Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Lehrstuhl für
INFORMATIK I
Algorithmen & Komplexität

Institut für Informatik

Exact Algorithms

Sommer Term 2020

Lecture 3. Minimum Dominating Set

Based on: [Exact Exponential Algorithms: §3.2]
Further discussions: [Parameterized Algorithms: §6.1]

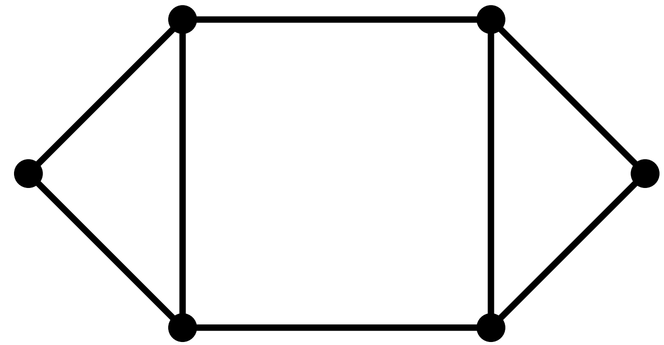
(slides by J. Spoerhase, Th. van Dijk, S. Chaplick, and A. Wolff)

Alexander Wolff

Lehrstuhl für Informatik I

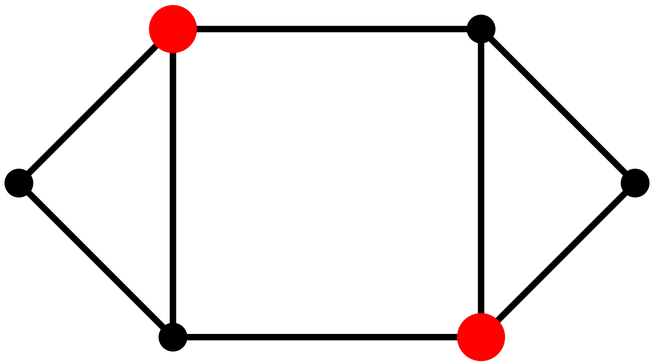
Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if



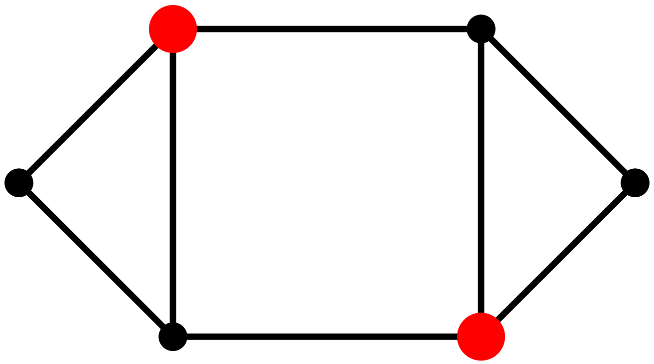
Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if



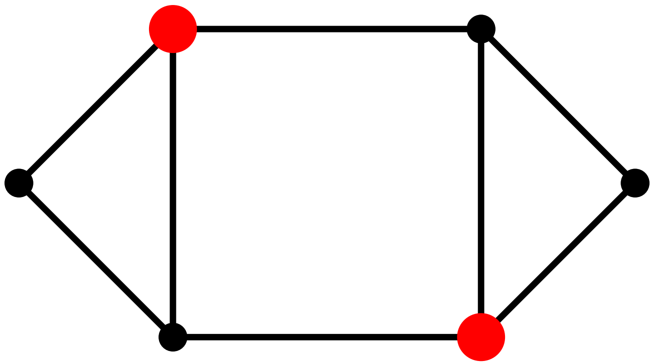
Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.



Dominating Sets

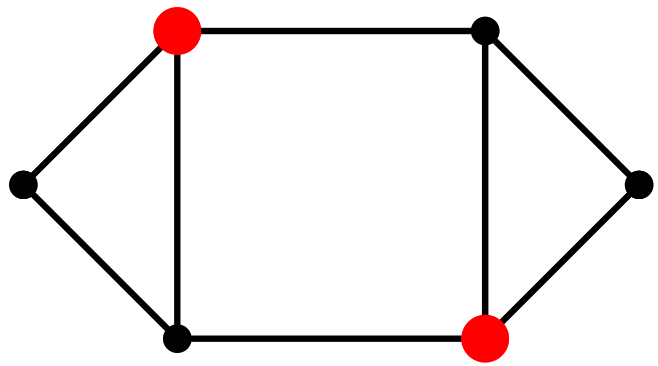
Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.



Example Application:

Dominating Sets

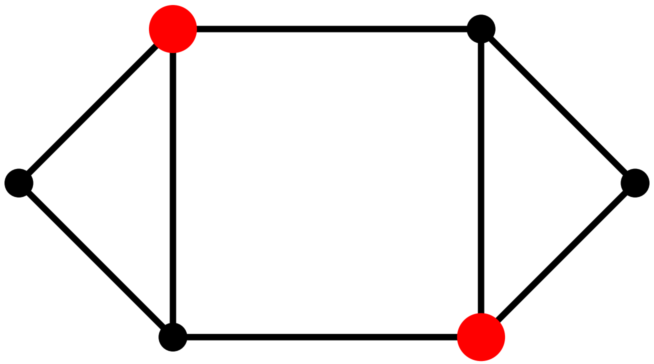
Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.



Example Application: Placement of cell towers.

Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.

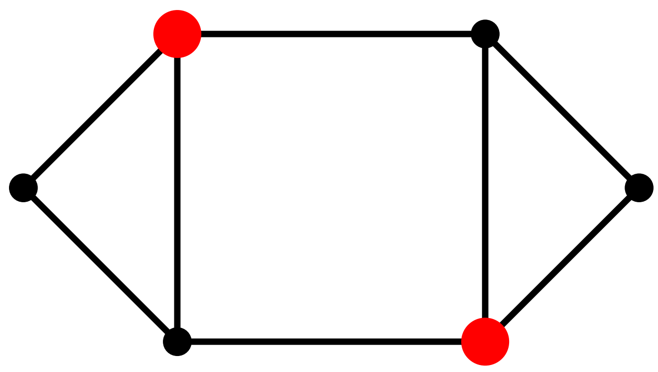


Example Application: Placement of cell towers.

Def. *Minimum Dominating Set*
Given:
Find:

Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.

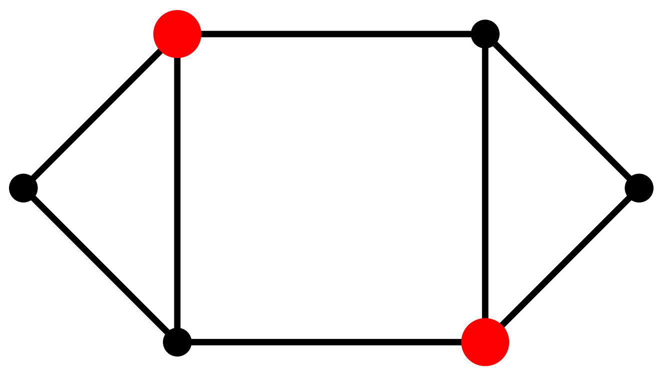


Example Application: Placement of cell towers.

Def. *Minimum Dominating Set*
Given: graph $G = (V, E)$,
Find:

Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.

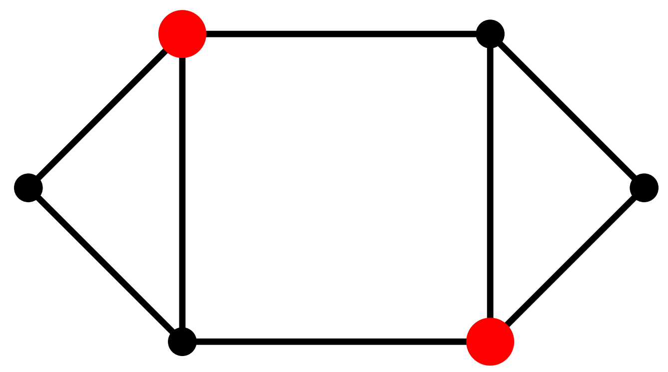


Example Application: Placement of cell towers.

Def. *Minimum Dominating Set*
Given: graph $G = (V, E)$,
Find: minimum-cardinality dominating set D of G

Dominating Sets

Def. For a graph $G = (V, E)$, a set $D \subseteq V$ *dominates* G if every vertex $u \in V \setminus D$ is adjacent to a vertex in D.



Example Application: Placement of cell towers.

Def. *Minimum Dominating Set*
Given: graph $G = (V, E)$,
Find: minimum-cardinality dominating set D of G

Domination number: $\gamma(G) := |D|$.

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

Algorithm recursiveIS(G)

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

Algorithm recursiveIS(G)
  if $V(G) = \emptyset$ **then**
    return $\emptyset$

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

Algorithm recursiveIS(G)
  if $V(G) = \emptyset$ **then**
    return $\emptyset$
  choose any $v \in V$
  return

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

Algorithm recursiveIS(G)
  if $V(G) = \emptyset$ **then**
    return $\emptyset$
  choose any $v \in V$
  return $\{v\} \cup$ recursiveIS(   )

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

```
Algorithm recursiveIS(G)
  if V(G) = ∅ then
    return ∅
  choose any v ∈ V
  return {v} ∪ recursiveIS(G − N[v])
```

Maximal Independent Sets

Def. An independent set I is *maximal* if no proper superset of it is an independent set.

Lemma. A maximal independent set can be found efficiently.

```
Algorithm recursiveIS(G)
  if V(G) = ∅ then
    return ∅
  choose any v ∈ V
  return {v} ∪ recursiveIS(G − N[v])
```

Obs. Every maximal independent set is a dominating set.

Algorithmic Approach for Min. Dom. Set

Brute Force:

Algorithmic Approach for Min. Dom. Set

Brute Force: (subset problem!)

Algorithmic Approach for Min. Dom. Set

Brute Force: $O^*(2^n)$ time (subset problem!)

Algorithmic Approach for Min. Dom. Set

Brute Force: $O^*(2^n)$ time (subset problem!)

An idea for a smarter algorithm:

- Find a maximal independent set I.
- If I is "small",
- If I is "big",

Algorithmic Approach for Min. Dom. Set

Brute Force: $O^*(2^n)$ time (subset problem!)

An idea for a smarter algorithm:

- Find a maximal independent set I.
- If I is "small", look at each $D \subseteq V$, $|D| < |I|$.
- If I is "big",

Algorithmic Approach for Min. Dom. Set

Brute Force: $O^*(2^n)$ time (subset problem!)

An idea for a smarter algorithm:

- Find a maximal independent set I.
- If I is "small", look at each $D \subseteq V$, $|D| < |I|$.
- If I is "big", use a *dynamic program* to ensure that vertices in I do not dominate each other.

Algorithmic Approach for Min. Dom. Set

Brute Force: $O^*(2^n)$ time (subset problem!)

An idea for a smarter algorithm:

- Find a maximal independent set I.
- If I is "small", look at each $D \subseteq V$, $|D| < |I|$.
- If I is "big", use a *dynamic program* to ensure that vertices in I do not dominate each other.

Lemma $\star$. Given a maximal independent set I of G, a minimum dominating set of G can be found in $O^*(2^{n-|I|})$ time.

Proof of Lemma $\star$

- Instead of *all* 2^n subsets of V, we consider their $2^{n-|I|}$ *projections* on $J = V \setminus I$

Proof of Lemma $\star$

- Instead of *all* 2^n subsets of V, we consider their $2^{n-|I|}$ *projections* on $J = V \setminus I$

- We test each subset J' of J and extend it to the *smallest* dominating set $D_{J'}$ of G such that $J \cap D_{J'} = J'$.

Proof of Lemma $\star$

- Instead of *all* 2^n subsets of V, we consider their $2^{n-|I|}$ *projections* on $J = V \setminus I$
- We test each subset J' of J and extend it to the *smallest* dominating set $D_{J'}$ of G such that $J \cap D_{J'} = J'$.

  $\Rightarrow \gamma(G) =$

Proof of Lemma $\star$

- Instead of *all* 2^n subsets of V, we consider their $2^{n-|I|}$ *projections* on $J = V \setminus I$

- We test each subset J' of J and extend it to the *smallest* dominating set $D_{J'}$ of G such that $J \cap D_{J'} = J'$.

  $\Rightarrow \gamma(G) = \min_{J' \subseteq J} |D_{J'}|$.

Proof of Lemma $\star$

- Instead of *all* 2^n subsets of V, we consider their $2^{n-|I|}$ *projections* on $J = V \setminus I$

- We test each subset J' of J and extend it to the *smallest* dominating set $D_{J'}$ of G such that $J \cap D_{J'} = J'$.

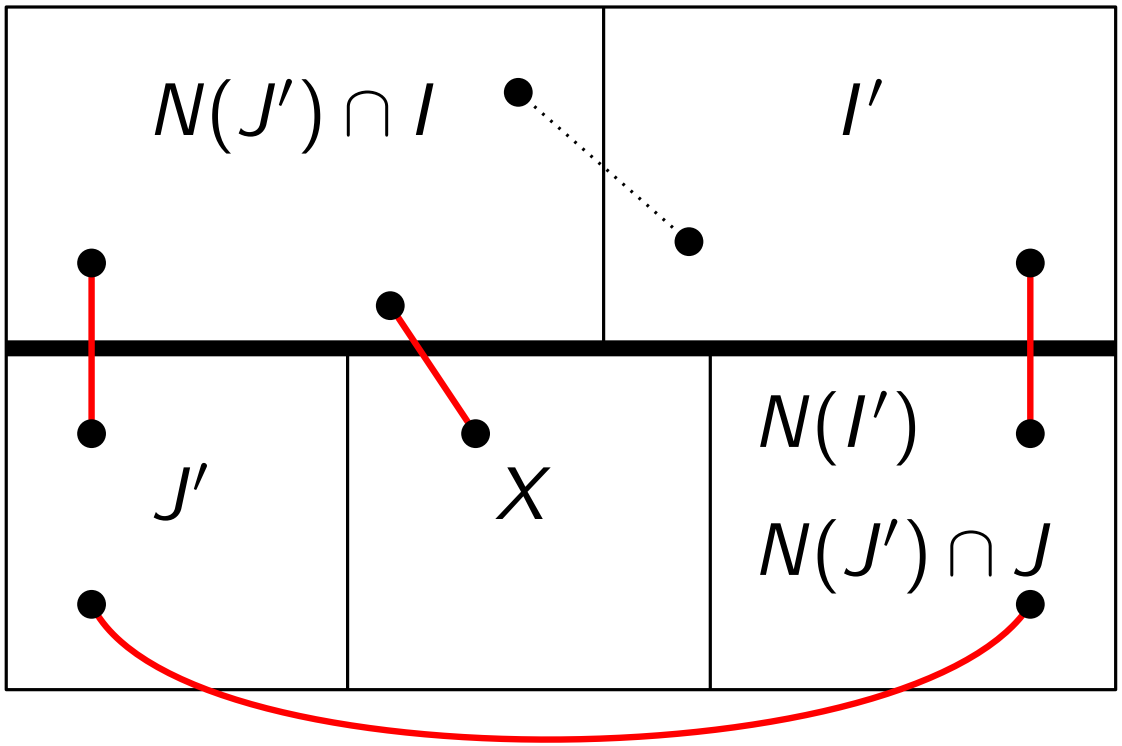
  $\Rightarrow \gamma(G) = \min_{J' \subseteq J} |D_{J'}|$.

- How can we find a smallest $D_{J'}$ for a given $J' \subseteq J$?

Proof (Lemma $\star$)

independent set: I

J

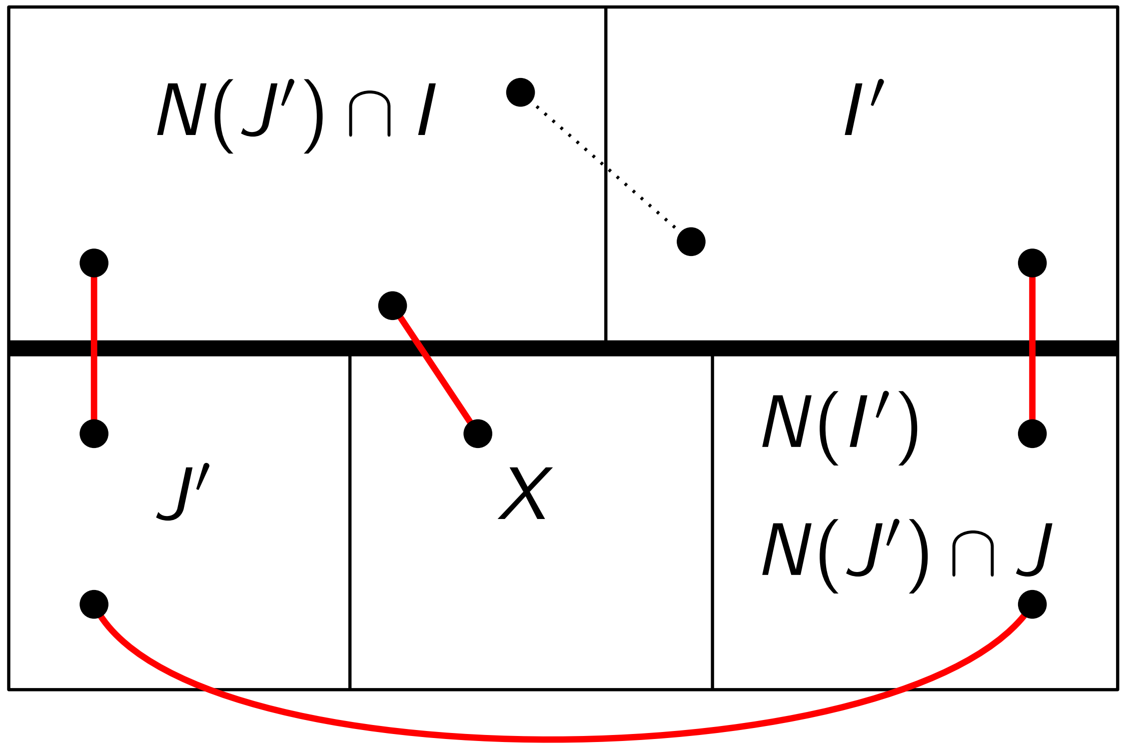


Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I

- $I' = I \setminus N(J')$ must be completely contained in $D_{J'}$

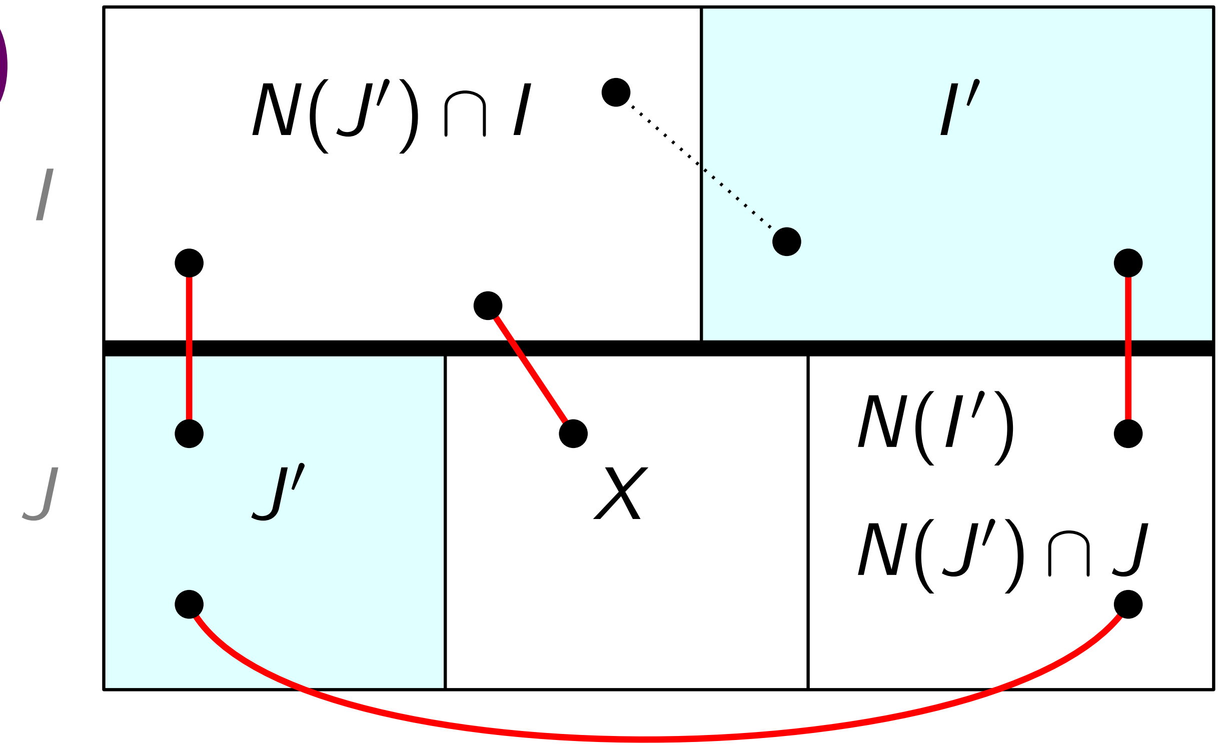


Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I

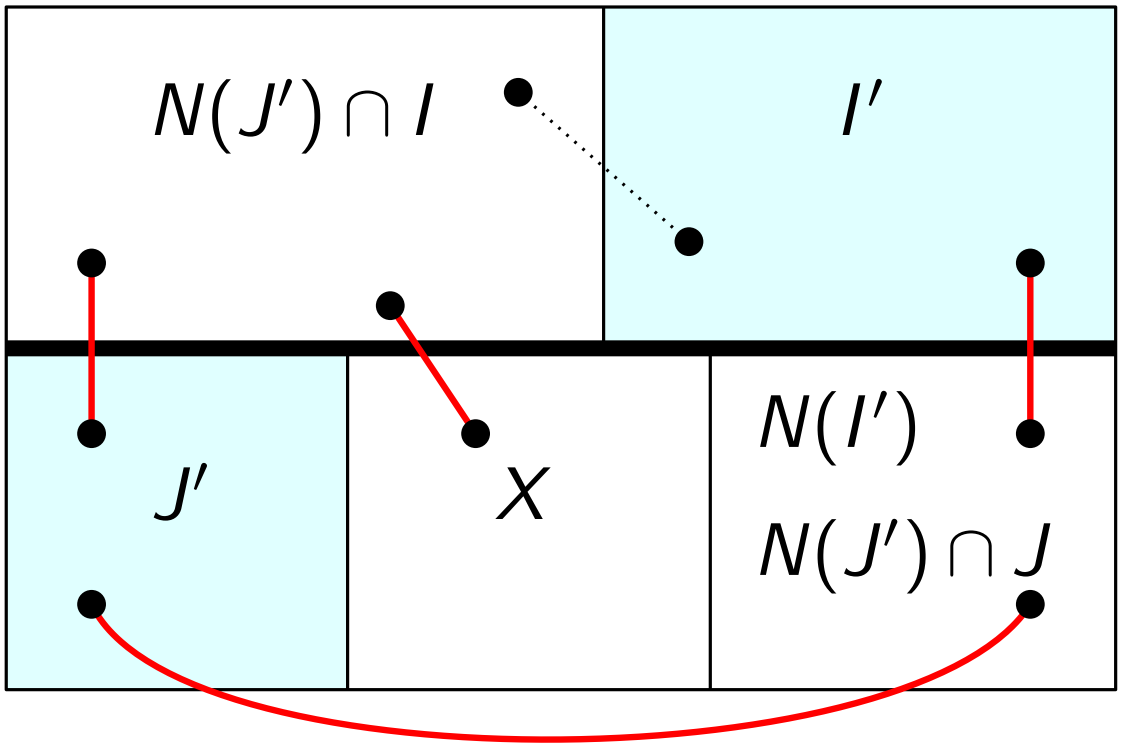
- $I' = I \setminus N(J')$ must be completely contained in $D_{J'}$ (since $D_{J'} \cap J = J'$!)



Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I

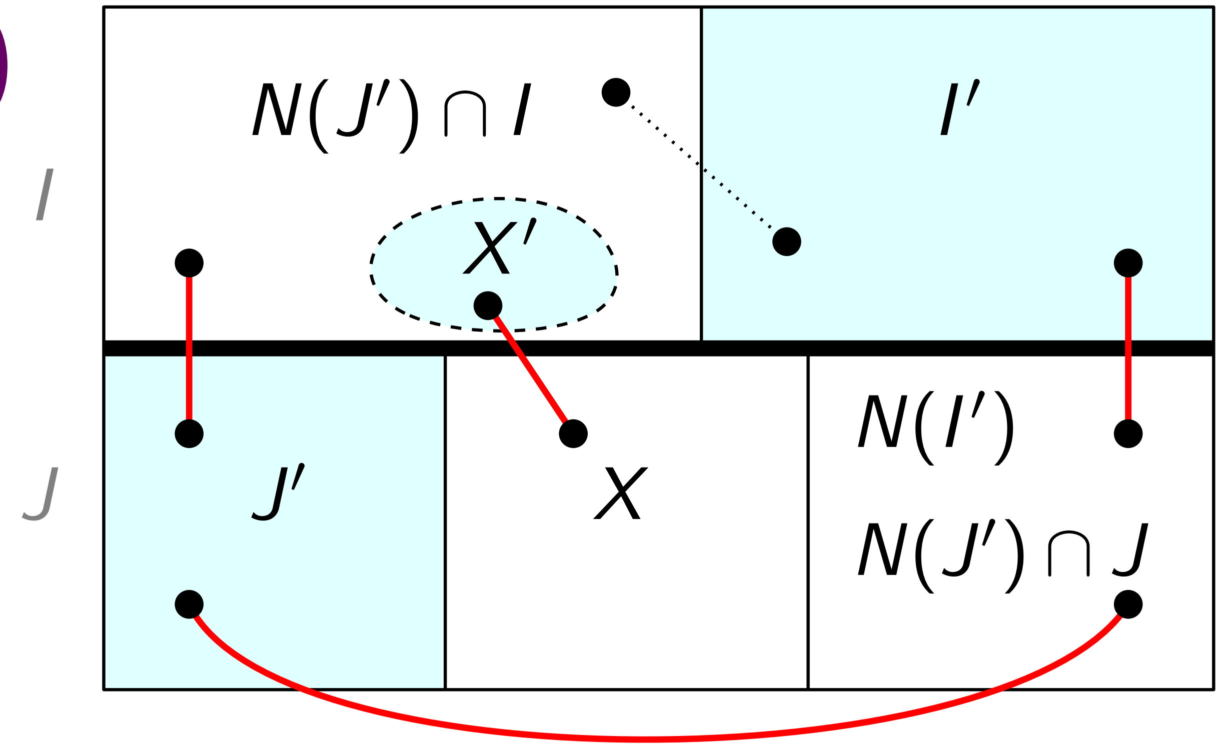


- $I' = I \setminus N(J')$ must be completely contained in $D_{J'}$ (since $D_{J'} \cap J = J'$!)
- The vertices not dominated by I' and J' are precisely $X := J \setminus (N[J'] \cup N(I'))$.

Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I



- $I' = I \setminus N(J')$ must be completely contained in $D_{J'}$ (since $D_{J'} \cap J = J'$!)

- The vertices not dominated by I' and J' are precisely $X := J \setminus (N[J'] \cup N(I'))$.

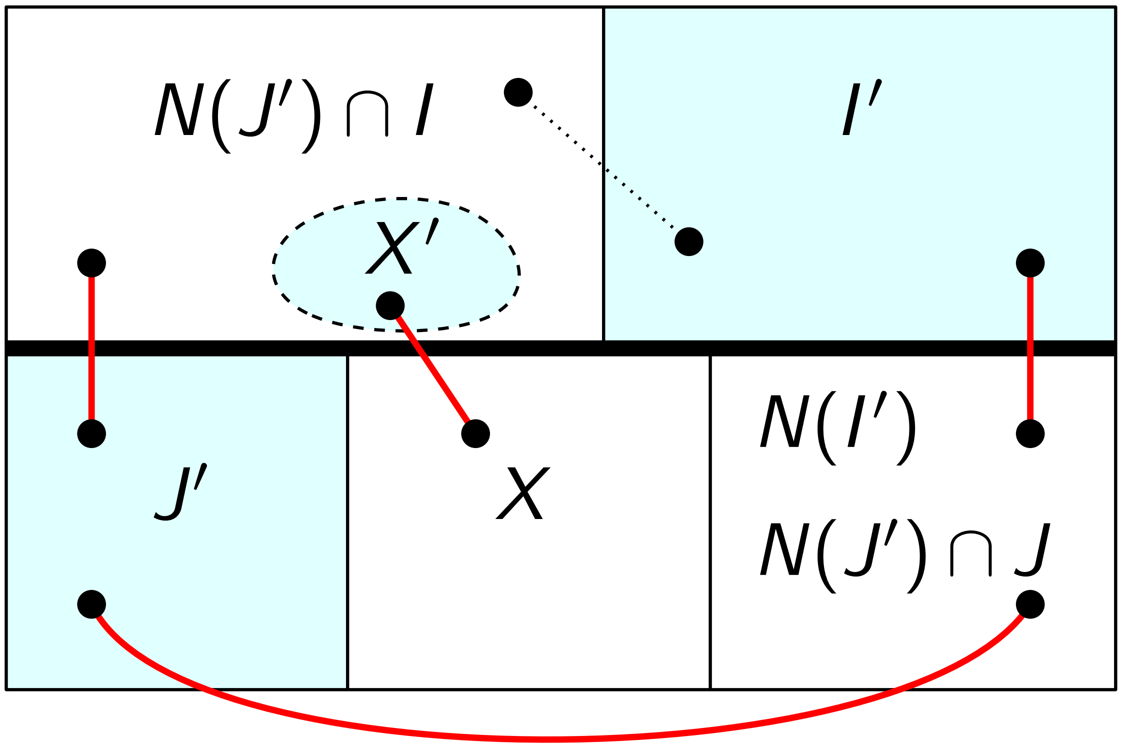
- Find the smallest set $X' \subseteq N(J') \cap I$ that dominates X.

Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I

J

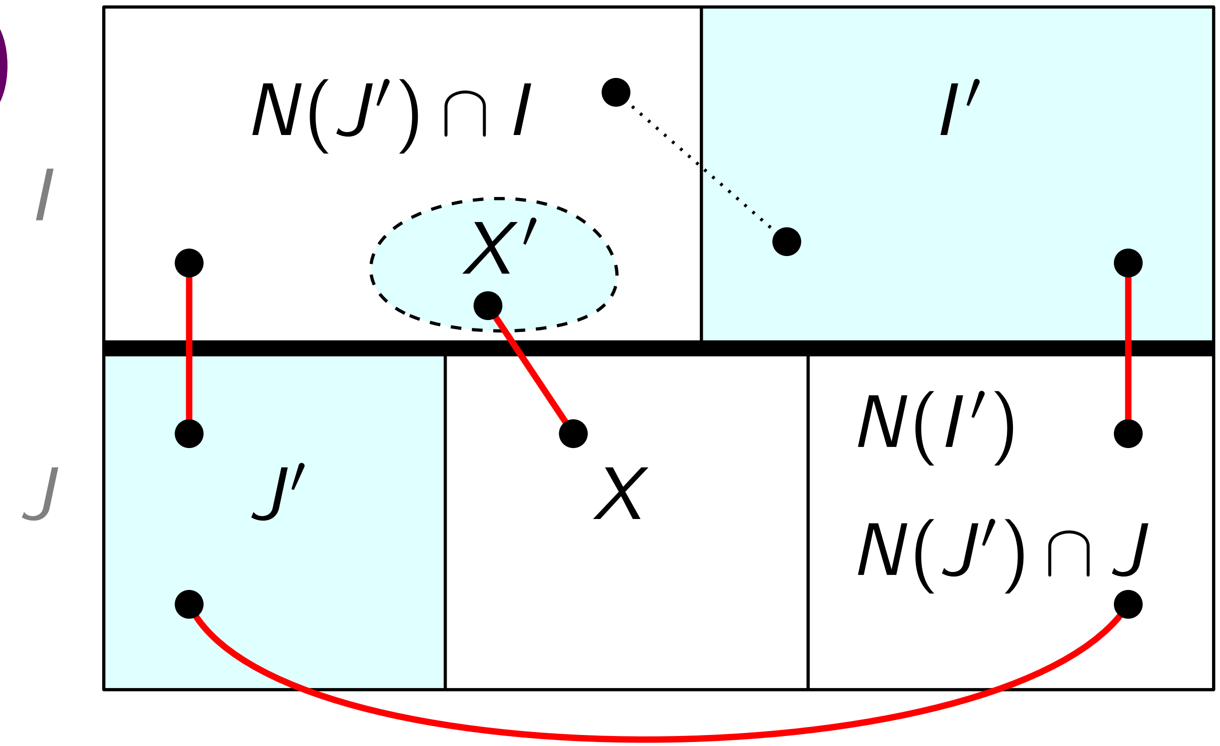


- $I' = I \setminus N(J')$ must be completely contained in $D_{J'}$ (since $D_{J'} \cap J = J'$!)
- The vertices not dominated by I' and J' are precisely $X := J \setminus (N[J'] \cup N(I'))$.
- Find the smallest set $X' \subseteq N(J') \cap I$ that dominates X.
- $\Rightarrow D_{J'} = J' \cup I' \cup X'$ dominates G.

Def. $N(S) = \bigcup_{v \in S} N(v)$ and $N[S] = N(S) \cup S$ for every $S \subseteq V$.

Proof (Lemma $\star$)

independent set: I



- Naive idea: find X' for each X separately.

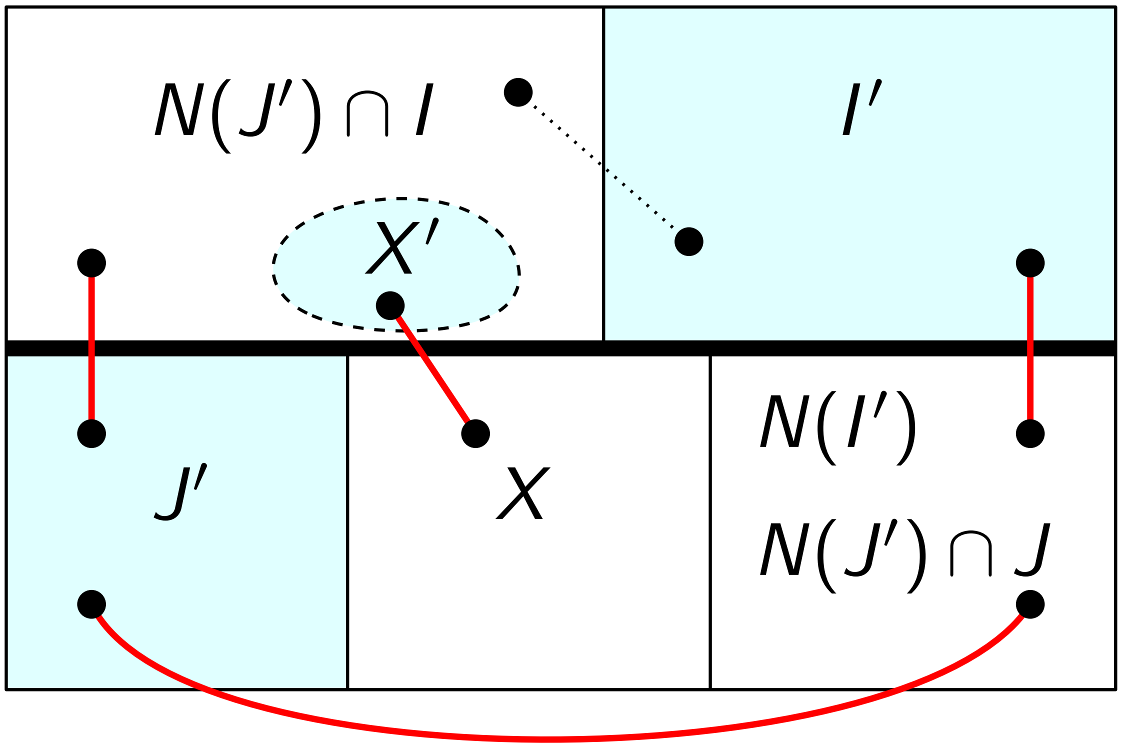
Proof (Lemma $\star$)

independent set: I

- Naive idea:
  find X' for each
  X separately.

  $\Rightarrow$ runtime $O^*(3^{|J|})$

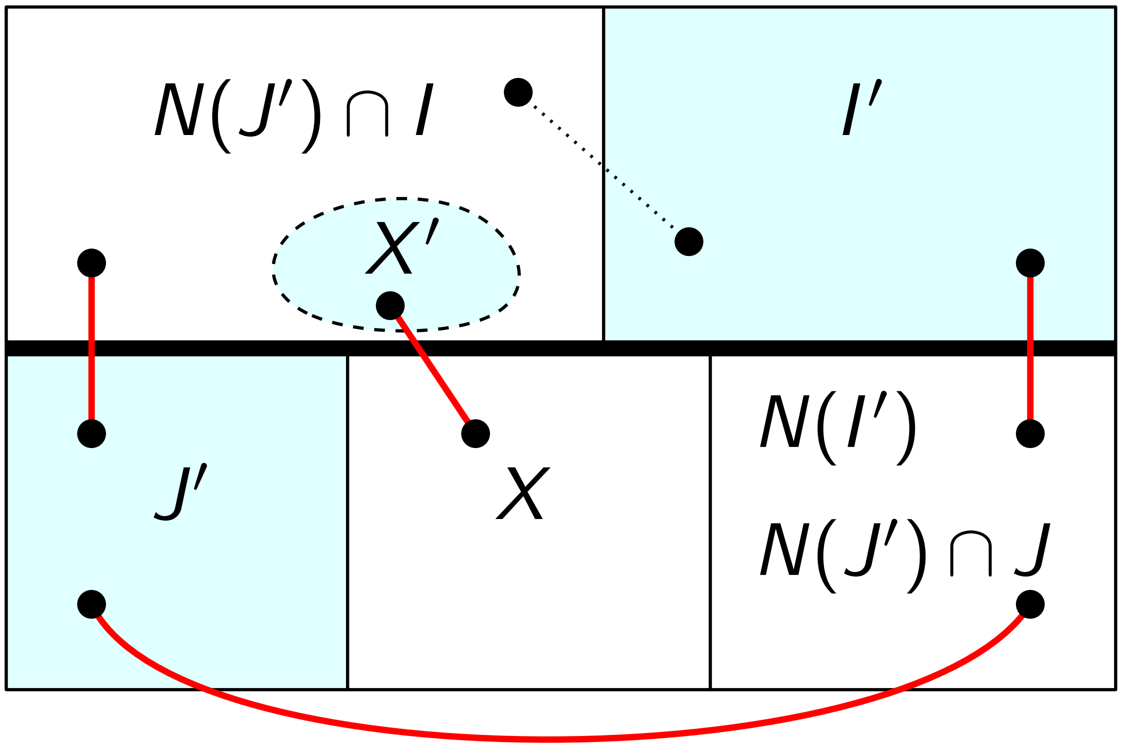
J



Proof (Lemma $\star$)

independent set: I

J

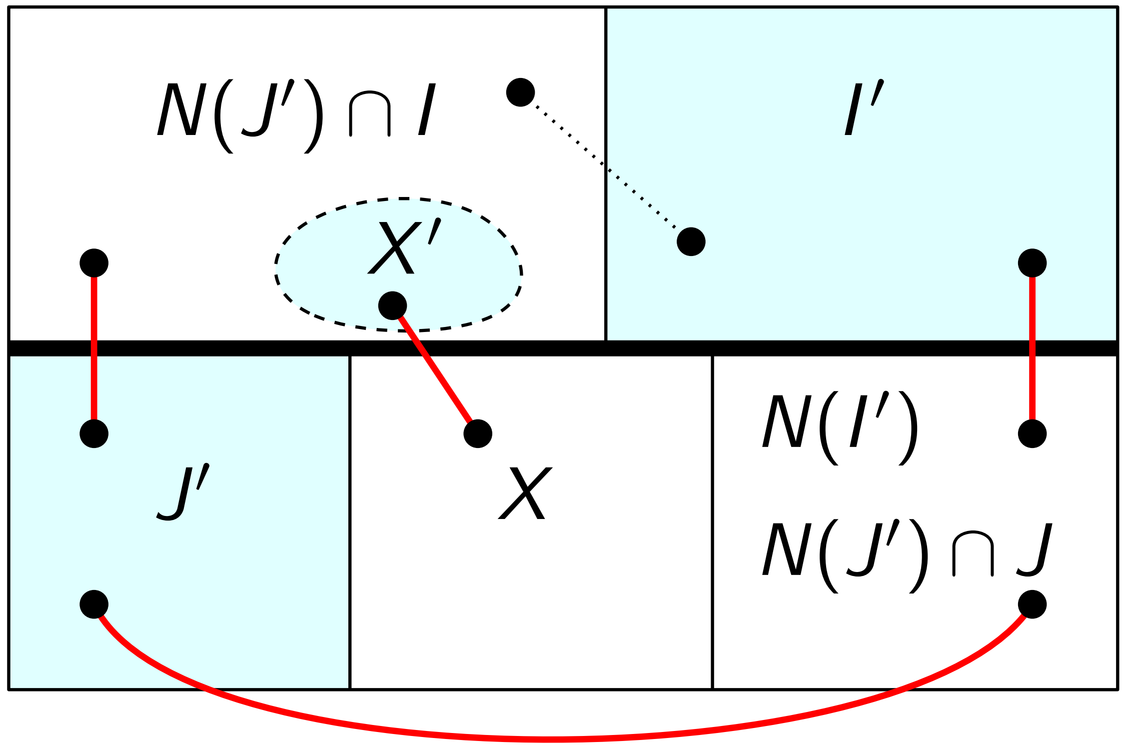


- Naive idea:
  find X' for each X separately.
  $\Rightarrow$ runtime $O^*(3^{|J|})$
- Better idea:
  For every subset $X \subseteq J$,
  we compute a minimum subset of I that dominates X.

Proof (Lemma $\star$)

independent set: I

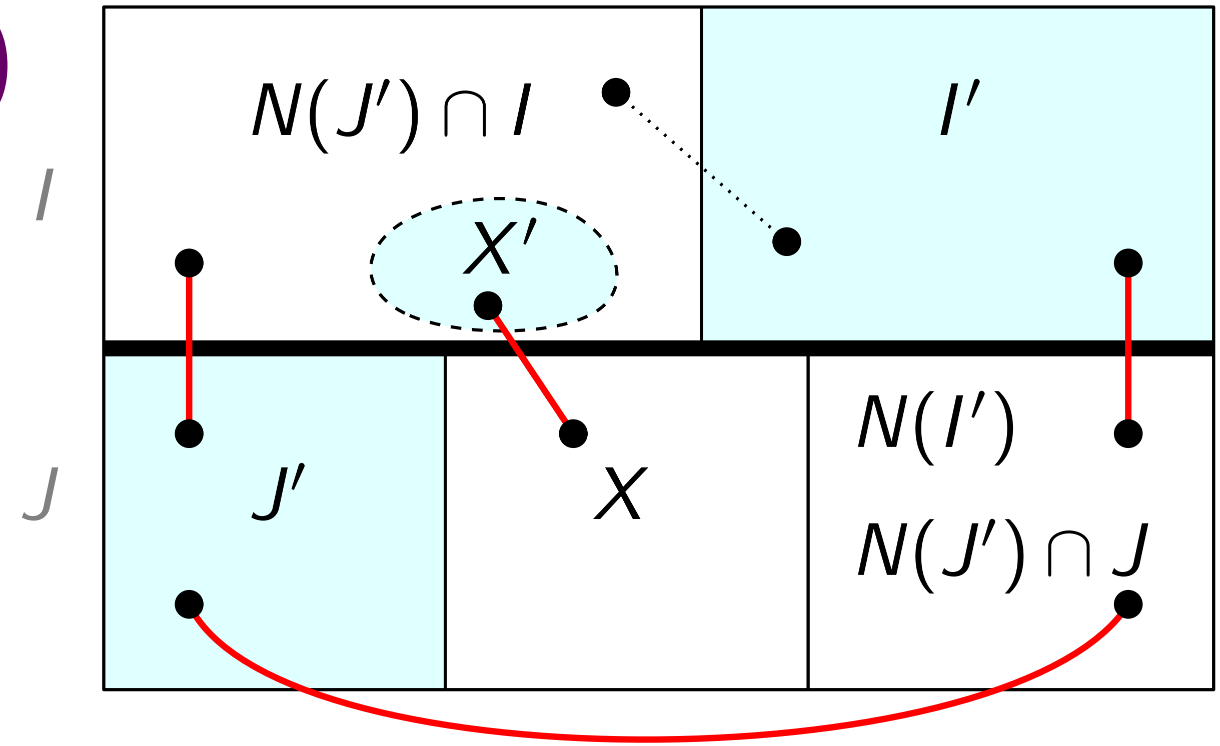
J



- Naive idea:
  find X' for each
  X separately.
  $\Rightarrow$ runtime $O^*(3^{|J|})$
- Better idea:
  For every subset $X \subseteq J$,
  we compute a minimum subset of I that dominates X.
- Let $I := \{v_1, \ldots, v_k\}$, $X \subseteq J$ and define:
  $T[X, \ell] :=$ a smallest subset of $\{v_1, \ldots, v_\ell\}$ dominating X.

Proof (Lemma $\star$)

independent set: I

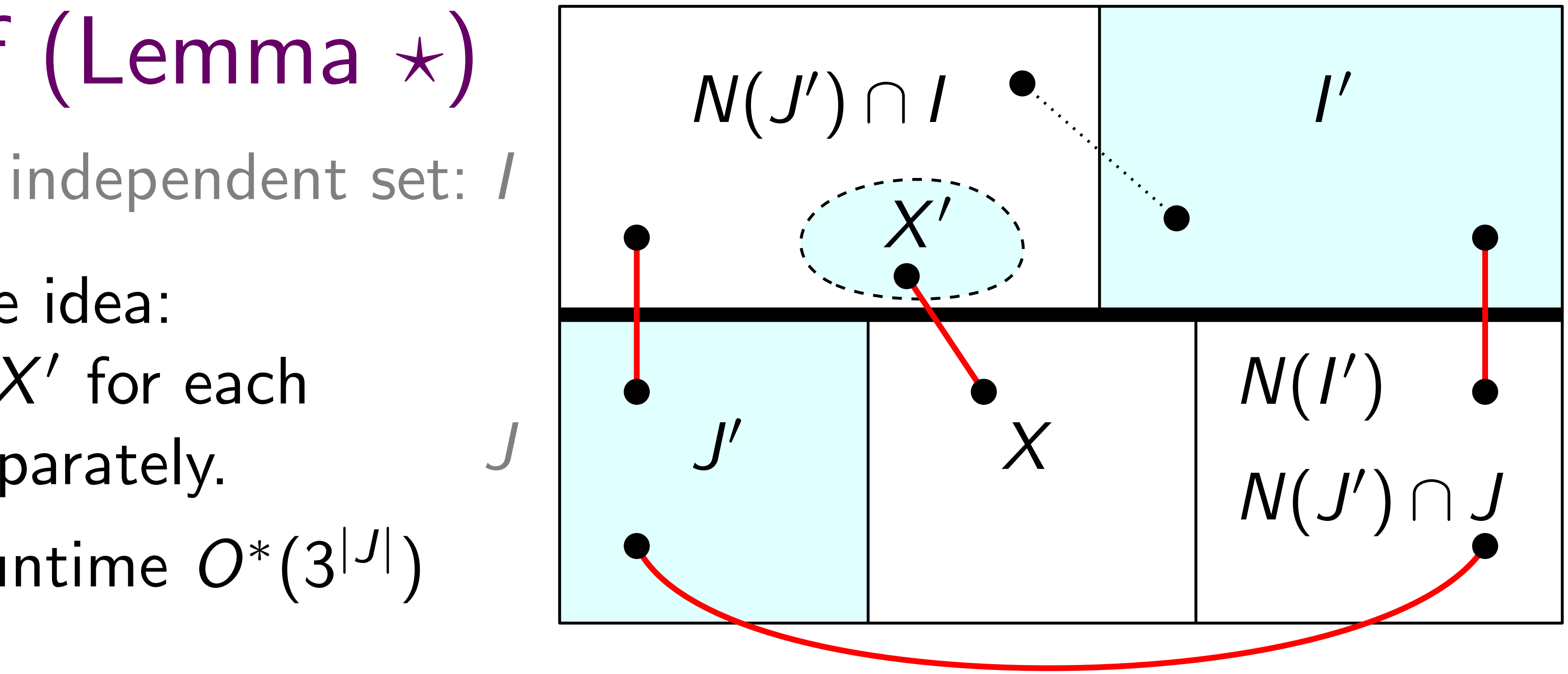


- Naive idea:
  find X' for each
  X separately.
  $\Rightarrow$ runtime $O^*(3^{|J|})$

- Better idea:
  For every subset $X \subseteq J$,
  we compute a minimum subset of I that dominates X.

- Let $I := \{v_1, \ldots, v_k\}$, $X \subseteq J$ and define:
  $T[X, \ell] :=$ a smallest subset of $\{v_1, \ldots, v_\ell\}$ dominating X.
  $\Rightarrow X' =$

Proof (Lemma $\star$)



- Naive idea:
  find X' for each
  X separately.
  $\Rightarrow$ runtime $O^*(3^{|J|})$
- Better idea:
  For every subset $X \subseteq J$,
  we compute a minimum subset of I that dominates X.
- Let $I := \{v_1, \ldots, v_k\}$, $X \subseteq J$ and define:
  $T[X, \ell] :=$ a smallest subset of $\{v_1, \ldots, v_\ell\}$ dominating X.
  $\Rightarrow X' = T[X, k]$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

- runtime $O^*(\quad)$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of} \begin{cases} T[X,\ell-1] \quad \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] \ \text{ if def'd.} \end{cases}$$

- runtime $O^*(2^{|J|})$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \le \ell \le k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] \quad \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] \;\; \text{if def'd.} \end{cases}$$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup X'$ that dominates G $\Rightarrow$ runtime:

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup \boxed{X'}$ that dominates $G \Rightarrow$ runtime:

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

as $T[X,k]$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup \boxed{X'}$ that dominates $G \Rightarrow$ runtime:

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of} \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

as $T[X,k]$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup \boxed{X'}$ that dominates G $\Rightarrow$ runtime: $O^*(2^{|J|})$

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

as $T[X,k]$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup \boxed{X'}$ that dominates G $\Rightarrow$ runtime: $O^*(2^{|J|})$

$\Rightarrow$ total runtime of the algorithm:

Proof (Lemma $\star$)

For each $X \subseteq J$: *Dynamic Program*

$$T[X,0] = \begin{cases} \emptyset & \text{if } X = \emptyset \\ \text{undef.} & \text{if } X \neq \emptyset \end{cases}$$

For $1 \leq \ell \leq k$:

$$T[X,\ell] = \text{smaller of } \begin{cases} T[X,\ell-1] & \text{and} \\ \{v_\ell\} \cup T[X \setminus N(v_\ell), \ell-1] & \text{if def'd.} \end{cases}$$

as $T[X,k]$

- runtime $O^*(2^{|J|})$
- For each of the $2^{|J|}$ sets $J' \subseteq J$, determine a smallest set $D_{J'} = J' \cup I' \cup \boxed{X'}$ that dominates G $\Rightarrow$ runtime: $O^*(2^{|J|})$

$\Rightarrow$ total runtime of the algorithm: $O^*(2^{|J|}) = O^*(2^{n-|I|})$ $\square$

Main Result

Thm.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime??

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime??

2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime?? TO DO: Analyse this more carefully!

2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime?? TO DO: Analyse this more carefully!

2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime?? TO DO: Analyse this more carefully!

2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO: Determine the value $\alpha^\star$ for α, to balance 1. and 2.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:
$\Rightarrow \gamma(G) \leq \alpha n$
$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime?? TO DO: Analyse this more carefully!

2. If $|I| > \alpha n$:
Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO: Determine the value $\alpha^\star$ for α, to balance 1. and 2.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\beta^n)$ time, for some $\beta < 2$.

Proof. Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:
$\Rightarrow \gamma(G) \leq \alpha n$
$\Rightarrow$ Try all αn-subsets of the given n vertices.
Runtime?? TO DO: Analyse this more carefully!

2. If $|I| > \alpha n$: (Note: If $\alpha \geq \frac{1}{2}$, then definitely use 2.)
Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO: Determine the value $\alpha^\star$ for α, to balance 1. and 2.

Helper Lemma

Lemma. For $\alpha \in (0, \frac{1}{2}]$, we have

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\Big(2^{h(\alpha)n}\Big),$$

where $h(\alpha) = -\alpha \log_2 \alpha - (1 - \alpha) \log_2(1 - \alpha)$.

Helper Lemma

Lemma. For $\alpha \in (0, \frac{1}{2}]$, we have

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right),$$

where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha) \log_2(1-\alpha)$.

(That's the binary entropy function.)

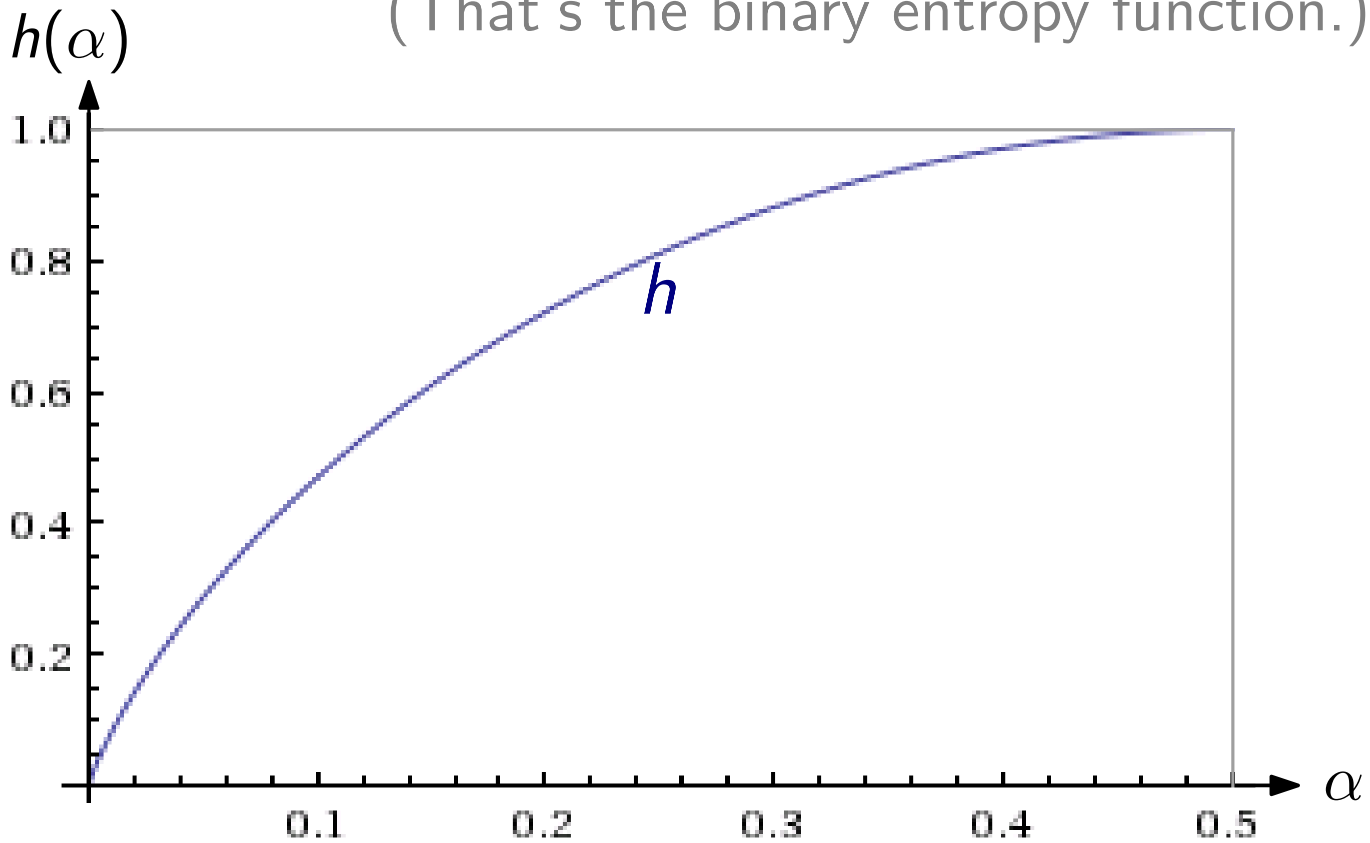
Helper Lemma

Lemma. For $\alpha \in (0, \frac{1}{2}]$, we have

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha) n}\right),$$

where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

(That's the binary entropy function.)



Helper Lemma

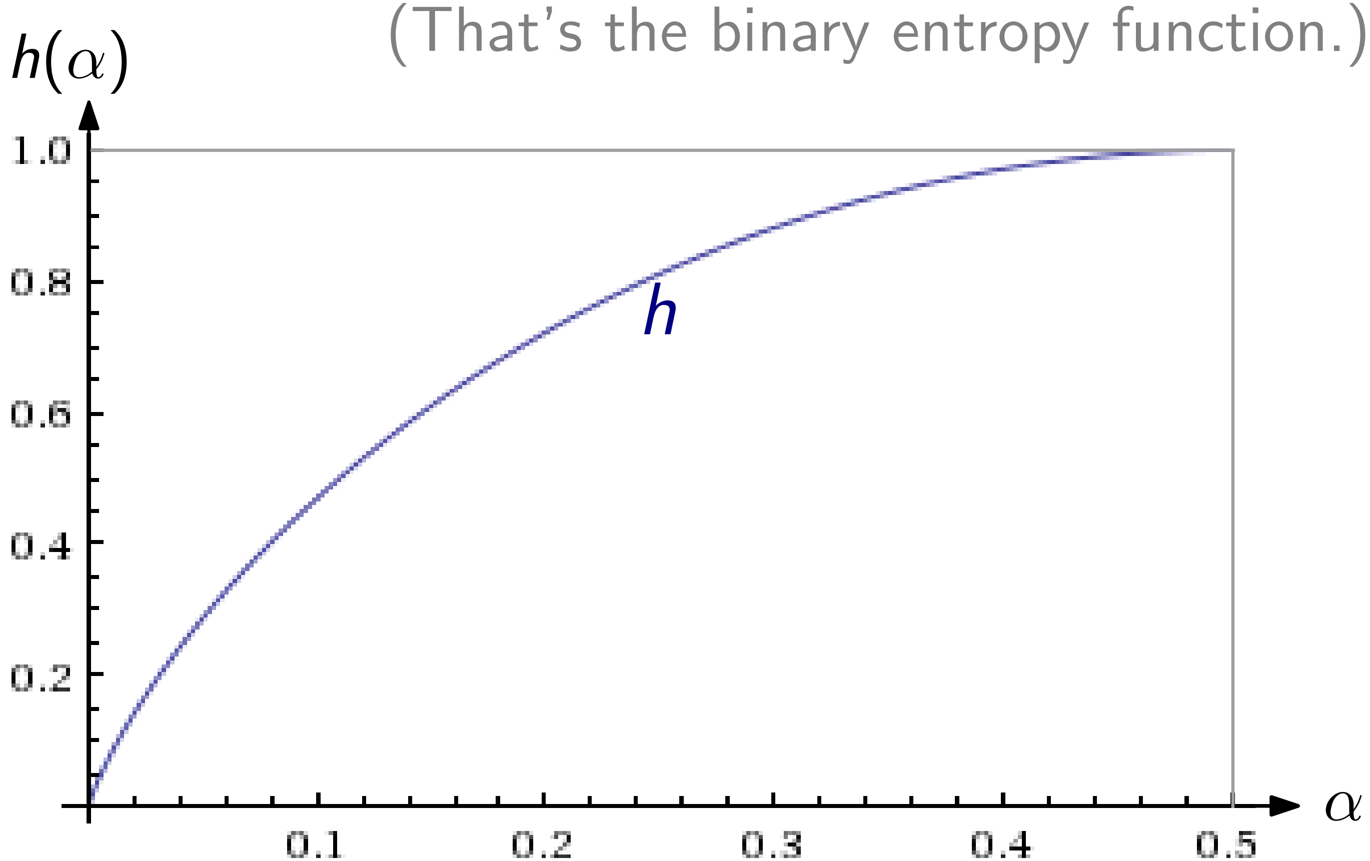
(Proof at the end!)

Lemma. For $\alpha \in (0, \frac{1}{2}]$, we have

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right),$$

where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

(That's the binary entropy function.)



Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\qquad ^n)$ time.

Proof.

Compute a maximal independent set I.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\quad{}^n)$ time.

Proof.

Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\quad\quad^n)$ time.

Proof.

Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\quad\quad^n)$ time.

Proof.

Compute a maximal independent set I.

1. If $|I| \le \alpha n$:

$\Rightarrow \gamma(G) \le \alpha n$

$\Rightarrow$ In $O^*(2^{h(\alpha)n})$ time, locate a minimum dominating set of cardinality $\le \alpha n$ (by brute force & helper lemma)

2. If $|I| > \alpha n$:

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(\quad\quad^{n})$ time.

Proof.

Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ In $O^*(2^{h(\alpha)n})$ time, locate a minimum dominating set of cardinality $\leq \alpha n$ (by brute force & helper lemma)

2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO: Determine the value $\alpha^\star$ for α, to balance 1. and 2.

Main Result

Thm. A minimum dominating set of a given graph can be found in $O(1.7088^n)$ time.

Proof.

Compute a maximal independent set I.

1. If $|I| \leq \alpha n$:

$\Rightarrow \gamma(G) \leq \alpha n$

$\Rightarrow$ In $O^*(2^{h(\alpha)n})$ time, locate a minimum dominating set of cardinality $\leq \alpha n$ (by brute force & helper lemma)

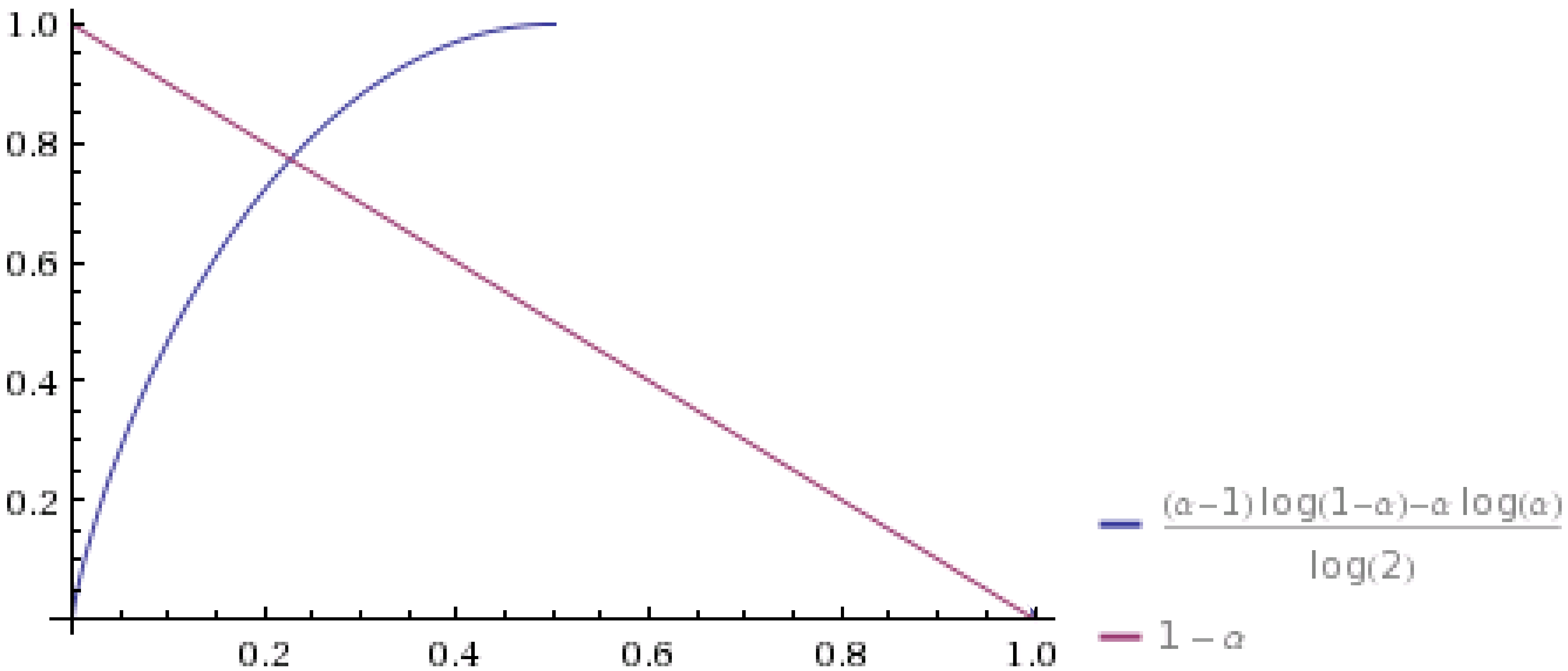
2. If $|I| > \alpha n$:

Apply Lemma $\star$ to obtain a minimum dominating set in $O^*(2^{(1-\alpha)\cdot n})$ time.

TO DO: Determine the value $\alpha^\star$ for α, to balance 1. and 2.

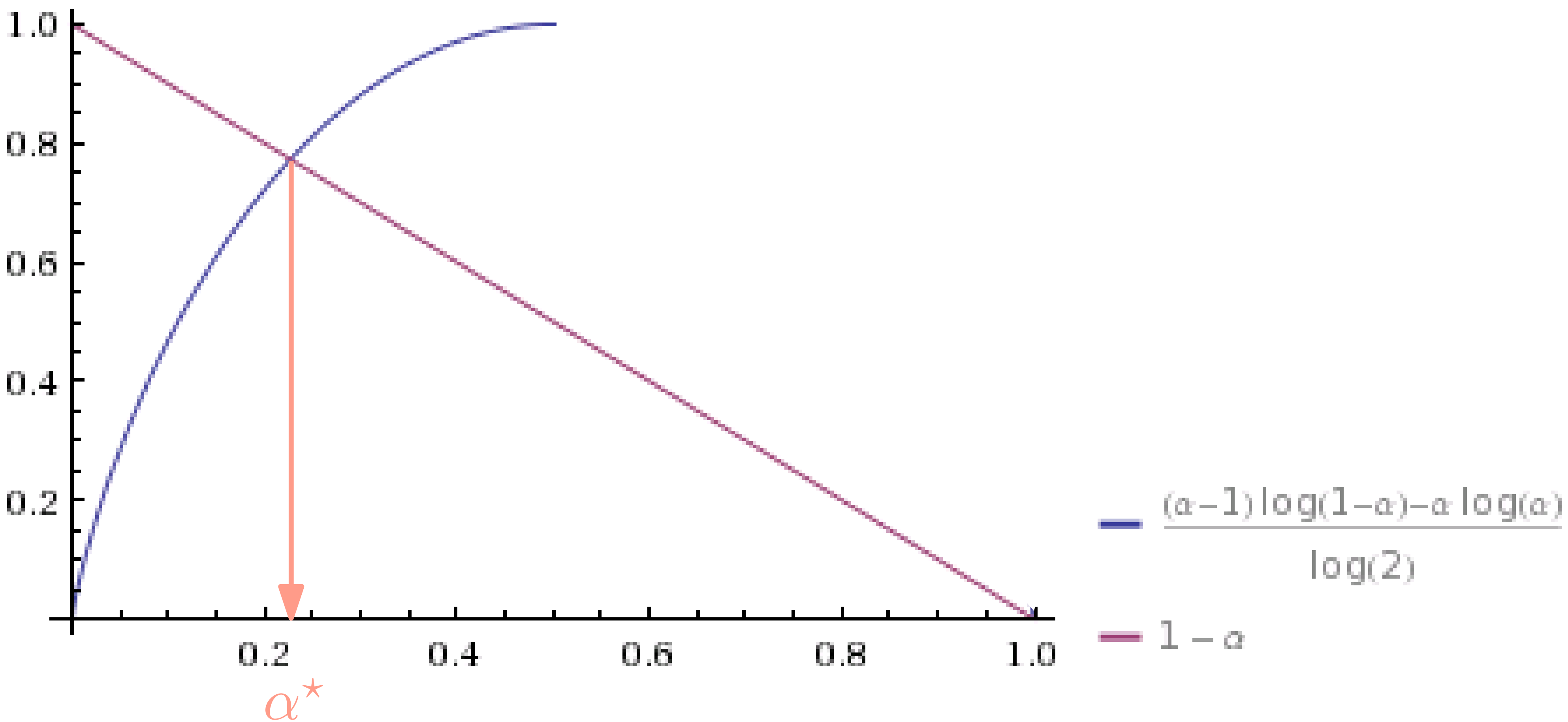
Finding $\alpha^\star$ and the Base

For $\alpha^\star =$ we have a total runtime of
$O^*($ $)$



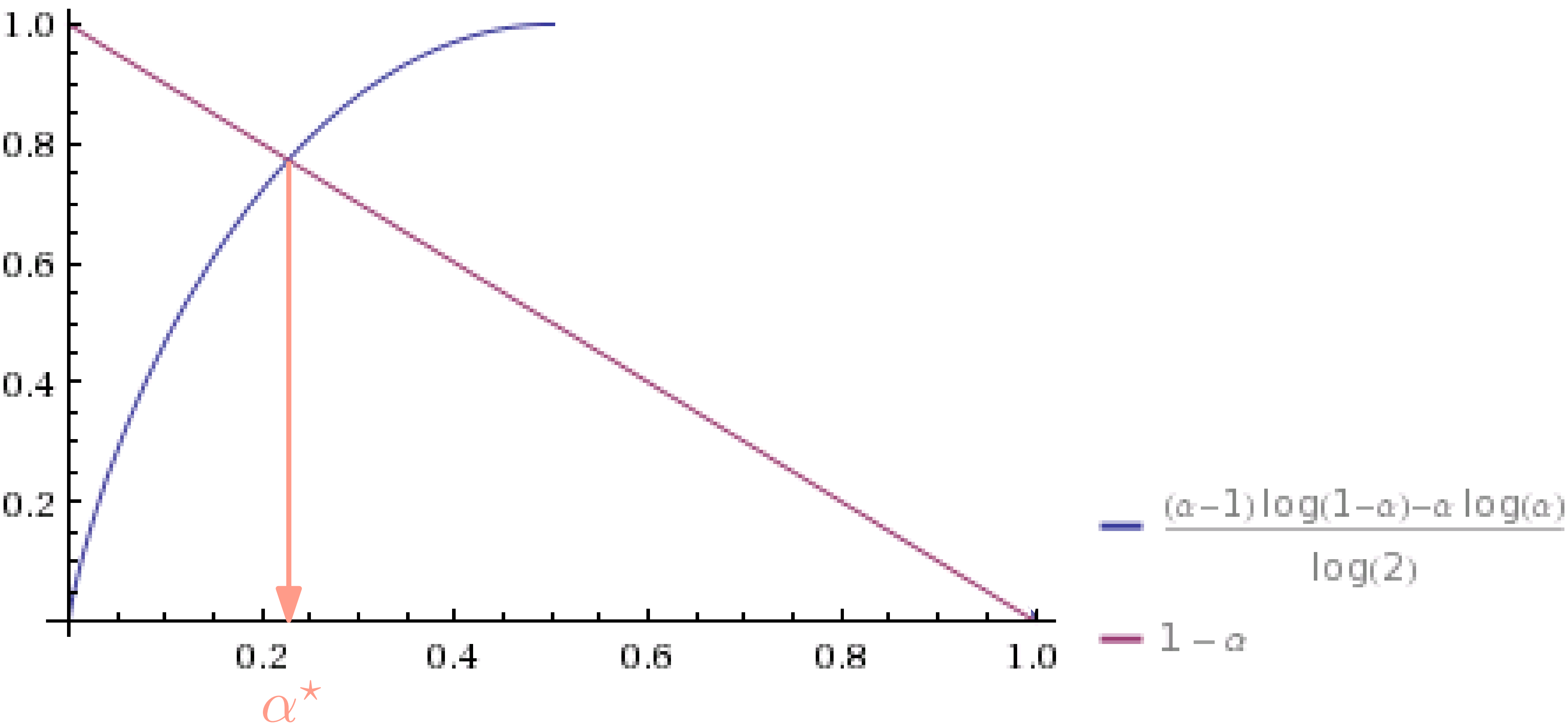
Finding $\alpha^\star$ and the Base

For $\alpha^\star =$ we have a total runtime of $O^*($ $)$



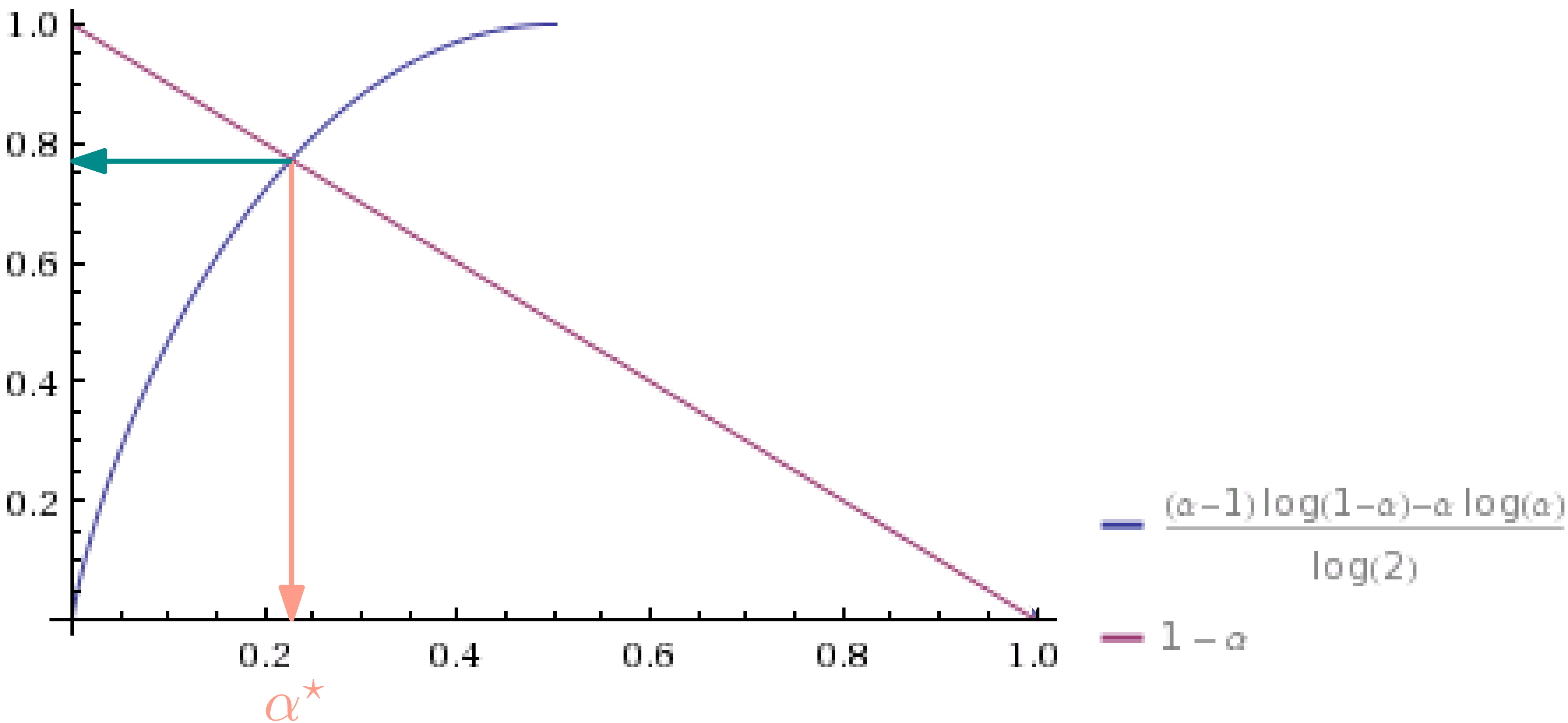
Finding $\alpha^\star$ and the Base

For $\alpha^\star = 0.22711$, we have a total runtime of
$O^*(\qquad)$



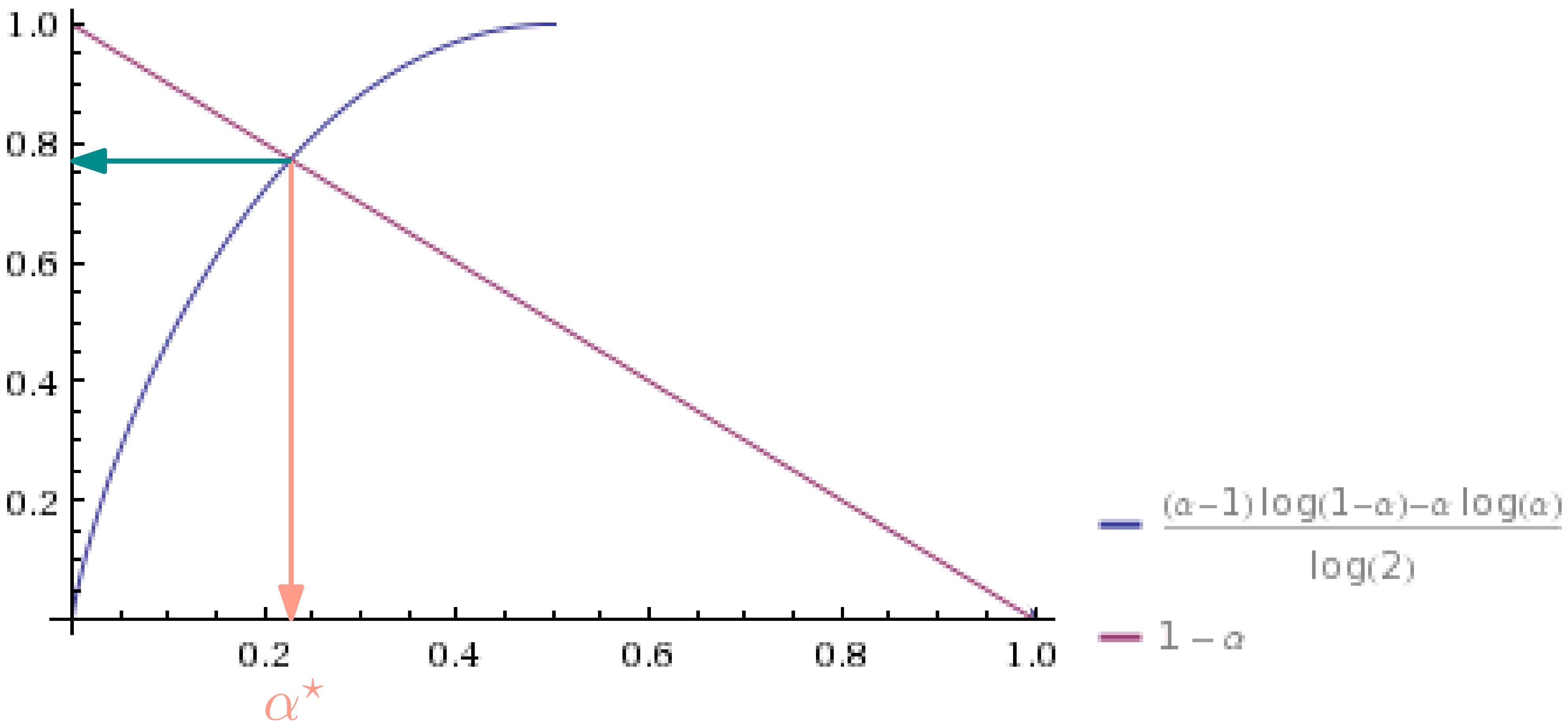
Finding $\alpha^\star$ and the Base

For $\alpha^\star = 0.22711$, we have a total runtime of
$O^*(\qquad)$



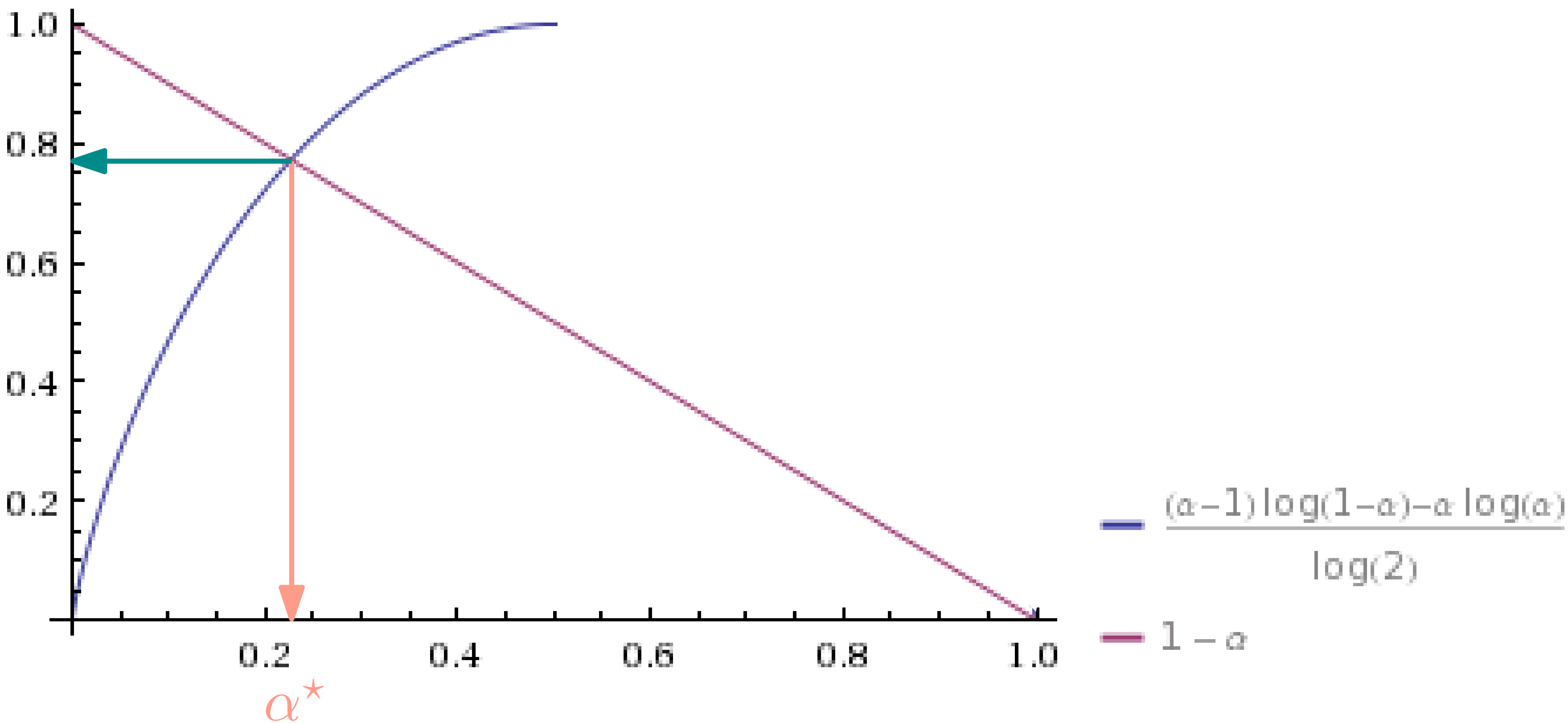
Finding $\alpha^\star$ and the Base

For $\alpha^\star = 0.22711$, we have a total runtime of $O^*(2^{0.7729n})$



Finding $\alpha^\star$ and the Base

For $\alpha^\star = 0.22711$, we have a total runtime of $O^*(2^{0.7729n}) = O(1.7088^n)$.



Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\big(2^{h(\alpha)n}\big)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha) n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha) \log_2 (1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Note: $\binom{n}{0} \leq \binom{n}{1} \leq \cdots \leq \binom{n}{\lceil n/2 \rceil}$.

Proof of the Helper Lemma

Recall the statement:

For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Note: $\binom{n}{0} \leq \binom{n}{1} \leq \cdots \leq \binom{n}{\lceil n/2 \rceil}$.

Stirlings formula:

Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Note: $\binom{n}{0} \leq \binom{n}{1} \leq \cdots \leq \binom{n}{\lceil n/2 \rceil}$.

Stirlings formula: $\sqrt{2\pi n}\left(\frac{n}{e}\right)^n \leq n! \leq 2\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha)n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Note: $\binom{n}{0} \leq \binom{n}{1} \leq \cdots \leq \binom{n}{\lceil n/2 \rceil}$.

Stirlings formula: $\sqrt{2\pi n}\left(\frac{n}{e}\right)^n \leq n! \leq 2\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

$$\binom{n}{k} = \#\text{ of } k\text{-element subsets of an } n\text{-element set}$$

Proof of the Helper Lemma

Recall the statement: For $\alpha \in (0, \frac{1}{2}]$, we have $\sum_{i=1}^{\alpha n} \binom{n}{i} \in O^*\left(2^{h(\alpha) n}\right)$, where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha)\log_2(1-\alpha)$.

$\alpha \in (0, \frac{1}{2}]$ implies

$$\sum_{i=1}^{\alpha n} \binom{n}{i} \leq \alpha n \cdot \binom{n}{\alpha n} \in O^*\left(\binom{n}{\alpha n}\right),$$

Note: $\binom{n}{0} \leq \binom{n}{1} \leq \cdots \leq \binom{n}{\lceil n/2 \rceil}$.

Stirlings formula: $\sqrt{2\pi n}\left(\frac{n}{e}\right)^n \leq n! \leq 2\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

$$\binom{n}{k} = \#\text{ of } k\text{-element subsets of an } n\text{-element set}$$

$$= \frac{n!}{k!(n-k)!}$$

Proof of the Helper Lemma (cont'd)

$$
\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(n/e)^n}{(\alpha n/e)^{\alpha n}\cdot((1-\alpha)n/e)^{(1-\alpha)n}}\right) \\
&= O^*\left(\qquad \qquad \qquad \right) \\
&= O^*\left(\qquad \qquad \qquad \qquad \right) \\
&= O^*\left(\qquad \right)
\end{aligned}
$$

Proof of the Helper Lemma (cont'd)

$$\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(\cancel{n/e})^n}{(\alpha \cancel{n/e})^{\alpha n} \cdot ((1-\alpha)\cancel{n/e})^{(1-\alpha)n}}\right) \\
&= O^*\left(\qquad \right) \\
&= O^*\left(\qquad \right) \\
&= O^*\left(\qquad \right)
\end{aligned}$$

Proof of the Helper Lemma (cont'd)

$$
\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(\cancel{n/e})^n}{(\alpha \cancel{n/e})^{\alpha n} \cdot ((1-\alpha)\cancel{n/e})^{(1-\alpha)n}}\right) \\
&= O^*\left(\alpha^{-\alpha n} \cdot (1-\alpha)^{-(1-\alpha)n}\right) \\
&= O^*\left(\qquad\qquad\qquad\qquad\qquad\qquad\right) \\
&= O^*\left(\qquad\quad\right)
\end{aligned}
$$

Proof of the Helper Lemma (cont'd)

$$\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(\cancel{n/e})^n}{(\alpha \cancel{n/e})^{\alpha n} \cdot ((1-\alpha)\cancel{n/e})^{(1-\alpha)n}}\right) \\
&= O^*\left(\alpha^{-\alpha n} \cdot (1-\alpha)^{-(1-\alpha)n}\right) \\
&= O^*\left(2^{-(\alpha \log_2 \alpha)\cdot n - (1-\alpha)\log_2(1-\alpha)\cdot n}\right) \\
&= O^*\left(\qquad \right)
\end{aligned}$$

Proof of the Helper Lemma (cont'd)

$$\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(\cancel{n}/\cancel{e})^n}{(\alpha \cancel{n}/\cancel{e})^{\alpha n} \cdot ((1-\alpha)\cancel{n}/\cancel{e})^{(1-\alpha)n}}\right) \\
&= O^*\left(\alpha^{-\alpha n} \cdot (1-\alpha)^{-(1-\alpha)n}\right) \\
&= O^*\left(2^{-(\alpha \log_2 \alpha)\cdot n-(1-\alpha)\log_2(1-\alpha)\cdot n}\right) \\
&= O^*\left(\qquad \right)
\end{aligned}$$

Note: $h(\alpha) = -\alpha \log_2(\alpha) - (1-\alpha)\log_2(1-\alpha)$

Proof of the Helper Lemma (cont'd)

$$\begin{aligned}
\binom{n}{\alpha n} &\in O^*\left(\frac{(\cancel{n}/\cancel{e})^n}{(\alpha \cancel{n}/\cancel{e})^{\alpha n} \cdot ((1-\alpha)\cancel{n}/\cancel{e})^{(1-\alpha)n}}\right) \\
&= O^*\left(\alpha^{-\alpha n} \cdot (1-\alpha)^{-(1-\alpha)n}\right) \\
&= O^*\left(2^{-(\alpha \log_2 \alpha)\cdot n - (1-\alpha)\log_2(1-\alpha)\cdot n}\right) \\
&= O^*\left(2^{h(\alpha)\cdot n}\right) \qquad \square
\end{aligned}$$

Note: $h(\alpha) = -\alpha \log_2(\alpha) - (1-\alpha)\log_2(1-\alpha)$