Aalto University
School of Science

Department of Computer Science

Combinatorics of Efficient Computations

Approximation Algorithms

Lecture 6: k-Center via Parametric Pruning

Joachim Spoerhase

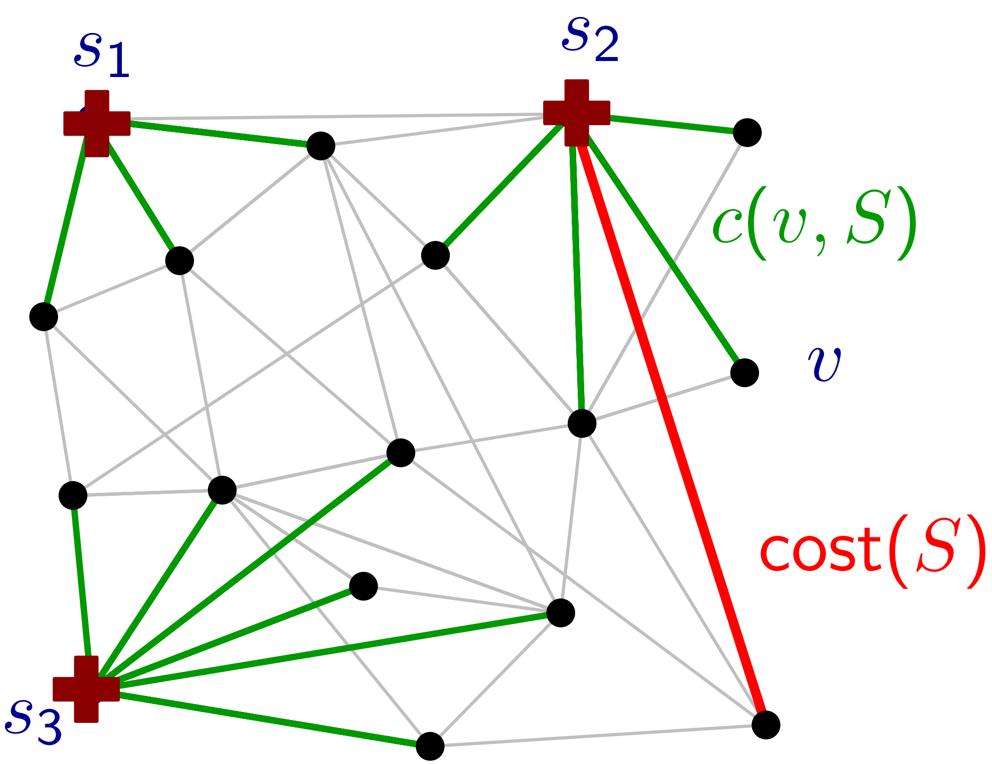
2019

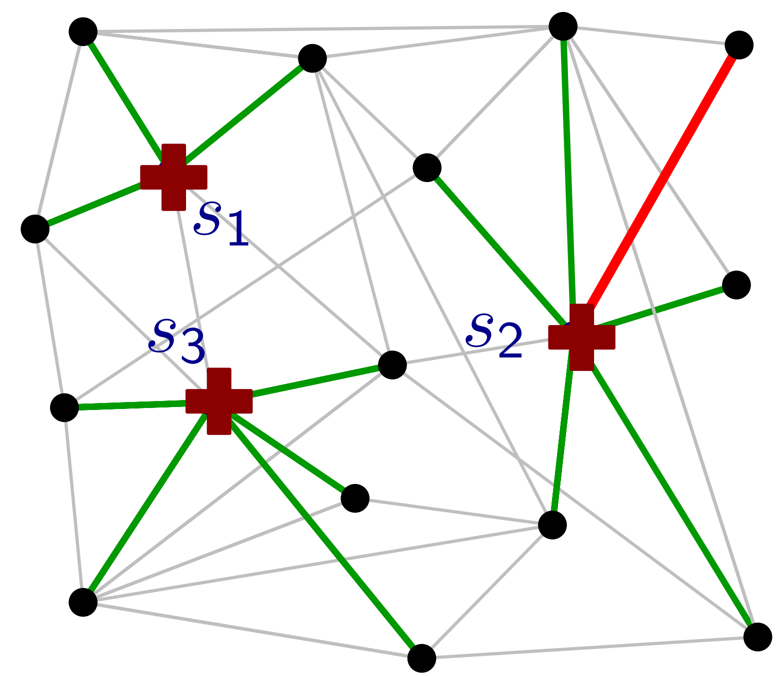
The metric k-Center-Problem

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 the a vertex in S.

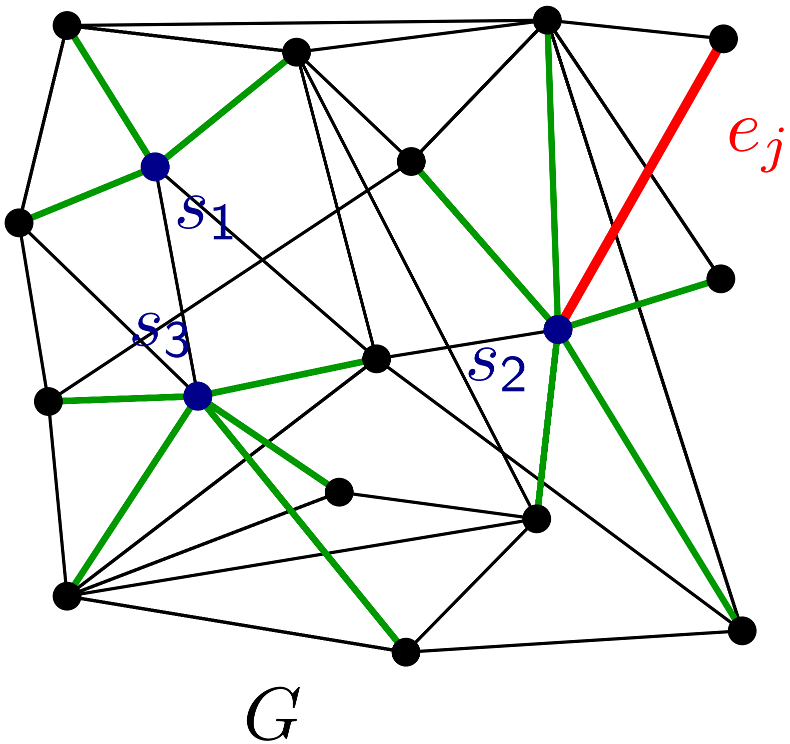
Find: A k-element vertex set S, such that $\text{cost}(S) := \max_{v \in V} c(v, S)$ is minimized.





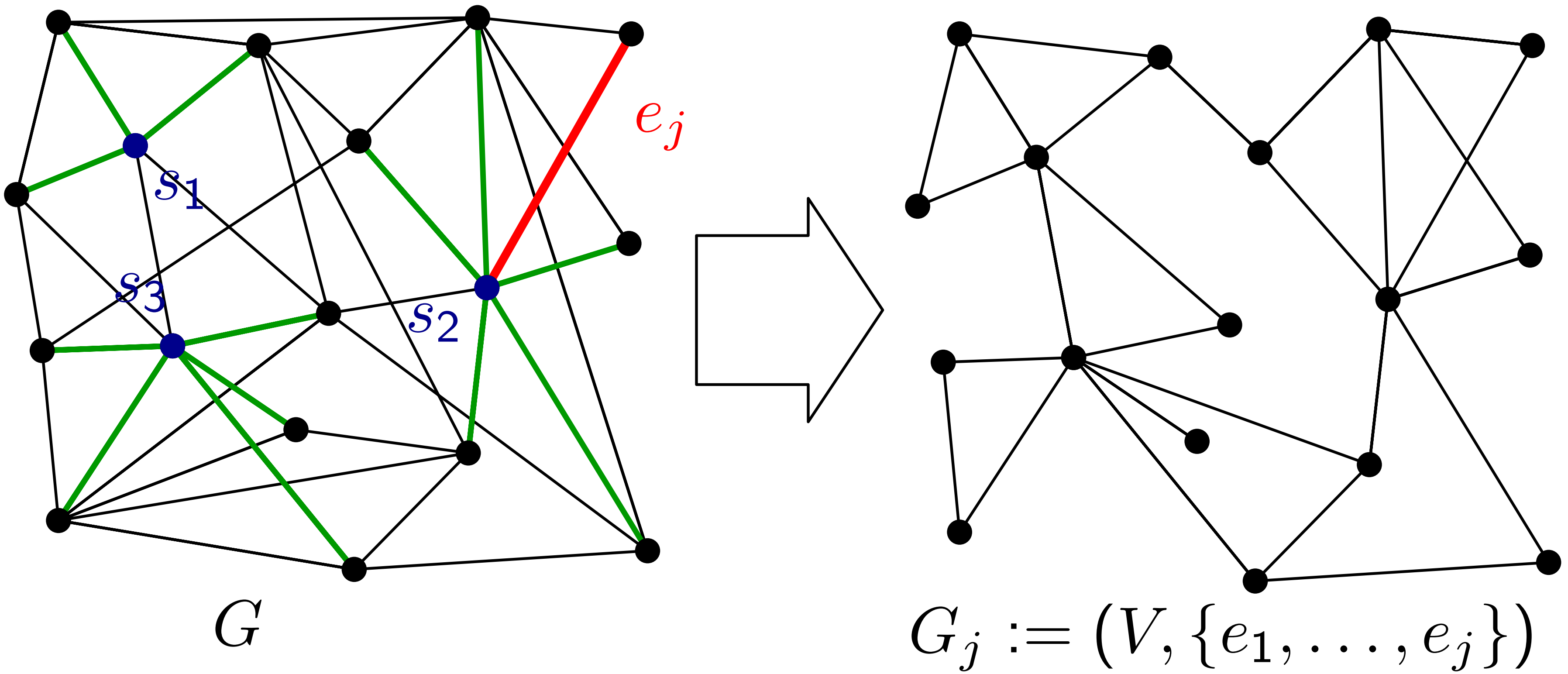
Parametric Pruning

Suppose we know OPT $= c(e_j)$ where $c(e_1) \leq \ldots \leq c(e_m)$.



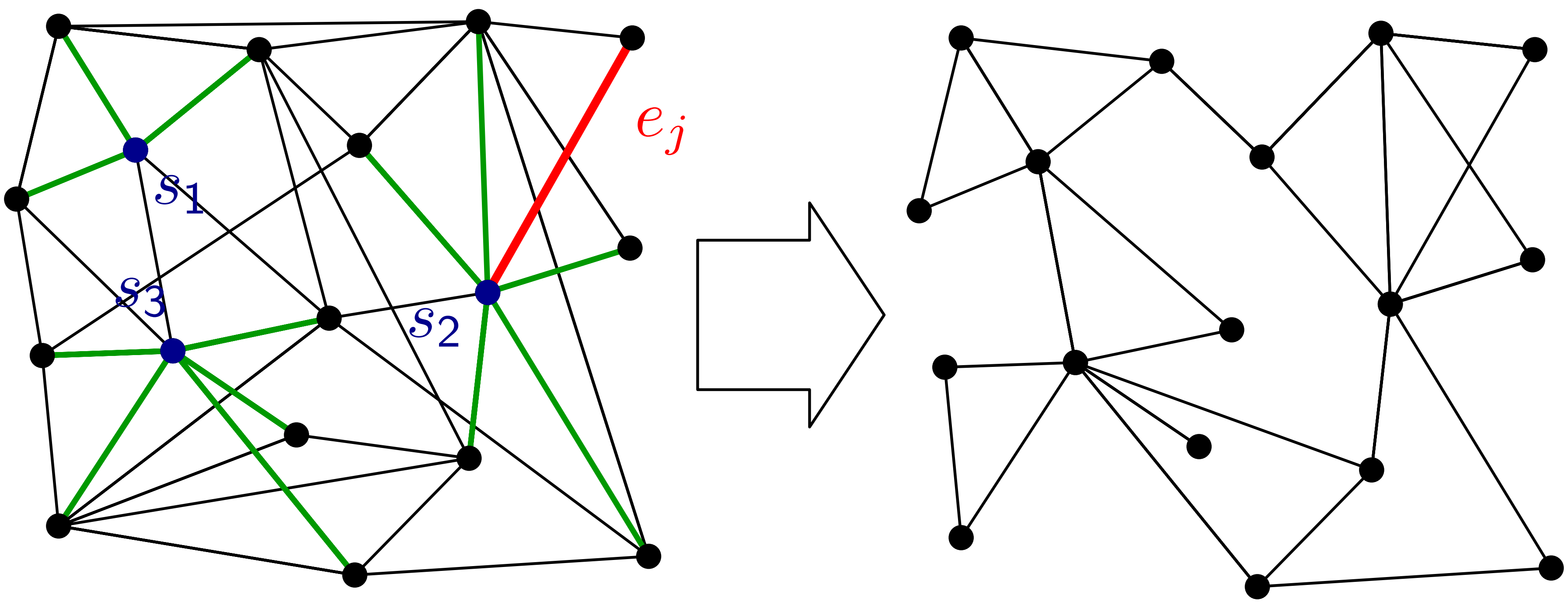
Parametric Pruning

Suppose we know $\text{OPT} = c(e_j)$ where $c(e_1) \leq \ldots \leq c(e_m)$.



Parametric Pruning

Suppose we know $\mathsf{OPT} = c(e_j)$ where $c(e_1) \leq \ldots \leq c(e_m)$.



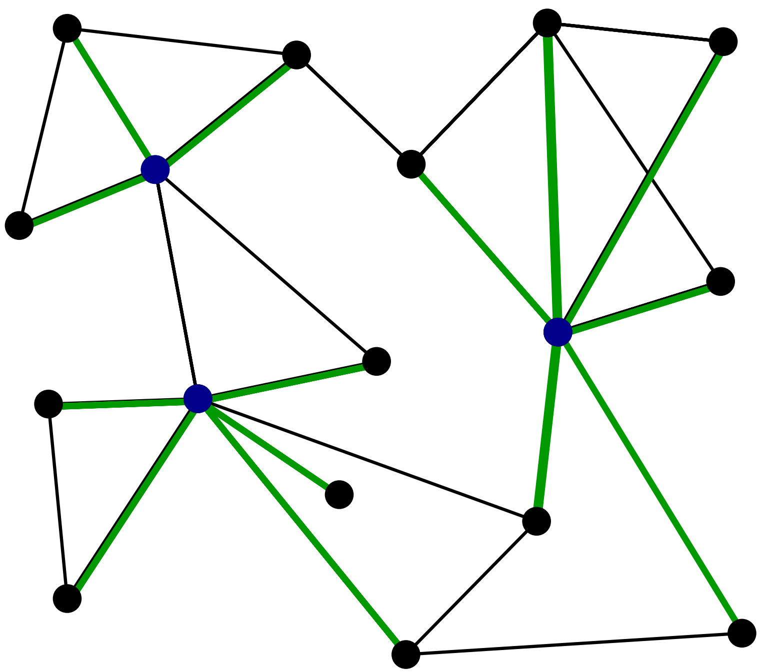
G

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

…try each G_i.

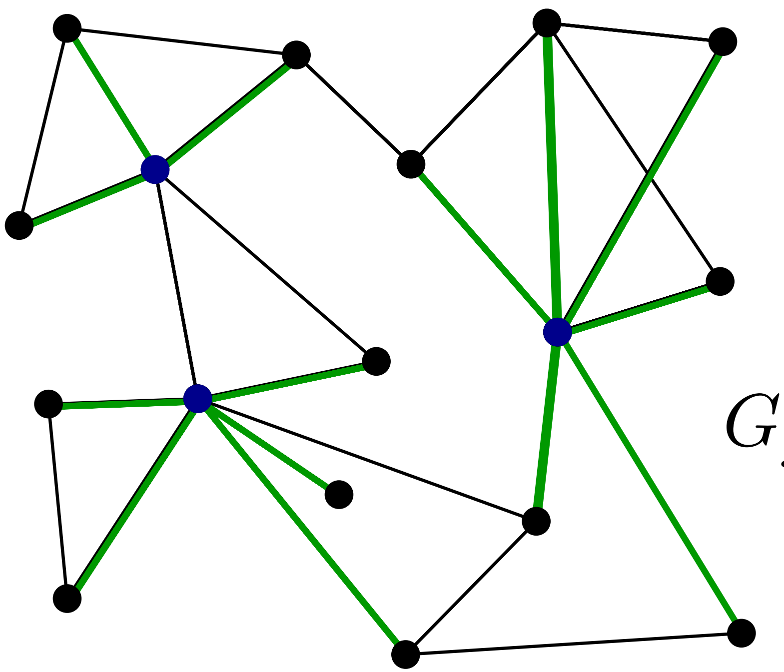
Dominating Set

Def. A vertex set D of a graph H is **dominating**, when 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 $\text{dom}(H)$.



Dominating Set

Def. A vertex set D of a graph H is **dominating**, when 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 $\text{dom}(H)$.

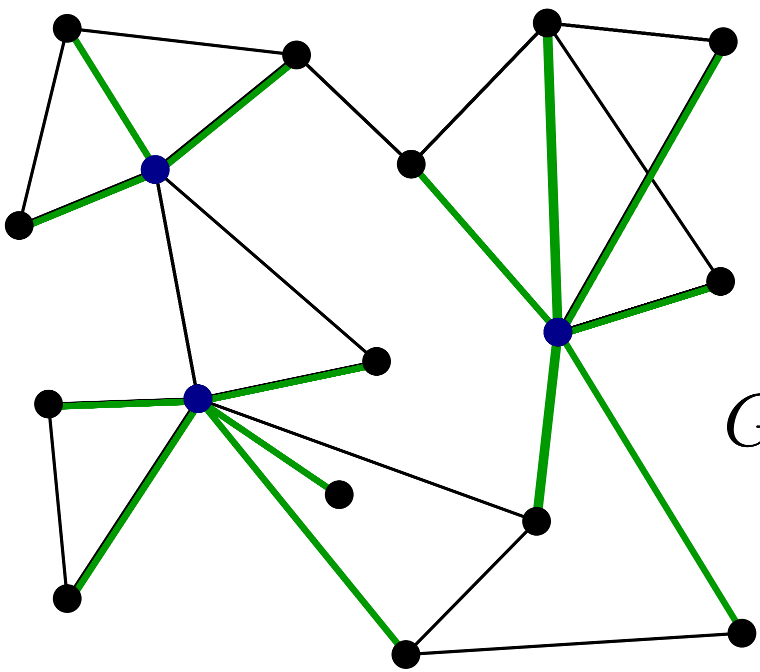


$$\text{dom}(G_j) \leq k$$

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

Dominating Set

Def. A vertex set D of a graph H is **dominating**, when 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 $\text{dom}(H)$.



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

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

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

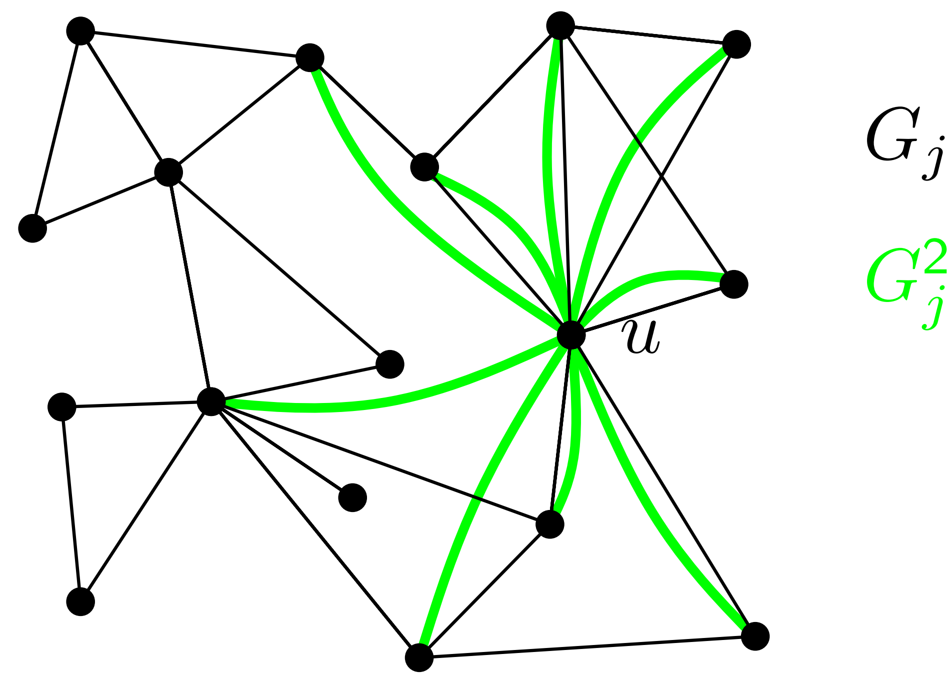
Square of a Graph

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

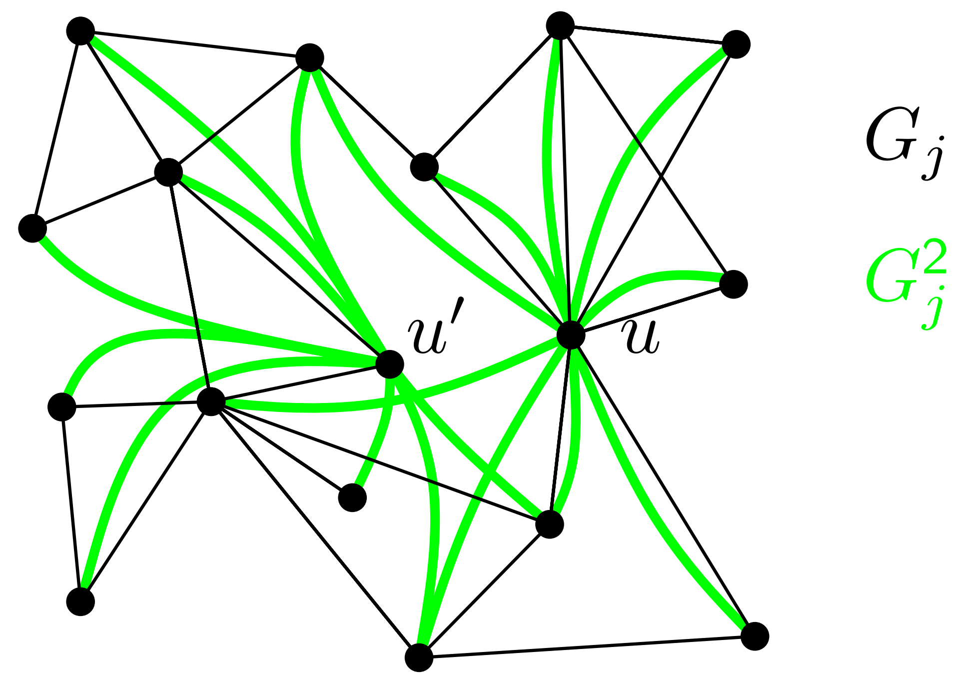
Def. The **square** H^2 of a graph H has the same vertex set as H. Additionally, two vertices $u \neq v$ are adjacent in H^2 when they are within distance **two** in H.



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.

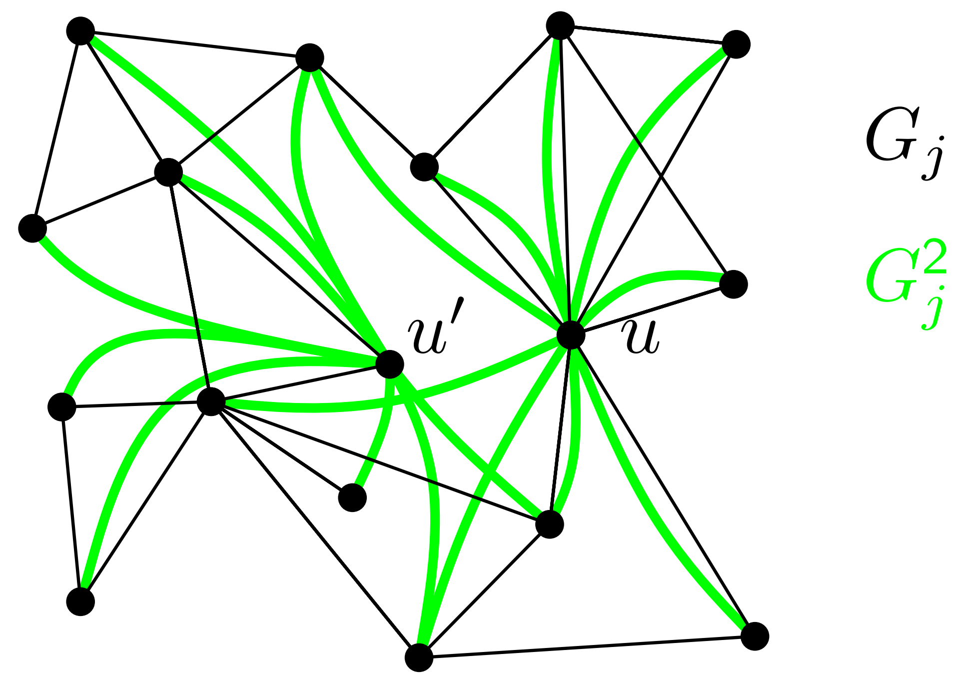


Square of a Graph

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.



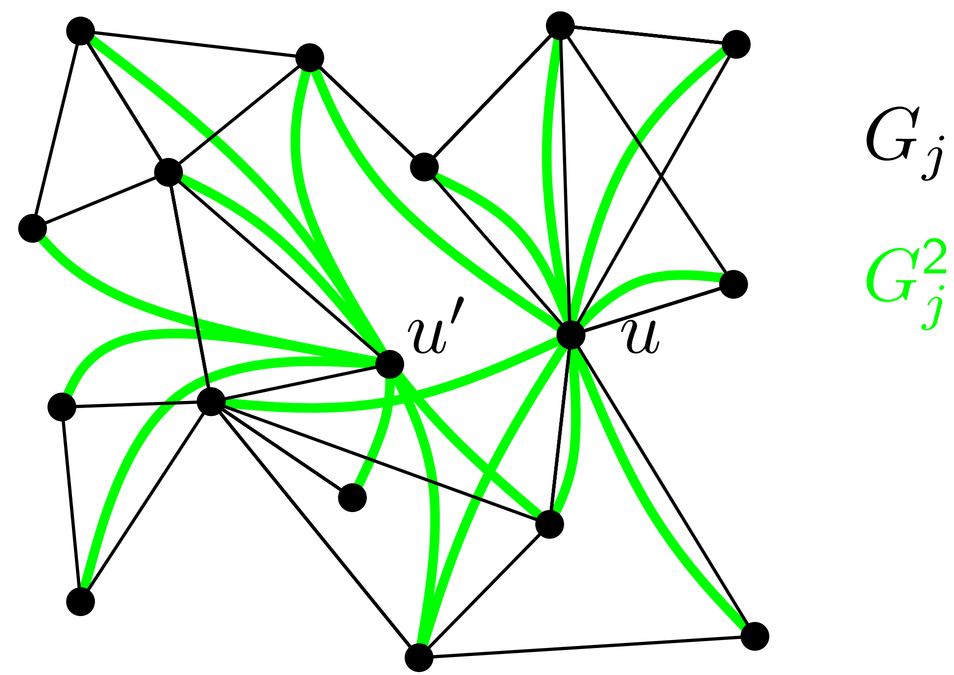
Square of a Graph

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?



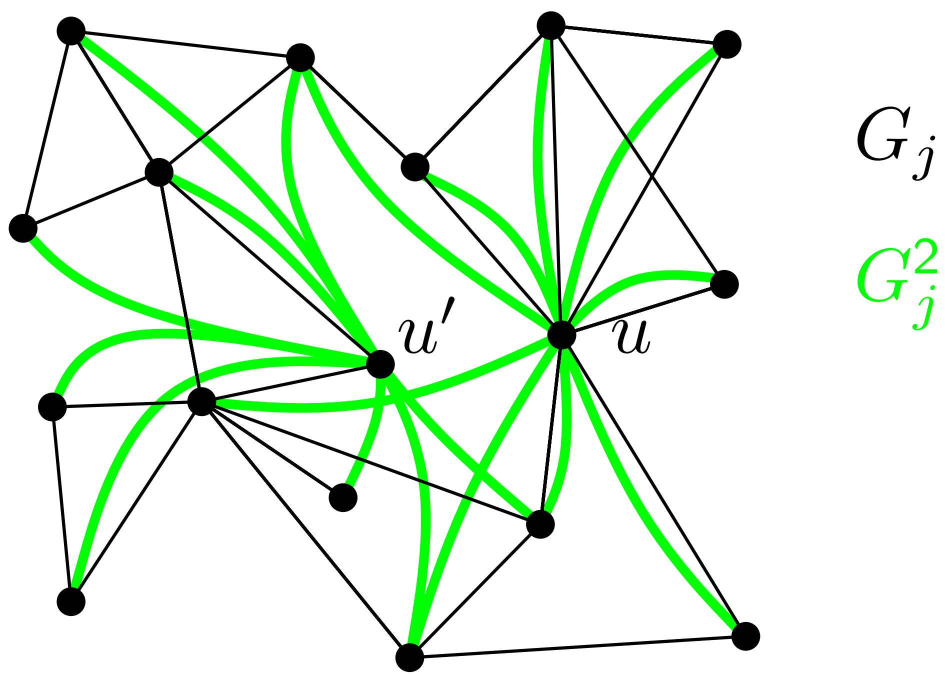
Square of a Graph

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.

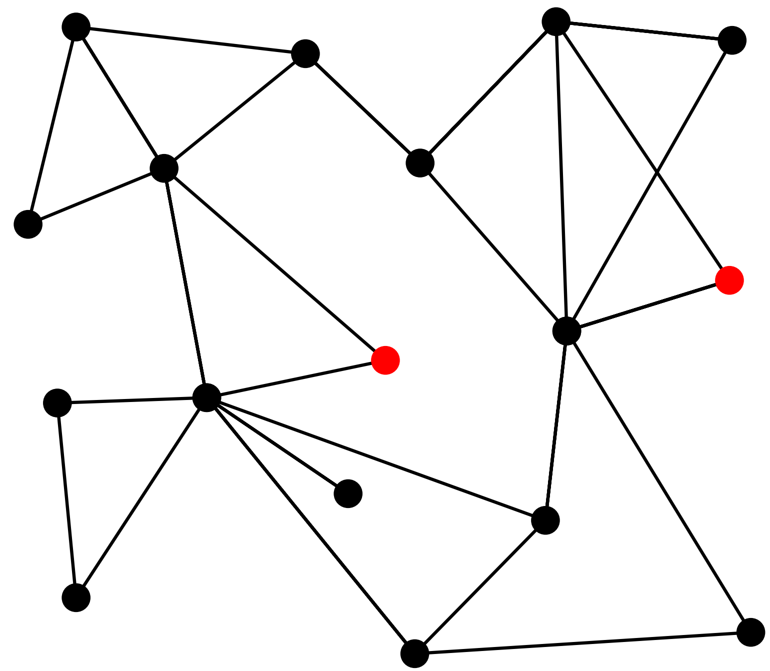
Obs. A dominating set in G_j^2 with $\leq k$ elements is already a 2-Approximation.

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



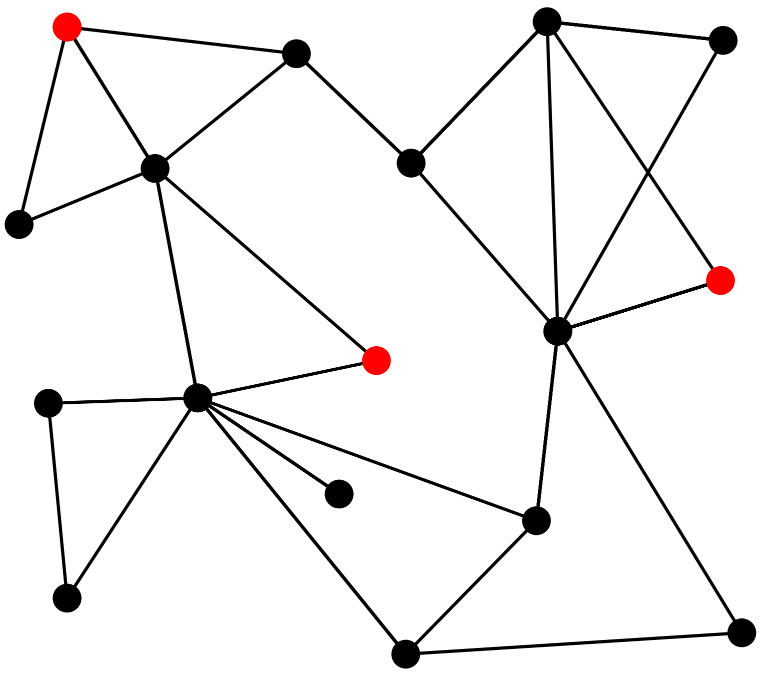
Independent sets

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



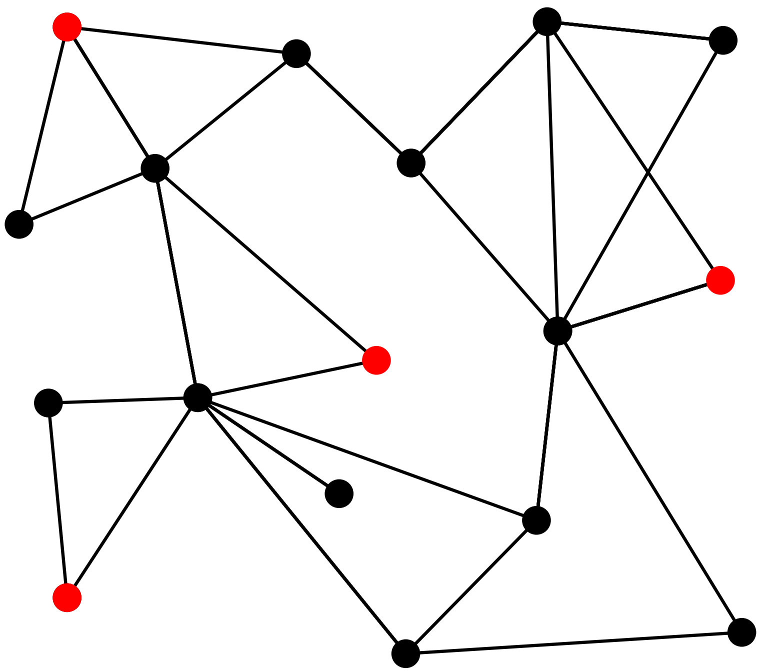
Independent sets

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



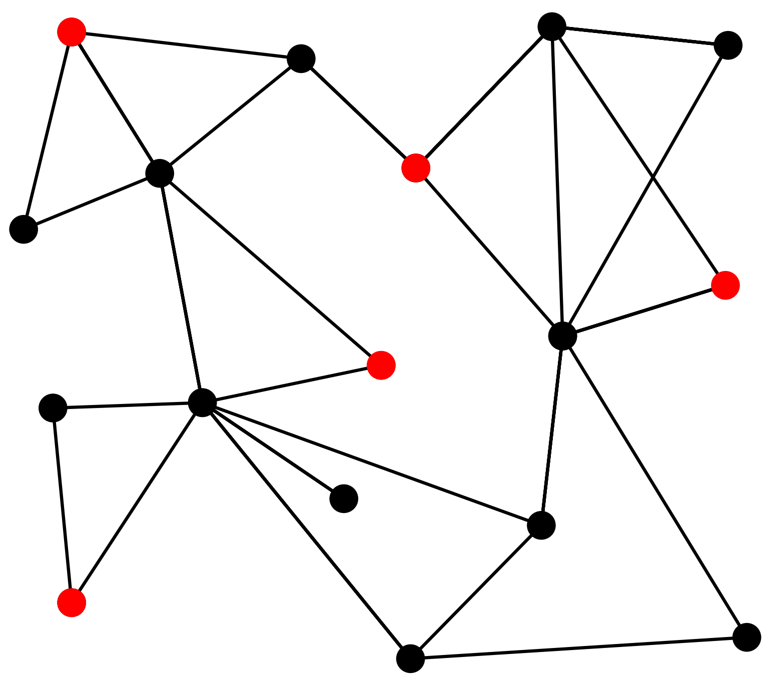
Independent sets

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



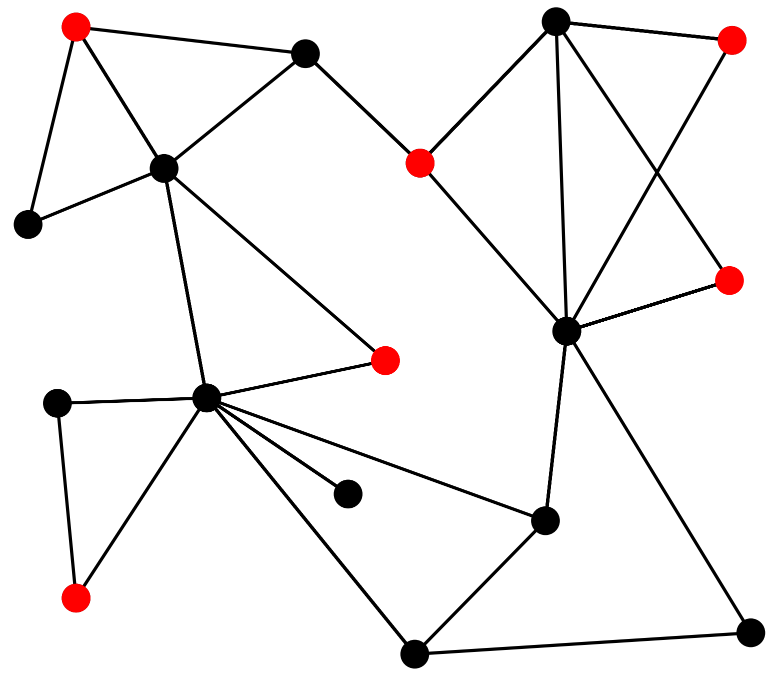
Independent sets

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



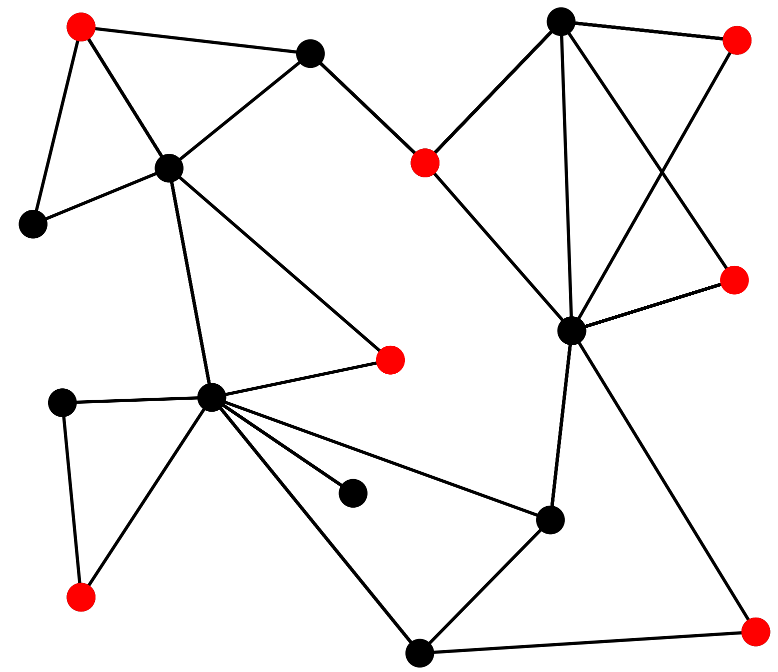
Independent sets

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



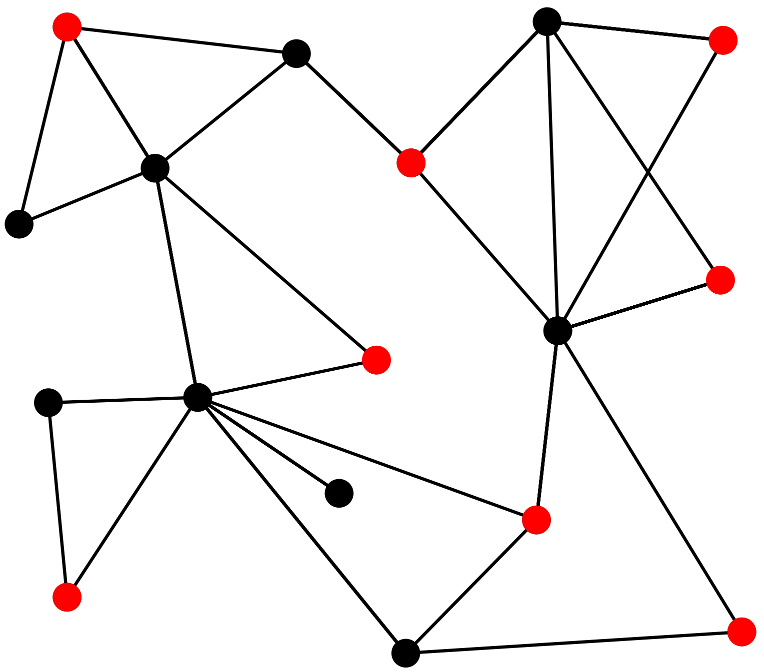
Independent sets

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



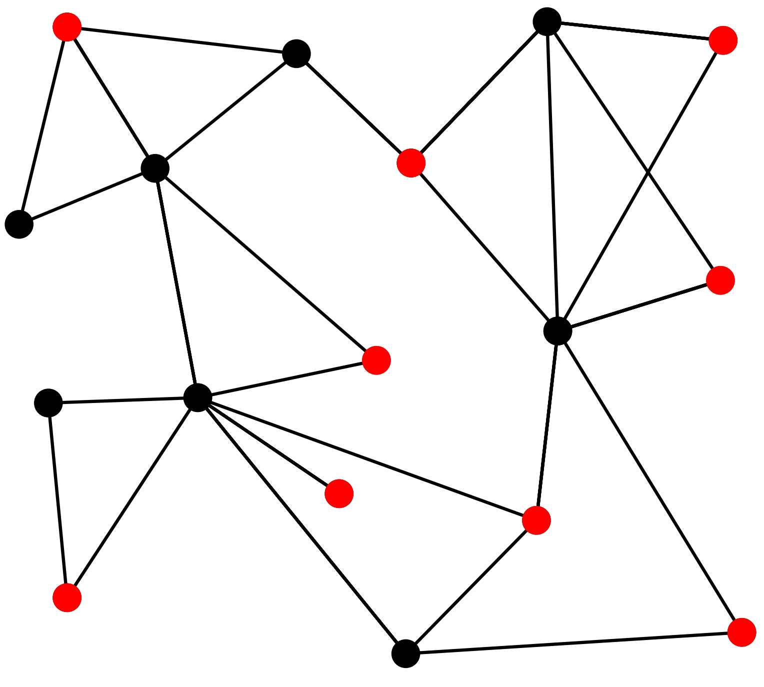
Independent sets

Def. A vertex set U in a graph is called **independent** (or **stable**), when 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

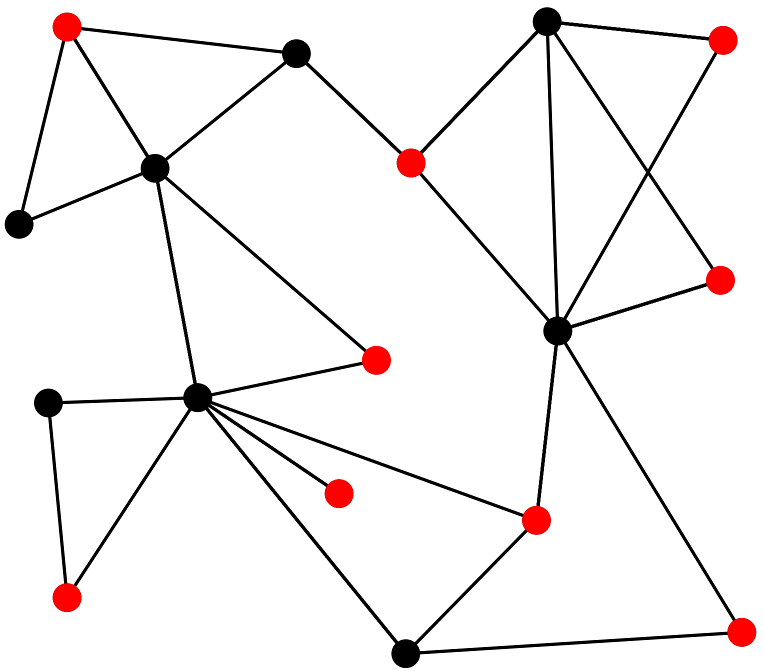
Def. A vertex set U in a graph is called **independent** (or **stable**), when no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.



Independent sets

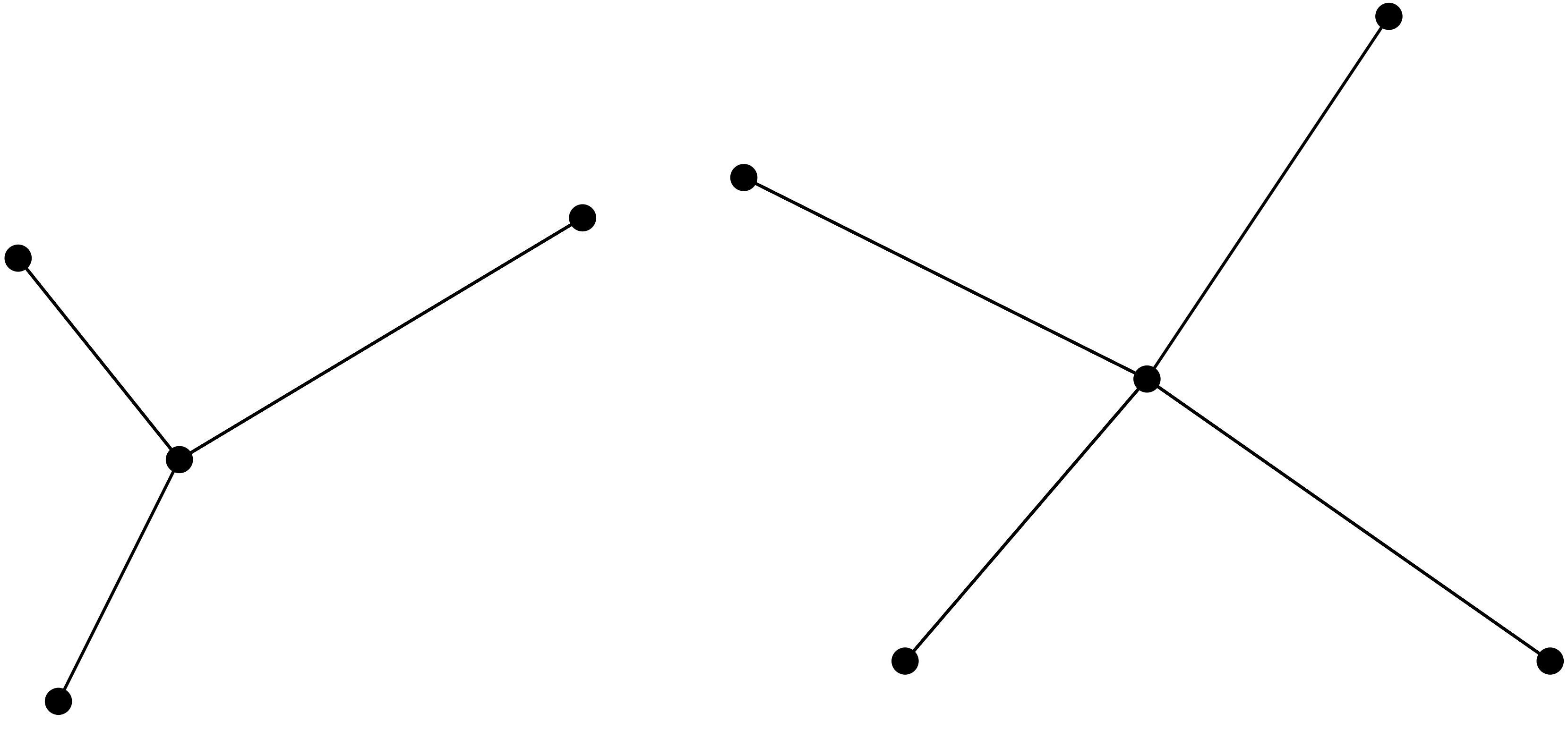
Def. A vertex set U in a graph is called **independent** (or **stable**), when no pair of vertices in U form an edge. An independent set is called **maximal** when no superset of it is an independent set.

Obs. Maximal independent sets are dominating sets :-)



Independent sets in H^2

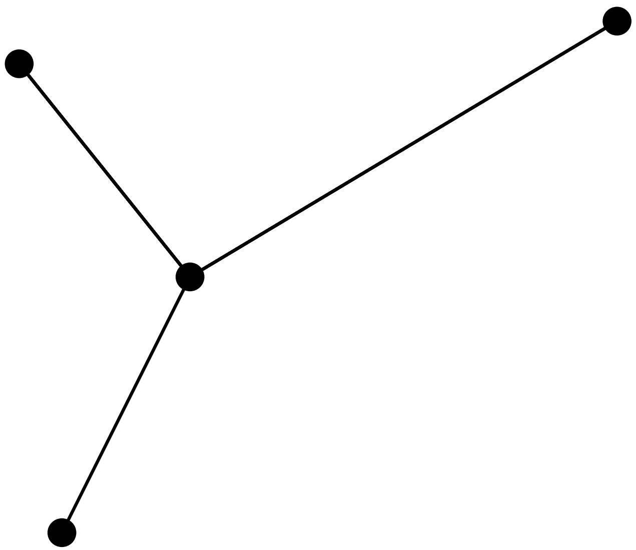
Lem. For a graph H and an independent set U in H^2, $|U| \leq \text{dom}(H)$

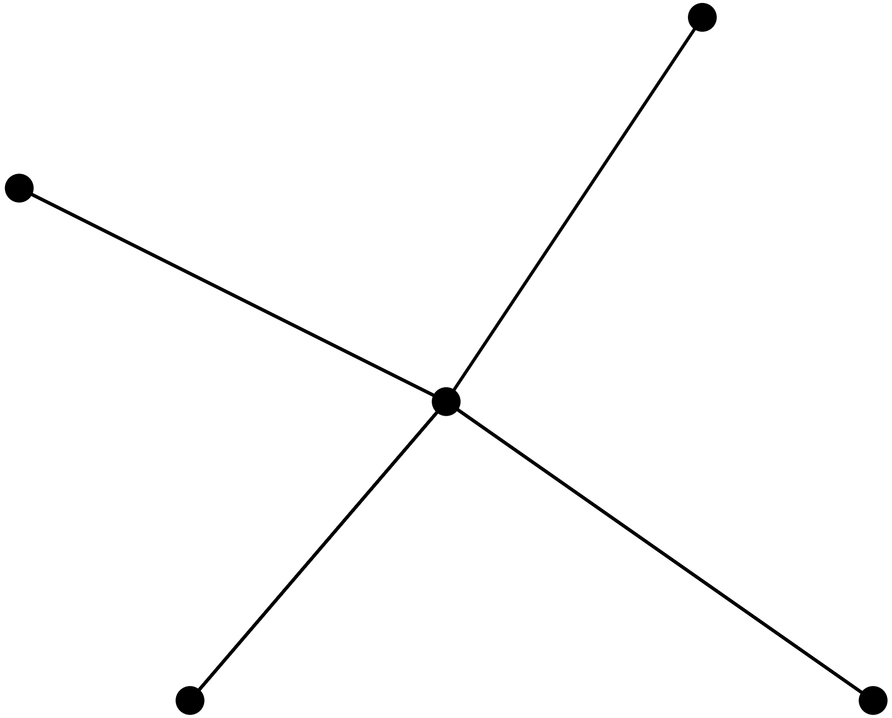


Independent sets in H^2

Lem. 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?

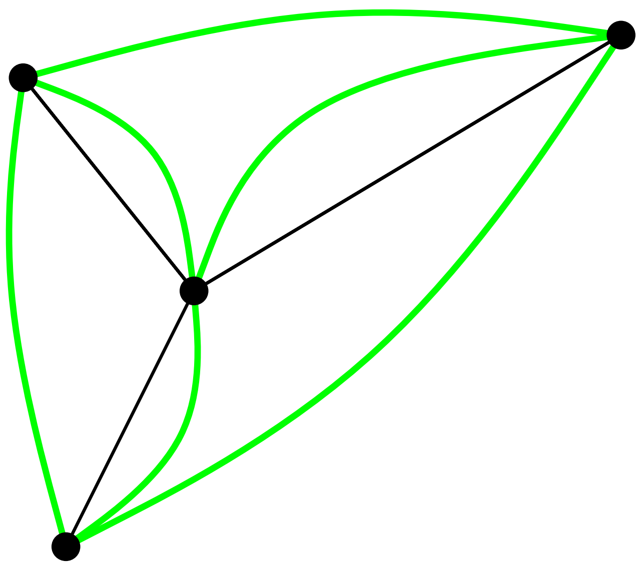


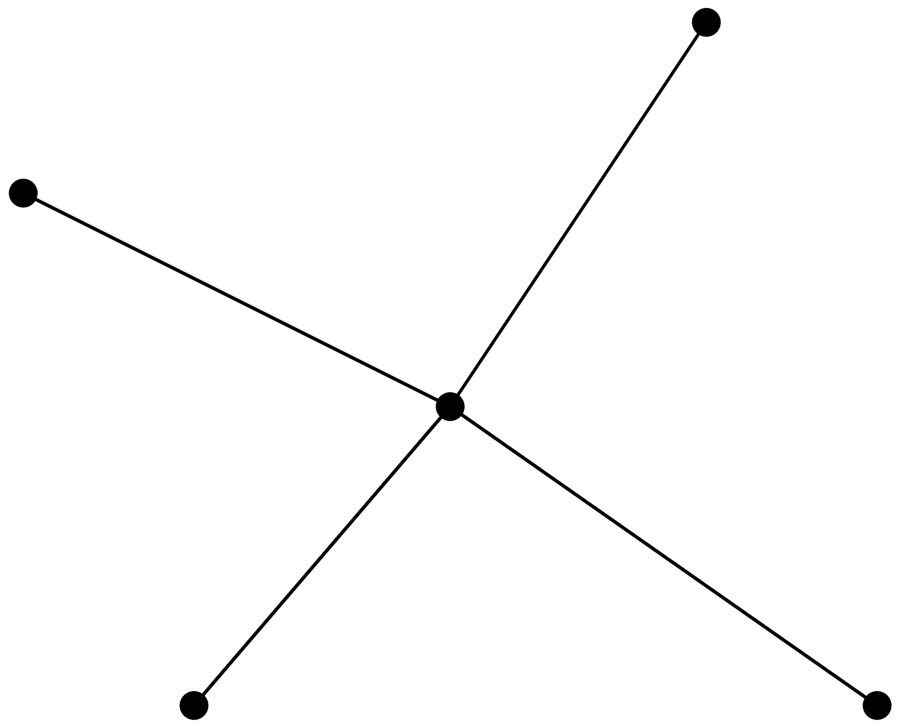


Independent sets in H^2

Lem. 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?

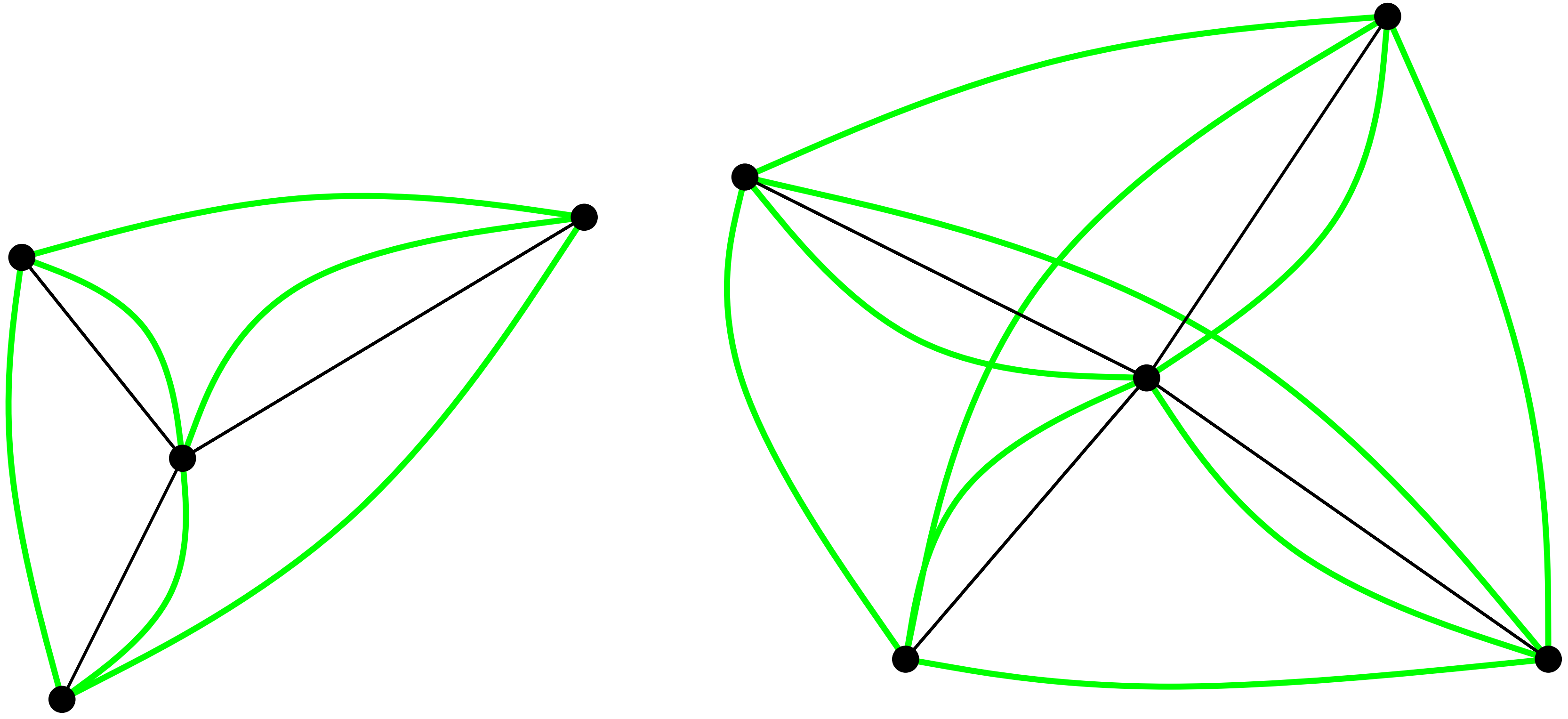




Independent sets in H^2

Lem. 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?



Factor-2 approx. for metric k-Center

Algorithm Metric-k-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

Lem. For j provided by the Algorithm, we have $c(e_j) \leq$ OPT.

Factor-2 approx. for metric k-Center

Algorithm Metric-k-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

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

Thm. The above algorithm is a factor-2 approximation algorithm for the metric k-Center problem.

Can we do better . . . ?

Can we do better . . . ?

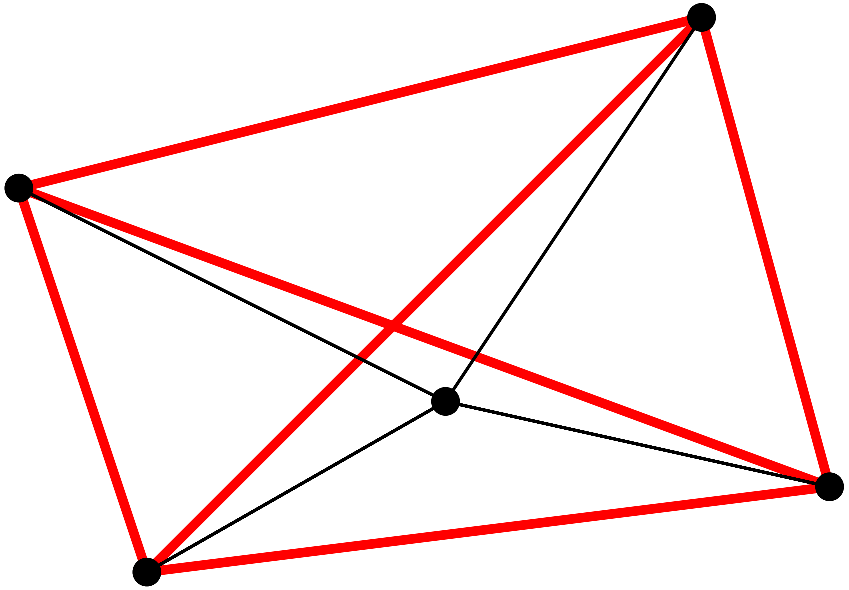
What about a tight example?

Can we do better . . . ?

What about a tight example?

black $\rightsquigarrow$ cost $=1$

red $\rightsquigarrow$ cost $=2$

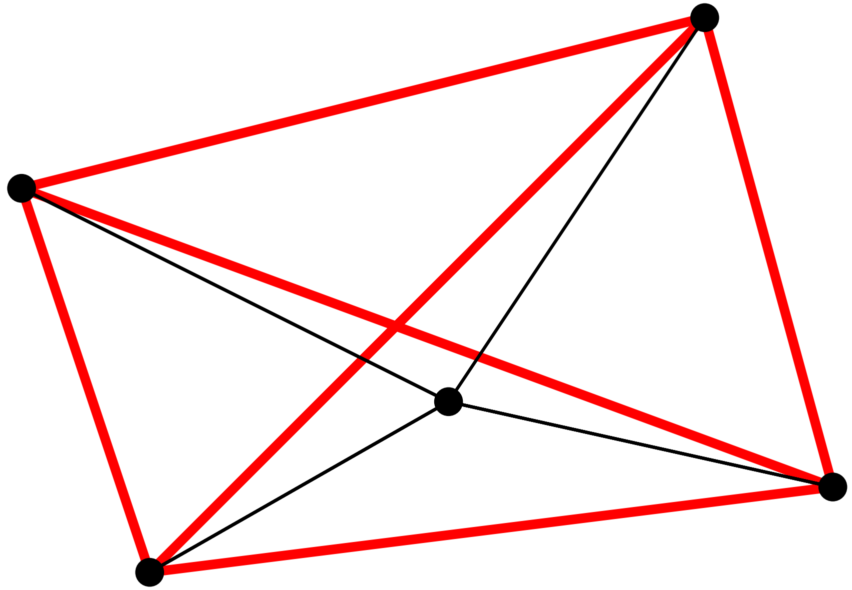


Can we do better . . . ?

What about a tight example?

black $\rightsquigarrow$ cost $=1$

red $\rightsquigarrow$ cost $=2$



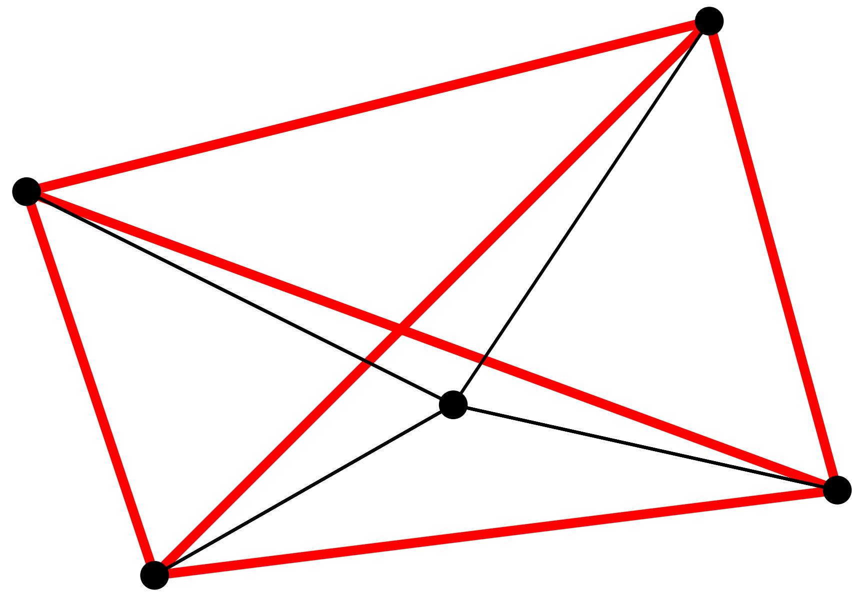
Thm. Assuming P $\neq$ NP, there is no factor-$(2-\epsilon)$ approximation algorithm for the metric k-Center problem, for any $\epsilon > 0$.

Can we do better . . . ?

What about a tight example?

black $\rightsquigarrow$ cost $=1$

red $\rightsquigarrow$ cost $=2$



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

Proof: Idea: reduce from dominating set to metric k-Center.

- If $\text{dom}(G) \leq k$, then opt k-center has cost ≤ 1.
- else ($\text{dom}(G) > k$), opt k-center has cost ≥ 2.

Metric k-CENTER problem

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 problem

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 problem

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

Metric ~~k~~-Center problem

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

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

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??

The weighted version

Algorithm metric-weighted-Center

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

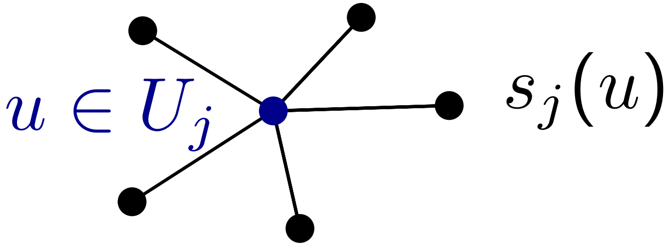
for $j = 1, \ldots, m$ **do**

  Construct G_j^2

  Find a maximal independent set U_j in G_j^2

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

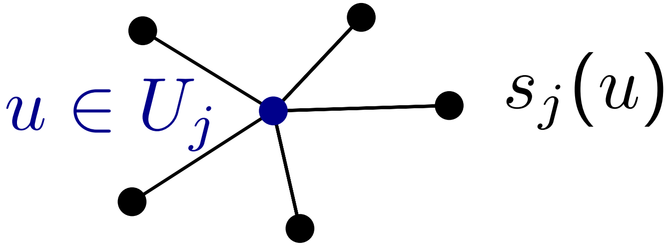
    return U_j



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

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 $N_{G_j}(u) \cup \{u\}$

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**

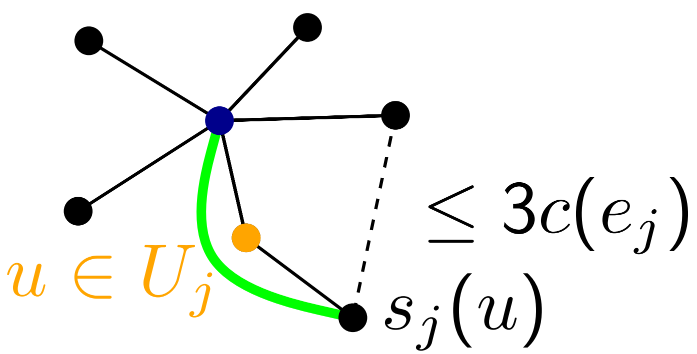
Construct G_j^2

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** $w(S_j) \leq W$

return ~~U_j~~ S_j



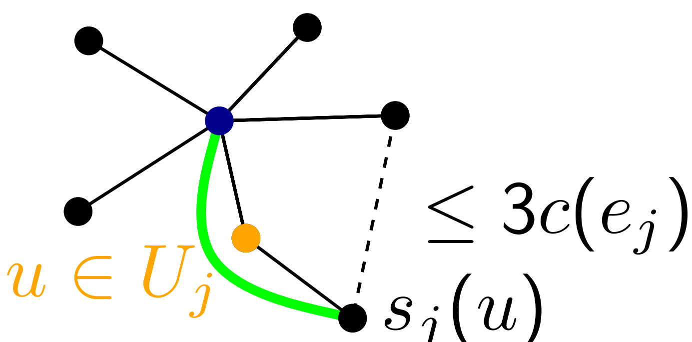
The weighted version

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** $w(S_j) \leq W$
  - **return** ~~U_j~~ S_j



Thm. The above is a factor-3 approximation algorithm for the metric weighted-CENTER problem.

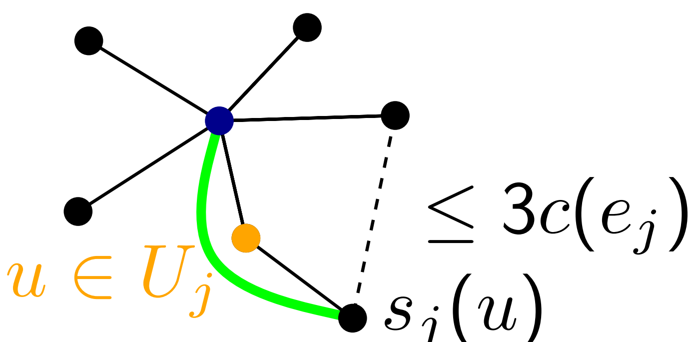
The weighted version

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** $w(S_j) \leq W$
  - **return** ~~U_j~~ S_j



Next Week:
Local Search
Min. Degree
Spanning Trees

Thm. The above is a factor-3 approximation algorithm for the metric weighted-Center problem.