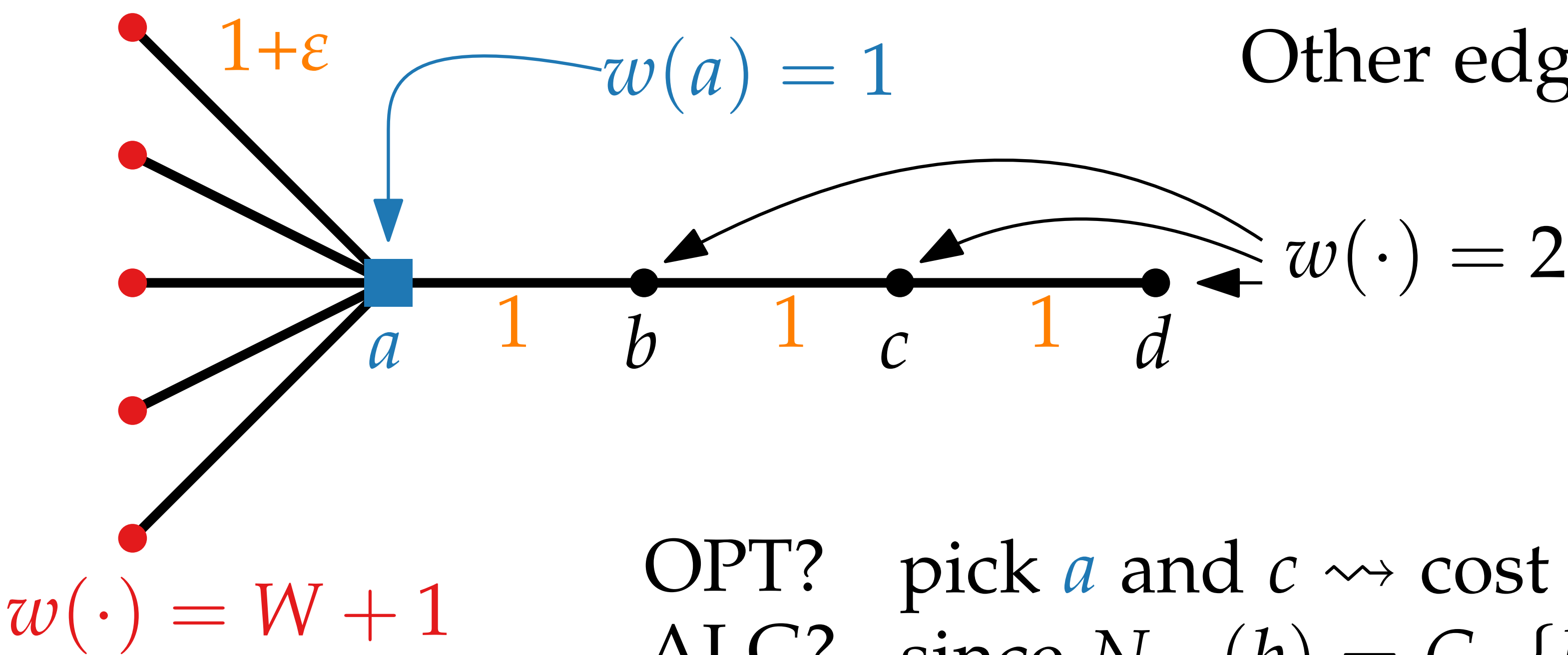
How can we generalize this to larger W?

Tight Example... ?

Here, we need to have a weight limit W, and edge costs satisfying the triangle inequality.

~~Consider $W = 3$~~

Other edge costs? 2



OPT? pick a and c $\rightsquigarrow$ cost $1 + \varepsilon$

ALG? since $N_{G^2}(b) = G$, $\{b\}$ is a maximal independent set in G^2

Thus, alg. picks a $\rightsquigarrow$ cost 3

How can we generalize this to larger W?