Submodular Optimization Under Uncertainty

Online, Dynamic and Streaming Algorithms

Carnegie Mellon University

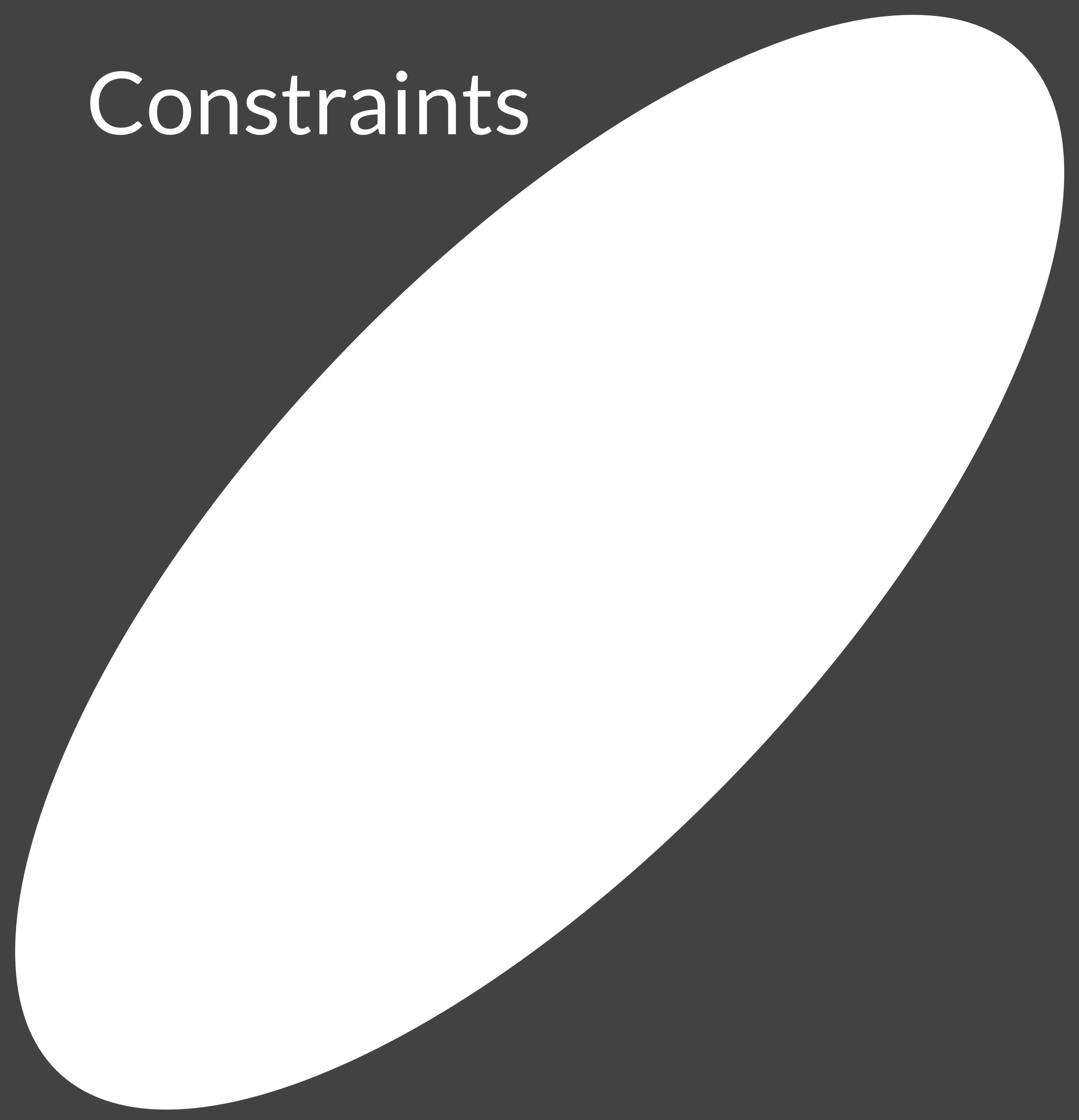
Roie Levin

Committee: Anupam Gupta, R. Ravi, David Woodruff, Chandra Chekuri, Seffi Naor

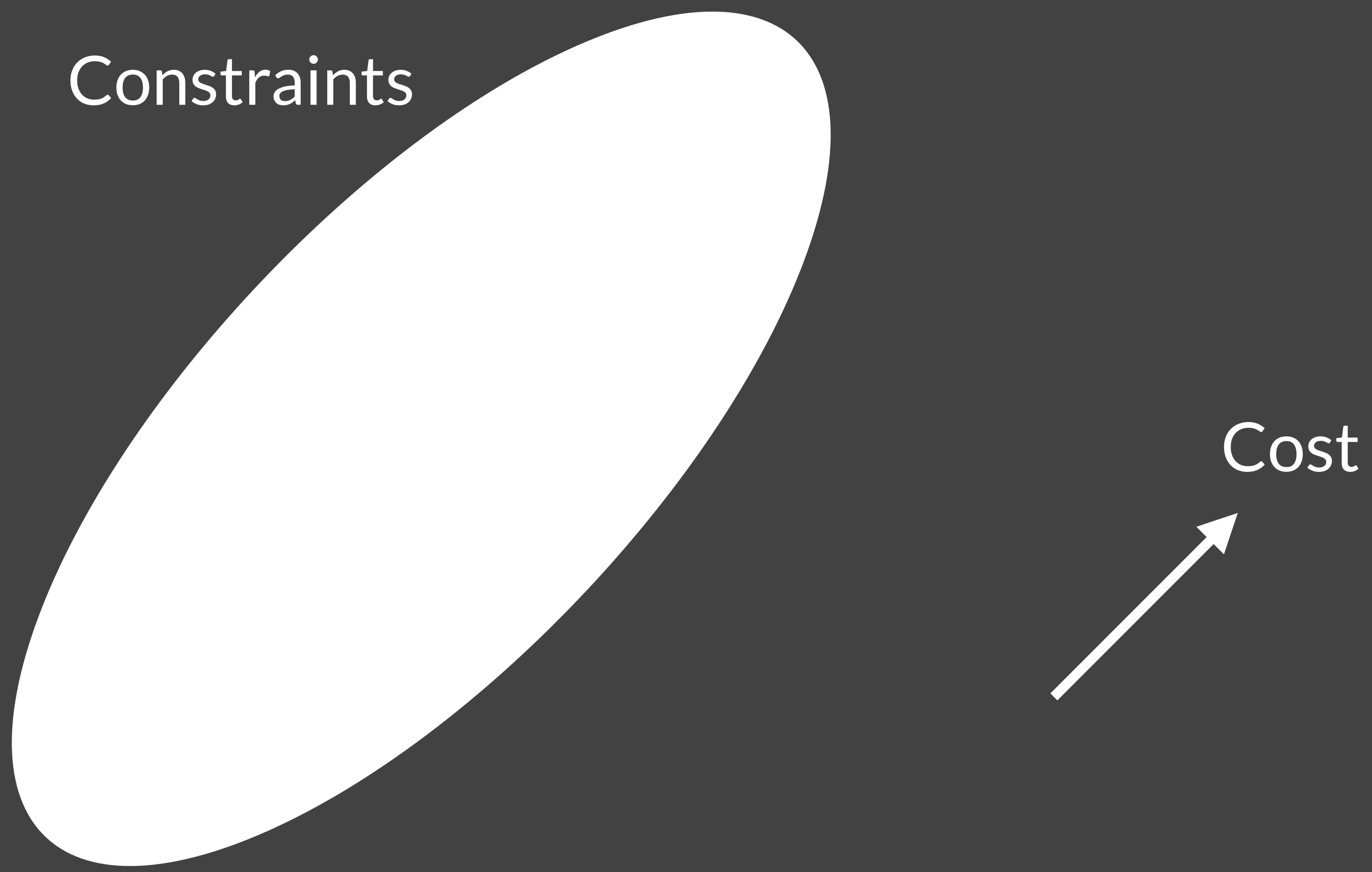
Intro

Classical Approximation Algorithms

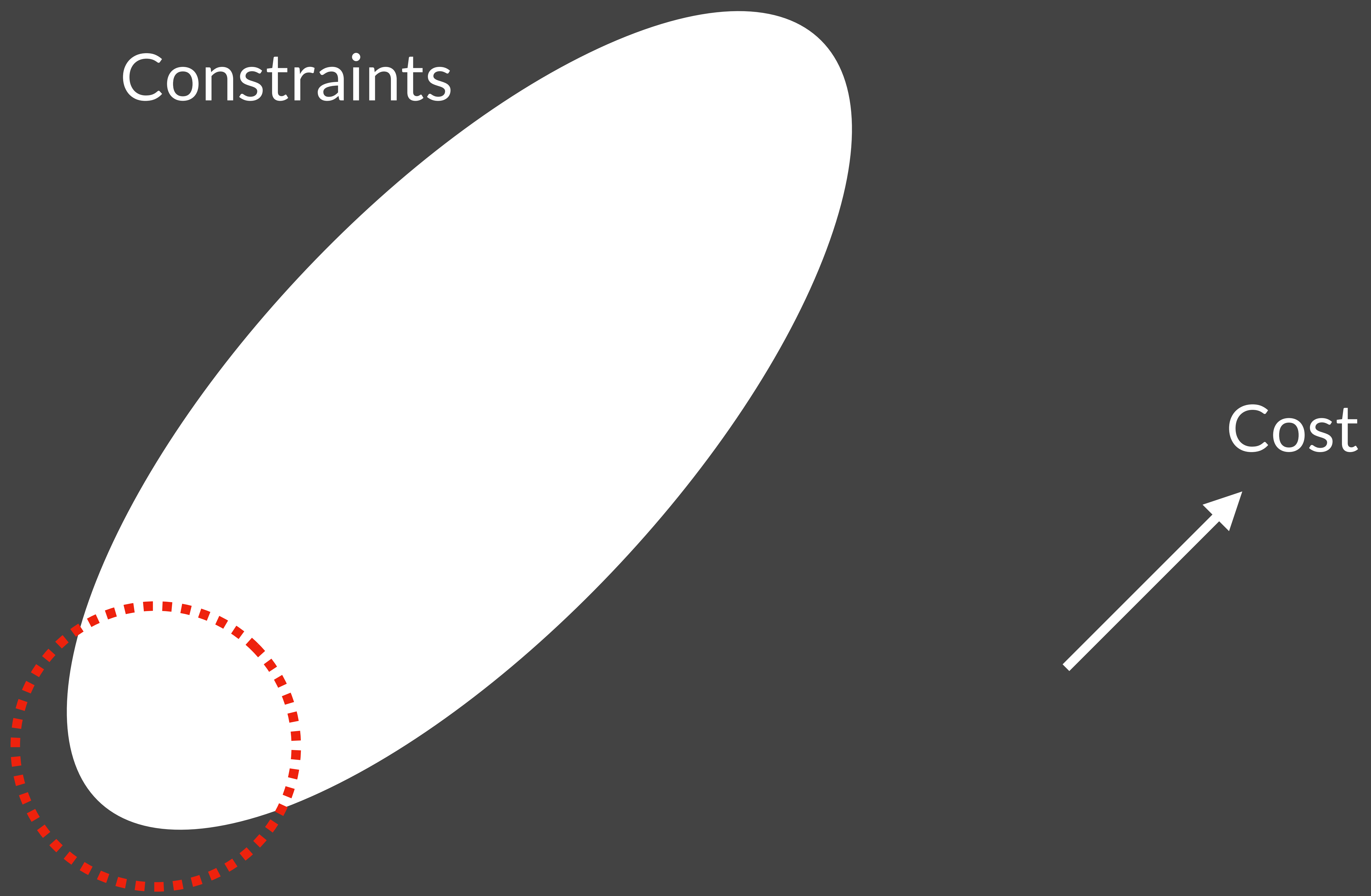
Classical Approximation Algorithms



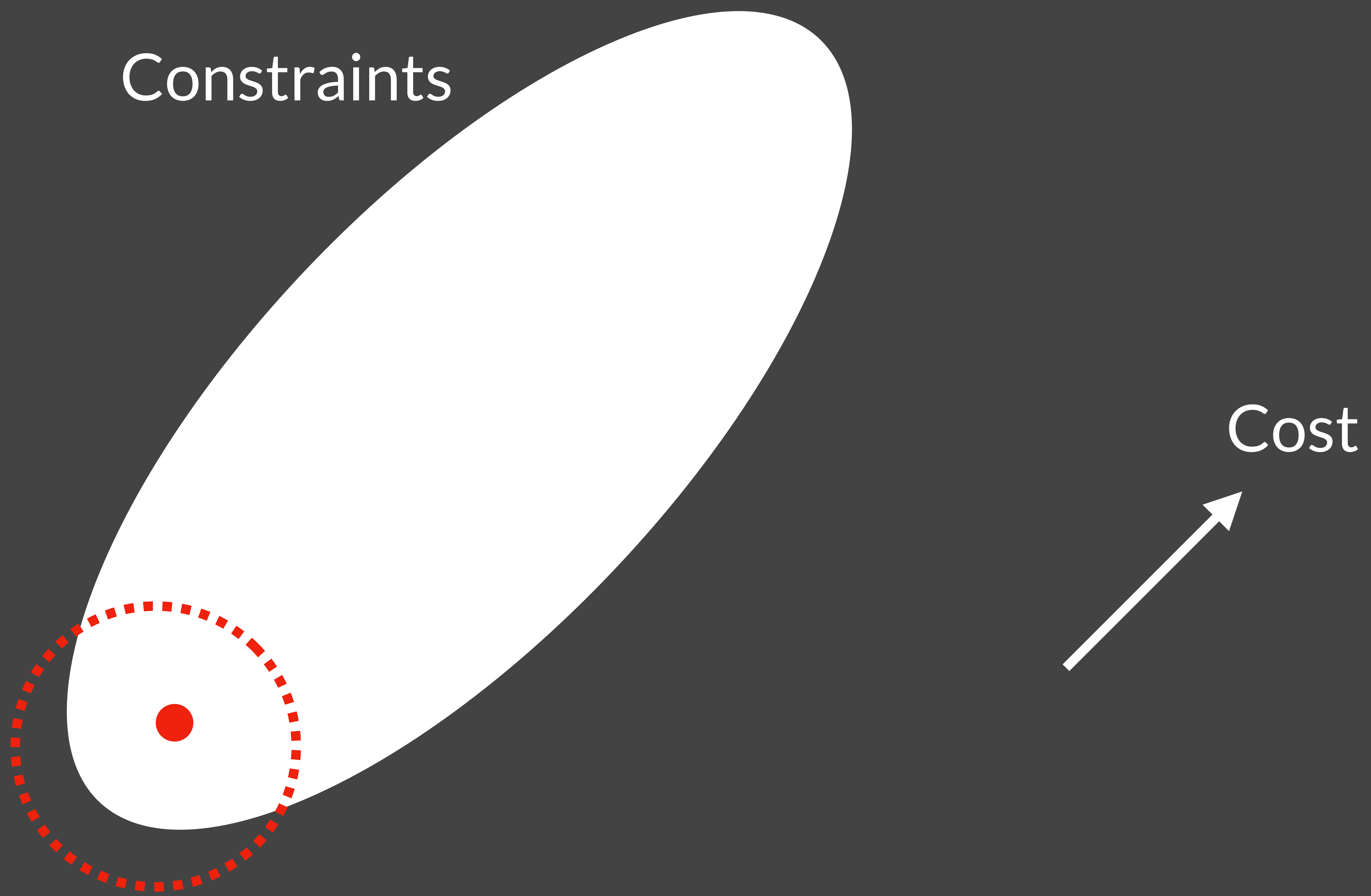
Classical Approximation Algorithms



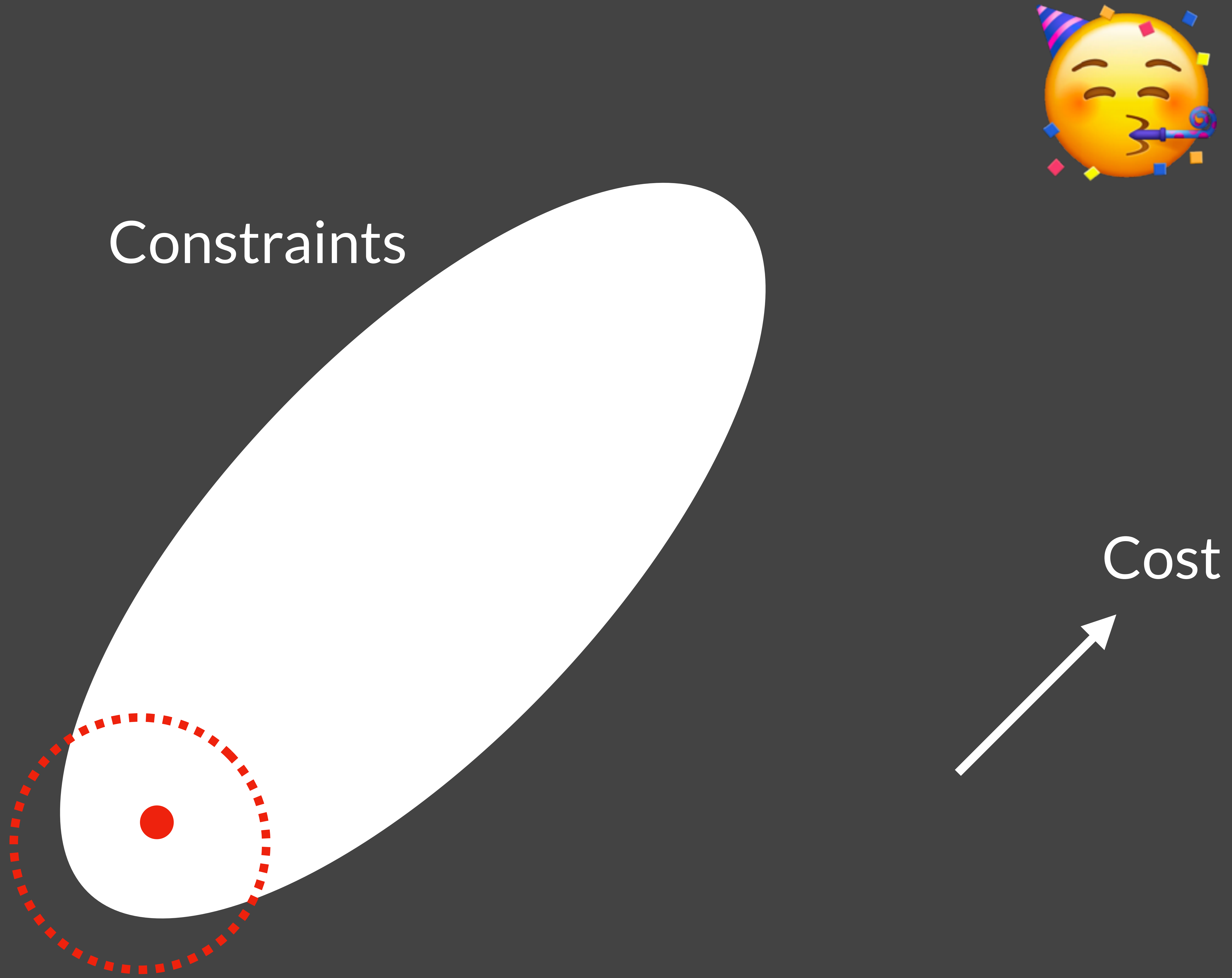
Classical Approximation Algorithms



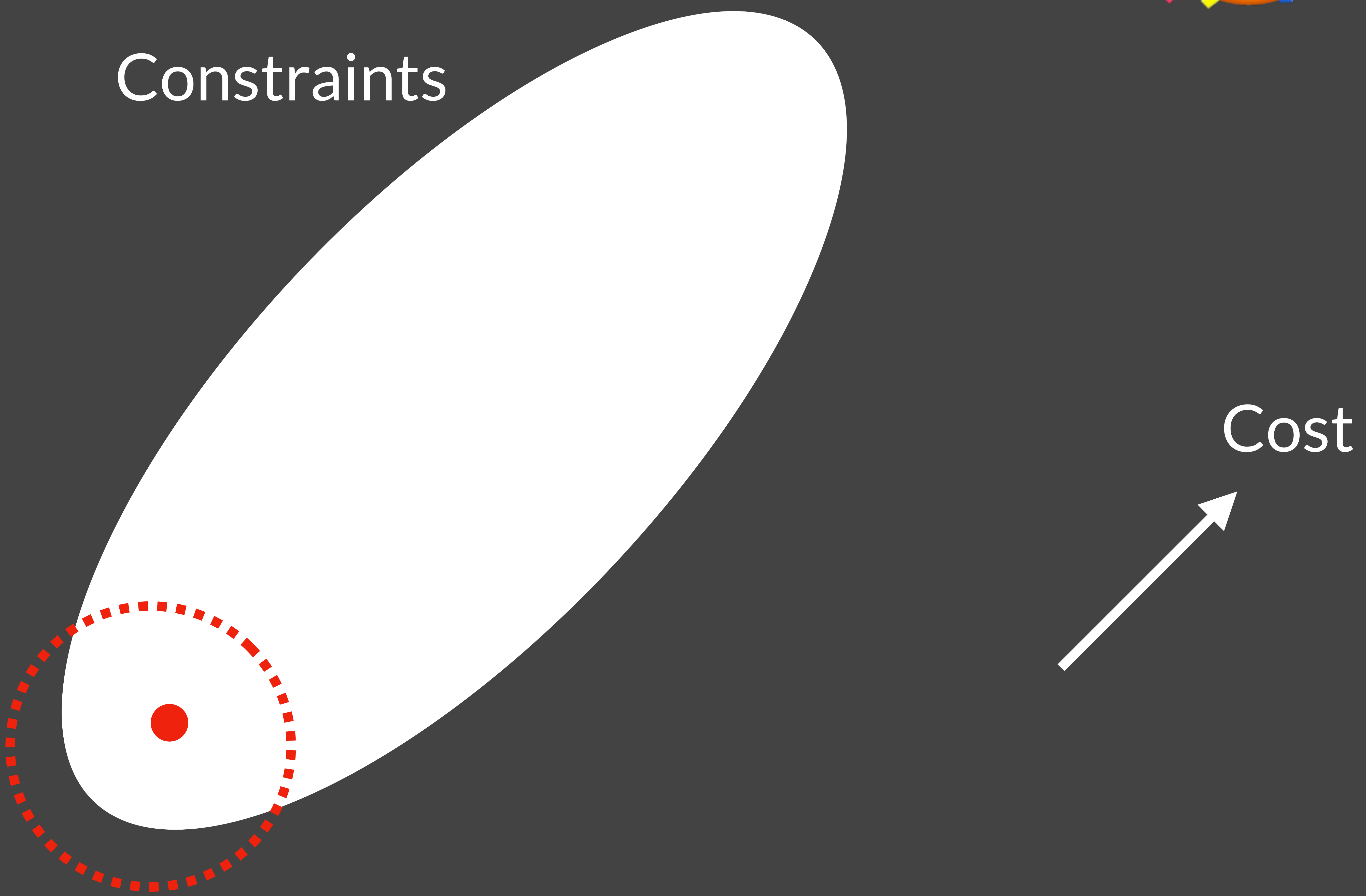
Classical Approximation Algorithms



Classical Approximation Algorithms

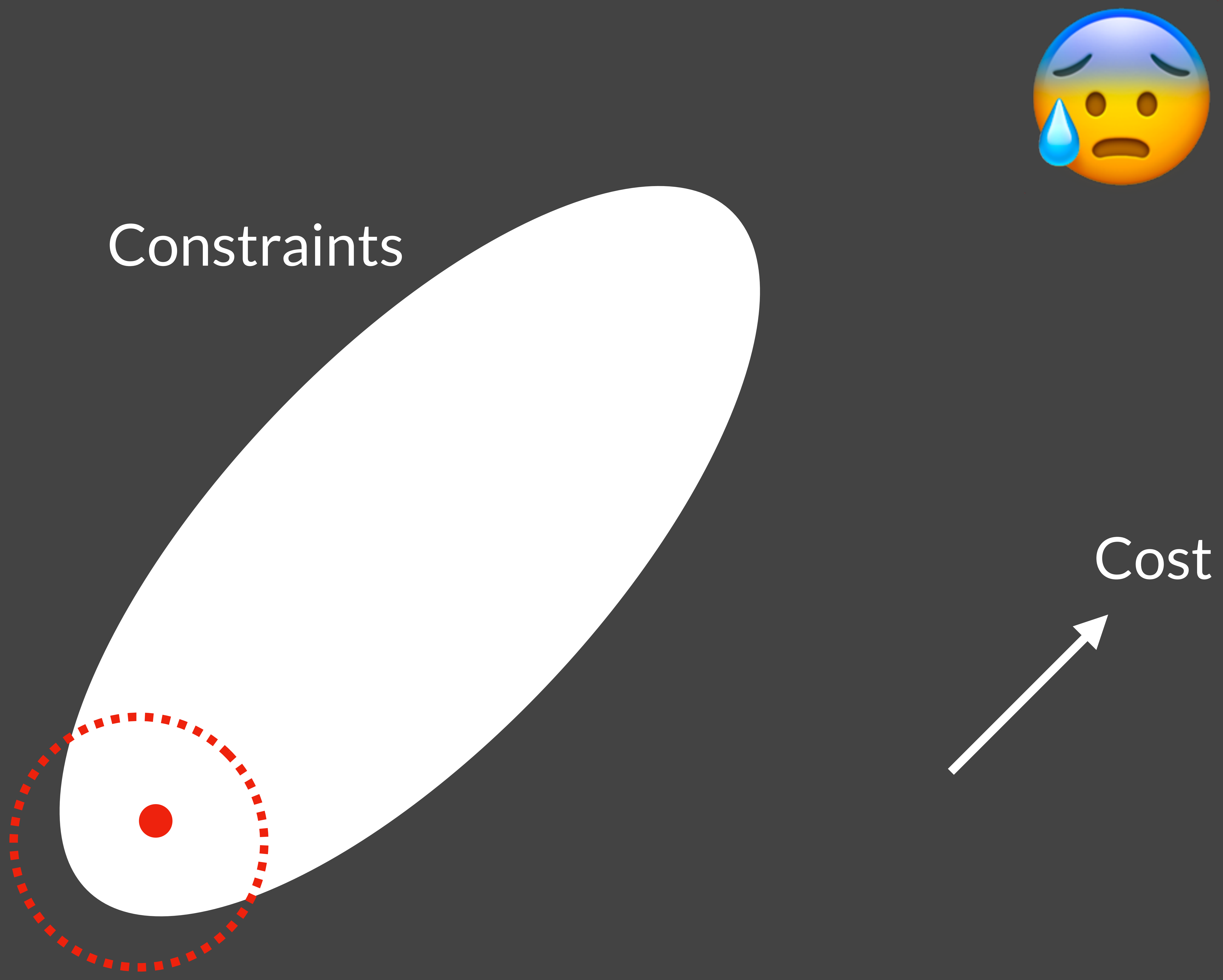


Classical Approximation Algorithms



Unrealistic to expect full/perfect information!

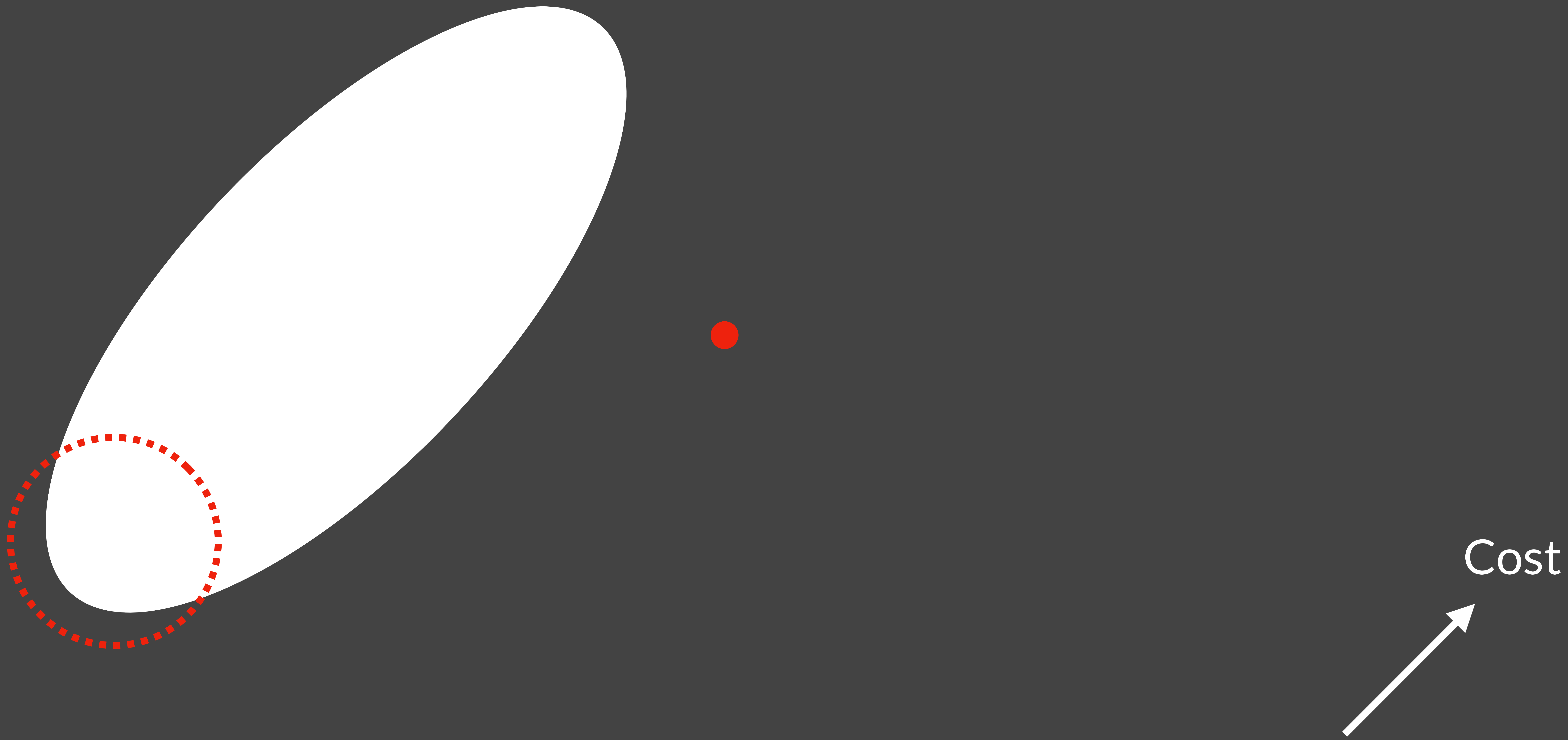
Classical Approximation Algorithms



Unrealistic to expect full/perfect information!

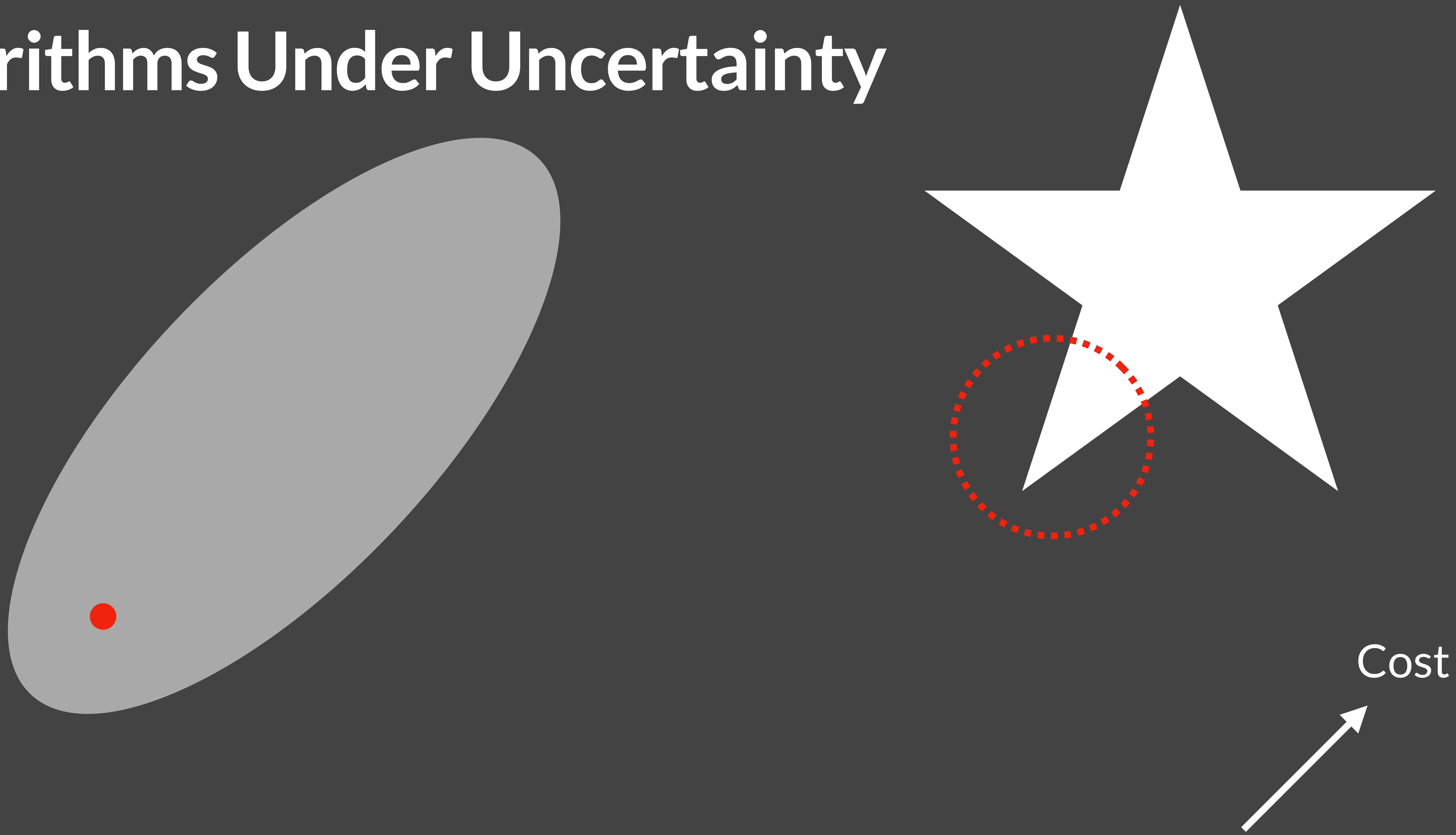
Algorithms Under Uncertainty

Algorithms Under Uncertainty

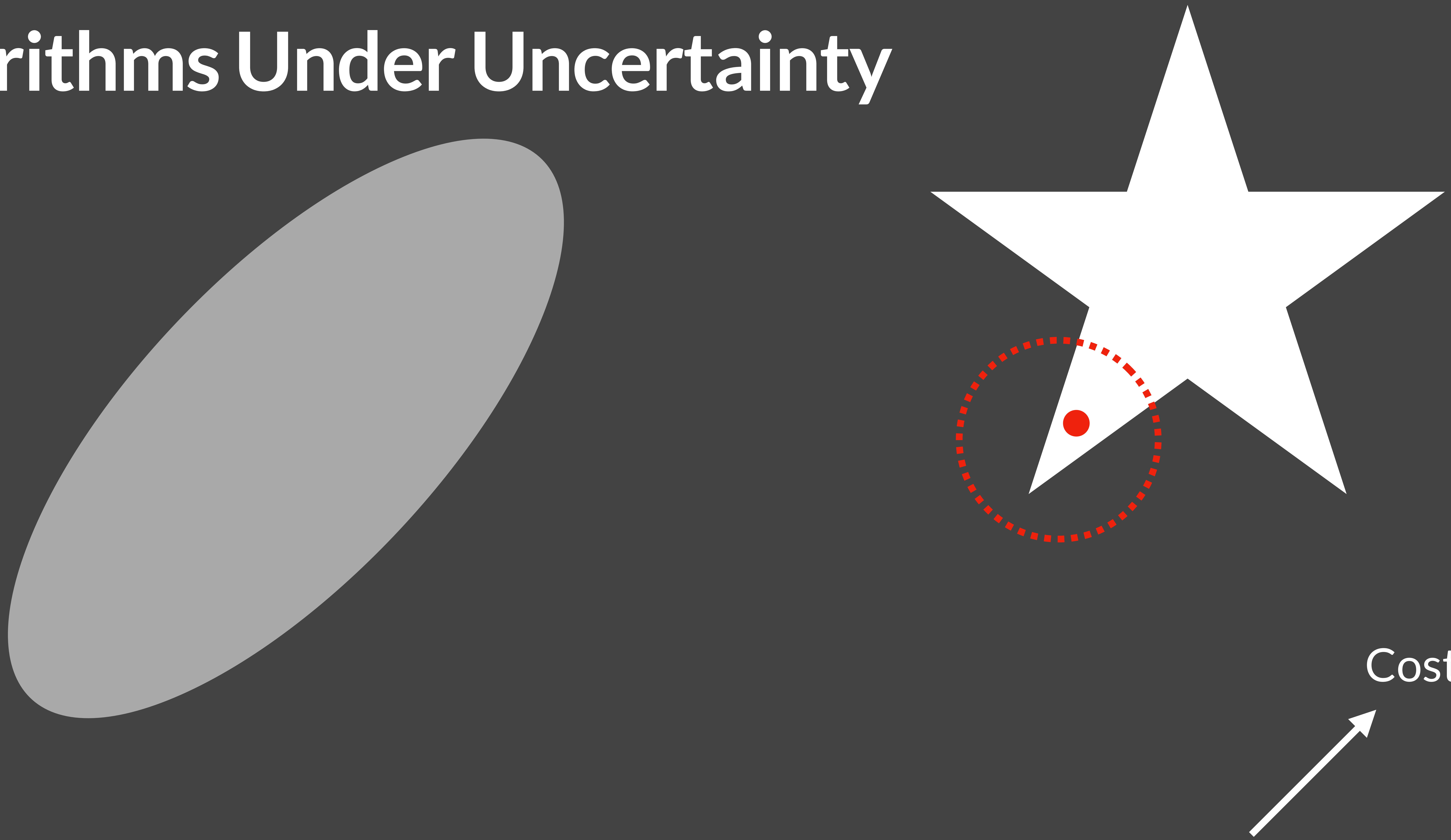


Algorithms Under Uncertainty

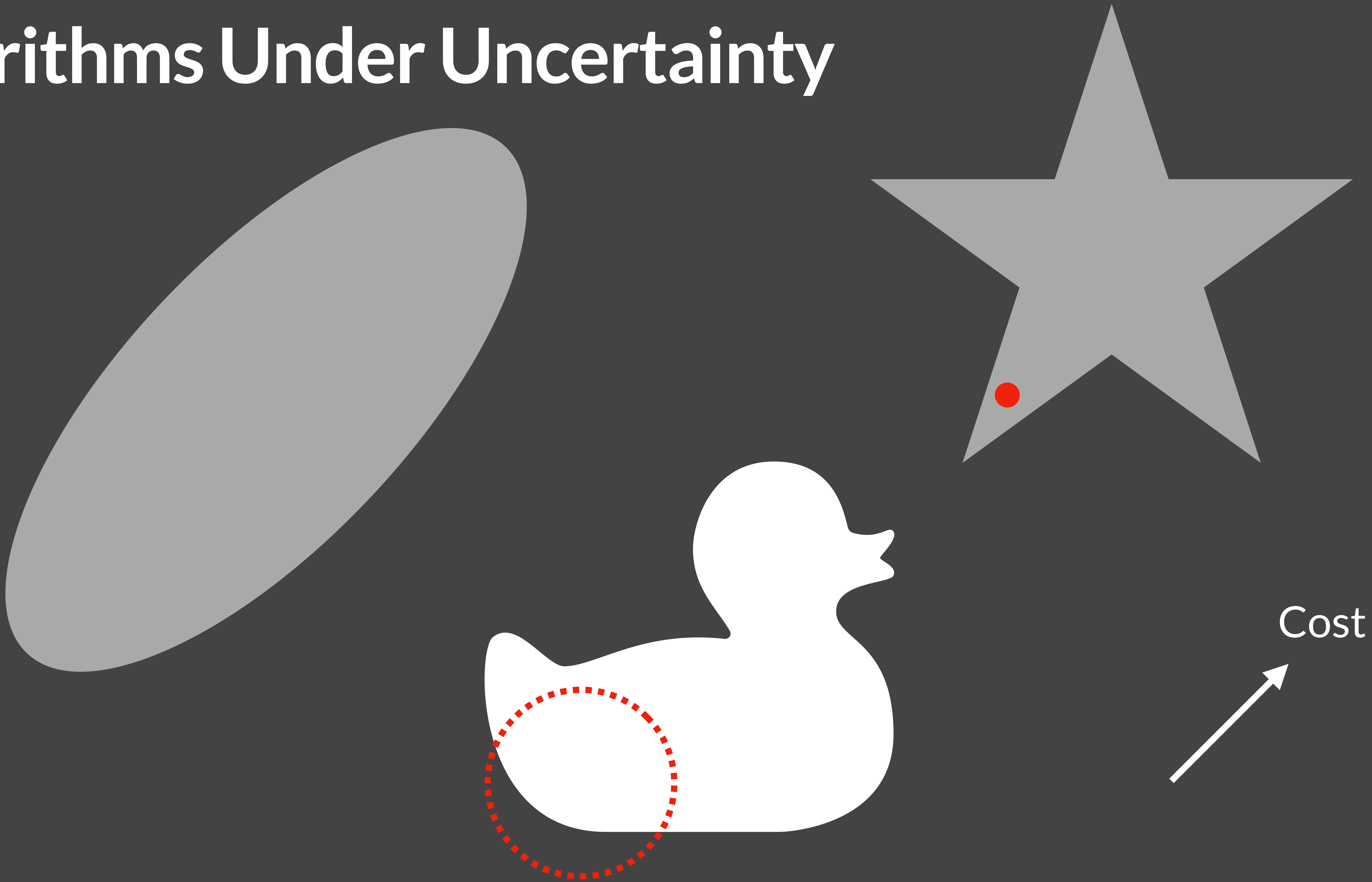
Algorithms Under Uncertainty



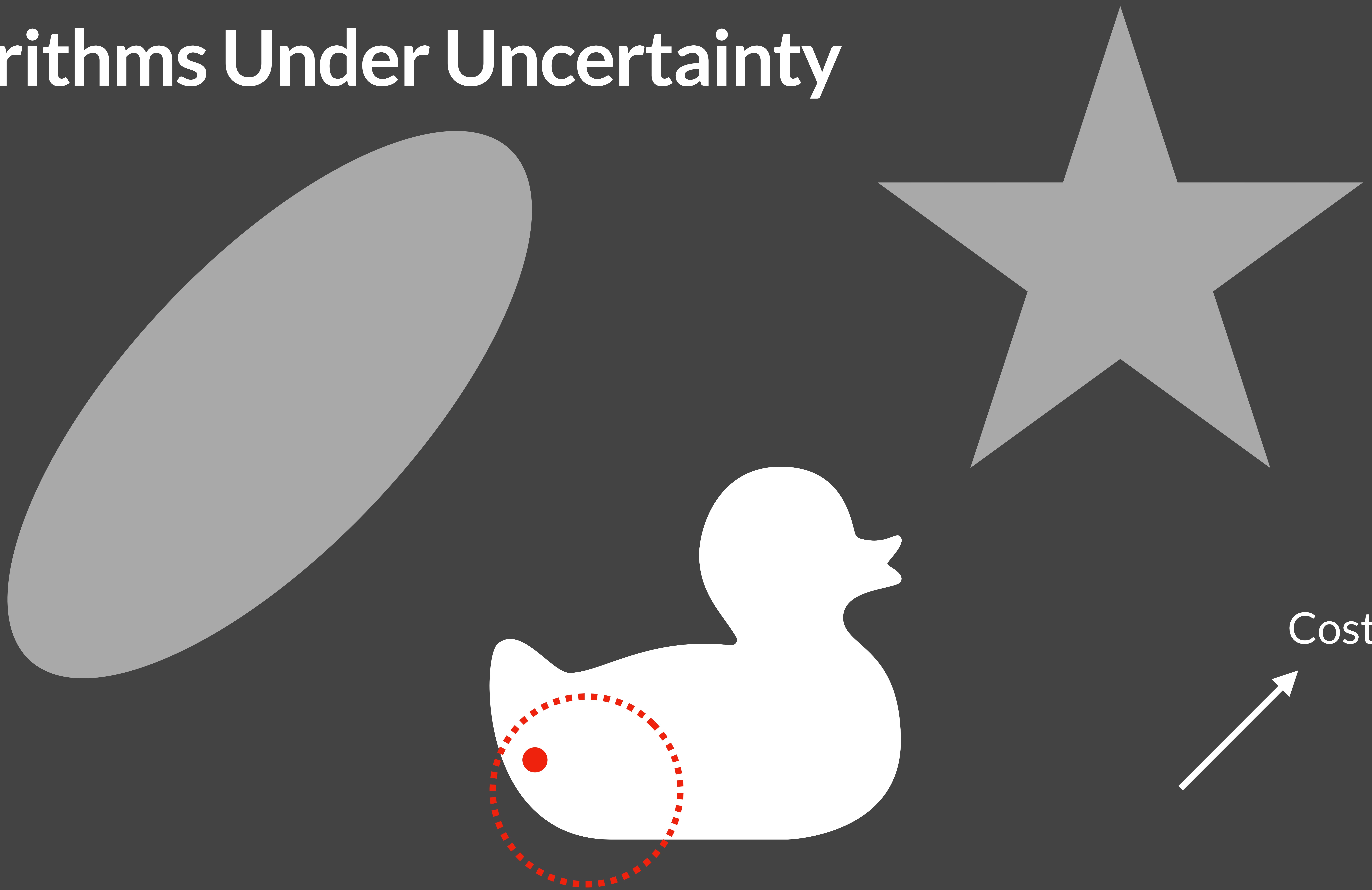
Algorithms Under Uncertainty



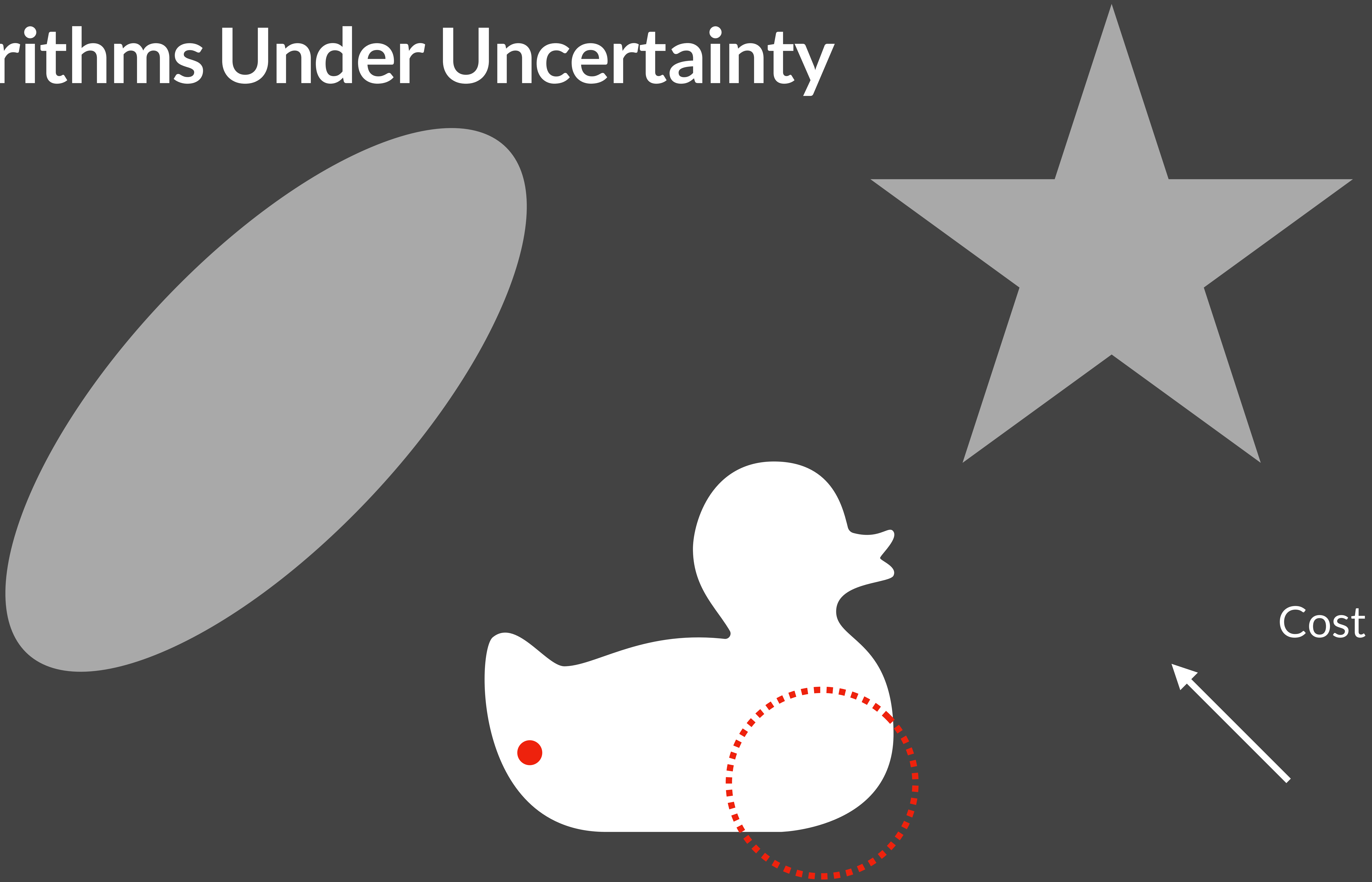
Algorithms Under Uncertainty



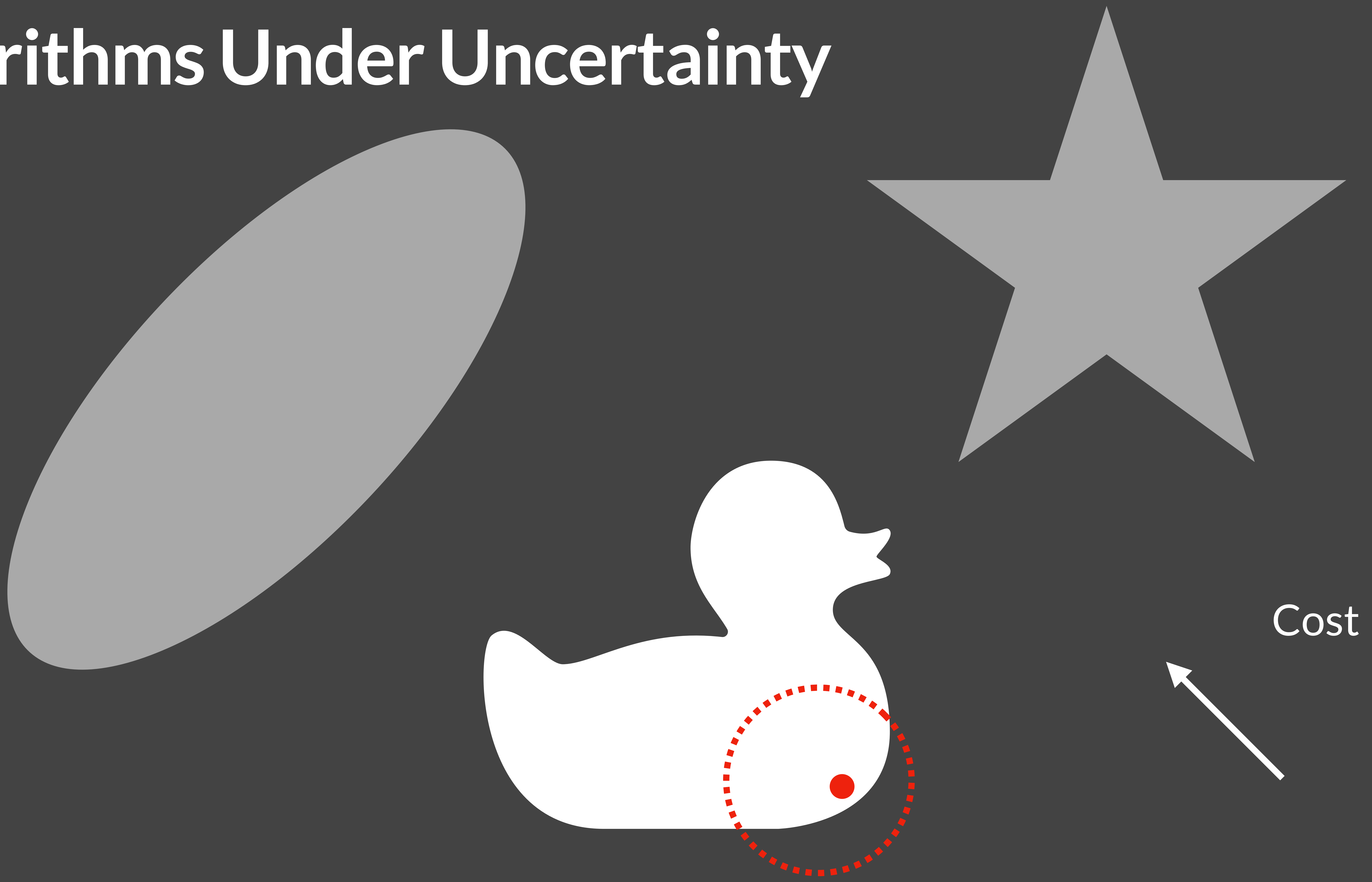
Algorithms Under Uncertainty



Algorithms Under Uncertainty

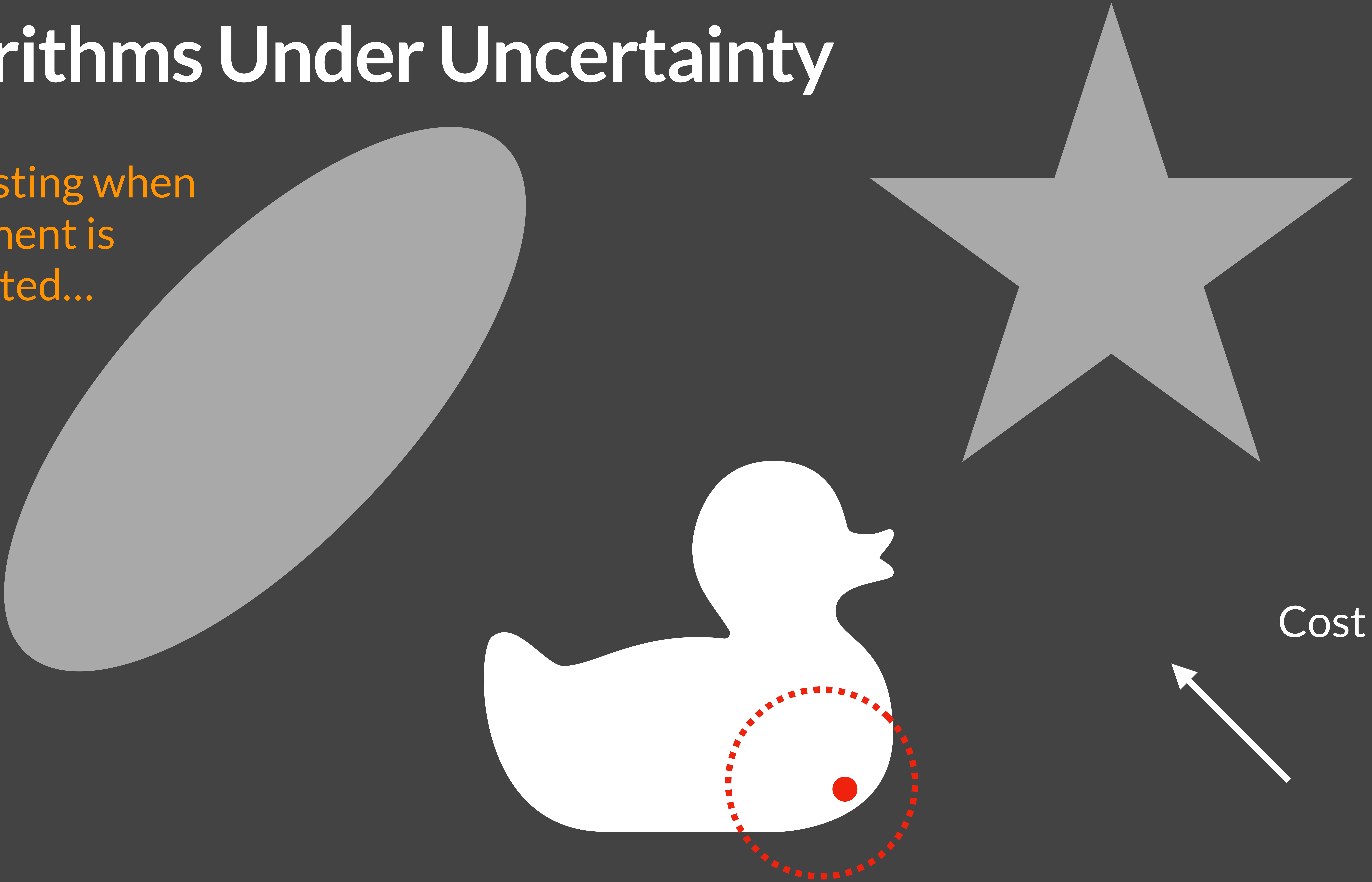


Algorithms Under Uncertainty



Algorithms Under Uncertainty

Interesting when movement is restricted...



Algorithms Under Uncertainty

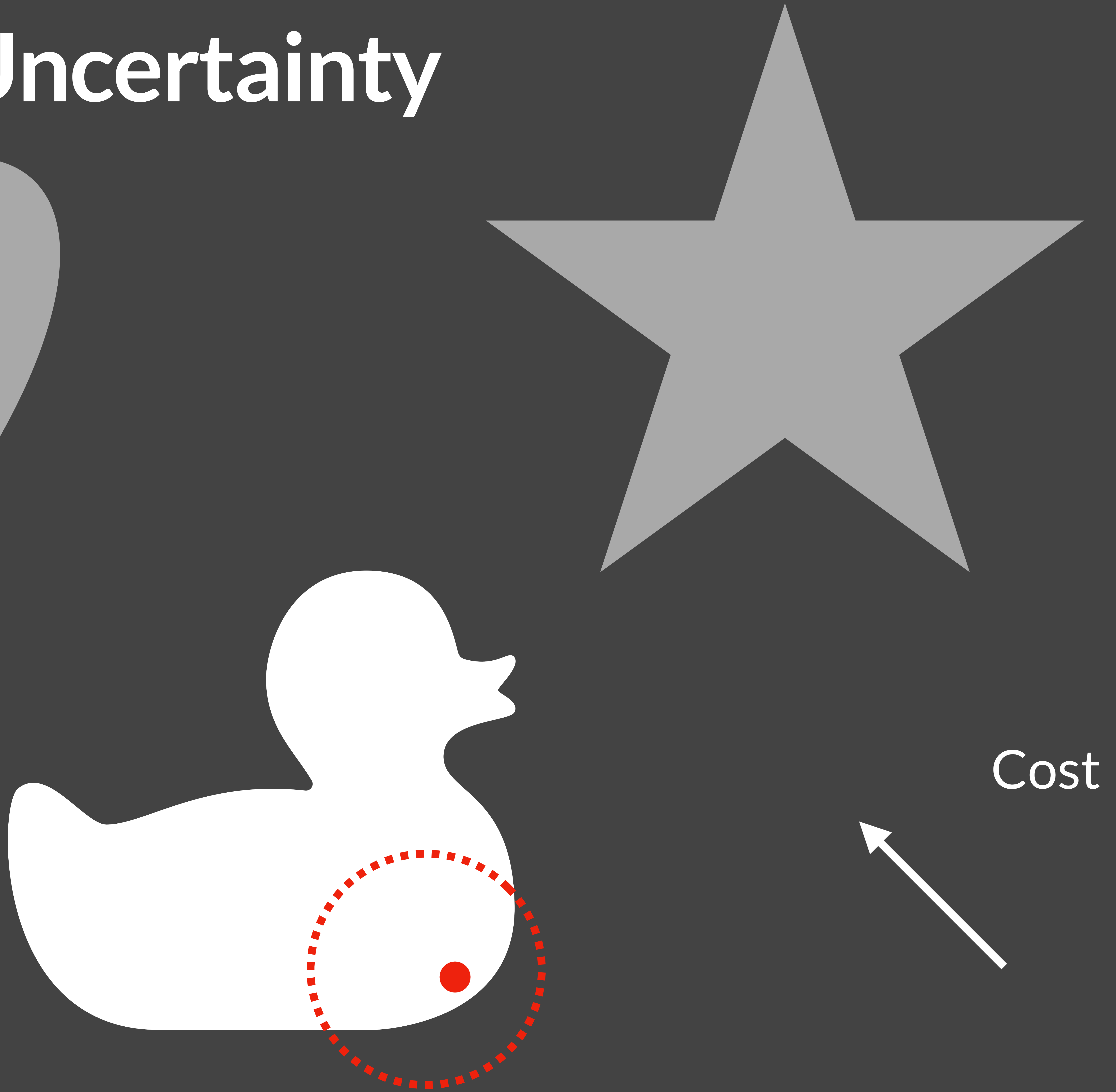
Interesting when movement is restricted…

Thesis studies 3 restrictions:

Online — monotone solution

Dynamic — low movement

Streaming — low memory



Algorithms Under Uncertainty

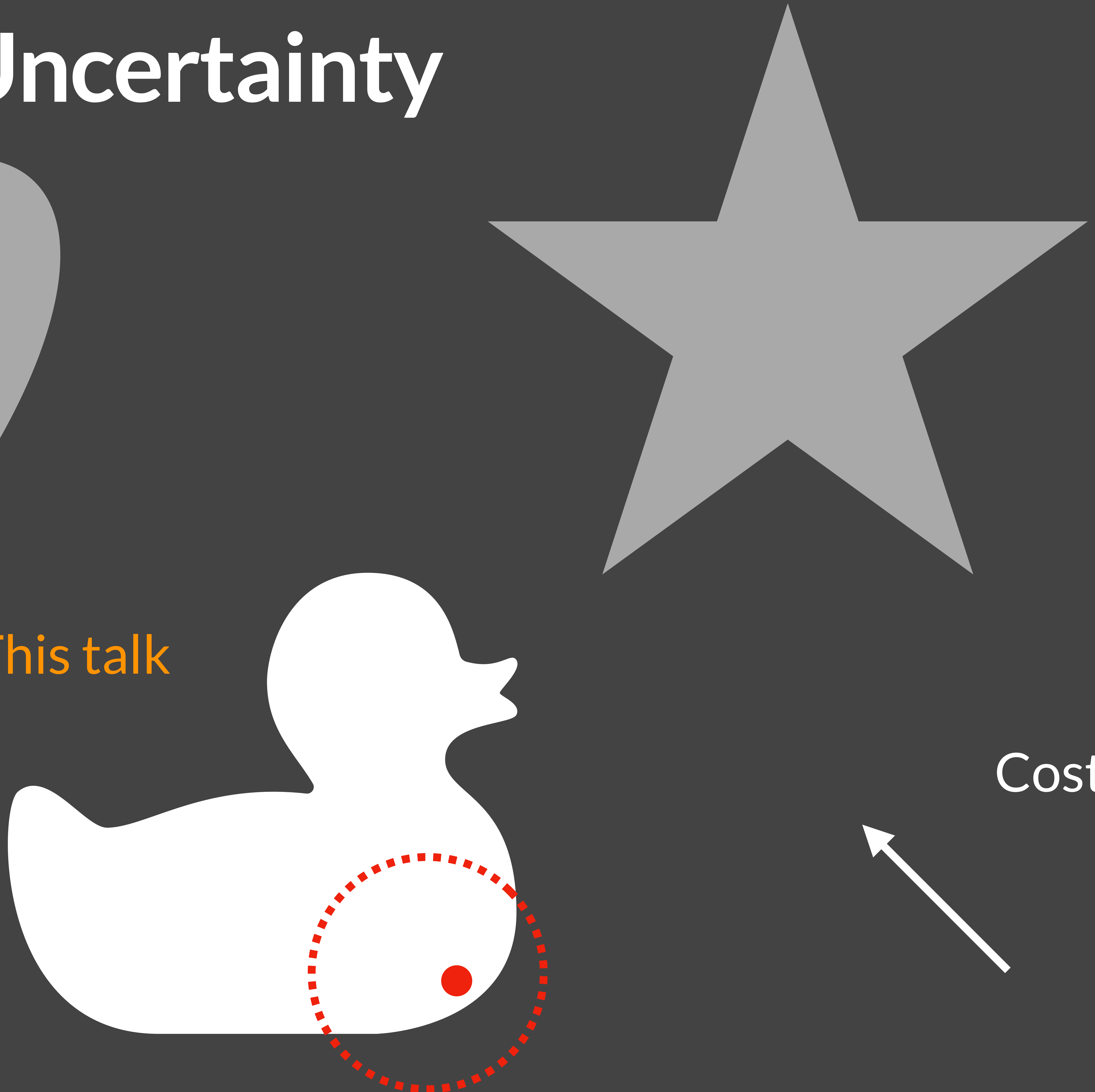
Interesting when movement is restricted...

Thesis studies 3 restrictions:

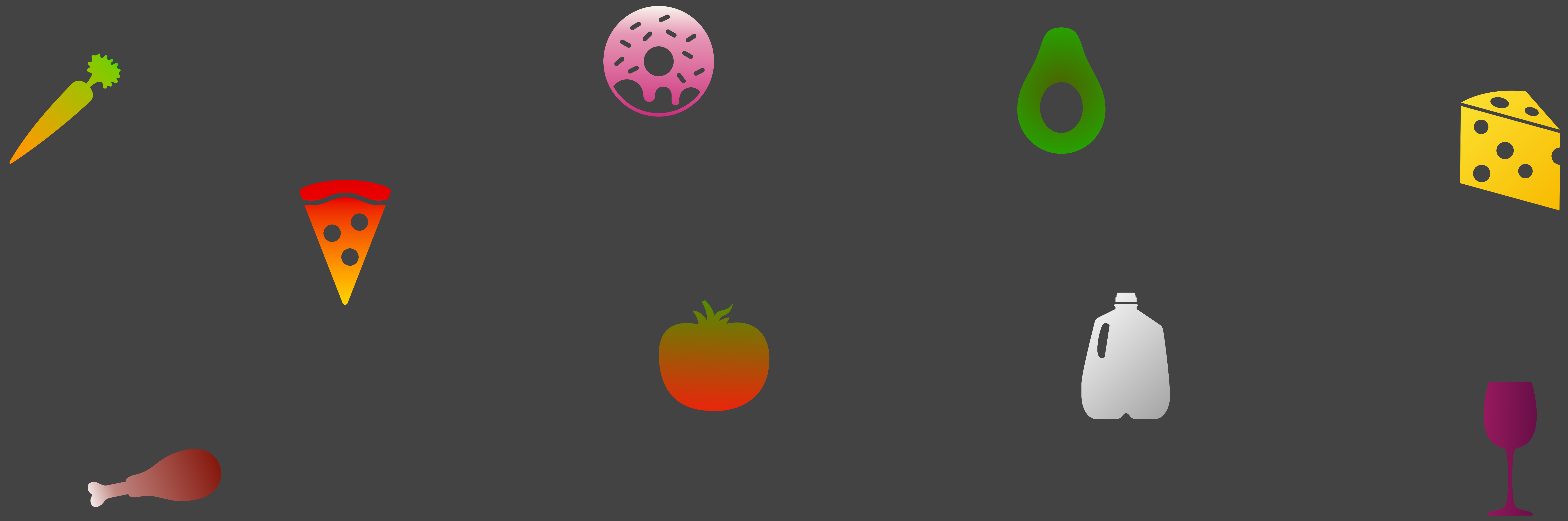
Online — monotone solution

Dynamic — low movement

Streaming — low memory



This Talk: Submodular Cover [Wolsey 82]

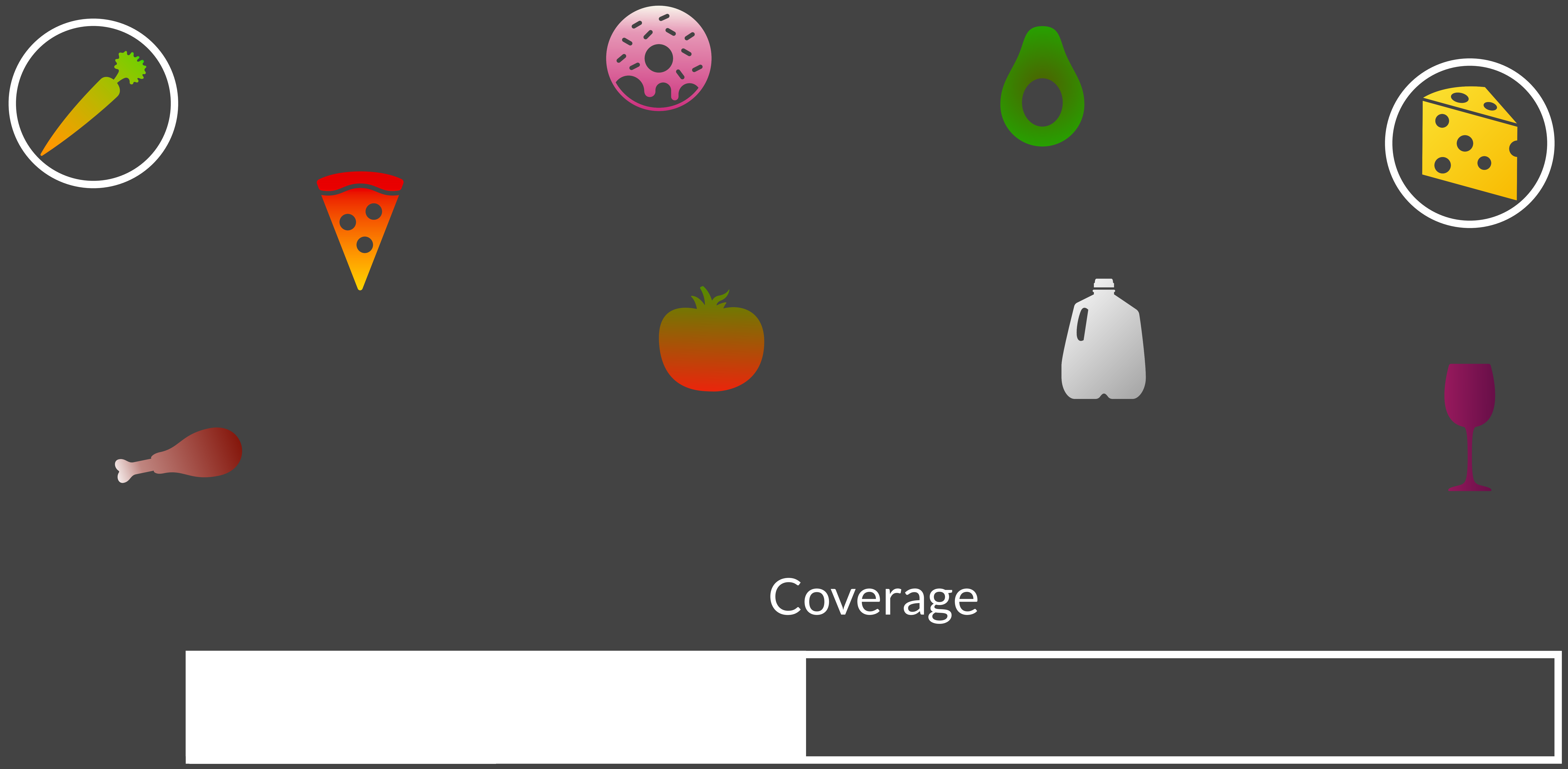


Coverage

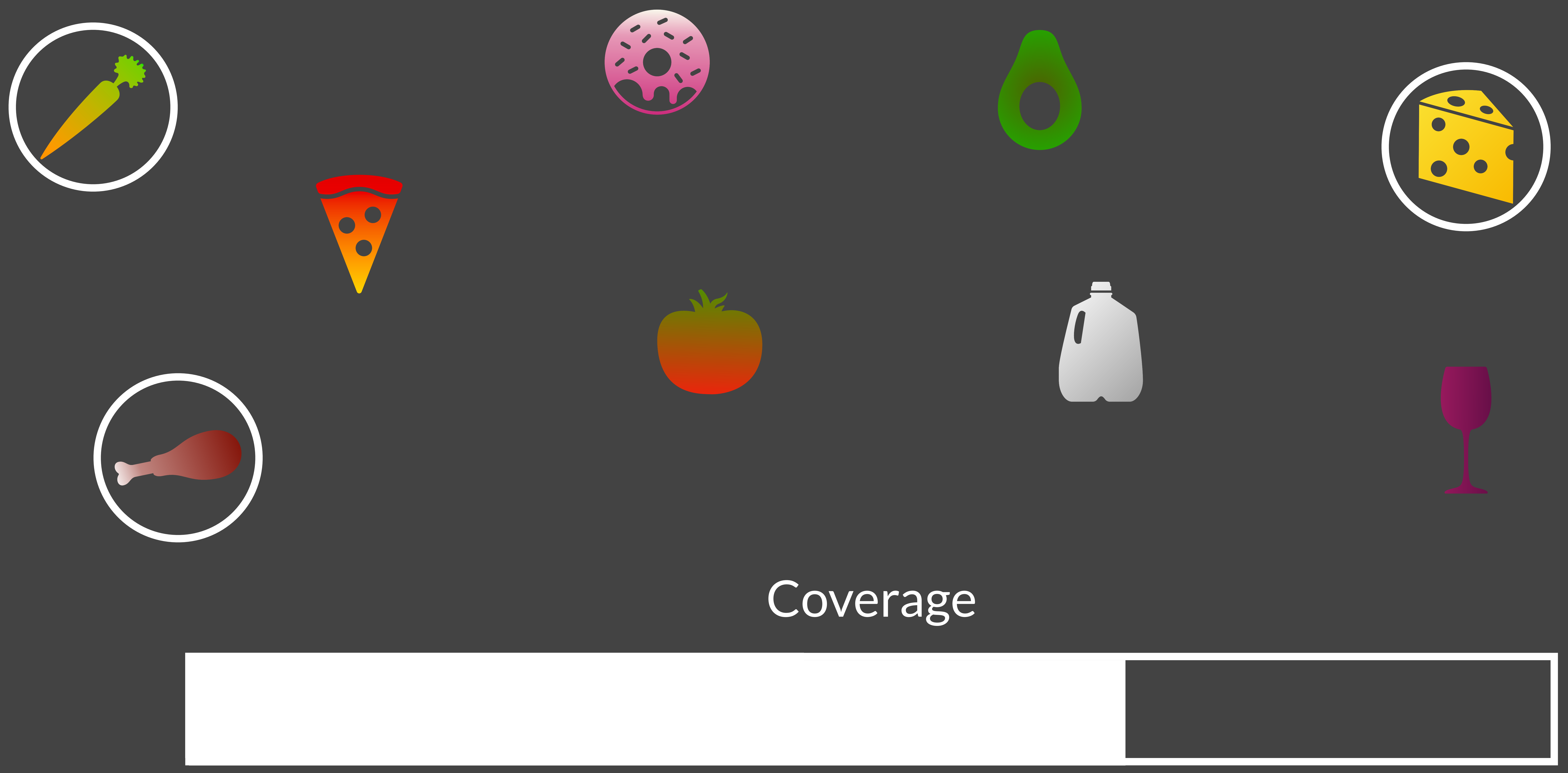
This Talk: Submodular Cover [Wolsey 82]

Coverage

This Talk: Submodular Cover [Wolsey 82]



This Talk: Submodular Cover [Wolsey 82]



This Talk: Submodular Cover [Wolsey 82]

Coverage

This Talk: Submodular Cover [Wolsey 82]

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$
- Coverage "Quality": $f(S)$

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$
- Coverage “Quality”: $f(S)$

Want **min cost** solution with **max coverage**!

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$
- Coverage "Quality": $f(S)$

Want **min cost** solution with **max coverage**!

$f : 2^{\mathcal{N}} \to \mathbb{R}$ is **monotone**, **nonnegative** and **submodular**.

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$
- Coverage "Quality": $f(S)$

$$\min_{S \subseteq \mathcal{N}} c(S)$$

$$f(S) \geq f(\mathcal{N})$$

$$S \in \{0,1\}^m$$

Want **min cost** solution with **max coverage**!

$f : 2^{\mathcal{N}} \to \mathbb{R}$ is **monotone**, **nonnegative** and **<u>submodular</u>**.

This Talk: Submodular Cover [Wolsey 82]

- Universe of choices: $\mathcal{N} = \{u_1, u_2, \ldots, u_m\}$
- Solution: $S \subseteq \mathcal{N}$
- Cost: $c(S)$
- Coverage "Quality": $f(S)$

$$\min_{S \subseteq \mathcal{N}} c(S)$$

$$f(S) \geq f(\mathcal{N})$$

$$S \in \{0,1\}^m$$

Want **min cost** solution with **max coverage**!

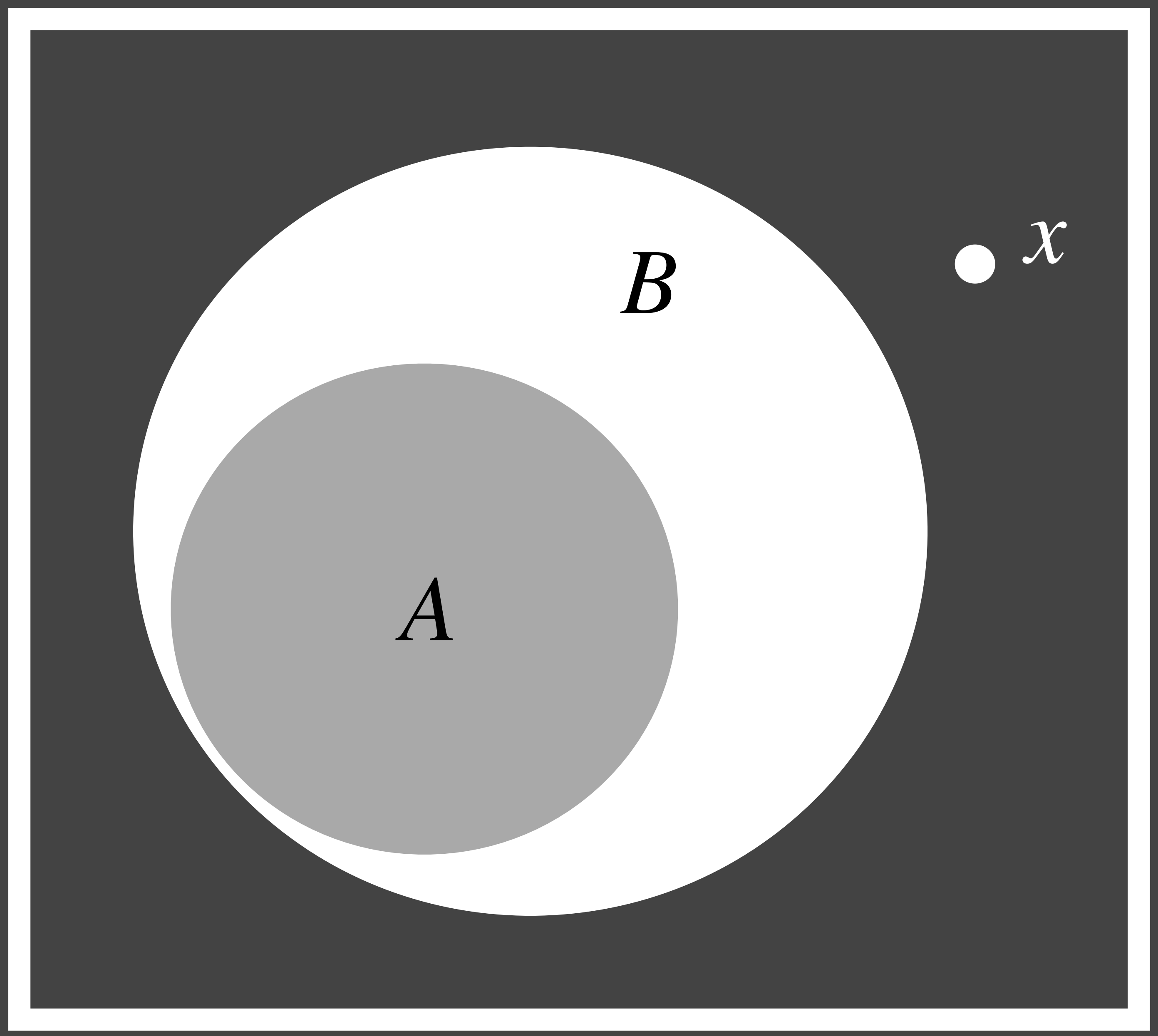
$f : 2^{\mathcal{N}} \to \mathbb{R}$ is **monotone**, **nonnegative** and <u>**submodular**</u>.

<u>This talk:</u>

f integer valued,
all costs are 1.

Submodularity

Submodularity

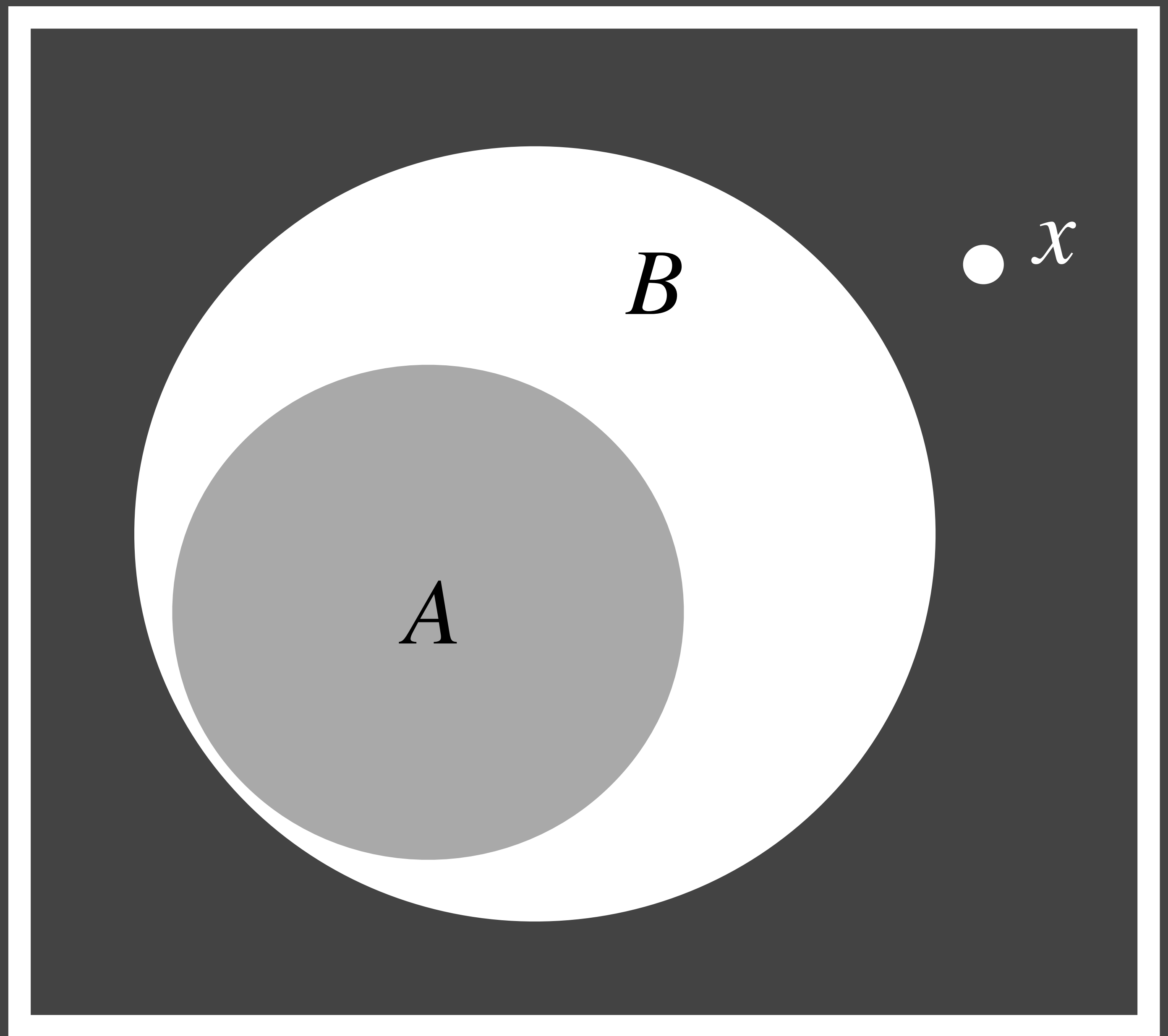


Definition: f is ***submodular*** *if*, $\forall A \subseteq B, x \notin B,$

Submodularity

Definition: f is ***submodular*** *if*, $\forall A \subseteq B, x \notin B$,

$$f(A+x) - f(A) \geq f(B+x) - f(B)$$

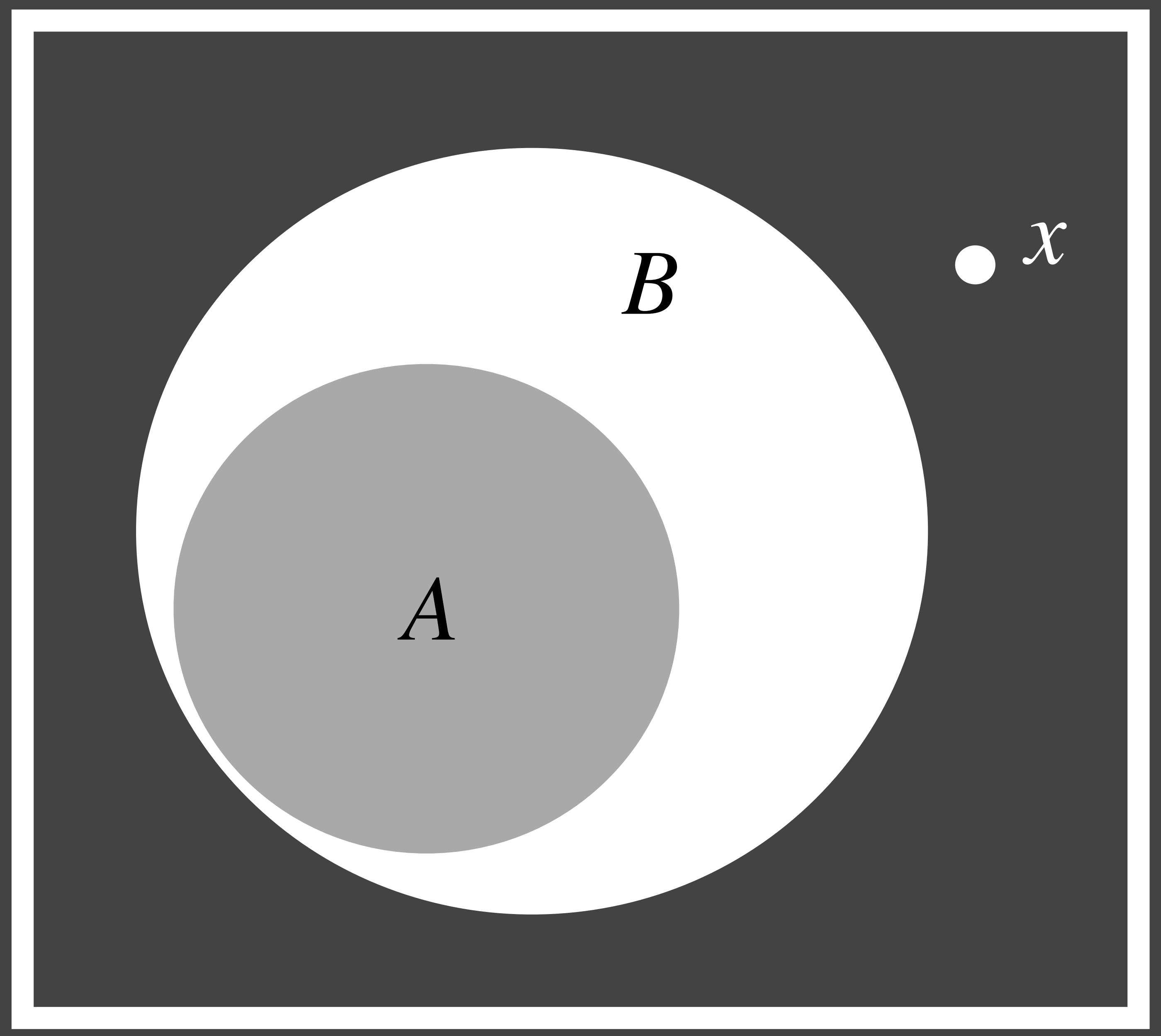


Submodularity

Definition: f is ***submodular*** *if,* $\forall A \subseteq B, x \notin B,$

$$f(A+x) - f(A) \geq f(B+x) - f(B)$$

$$f(x \mid A) \geq f(x \mid B)$$

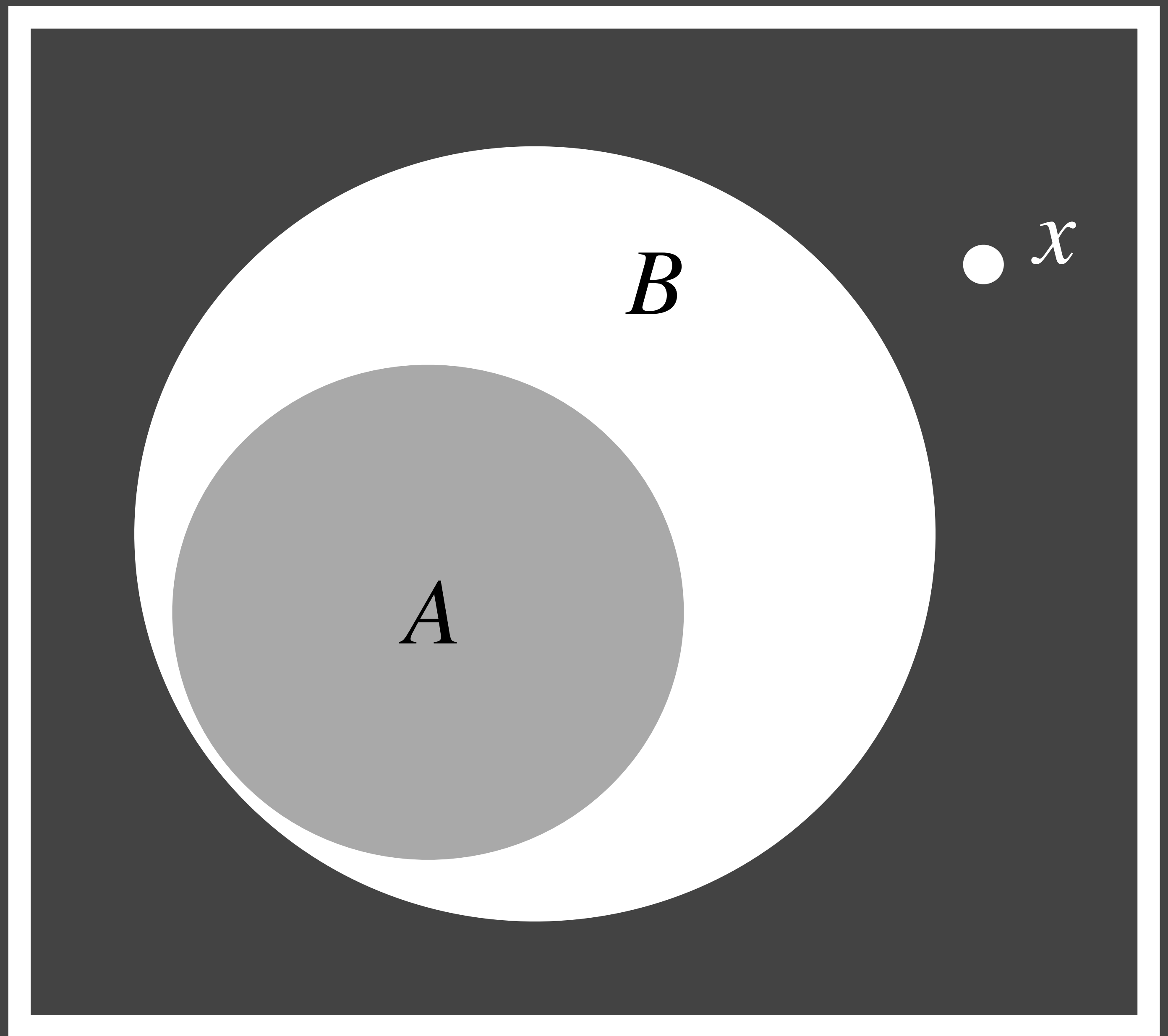


Submodularity

Definition: f is ***submodular*** if, $\forall A \subseteq B, x \notin B,$

$$f(A + x) - f(A) \geq f(B + x) - f(B)$$

$$f(x \mid A) \geq f(x \mid B)$$



$f($🍕$\mid$🥕$) \geq f($🍕$\mid$🥕, 🍩$)$

Why should we care about Submodular Cover?

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover
- Sensor Placement/ Robot Exploration
- Resource Allocation
- Influence Maximization in Social Networks

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover
- Sensor Placement/ Robot Exploration
- Resource Allocation
- Influence Maximization in Social Networks
- Feature Selection
- Document Summarization

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover
- Sensor Placement/ Robot Exploration
- Resource Allocation
- Influence Maximization in Social Networks
- Feature Selection
- Document Summarization

Popular to reduce to Submodular Cover!

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover
- Sensor Placement/ Robot Exploration
- Resource Allocation
- Influence Maximization in Social Networks
- Feature Selection
- Document Summarization

Popular to reduce to Submodular Cover!

[Goyal+ 13][Loukides Gwadera 16][Zheng+ 17][Andreev+ 09][Lee+ 13][Lukovszki+ 18][Poularakis+ 17][Krause+ 08][Korsarz Nutov 15][Jorgensen+ 17][Chen+ 18][Beinhofer+ 13][Tzoumas+ 16][Tong+ 17][Liu+ 16][Mafuta Walingo 16][Yang+ 15][Rahimian Preciado 15][Izumi+ 10][Wu+ 15], etc...

Why should we care about Submodular Cover?

Highly expressive! Examples of Submodular Cover:

- Set Cover (Hitting Set)
- Partial Set Cover
- Capacitated Set Cover
- Sensor Placement/ Robot Exploration
- Resource Allocation
- Influence Maximization in Social Networks
- Feature Selection
- Document Summarization

Popular to reduce to Submodular Cover!

[Goyal+ 13][Loukides Gwadera 16][Zheng+ 17][Andreev+ 09][Lee+ 13][Lukovszki+ 18][Poularakis+ 17][Krause+ 08][Korsarz Nutov 15][Jorgensen+ 17][Chen+ 18][Beinhofer+ 13][Tzoumas+ 16][Tong+ 17][Liu+ 16][Mafuta Walingo 16][Yang+ 15][Rahimian Preciado 15][Izumi+ 10][Wu+ 15], etc...

Porting submod cover to uncertain settings automatically ports all applications!

Why should we care about Submodular Cover?

Why should we care about Submodular Cover?

Also, can approximate efficiently! Greedy gives $\log f(\mathcal{N}) + 1$ approx [Wolsey 82].

Why should we care about Submodular Cover?

Also, can approximate efficiently! Greedy gives $\log f(\mathcal{N}) + 1$ approx [Wolsey 82].

$\Rightarrow \log n + 1$ approx for Set Cover.

Why should we care about Submodular Cover?

Also, can approximate efficiently! Greedy gives $\log f(\mathcal{N}) + 1$ approx [Wolsey 82].

$\Rightarrow \log n + 1$ approx for Set Cover.

Optimal in poly time, unless P=NP [Feige 98][Dinur Steurer 14].

Why should we care about Submodular Cover?

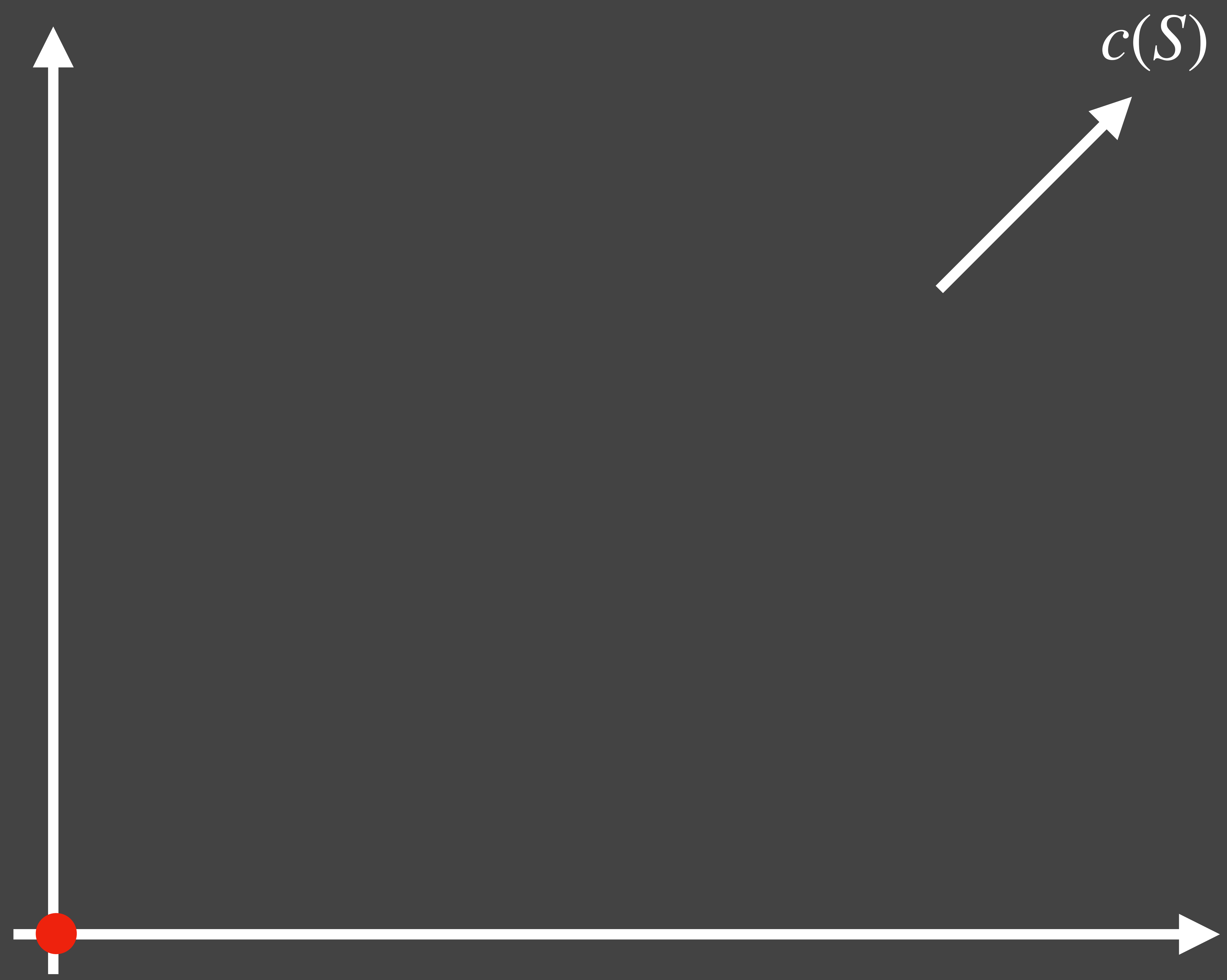
Also, can approximate efficiently! Greedy gives $\log f(\mathcal{N}) + 1$ approx [Wolsey 82].

$\Rightarrow \log n + 1$ approx for Set Cover.

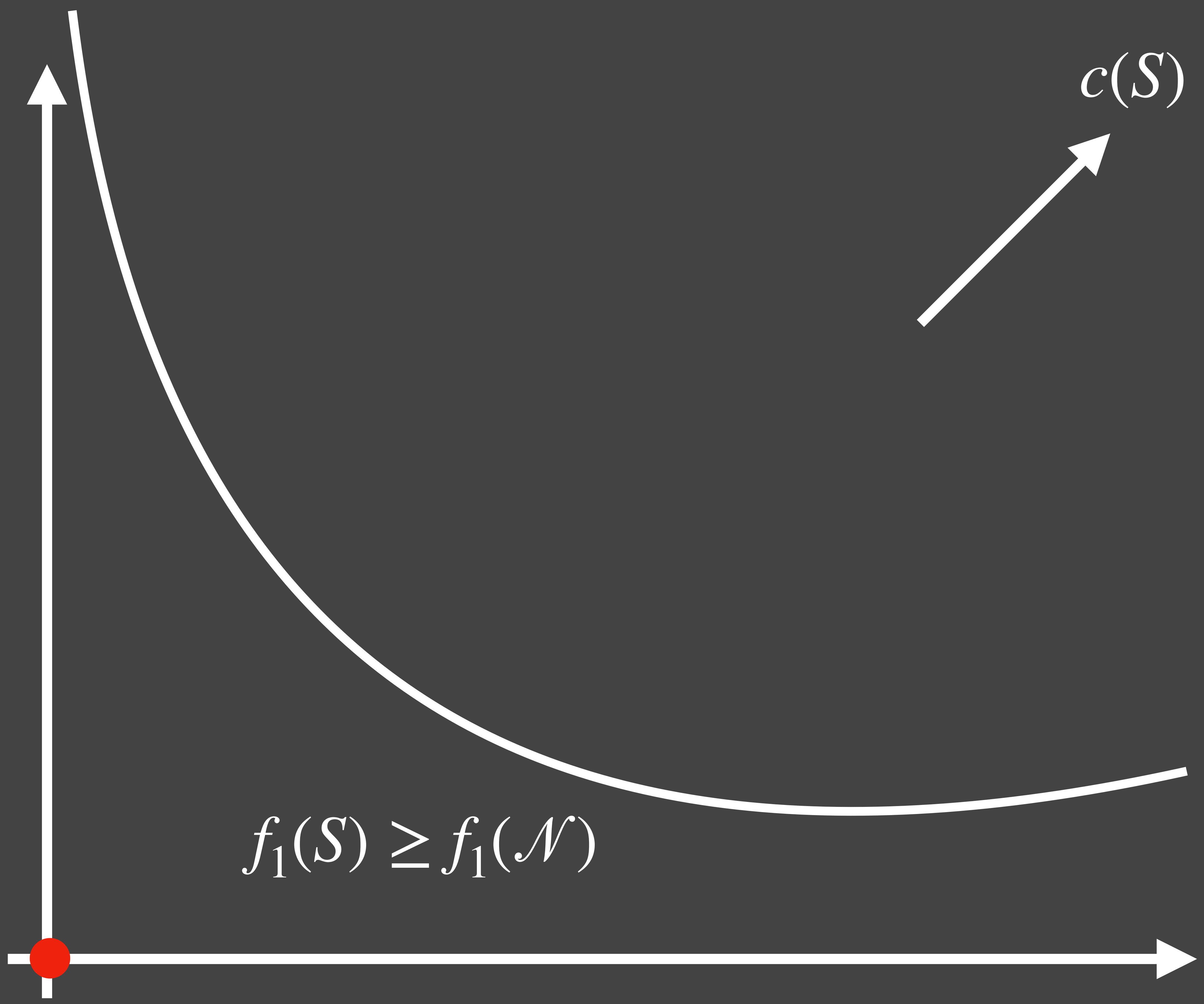
Optimal in poly time, unless P=NP [Feige 98][Dinur Steurer 14].

Sweet spot between generality and tractability!

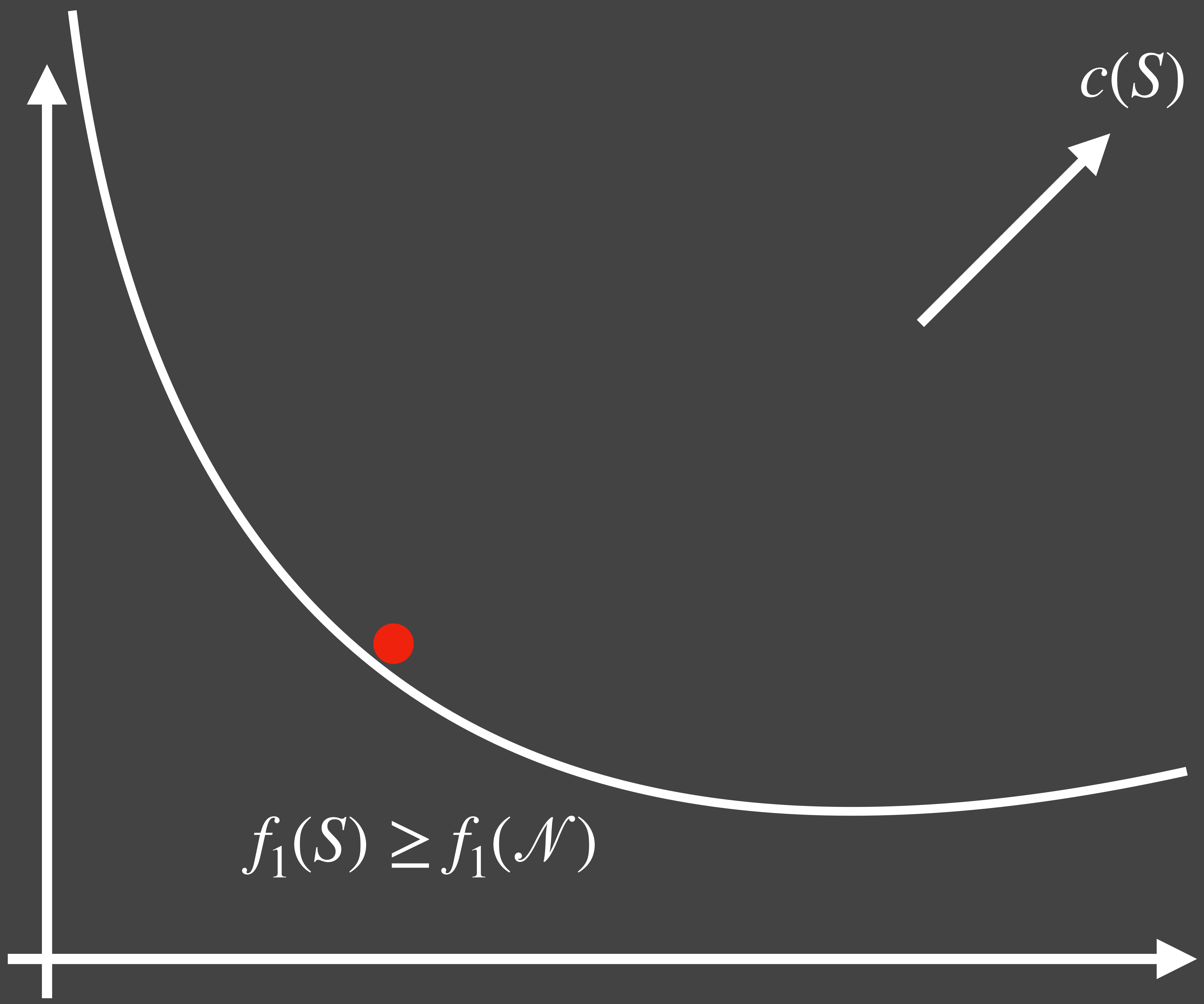
Online/Dynamic Submodular Cover



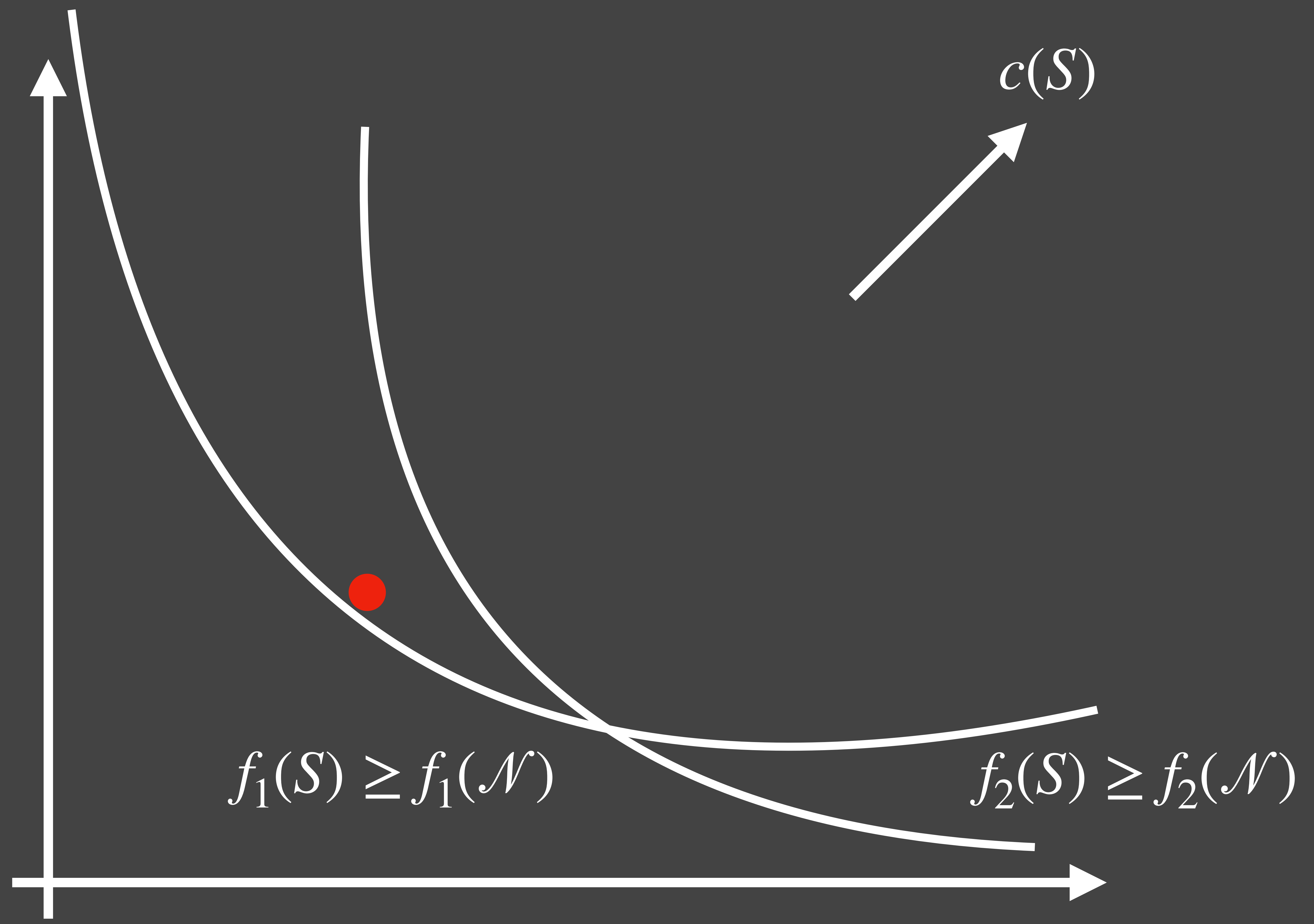
Online/Dynamic Submodular Cover



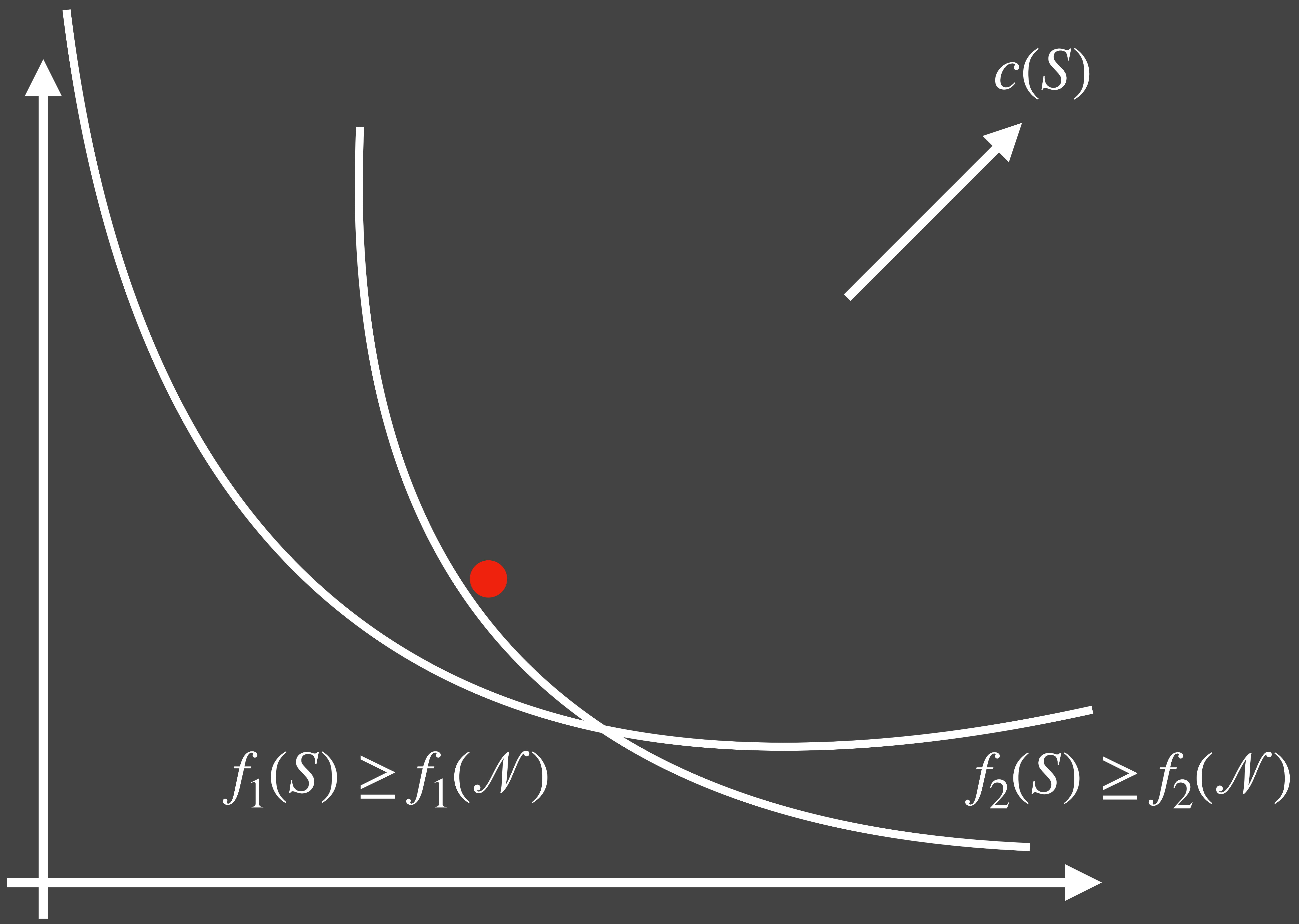
Online/Dynamic Submodular Cover



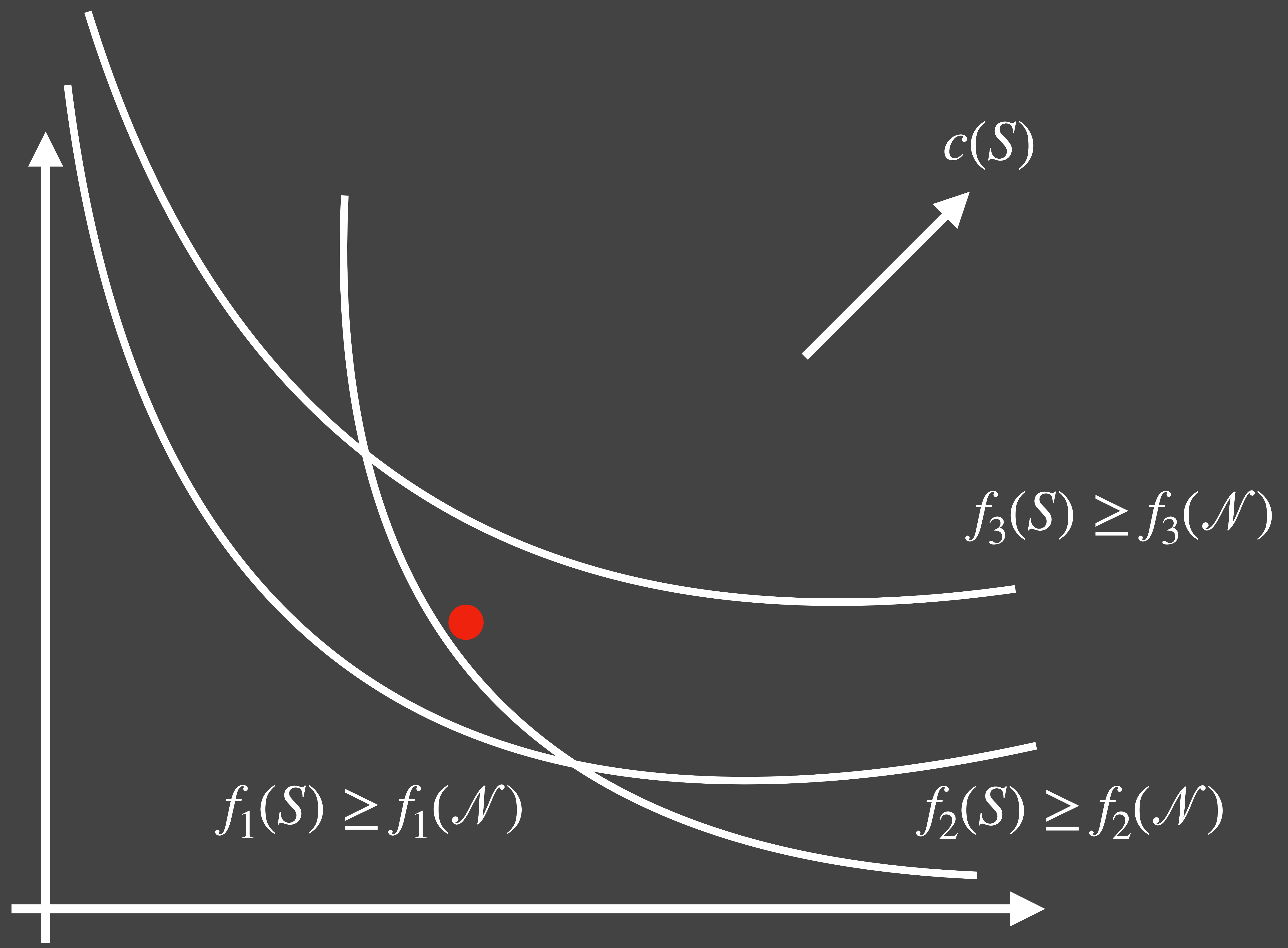
Online/Dynamic Submodular Cover



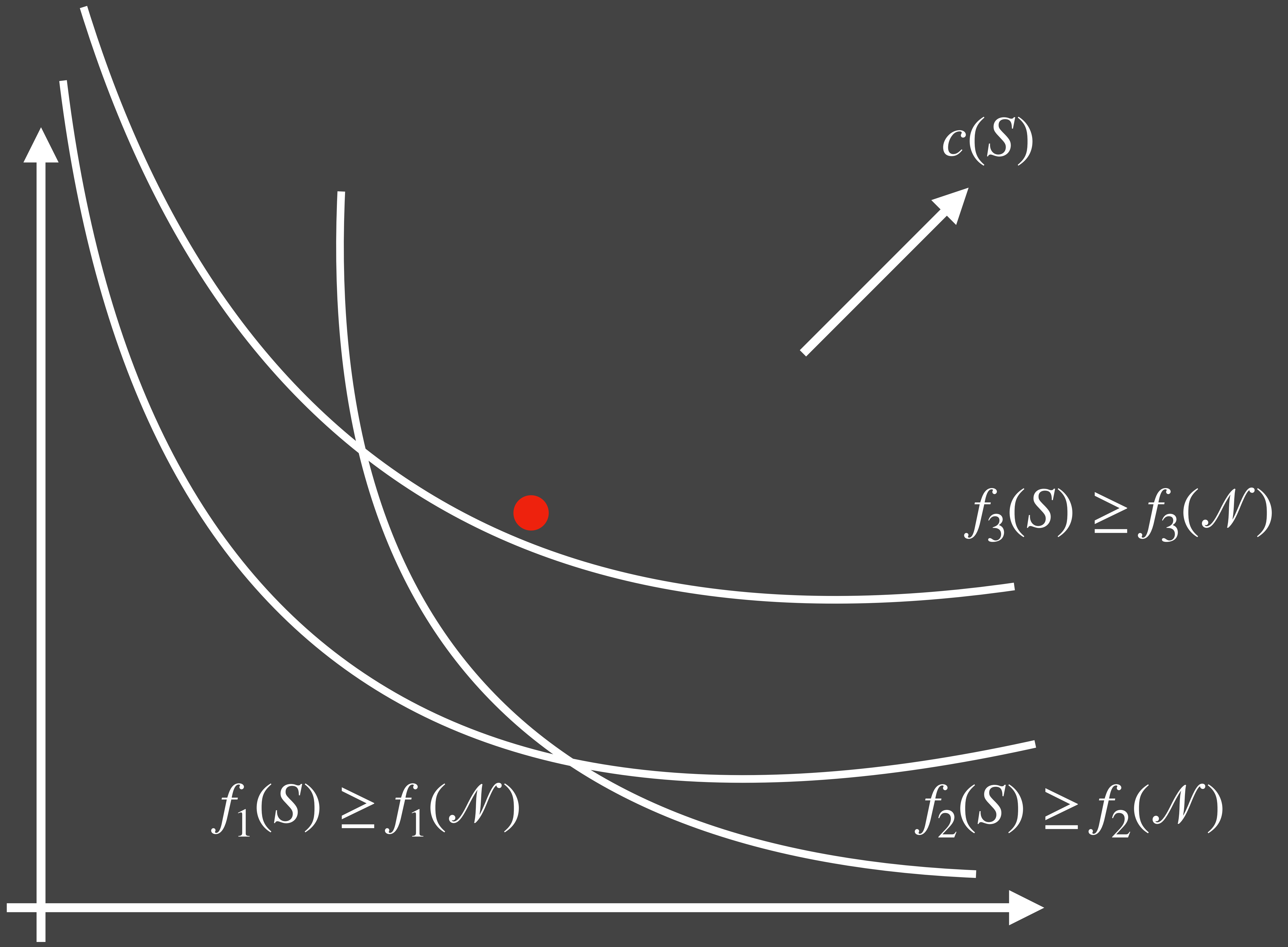
Online/Dynamic Submodular Cover



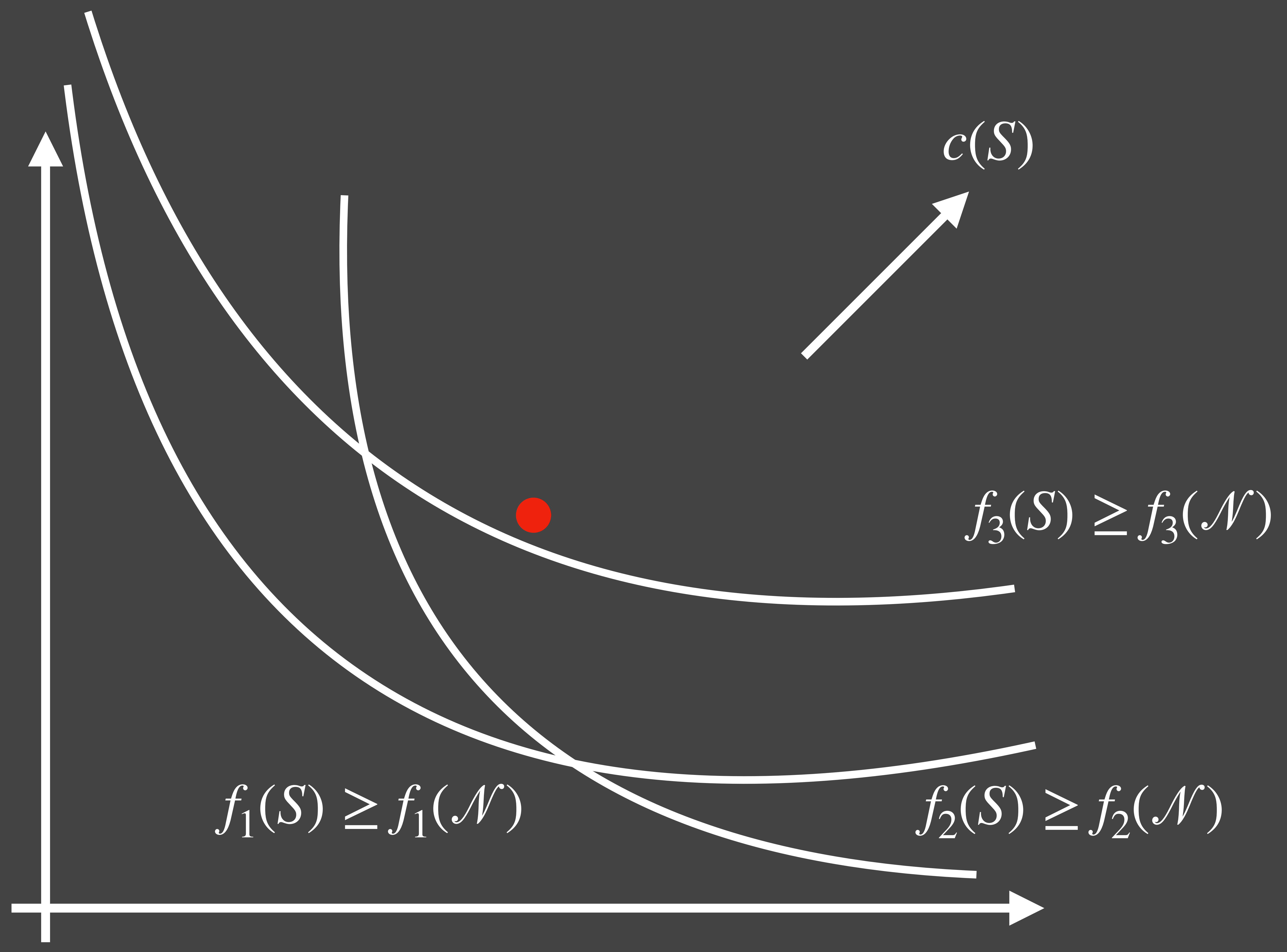
Online/Dynamic Submodular Cover



Online/Dynamic Submodular Cover

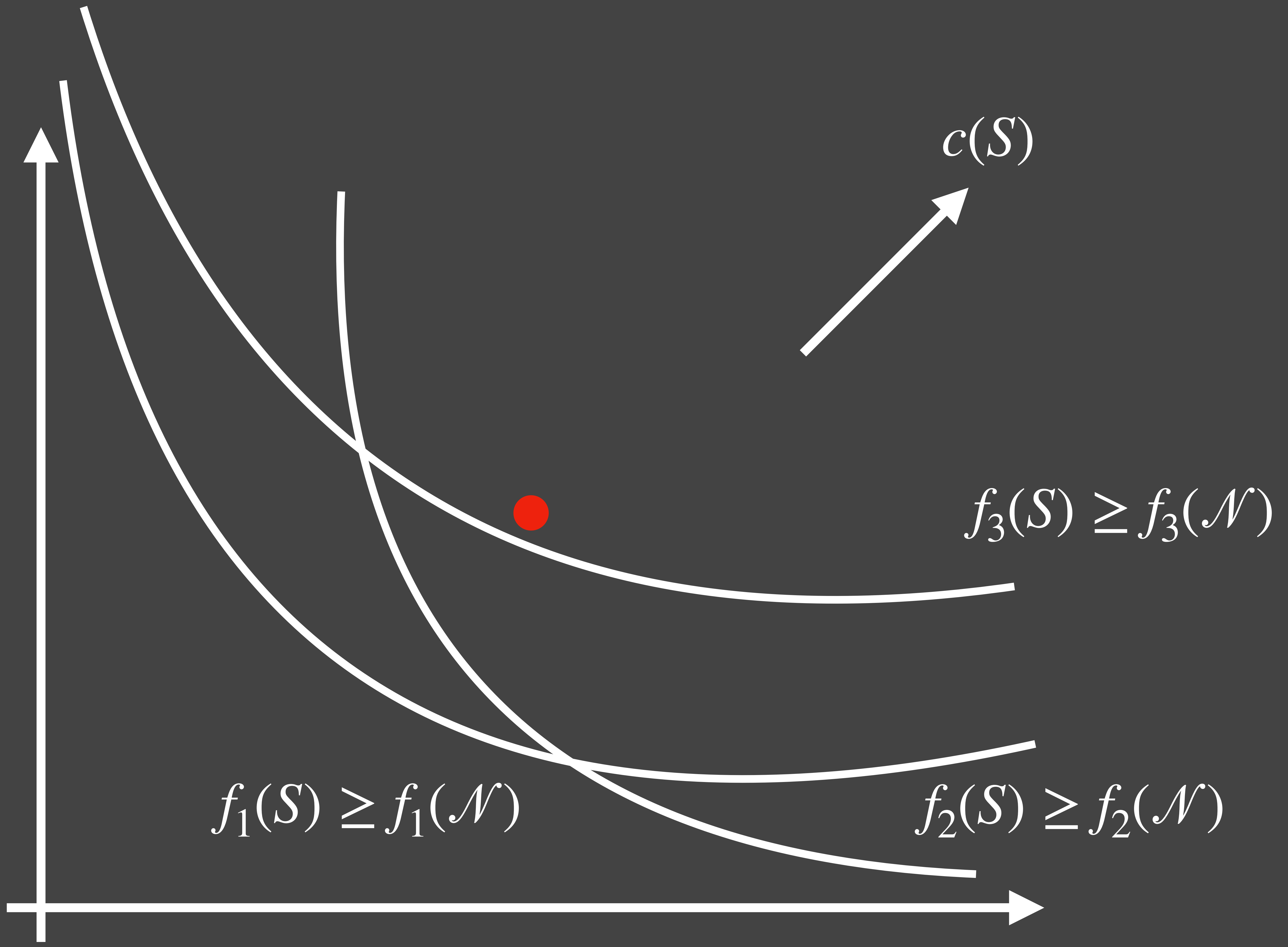


Online/Dynamic Submodular Cover



$$F = \sum_i f_i$$

Online/Dynamic Submodular Cover



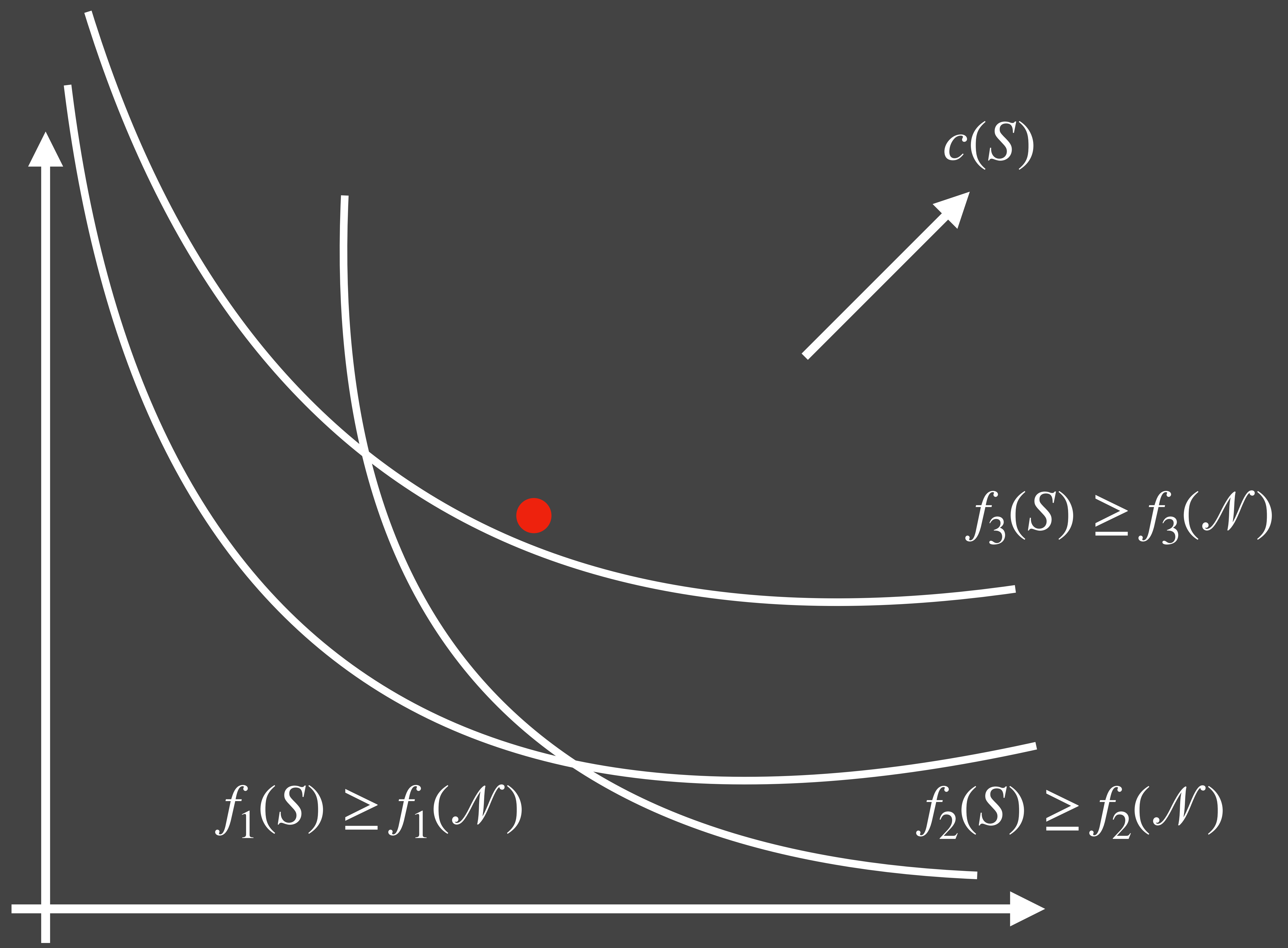
$$F = \sum_i f_i$$

$$\min_{S \subseteq \mathcal{N}} c(S)$$

$$F(S) \geq F(\mathcal{N})$$

$$S \in \{0,1\}^m$$

Online/Dynamic Submodular Cover



$$F = \sum_i f_i$$

$$\min_{S \subseteq \mathcal{N}} c(S)$$

$$F(S) \geq F(\mathcal{N})$$

$$S \in \{0,1\}^m$$

<u>This talk</u>: $f_i(\mathcal{N}) = f(\mathcal{N})$, all same.

My PhD Work

Online	Dynamic	Streaming
The Online Submodular Cover Problem [Gupta, L., SODA 20]	Fully-Dynamic Submodular Cover with Bounded Recourse [Gupta, L., FOCS 20]	Streaming Submodular Matching Meets the Primal Dual Method [L., Wajc, SODA 21]
Random Order Set Cover is as Easy as Offline [Gupta, Kehne, L., FOCS 21]		Robust Subspace Approximation in a Stream [L., Sevekari, Woodruff, NeurIPS 18]
Competitive Algorithms for Block-Aware Caching [Coester, Naor, L., Talmon, SPAA 22] New!		

… and Offline

Finding Skewed Subcubes Under a Distribution [Gopalan, L., Wieder ITCS 20]

$\in$ Thesis

$\notin$ Thesis

My PhD Work

Online	Dynamic	Streaming
The Online Submodular Cover Problem [Gupta, L., SODA 20]	Fully-Dynamic Submodular Cover with Bounded Recourse [Gupta, L., FOCS 20]	Streaming Submodular Matching Meets the Primal Dual Method [L., Wajc, SODA 21]
Random Order Set Cover is as Easy as Offline [Gupta, Kehne, L., FOCS 21]		Robust Subspace Approximation in a Stream [L., Sevekari, Woodruff, NeurIPS 18]
Competitive Algorithms for Block-Aware Caching [Coester, Naor, L., Talmon, SPAA 22] New!		

This Talk

… and Offline

Finding Skewed Subcubes Under a Distribution [Gopalan, L., Wieder ITCS 20]

$\in$ Thesis

$\notin$ Thesis

Talk Outline

➔ Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

Part III — Random Order Online Set Cover

Conclusion

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

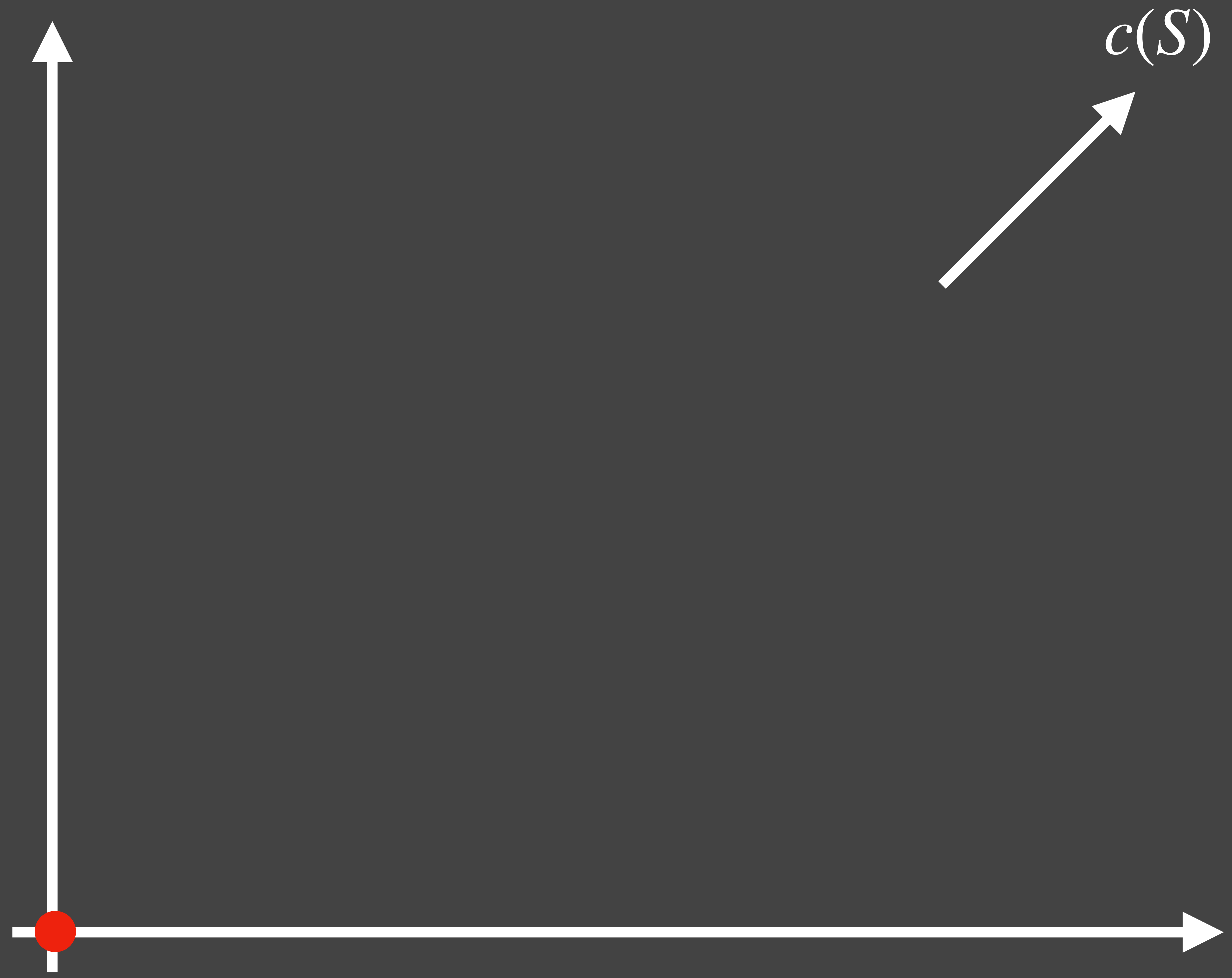
Part III — Random Order Online Set Cover

Conclusion

Part I — Online/Dynamic Submodular Cover

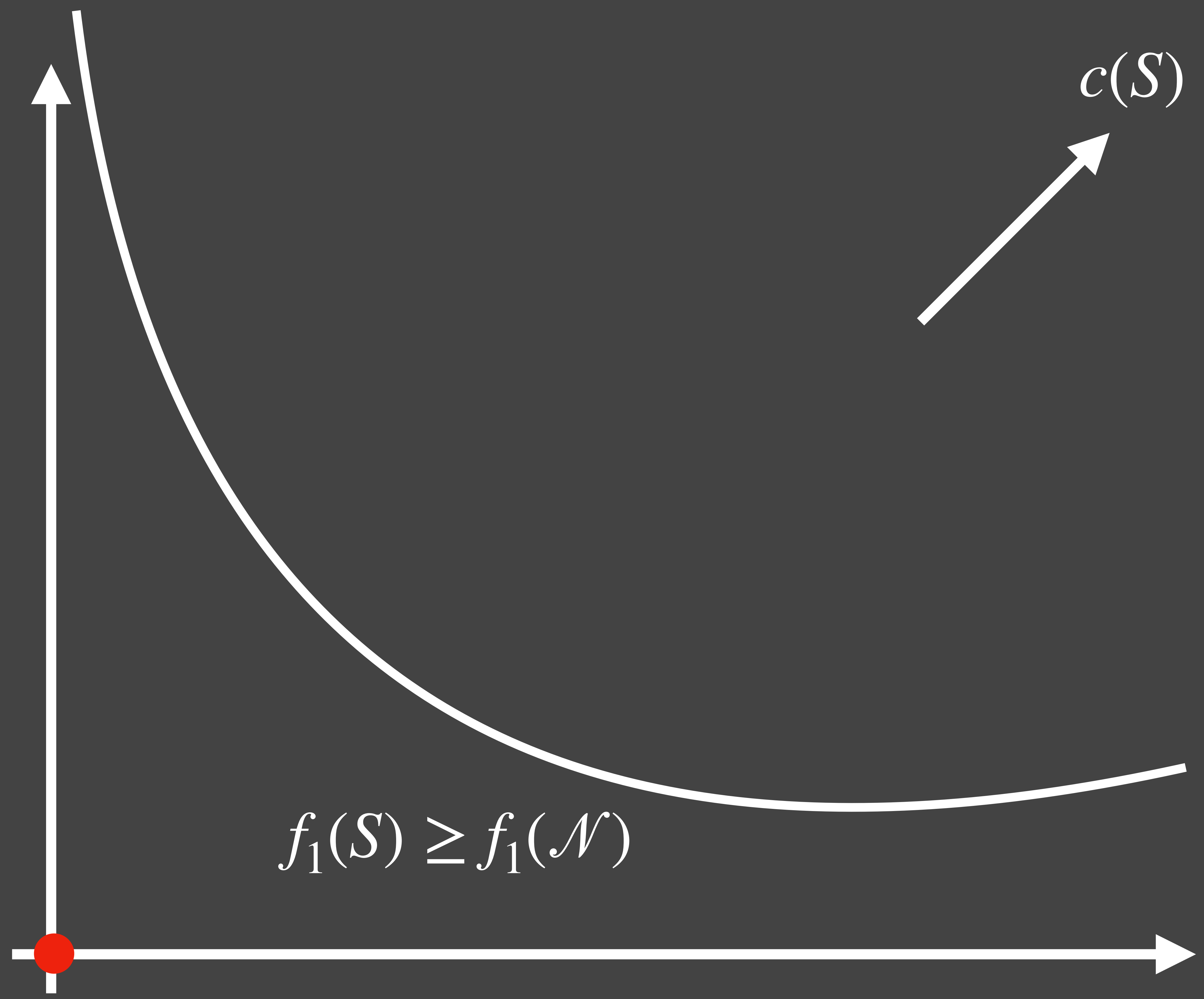
with Anupam Gupta

Online Submodular Cover



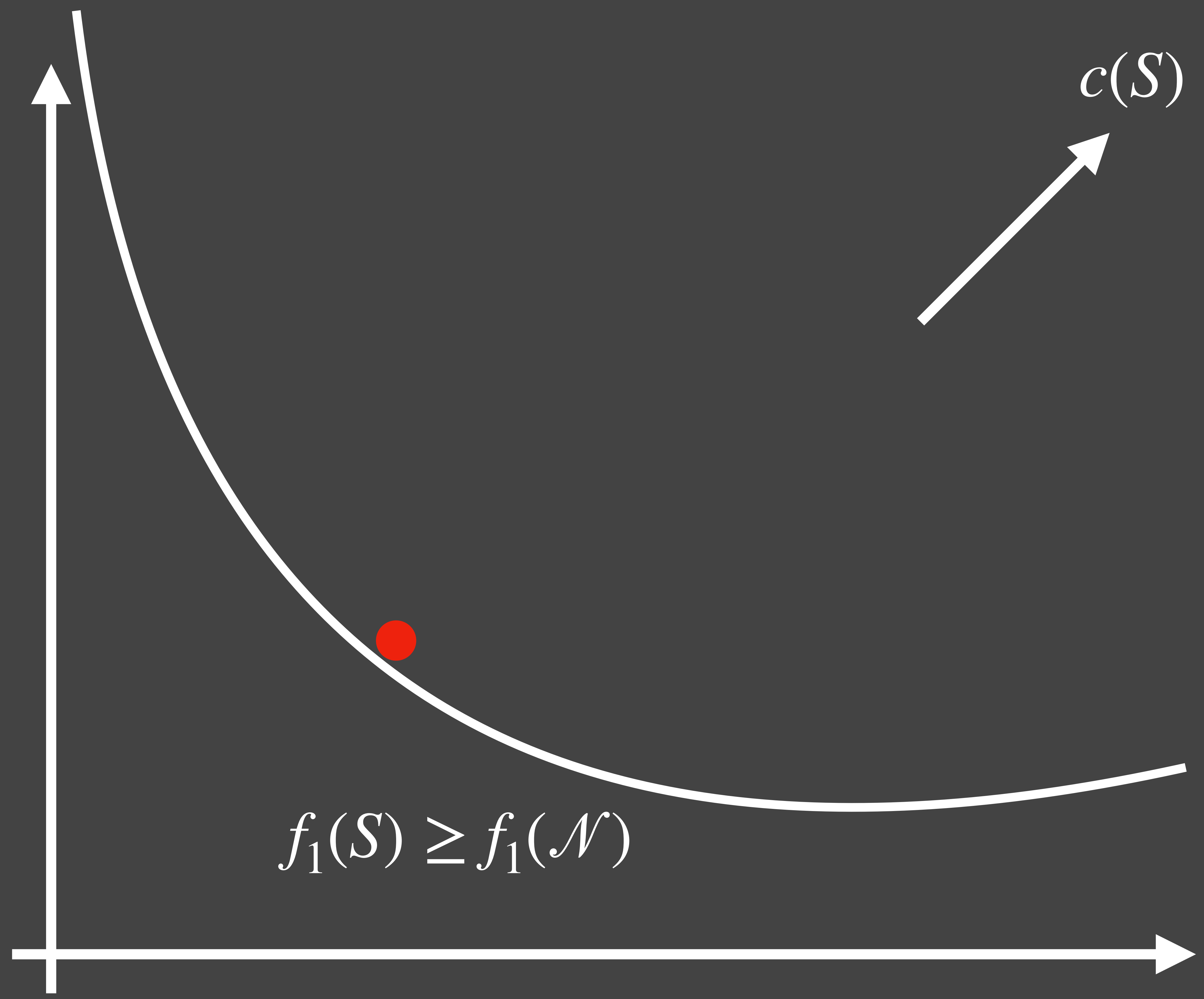
$$F = \sum_i f_i$$

Online Submodular Cover



$$F = \sum_i f_i$$

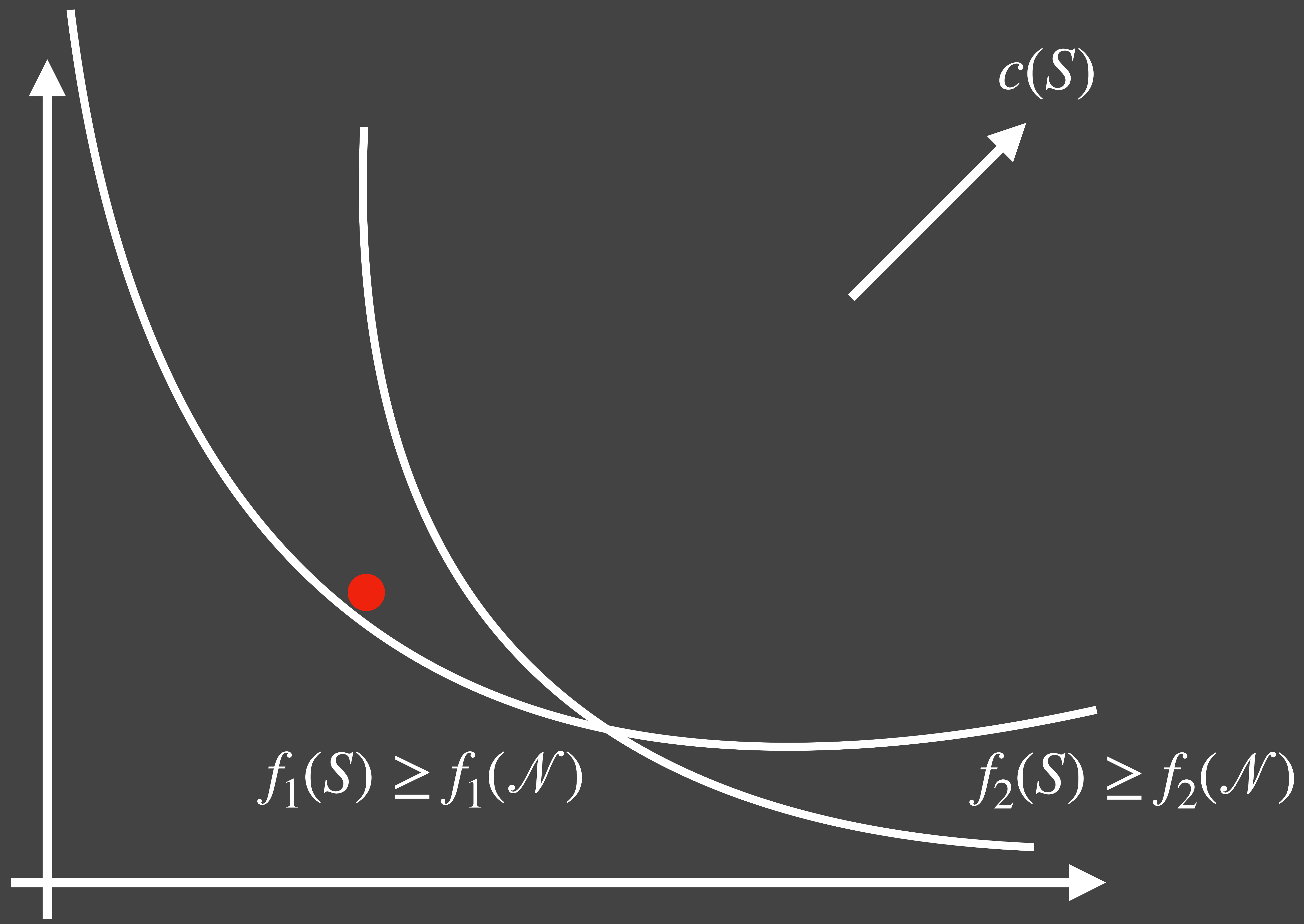
Online Submodular Cover



$c(S)$

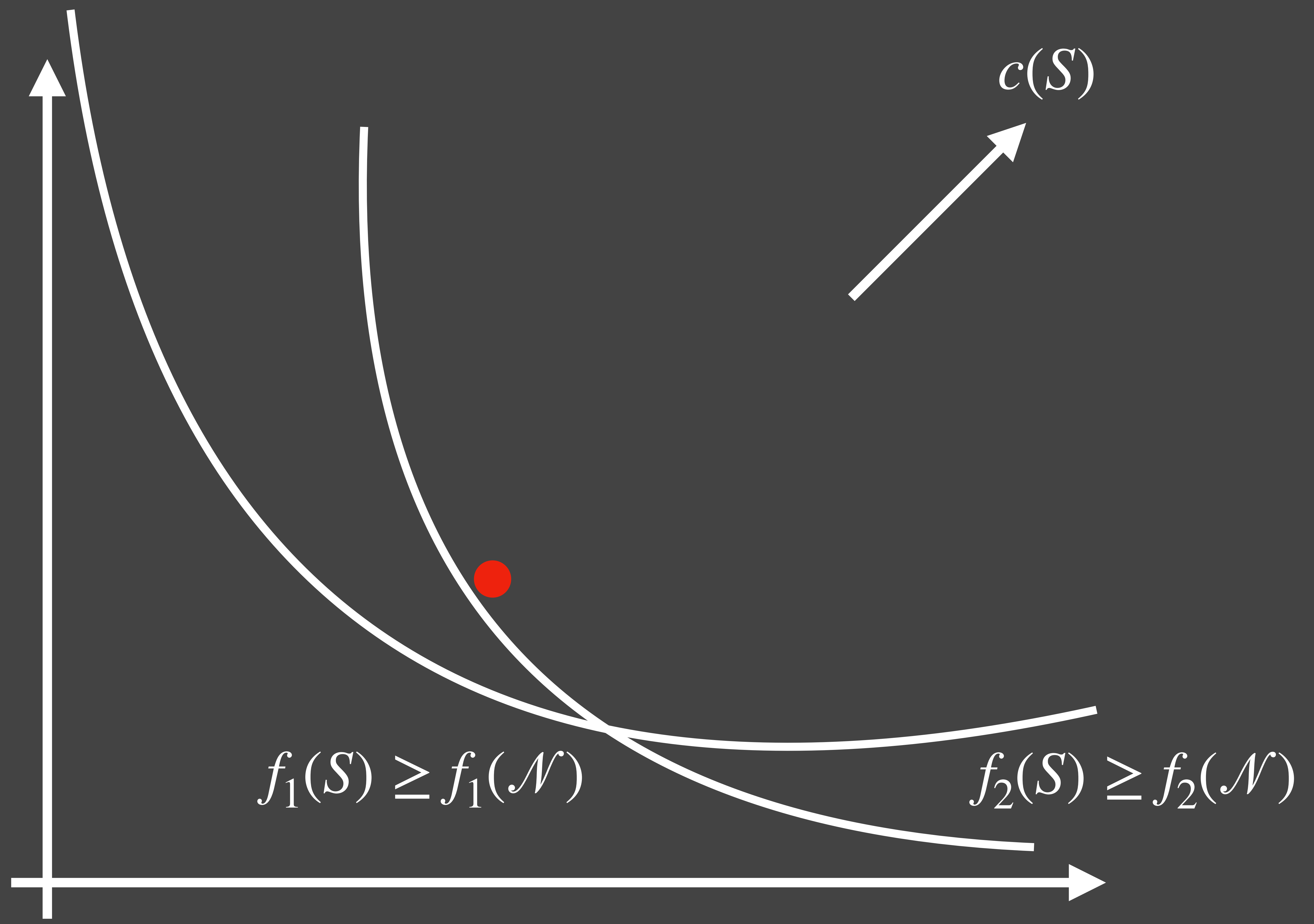
$$F = \sum_i f_i$$

Online Submodular Cover



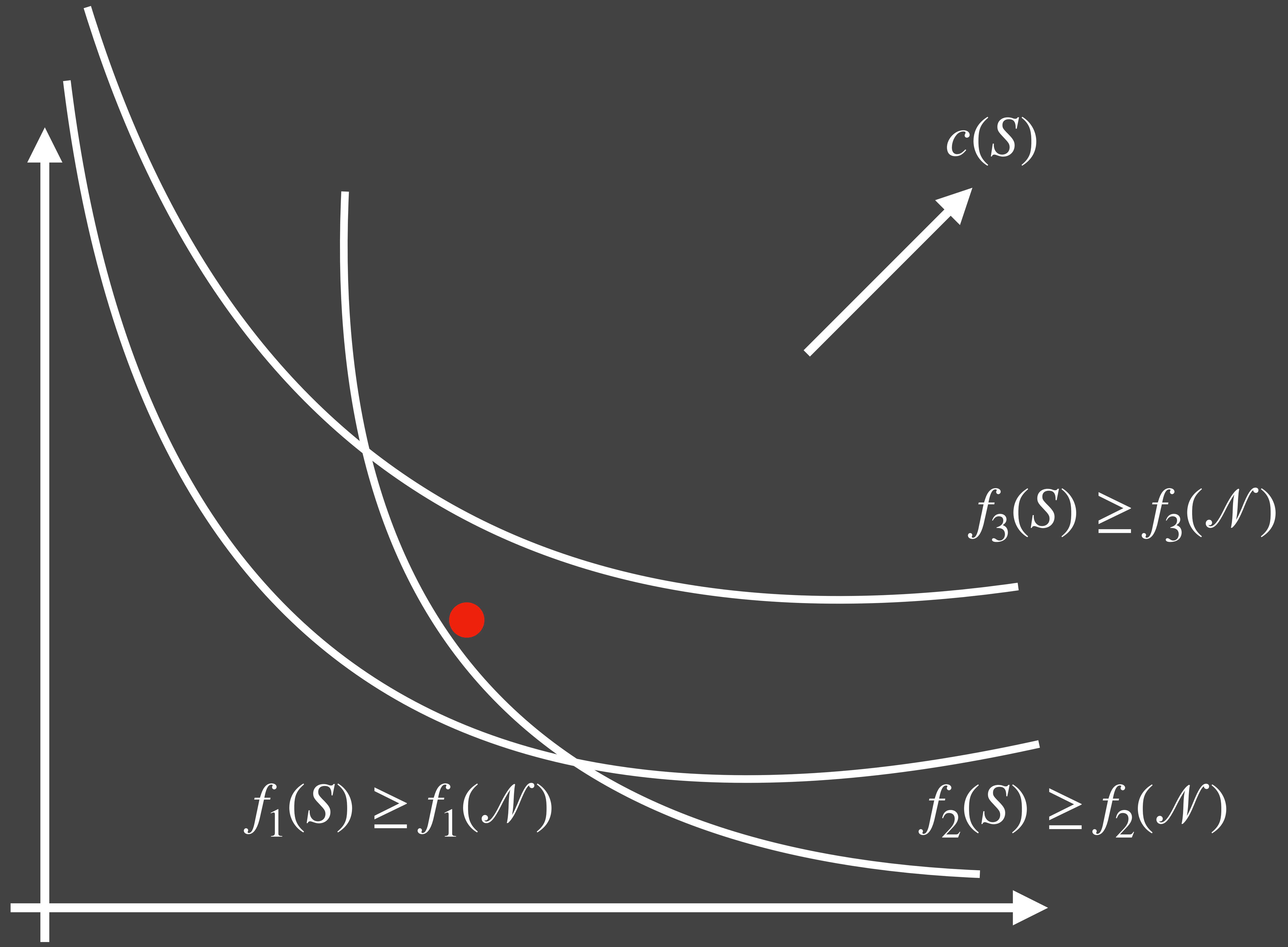
$$F = \sum_i f_i$$

Online Submodular Cover



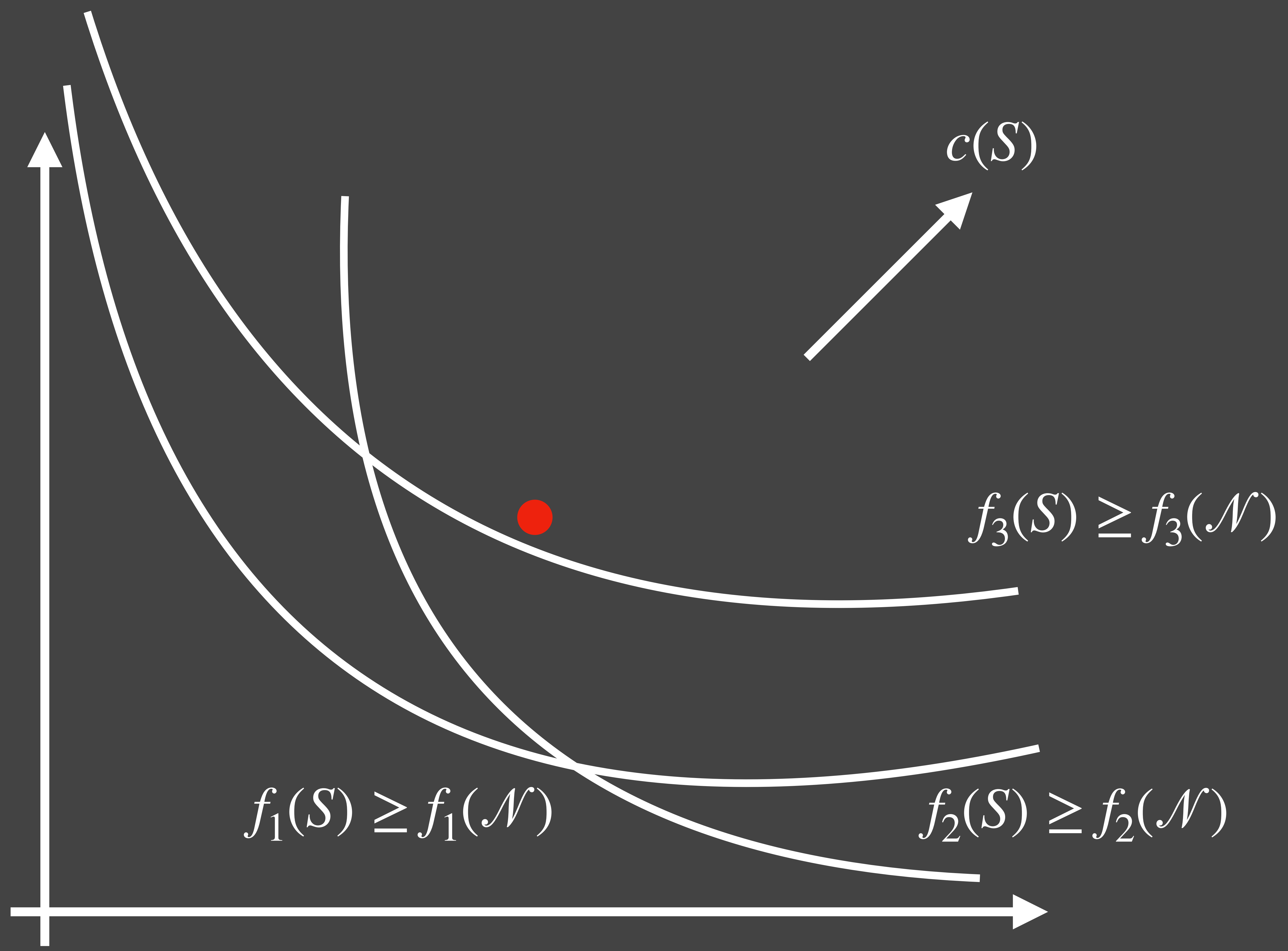
$$F = \sum_i f_i$$

Online Submodular Cover



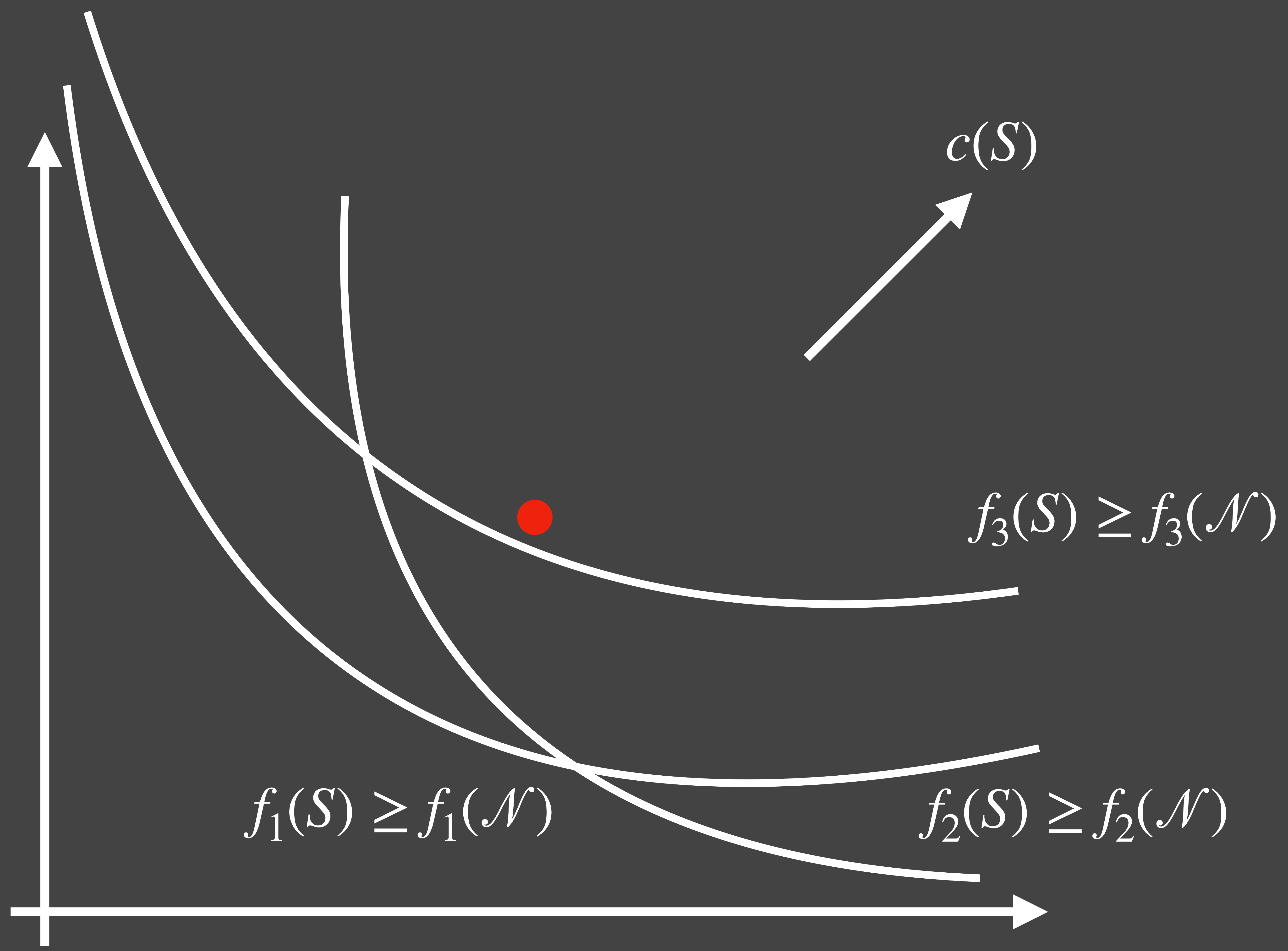
$$F = \sum_i f_i$$

Online Submodular Cover



$$F = \sum_i f_i$$

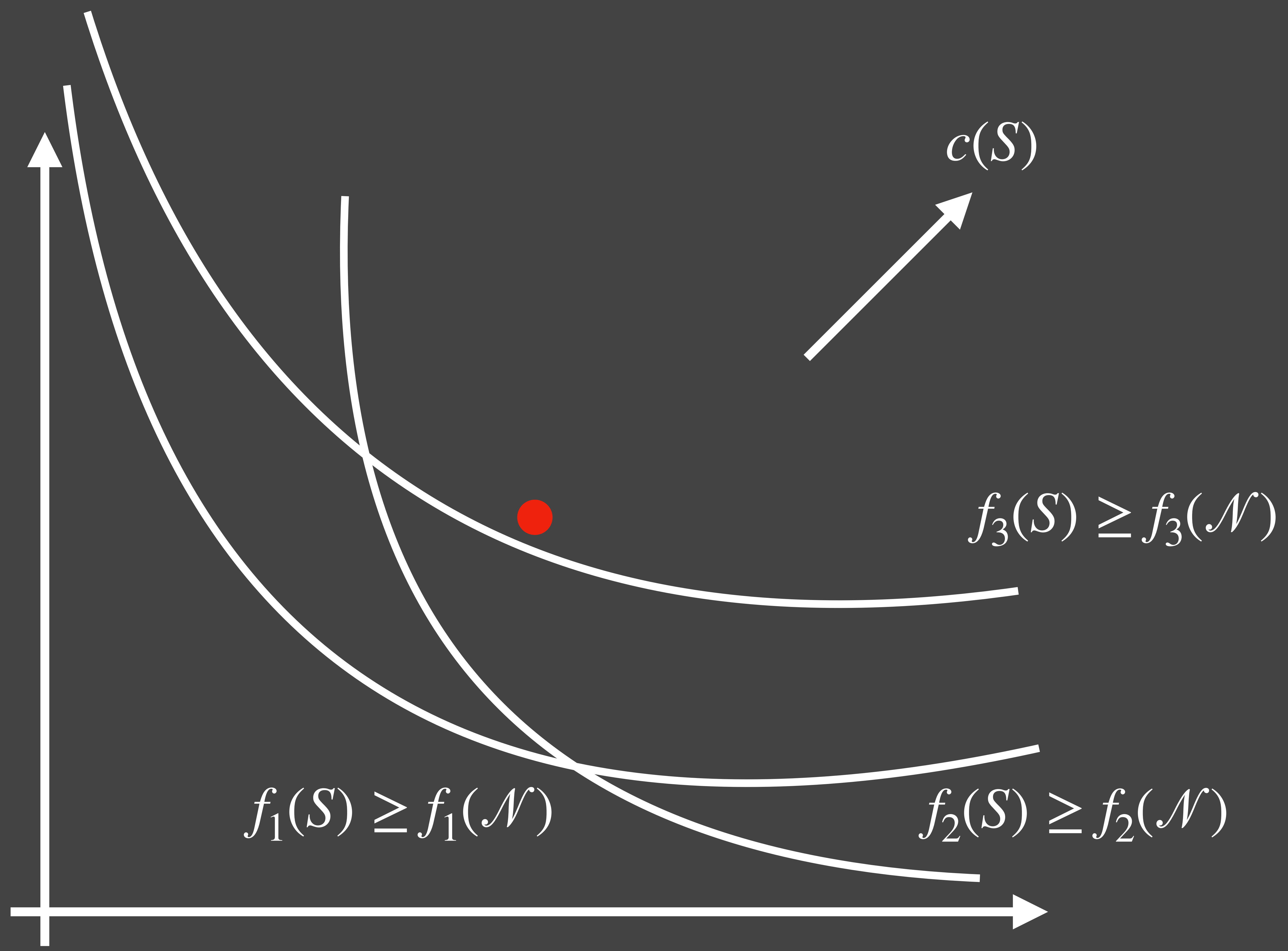
Online Submodular Cover



$$F = \sum_i f_i$$

Decisions are irrevocable!!

Online Submodular Cover



$$F = \sum_i f_i$$

Decisions are irrevocable!!

S can only grow over time...

Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]

s_1

s_2

s_3

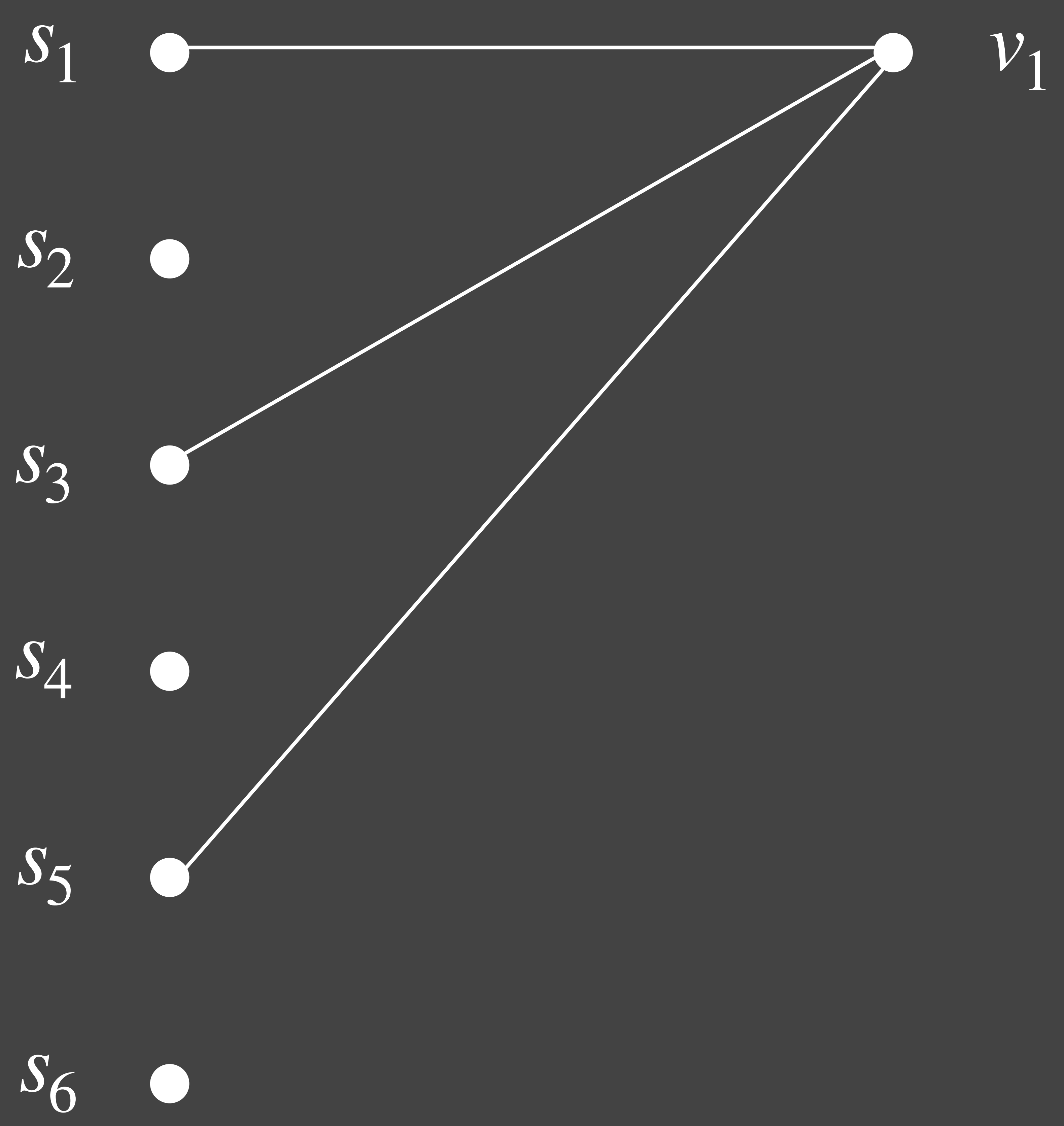
s_4

s_5

s_6

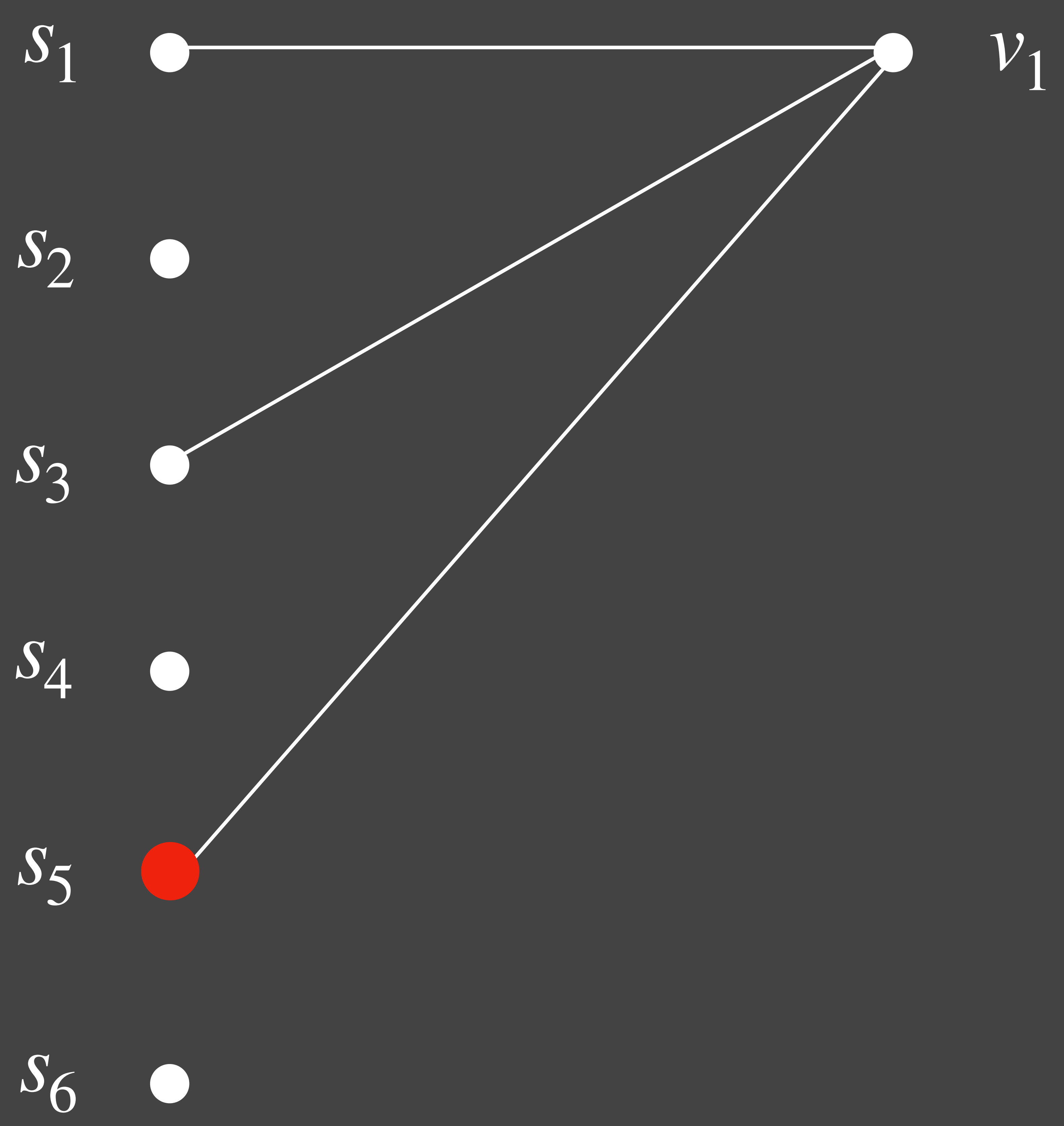
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



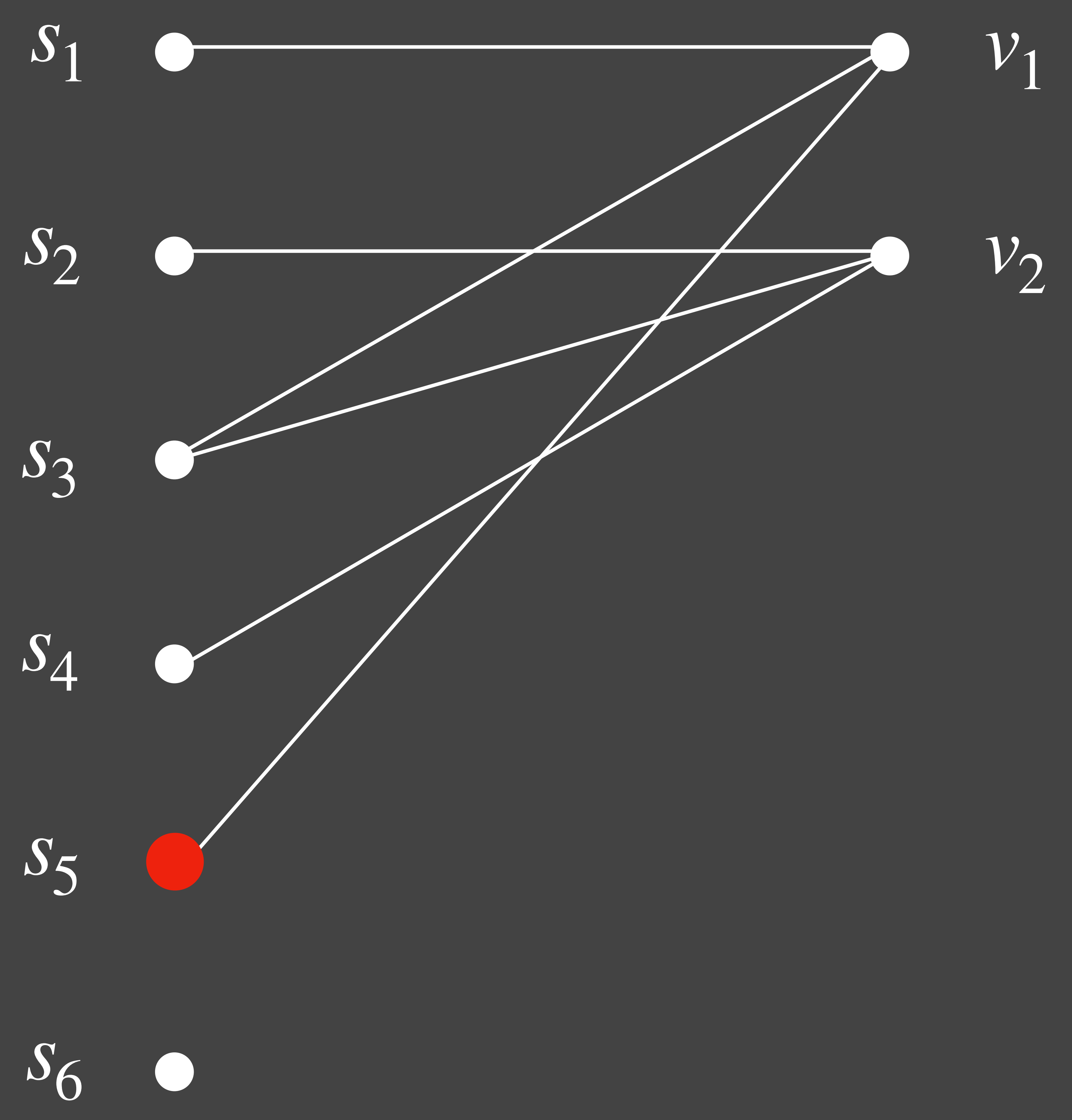
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



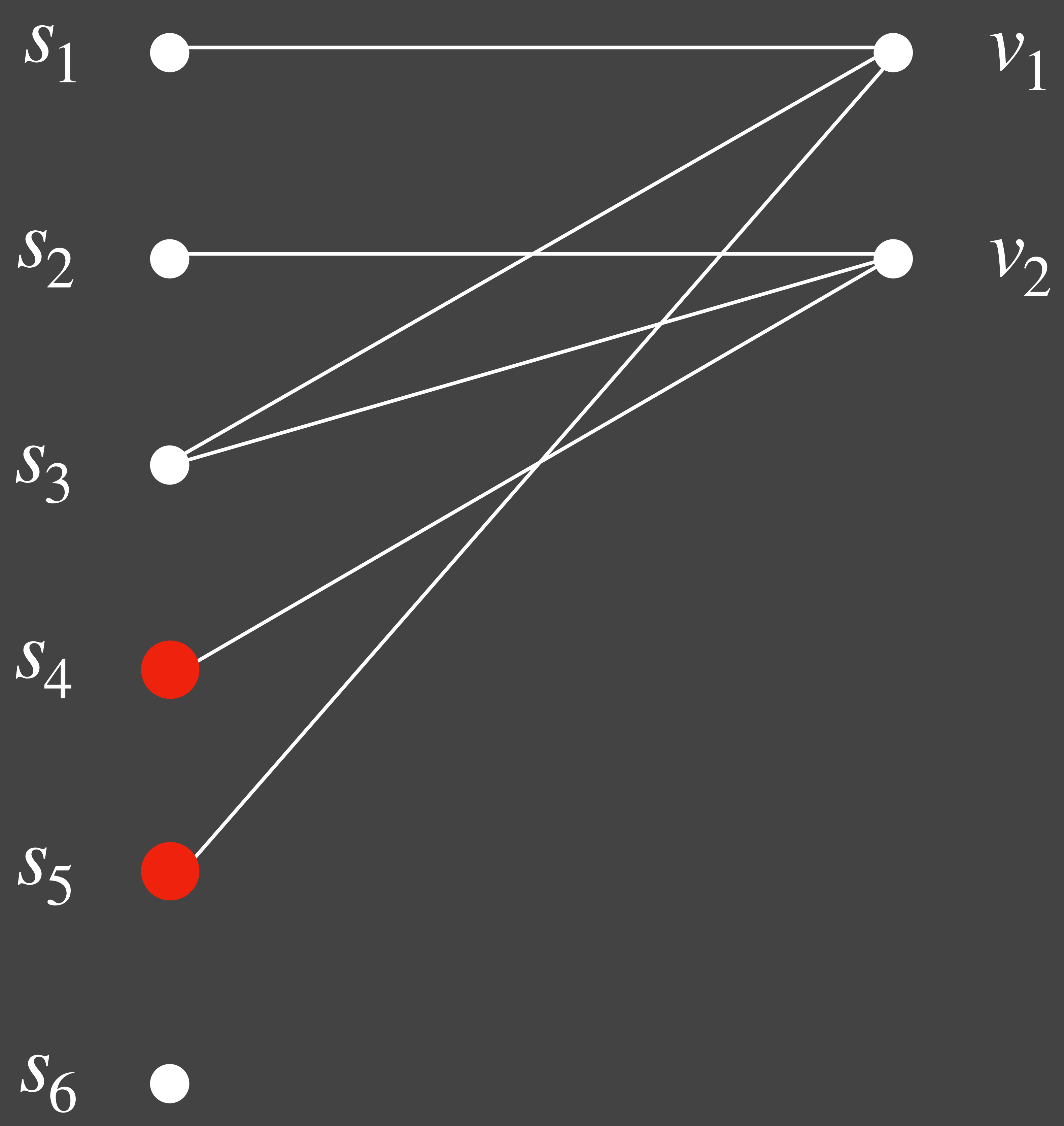
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



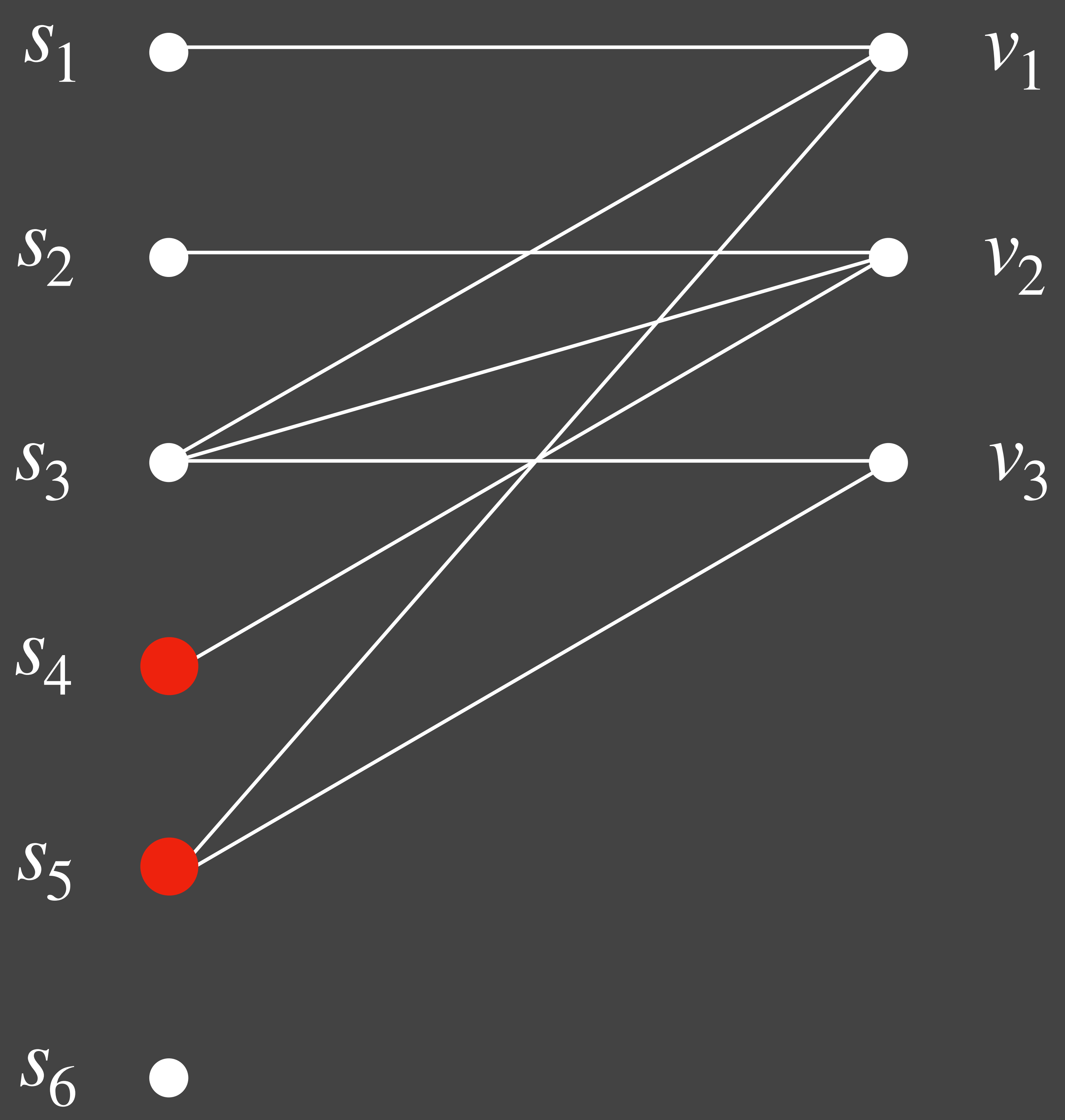
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



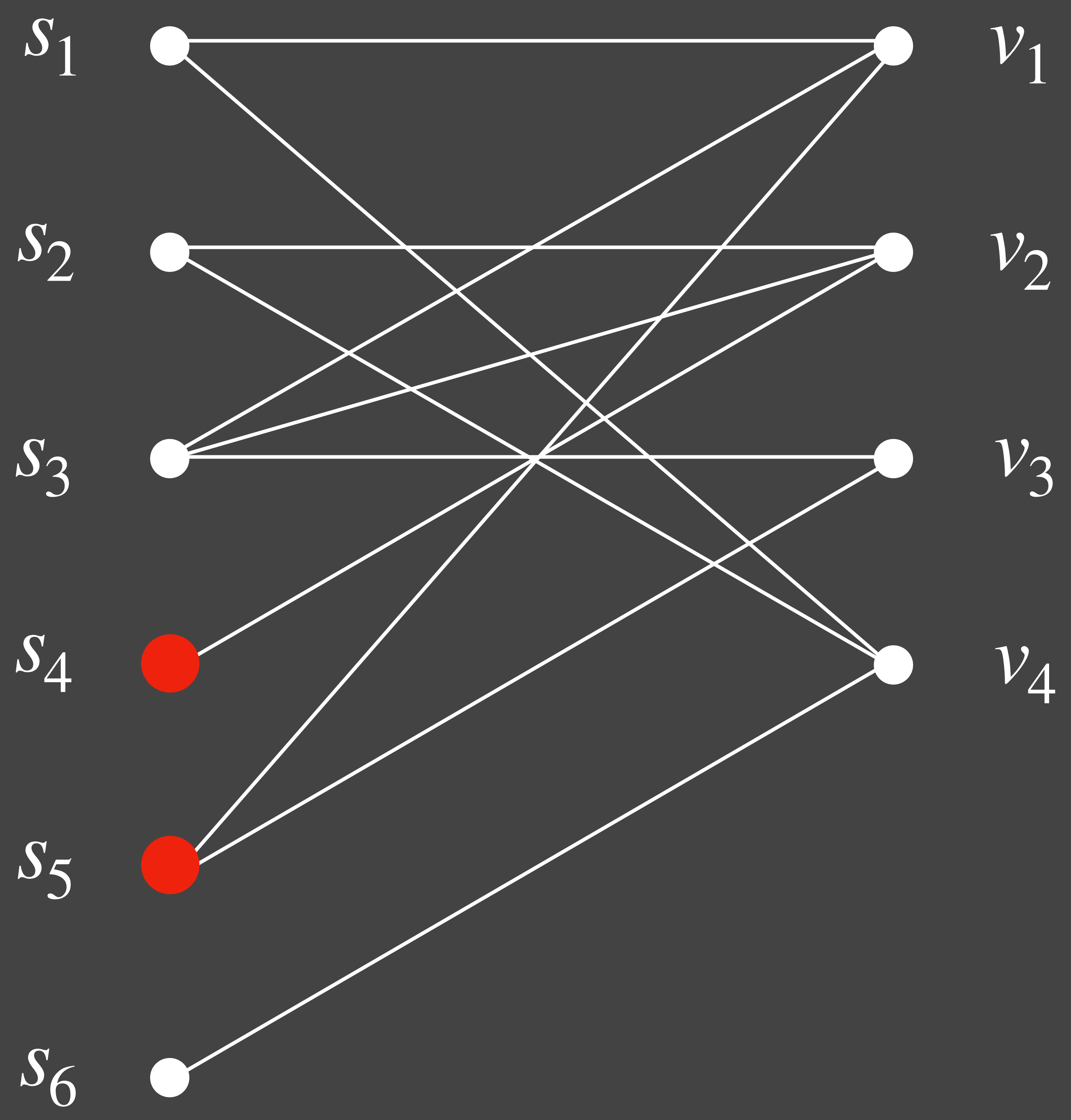
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



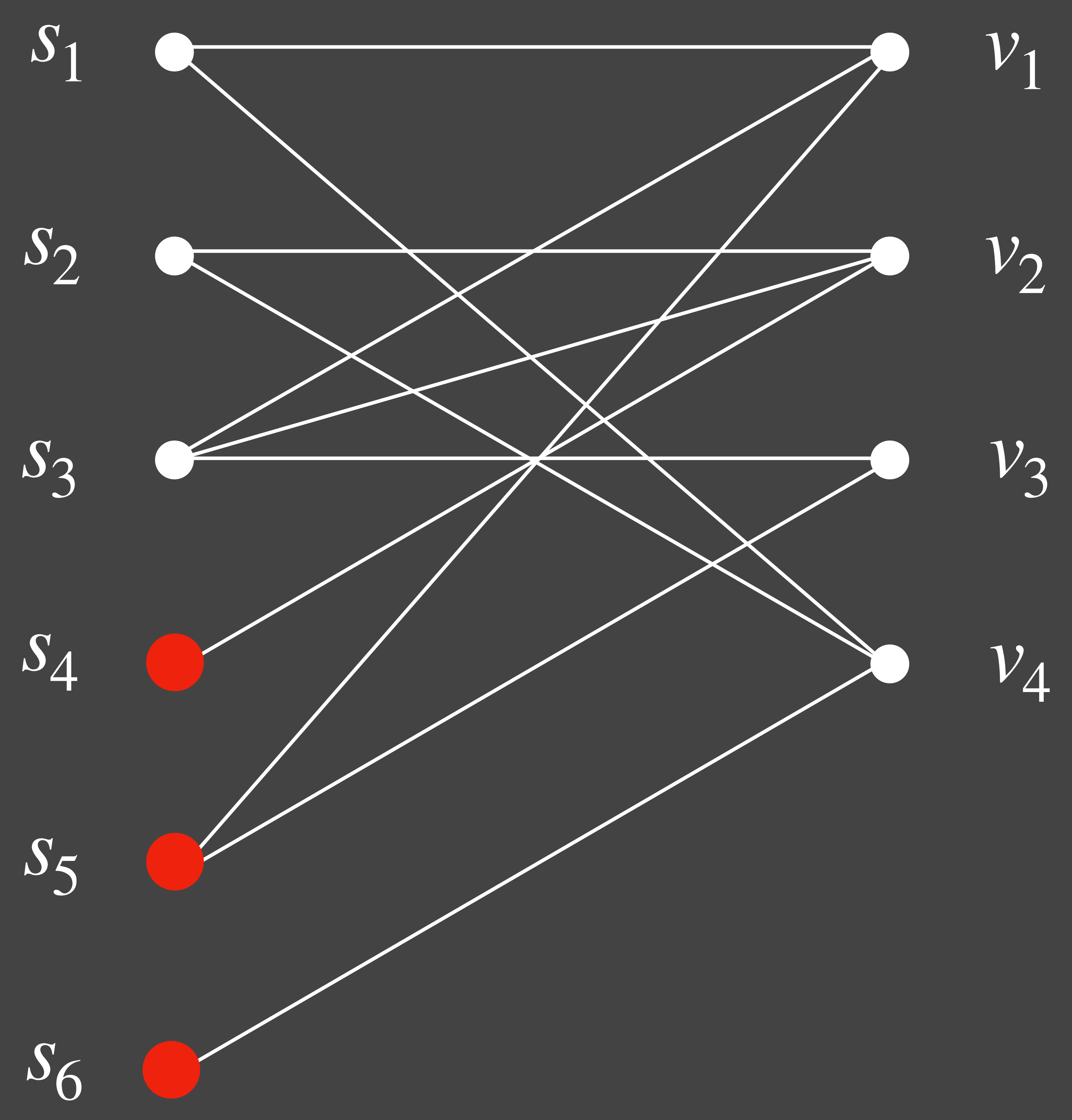
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



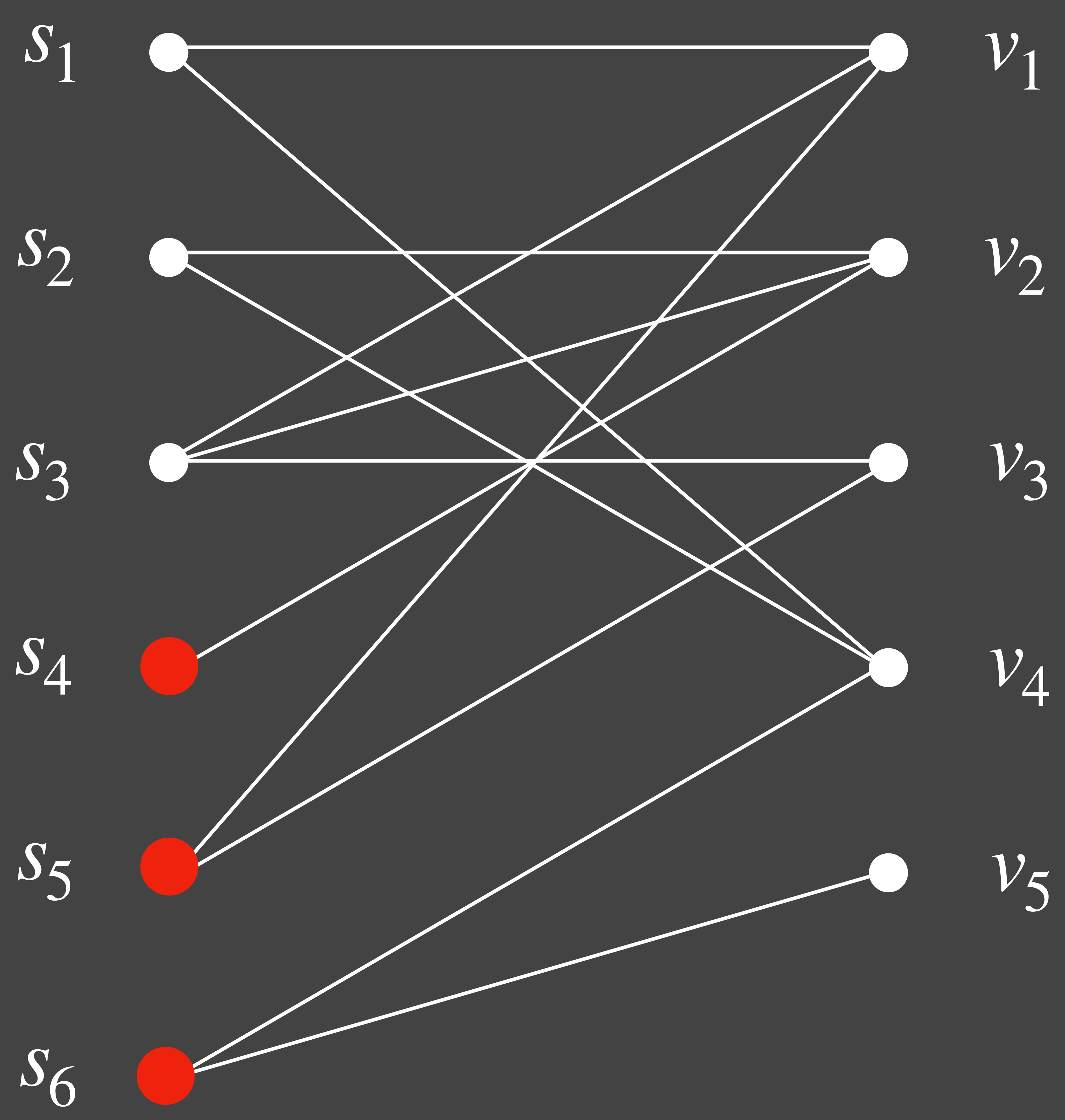
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



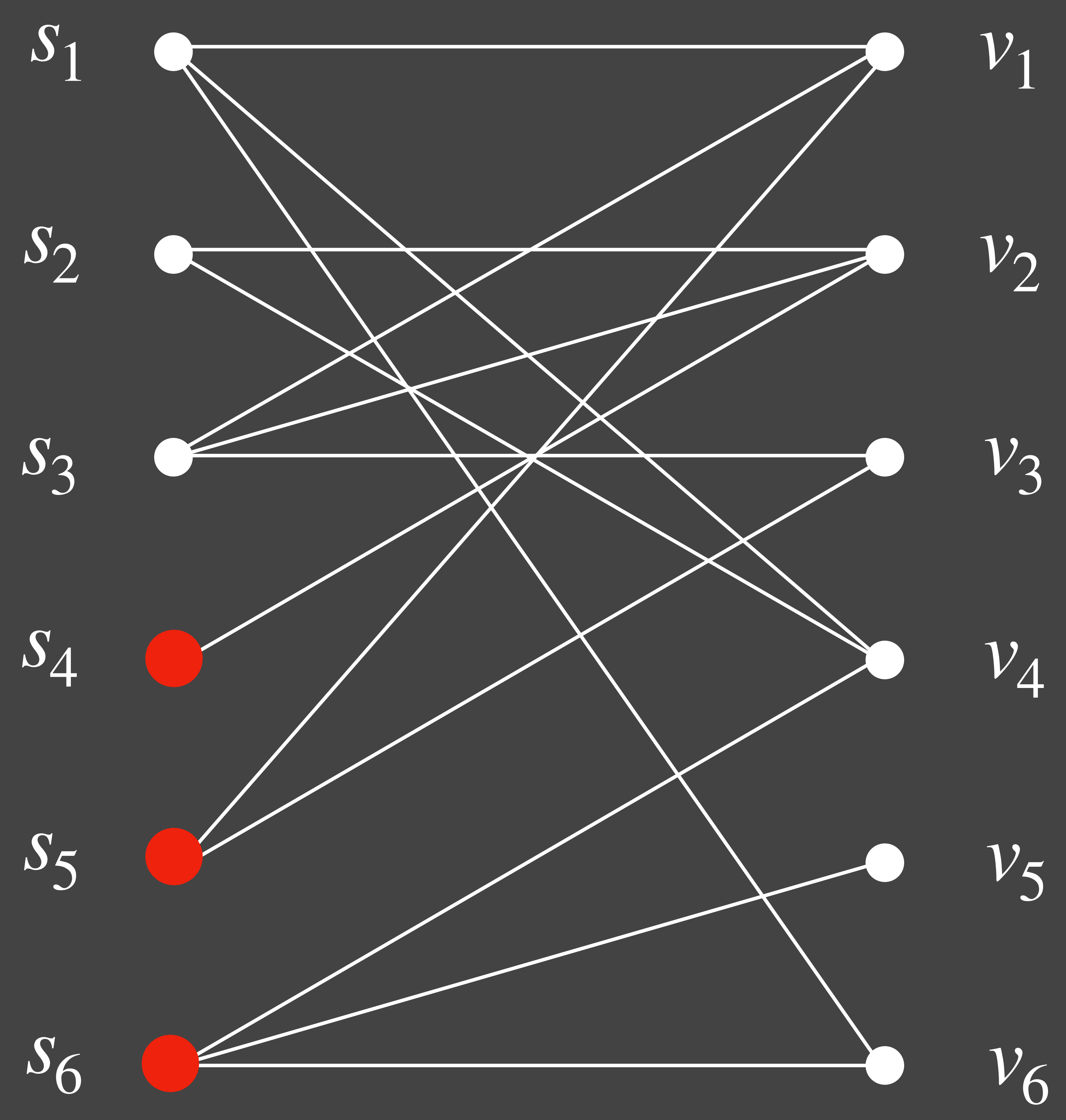
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



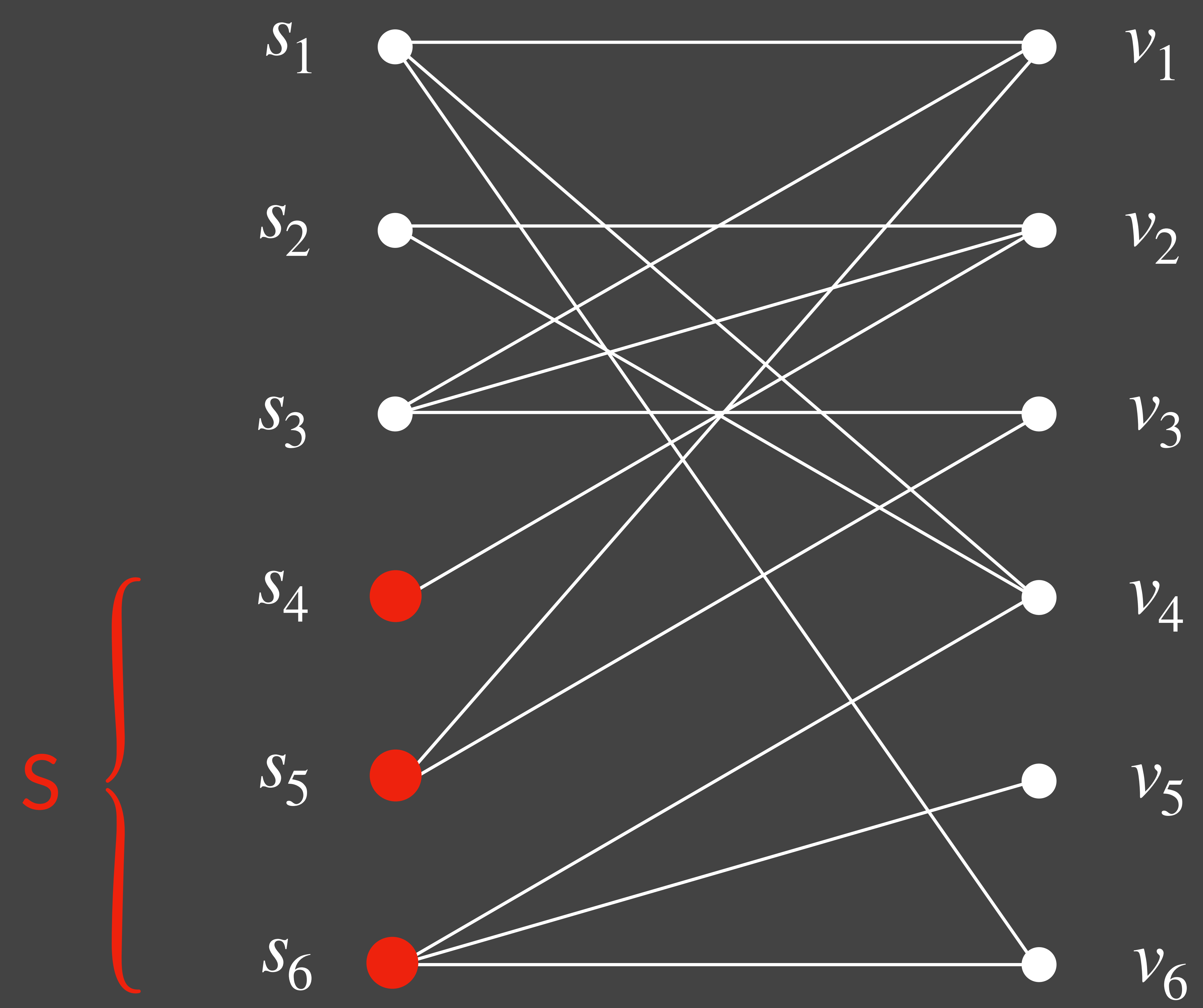
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



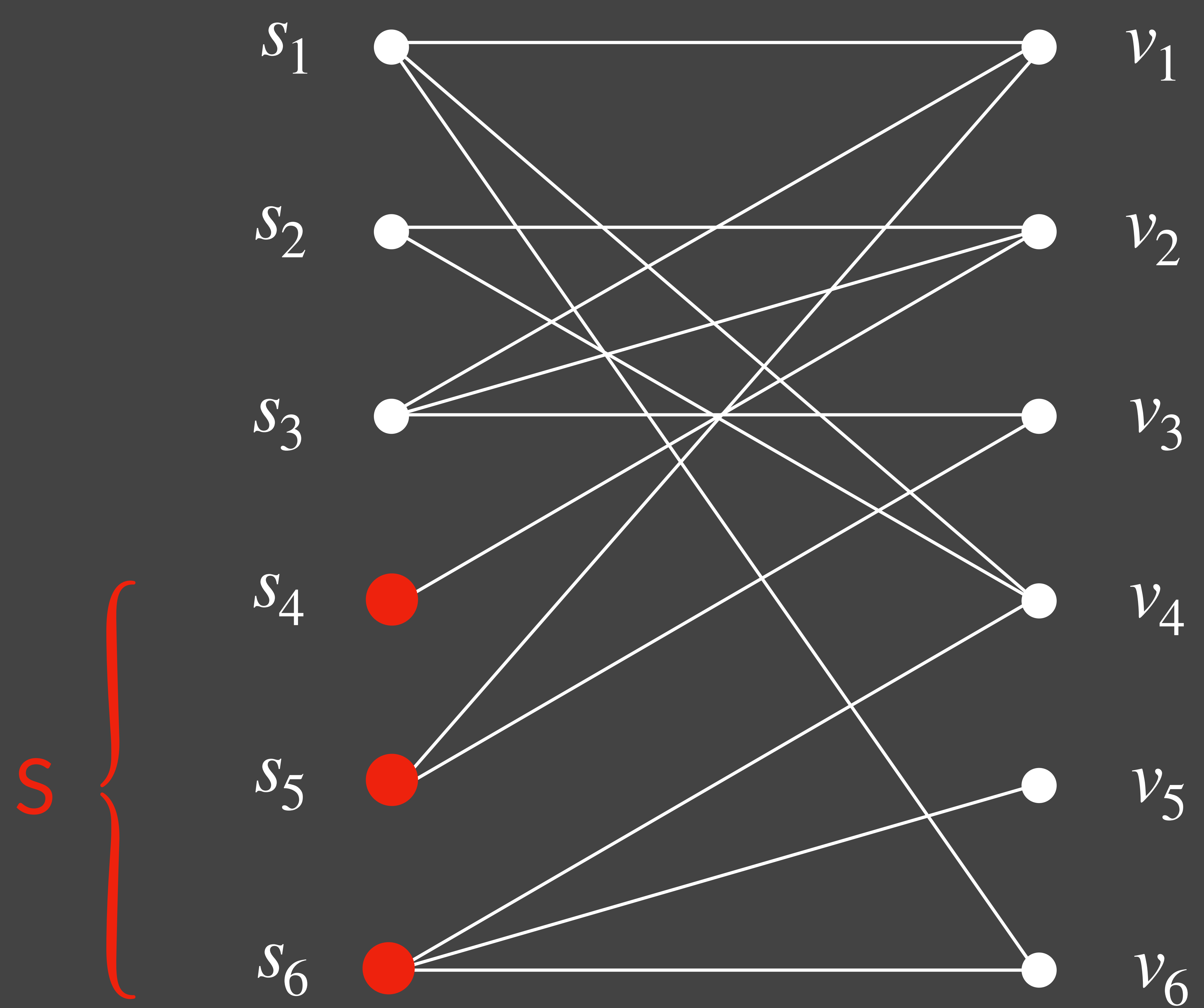
Special Case: Online Set Cover

[Alon Awerbuch Azar Buchbinder Naor 03]



Special Case: Online Set Cover

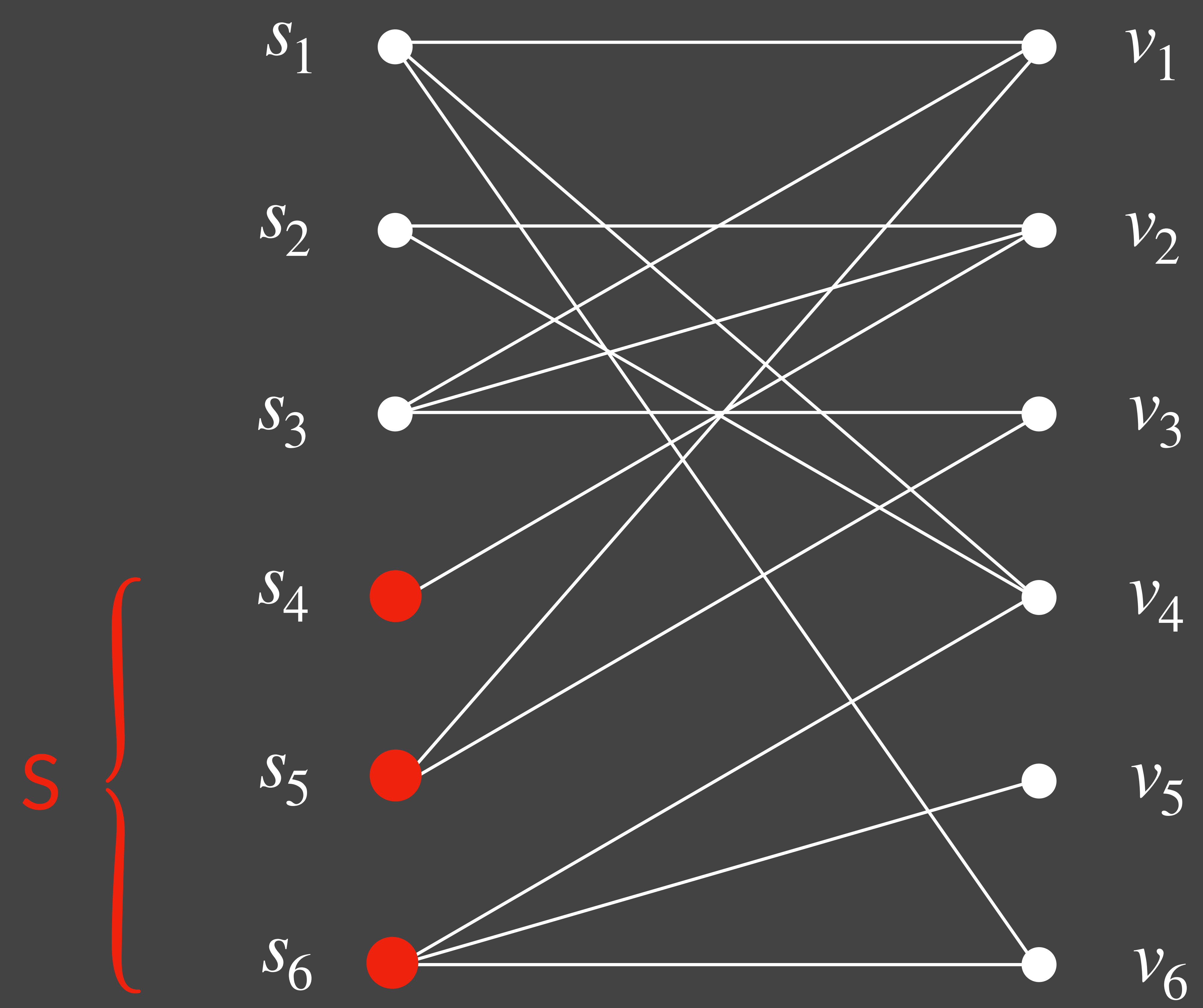
[Alon Awerbuch Azar Buchbinder Naor 03]



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

Special Case: Online Set Cover

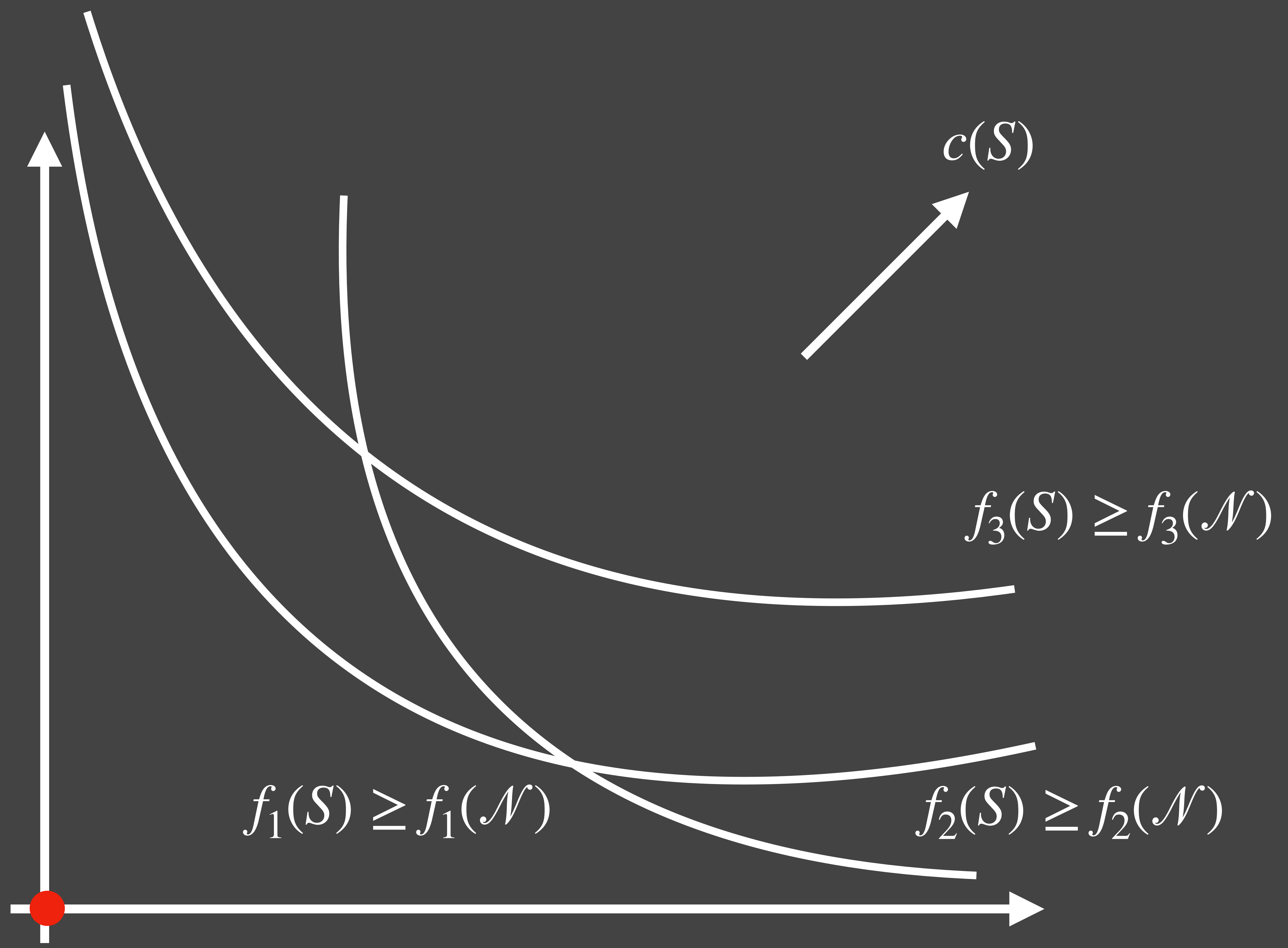
[Alon Awerbuch Azar Buchbinder Naor 03]



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

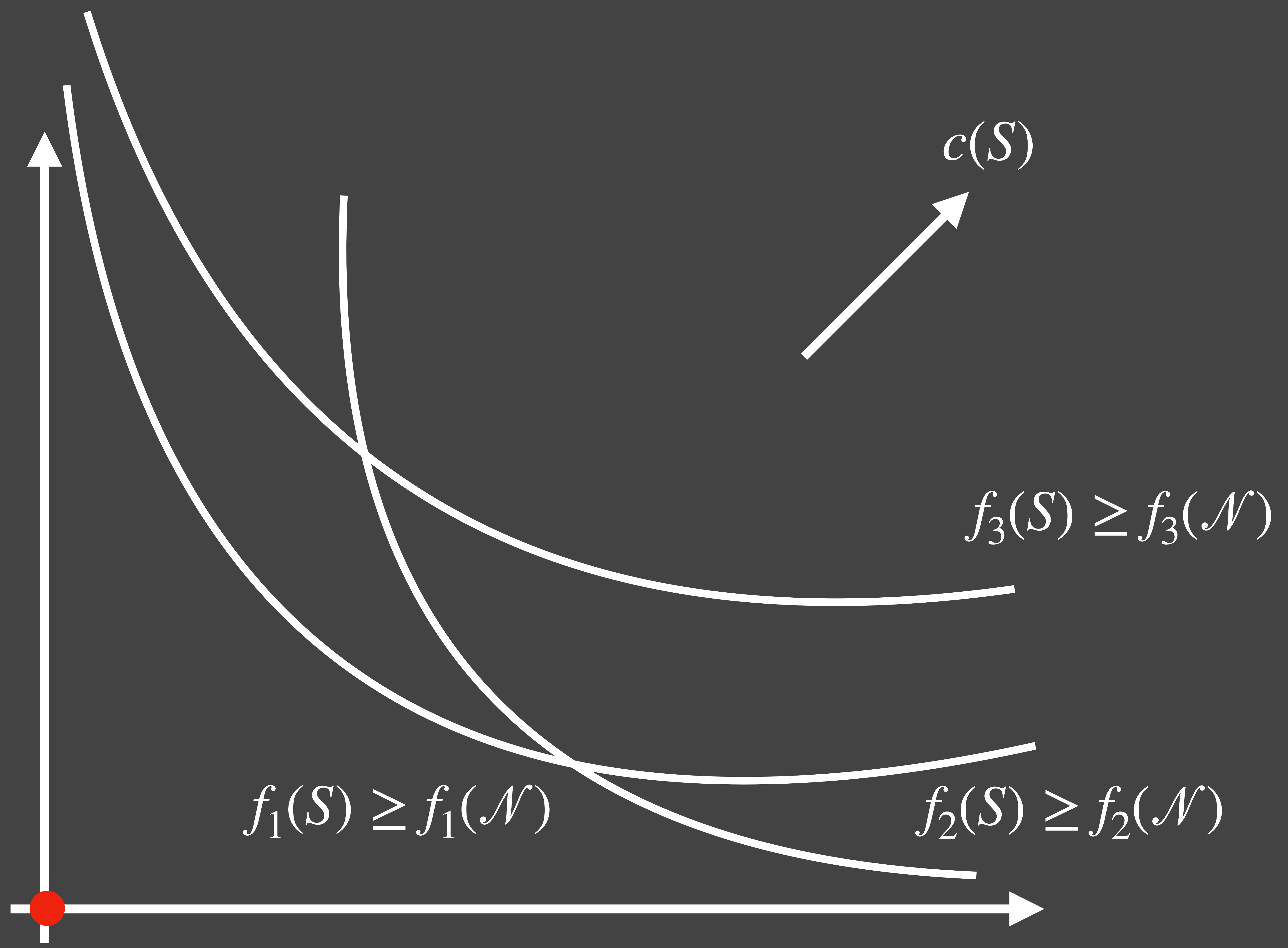
$$F = \sum_i f_i = \text{\# elements covered}$$

Online Submodular Cover Results



$$F = \sum_i f_i$$

Online Submodular Cover Results



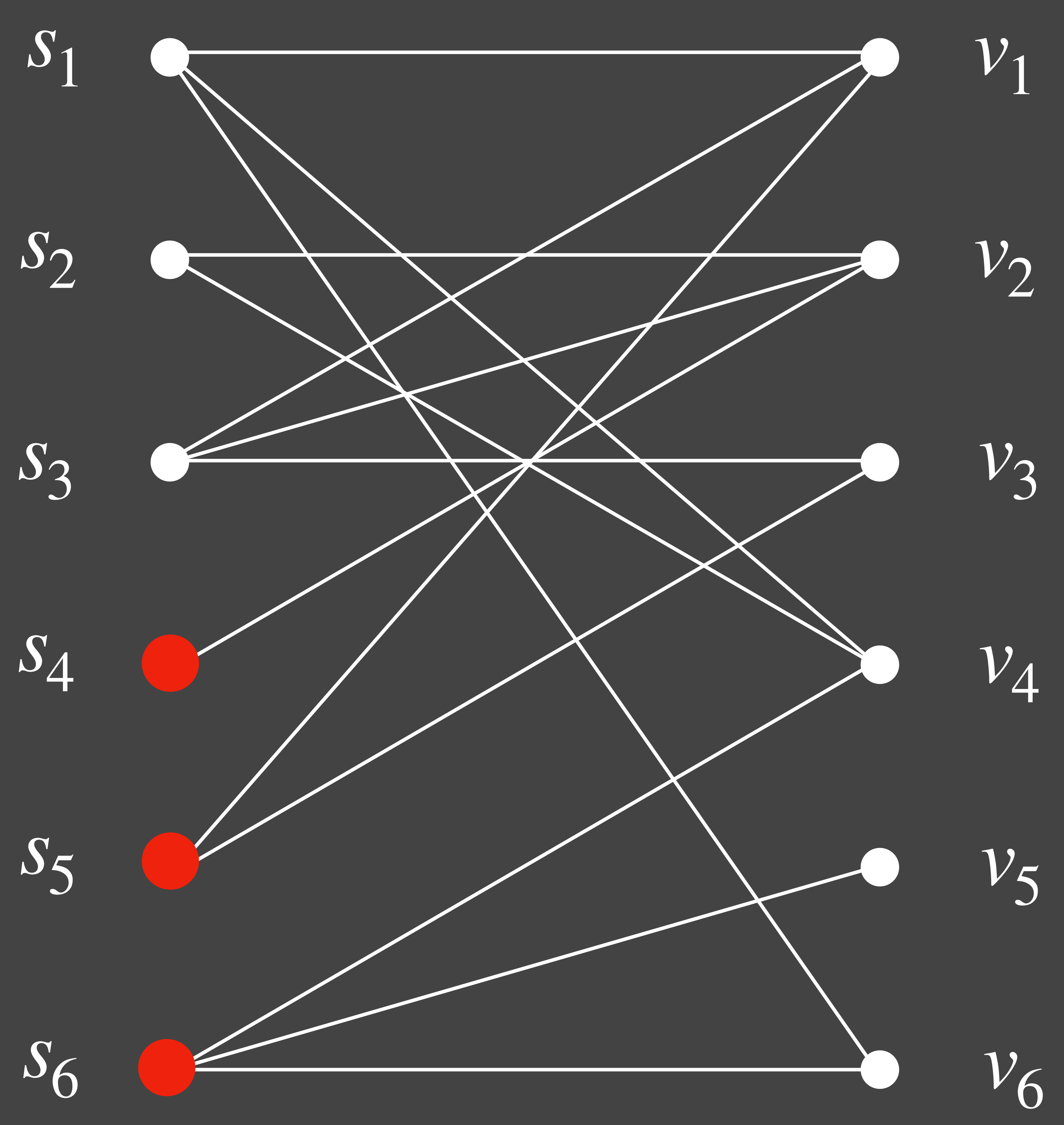
$$F = \sum_i f_i$$

Theorem [Gupta L. SODA20]:

*There is a **randomized poly time** algo for **Online Submod Cover** with expected competitive ratio:*

$$O(\log m \cdot \log F(\mathcal{N})).$$

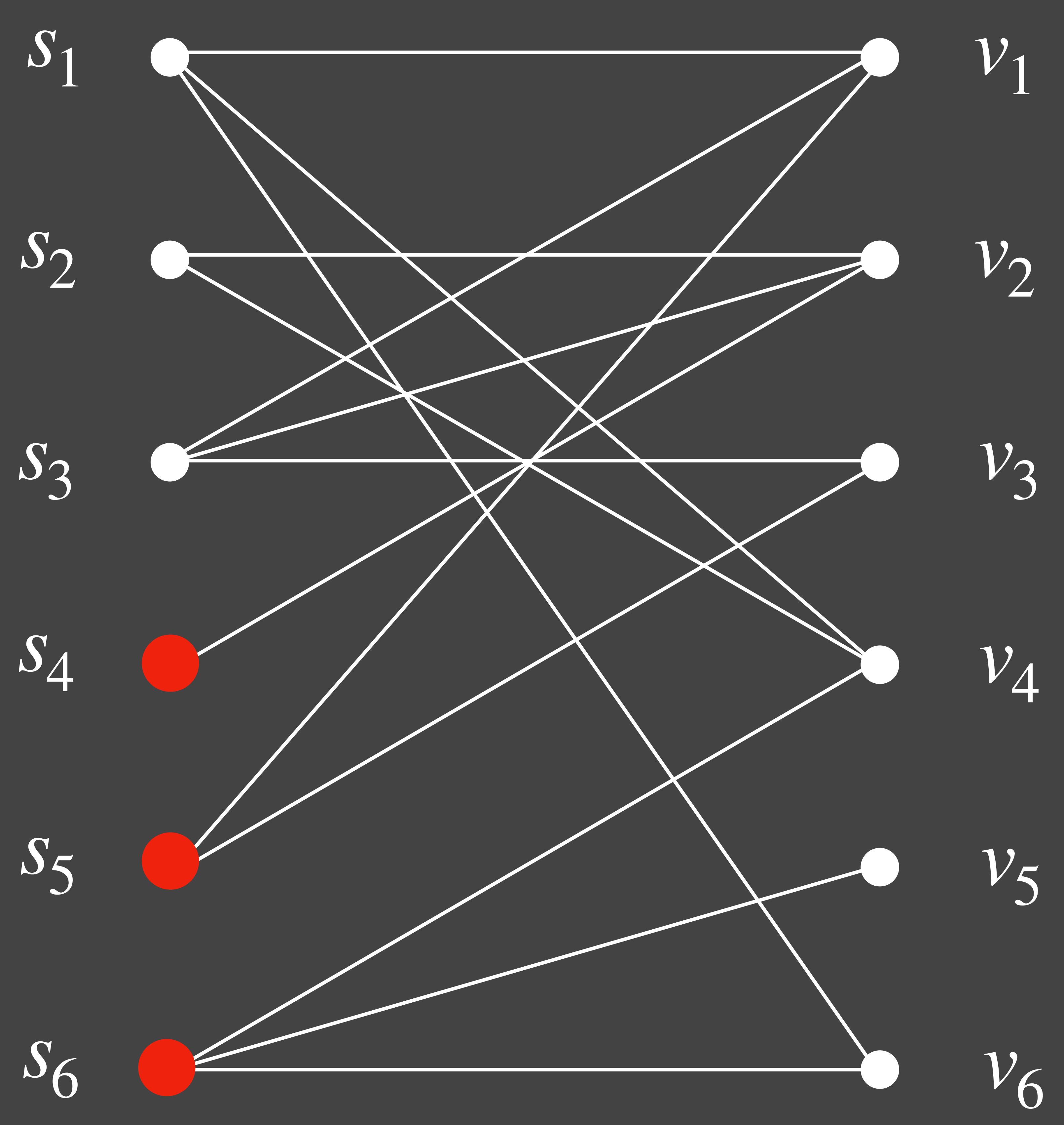
Special Case: Online Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

$$F = \sum_i f_i = \text{\# elements covered}$$

Special Case: Online Set Cover



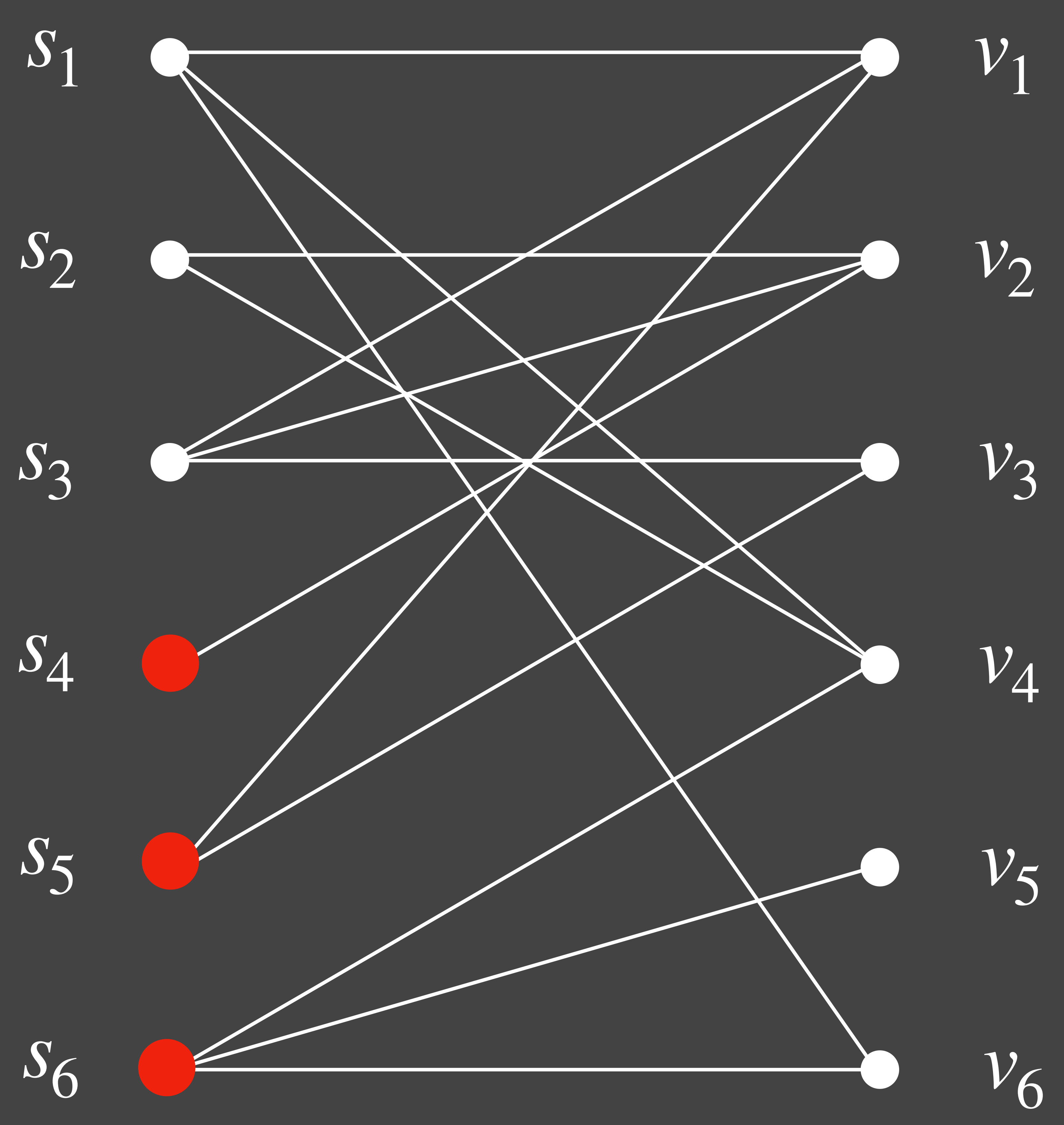
$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

$$F = \sum_i f_i = \text{\# elements covered}$$

Theorem (Online):

$O(\log m \cdot \log F(\mathcal{N}))$.

Special Case: Online Set Cover



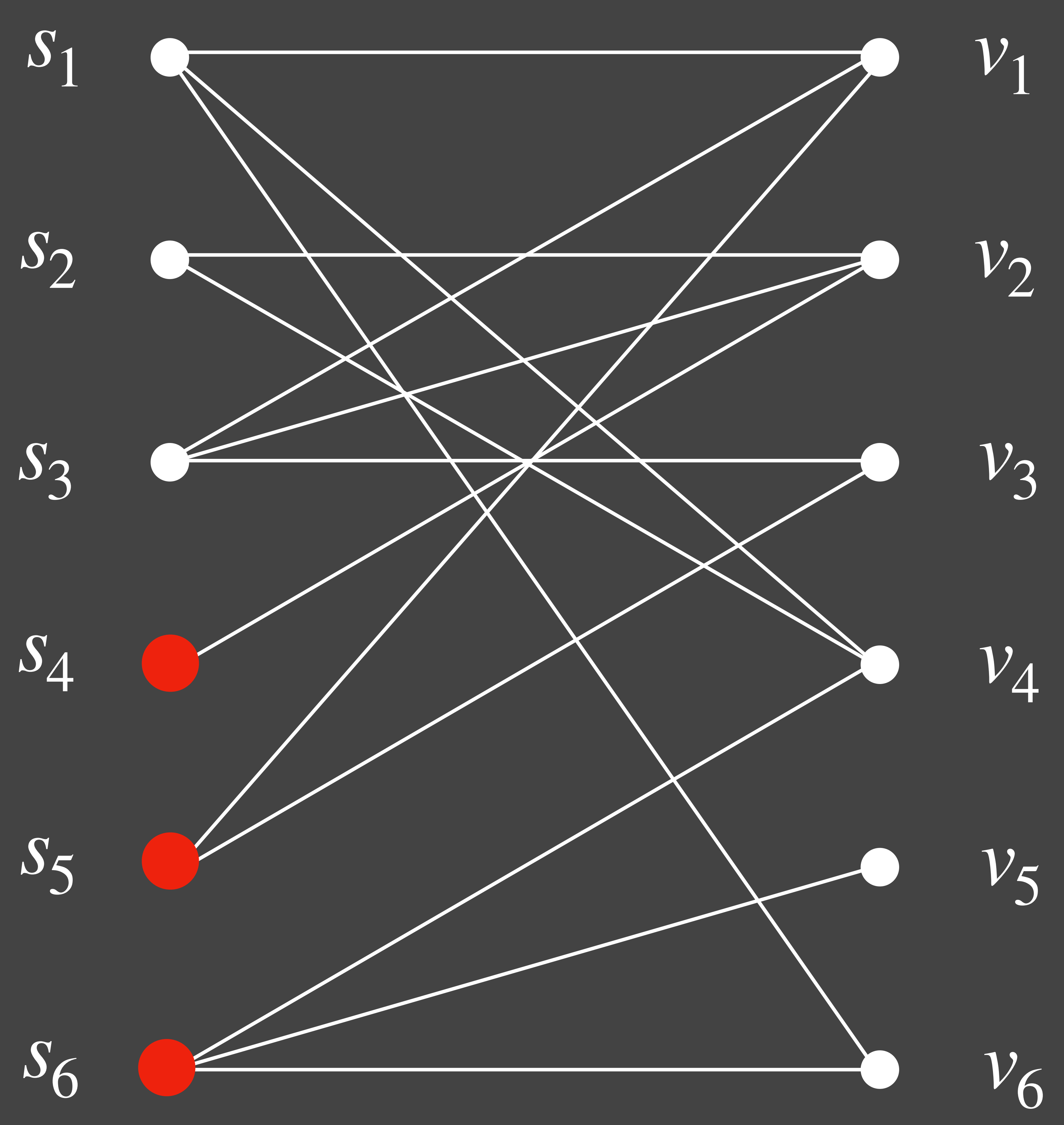
$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

$$F = \sum_i f_i = \text{\# elements covered}$$

<u>Theorem (Online)</u>:

$O(\log m \cdot \log n)$.

Special Case: Online Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

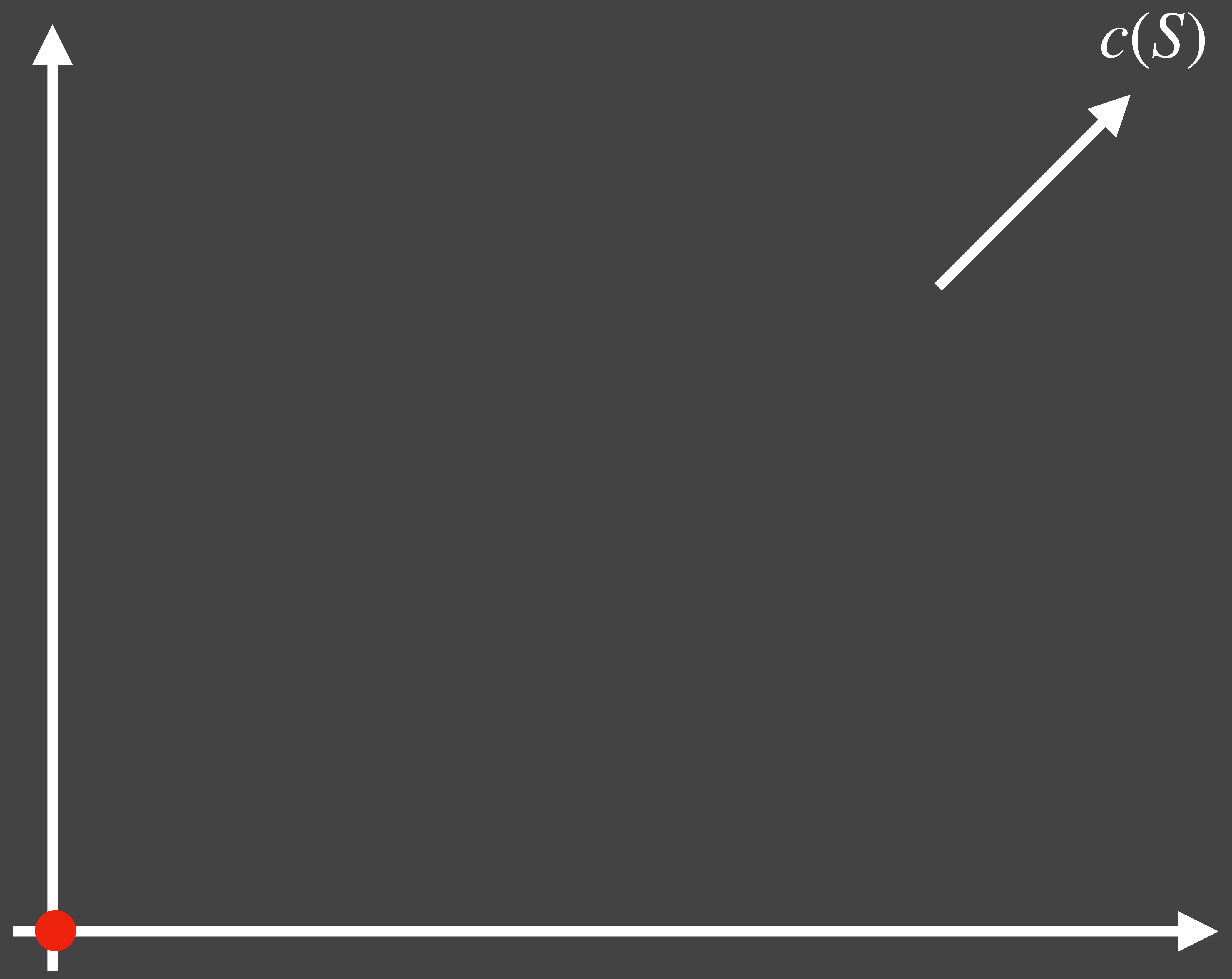
$$F = \sum_i f_i = \text{\# elements covered}$$

Theorem (Online):

$O(\log m \cdot \log n)$.

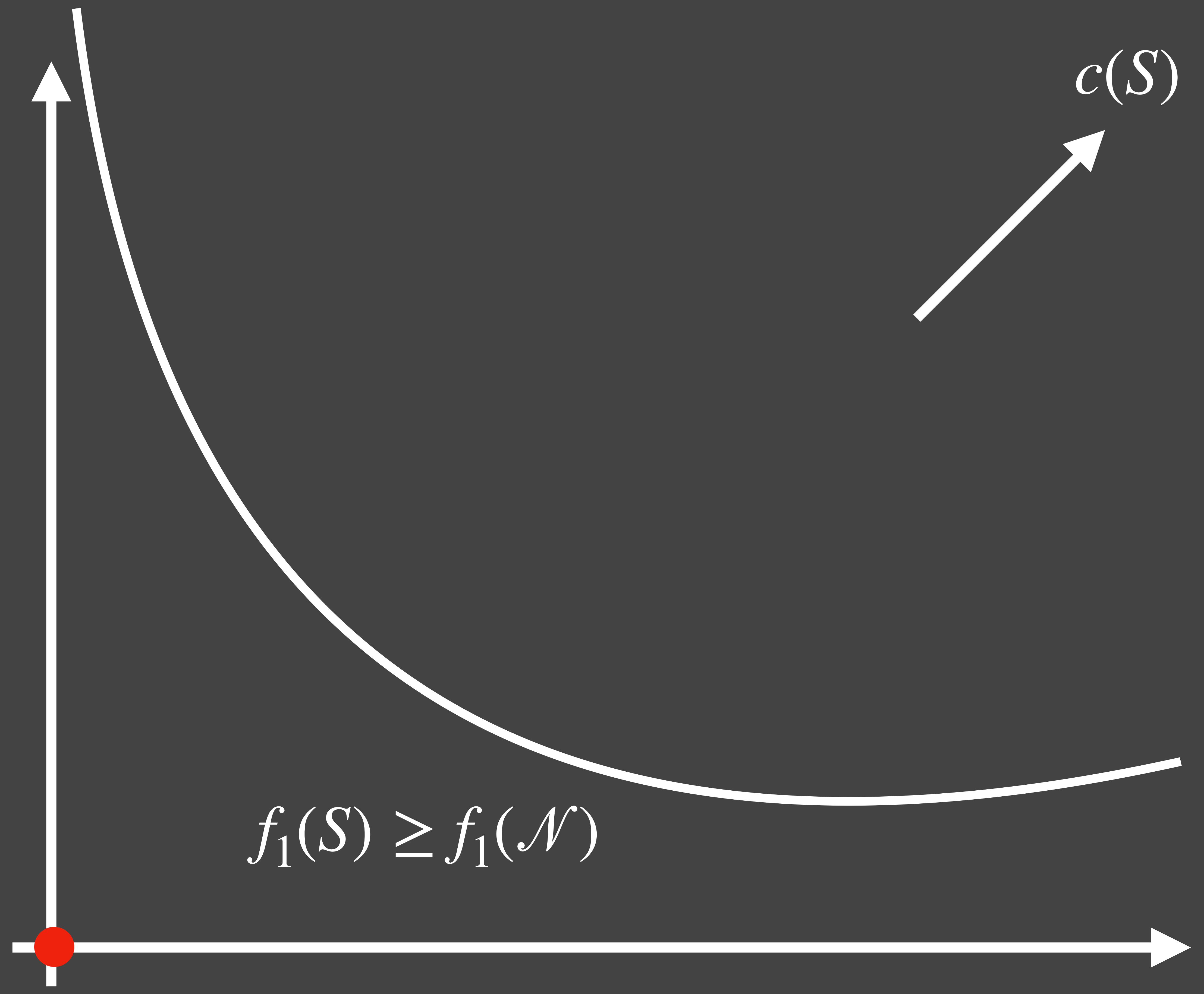
Generalizes [Alon+ 03]

Fully-Dynamic Submodular Cover



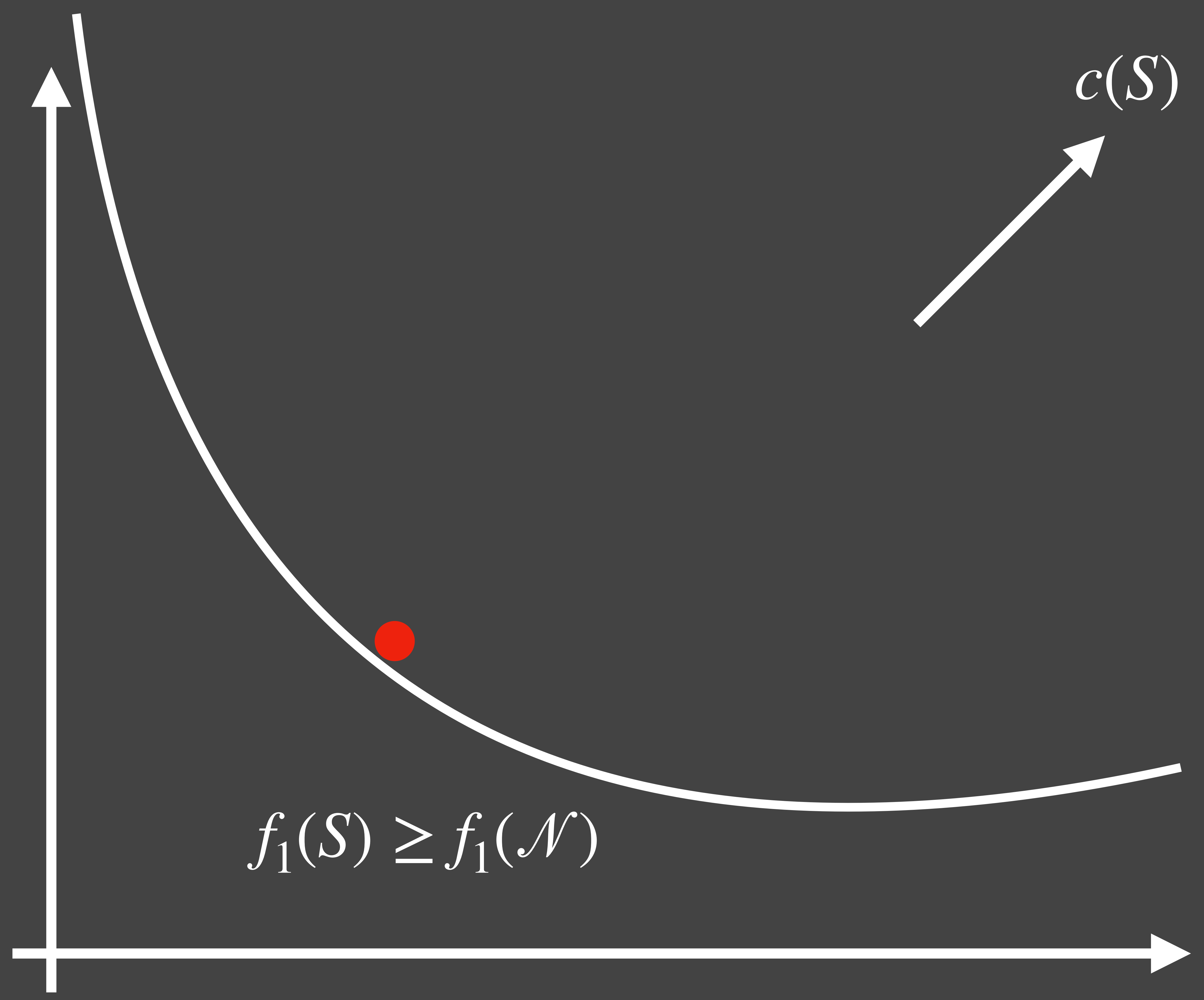
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



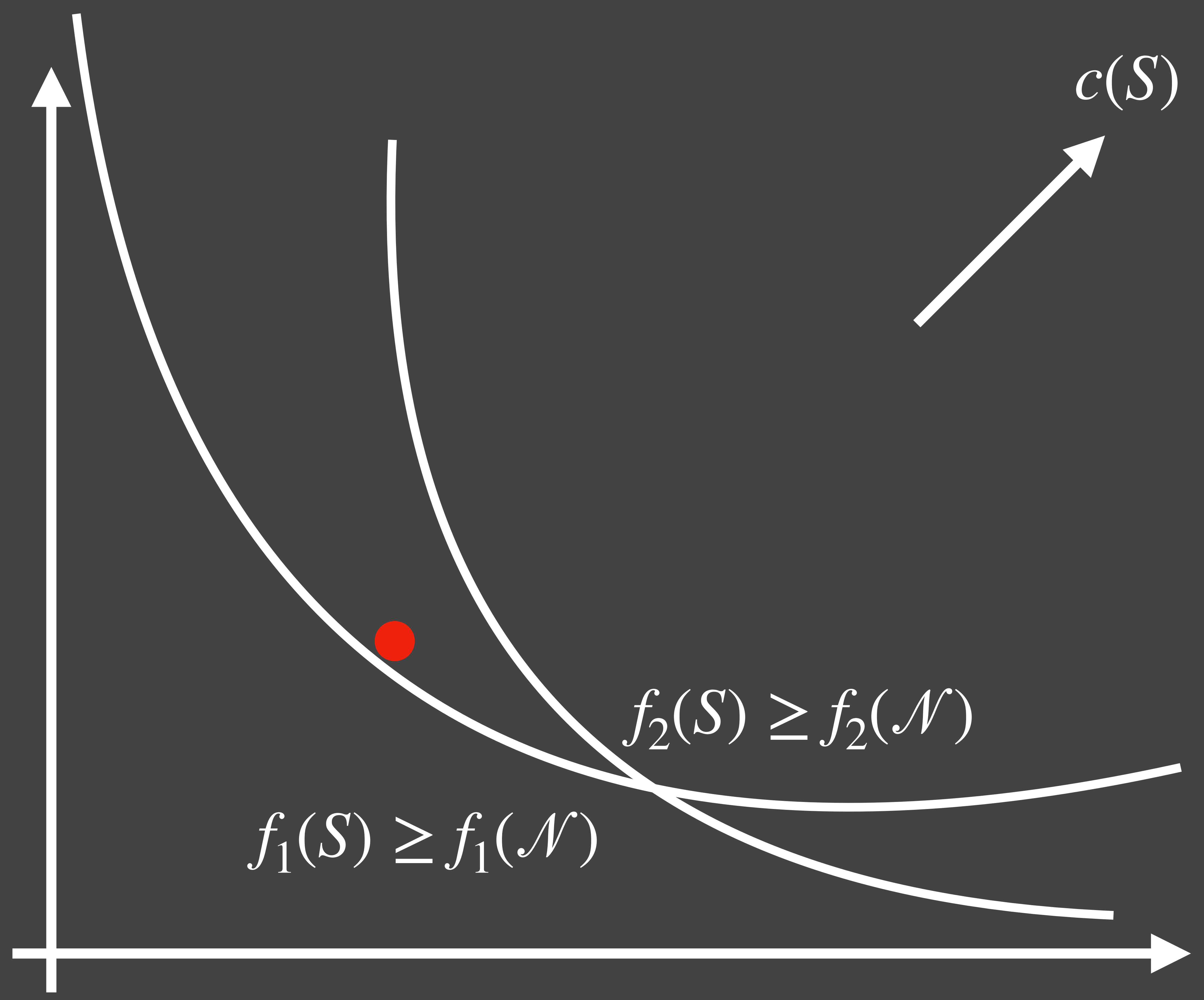
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



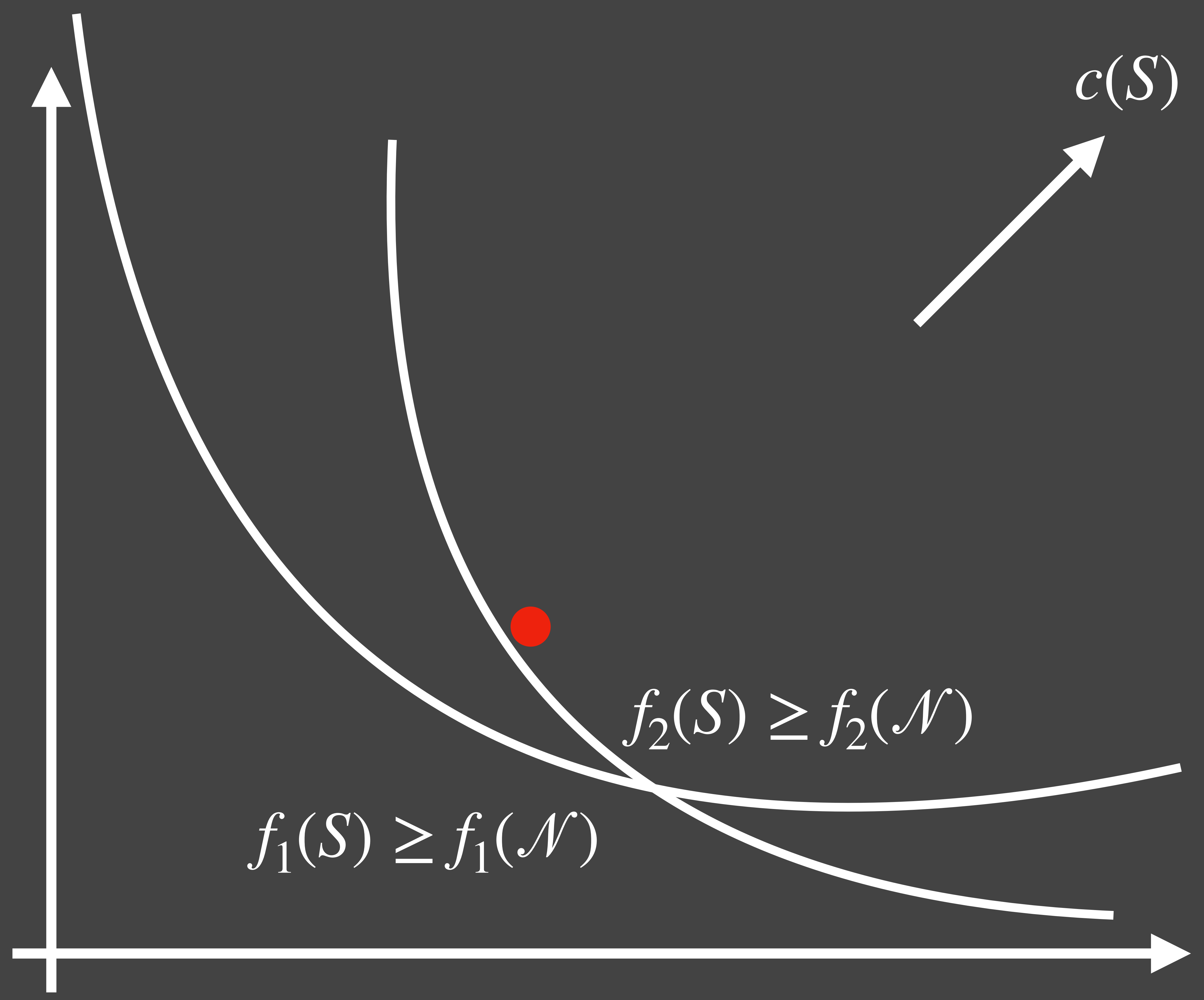
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



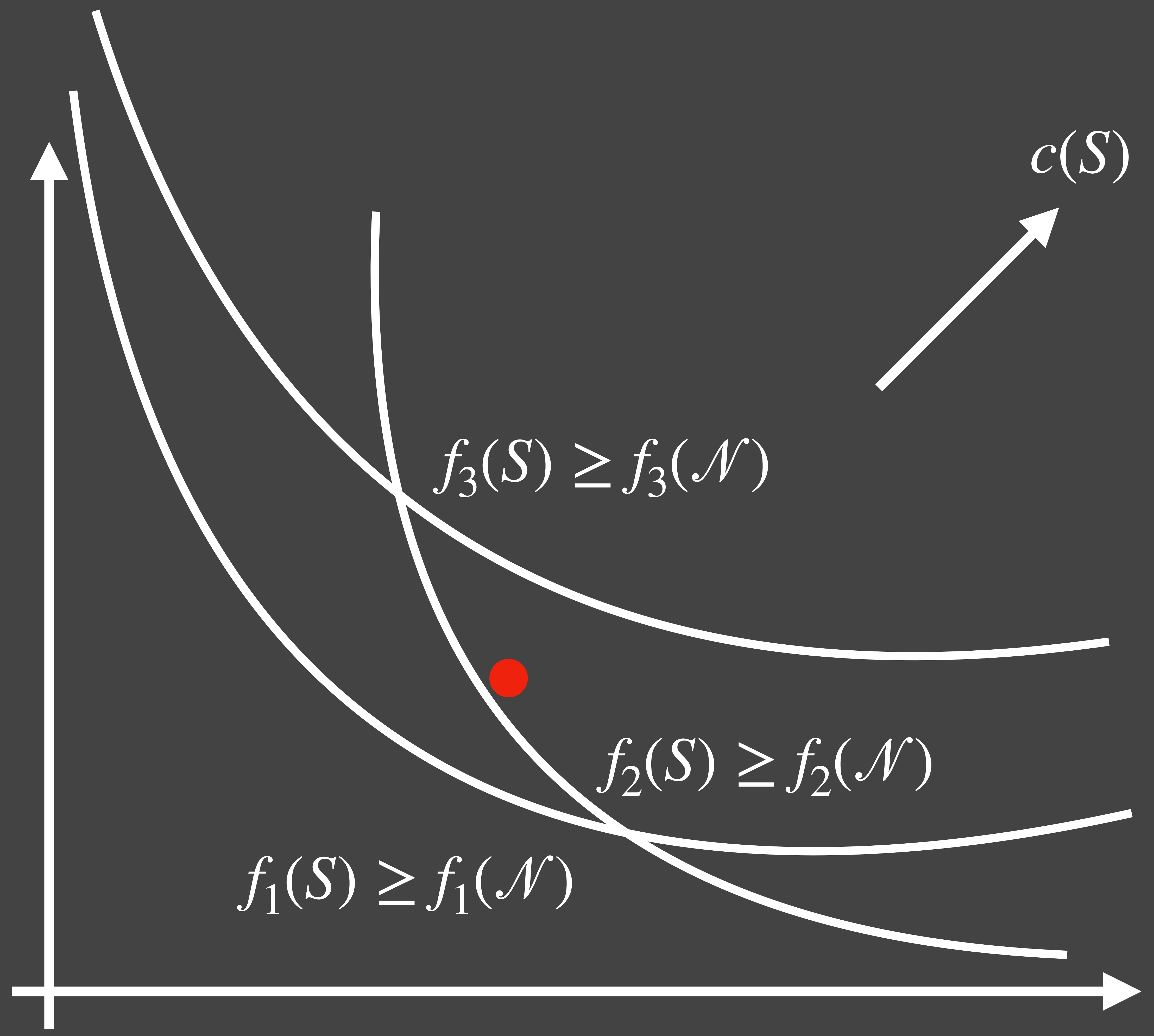
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



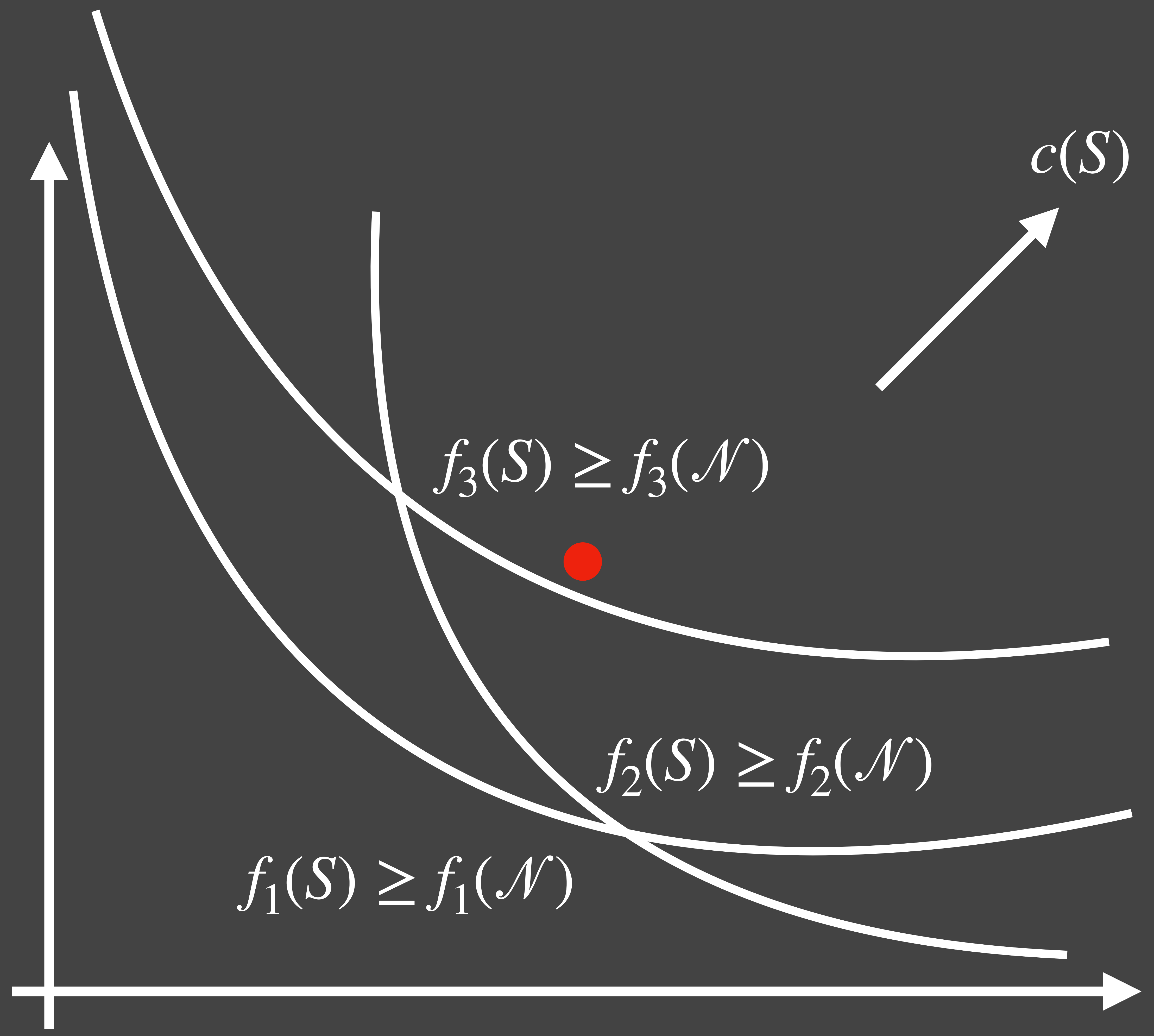
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



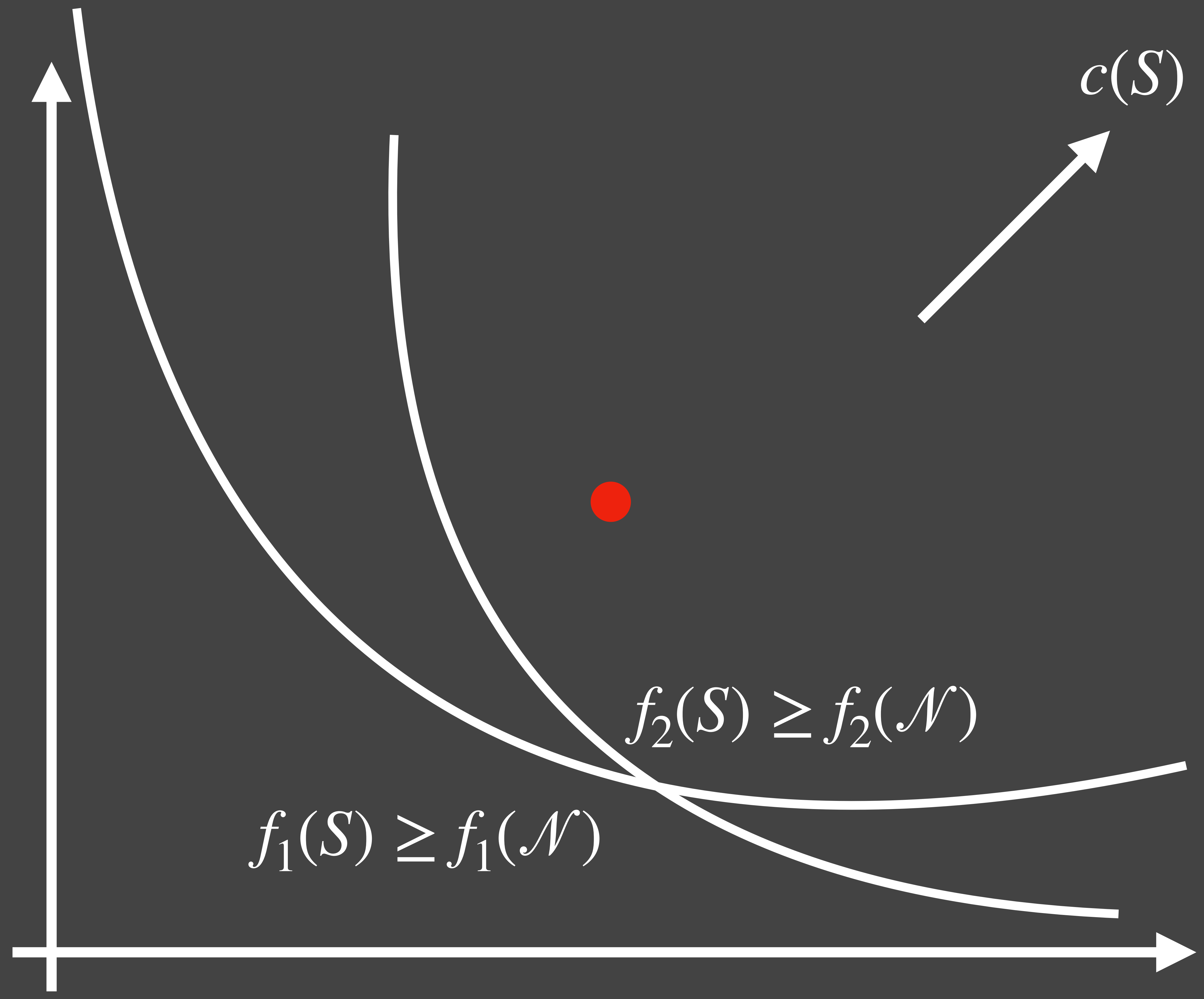
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



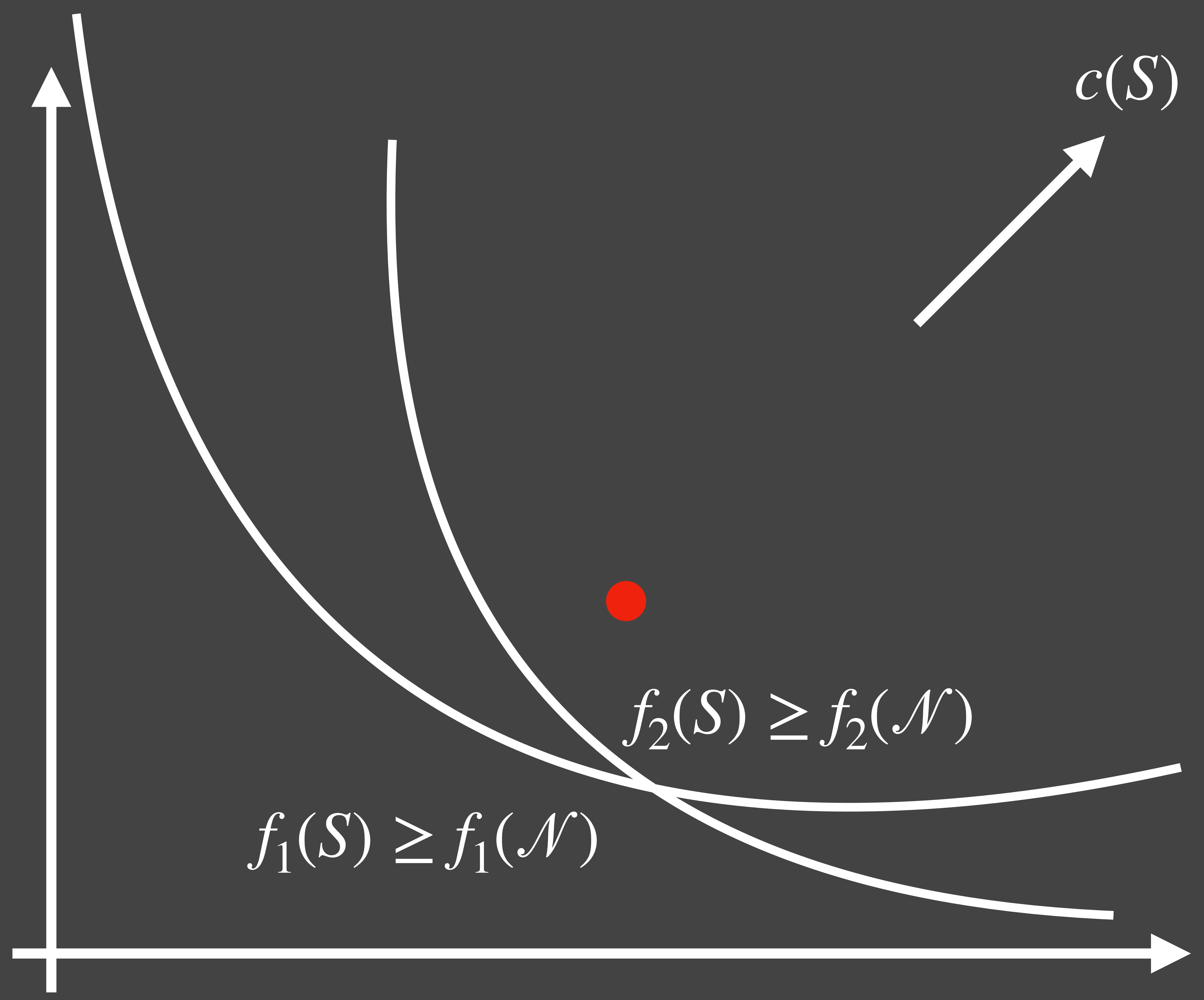
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



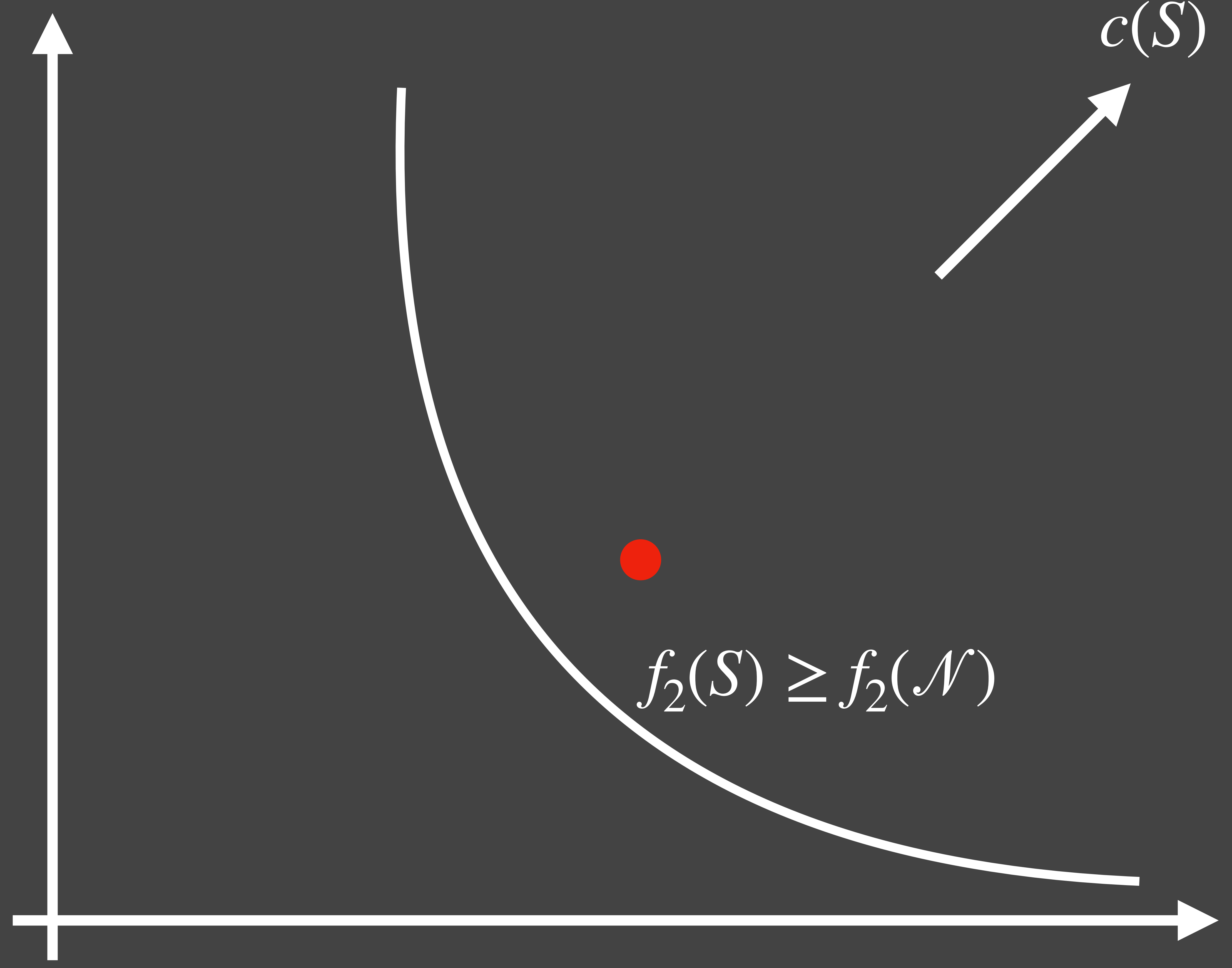
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



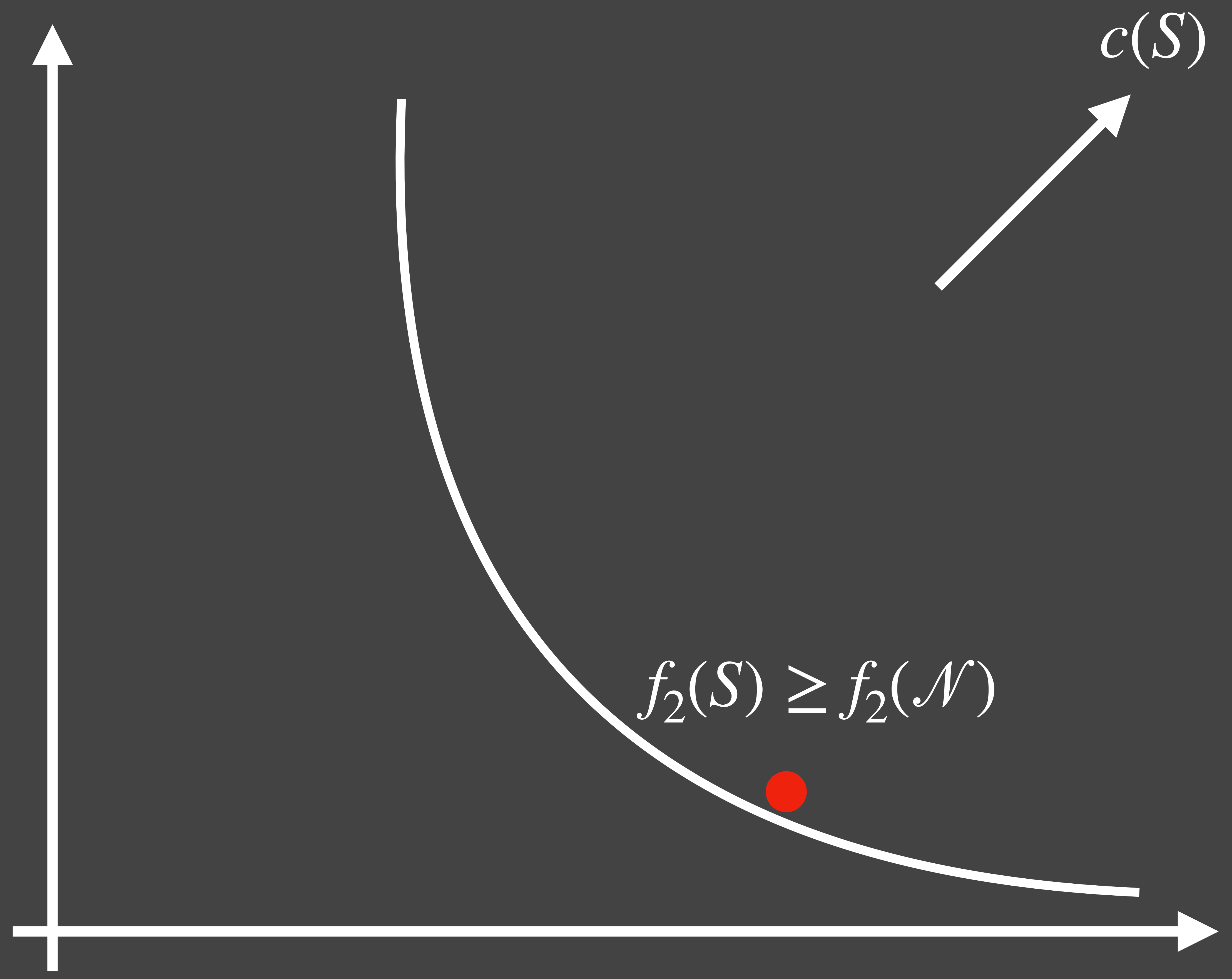
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



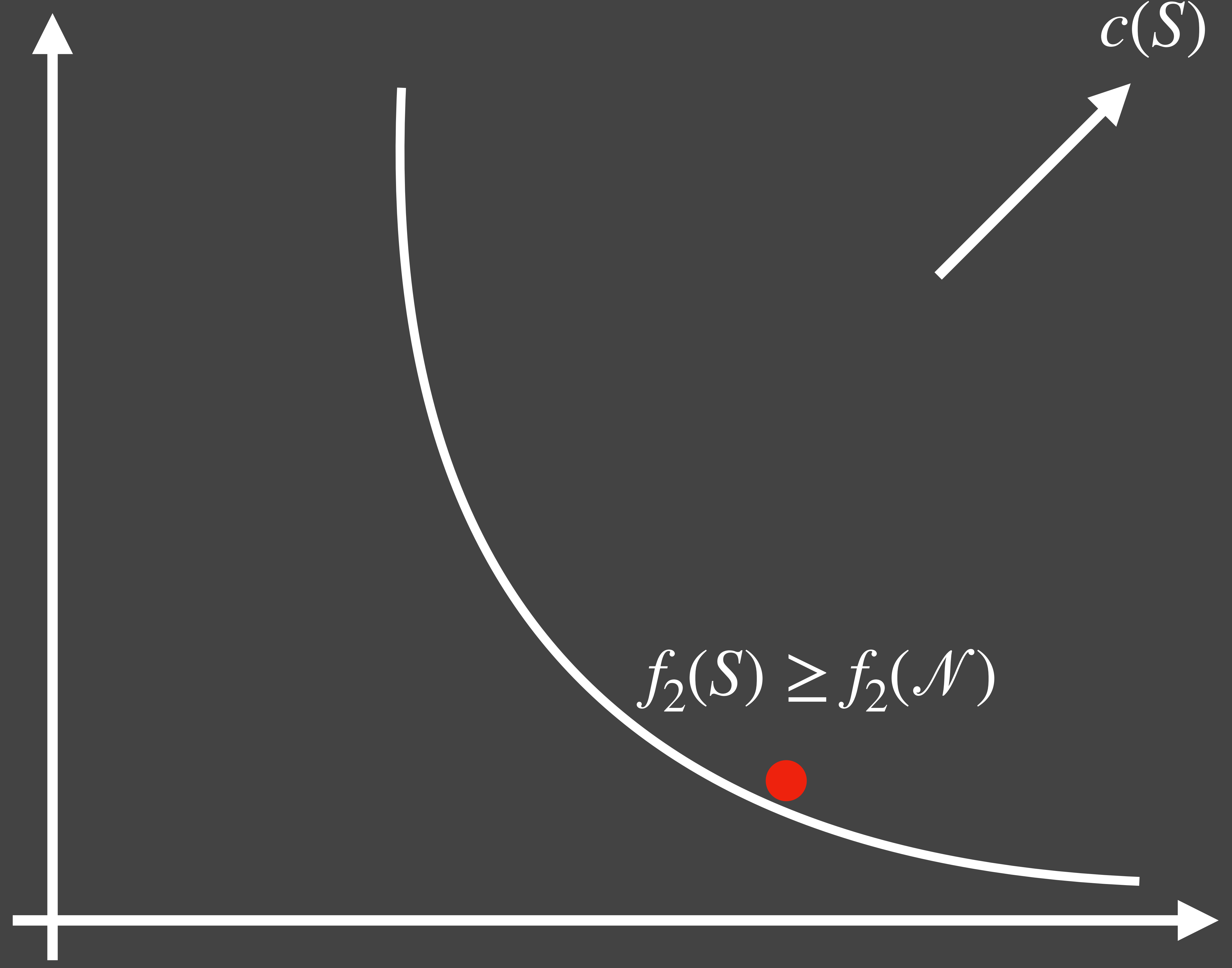
$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover



$$F = \sum_i f_i$$

Fully-Dynamic Submodular Cover

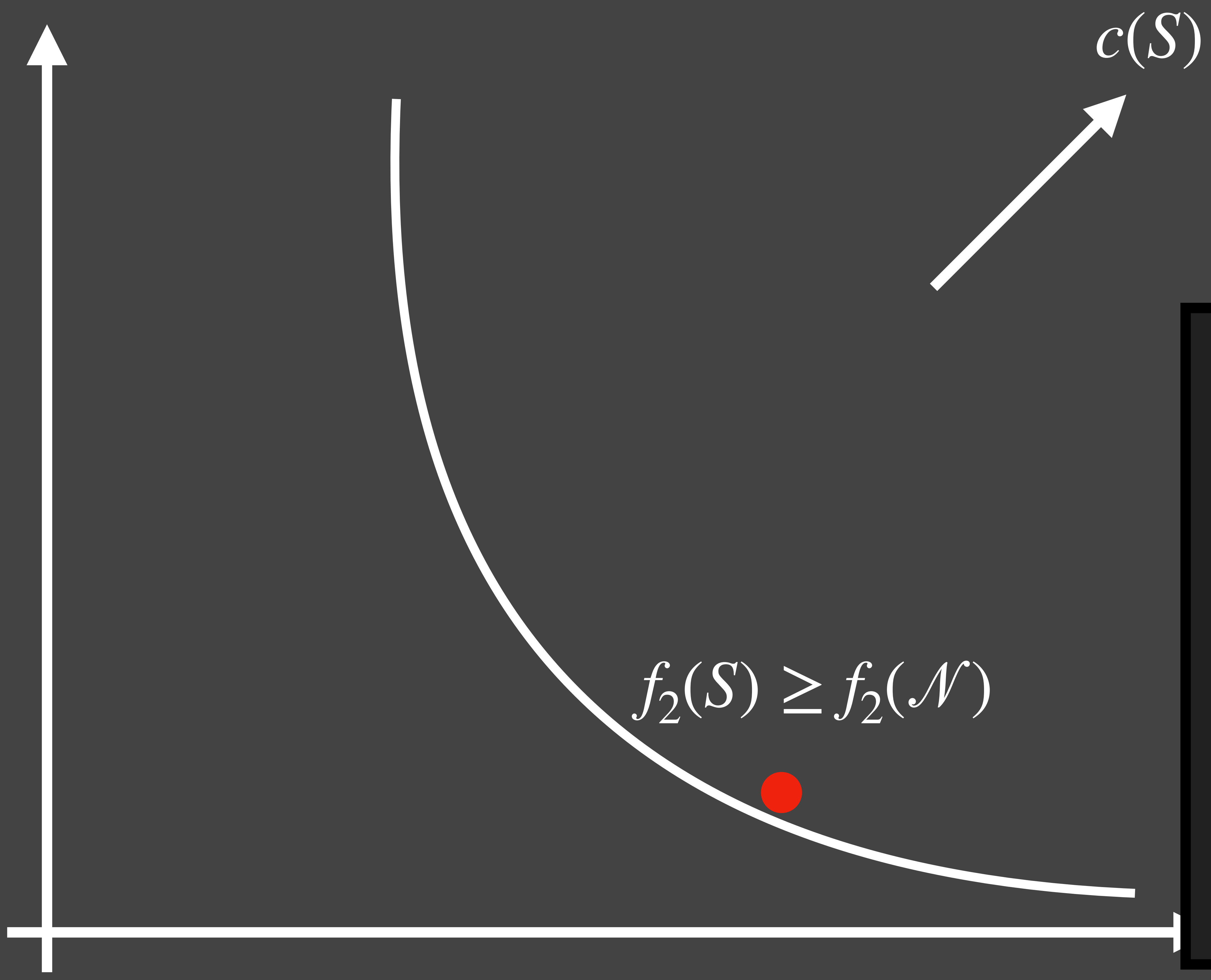


$$F = \sum_i f_i$$

Definition: Recourse

$$\sum_t |S^t \triangle S^{t-1}|$$

Fully-Dynamic Submodular Cover



$$F = \sum_i f_i$$

Definition: Recourse

$$\sum_t |S^t \triangle S^{t-1}|$$

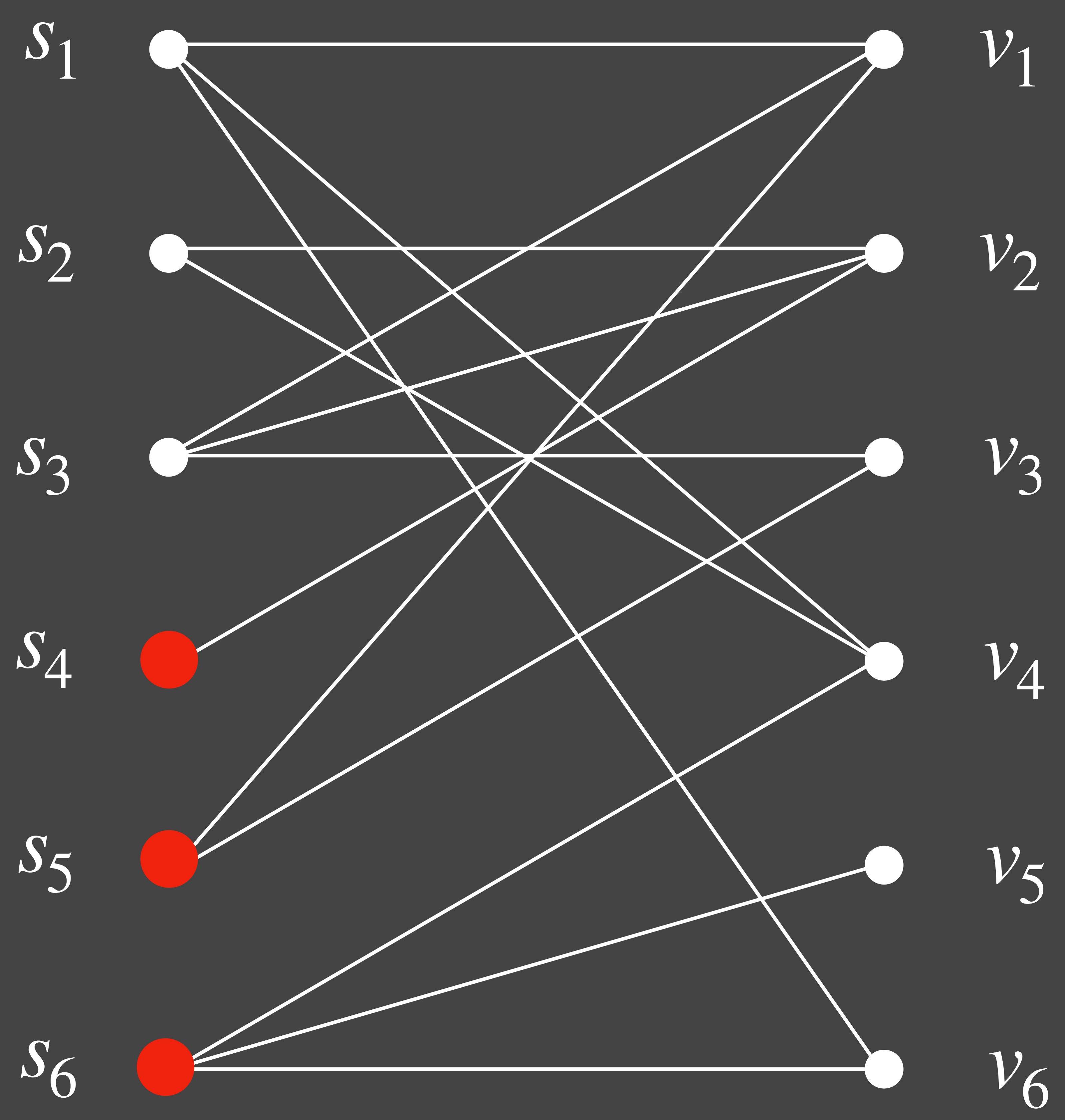
Theorem [Gupta **L.** FOCS 20]:

*There is a **deterministic poly time** algorithm for **Fully-Dynamic Submodular** Cover with:*

(i) *competitive ratio* $O(\log F(\mathcal{N}))$.

(ii) *average recourse* $\tilde{O}(f(\mathcal{N}))$.

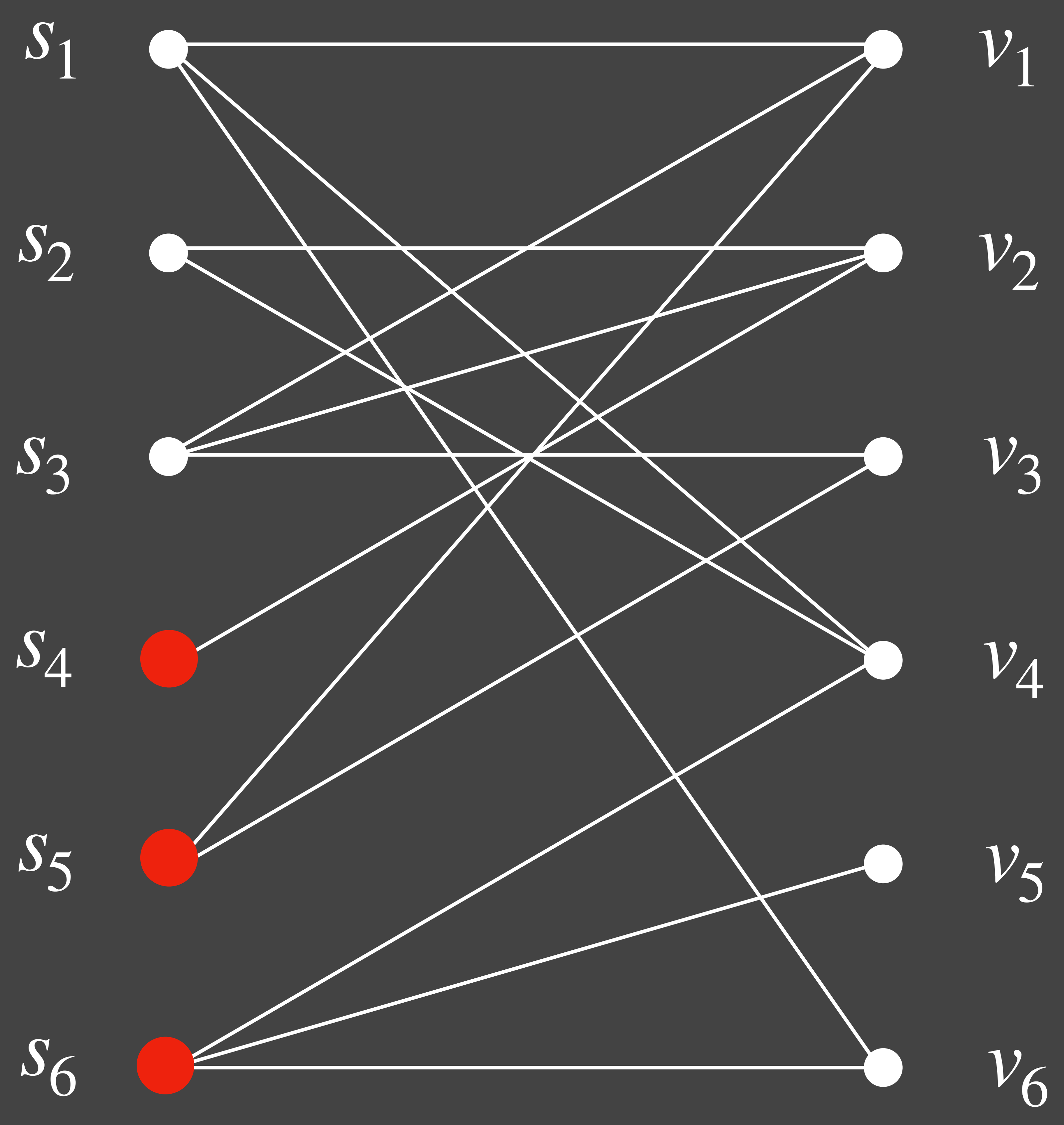
Special Case: Dynamic Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

$$F = \sum_i f_i = \text{\# elements covered}$$

Special Case: Dynamic Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

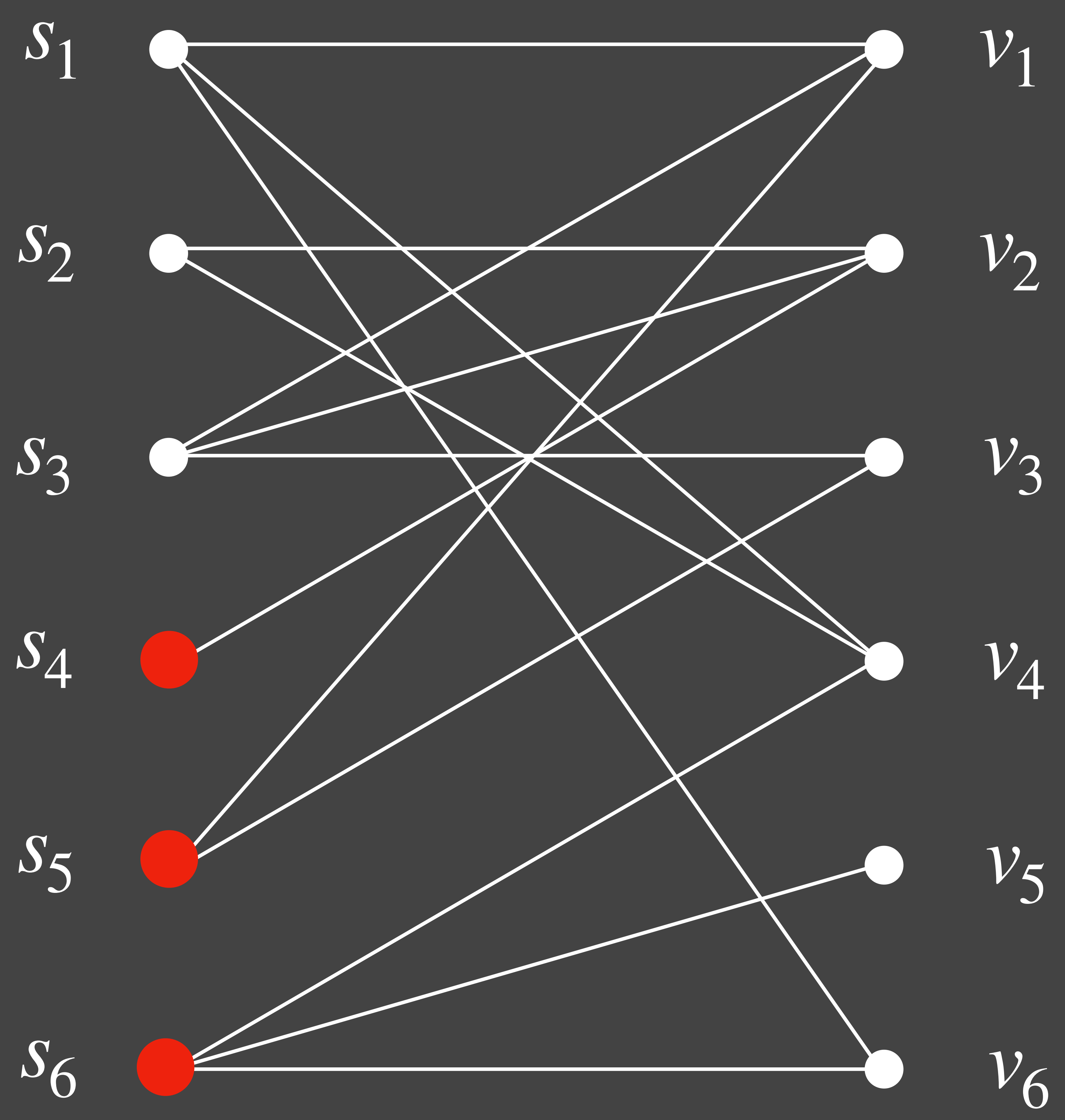
$$F = \sum_i f_i = \text{\# elements covered}$$

Theorem (Dynamic):

(i) *competitive ratio* $O(\log F(\mathcal{N}))$.

(ii) *average recourse* $O(f(\mathcal{N}))$.

Special Case: Dynamic Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

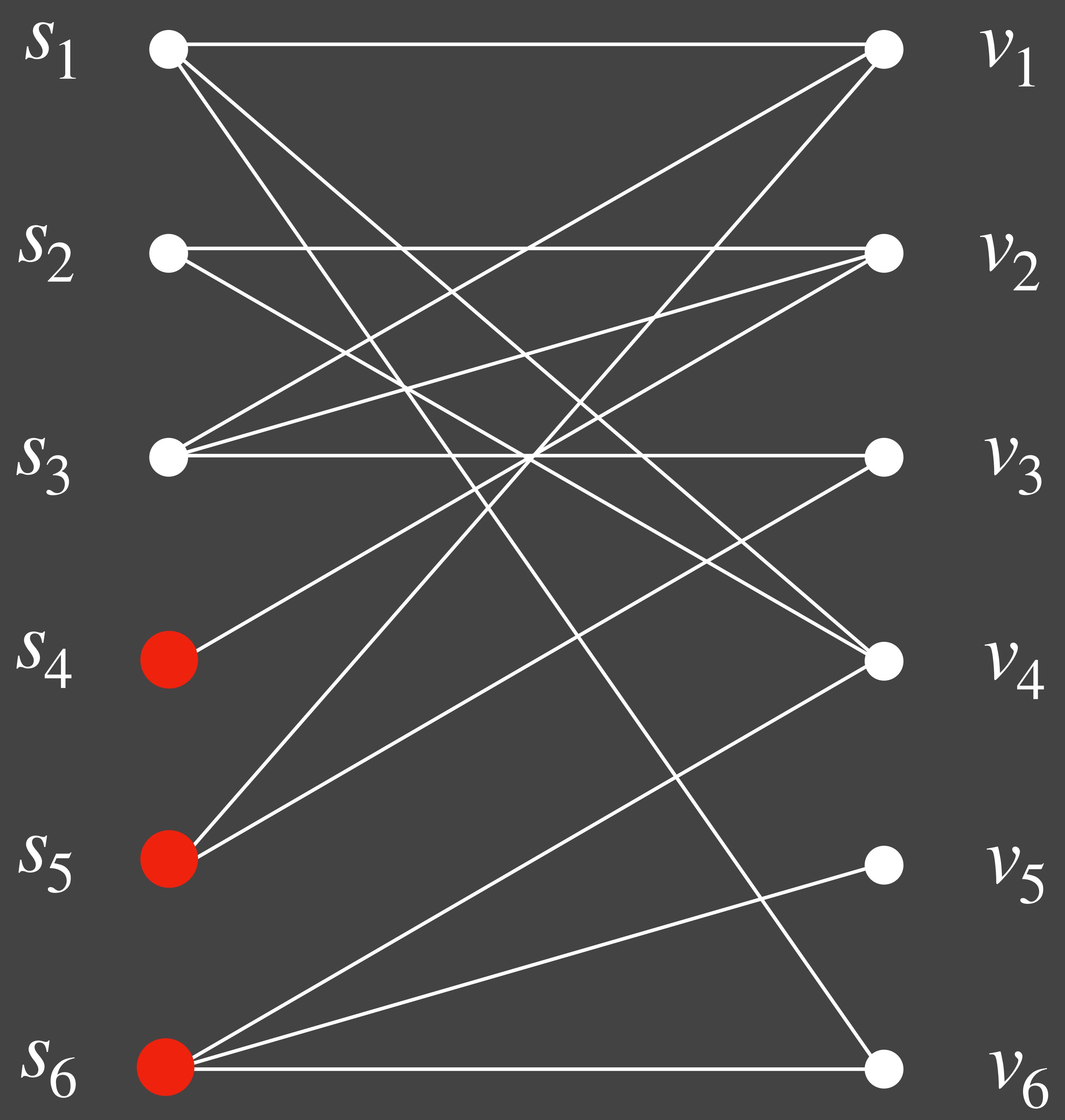
$$F = \sum_i f_i = \#\text{ elements covered}$$

Theorem (Dynamic):

(i) *competitive ratio* $O(\log n)$.

(ii) *average recourse* $O(1)$.

Special Case: Dynamic Set Cover



$$f_i(S) = \begin{cases} 1 & \text{if } v_i \text{ covered by } S \\ 0 & \text{otherwise} \end{cases}$$

$$F = \sum_i f_i = \#\text{ elements covered}$$

Theorem (Dynamic):

(i) *competitive ratio* $O(\log n)$.

(ii) *average recourse* $O(1)$.

Generalizes [Gupta Kumar Krishnaswamy Panigrahi 17]

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**

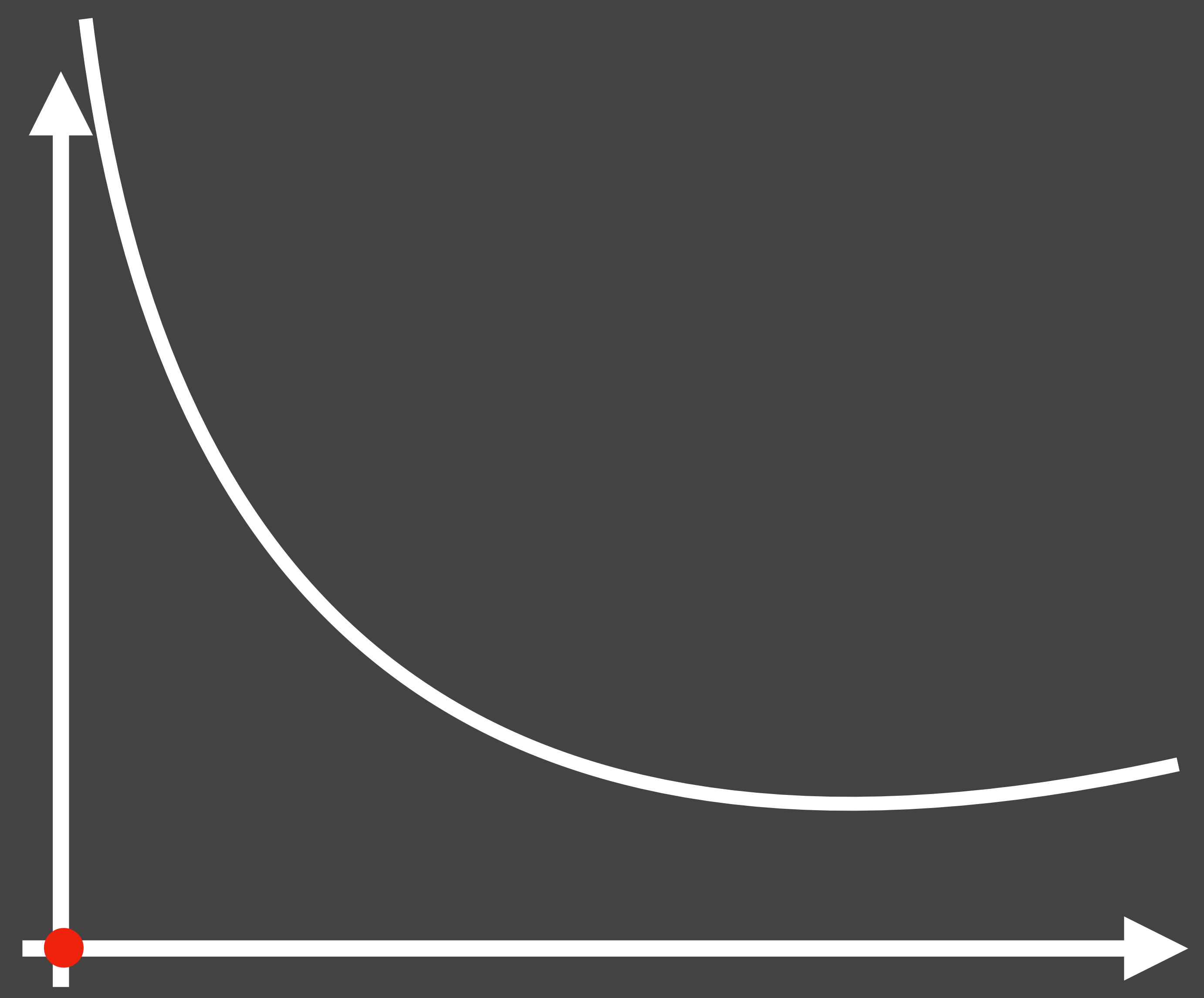
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



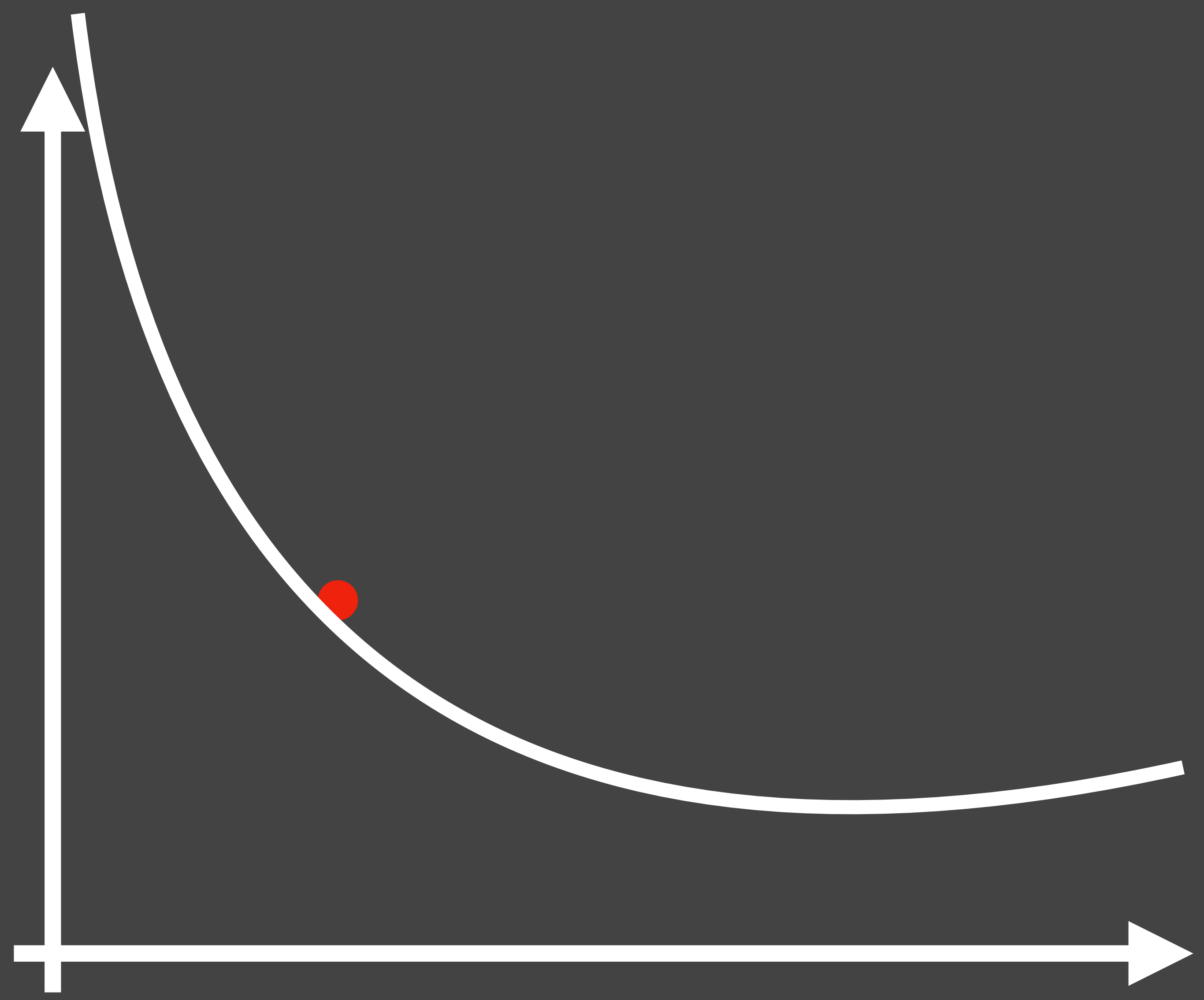
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



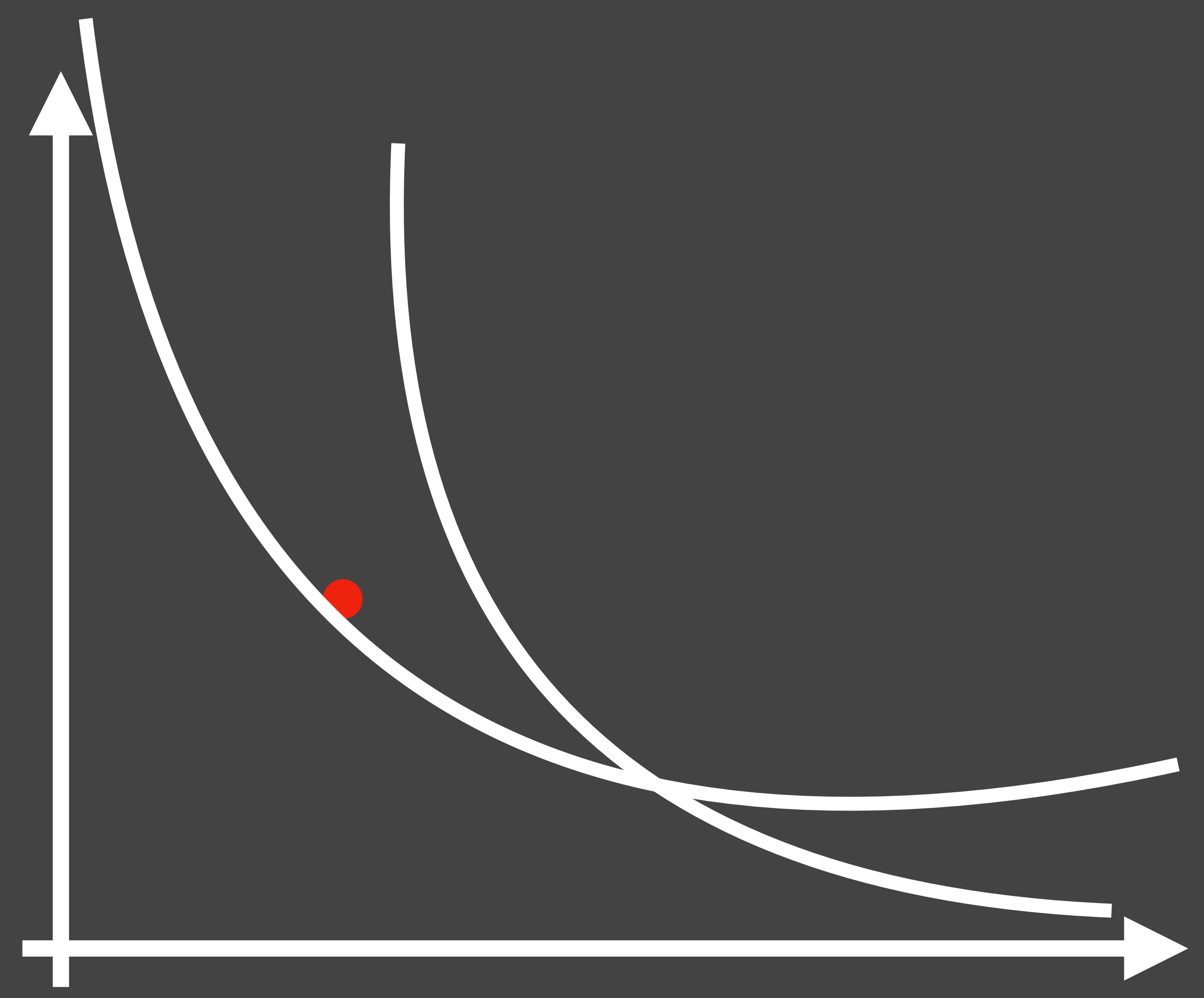
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



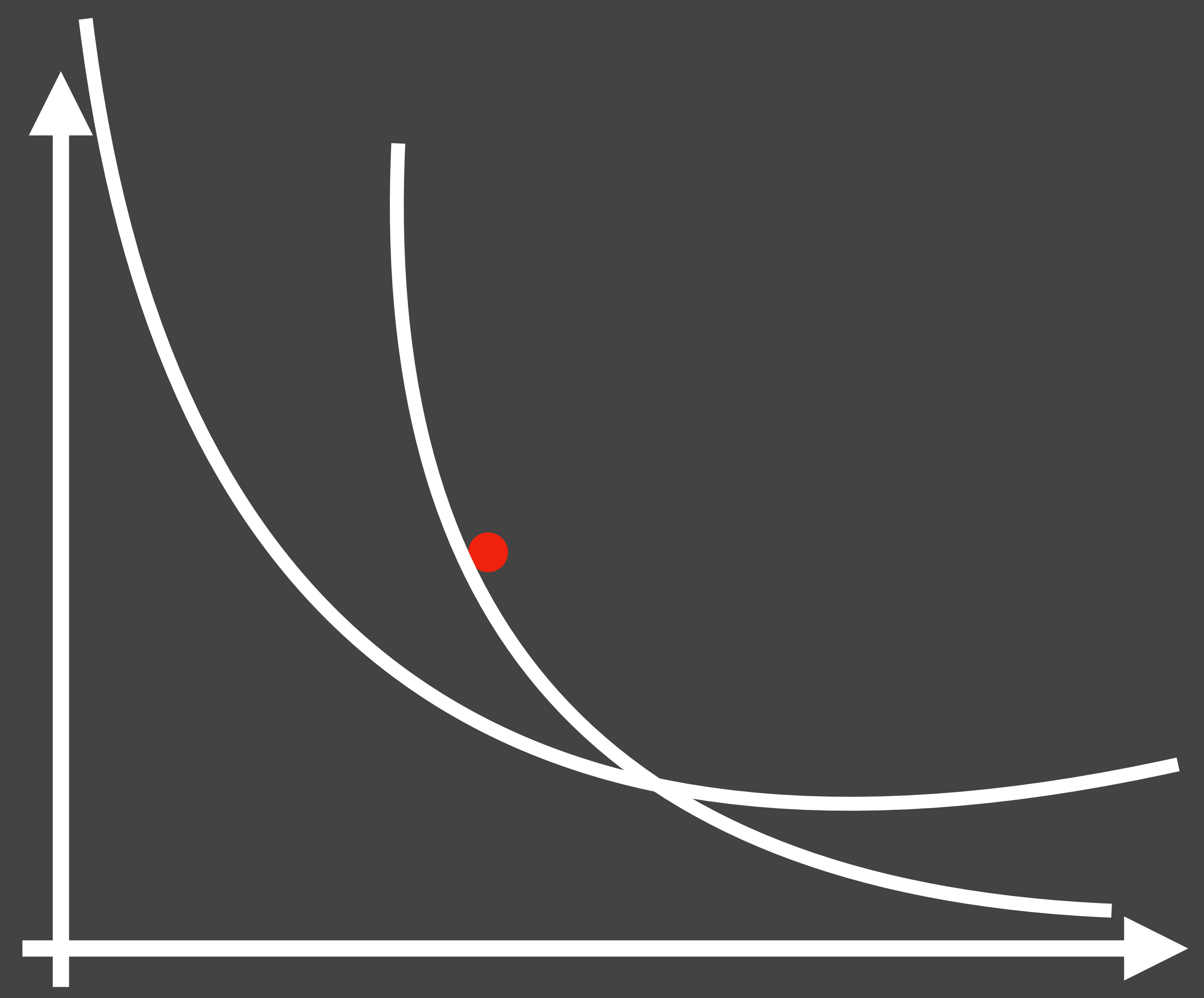
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



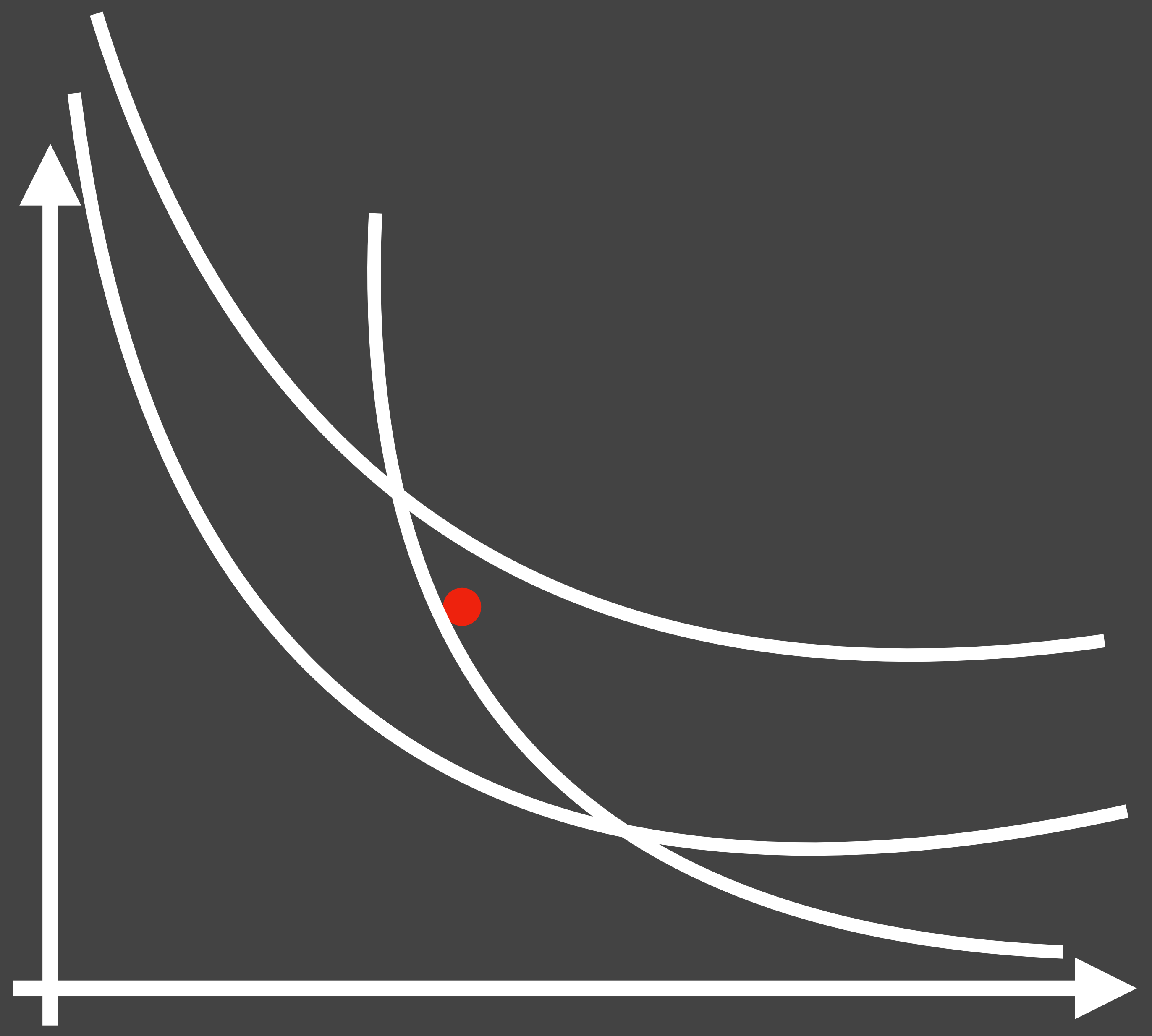
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



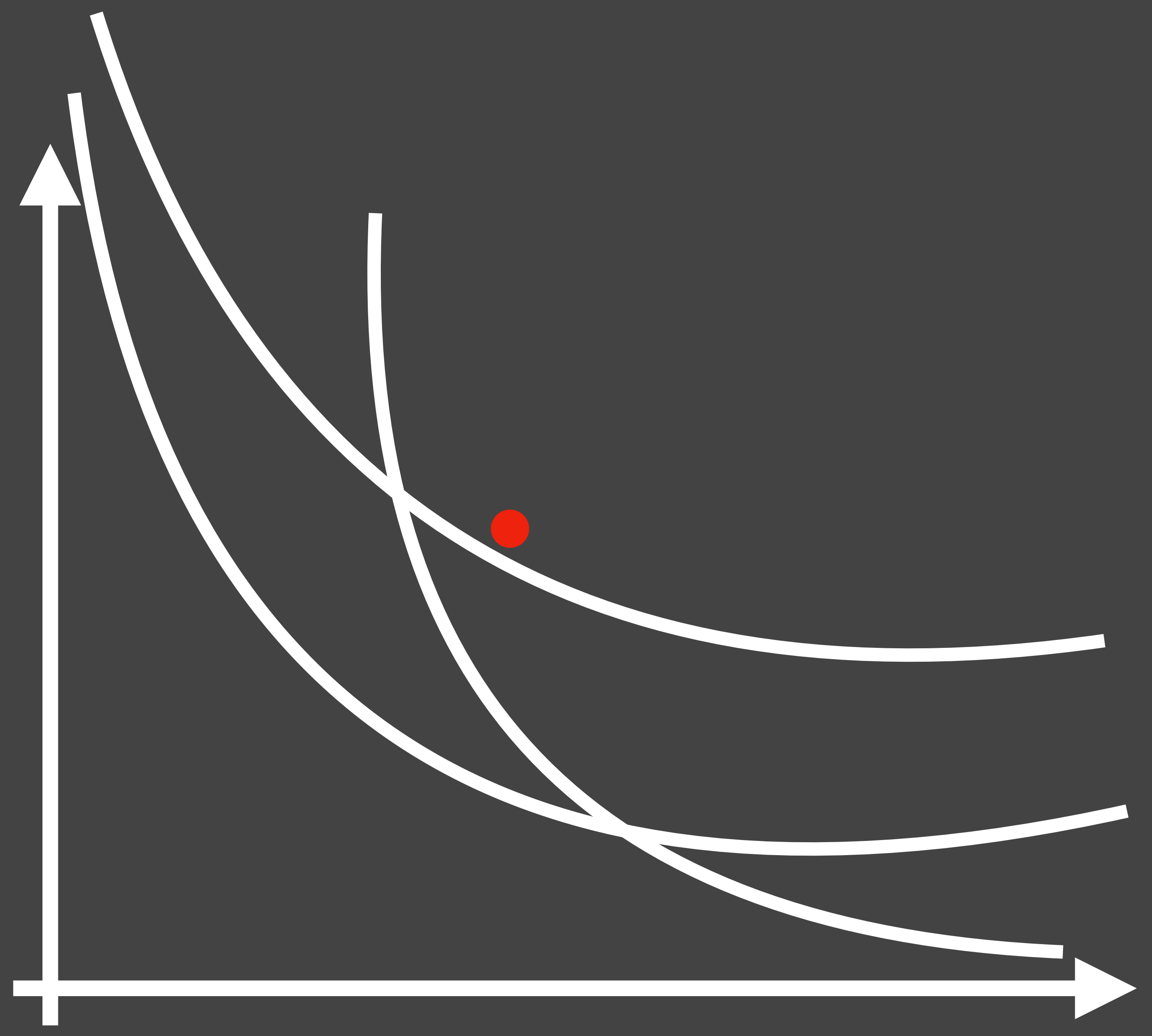
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**



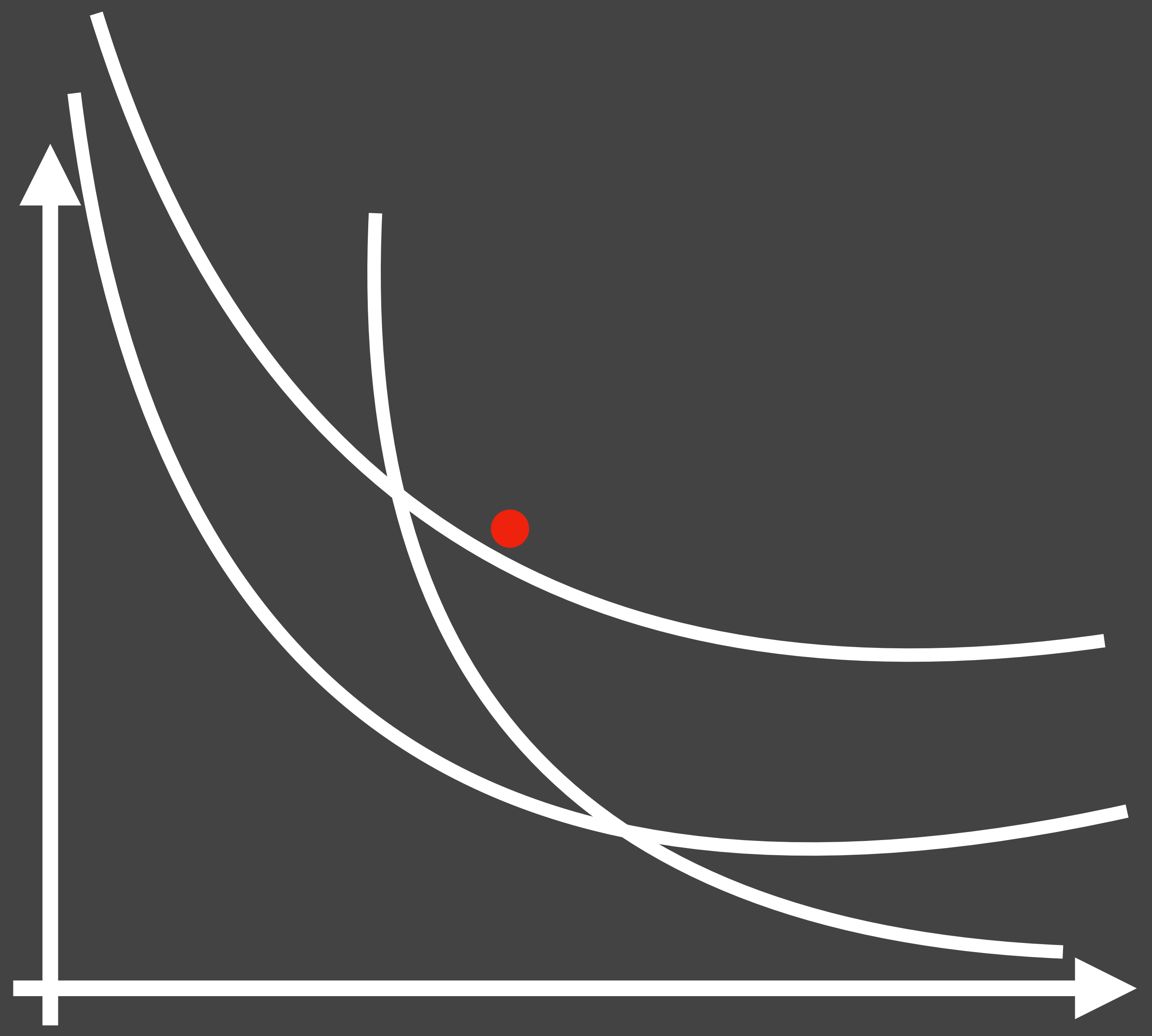
Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Recap so far

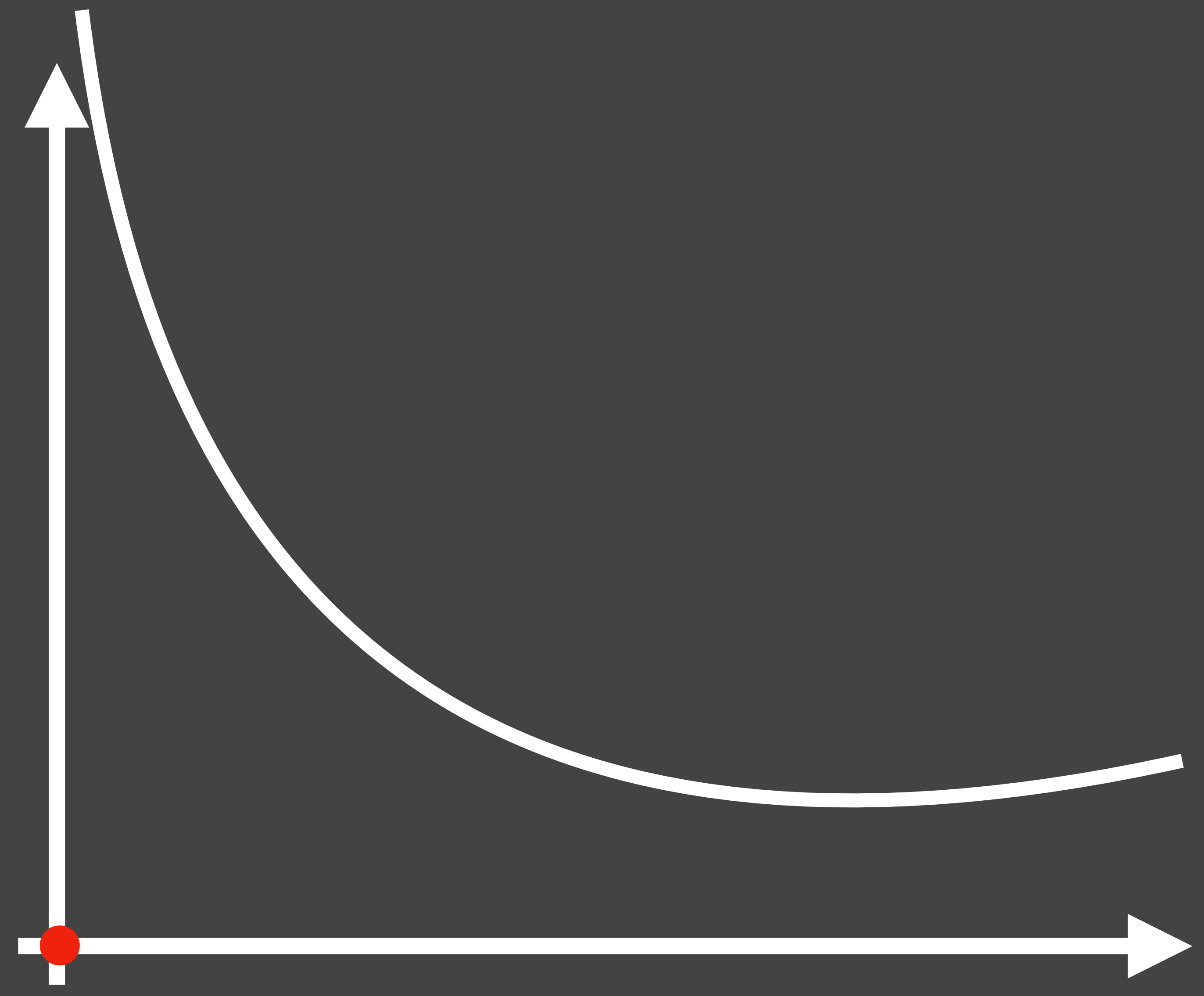
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

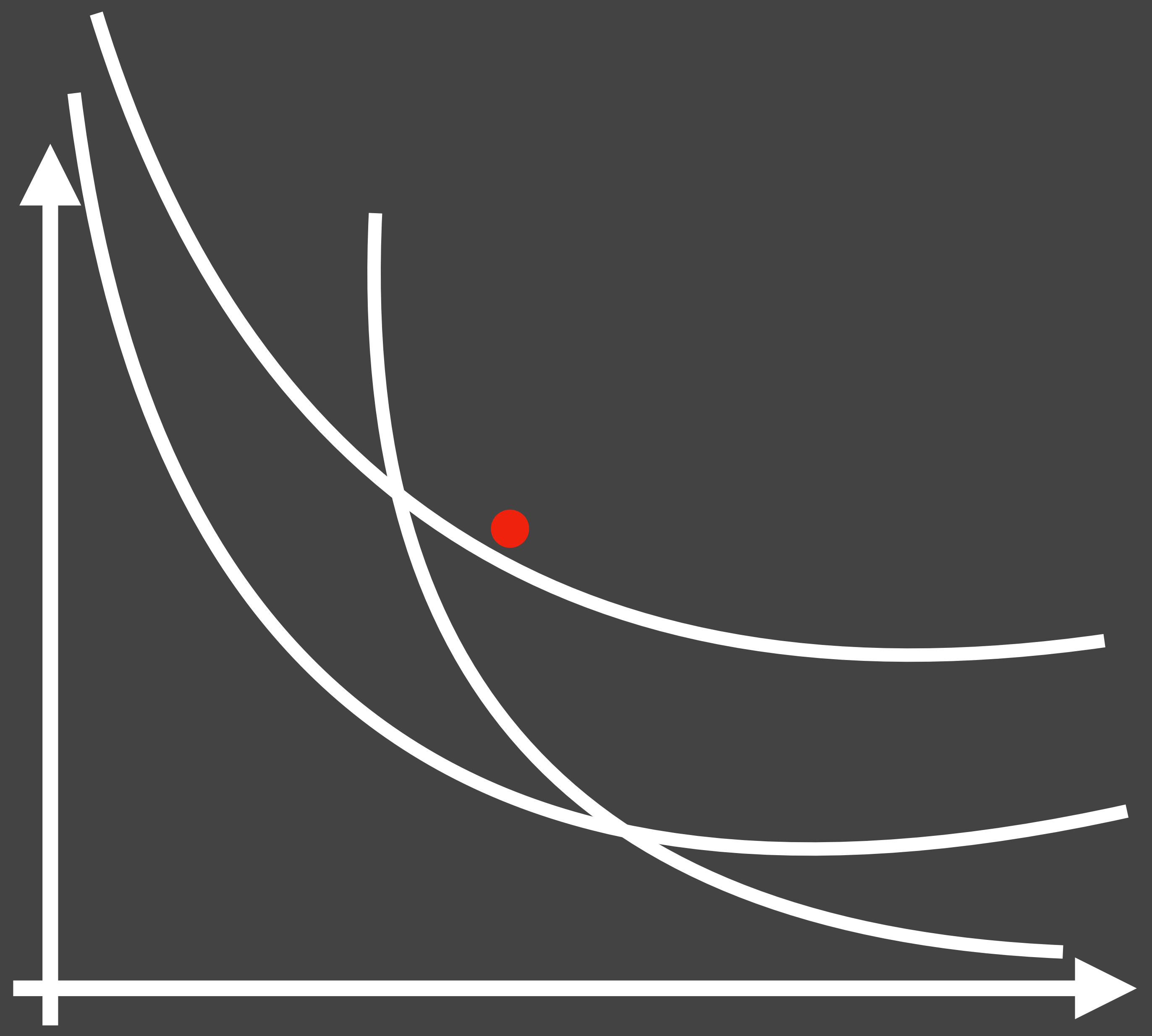
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

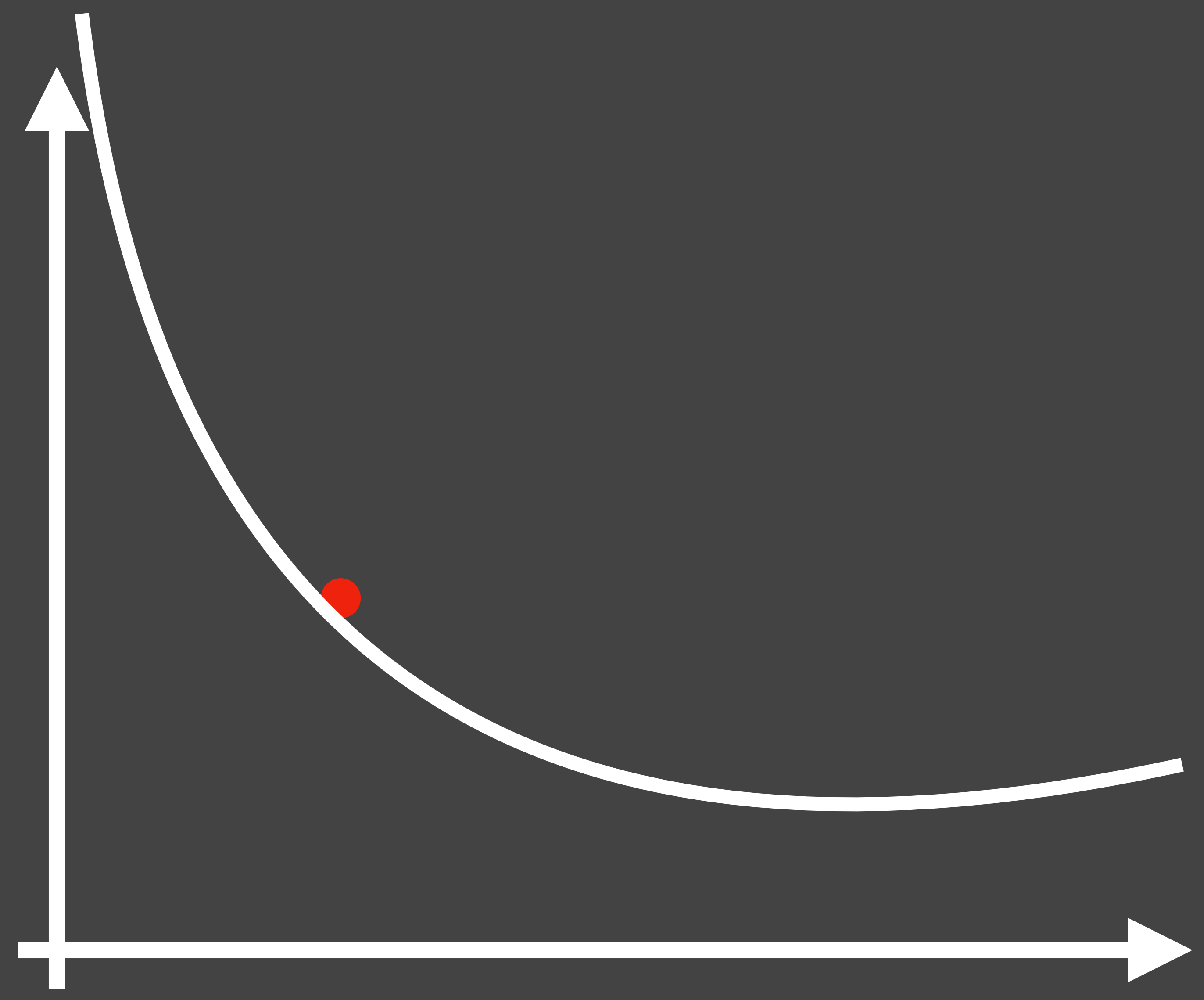
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

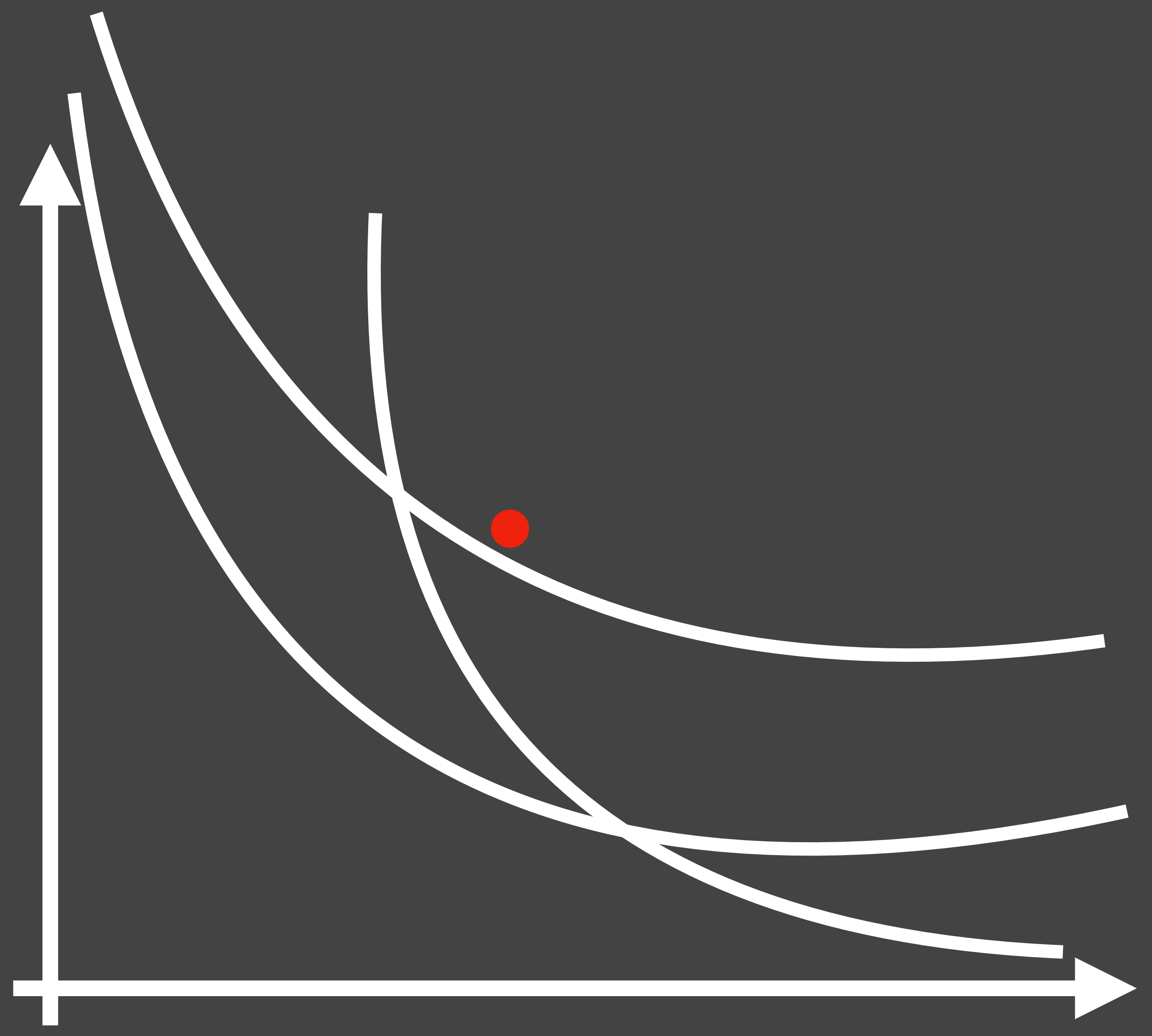
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

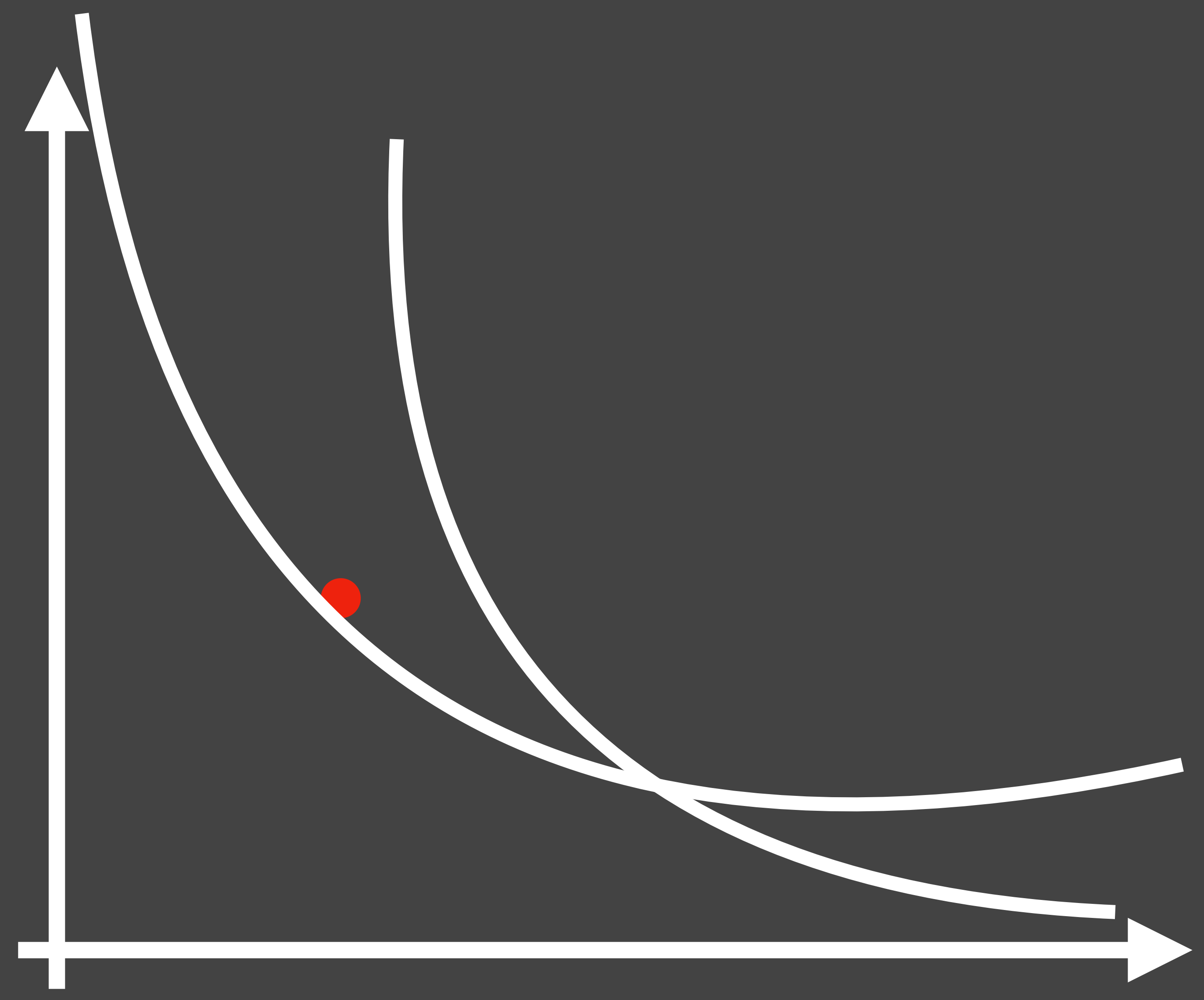
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

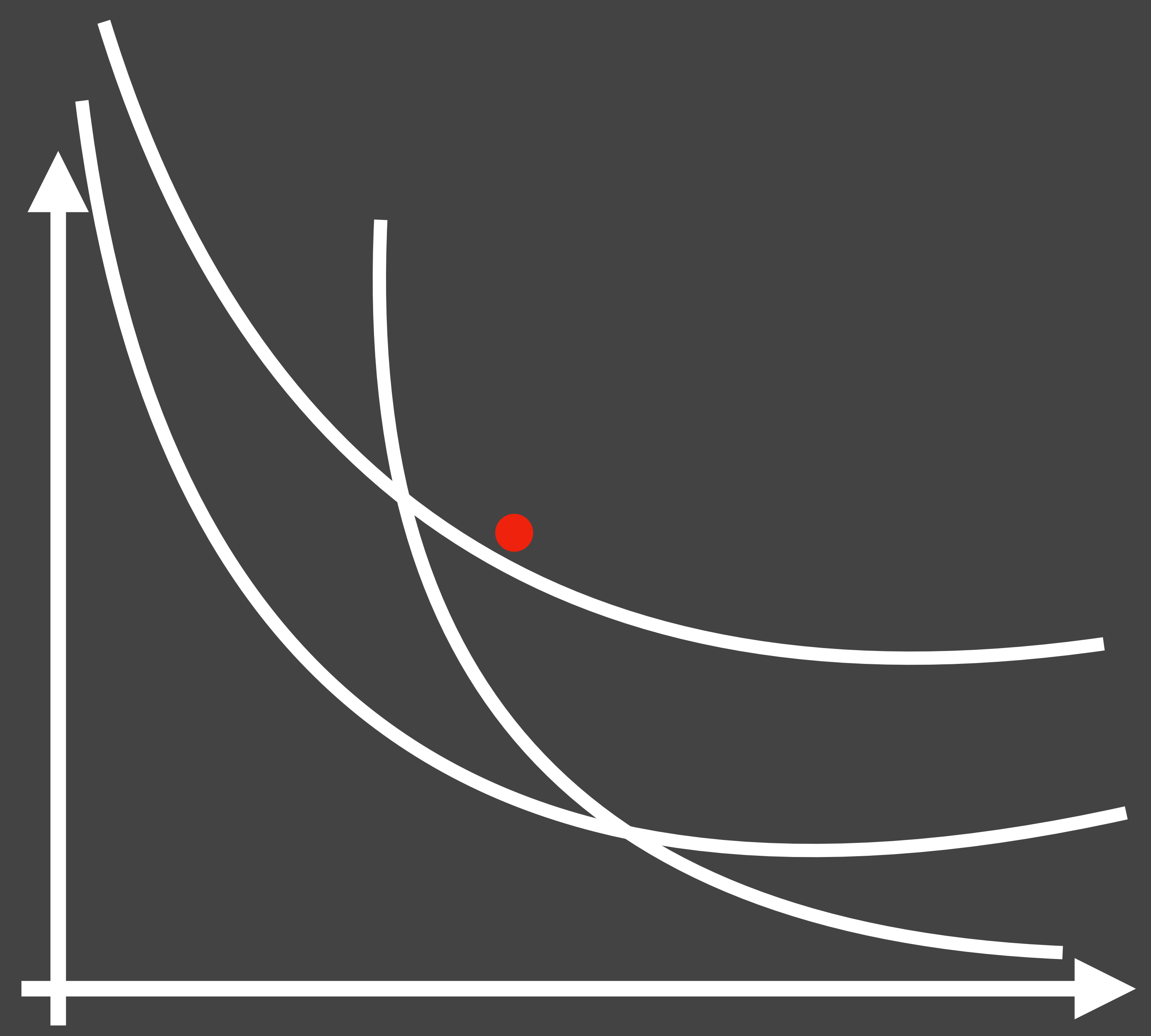
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

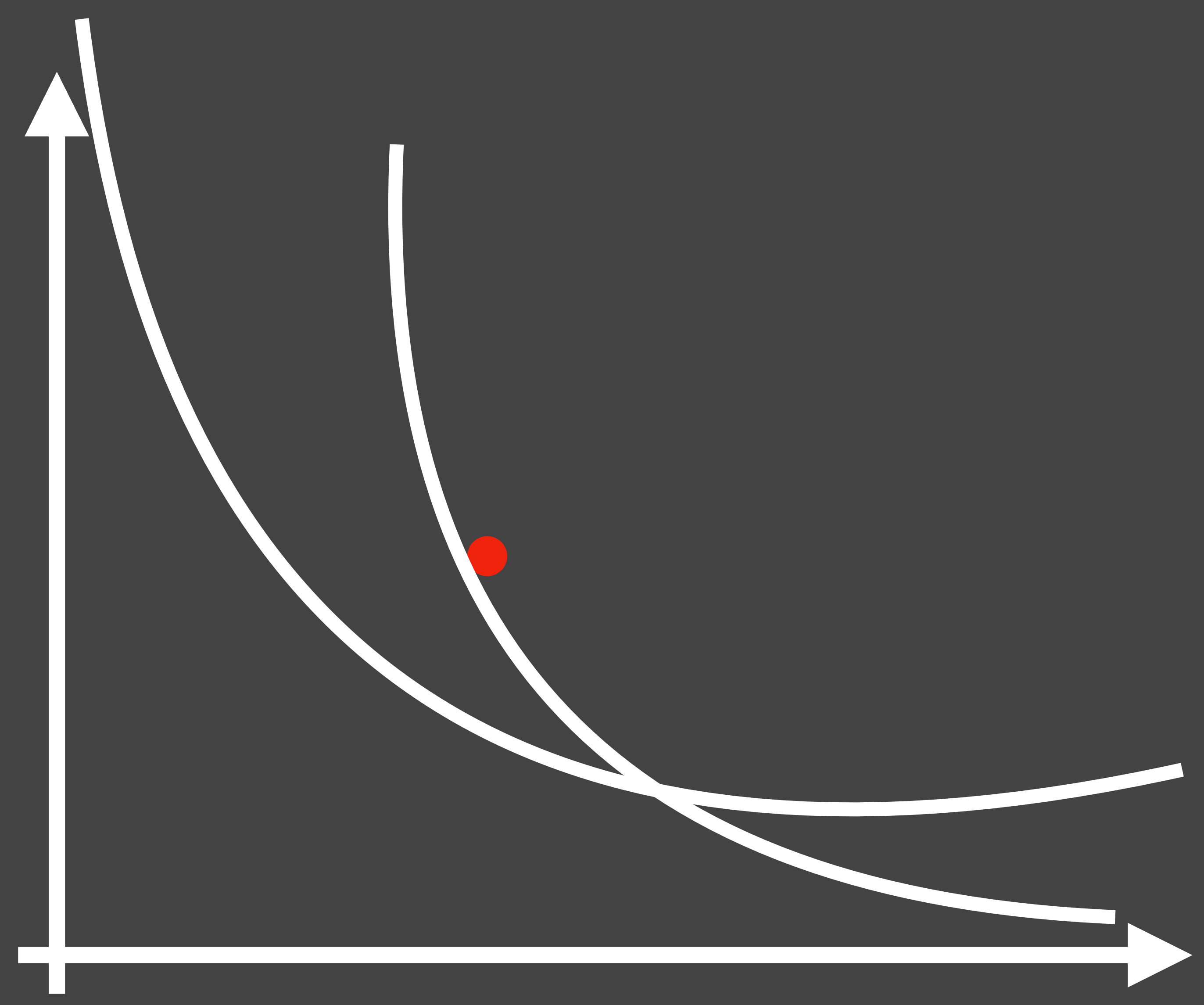
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

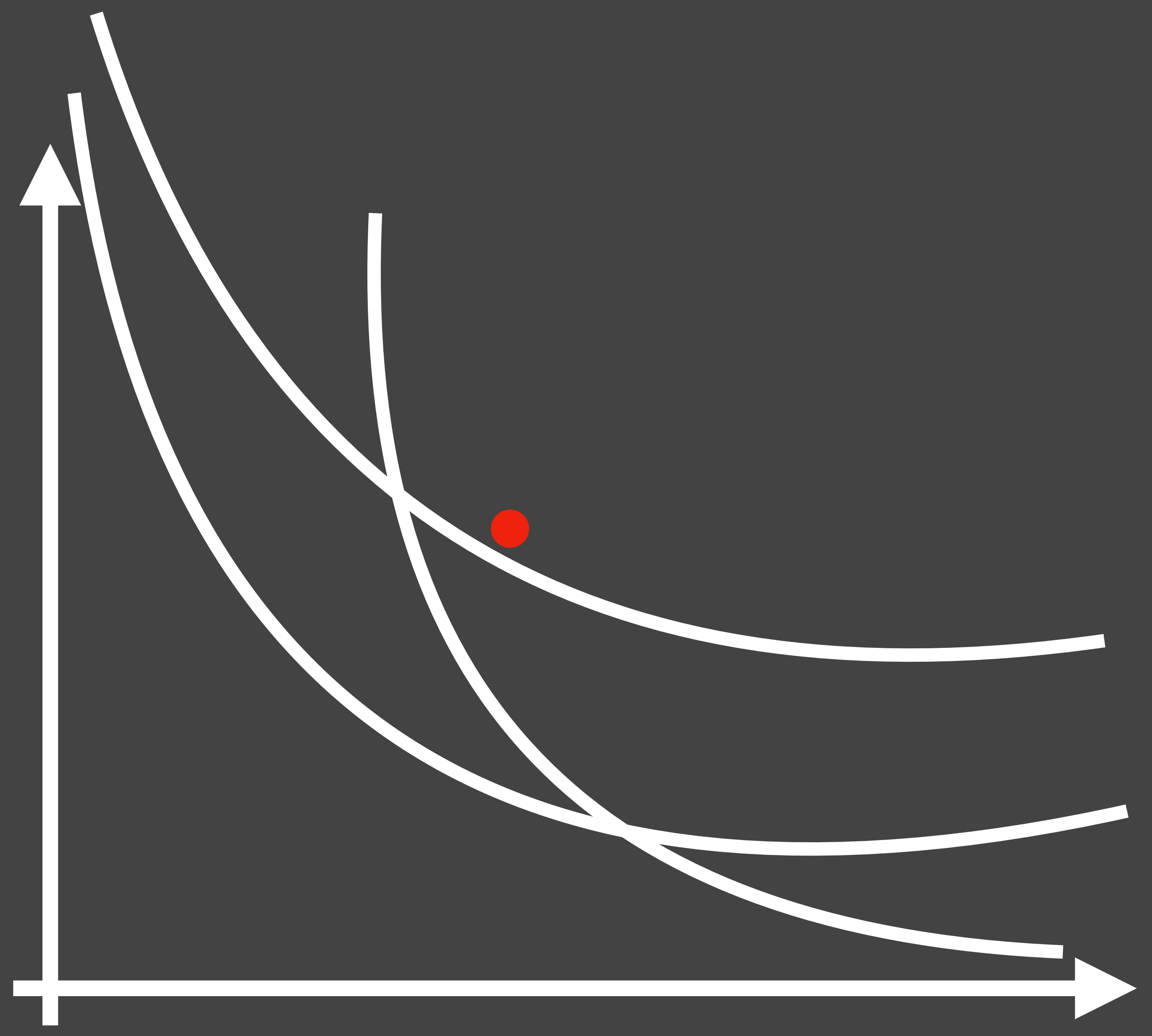
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

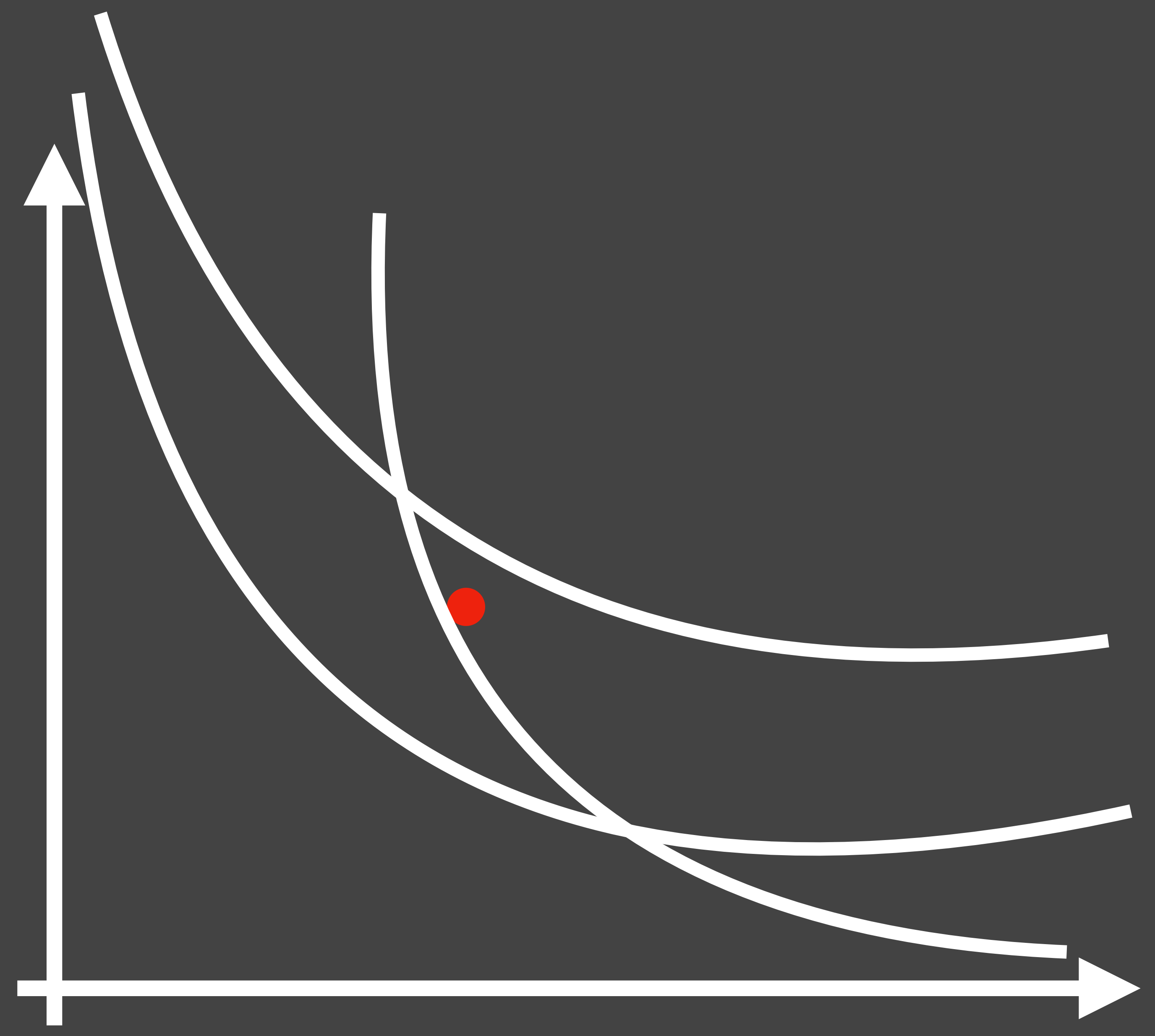
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

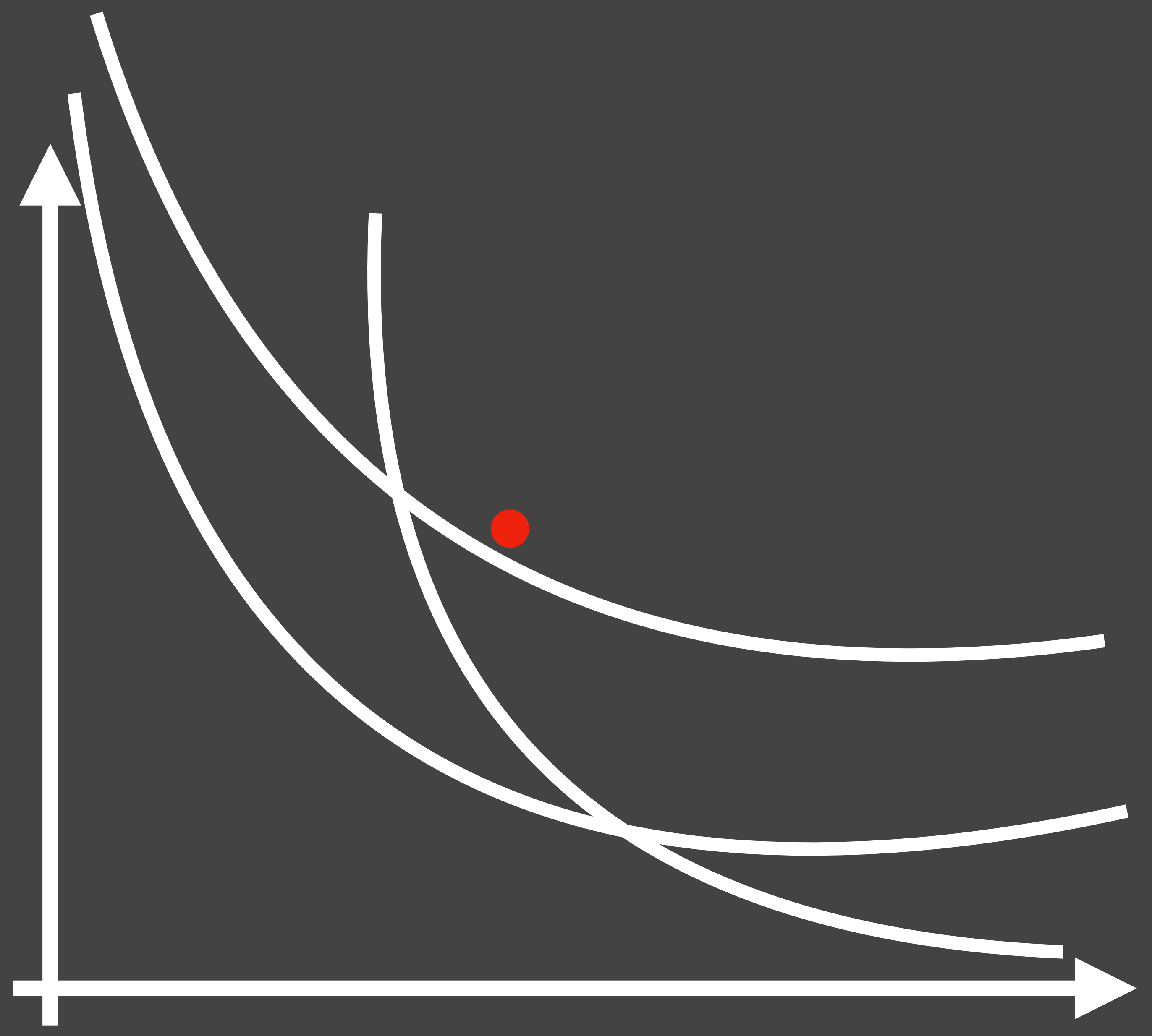
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

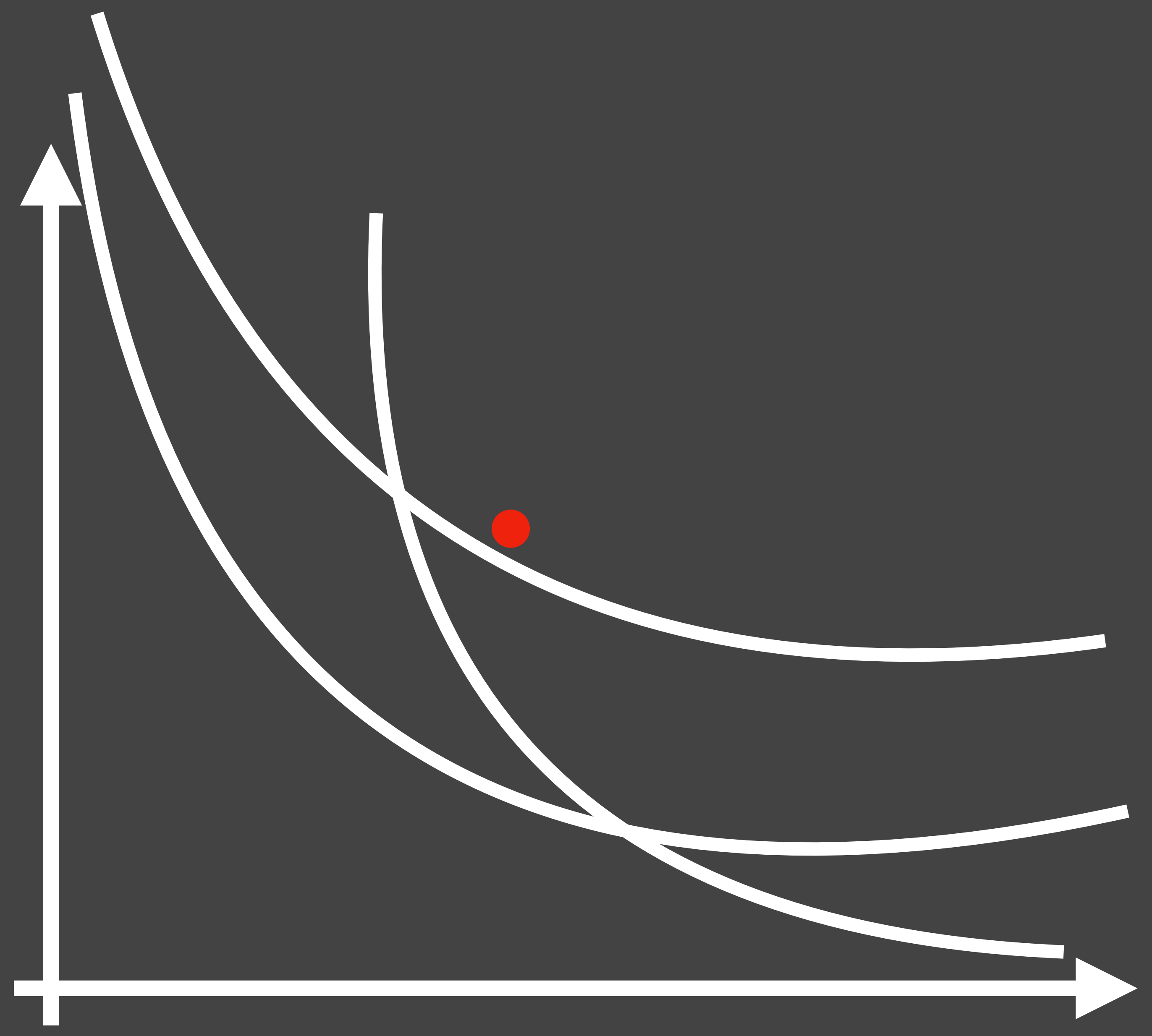
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

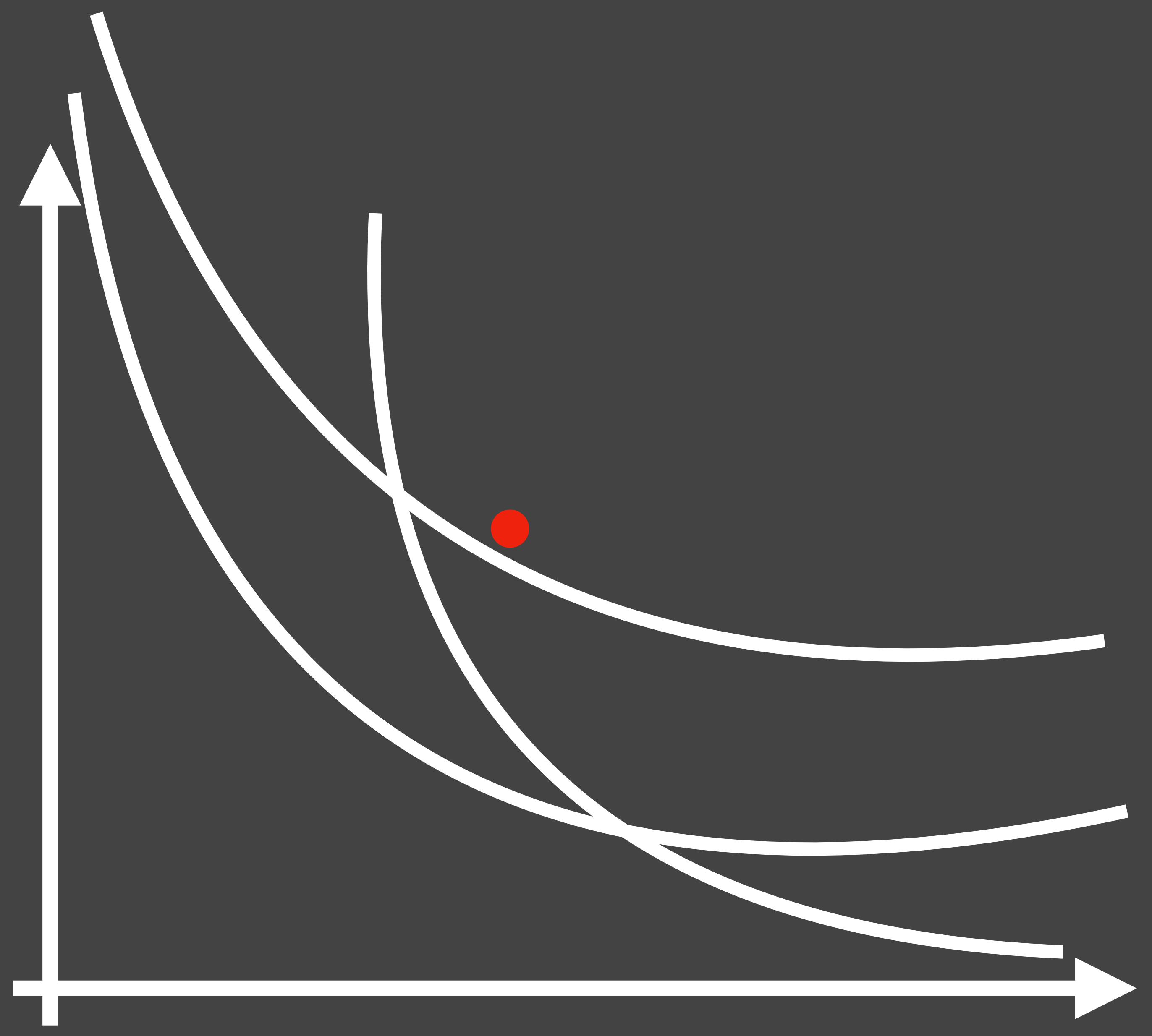
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

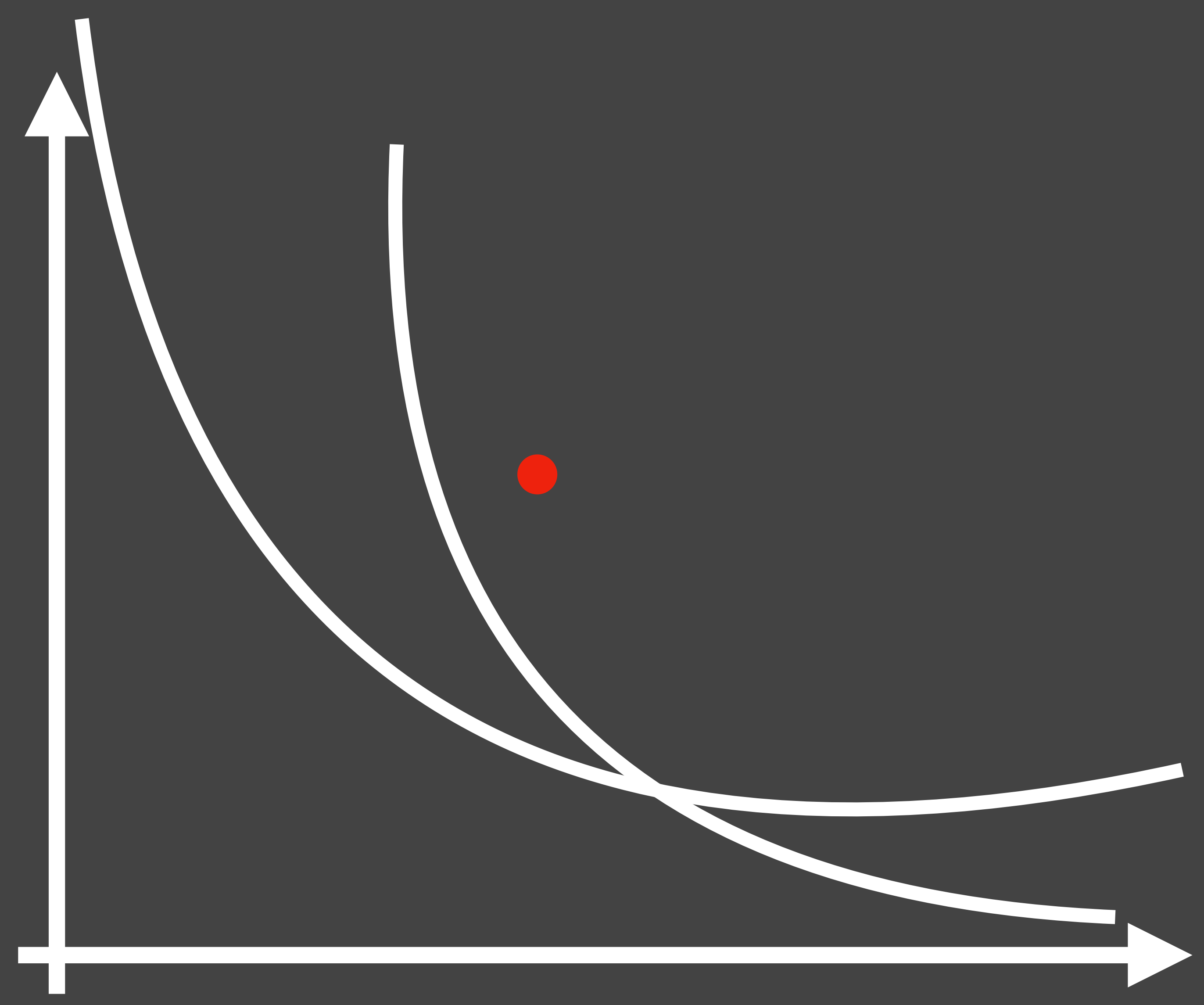
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

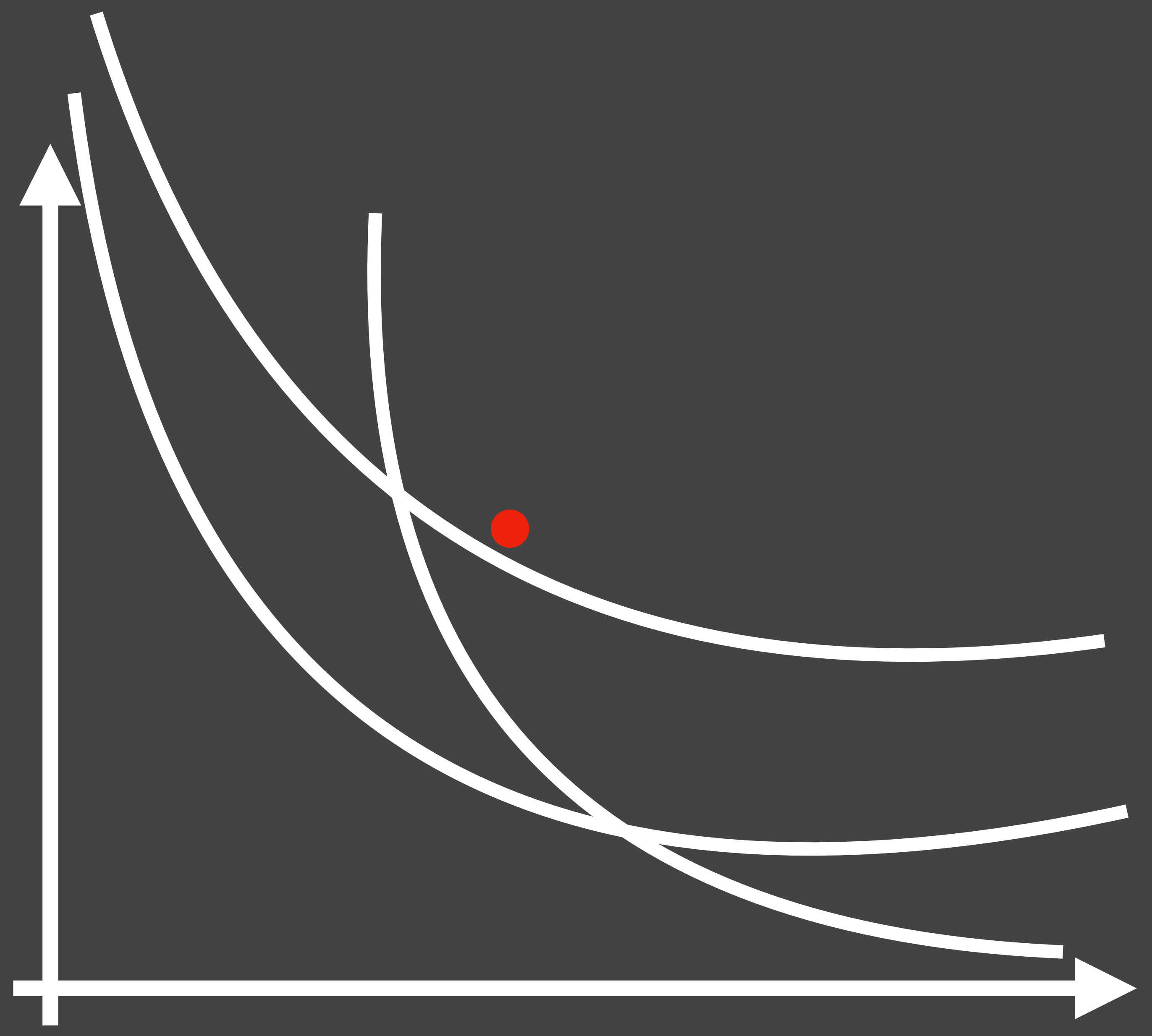
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

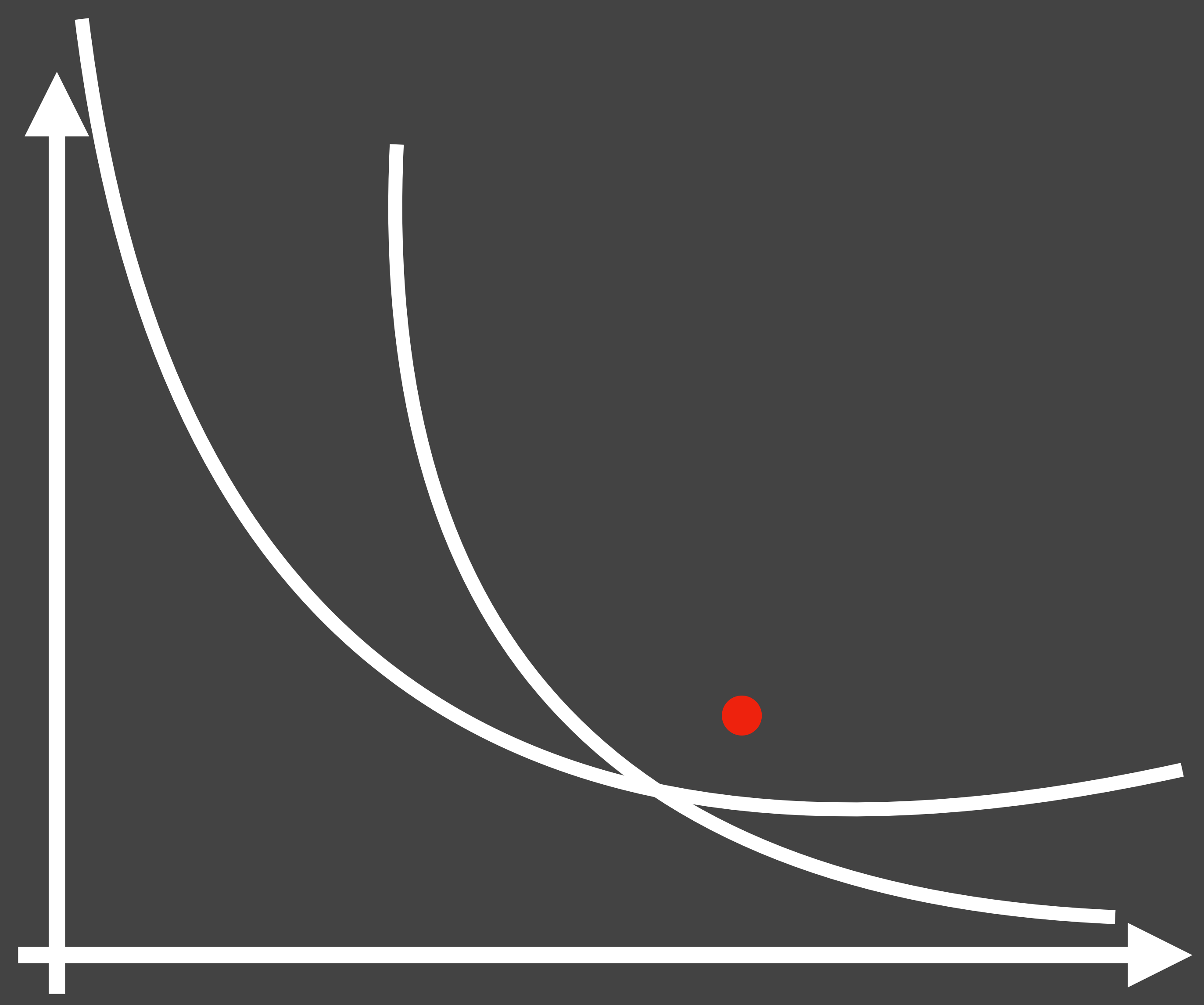
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

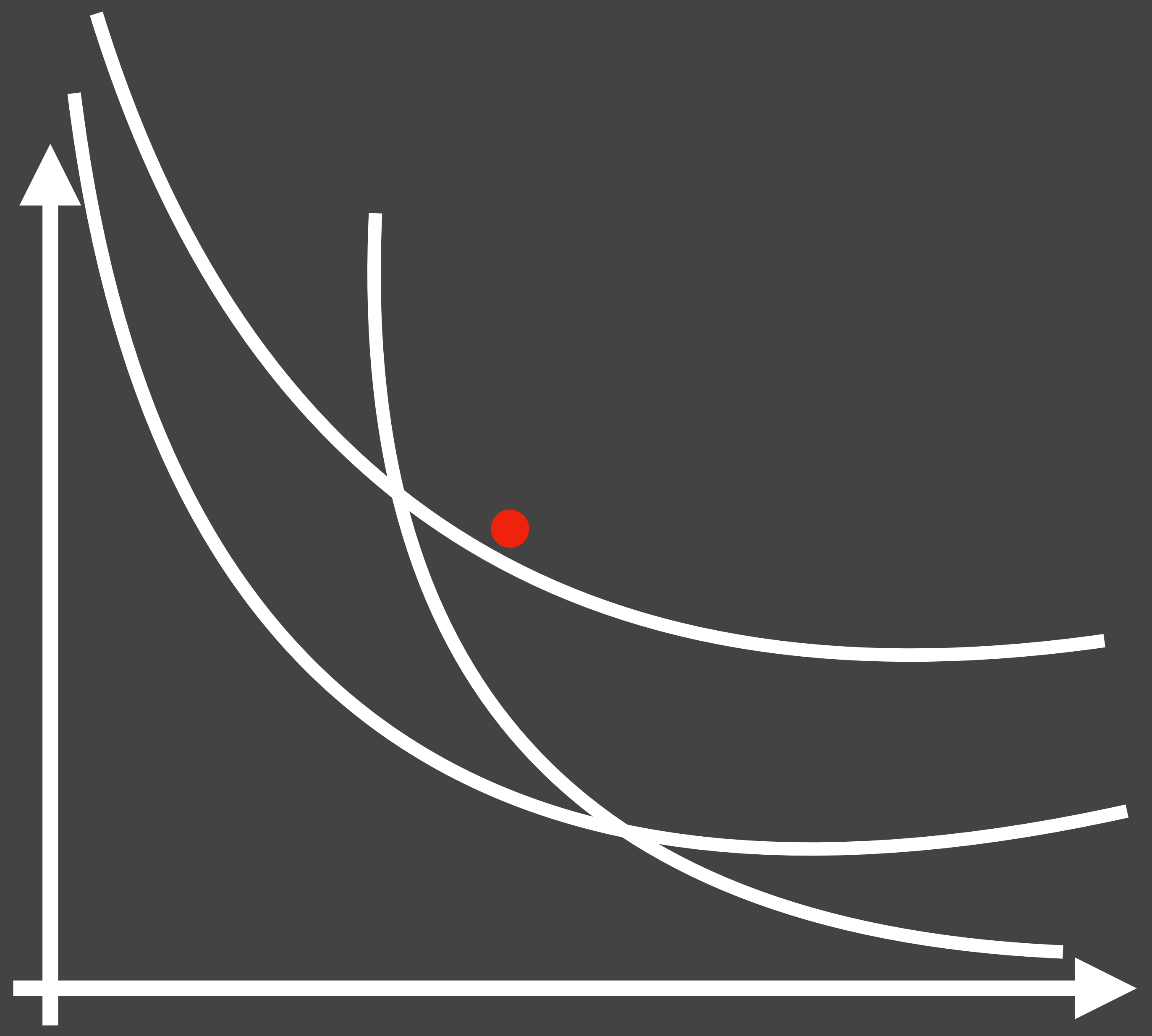
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

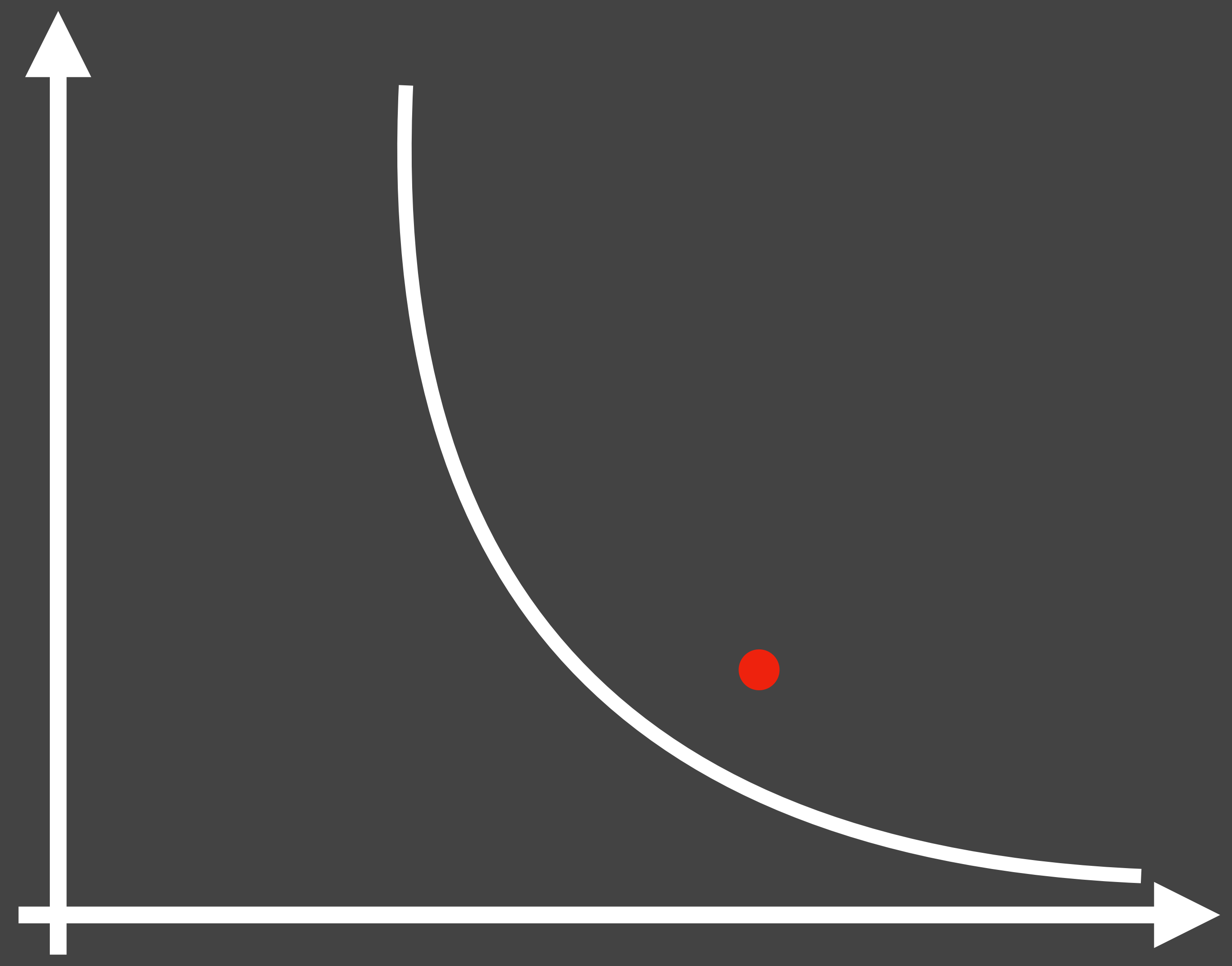
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

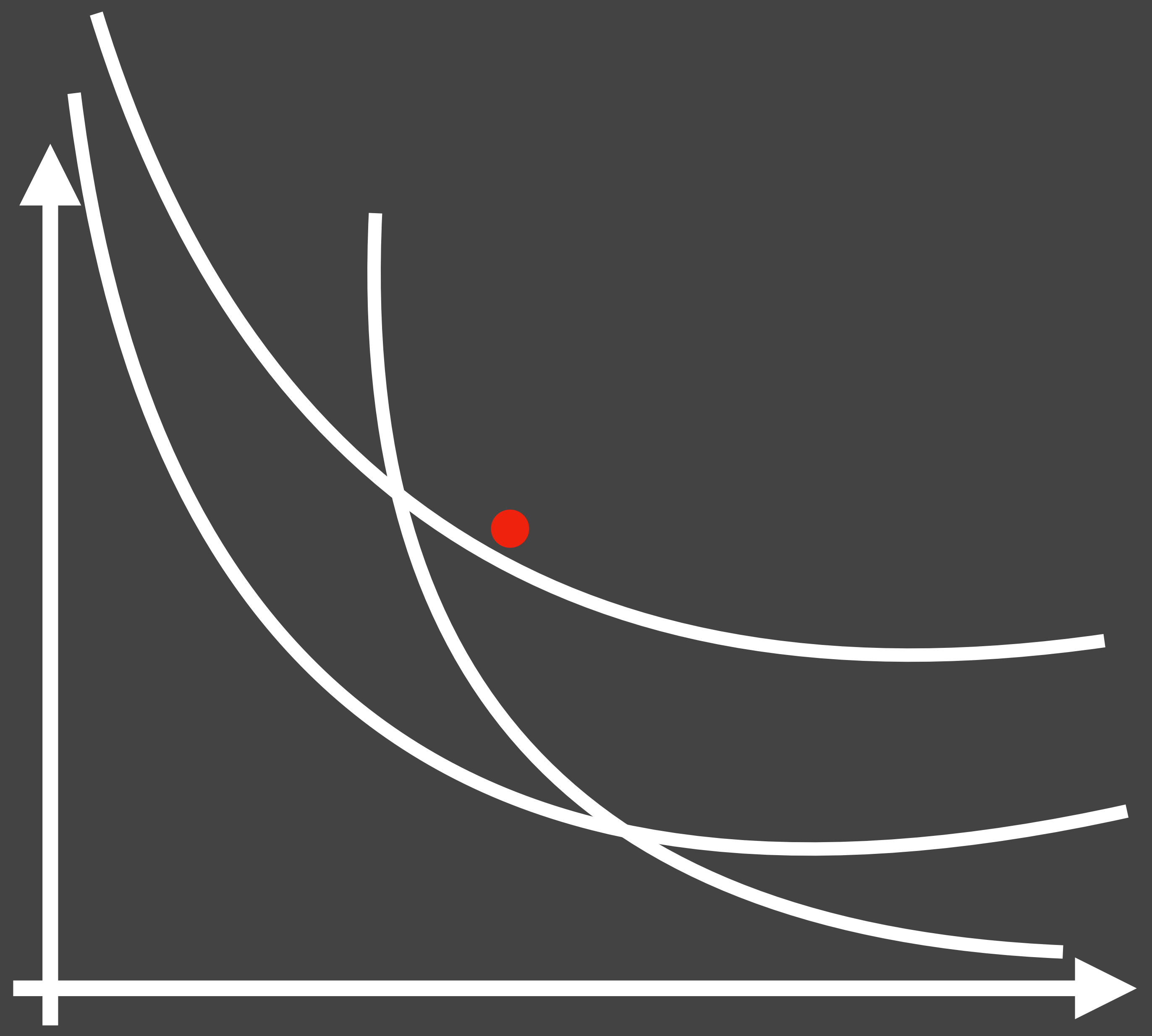
- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

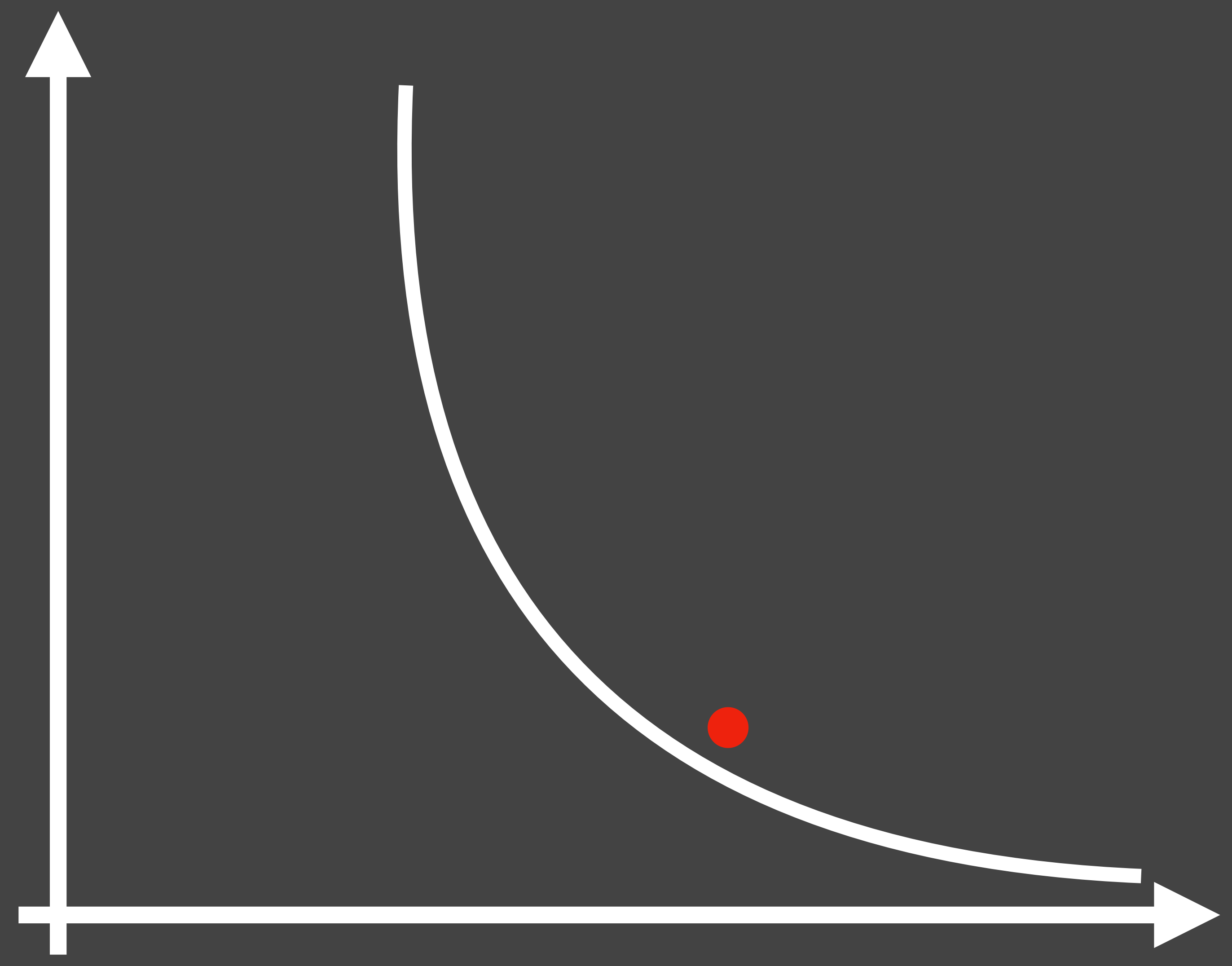
Part I: Online

- Inserts Only
- Decisions are **irrevocable**



Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**



Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**

Theorem (Online) [Gupta L. SODA 20]:

Competitive ratio $O(\log n \ \log F(\mathcal{N}))$.

Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Theorem (Dynamic) [Gupta L. FOCS 20]:

(i) *Competitive ratio* $O(\log F(\mathcal{N}))$.

(ii) *Average recourse* $\tilde{O}(f(\mathcal{N}))$.

Recap so far

Part I: Online

- Inserts Only
- Decisions are **irrevocable**

Theorem (Online) [Gupta L. SODA 20]:

Competitive ratio $O(\log n \; \log F(\mathcal{N}))$.

Part II: Fully- Dynamic

- Inserts + Deletes
- Want minimum # edits, a.k.a. **recourse.**

Theorem (Dynamic) [Gupta L. FOCS 20]:

(i) *Competitive ratio* $O(\log F(\mathcal{N}))$.

(ii) *Average recourse* $\tilde{O}(f(\mathcal{N}))$.

Modeling power of Submod Cover + robustness to uncertainty of Online/Dynamic algos.

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

Part III — Random Order Online Set Cover

Conclusion

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

Part III — Random Order Online Set Cover

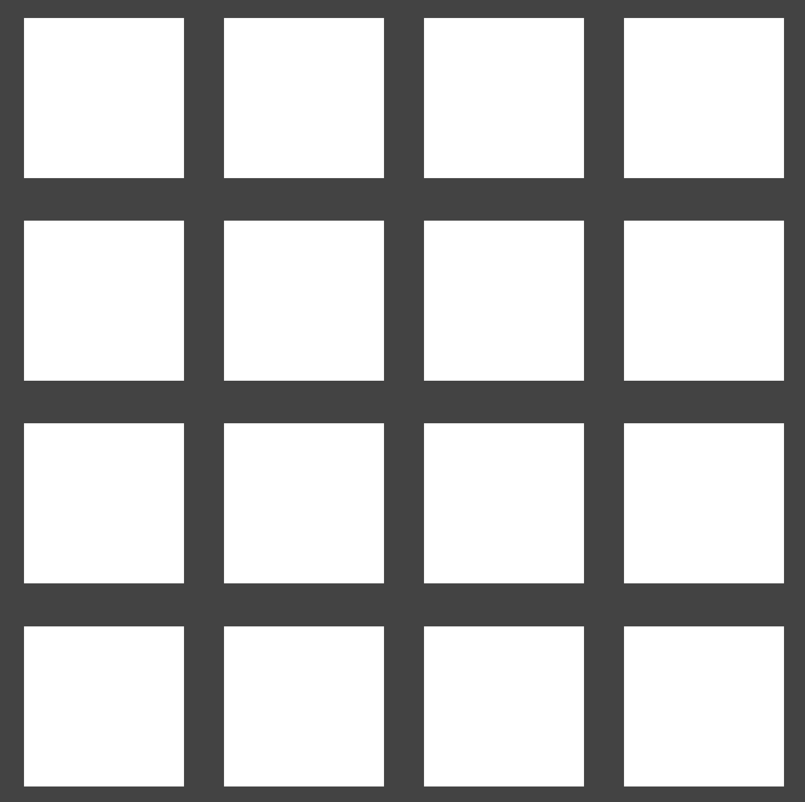
Conclusion

Part II — Application: Block-Aware Caching

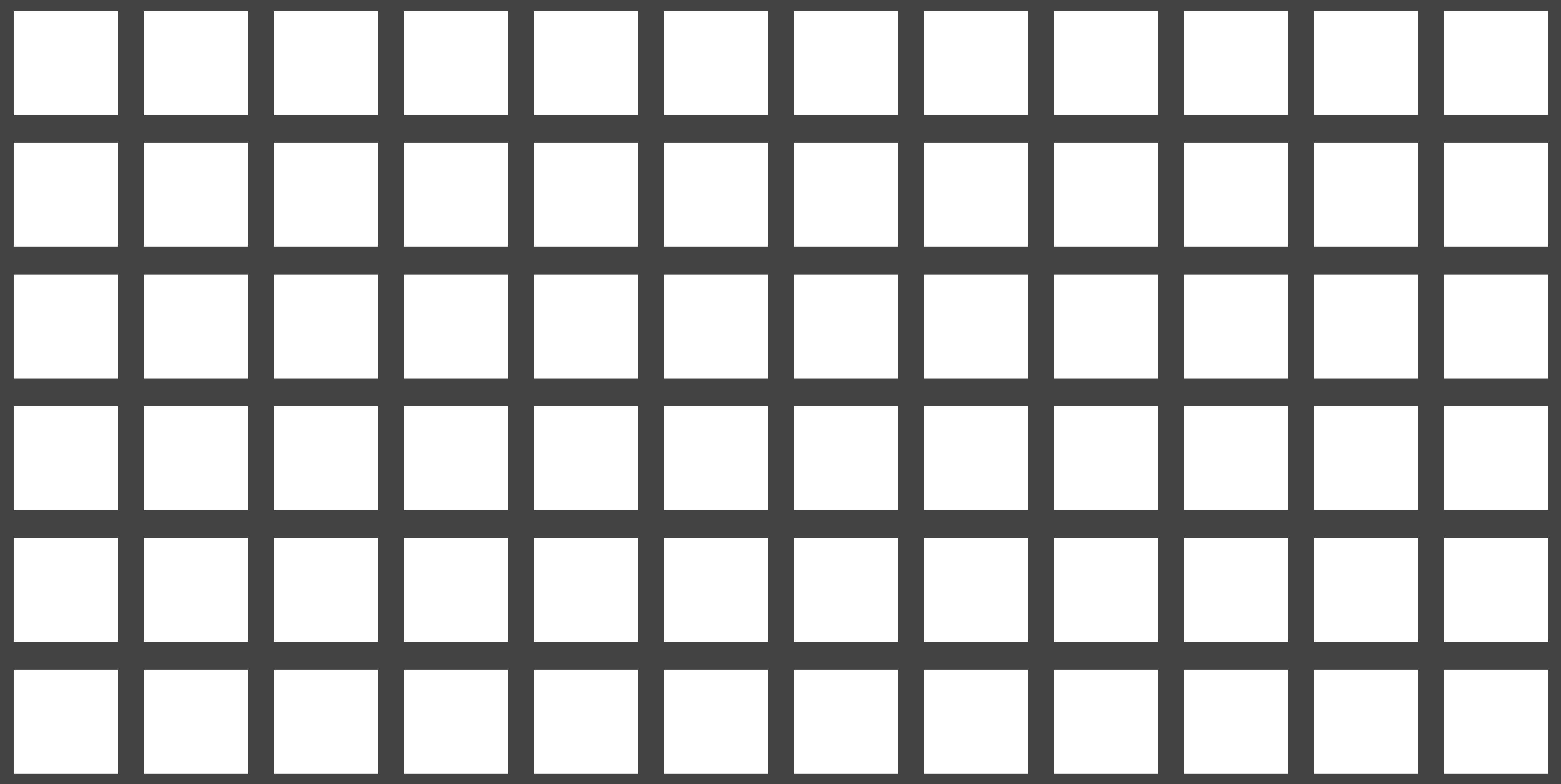
with Christian Coester, Seffi Naor, Ohad Talmon

Classic Caching

Cache of size k

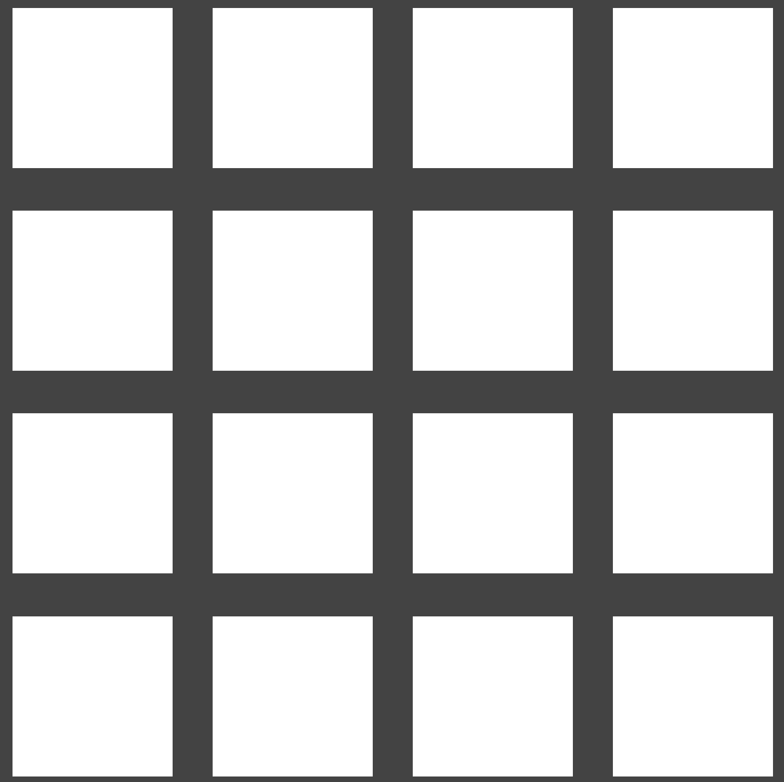


n total pages

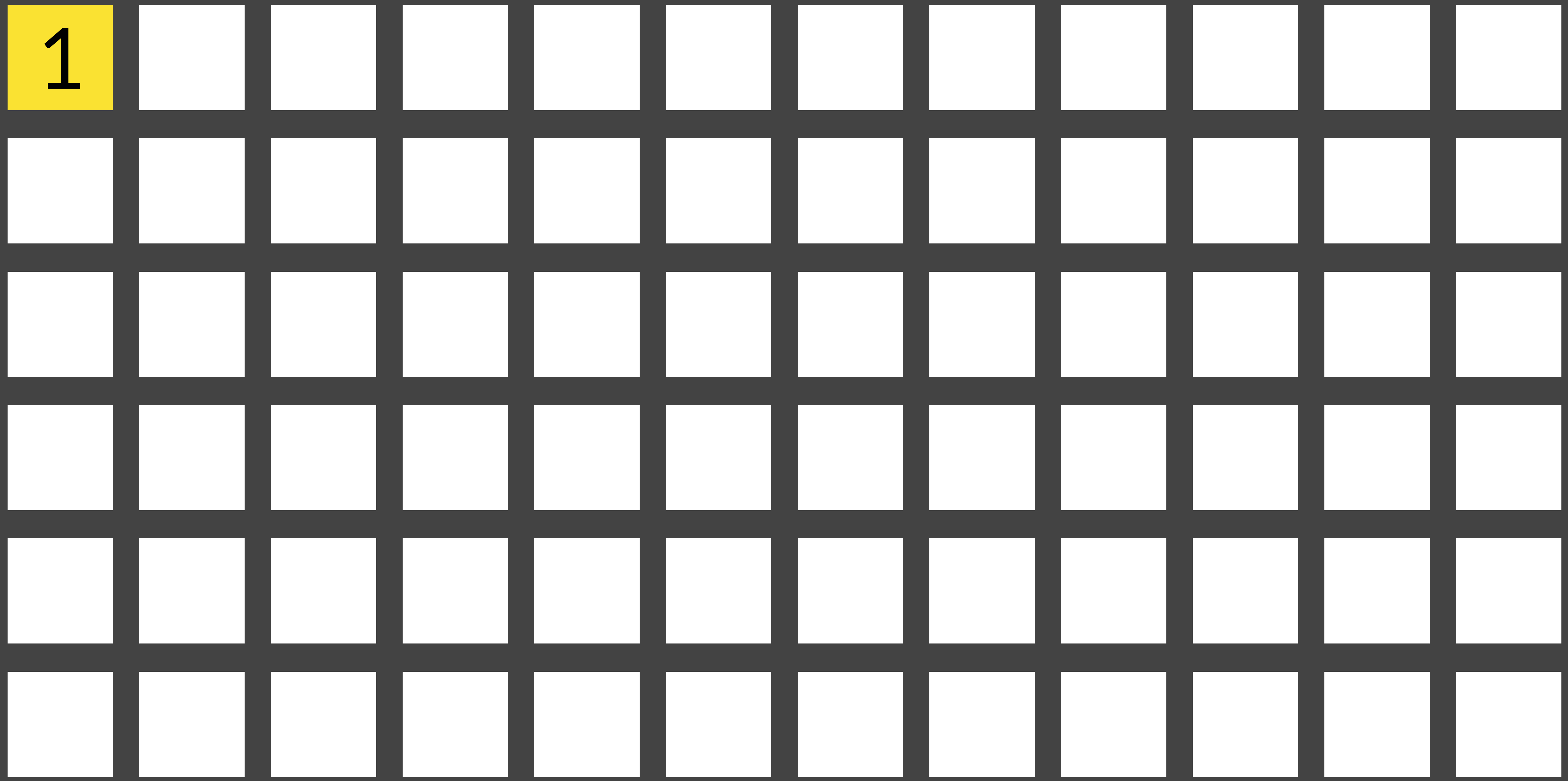


Classic Caching

Cache of size k

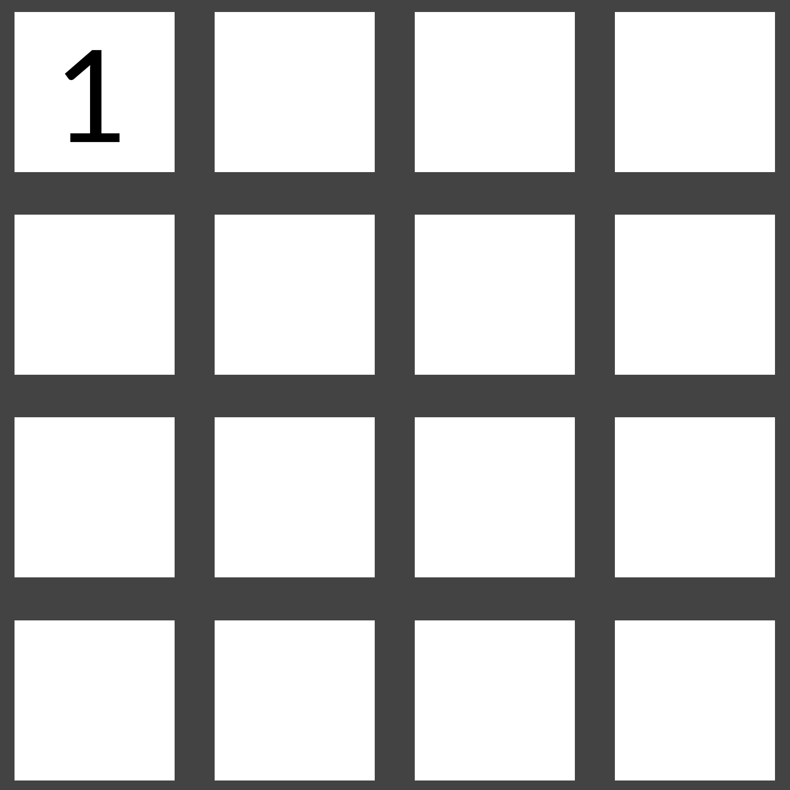


n total pages

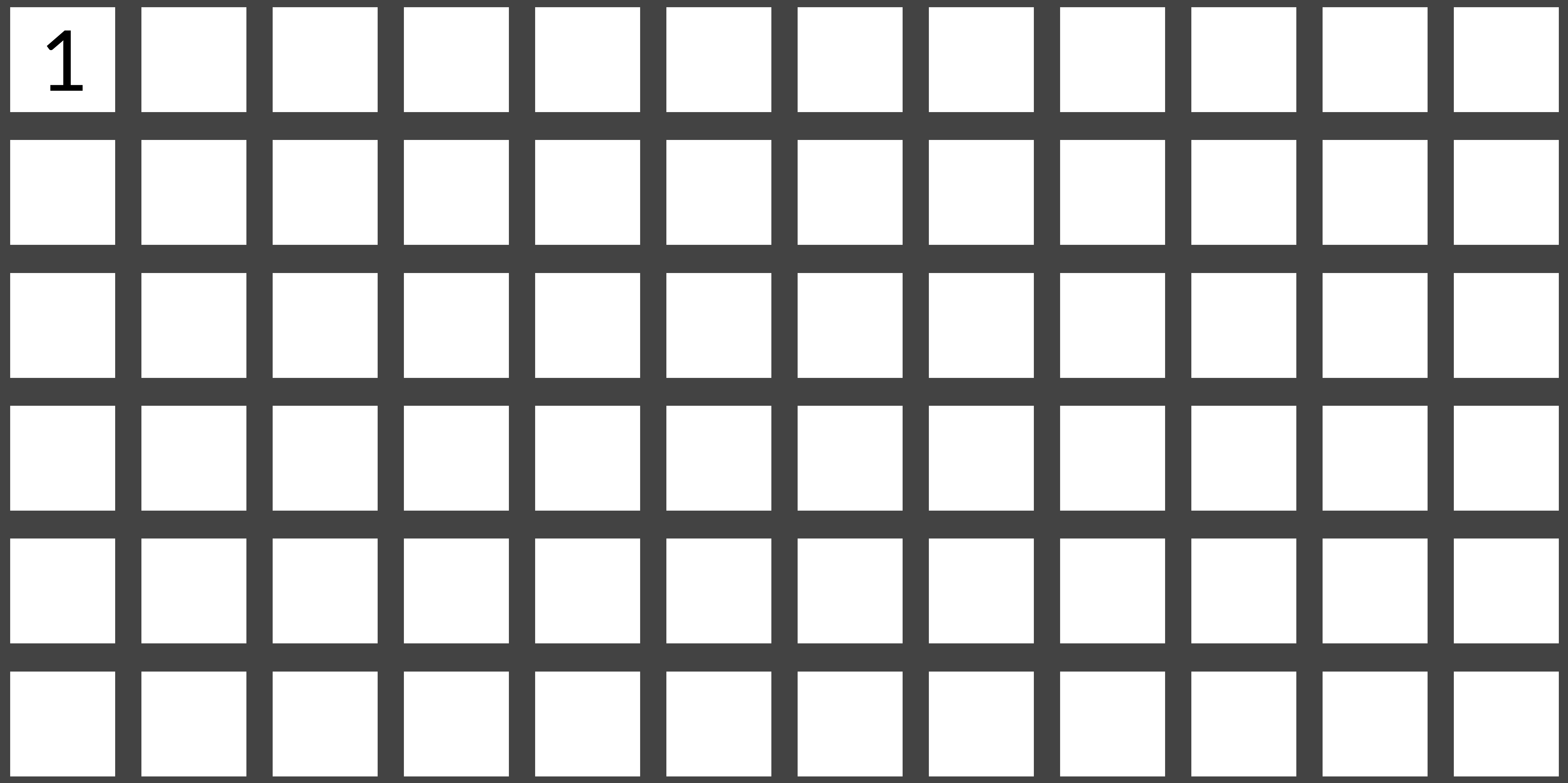


Classic Caching

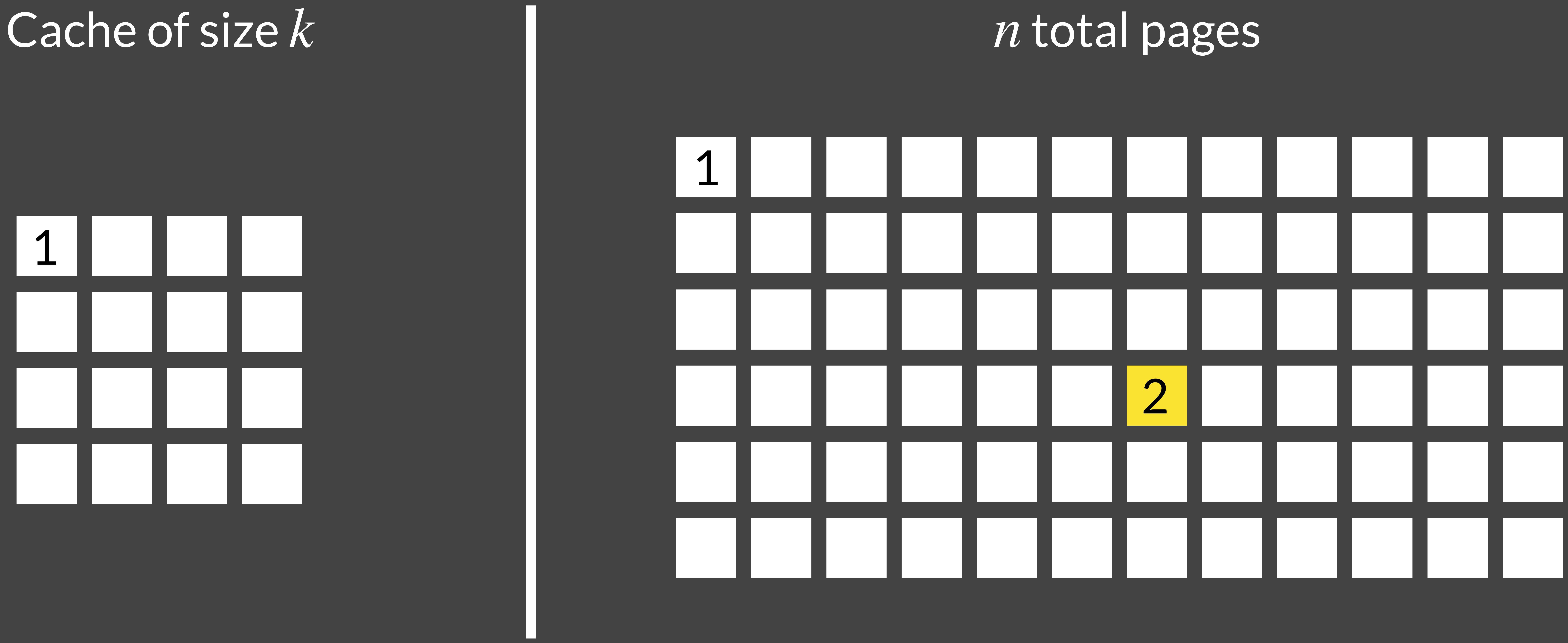
Cache of size k



n total pages

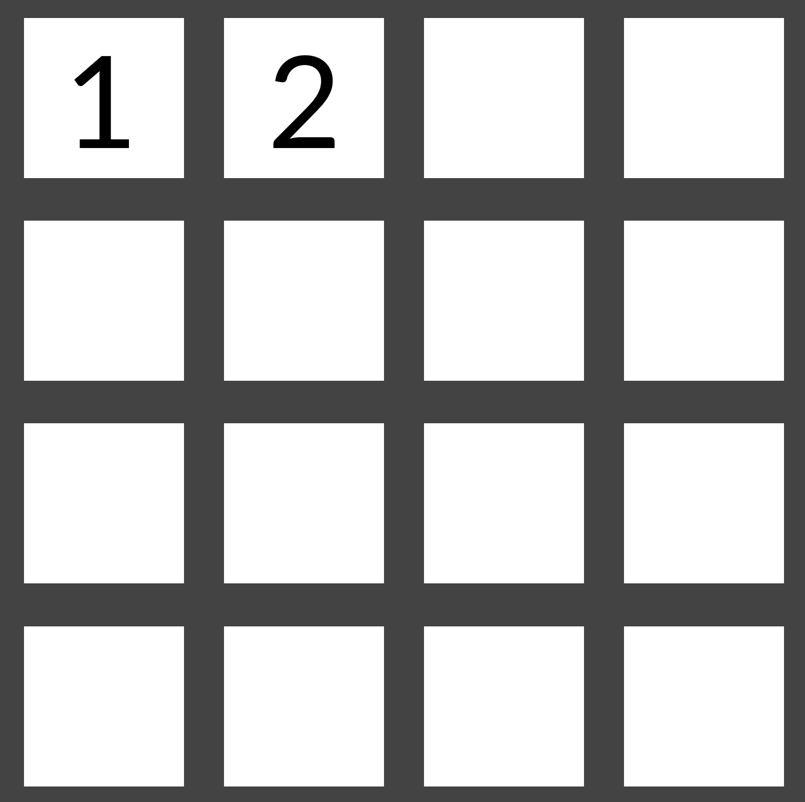


Classic Caching

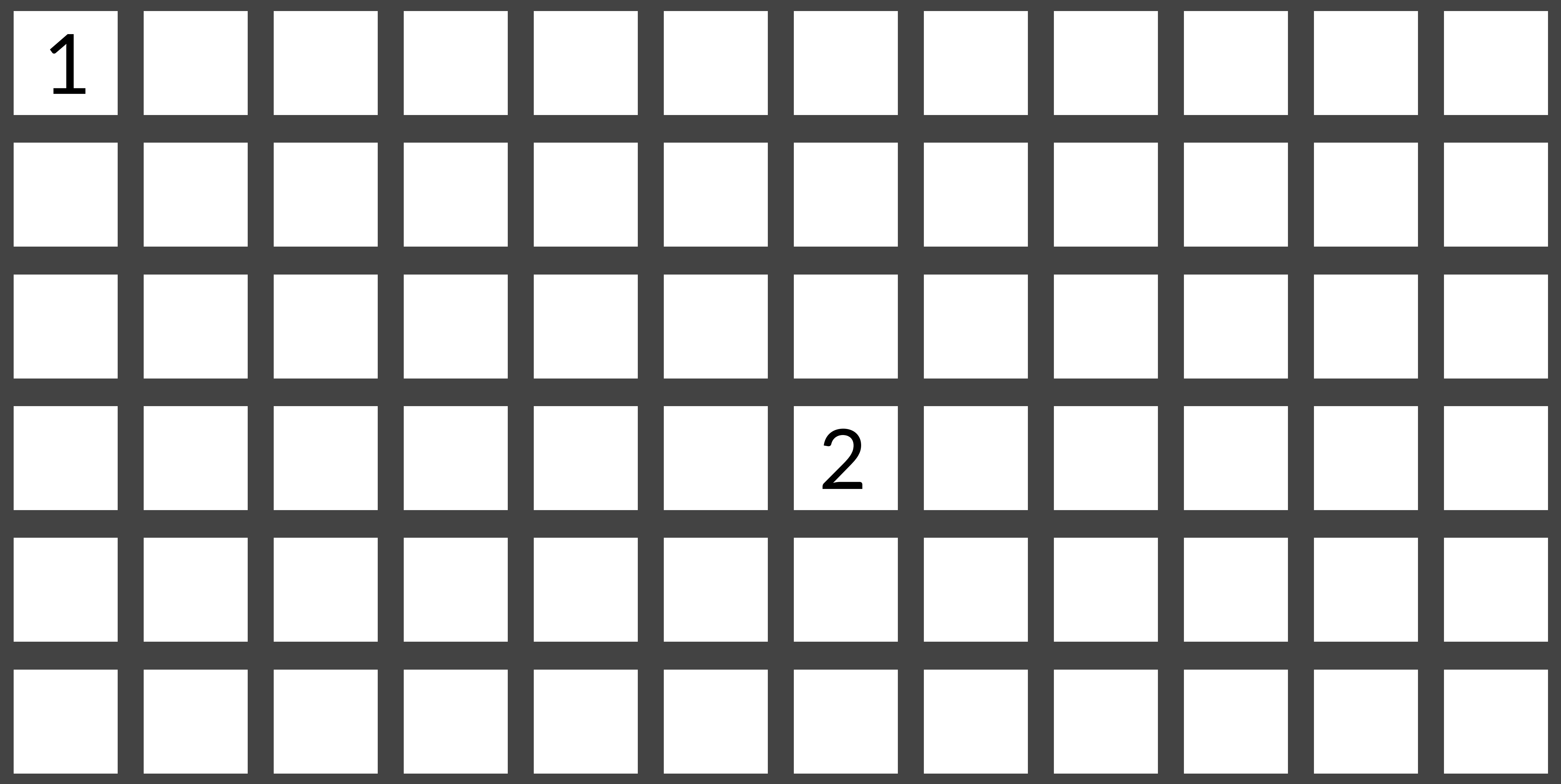


Classic Caching

Cache of size k

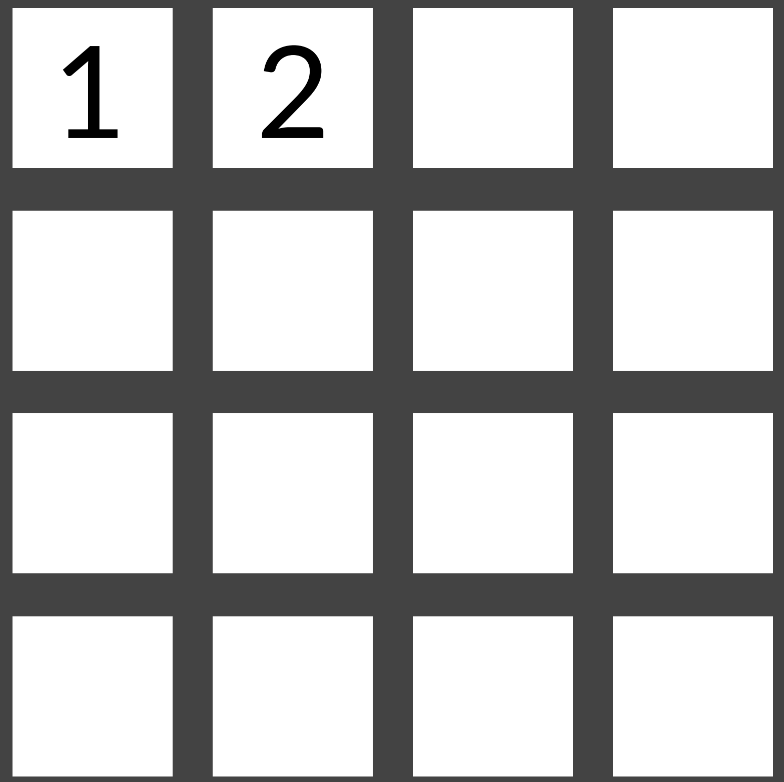


n total pages

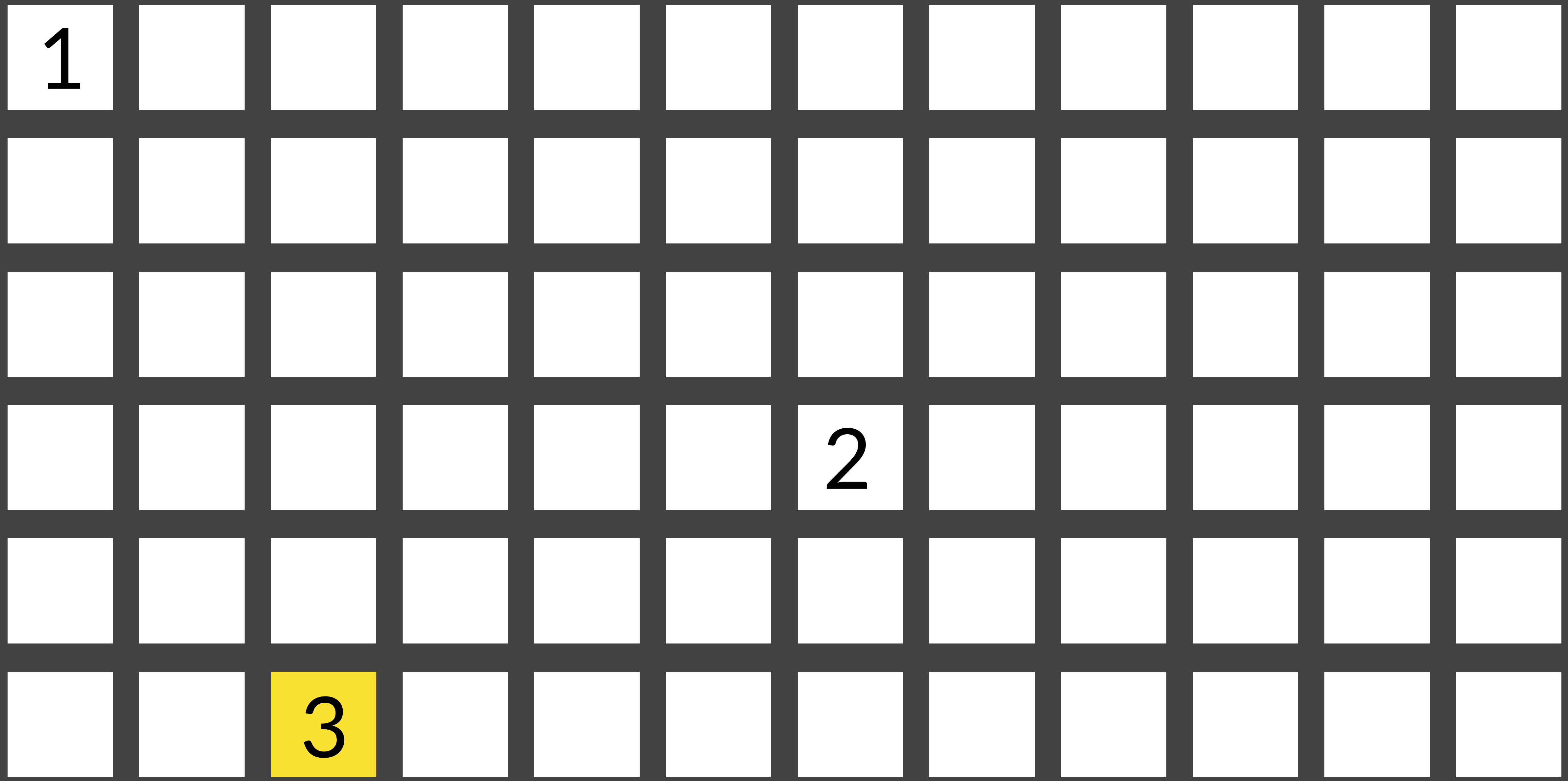


Classic Caching

Cache of size k

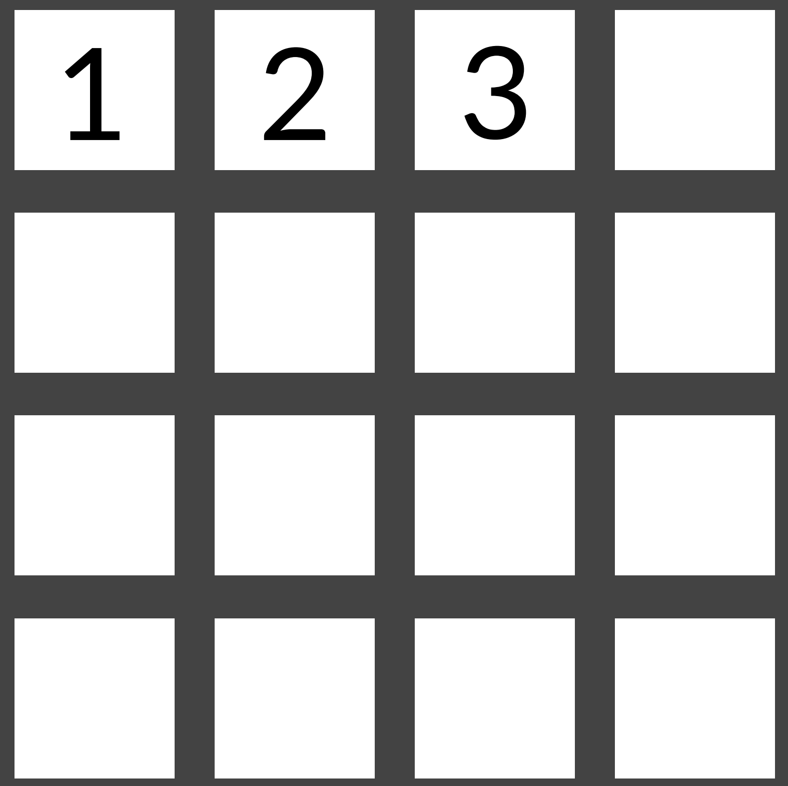


n total pages

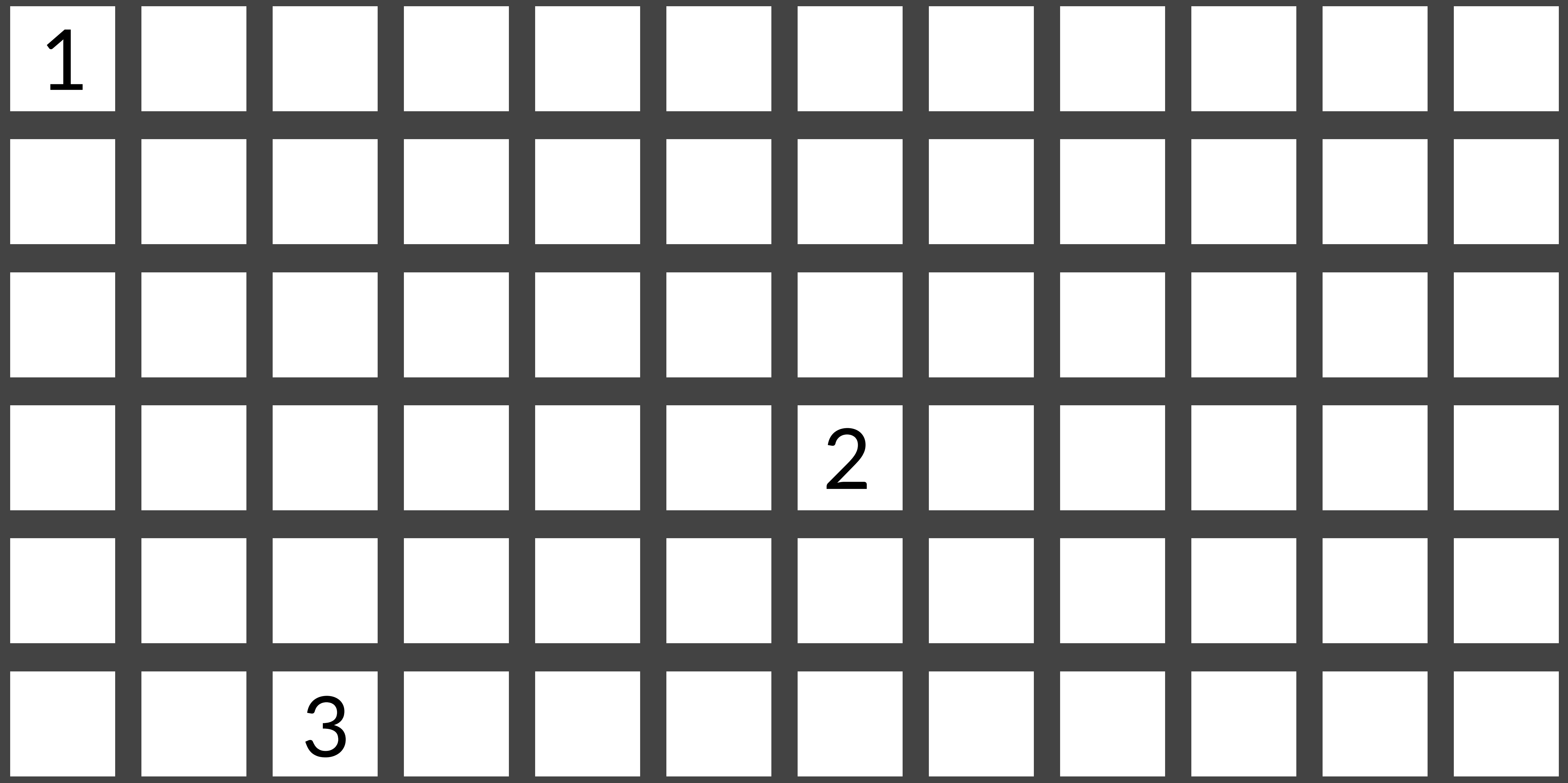


Classic Caching

Cache of size k

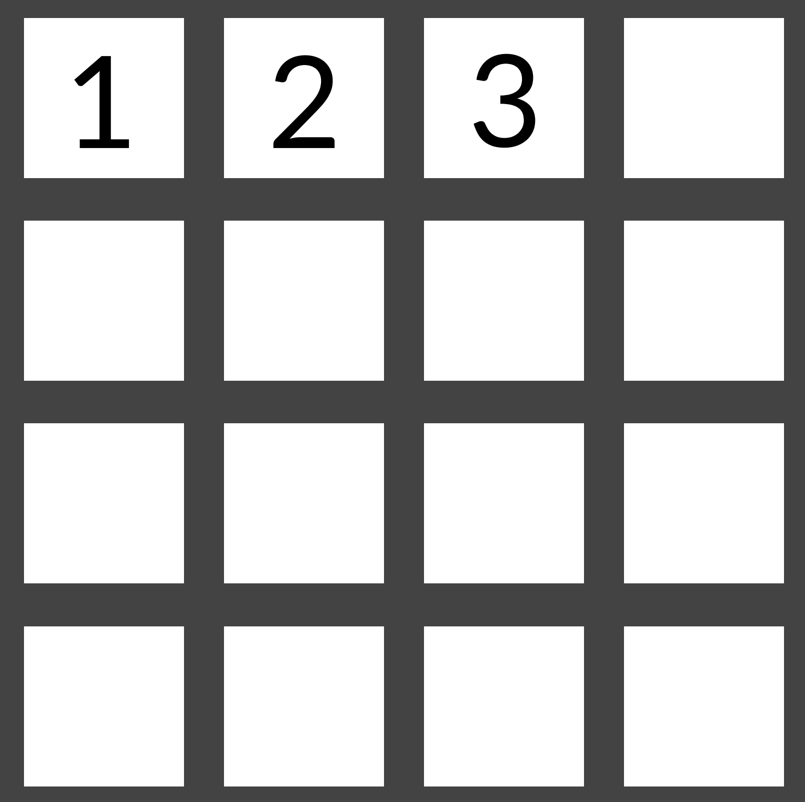


n total pages

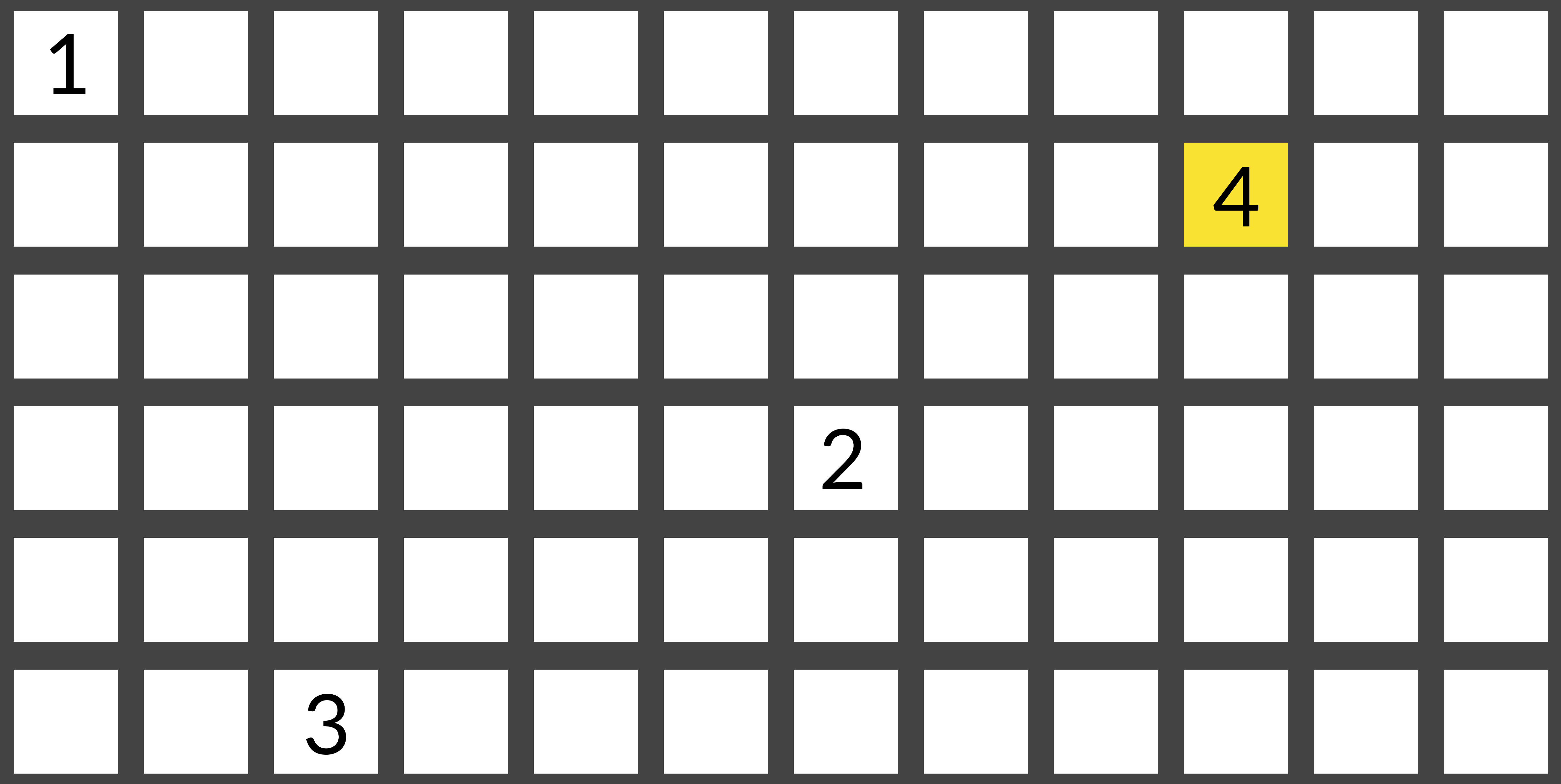


Classic Caching

Cache of size k

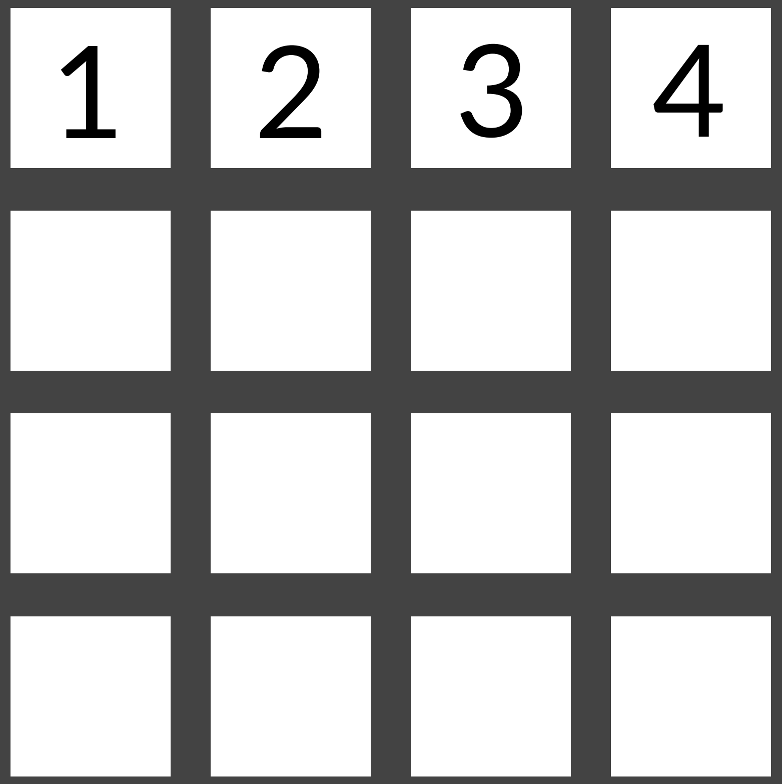


n total pages

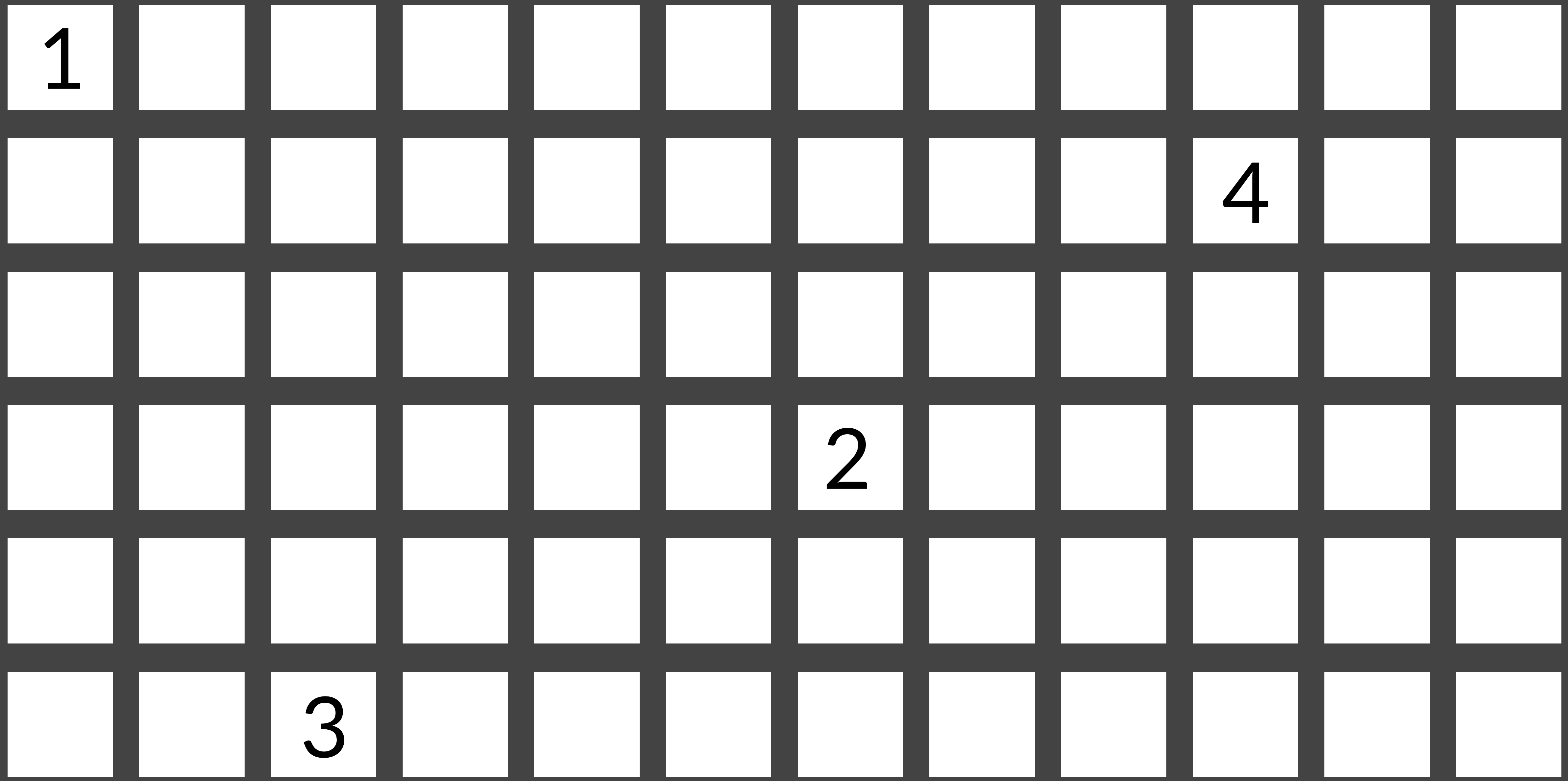


Classic Caching

Cache of size k

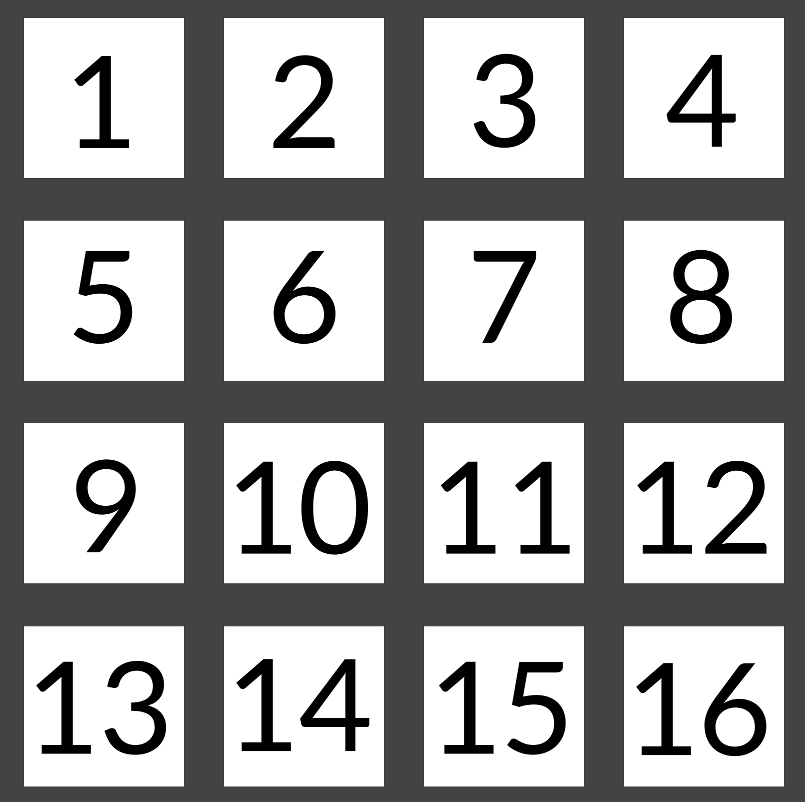


n total pages

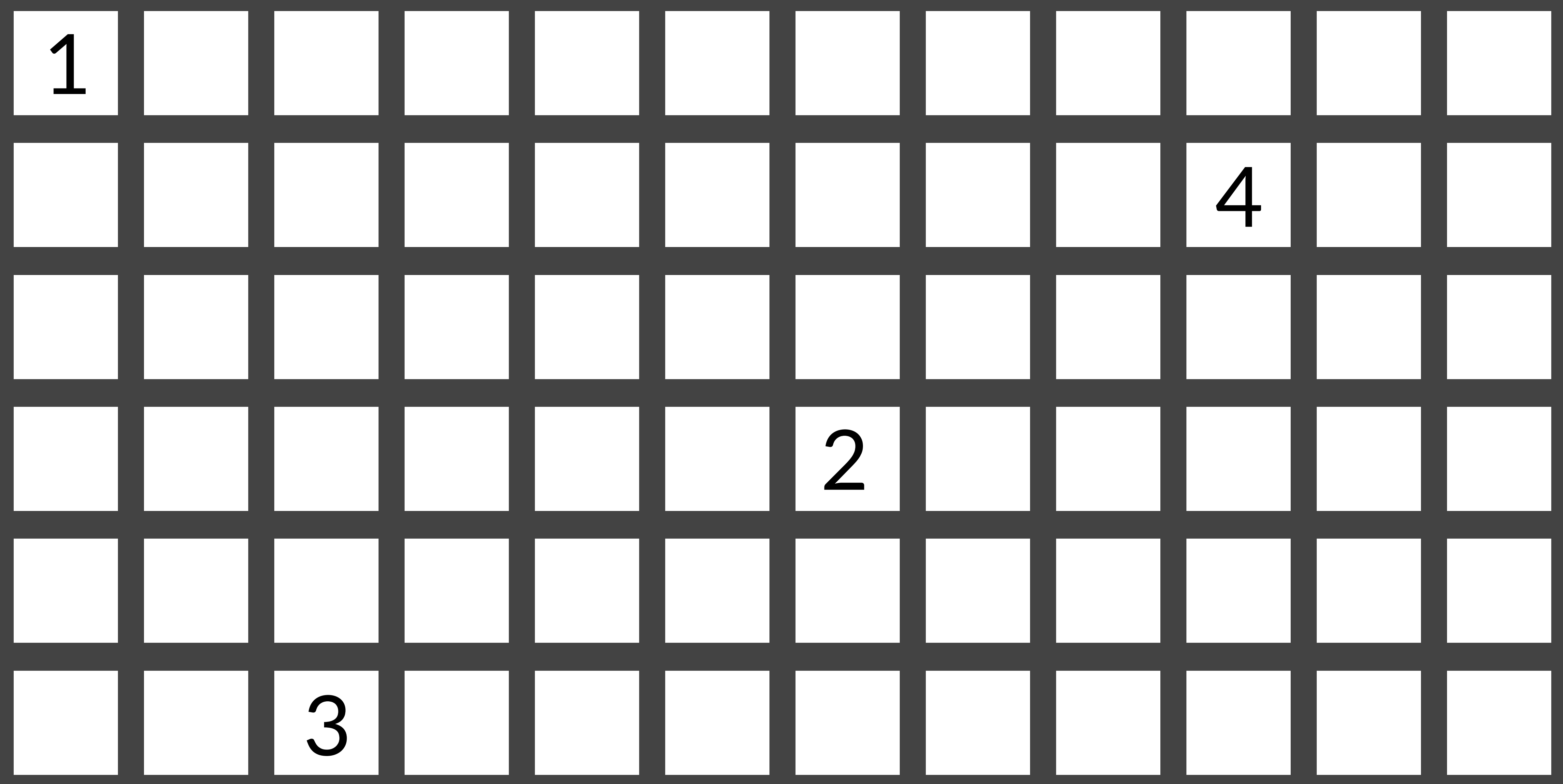


Classic Caching

Cache of size k

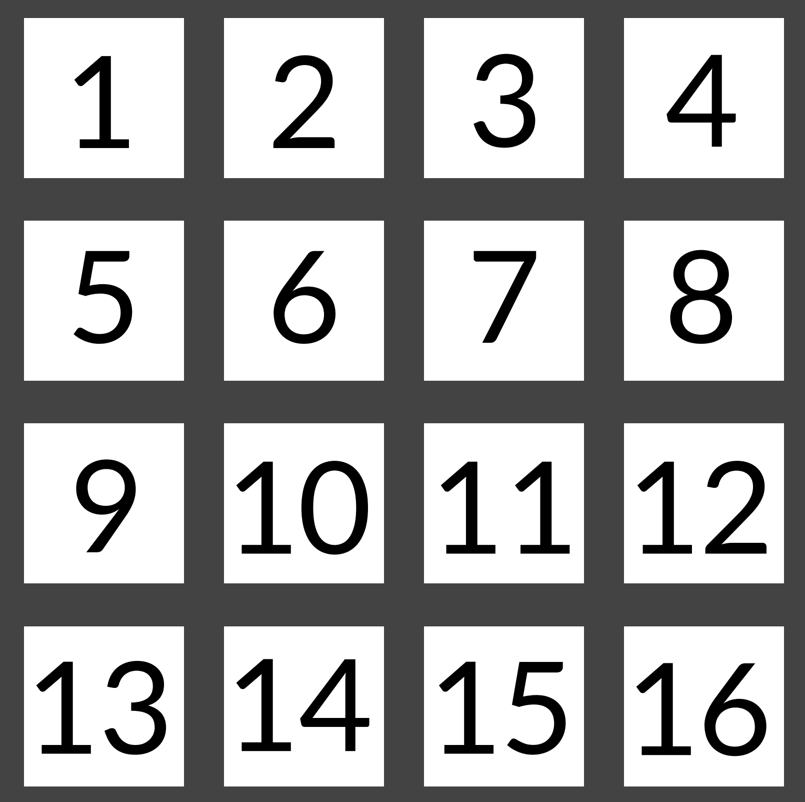


n total pages

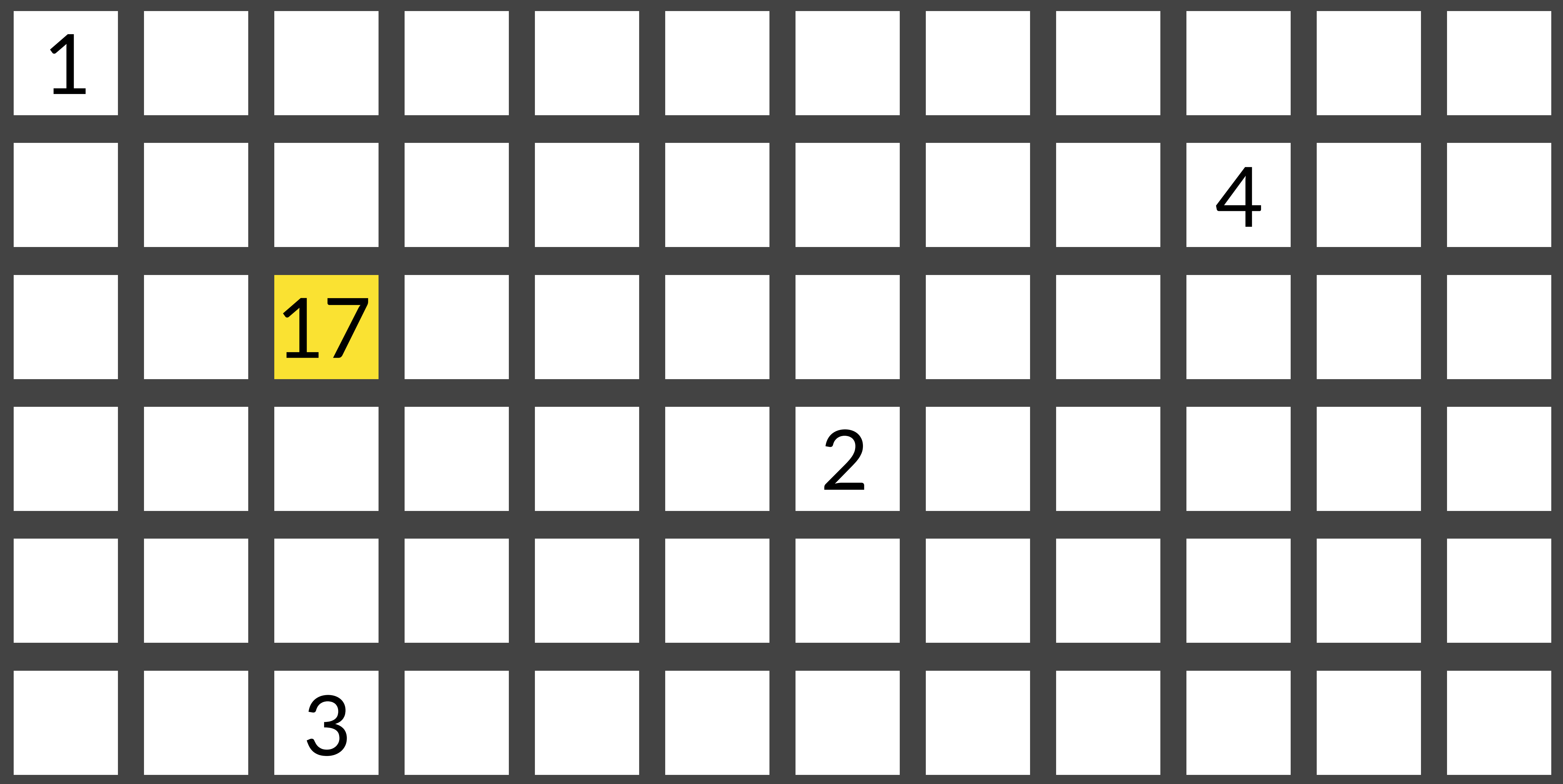


Classic Caching

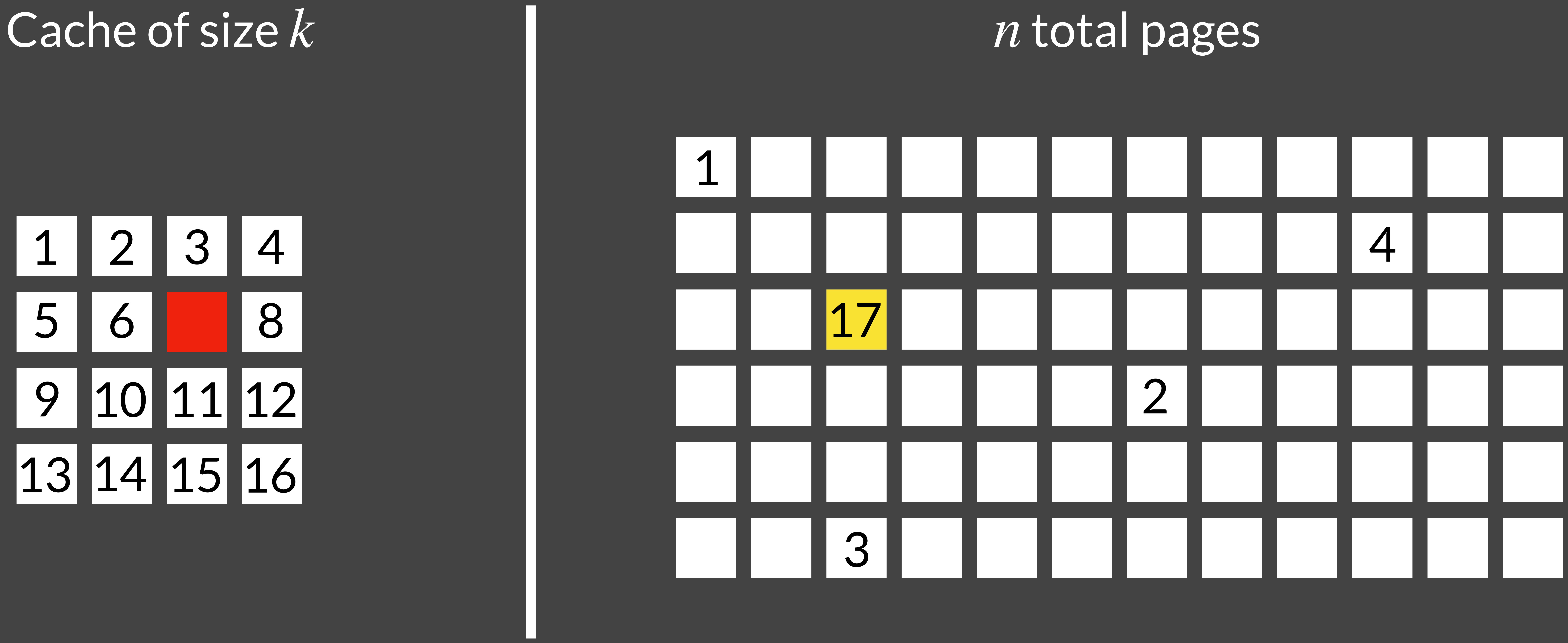
Cache of size k



n total pages

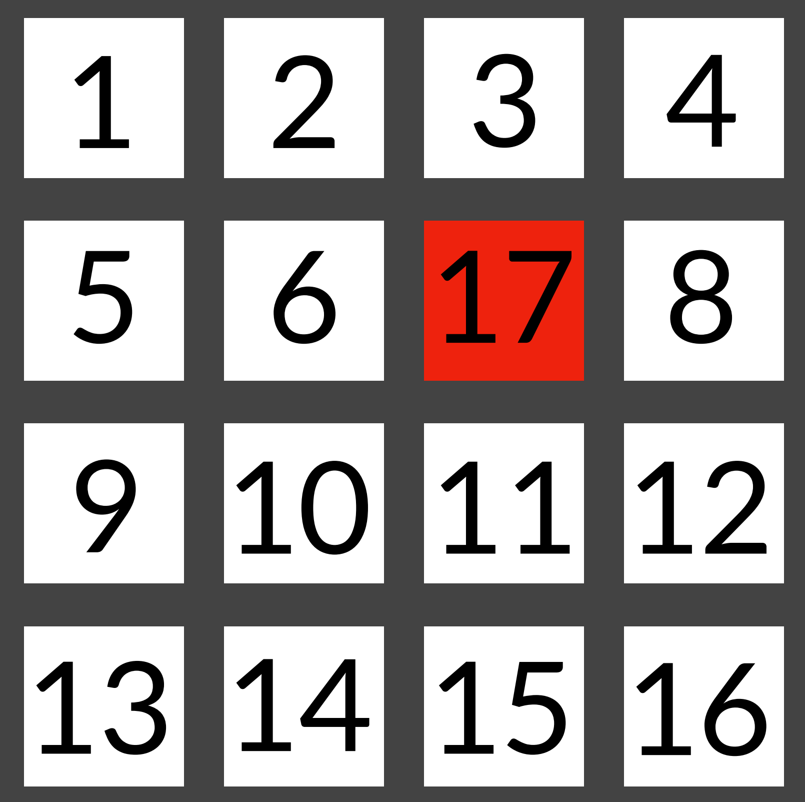


Classic Caching

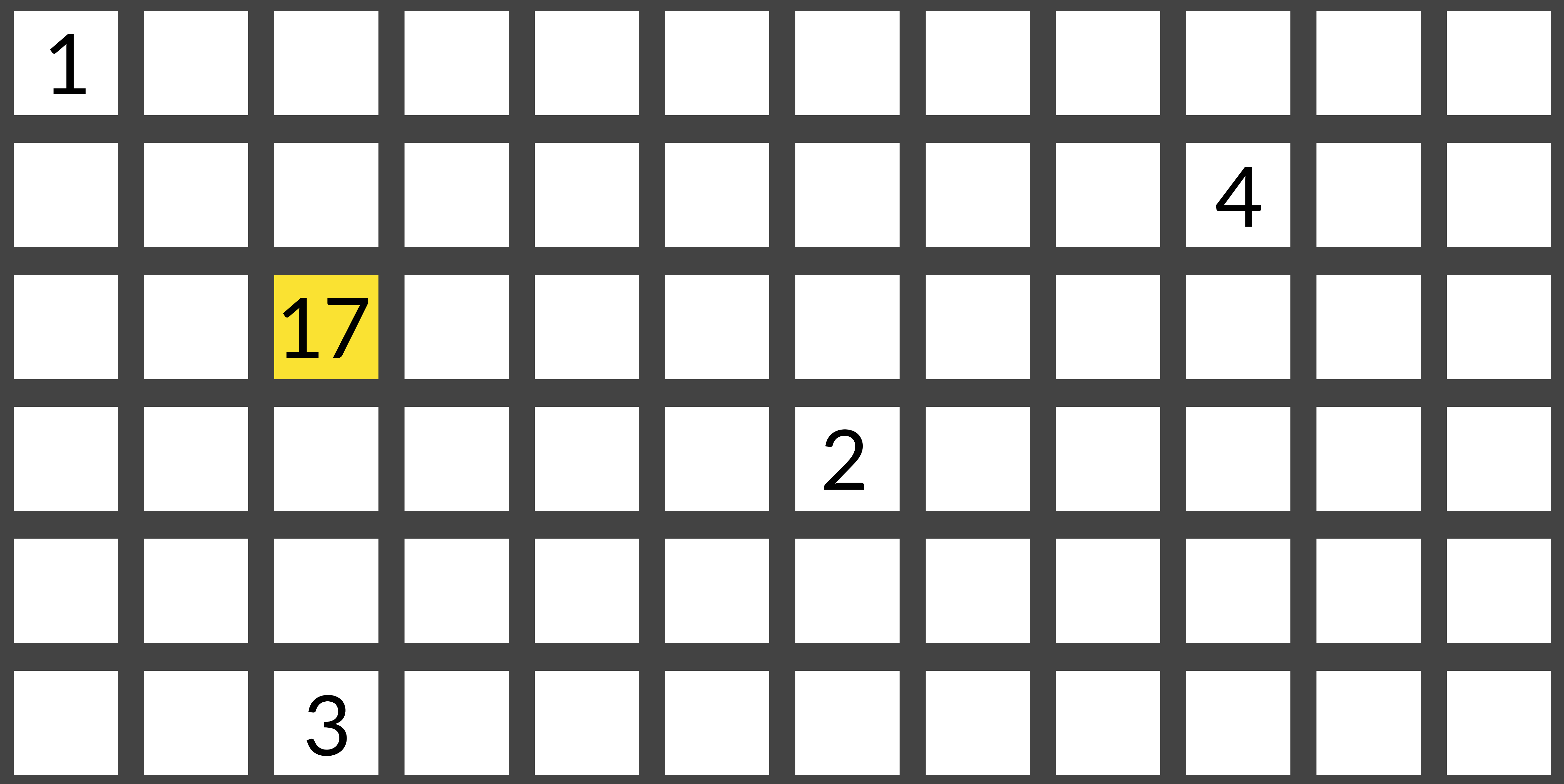


Classic Caching

Cache of size k

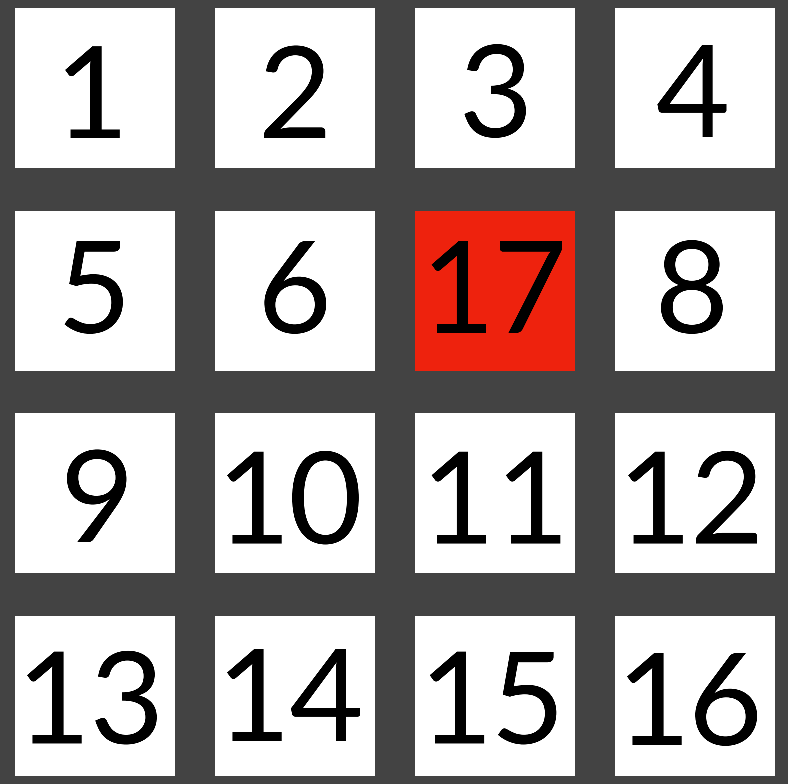


n total pages

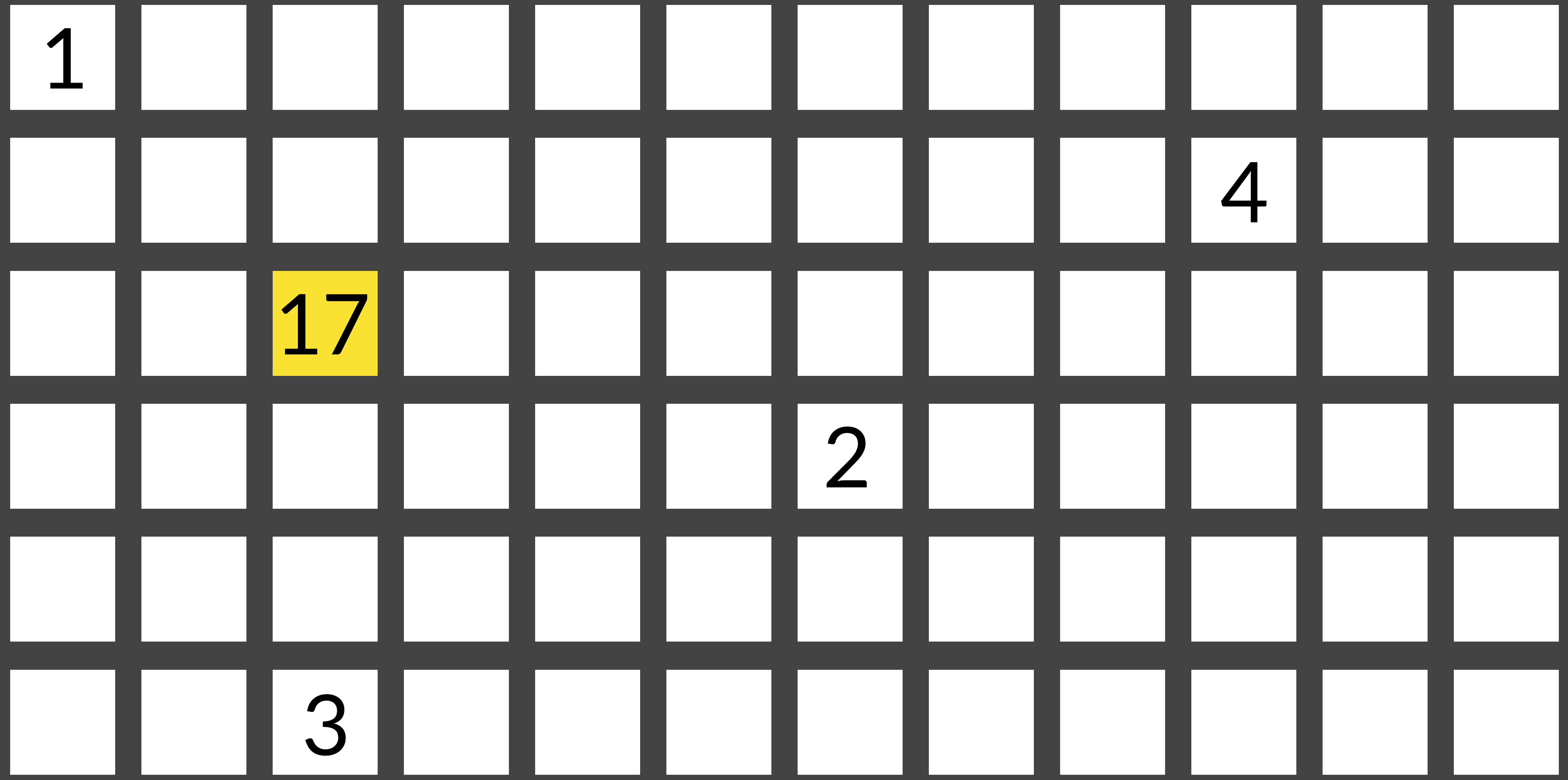


Classic Caching

Cache of size k



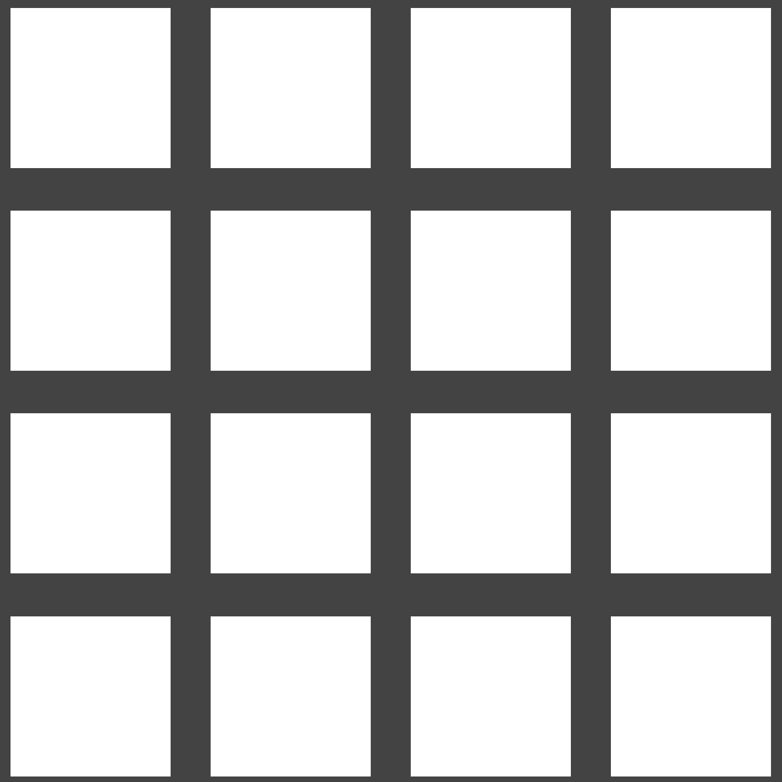
n total pages



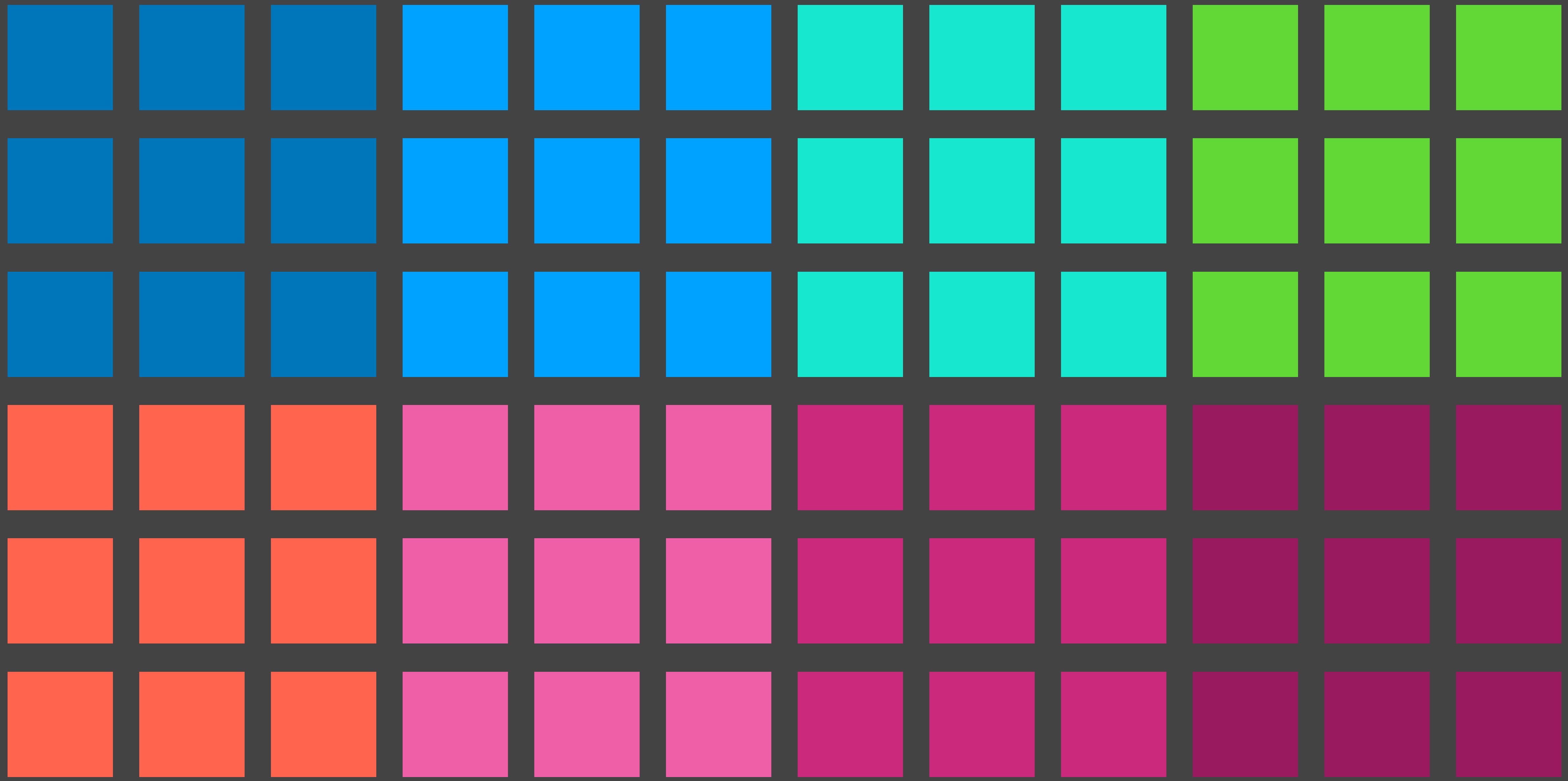
Goal is to minimize number of evictions!

Block-Aware Caching [Beckmann+ 2021]

Cache of size k

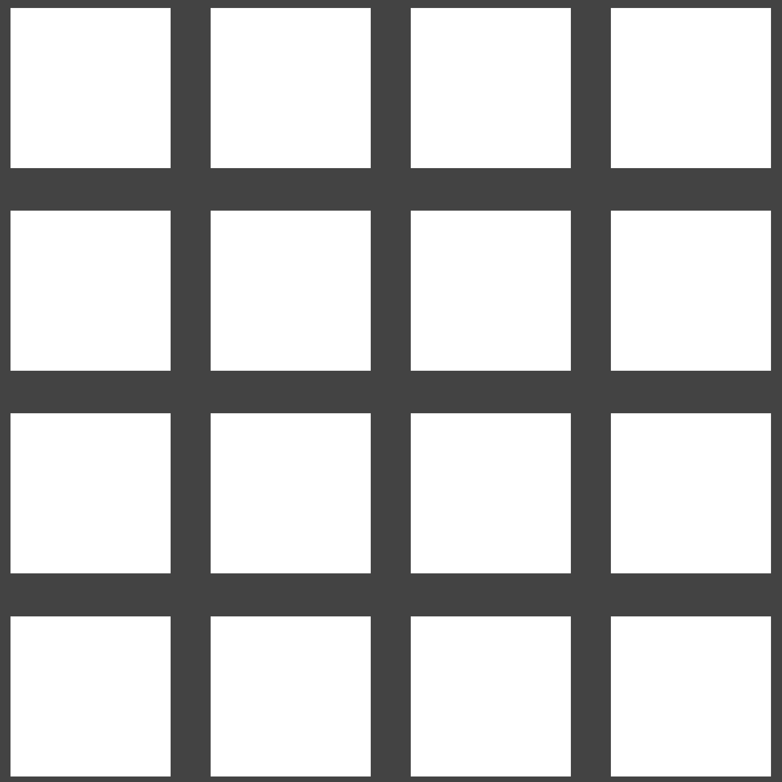


n total pages, **in blocks of size β**

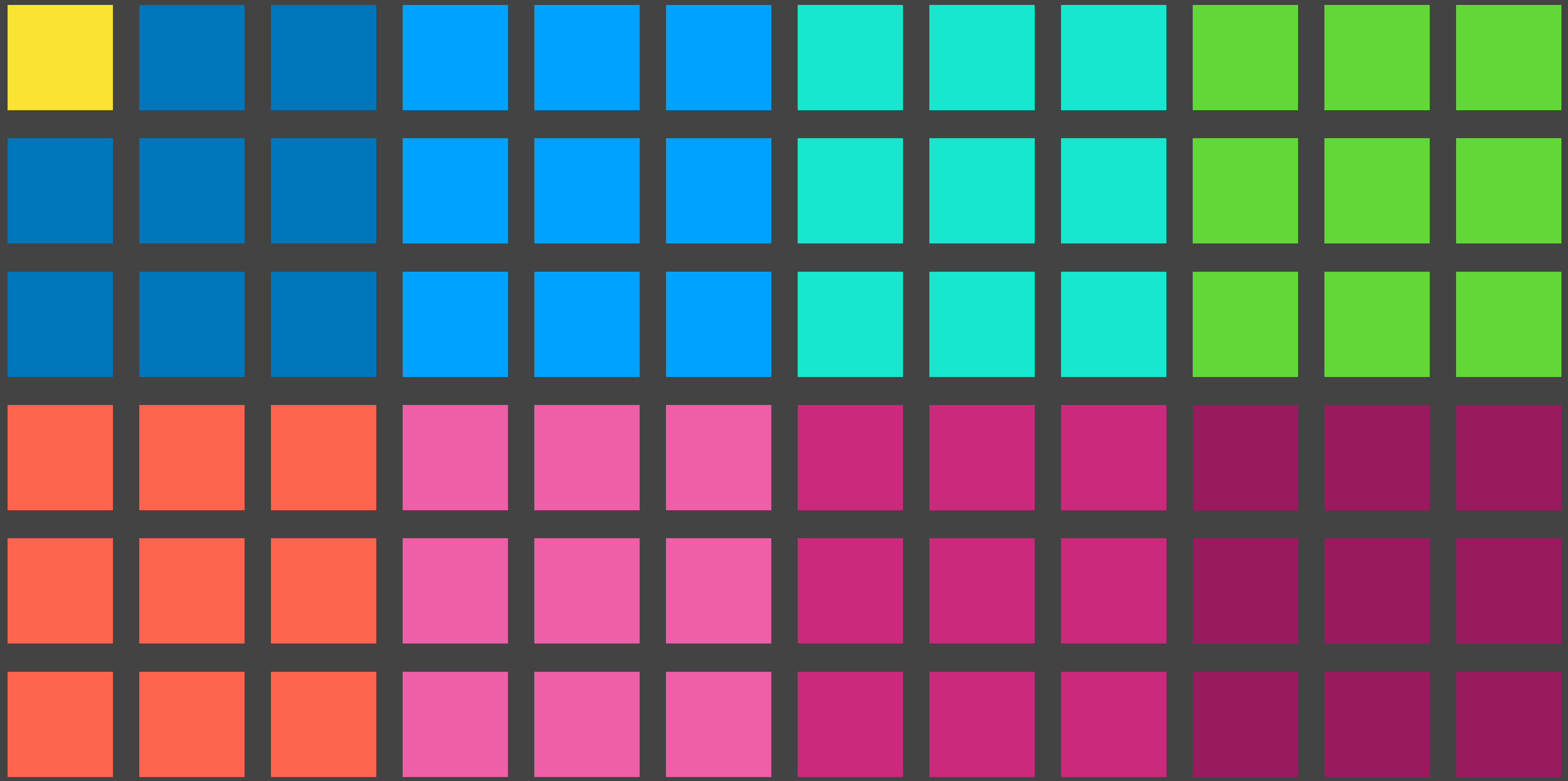


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

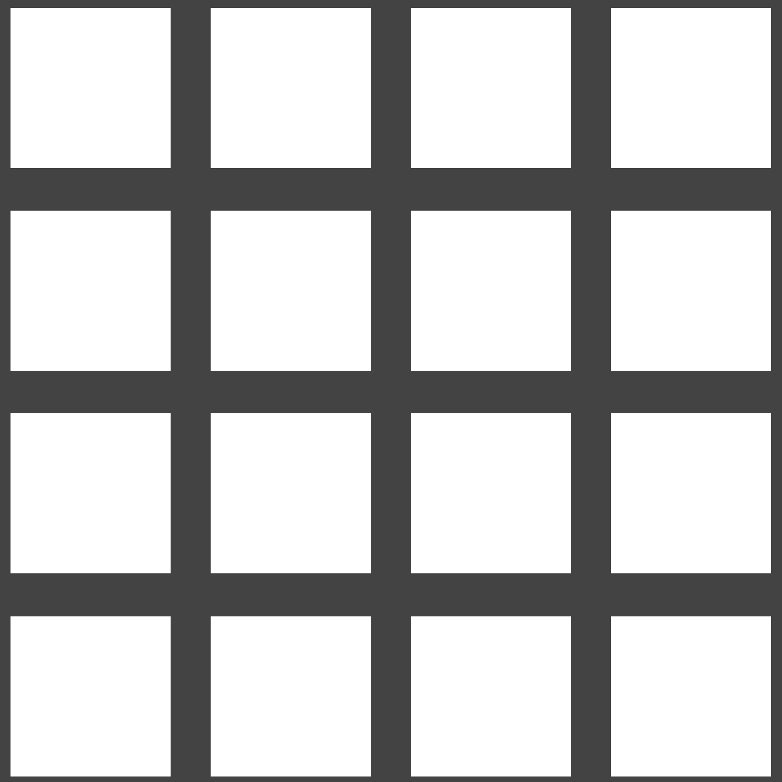


n total pages, **in blocks of size β**

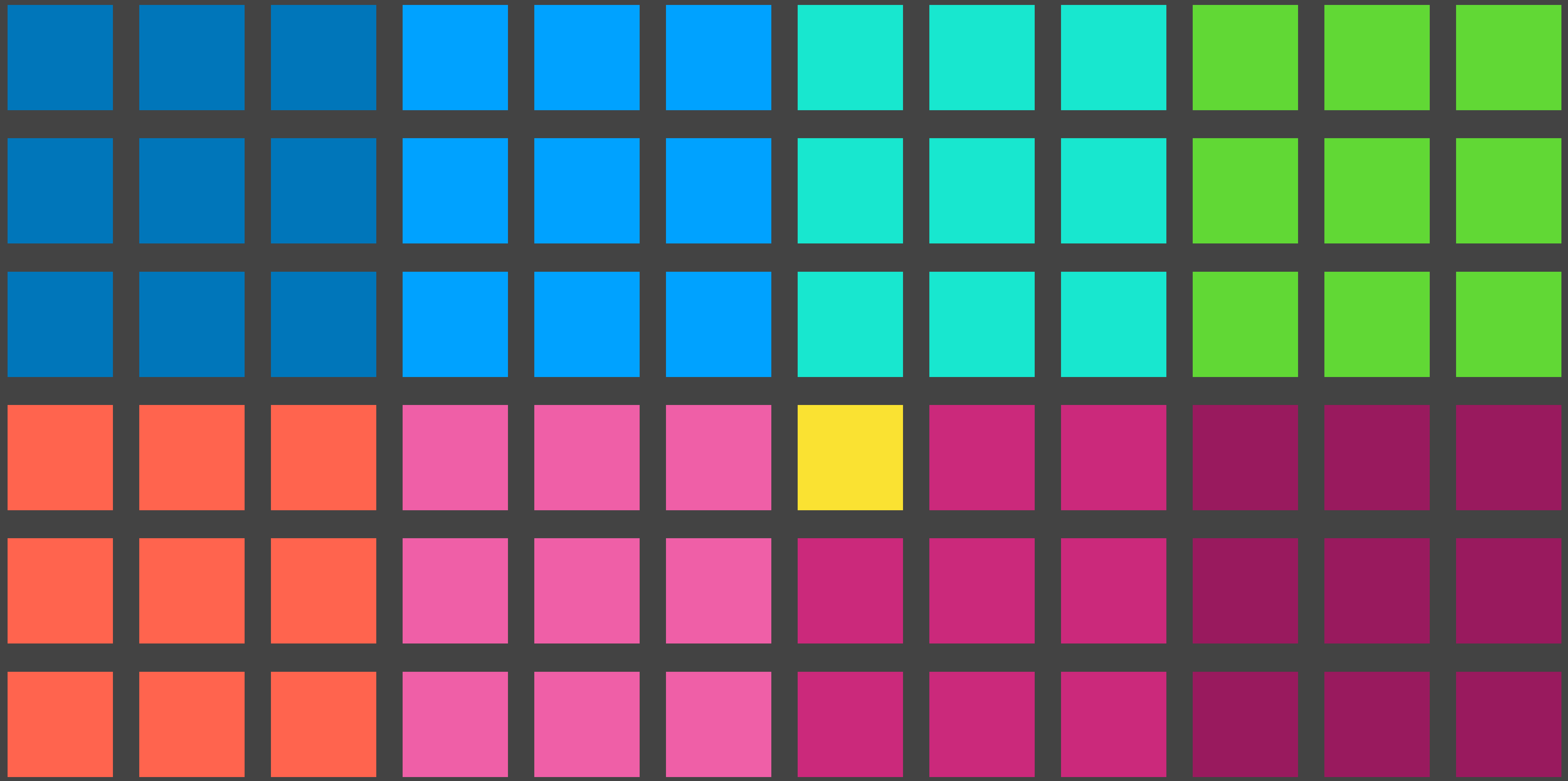


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

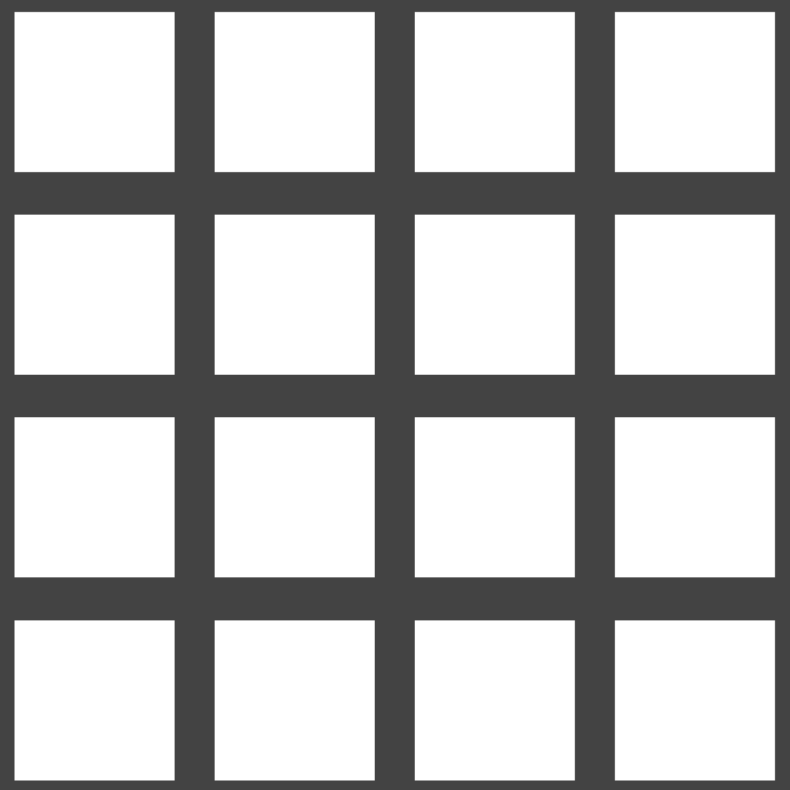


n total pages, **in blocks of size β**

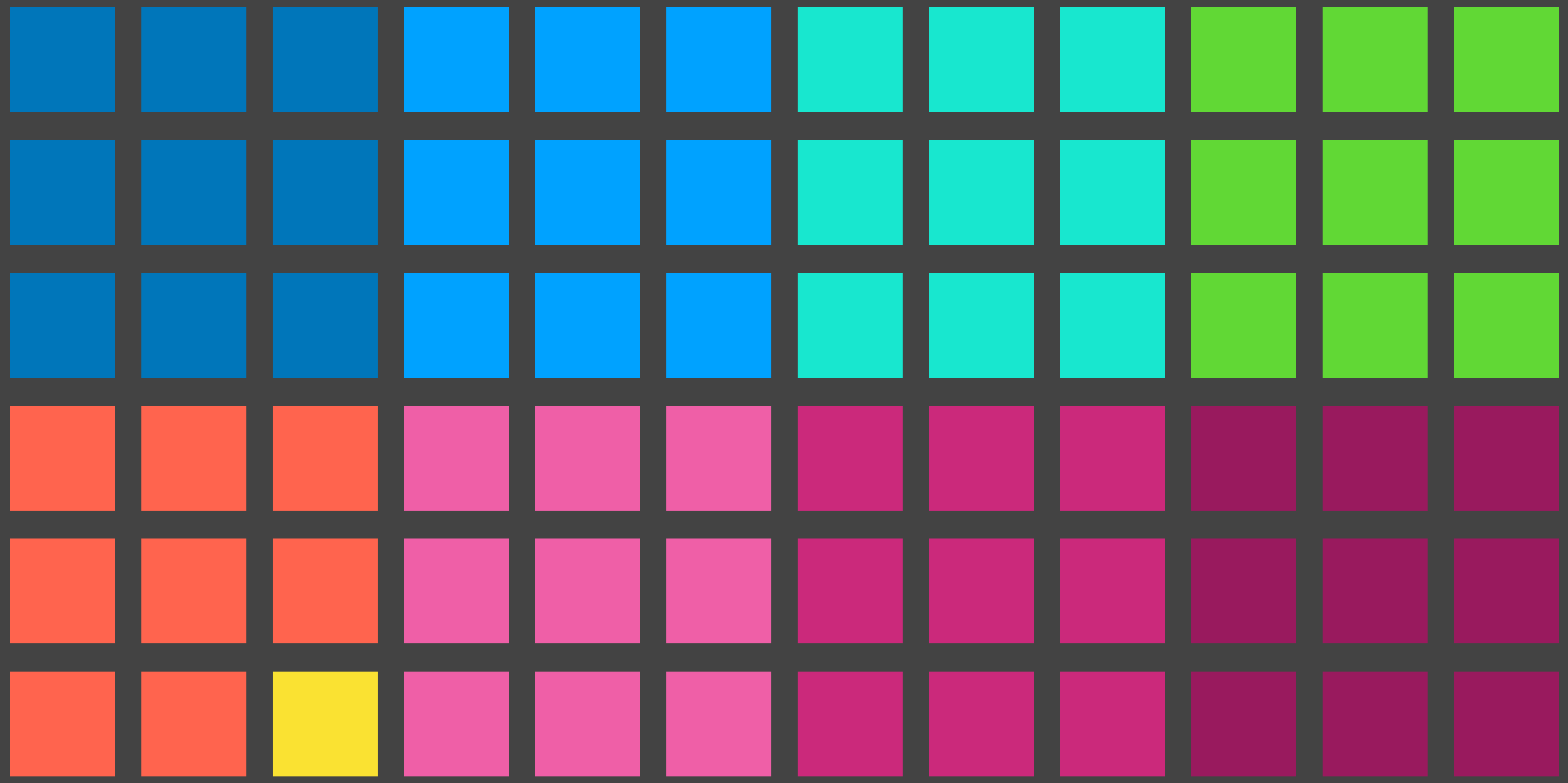


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

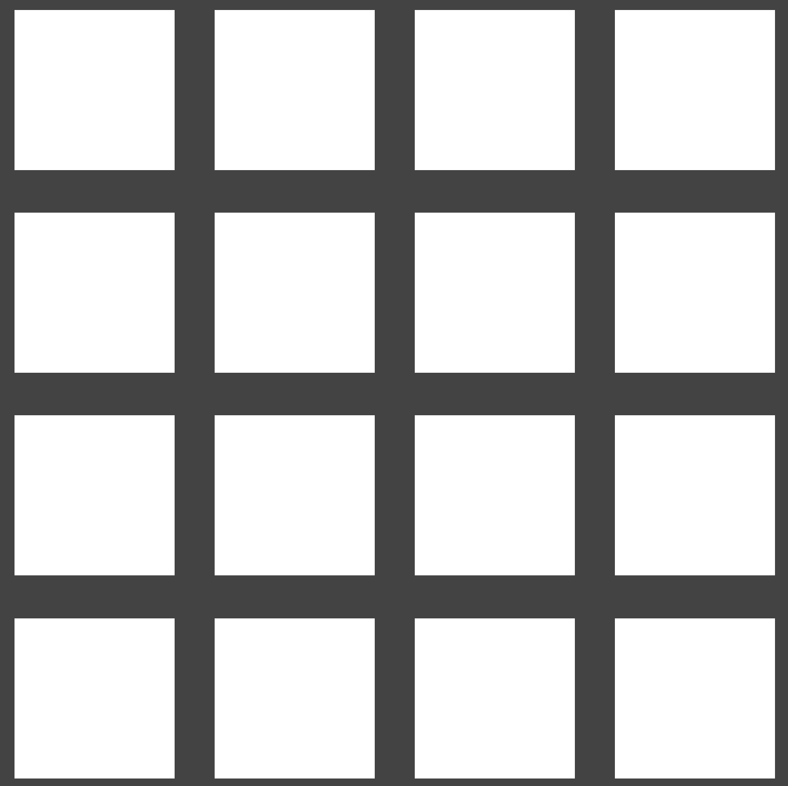


n total pages, **in blocks of size β**

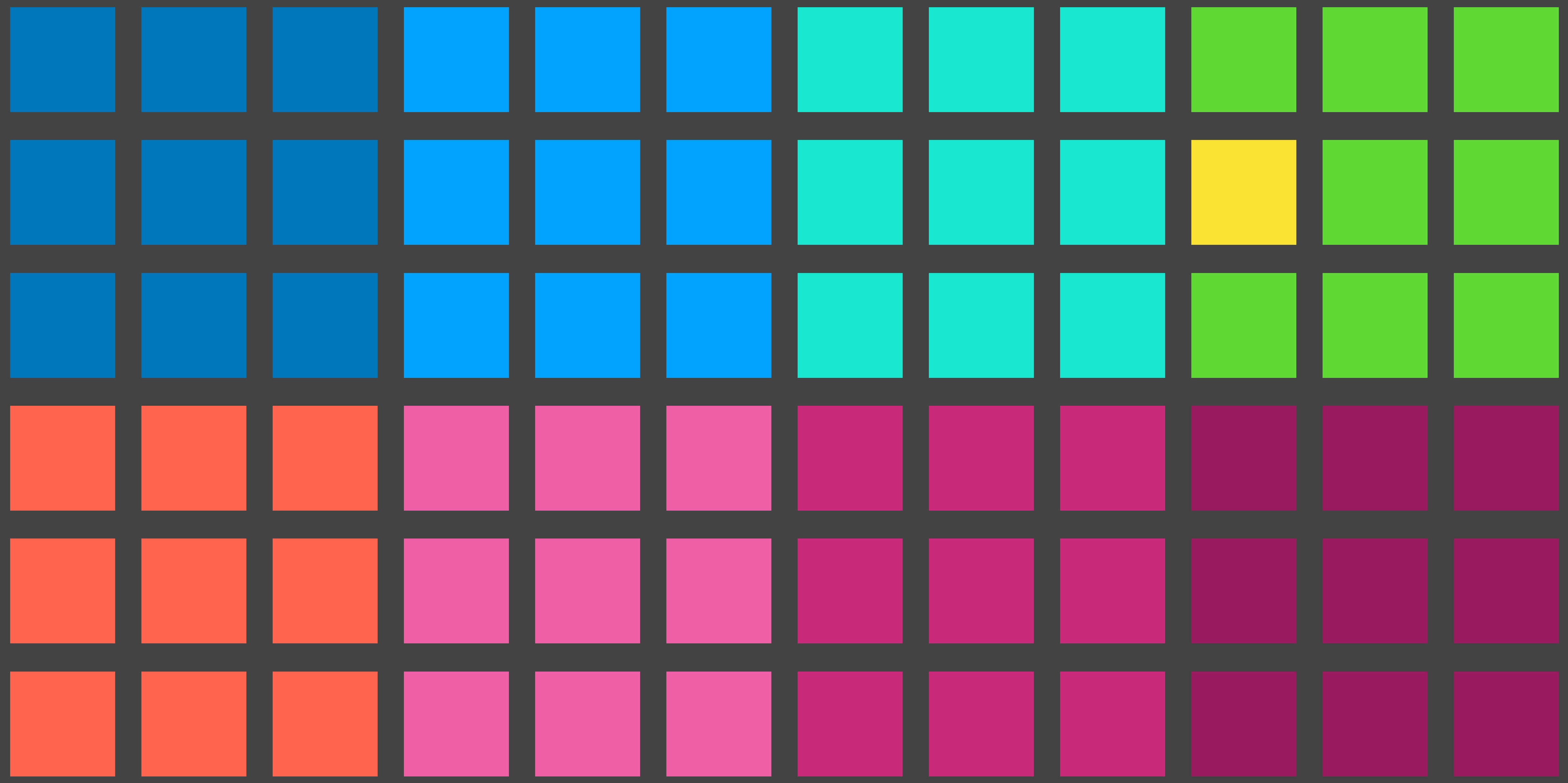


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

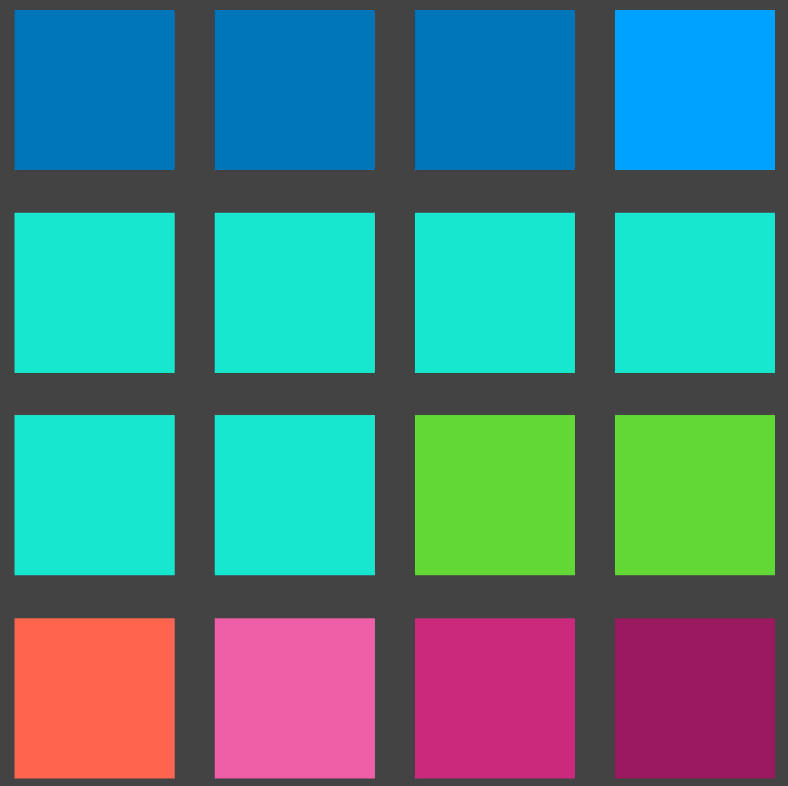


n total pages, **in blocks of size β**

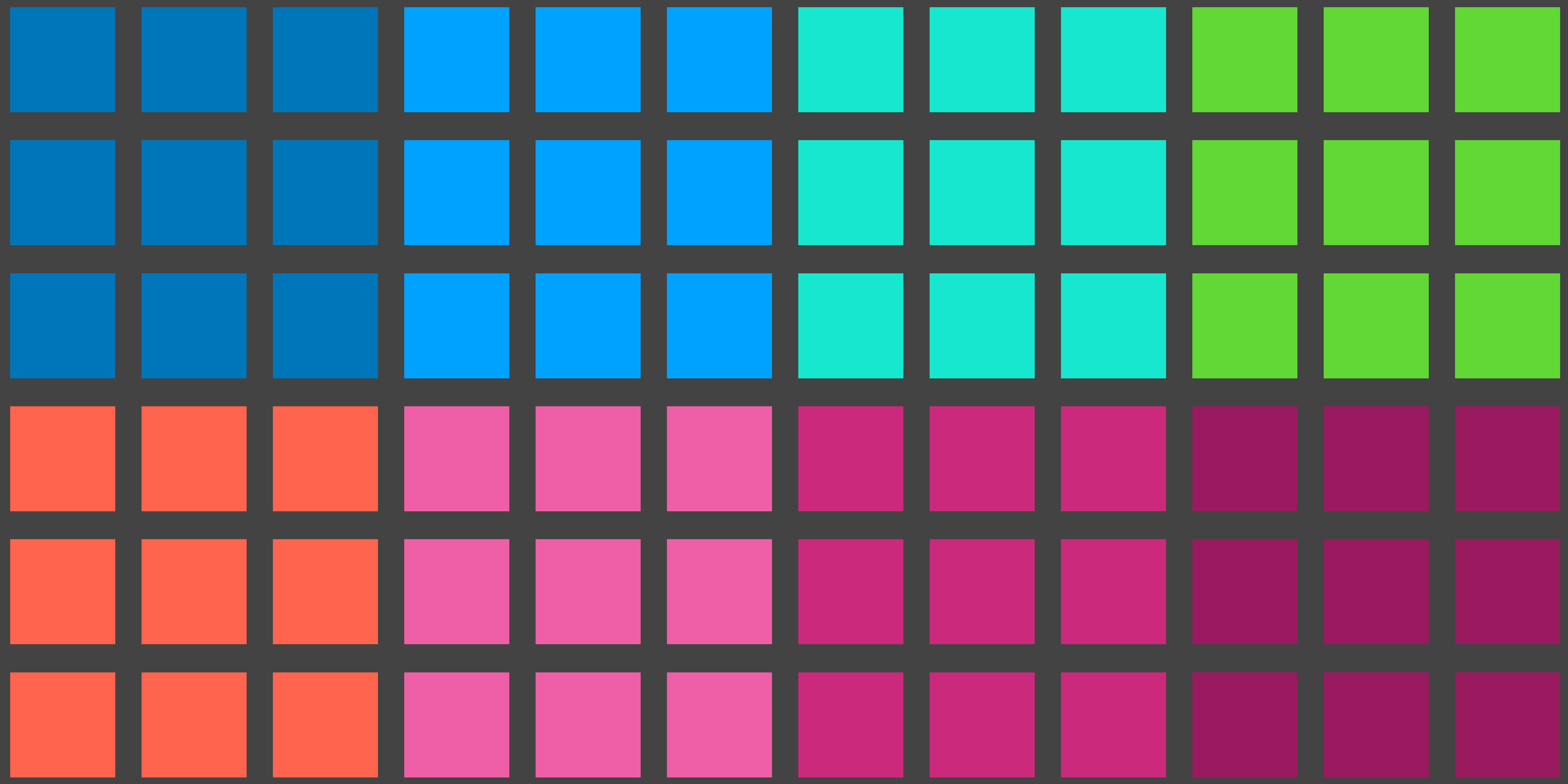


Block-Aware Caching [Beckmann+ 2021]

Cache of size k



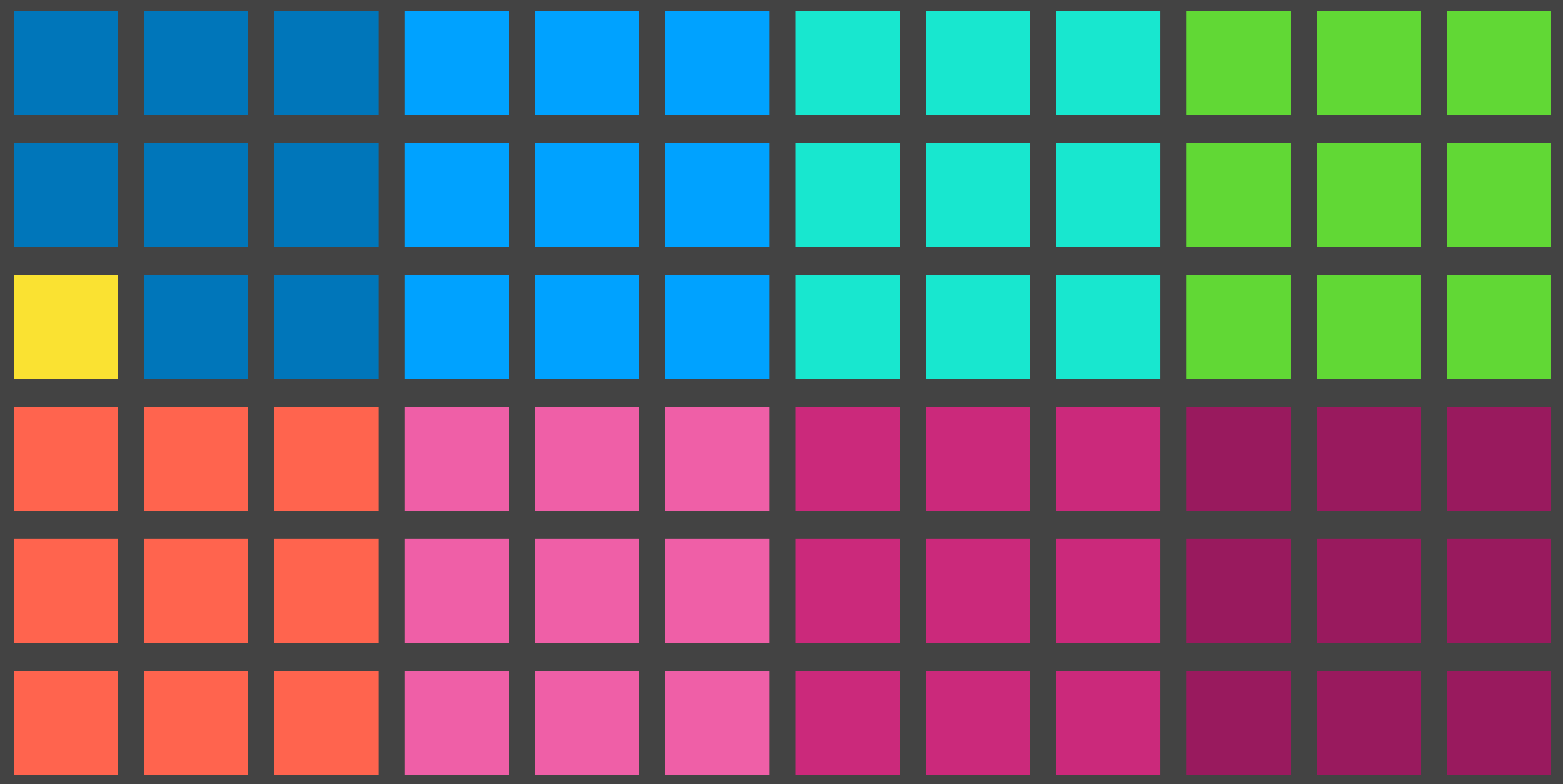
n total pages, **in blocks of size β**



Block-Aware Caching [Beckmann+ 2021]

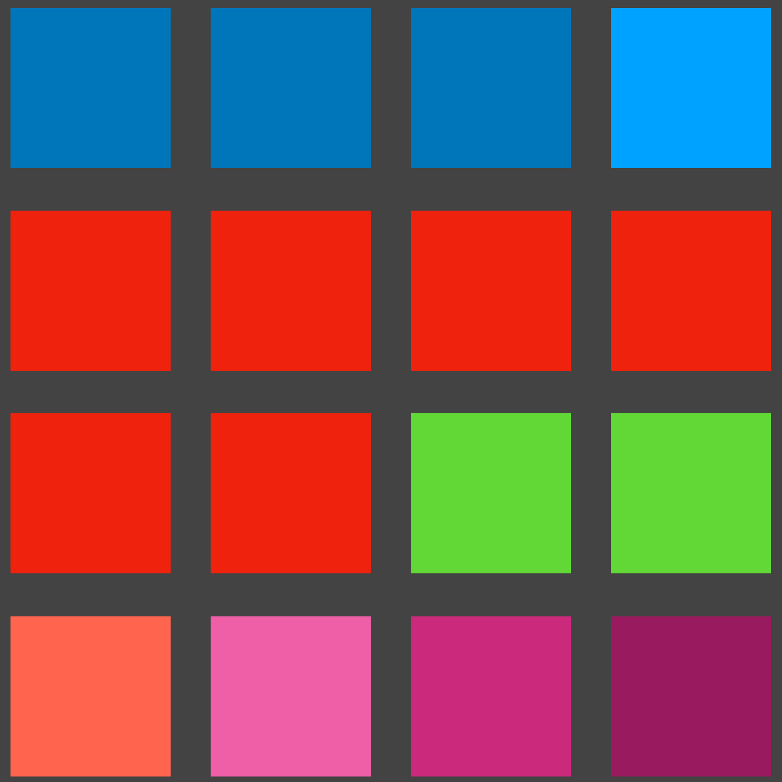
Cache of size k

n total pages, **in blocks of size β**

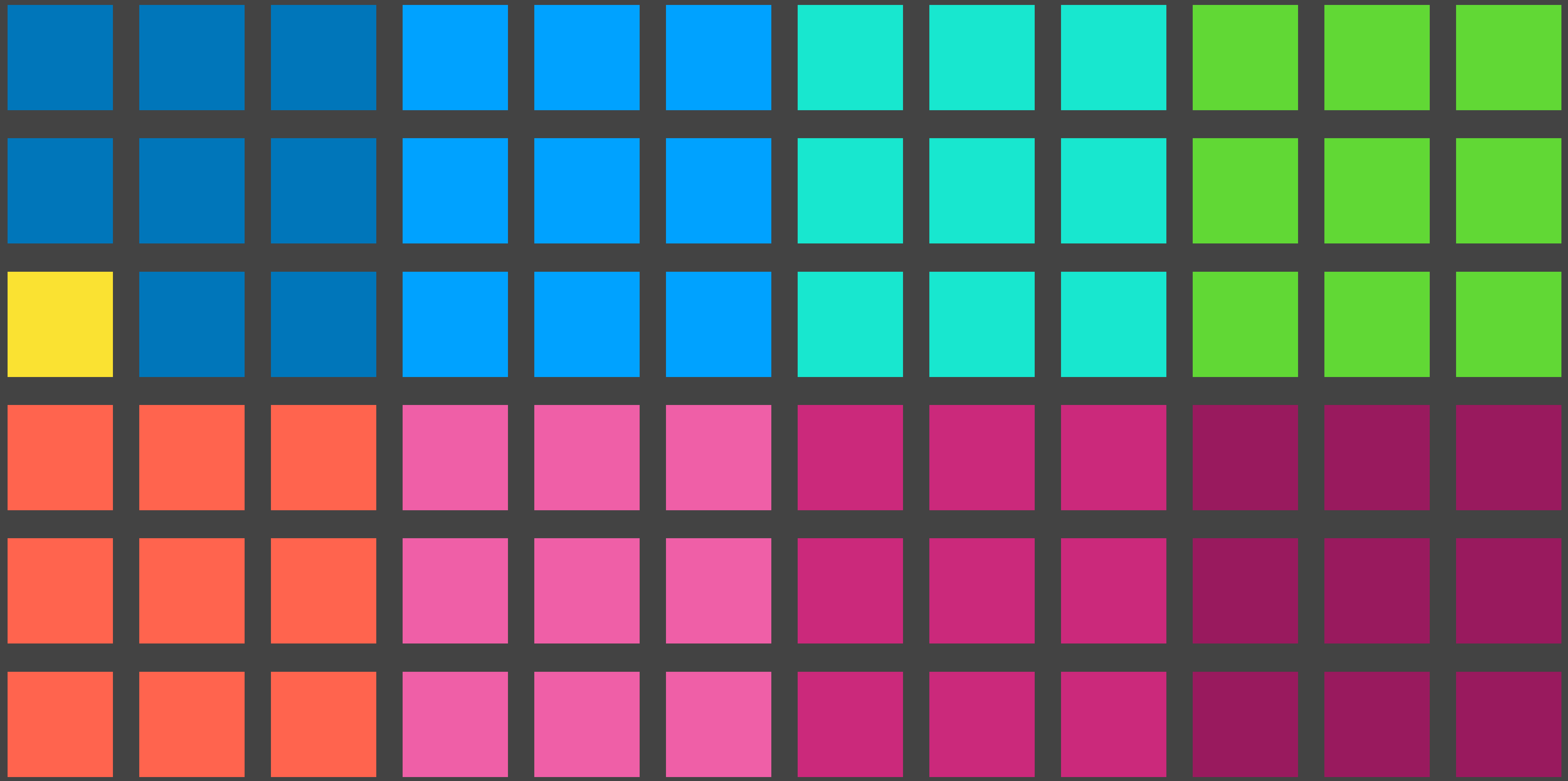


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

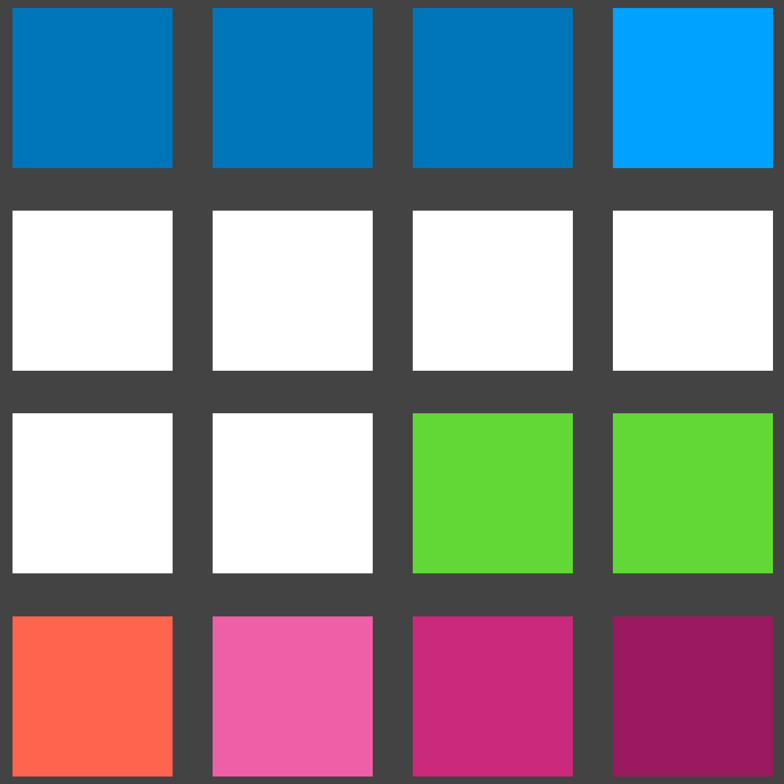


n total pages, **in blocks of size β**

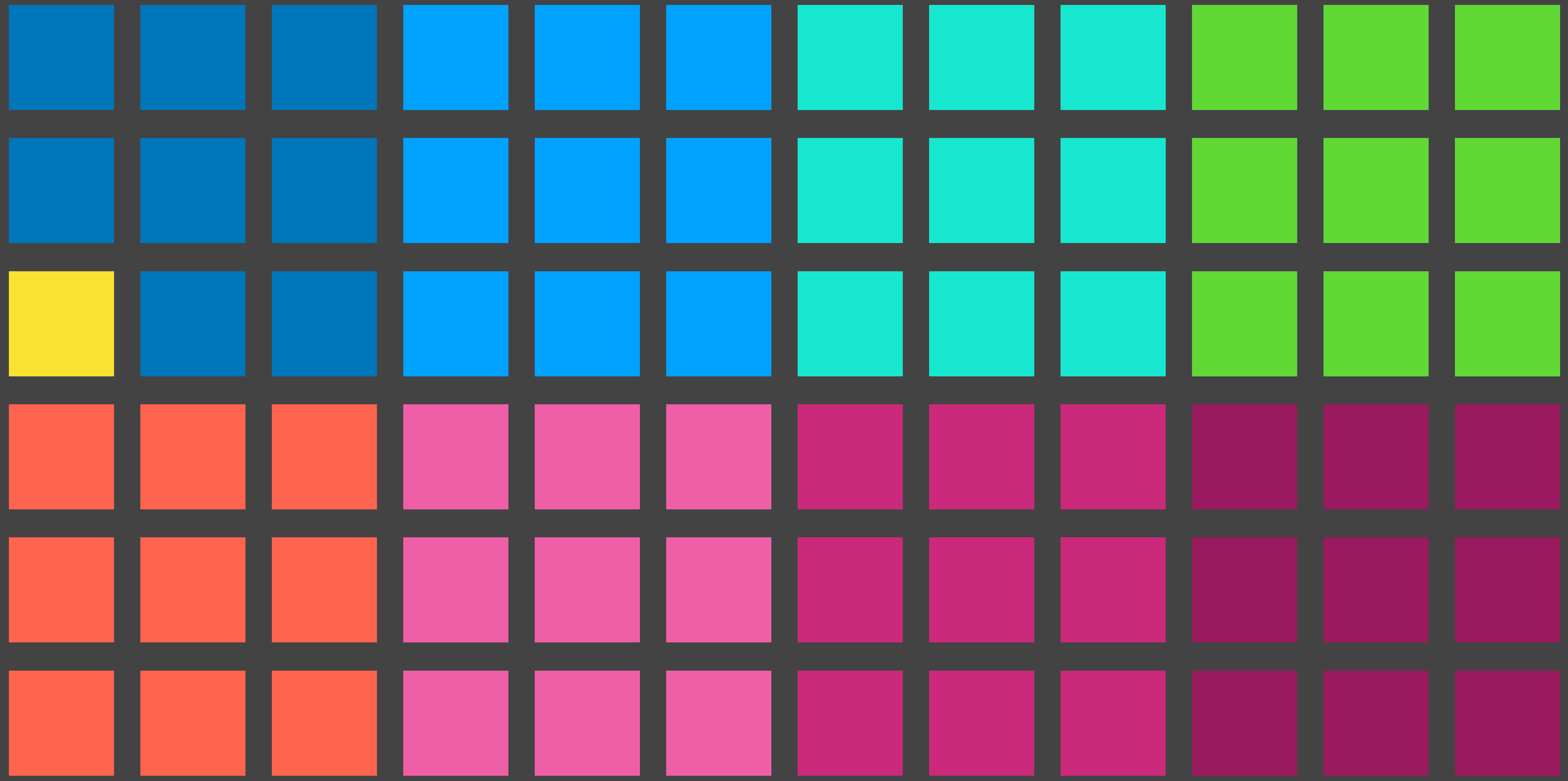


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

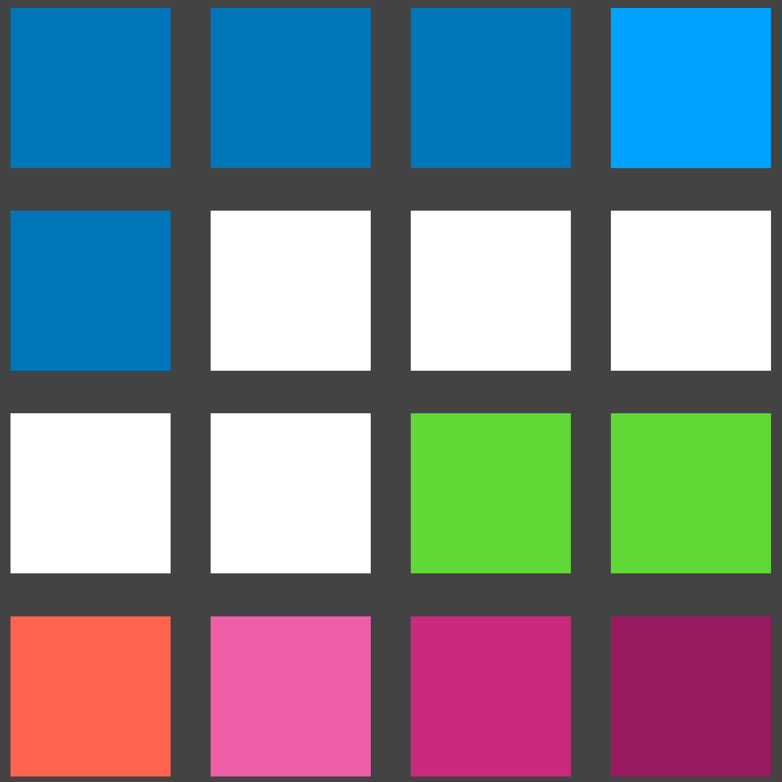


n total pages, **in blocks of size β**

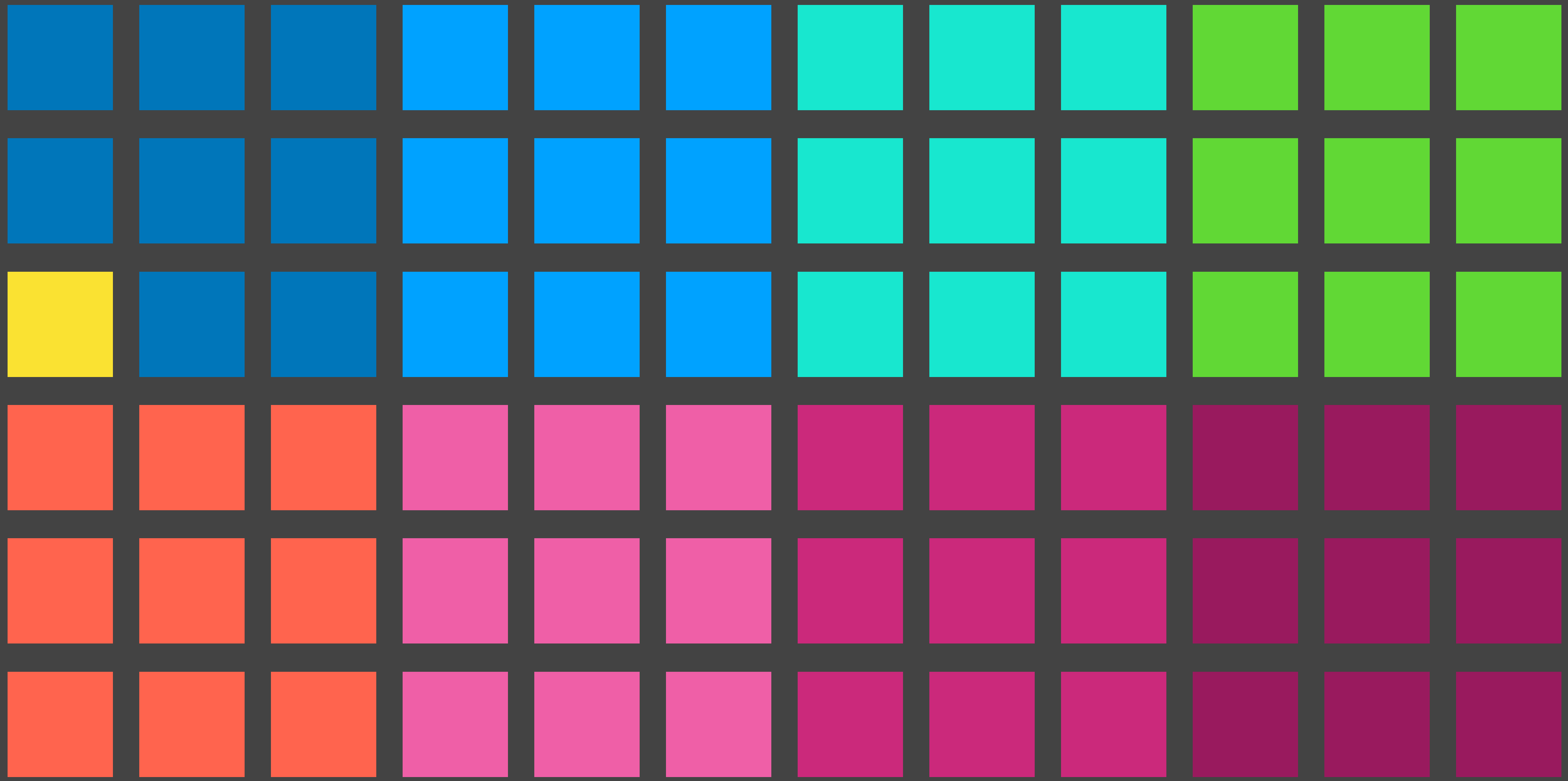


Block-Aware Caching [Beckmann+ 2021]

Cache of size k

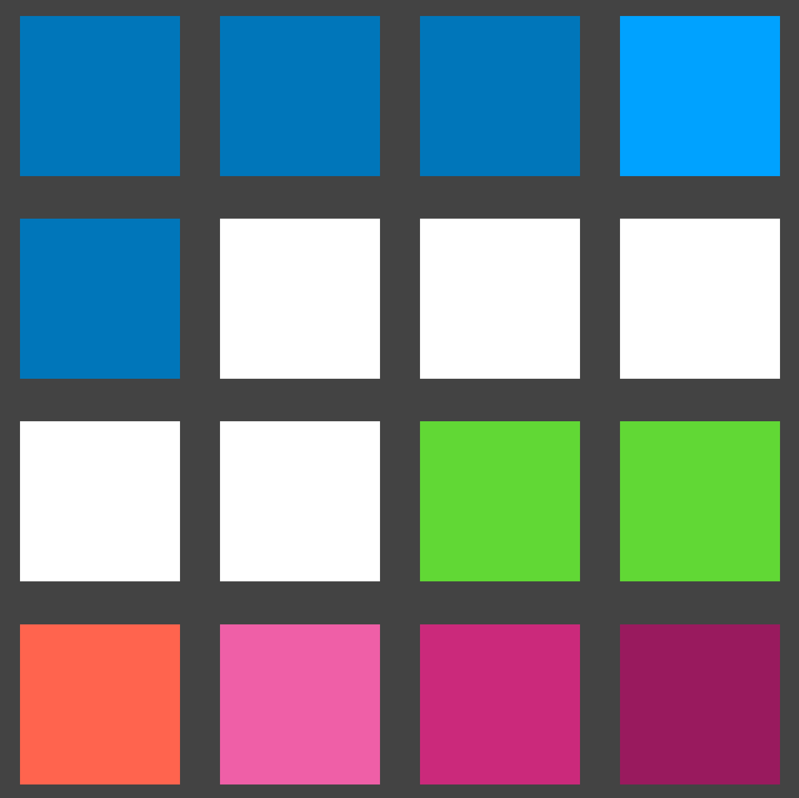


n total pages, **in blocks of size β**

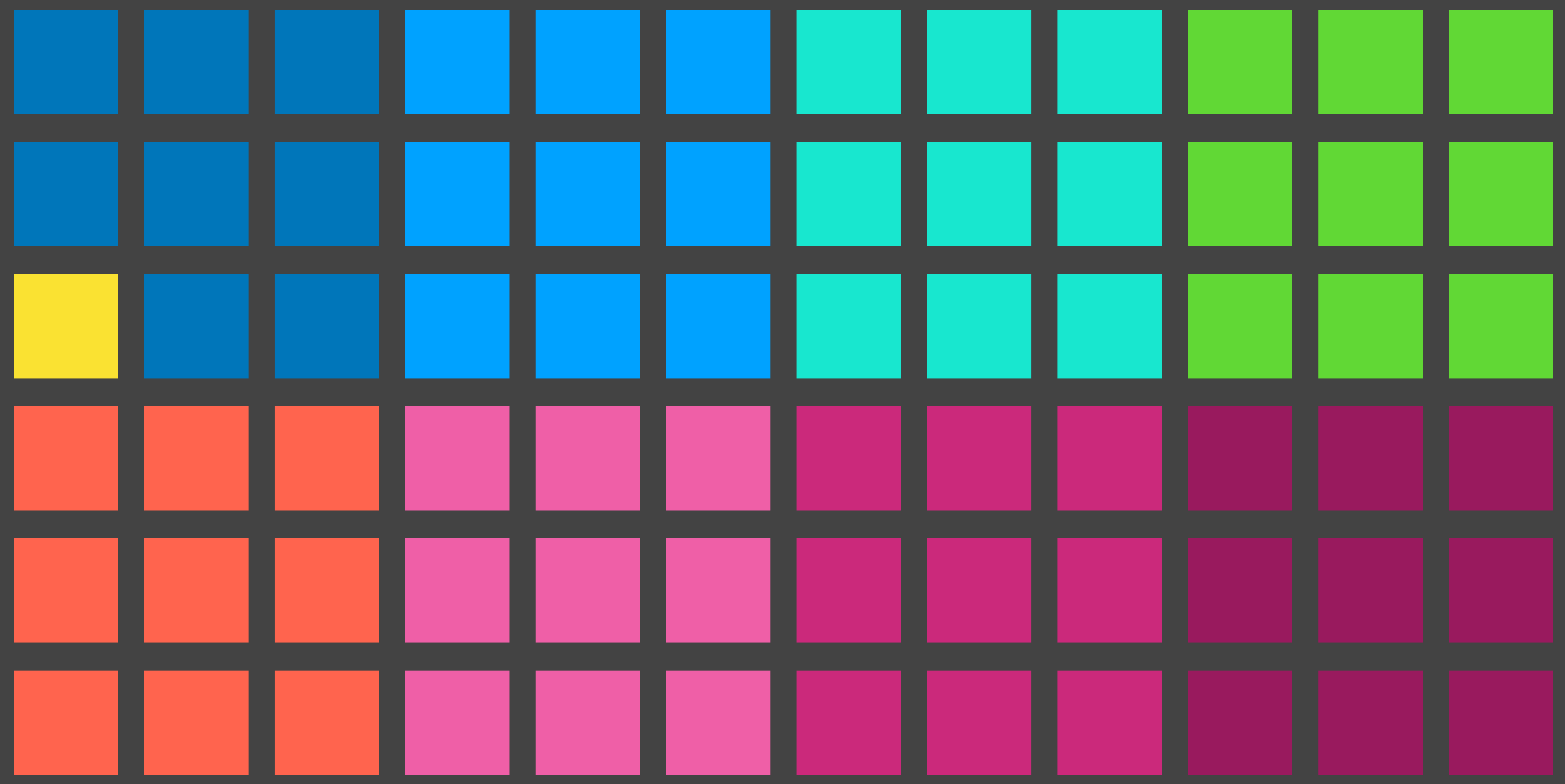


Block-Aware Caching [Beckmann+ 2021]

Cache of size k



n total pages, **in blocks of size β**



Goal is to minimize number of **blocks** evicted!

Results [Coester, Naor L., Talmon, SPAA 22]

New!

	Classic	Block-Aware
Offline	1	
Deterministic Online	k	
Randomized Online	O(log k)	

Results [Coester, Naor L., Talmon, SPAA 22]

New!

	Classic	Block-Aware
Offline	1	β
Deterministic Online	k	β k
Randomized Online	O(log k)	O(β log k)

Trivial!

Results [Coester, Naor L., Talmon, SPAA 22]

New!

	Classic	Block-Aware
Offline	1	O(log k)
Deterministic Online	k	k
Randomized Online	O(log k)	O($\log^2$ k)

Our Result

Results [Coester, Naor L., Talmon, SPAA 22]

New!

	Classic	Block-Aware
Offline	1	O(log k)
Deterministic Online	k	k
Randomized Online	O(log k)	O($\log^2$ k)

Our Result

Also show $\Omega(\beta)$ lower bound for randomized algorithms in **fetching cost model…**

Results [Coester, Naor L., Talmon, SPAA 22]

New!

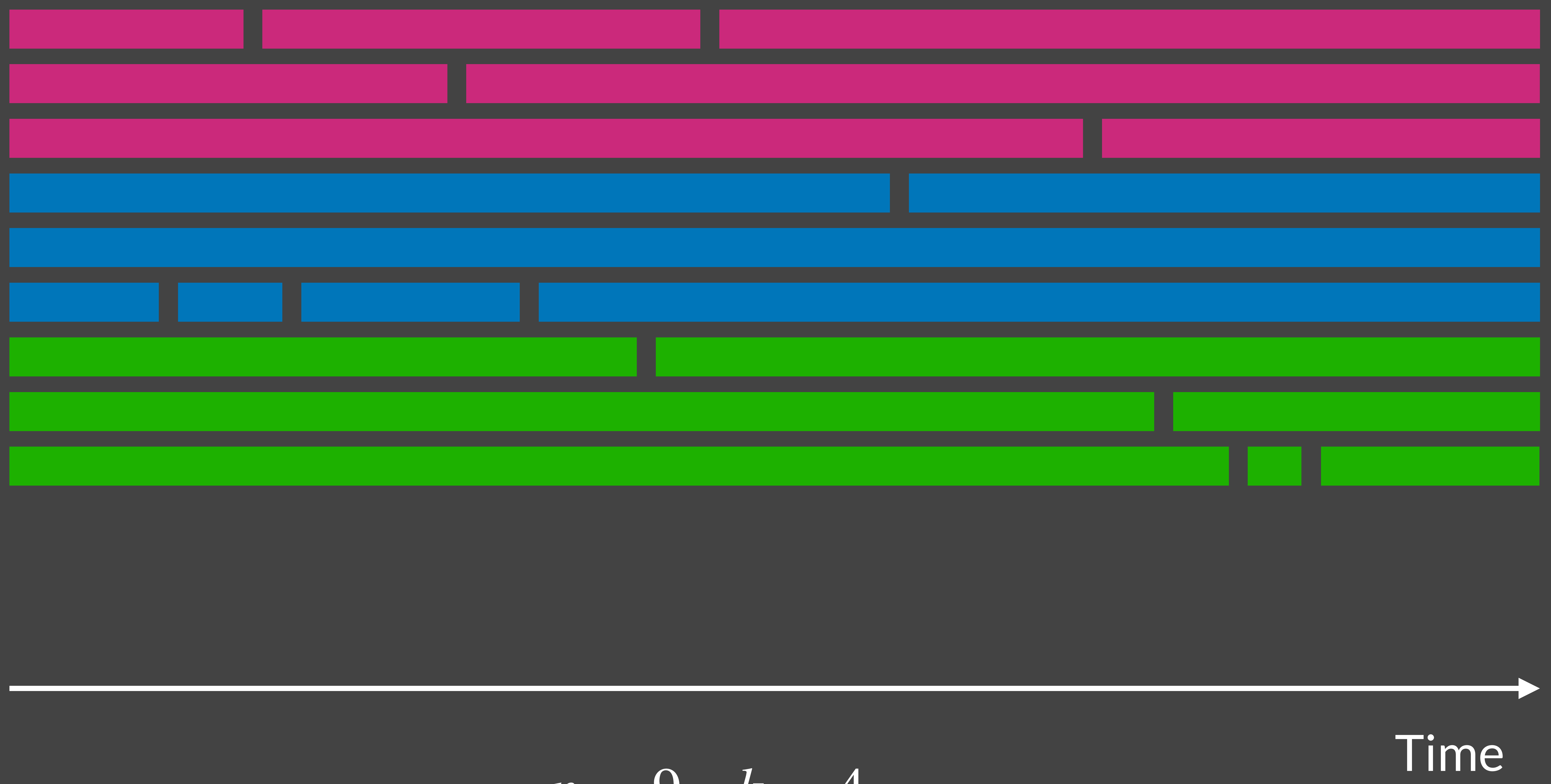
	Classic	Block-Aware
Offline	1	O(log k)
Deterministic Online	k	k
Randomized Online	O(log k)	O(log^2 k)

Our Result

Also show $\Omega(\beta)$ lower bound for randomized algorithms in **fetching cost model…**

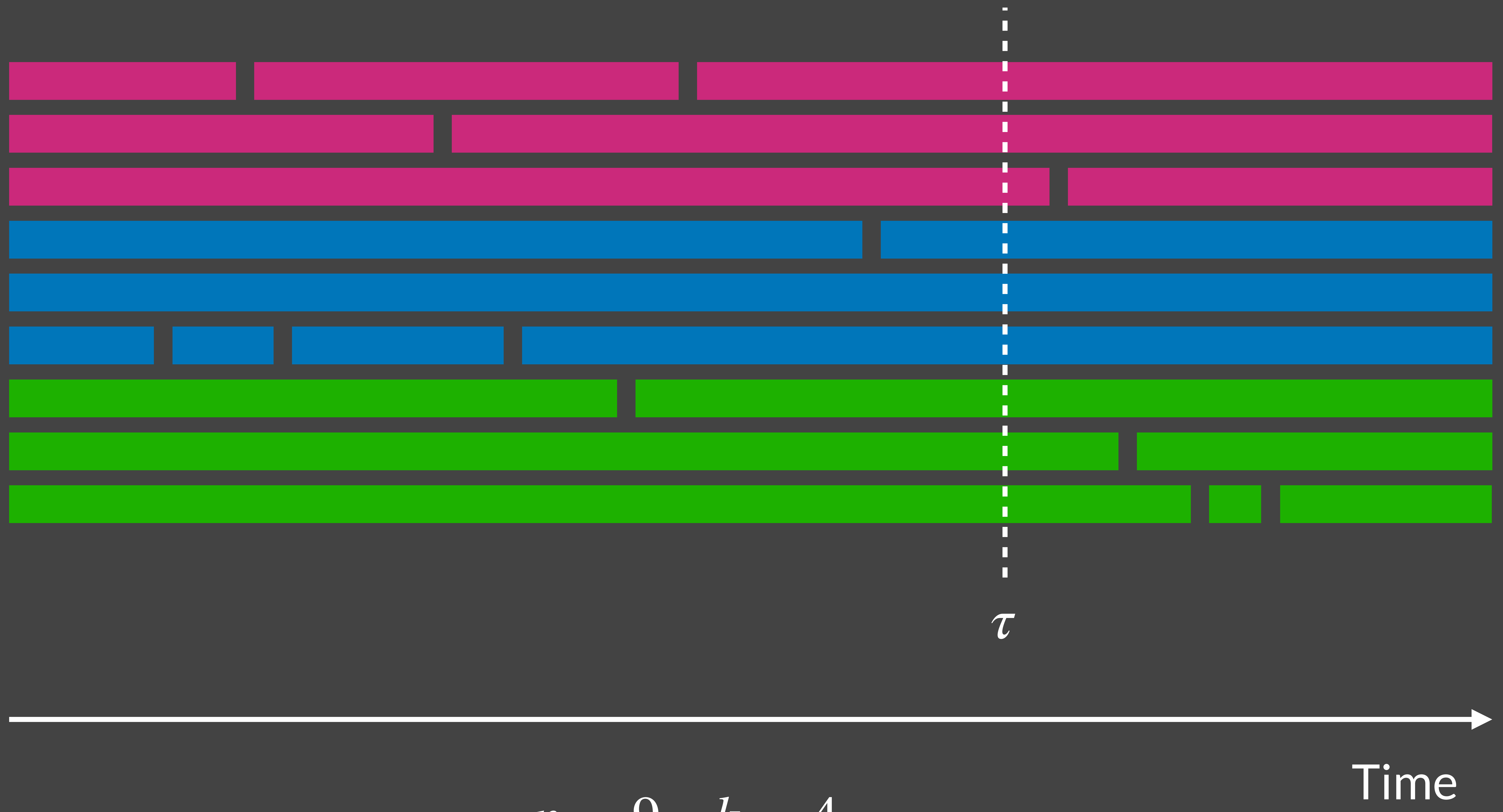
… separation of eviction/ fetching cost models!

What does this have to do with Submodular Cover?



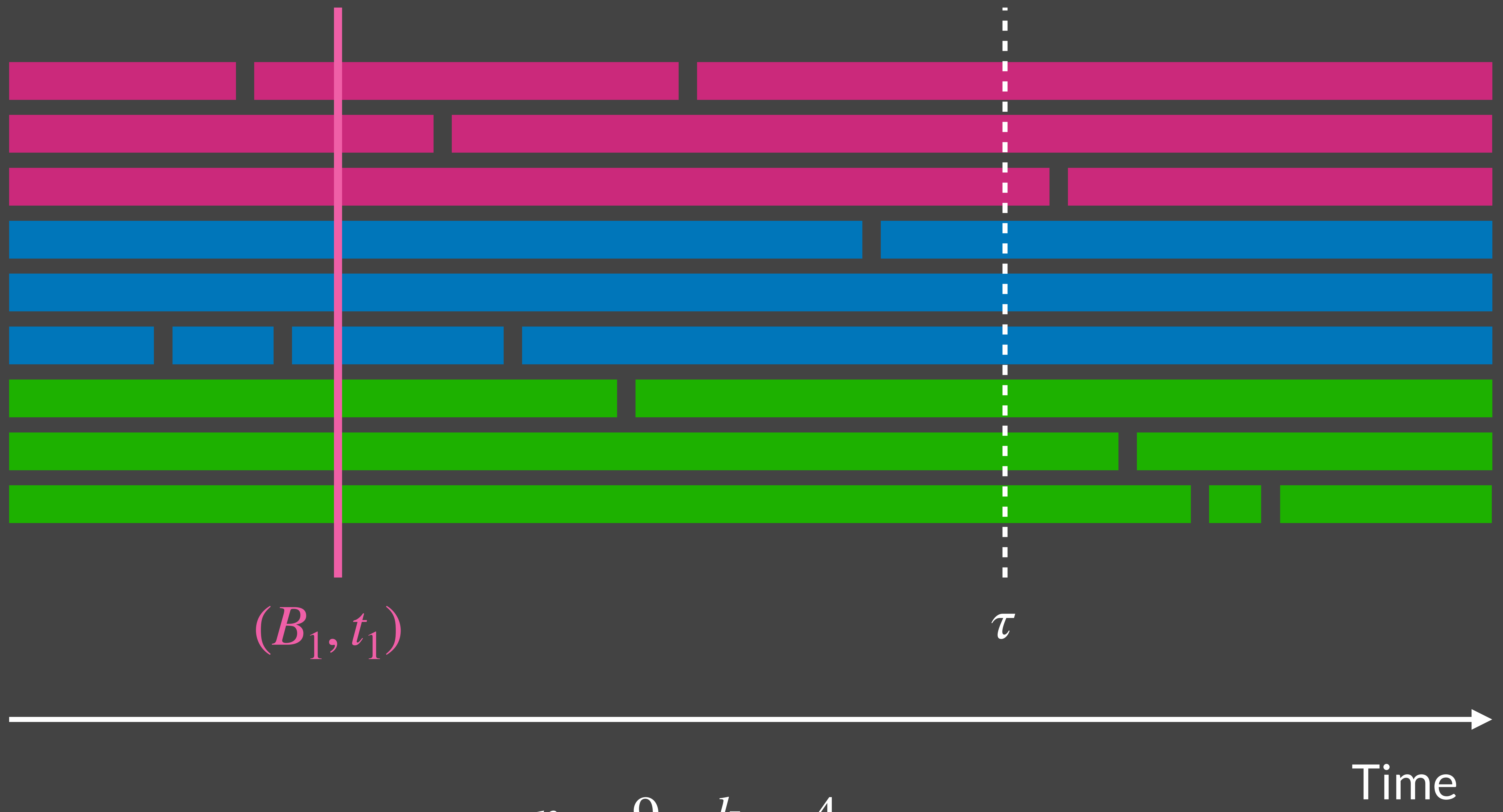
$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?



$$n = 9, \quad k = 4$$

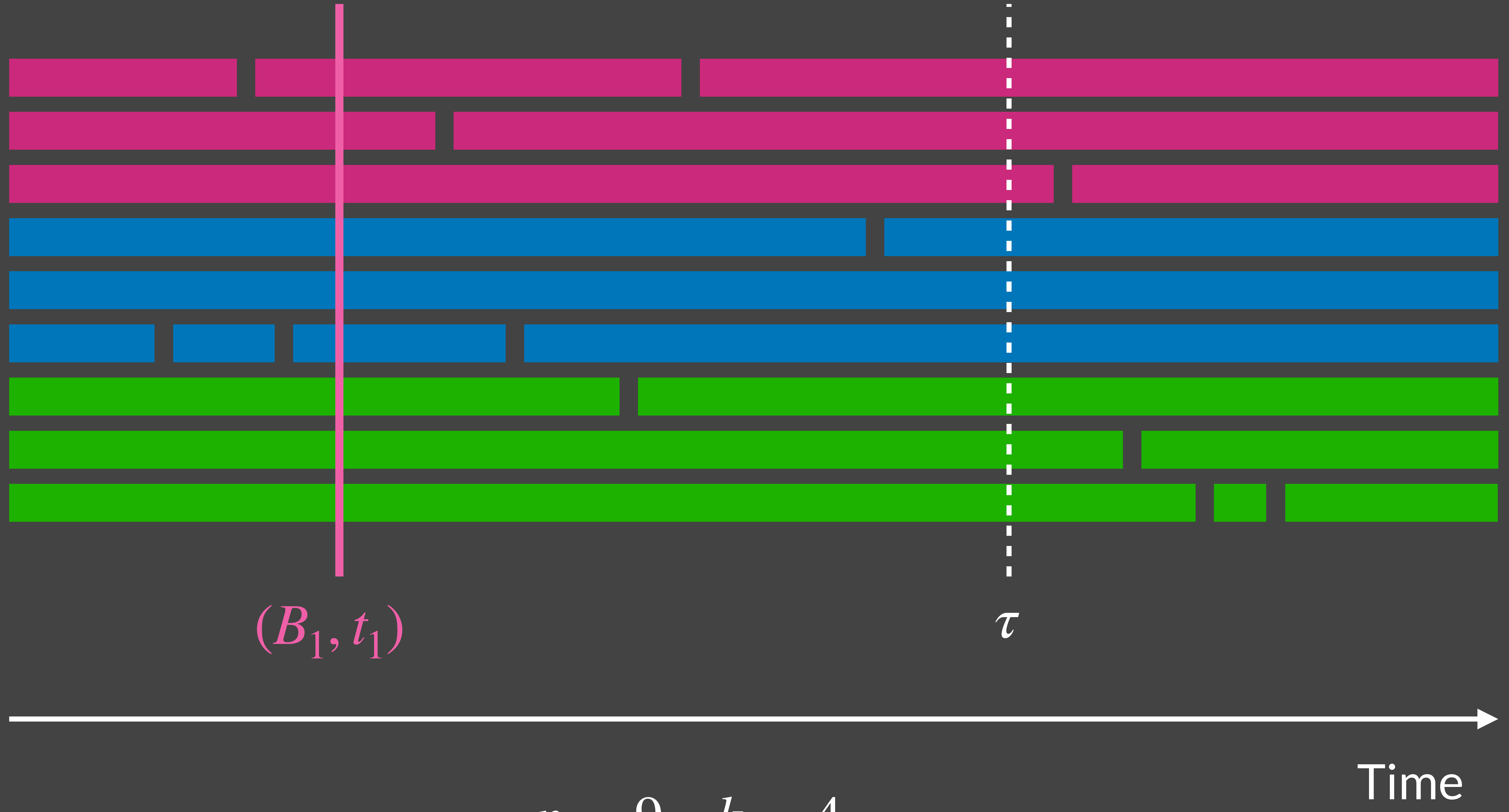
What does this have to do with Submodular Cover?



$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?

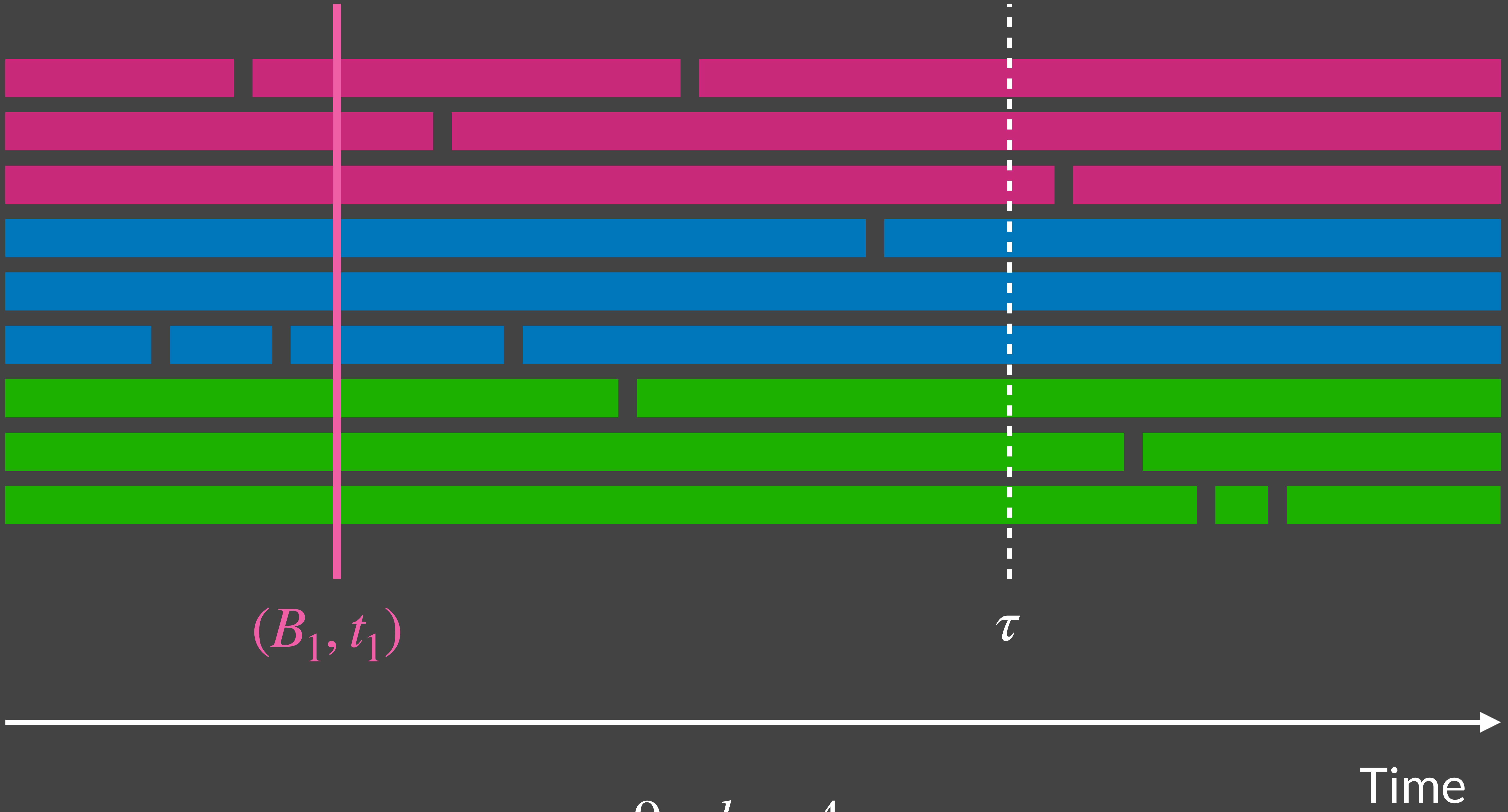
Reduces overflow at time τ by ____.



$$n = 9, \quad k = 4$$

What does this have to do with Submodular Cover?

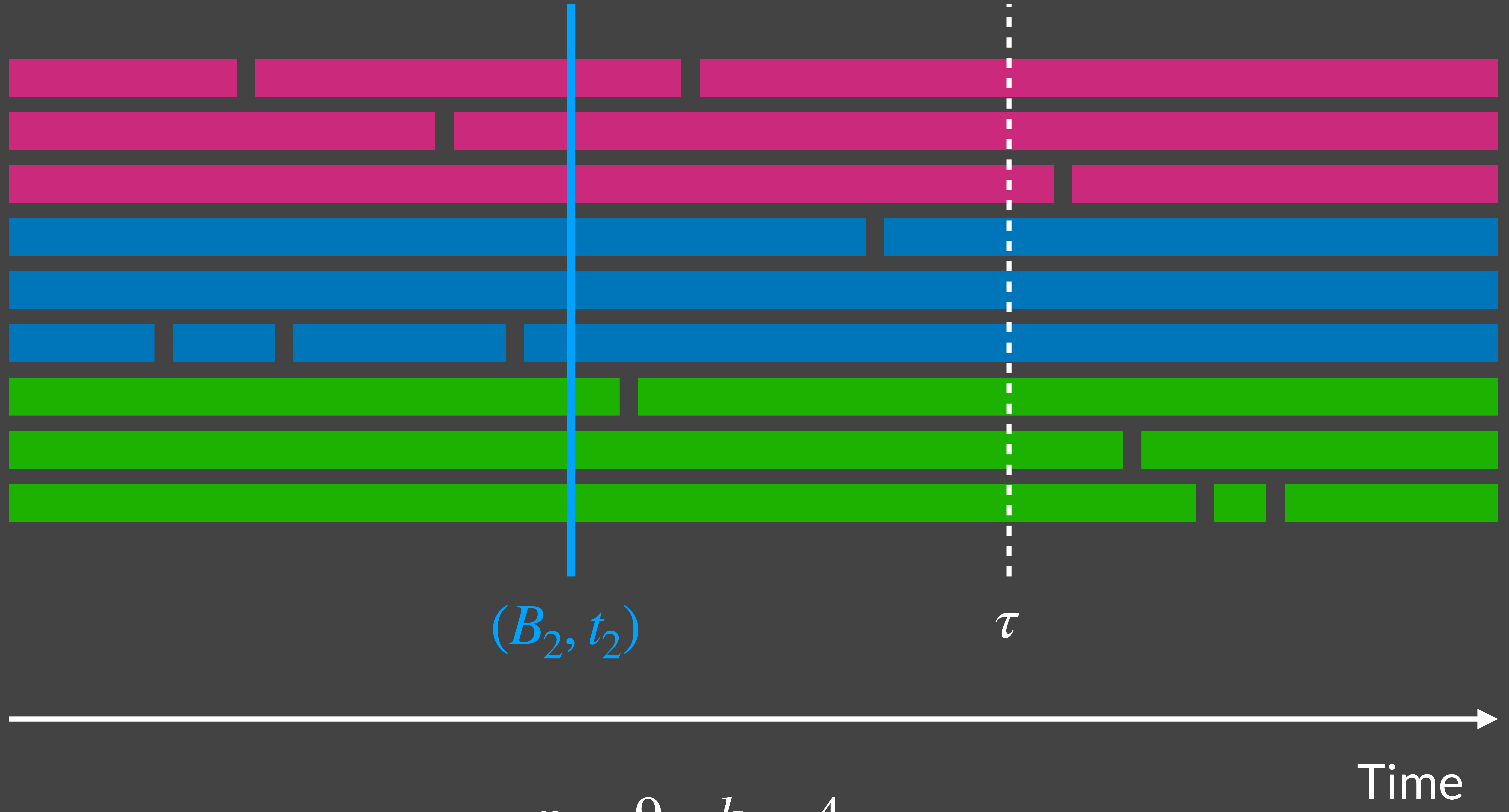
Reduces overflow at time τ by $\underline{\ 1\ }$.



$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?

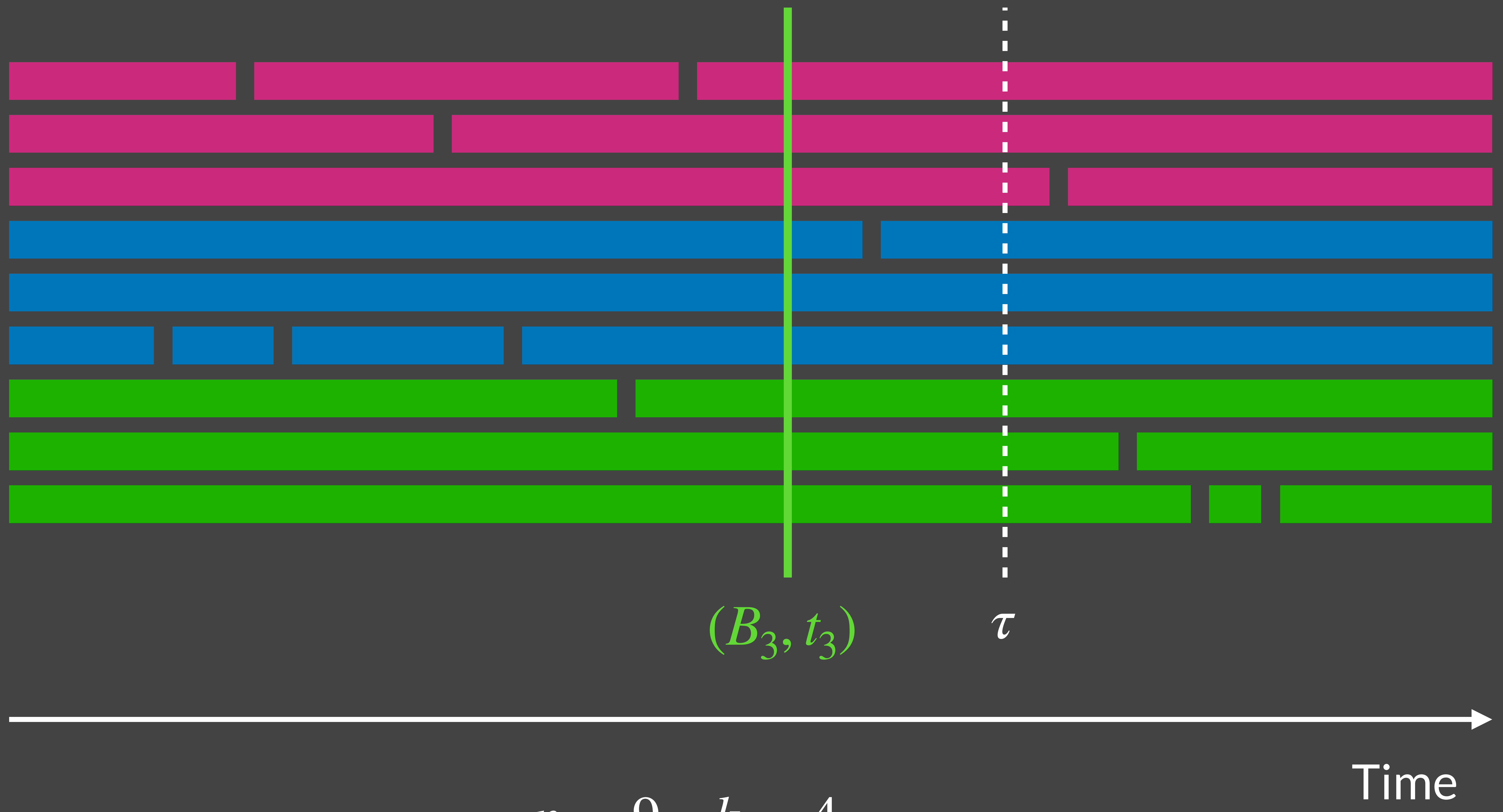
Reduces overflow at time τ by <u>2</u>.



$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?

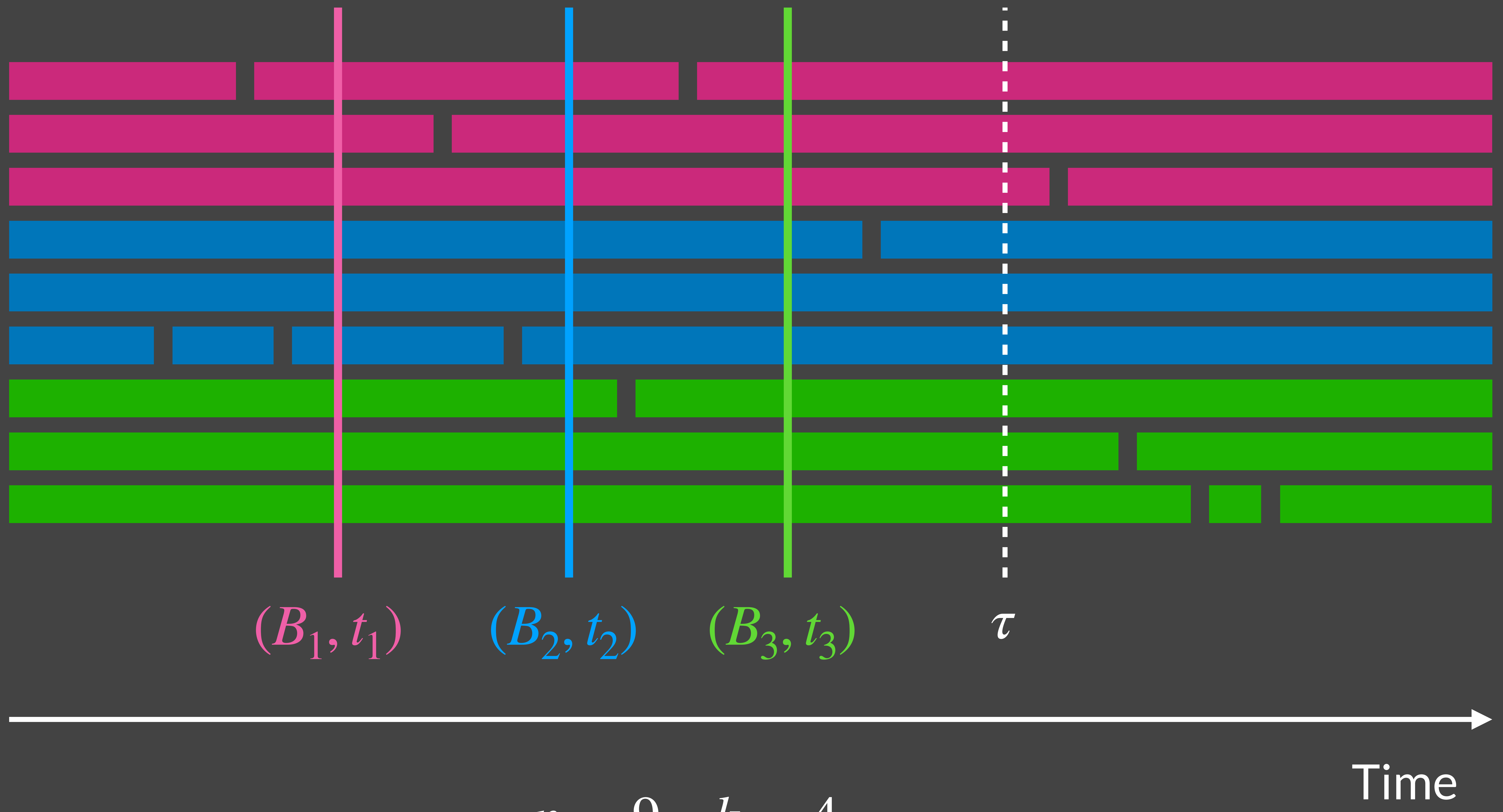
Reduces overflow at time τ by $\underline{\ 3\ }$.



$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?

Reduces overflow at time τ by _5_.

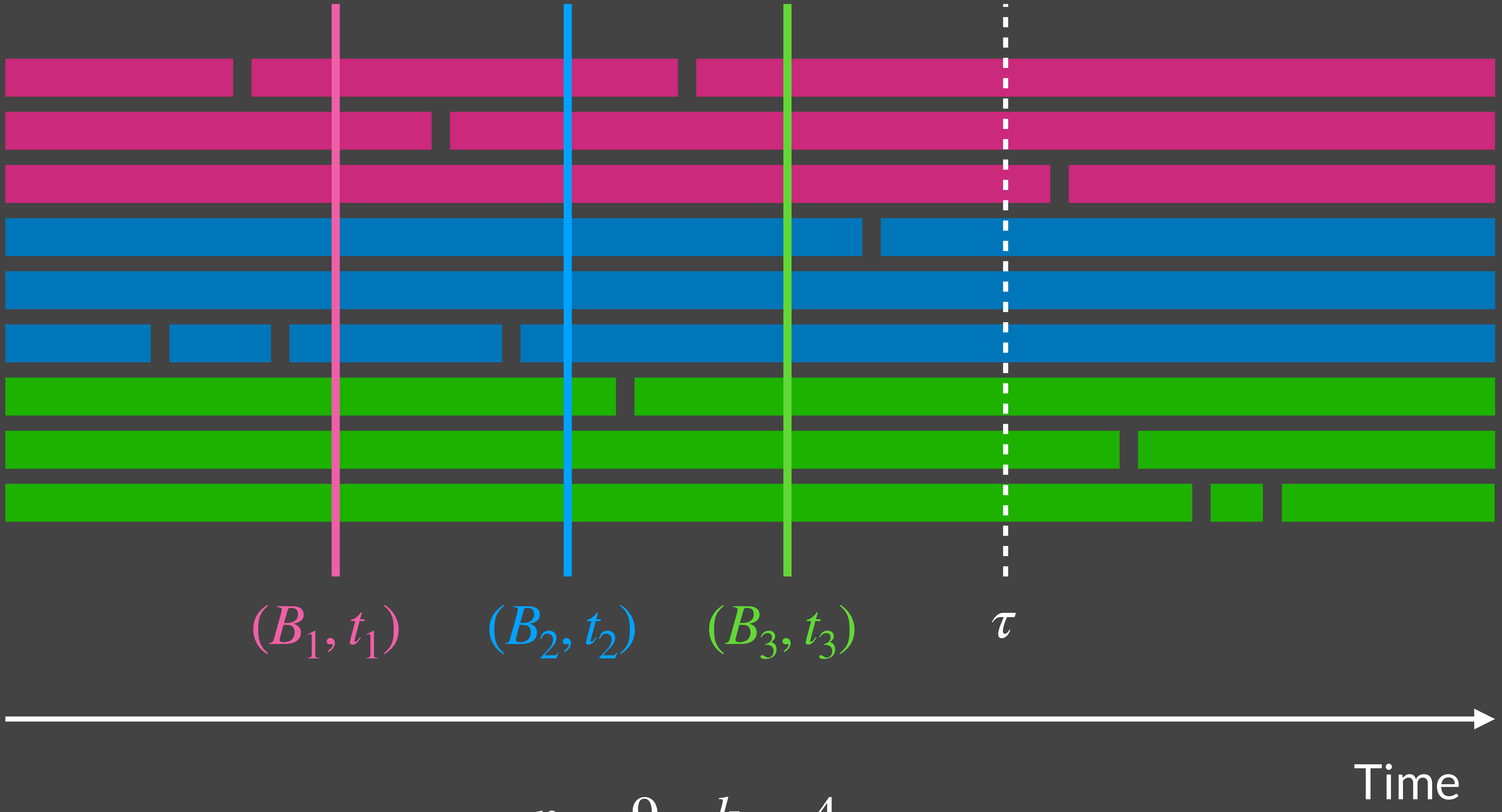


$n = 9, \quad k = 4$

What does this have to do with Submodular Cover?

Reduces overflow at time τ by <u>5</u>.

f^{τ} := "reduction in overflow at time τ" is submodular!



$$n = 9, \quad k = 4$$

Formulation as Submodular Cover

Formulation as Submodular Cover

$$\min_S |S|$$

$$\forall \tau : \quad f^\tau(S) \geq n - k$$

Formulation as Submodular Cover

$$\min_S |S|$$

$$\forall \tau : \quad f^\tau(S) \geq n - k$$

Where S is an eviction schedule, e.g. $S = \{(B_1, t_1), (B_2, t_2), \ \ldots\}$

Formulation as Submodular Cover

$$\min_{S} |S|$$

$$\forall \tau : \quad f^{\tau}(S) \geq n - k$$

Where S is an eviction schedule, e.g. $S = \{(B_1, t_1), (B_2, t_2), \ \ldots\}$

This is an instance of Online Submodular Cover!

Formulation as Submodular Cover

$$\min_S |S|$$

$$\forall \tau : \quad f^\tau(S) \geq n - k$$

Where S is an eviction schedule, e.g. $S = \{(B_1, t_1), (B_2, t_2), \ \ldots\}$

This is an instance of Online Submodular Cover!

Bounds from Part I <u>too weak</u>, depend on total time T.

Formulation as Submodular Cover

$$\min_{S} |S|$$

$$\forall \tau : \quad f^{\tau}(S) \geq n - k$$

Where S is an eviction schedule, e.g. $S = \{(B_1, t_1), (B_2, t_2), \ \ldots\}$

This is an instance of Online Submodular Cover!

Bounds from Part I too weak, depend on total time T.

We show our bounds via finer analysis… but reuse some ideas!

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

Part III — Random Order Online Set Cover

Conclusion

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

➡ Part III — Random Order Online Set Cover

Conclusion

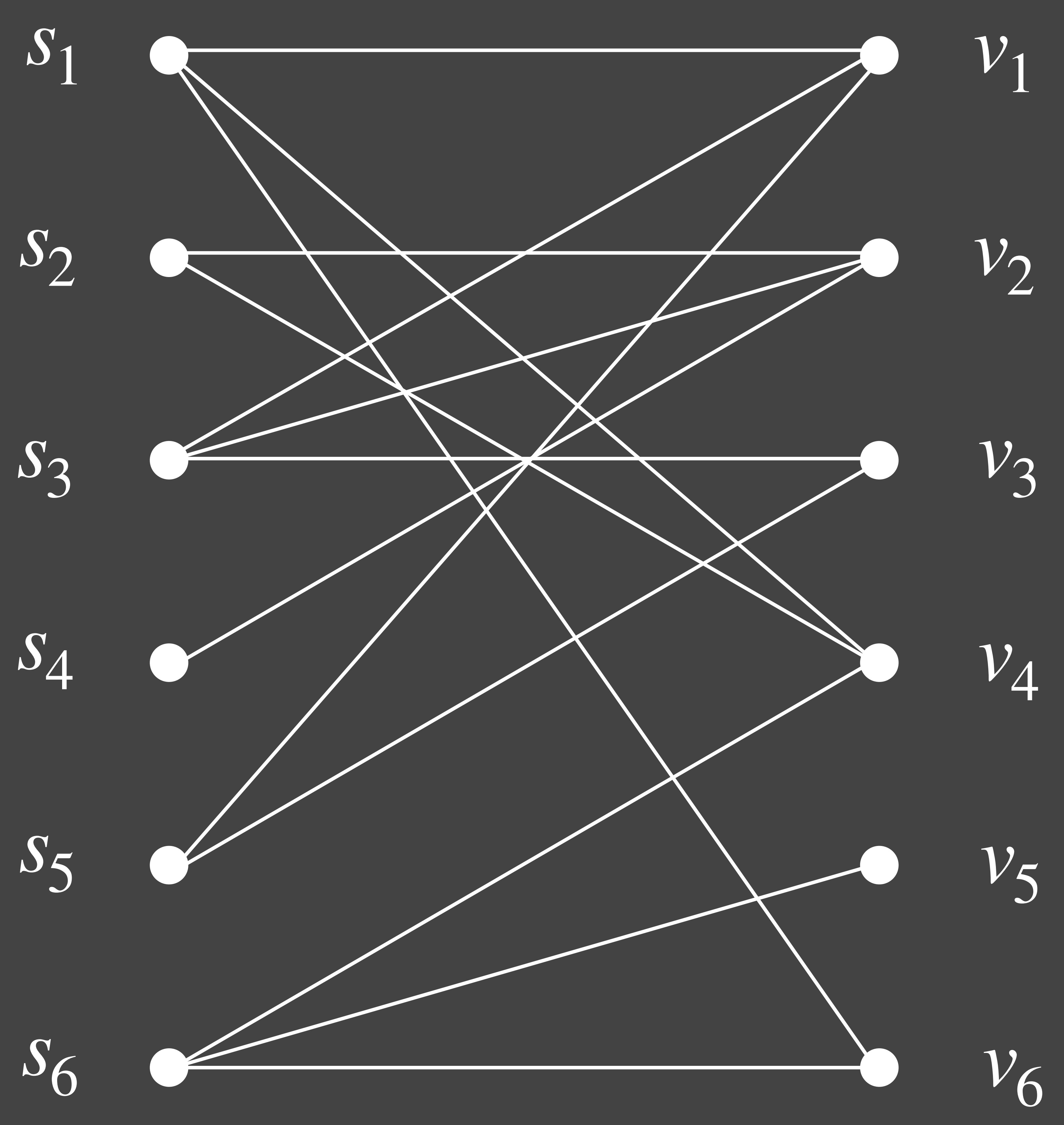
Part III — Random Order Online Set Cover

with Anupam Gupta and Gregory Kehne

Random Order (RO) Online Set Cover

$\mathcal{S}$

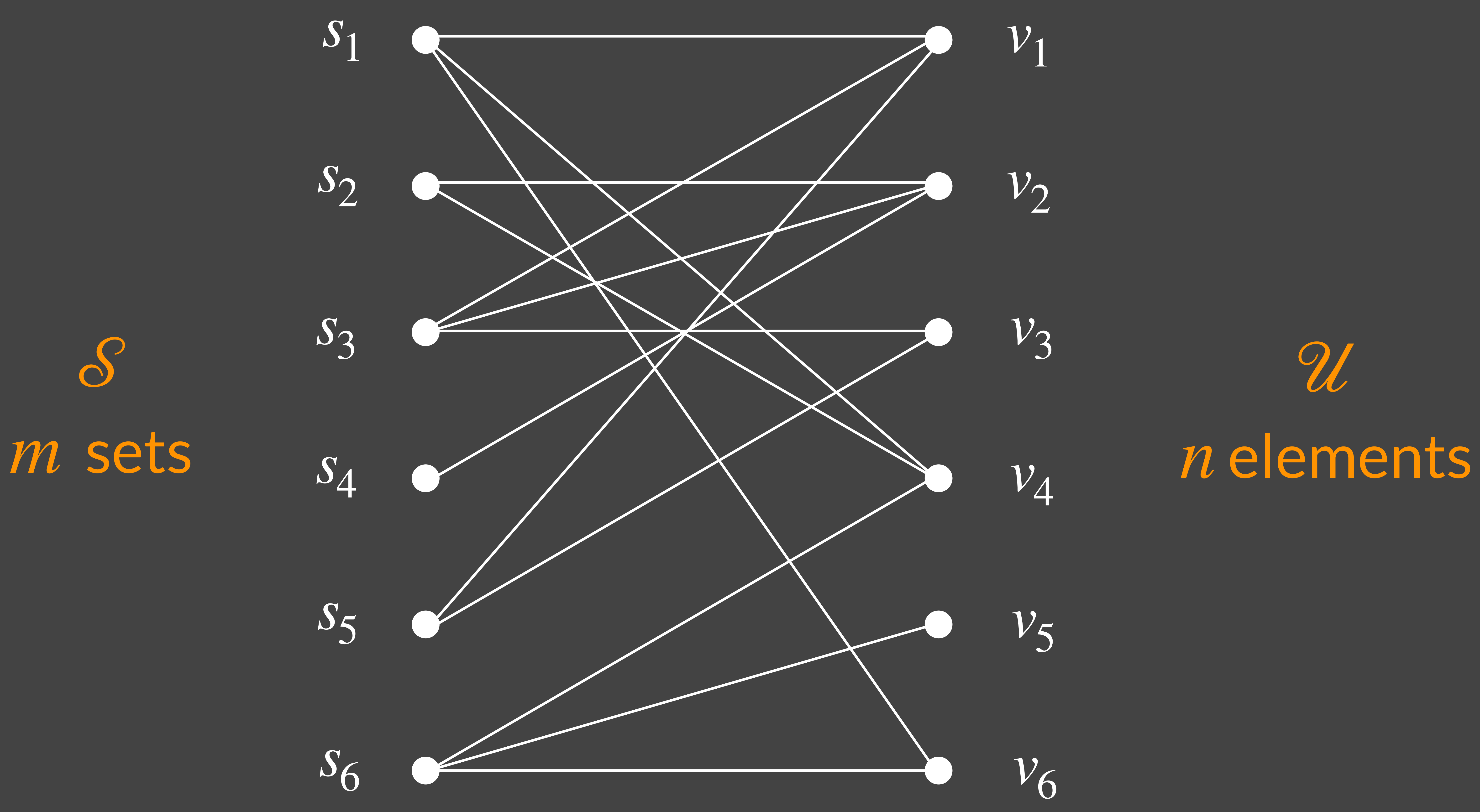
m sets



$\mathcal{U}$

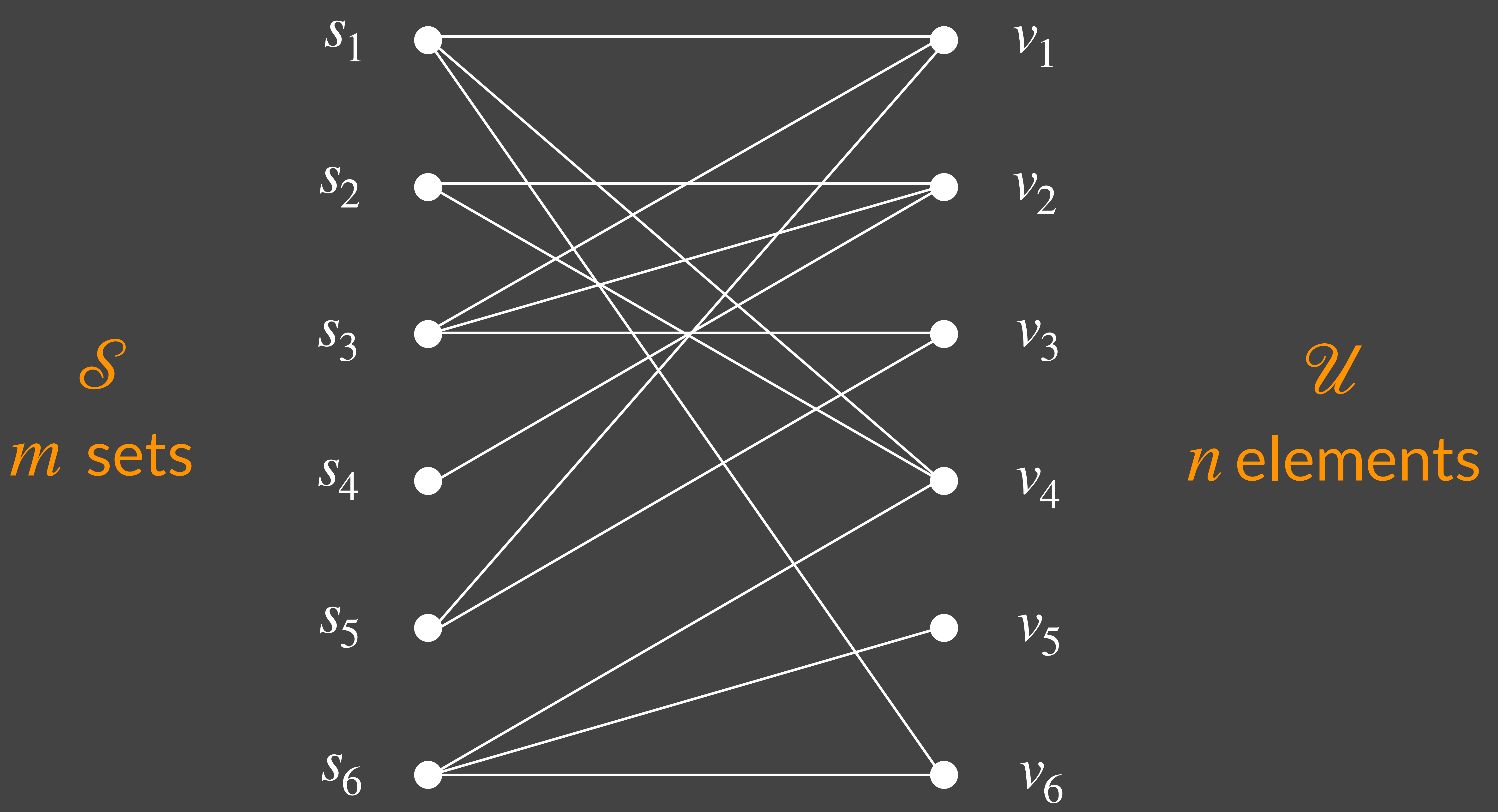
n elements

Random Order (RO) Online Set Cover

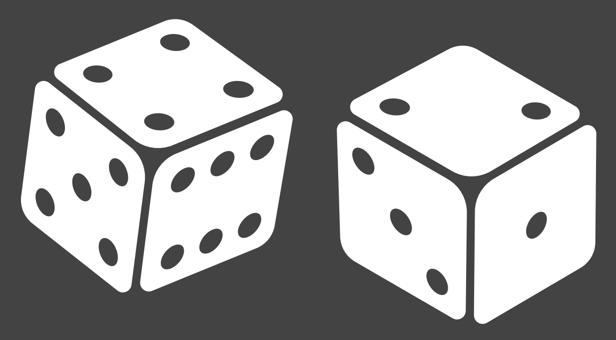


v_1
v_2
v_3
v_4
v_5
v_6

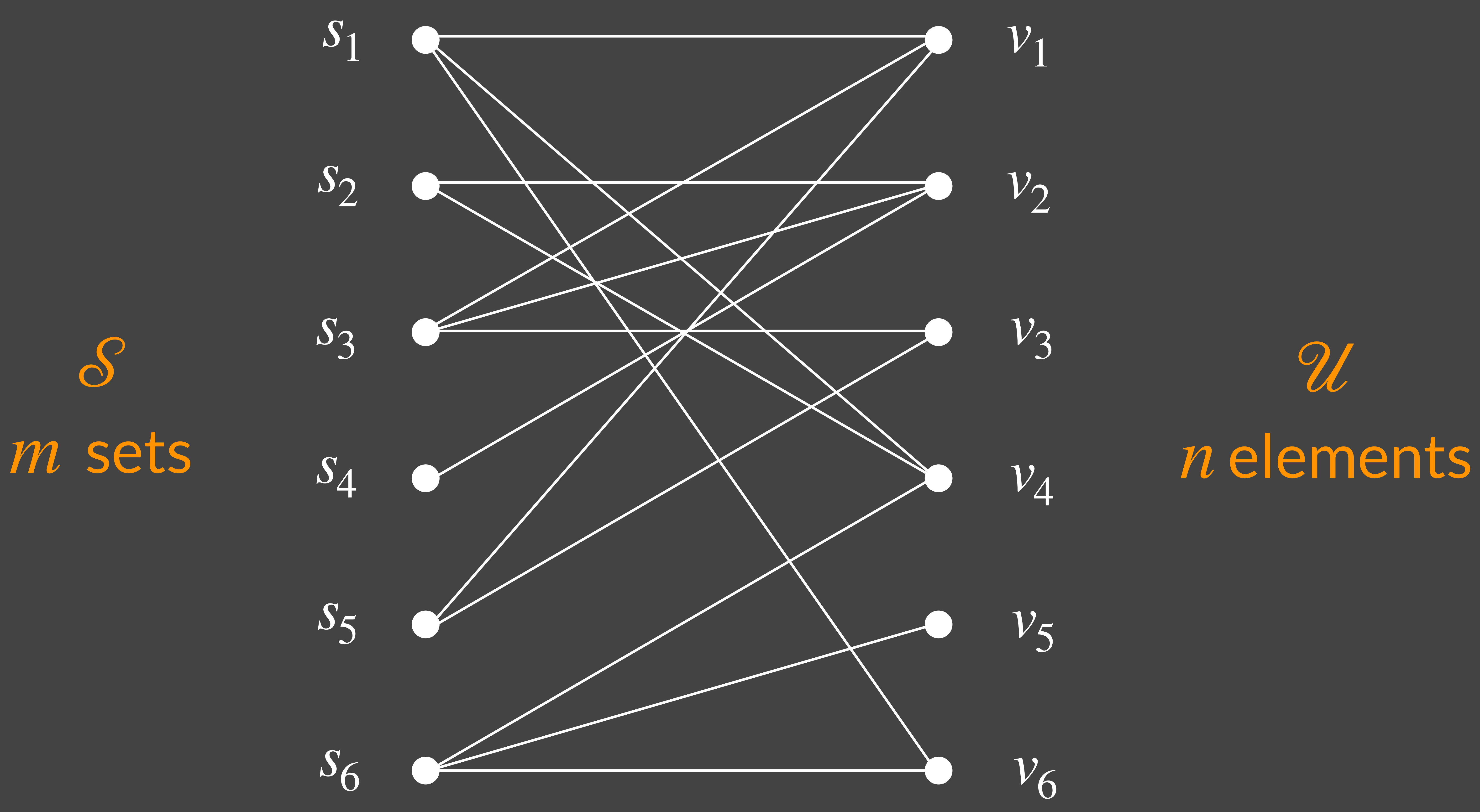
Random Order (RO) Online Set Cover



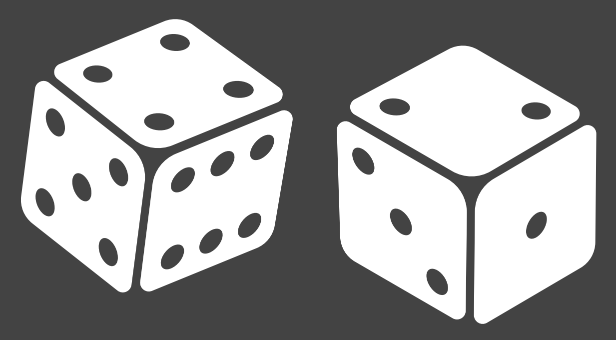
v_1
v_2
v_3
v_4
v_5
v_6



Random Order (RO) Online Set Cover



v_1
v_4
v_5
v_6
v_2
v_3



Random Order (RO) Online Set Cover

$\mathcal{S}$

m sets

s_1

s_2

s_3

s_4

s_5

s_6

$\mathcal{U}$

n elements

v_1

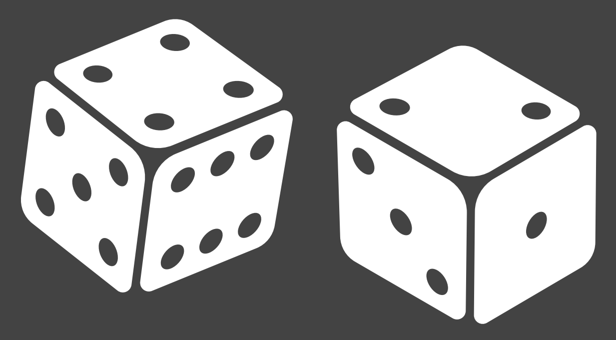
v_4

v_5

v_6

v_2

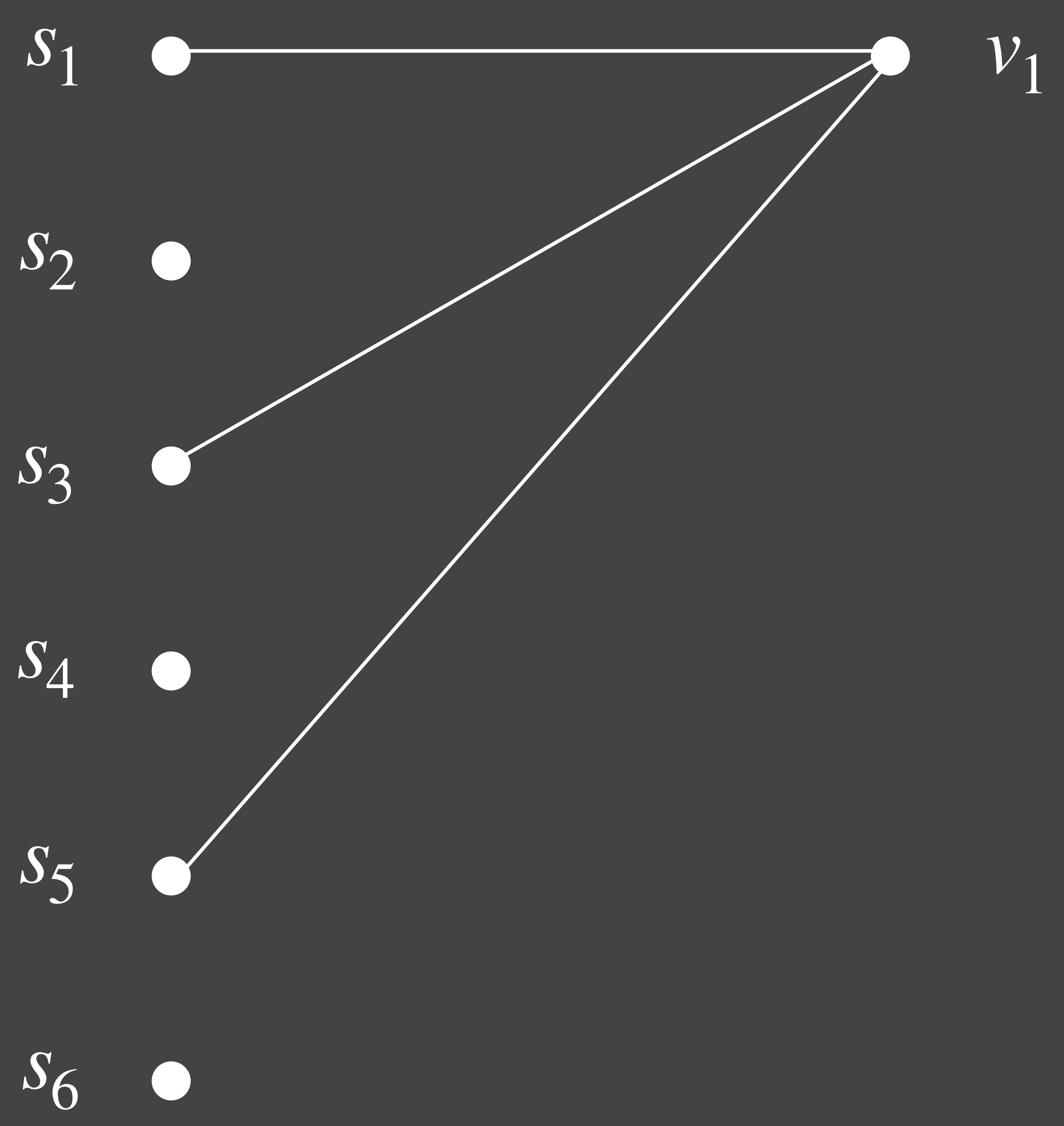
v_3



Random Order (RO) Online Set Cover

$\mathcal{S}$

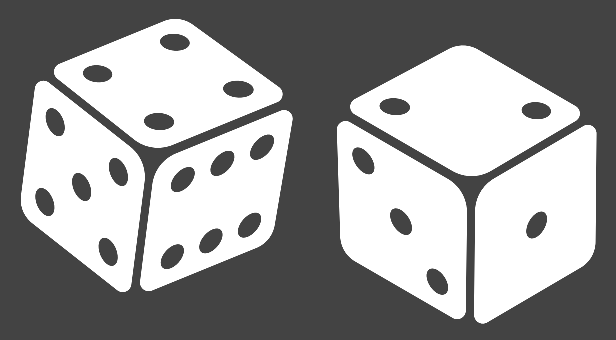
m sets



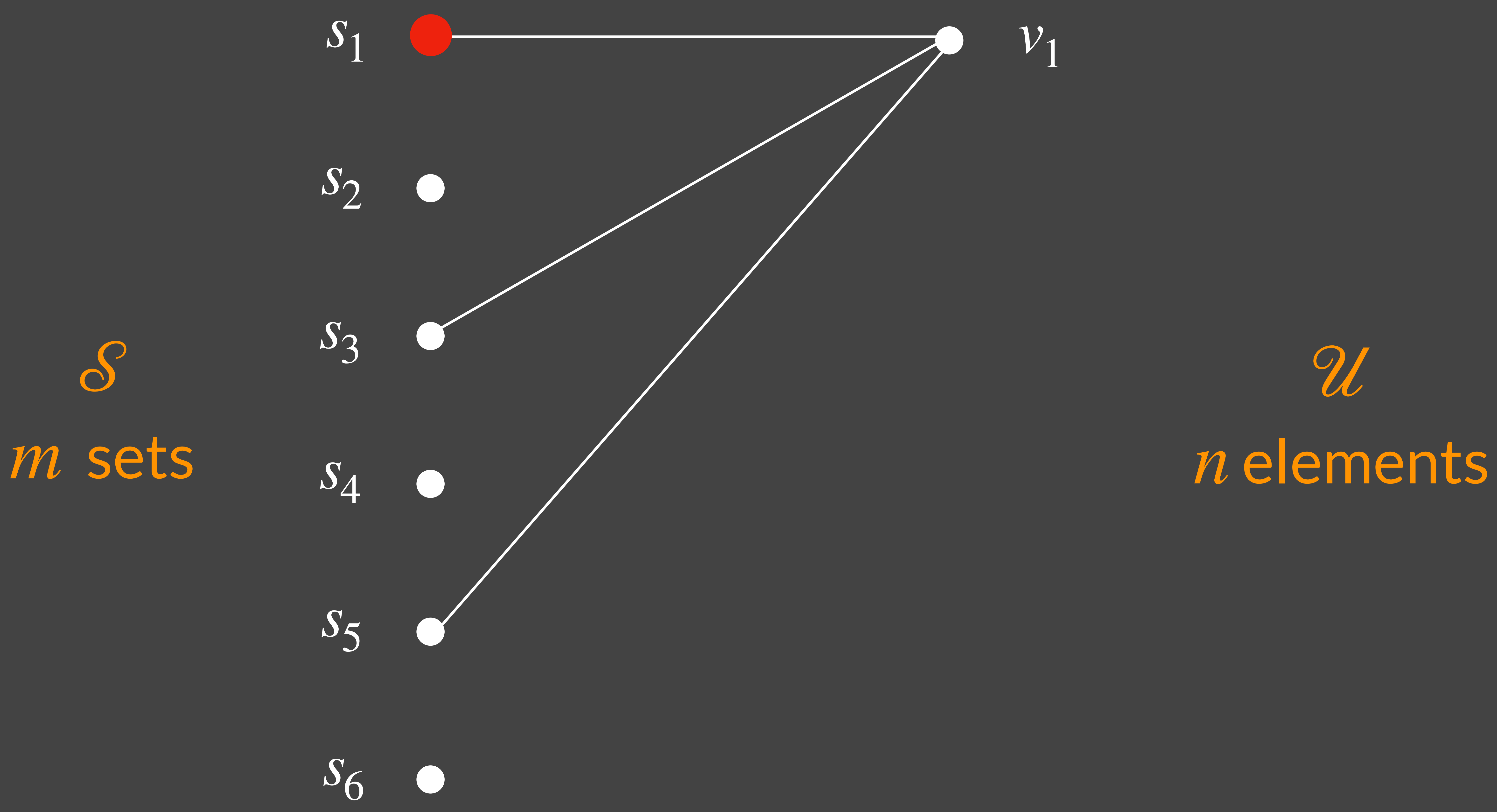
$\mathcal{U}$

n elements

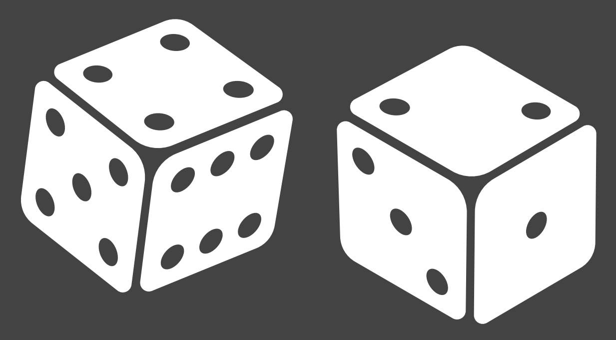
v_1
v_4
v_5
v_6
v_2
v_3



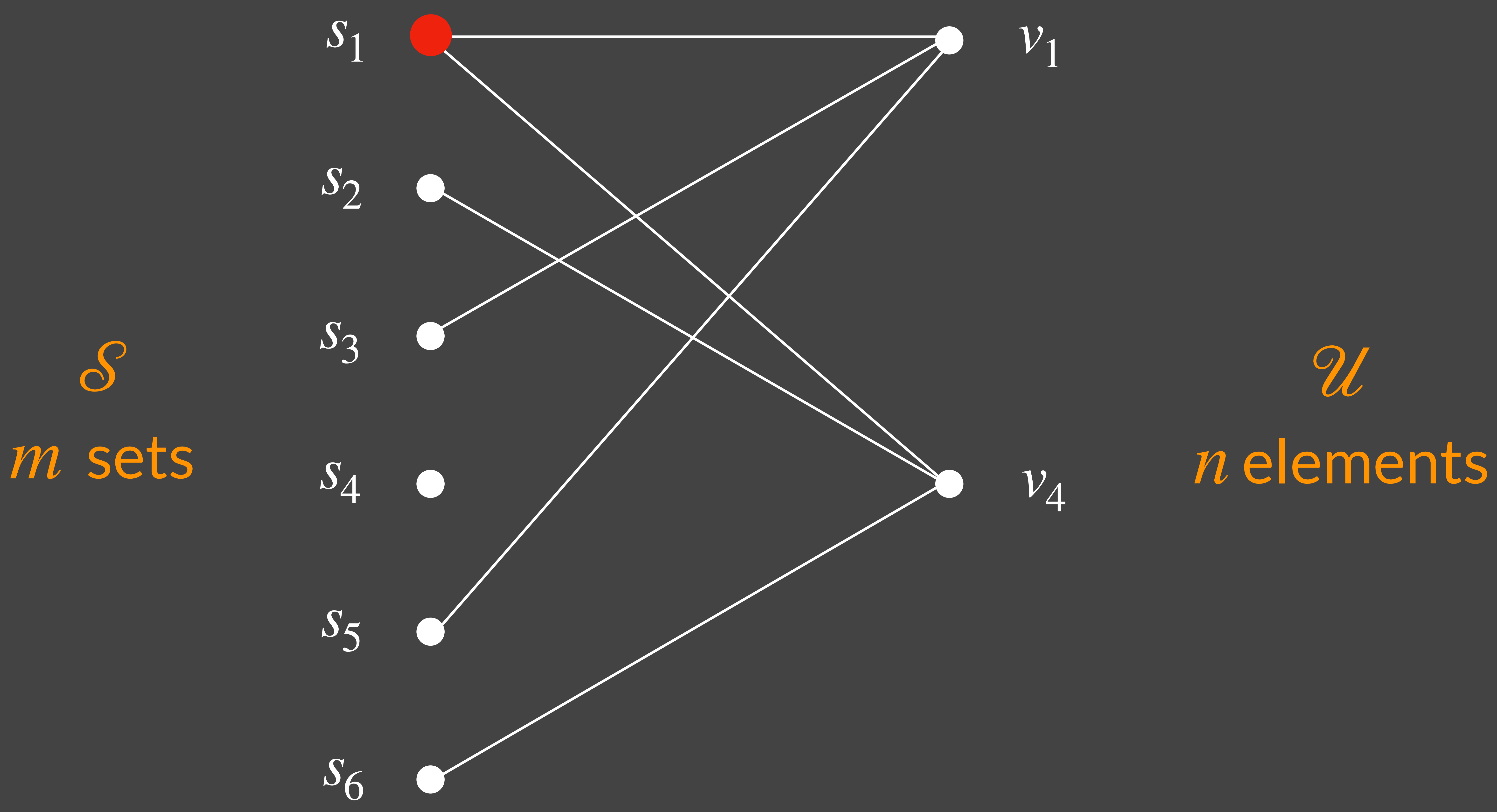
Random Order (RO) Online Set Cover



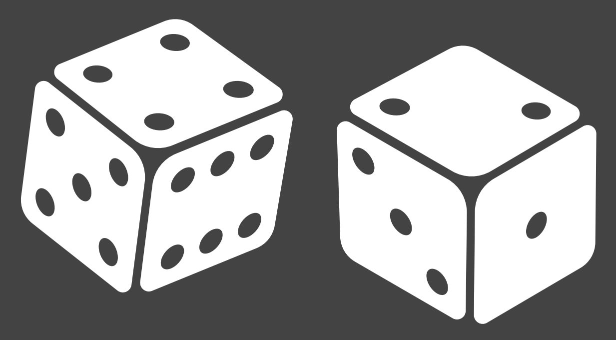
v_1
v_4
v_5
v_6
v_2
v_3



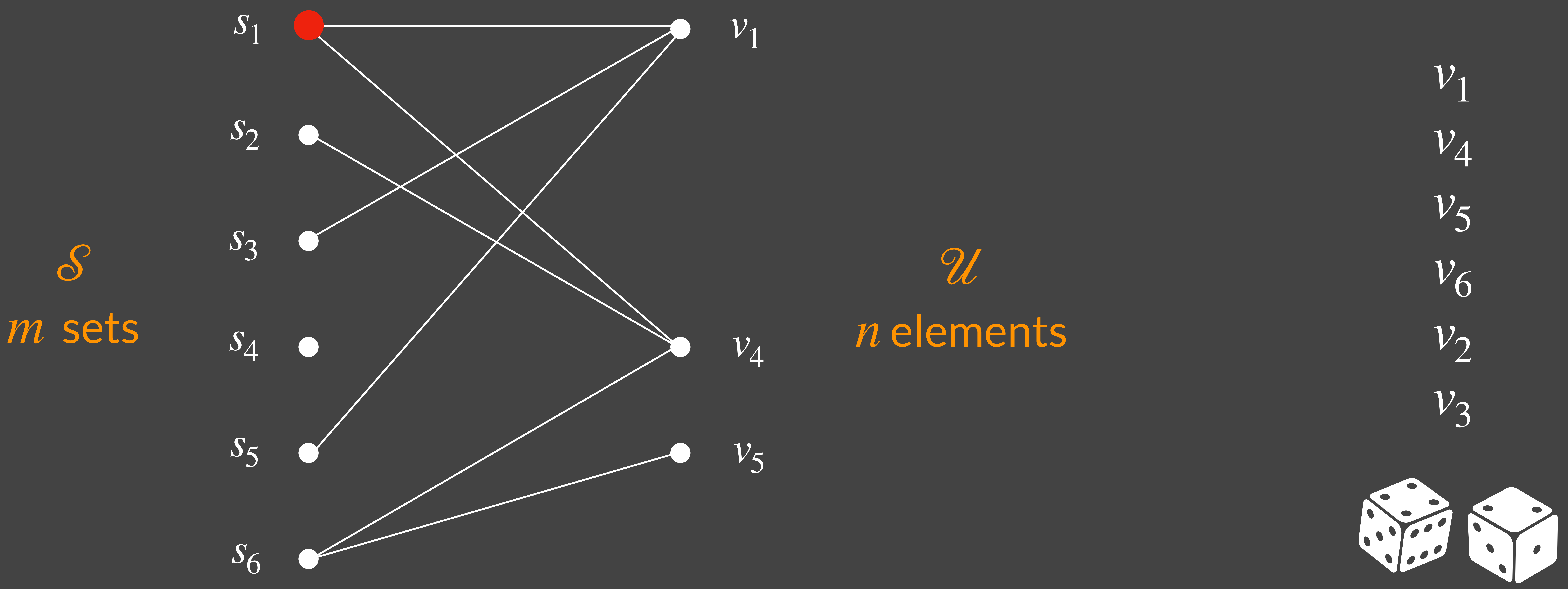
Random Order (RO) Online Set Cover



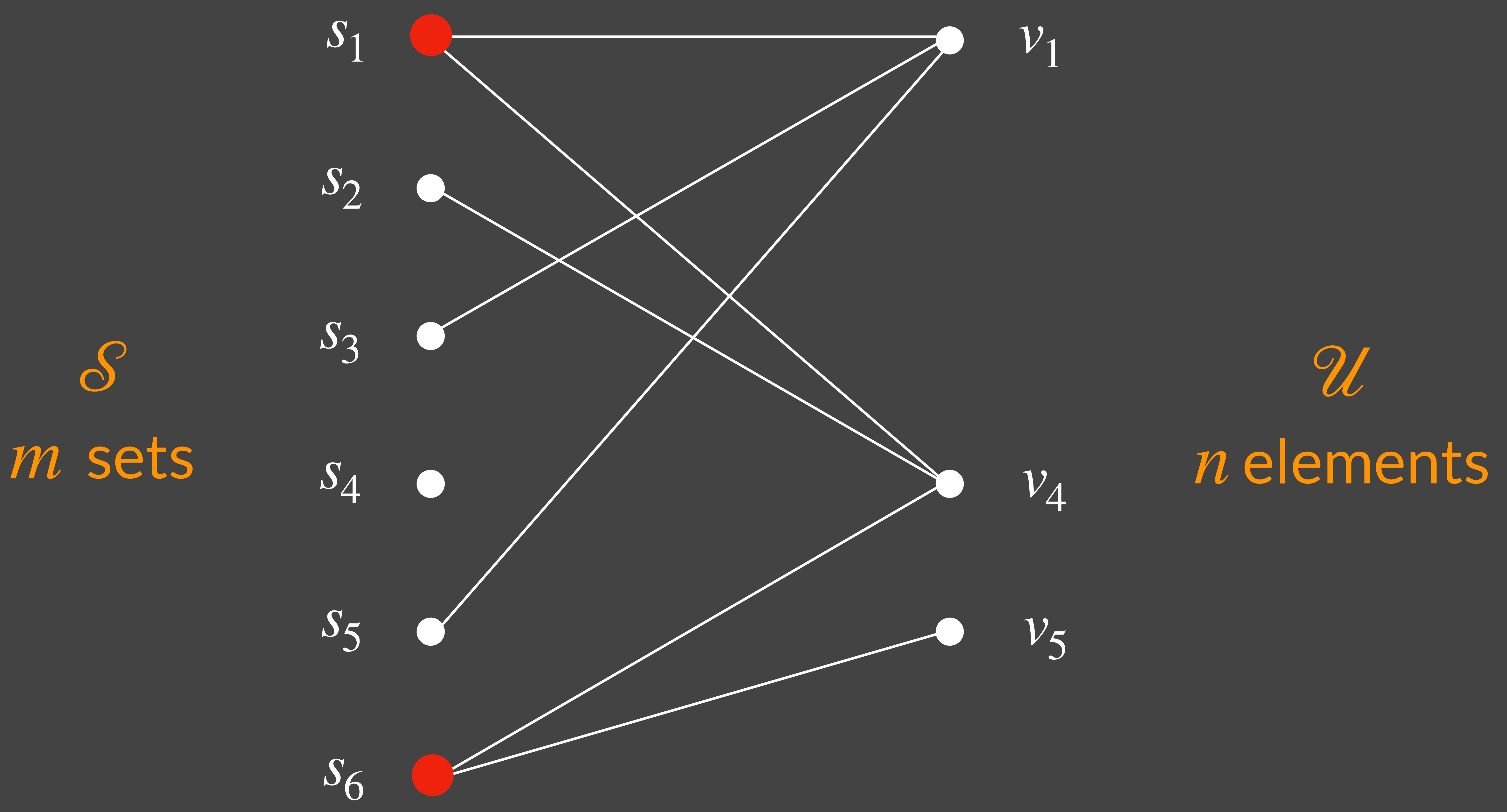
v_1
v_4
v_5
v_6
v_2
v_3



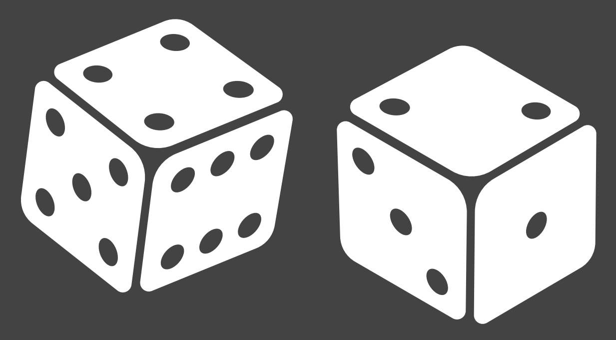
Random Order (RO) Online Set Cover



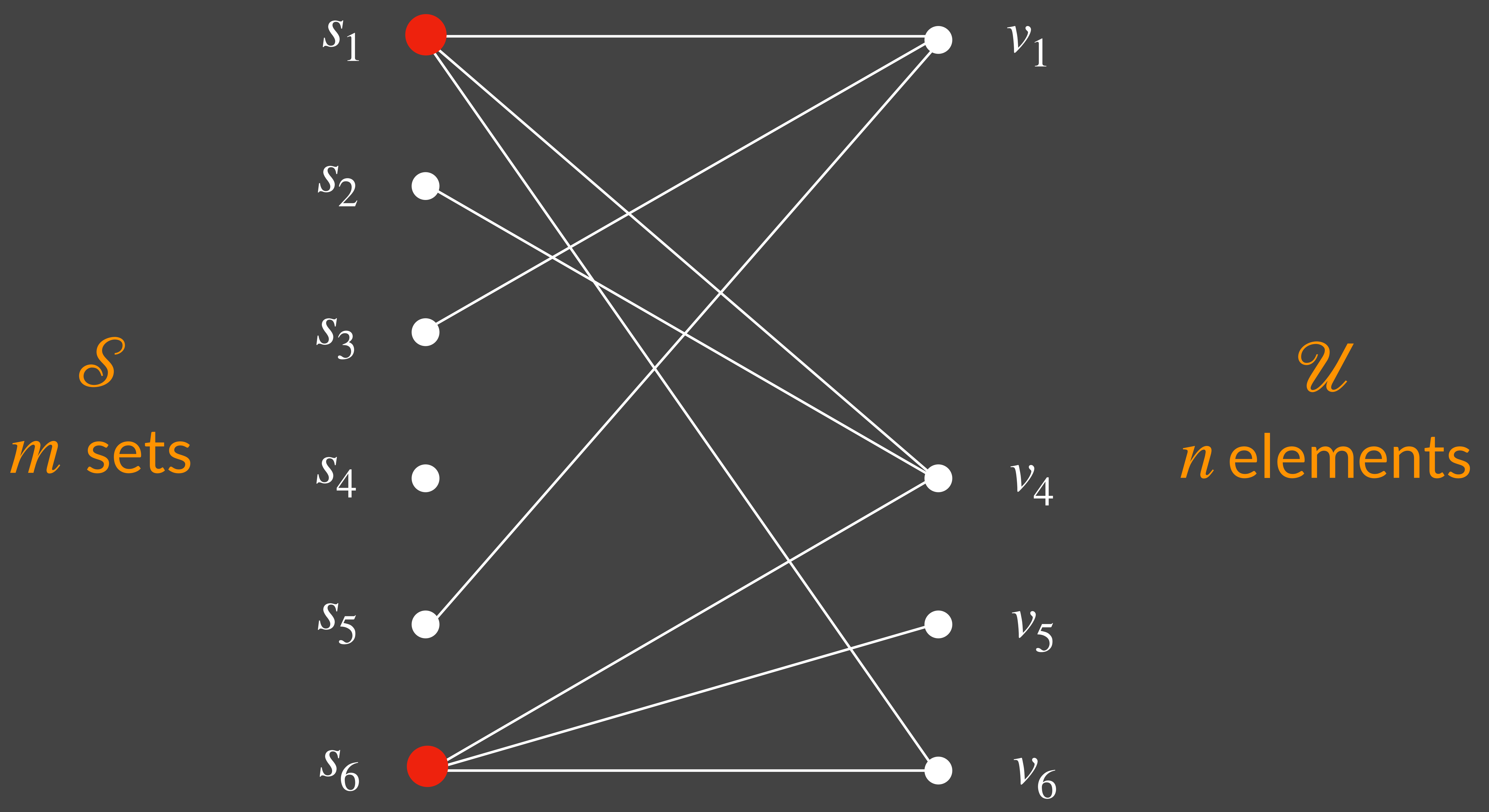
Random Order (RO) Online Set Cover



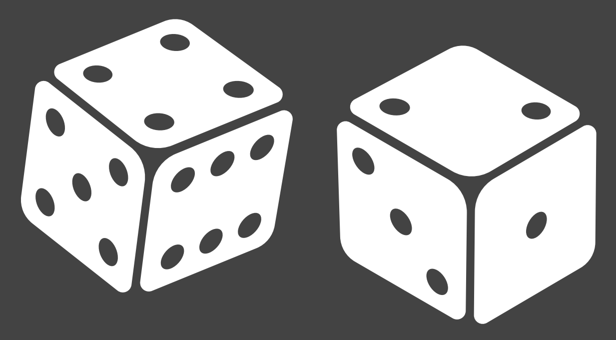
v_1
v_4
v_5
v_6
v_2
v_3



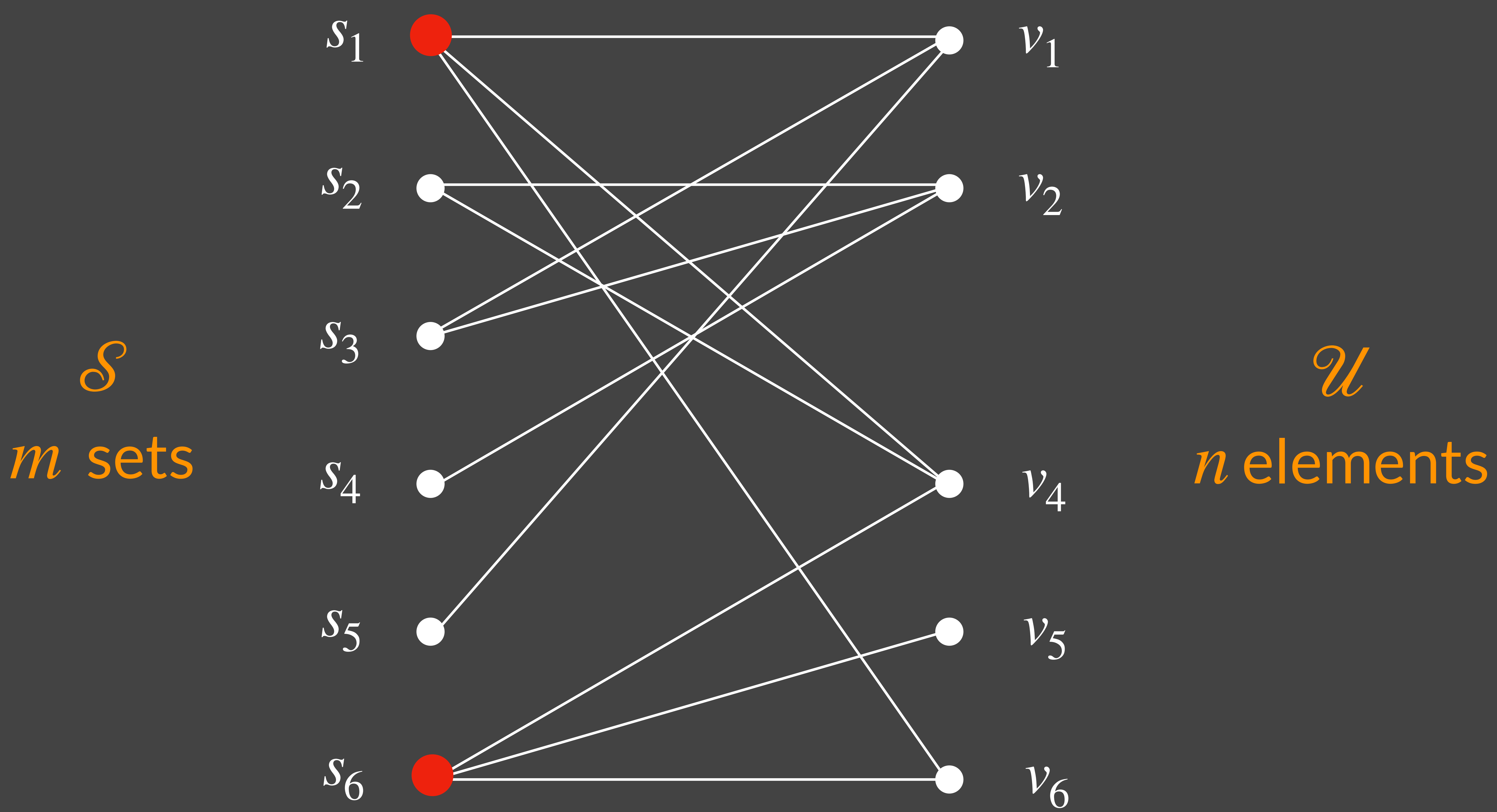
Random Order (RO) Online Set Cover



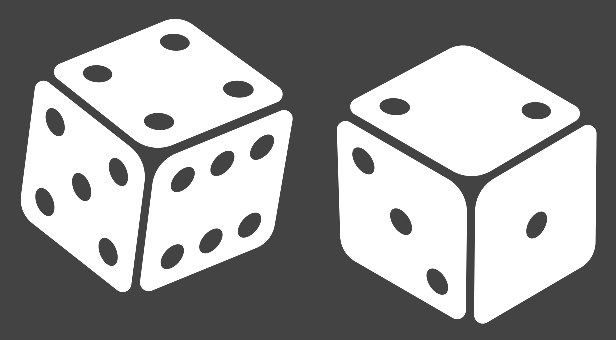
v_1
v_4
v_5
v_6
v_2
v_3



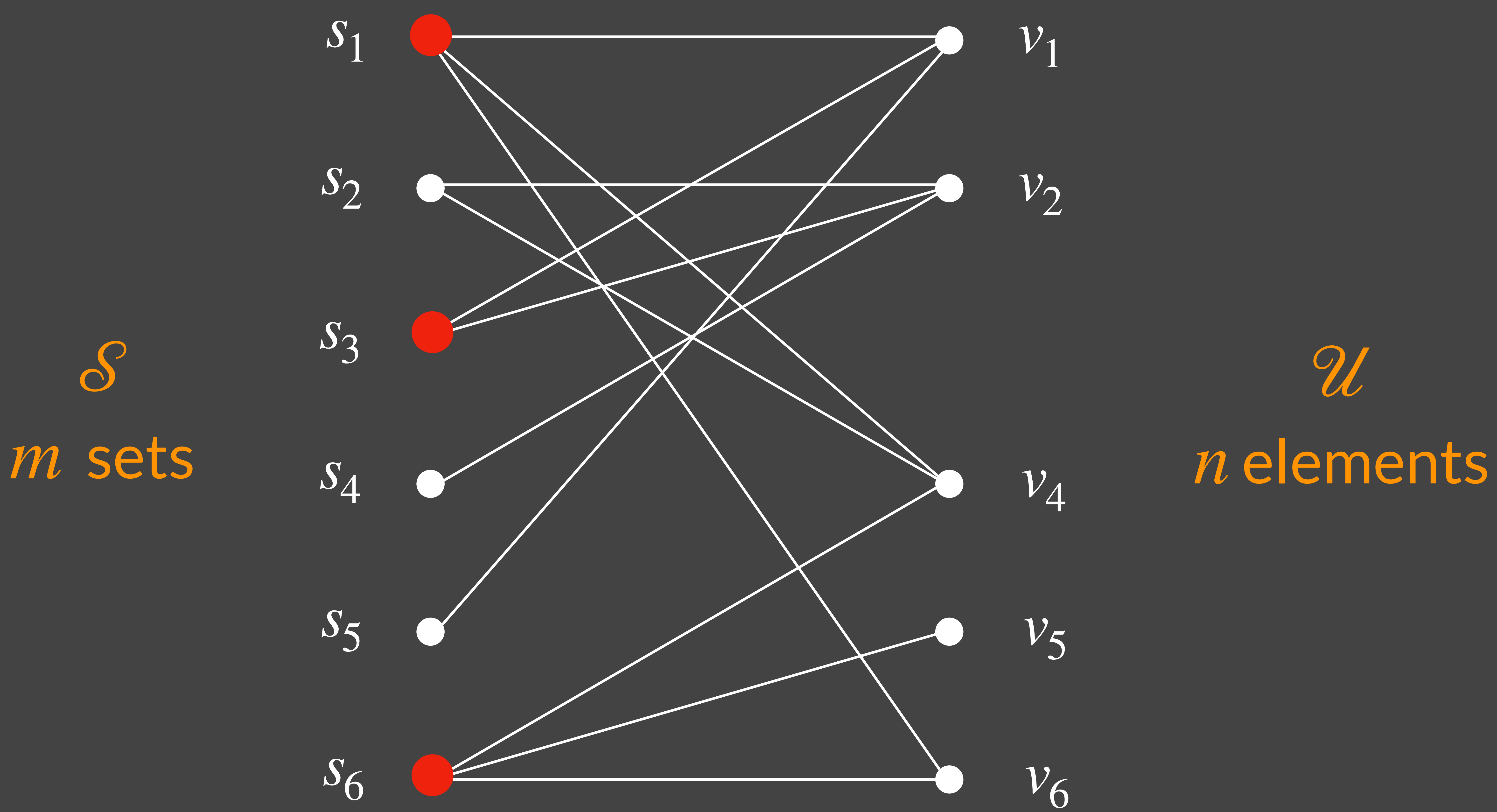
Random Order (RO) Online Set Cover



v_1
v_4
v_5
v_6
v_2
v_3



Random Order (RO) Online Set Cover

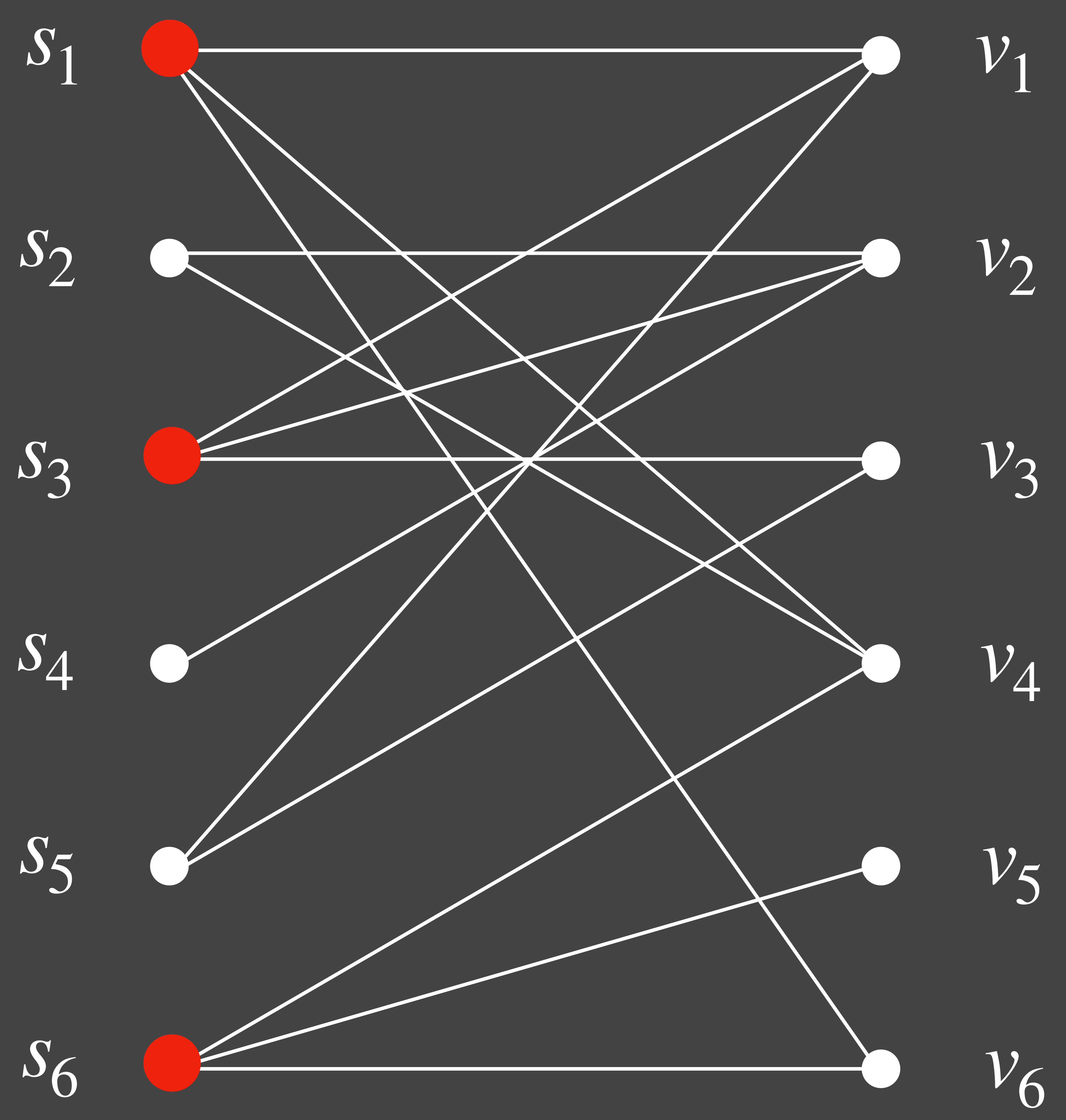


v_1
v_4
v_5
v_6
v_2
v_3

Random Order (RO) Online Set Cover

$\mathcal{S}$

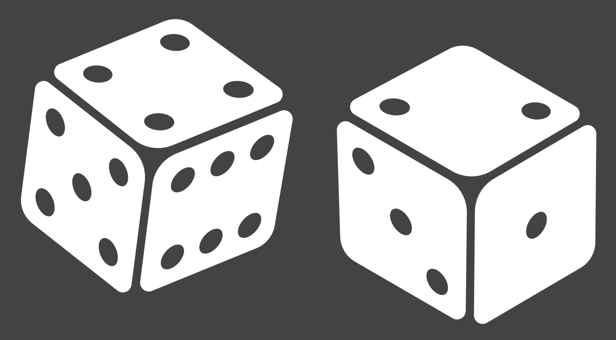
m sets



$\mathcal{U}$

n elements

v_1
v_4
v_5
v_6
v_2
v_3



What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	???

What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	???

$\Omega(\log m)$ even for fractional algorithms in RO! [BuchbinderNaor09]

strategy $\Omega(\log n \log m)$...

What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	???

$\Omega(\log m)$ even for fractional algorithms in RO! [BuchbinderNaor09]

strategy $\Omega(\log n \log m)$...

Believable

$o(\log n \log m)$ not possible...

What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	???

Theorem [Gupta Kehne **L.** 21]:

*There is a **randomized poly time** algorithm for RO **Online Set** Cover with competitive ratio $O(\log mn)$.*

What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	O(log mn) Our work

Theorem [Gupta Kehne **L.** 21]:

*There is a **randomized poly time** algorithm for RO **Online Set** Cover with competitive ratio* $O(\log mn)$.

What is known?

Offline	log n + 1 [Johnson74],[Lovasz75], [Chvatal79]
Adversarial Online	O(log n log m) [Alon+03] [BuchbinderNaor09]
Stochastic Online	O(log mn) [Gupta Grandoni Leonardi Miettinen Sankowski Singh 08]
RO	O(log mn) Our work

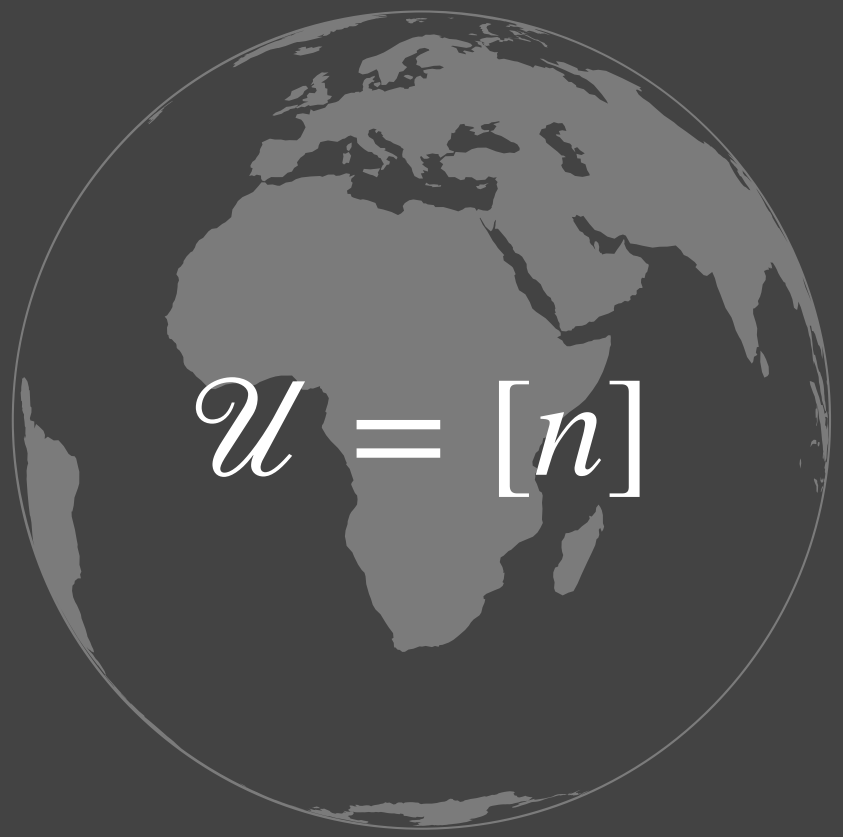
Theorem [Gupta Kehne **L.** 21]:

*There is a **randomized poly time** algorithm for RO **Online Set** Cover with competitive ratio* $O(\log mn)$.

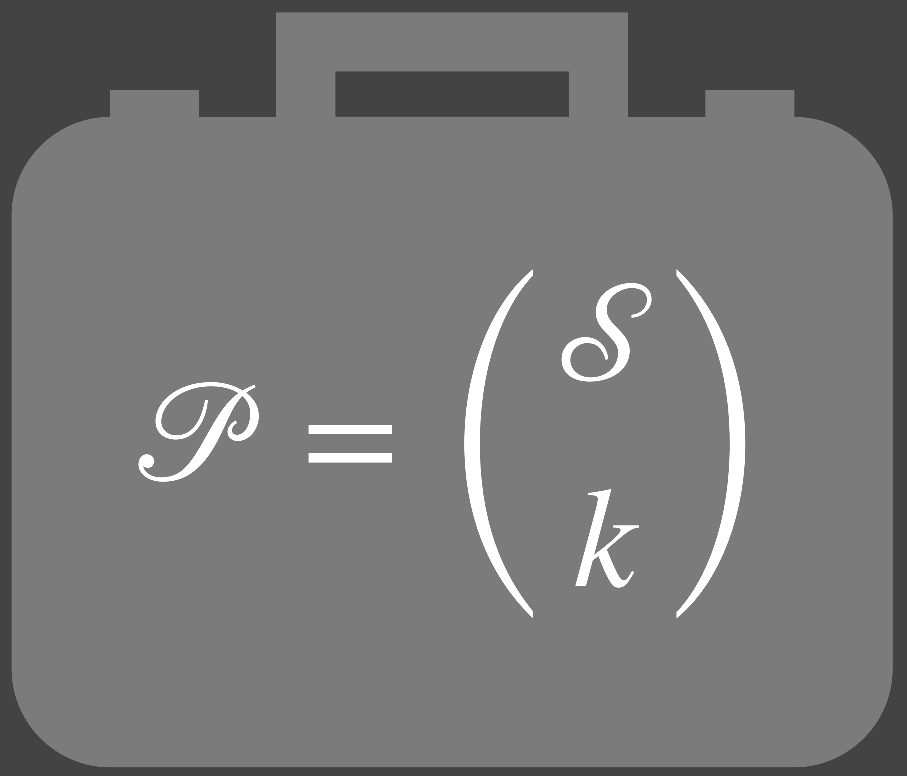
New algorithm! We show how to learn distribution & solve at same time.

RO Set Cover
(Exponential Time Warmup)

RO Set Cover
(Exponential Time Warmup)



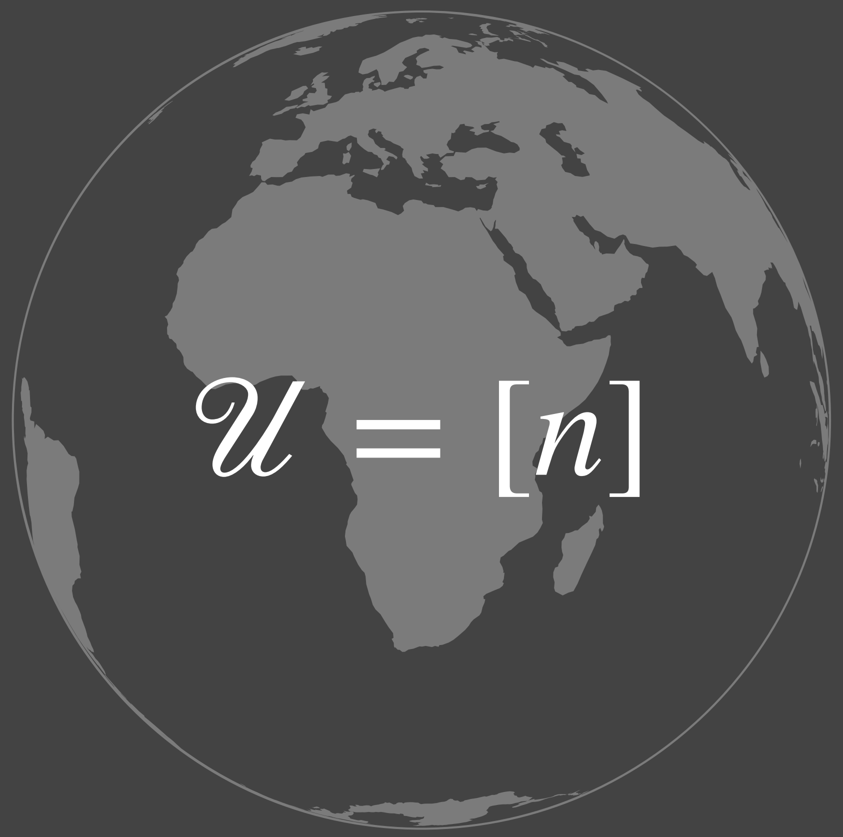
$$\mathscr{U} = [n]$$



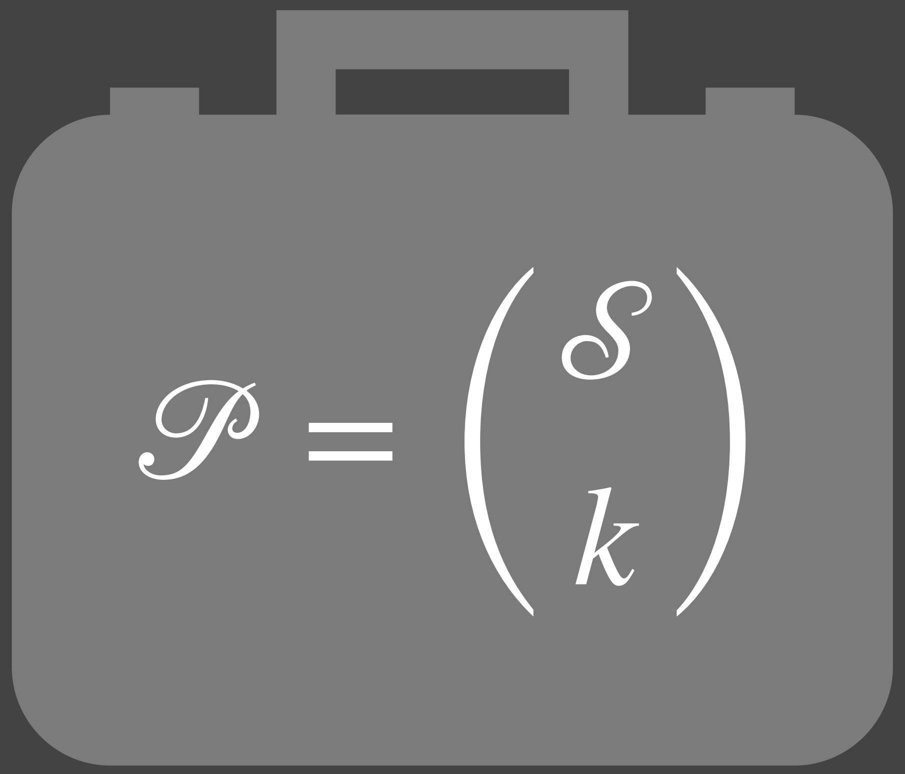
$$\mathscr{P} = \binom{\mathscr{S}}{k}$$

$k \approx |OPT|$

RO Set Cover
(Exponential Time Warmup)



$\mathscr{U} = [n]$

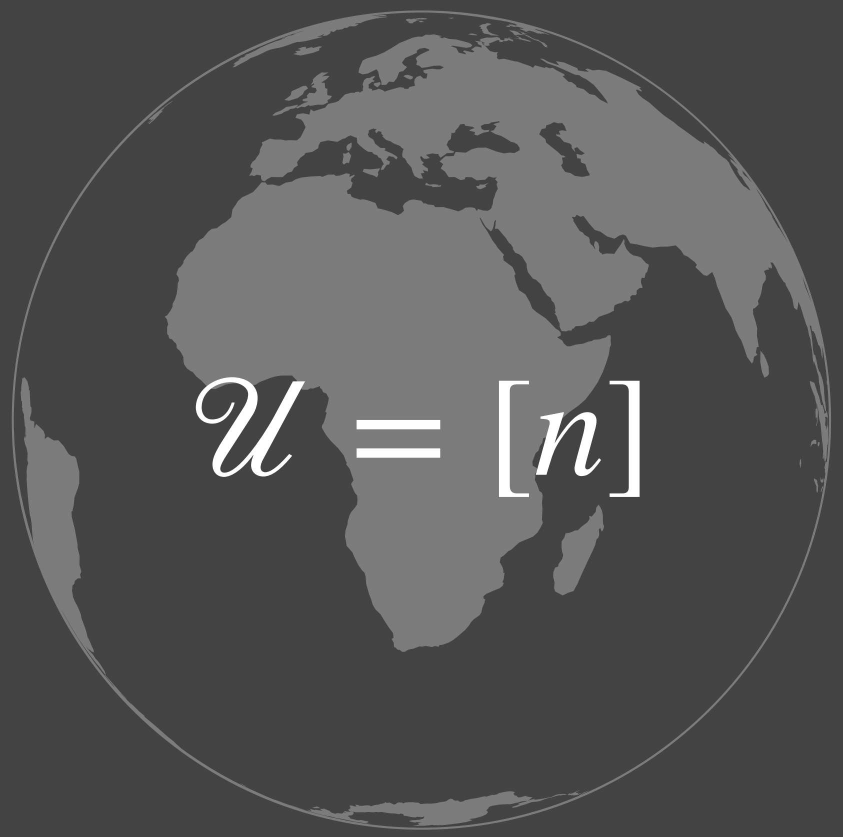


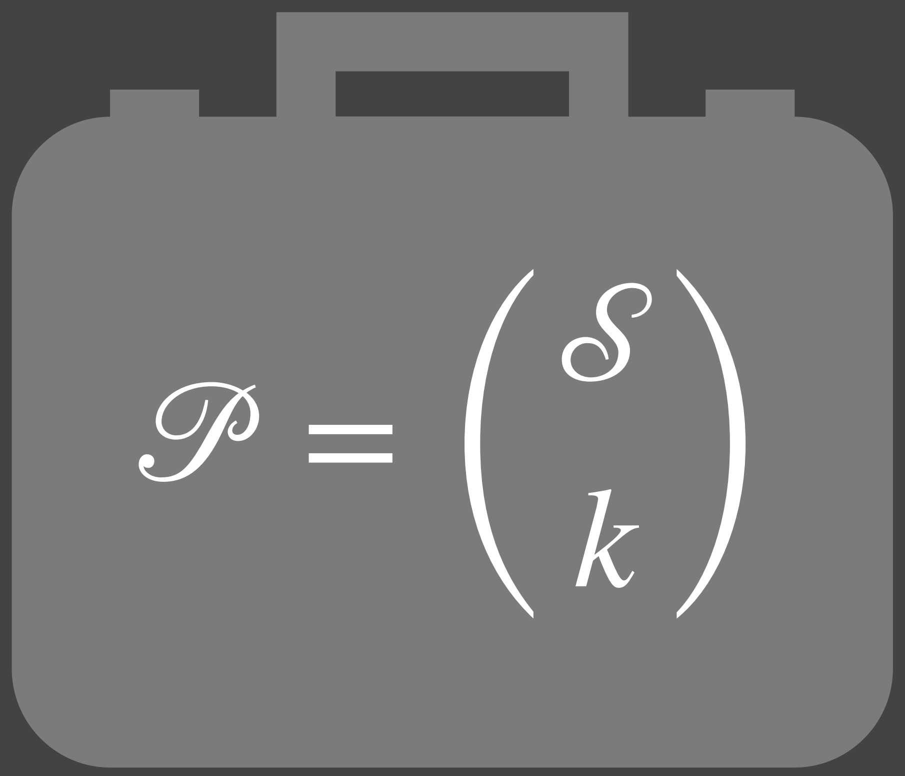
$\mathscr{P} = \binom{\mathscr{S}}{k}$

$k \approx |OPT|$

@ time t:

RO Set Cover
(Exponential Time Warmup)



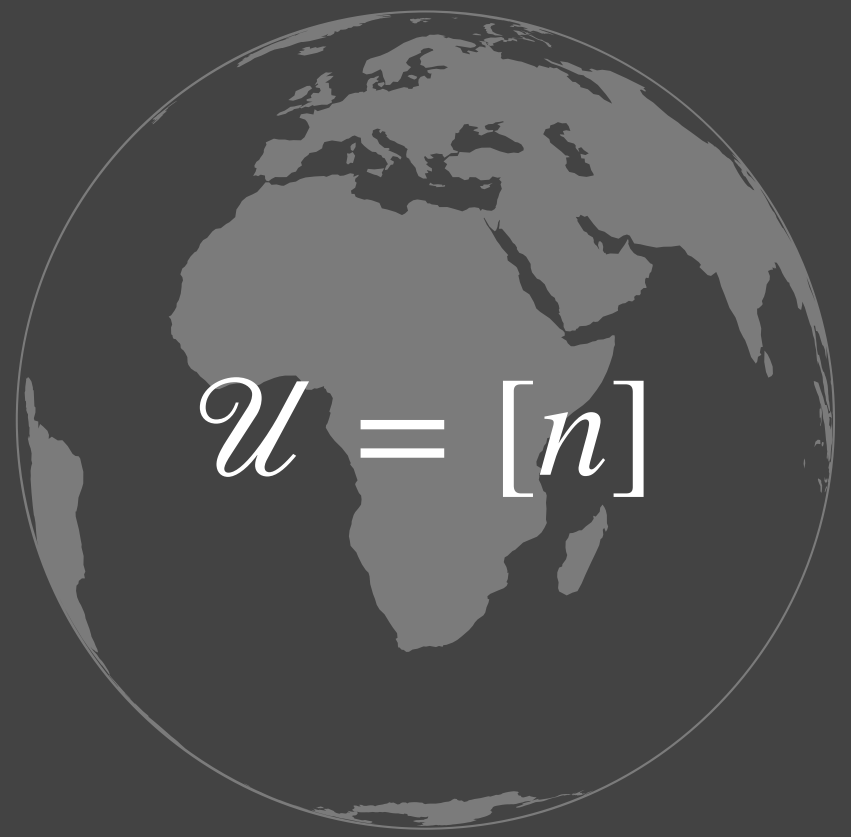


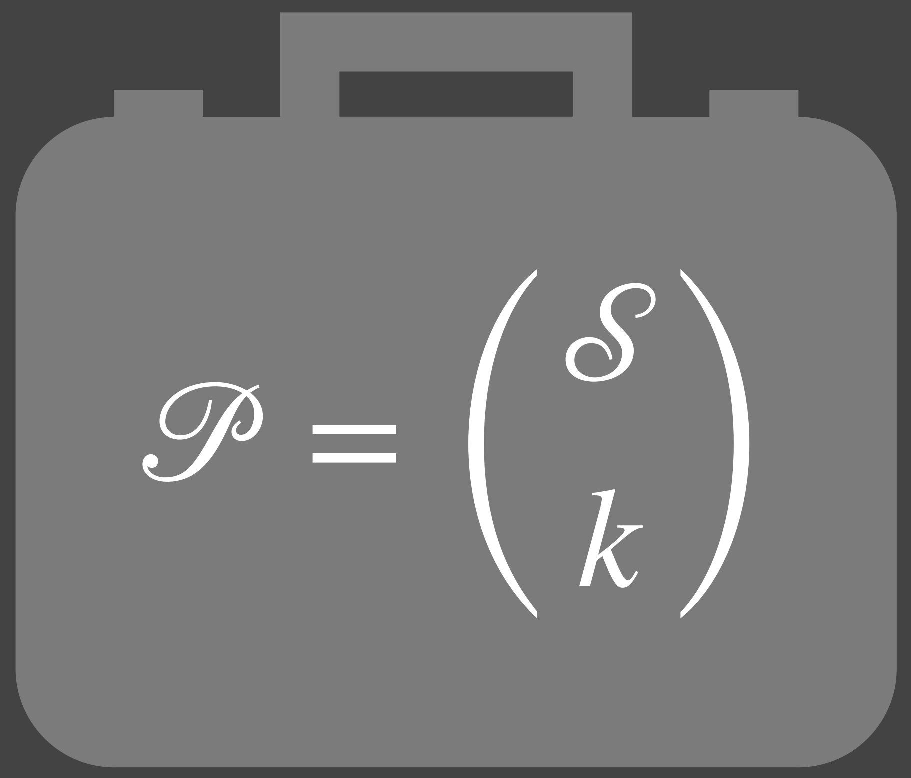
$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

RO Set Cover
(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

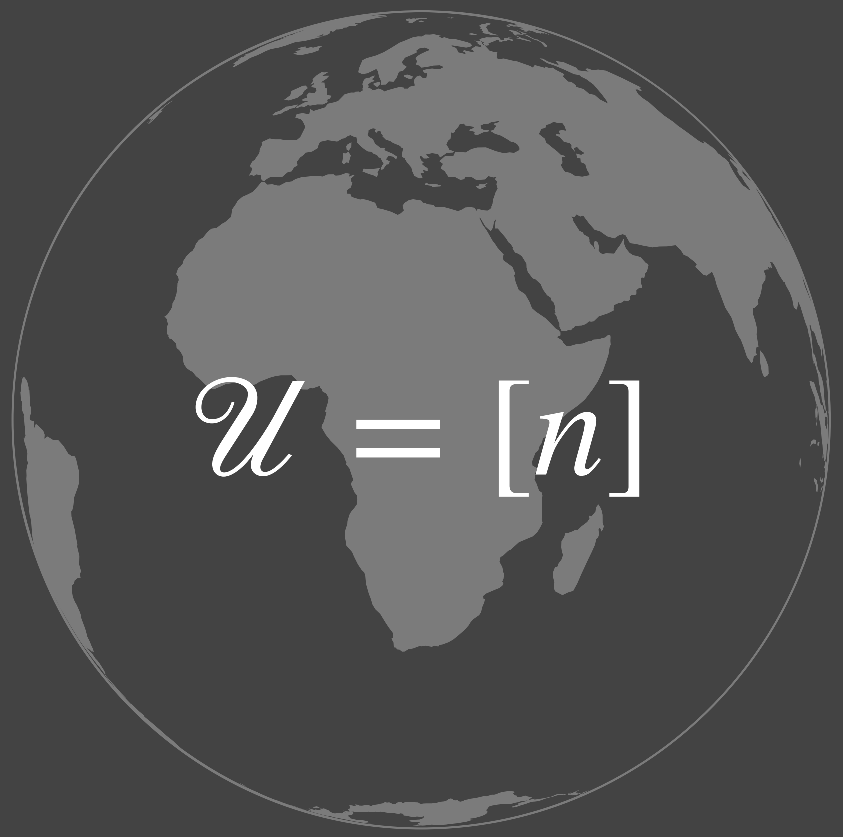
If v^t covered, do nothing.

Else:

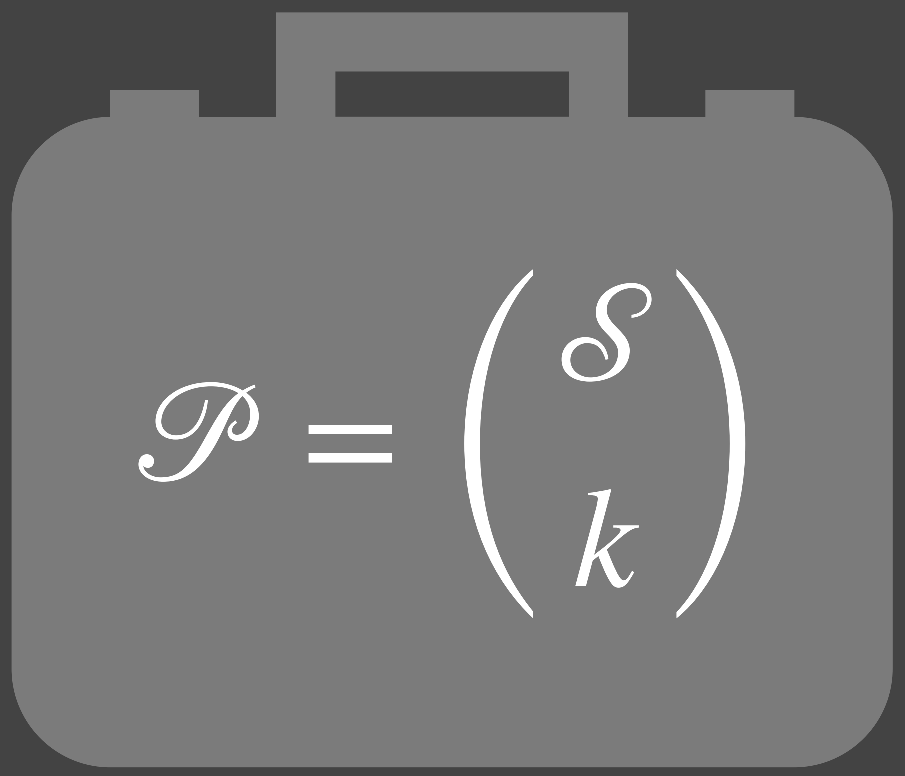
(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

RO Set Cover
(Exponential Time Warmup)



$\mathcal{U} = [n]$



$\mathcal{P} = \binom{\mathcal{S}}{k}$

$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

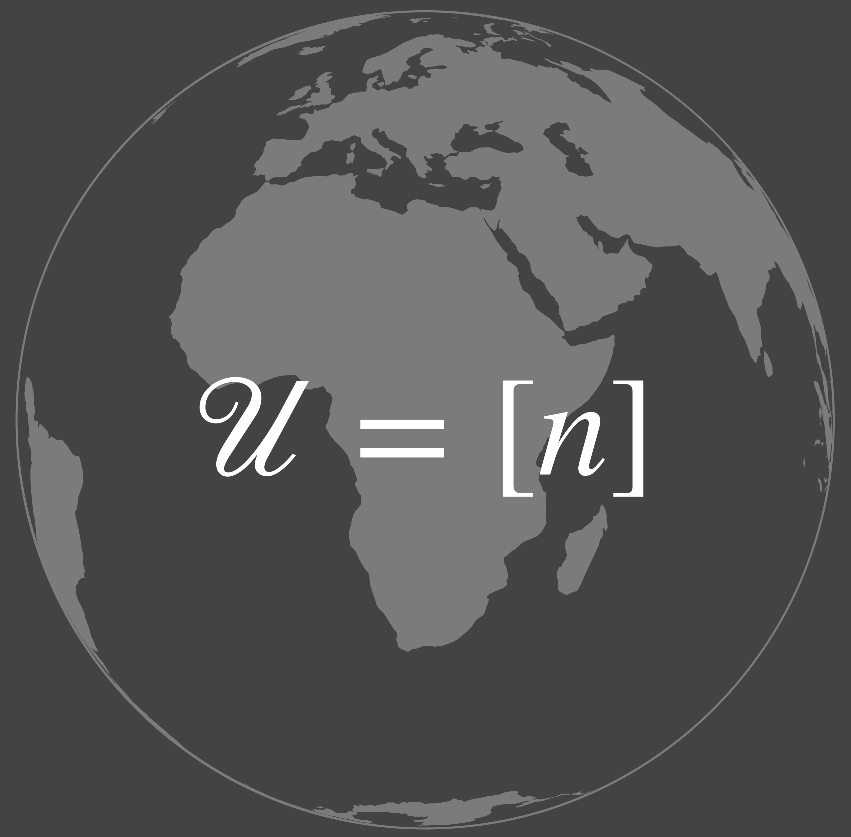
Else:

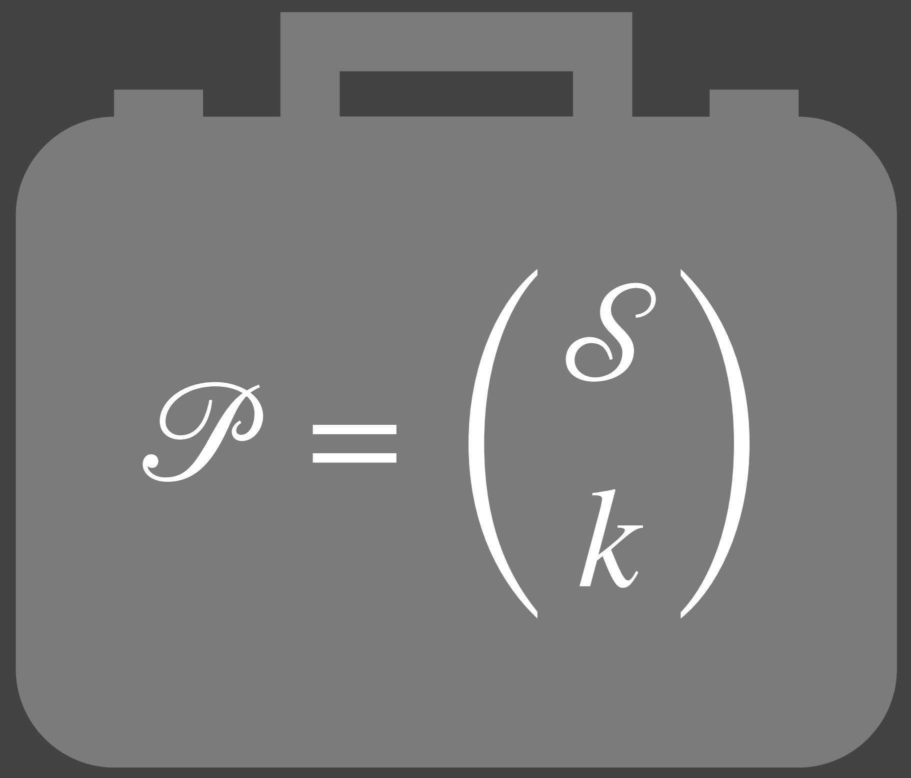
(I) choose $T \sim \mathcal{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathcal{P}$.

Buy arbitrary set to cover v^t.

RO Set Cover
(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

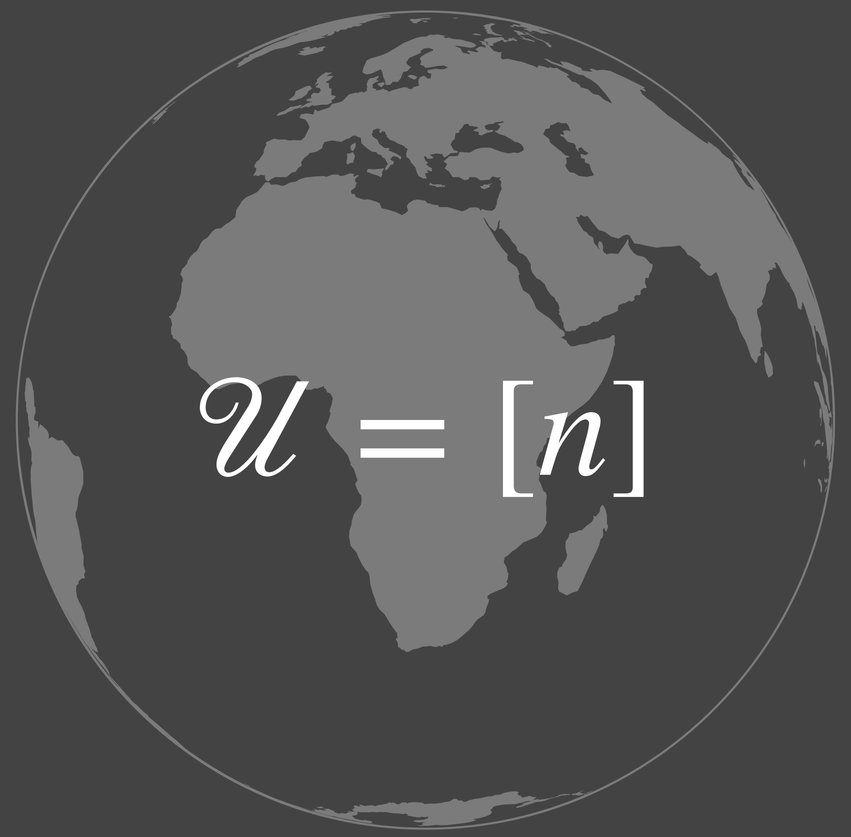
(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ of $T \in \mathscr{P}$ cover $\geq 1/2$ of $\mathscr{U}$.

Case 2: $\geq 1/2$ of $T \in \mathscr{P}$ cover $< 1/2$ of $\mathscr{U}$.

RO Set Cover
(Exponential Time Warmup)



$\mathscr{U} = [n]$

$\mathscr{P} = \binom{\mathscr{S}}{k}$

$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

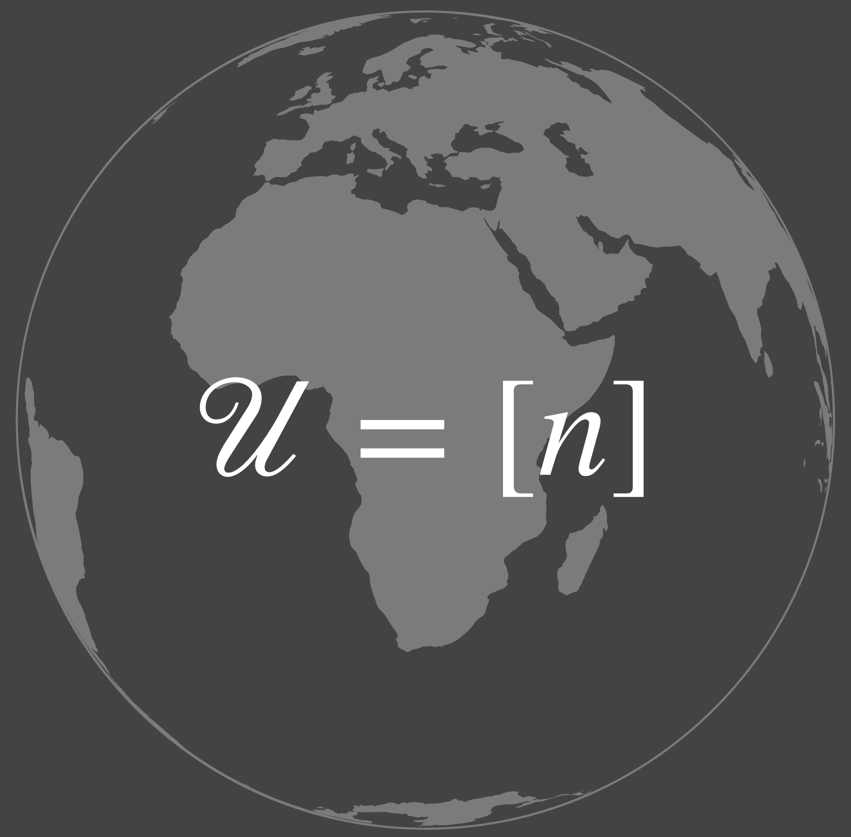
Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

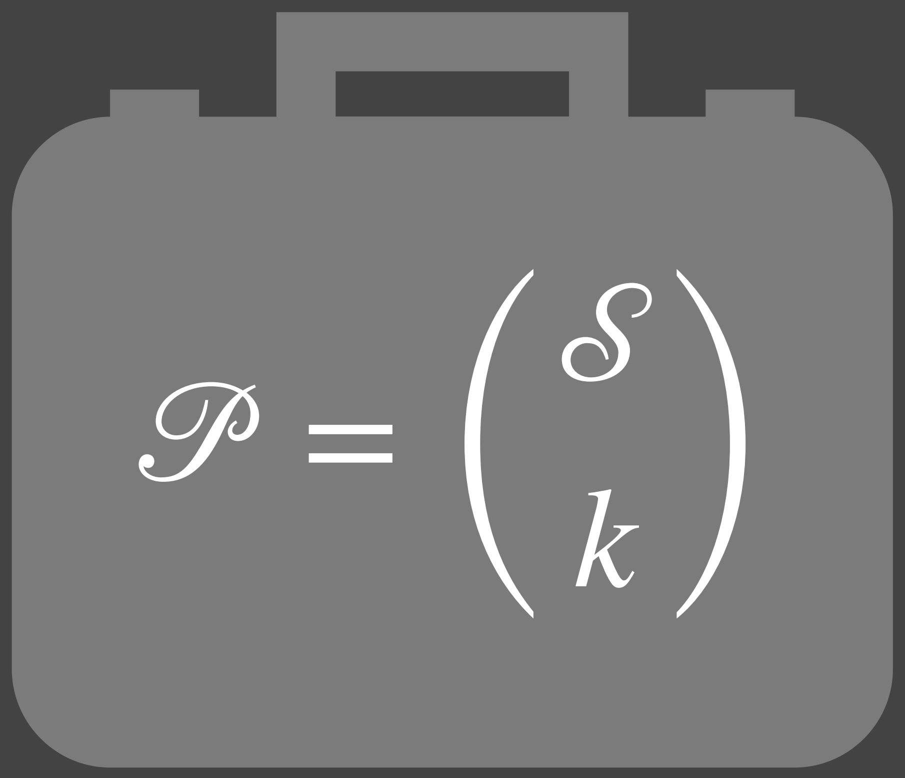
$\Rightarrow R$ covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

RO Set Cover

(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

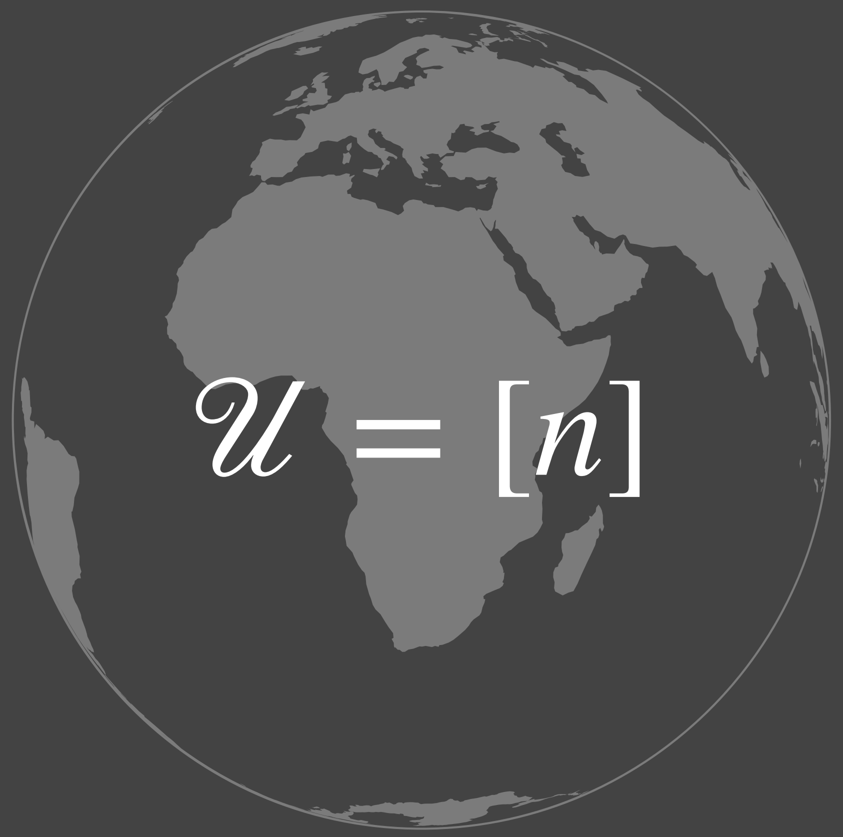
Case 1: $\geq 1/2$ of $T \in \mathscr{P}$ cover $\geq 1/2$ of $\mathscr{U}$.

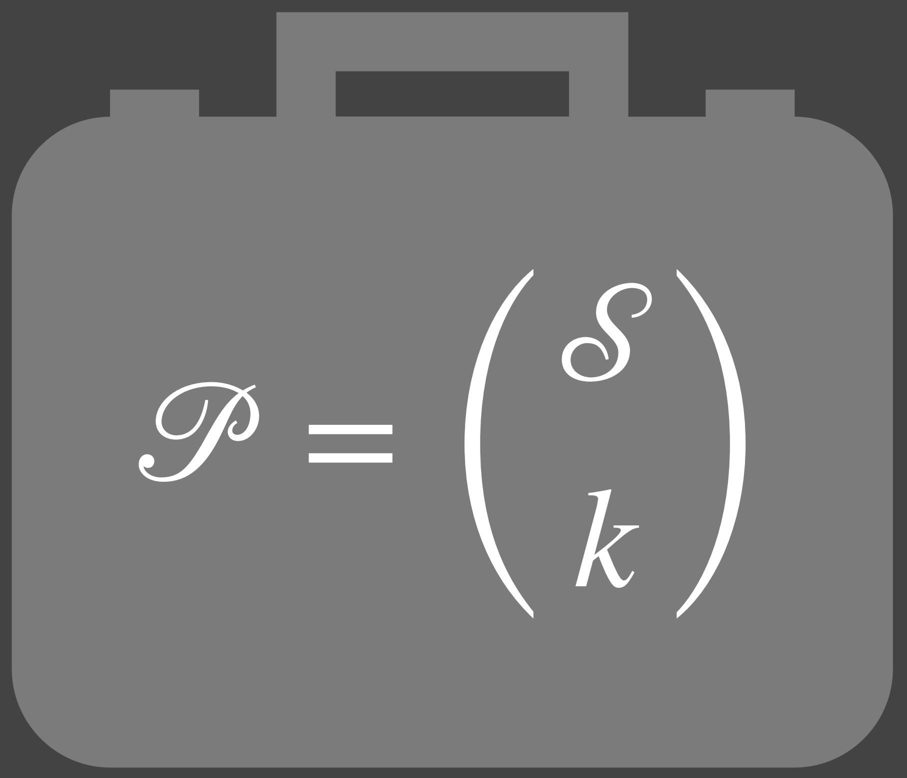
$\Rightarrow$ R covers $\frac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \frac{1}{4k}\right)$ in expectation.

Case 2: $\geq 1/2$ of $T \in \mathscr{P}$ cover $< 1/2$ of $\mathscr{U}$.

RO Set Cover
(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

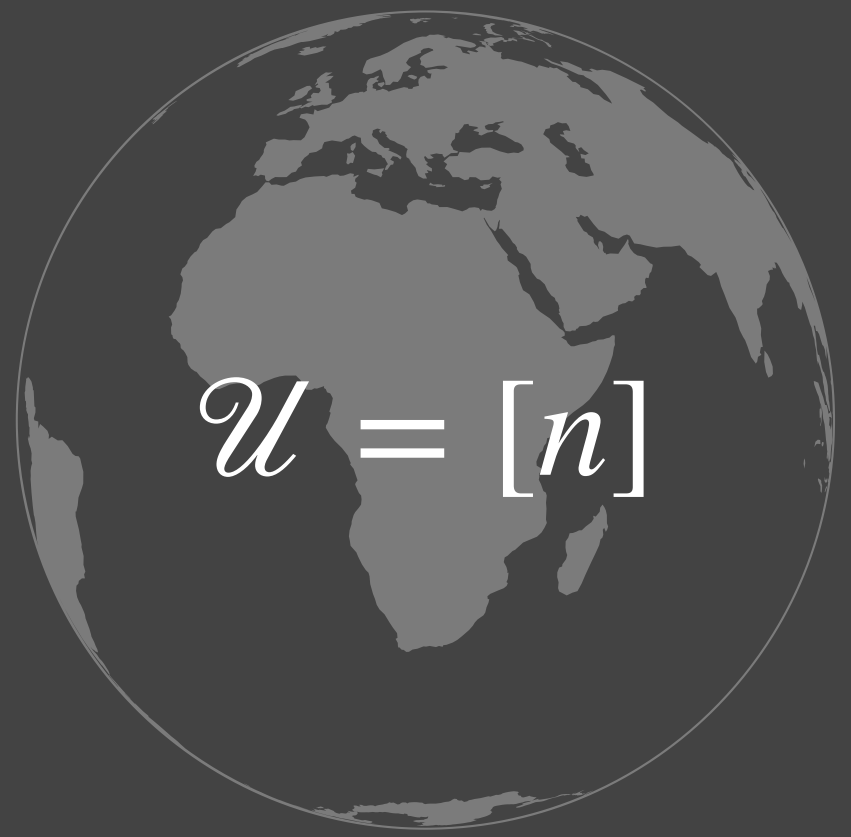
$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

RO Set Cover
(Exponential Time Warmup)



$$\mathscr{P} = \binom{\mathscr{S}}{k}$$

$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

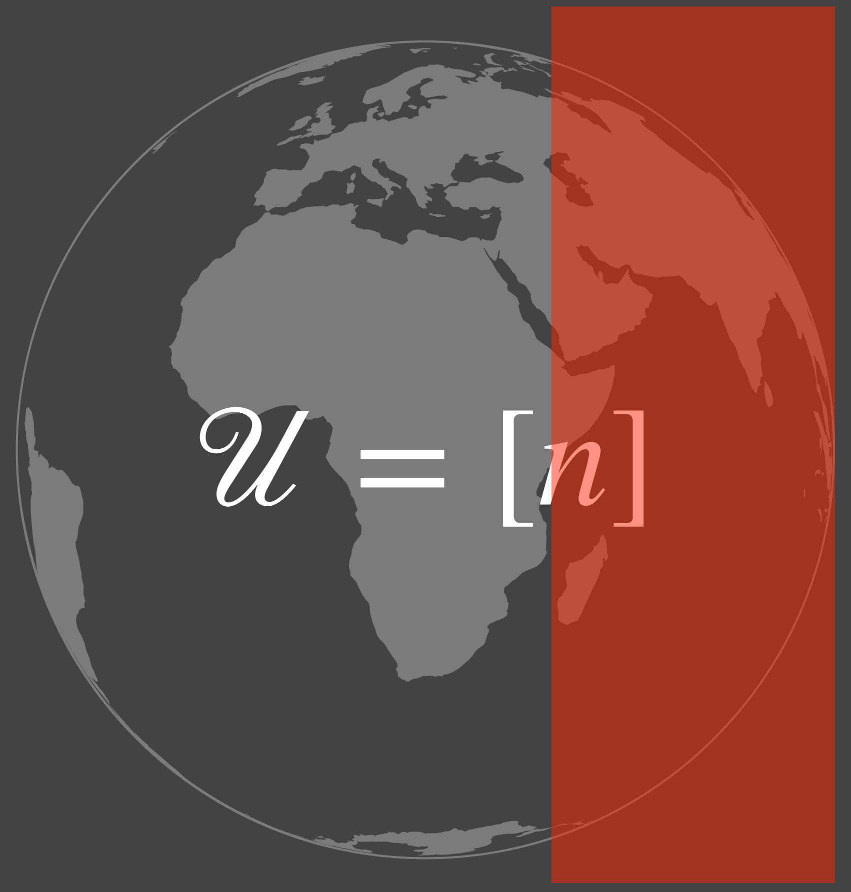
$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

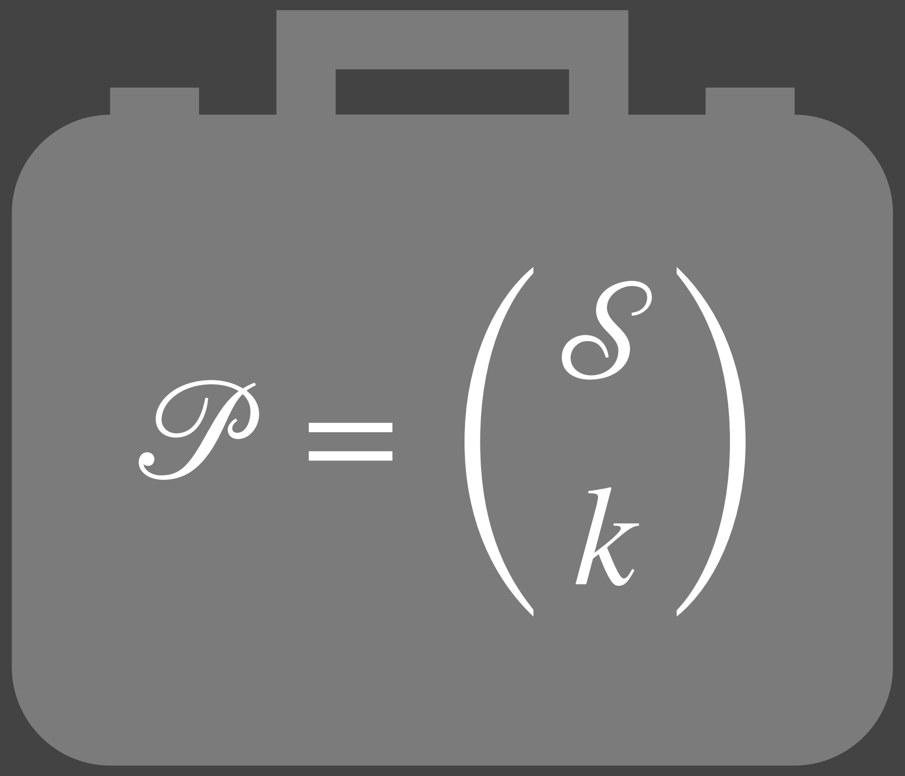
Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover
(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ of $T \in \mathscr{P}$ cover $\geq 1/2$ of $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

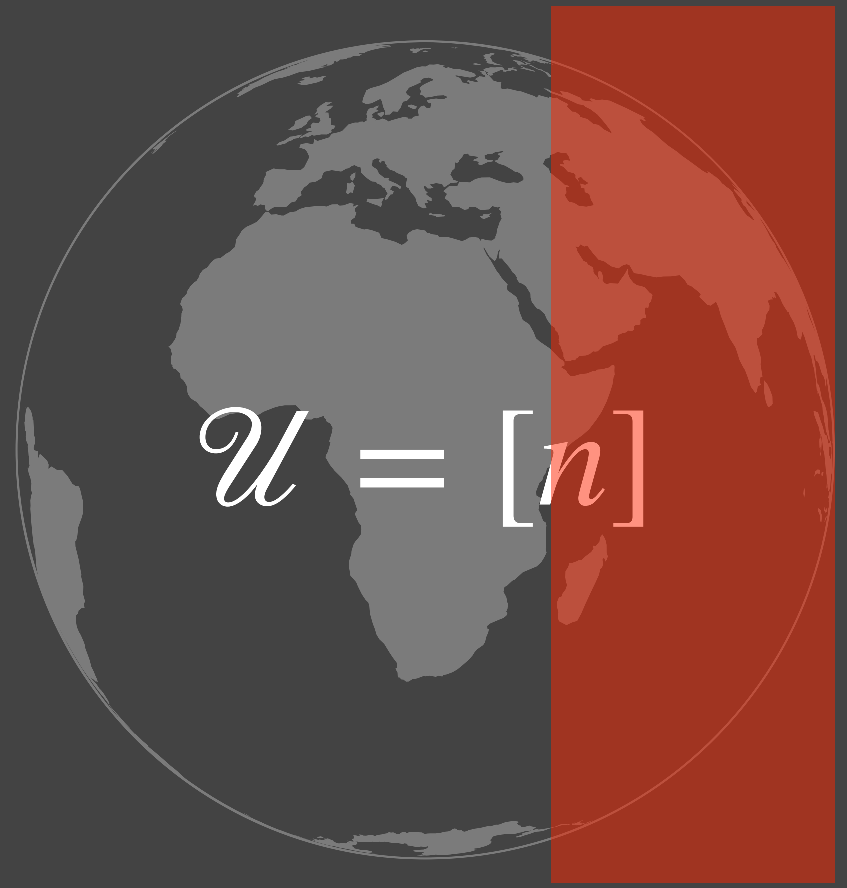
Case 2: $\geq 1/2$ of $T \in \mathscr{P}$ cover $< 1/2$ of $\mathscr{U}$.

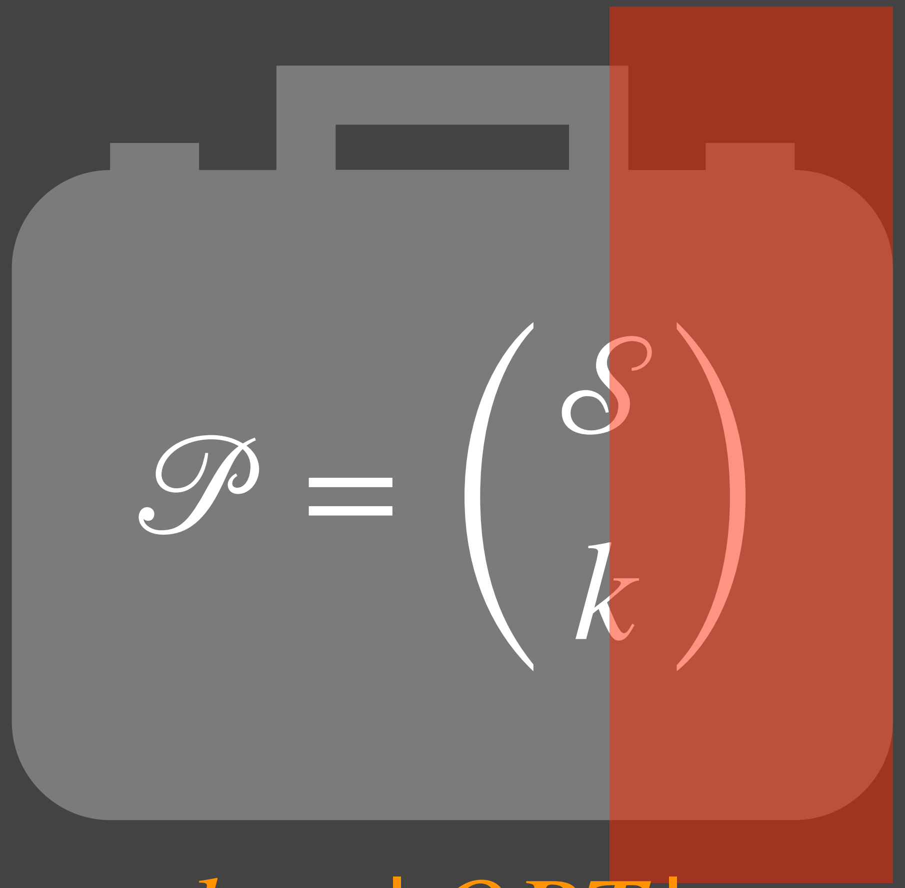
$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover

(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

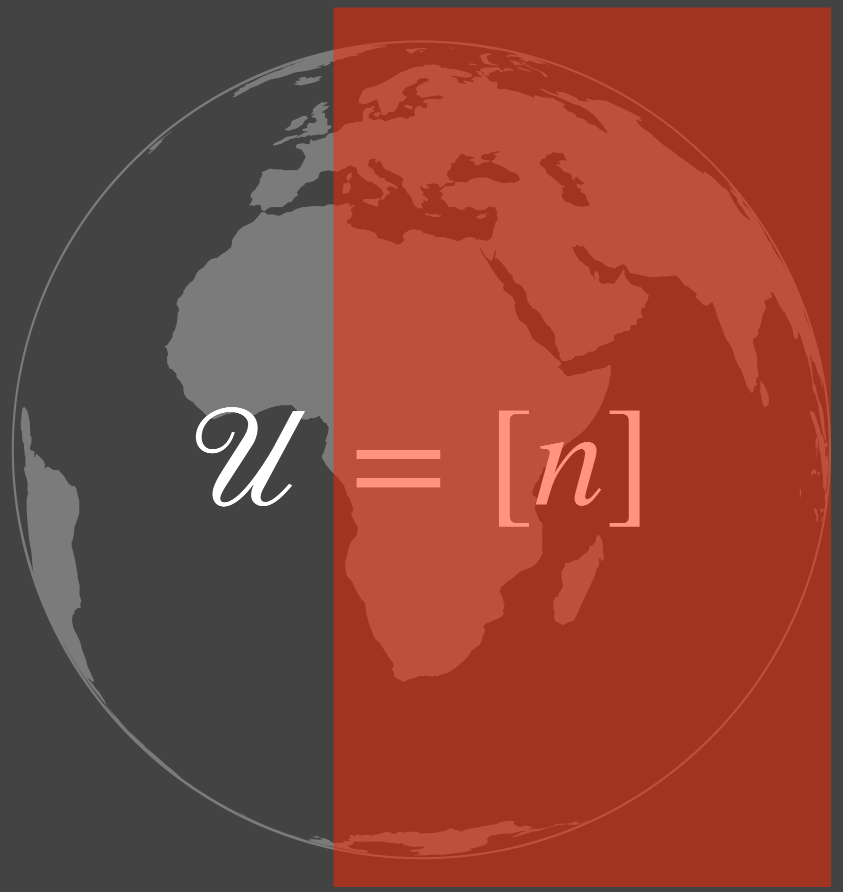
$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

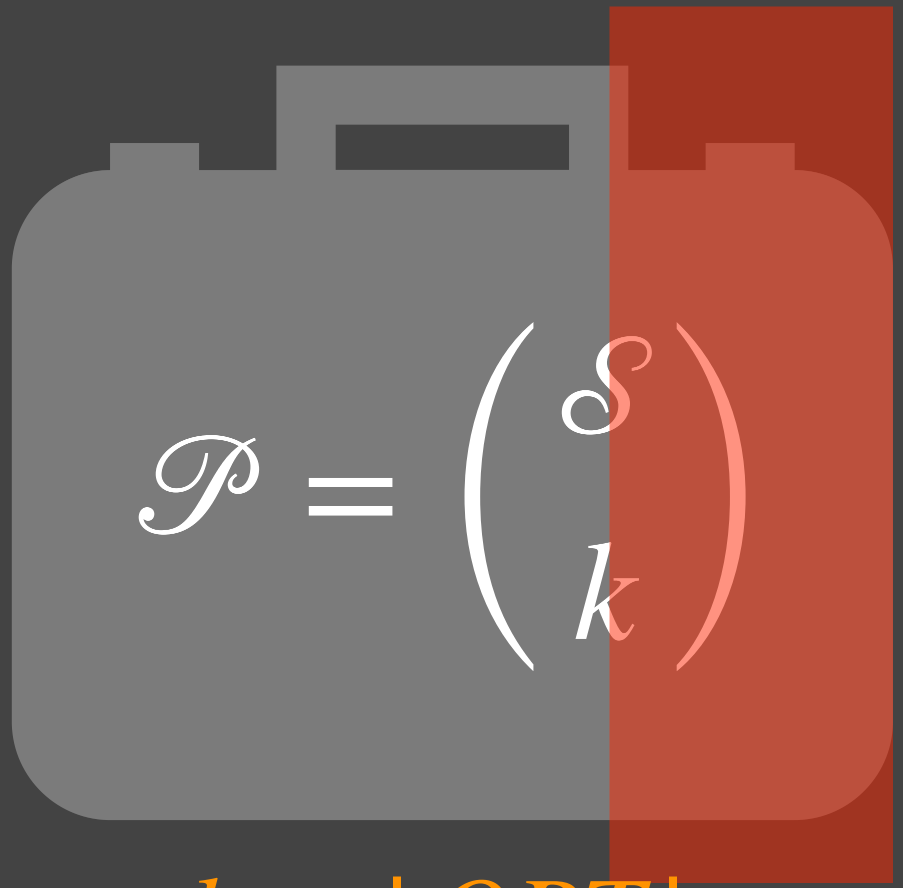
Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover
(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

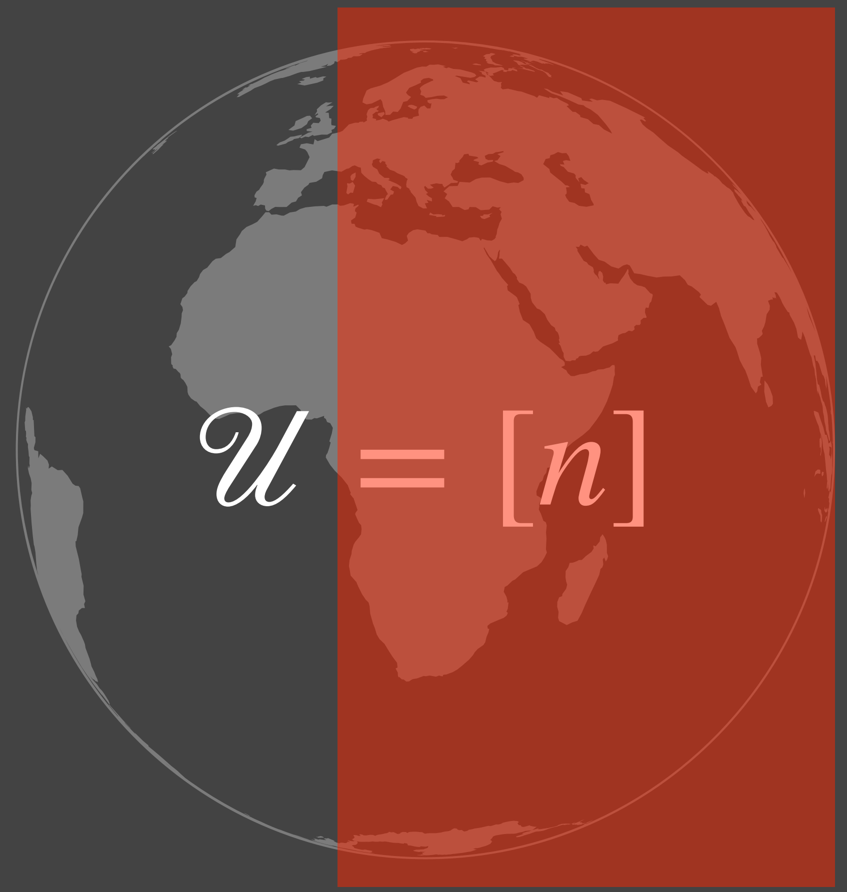
Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

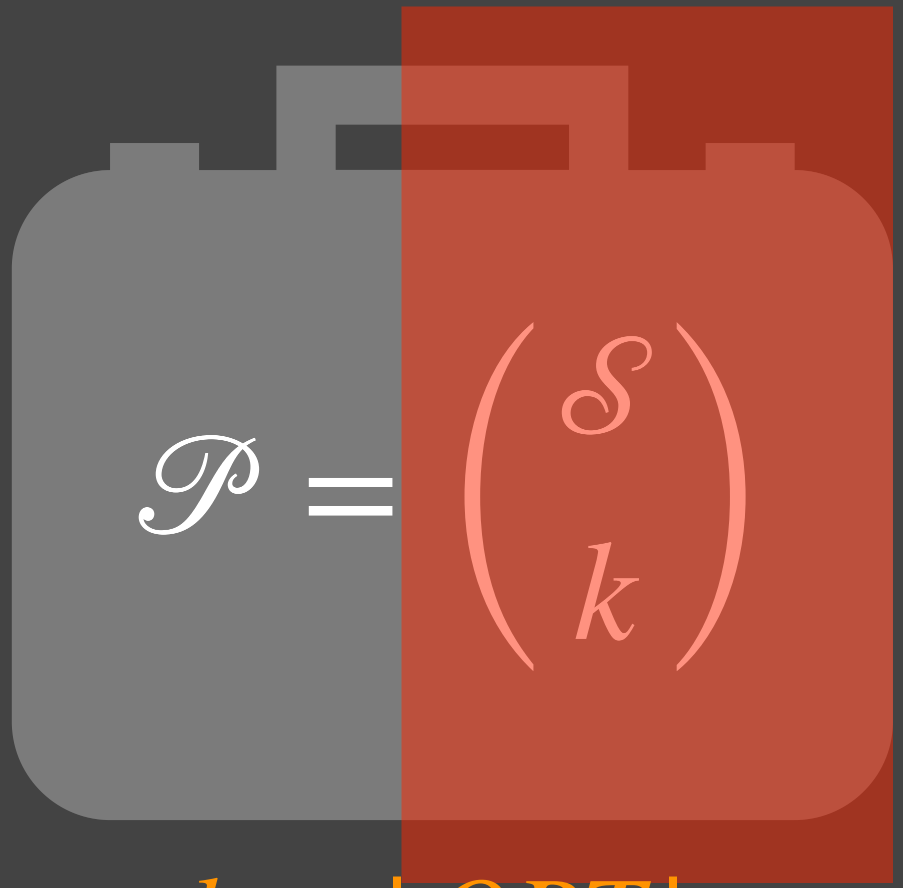
$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover

(Exponential Time Warmup)





$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ of $T \in \mathscr{P}$ cover $\geq 1/2$ of $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

Case 2: $\geq 1/2$ of $T \in \mathscr{P}$ cover $< 1/2$ of $\mathscr{U}$.

$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover

(Exponential Time Warmup)

$k \approx |OPT|$

@ time t:

If v^t covered, do nothing.

Else:

(I) choose $T \sim \mathscr{P}$, buy random $R \sim T$.

(II) "Prune" $T \not\ni v^t$ from $\mathscr{P}$.

Buy arbitrary set to cover v^t.

Case 1: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $\geq 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ R covers $\dfrac{|\mathscr{U}|}{4k}$ in expectation.

$\Rightarrow$ $\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

Case 2: $\geq 1/2$ **of** $T \in \mathscr{P}$ **cover** $< 1/2$ **of** $\mathscr{U}$.

$\Rightarrow$ $\geq 1/2$ of $T \in \mathscr{P}$ pruned w.p. $1/2$.

$\Rightarrow$ $\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover
(Exponential Time Warmup)

Case 1: (COVER)

$\mathscr{U}$ shrinks by $\left(1 - \frac{1}{4k}\right)$ in expectation.

Case 2: (LEARN)

$\mathscr{P}$ shrinks by $3/4$ in expectation.

RO Set Cover
(Exponential Time Warmup)

Case 1: (COVER)

$\mathscr{U}$ shrinks by $\left(1 - \frac{1}{4k}\right)$ in expectation.

Case 2: (LEARN)

$\mathscr{P}$ shrinks by $3/4$ in expectation.

$|\mathscr{U}|$ initially n, $\Rightarrow$ $O(k \log n)$ COVER steps suffice.

RO Set Cover
(Exponential Time Warmup)

Case 1: (COVER)

$\mathscr{U}$ shrinks by $\left(1 - \dfrac{1}{4k}\right)$ in expectation.

Case 2: (LEARN)

$\mathscr{P}$ shrinks by $3/4$ in expectation.

$|\mathscr{U}|$ initially n, $\Rightarrow$ $O(k \log n)$ COVER steps suffice.

$|\mathscr{P}|$ initially $\binom{m}{k}$, $\Rightarrow$ $O(k \log m)$ LEARN steps suffice.

RO Set Cover
(Exponential Time Warmup)

Case 1: (COVER)

$\mathscr{U}$ shrinks by $\left(1 - \frac{1}{4k}\right)$ in expectation.

Case 2: (LEARN)

$\mathscr{P}$ shrinks by $3/4$ in expectation.

$|\mathscr{U}|$ initially n, $\Rightarrow$ $O(k \log n)$ COVER steps suffice.

$|\mathscr{P}|$ initially $\binom{m}{k}$, $\Rightarrow$ $O(k \log m)$ LEARN steps suffice.

$\Rightarrow O(k \log mn)$ steps suffice.

RO Set Cover
(Exponential Time Warmup)

Case 1: (COVER)

$\mathscr{U}$ shrinks by $\left(1 - \frac{1}{4k}\right)$ in expectation.

Case 2: (LEARN)

$\mathscr{P}$ shrinks by $3/4$ in expectation.

$|\mathscr{U}|$ initially n, $\Rightarrow$ $O(k \log n)$ COVER steps suffice.

$|\mathscr{P}|$ initially $\binom{m}{k}$, $\Rightarrow$ $O(k \log m)$ LEARN steps suffice.

$\Rightarrow O(k \log mn)$ steps suffice.

But how to make polytime?

Can we reuse LEARN/ COVER intuition?

LearnOrCover
(Unit cost)

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/\|x\|_1$.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/||x||_1$.

Buy arbitrary set to cover v.

LearnOrCover
(Unit cost)

Idea! Measure convergence with potential function:

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/\|x\|_1$.

Buy arbitrary set to cover v.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/\|x\|_1$.

Buy arbitrary set to cover v.

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1 \, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/||x||_1$.

Buy arbitrary set to cover v.

$\sum_S x^*_S \log \frac{x^*_S}{x^t_S}$

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1 \, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/||x||_1$.

Buy arbitrary set to cover v.

$$\sum_S x_S^* \log \frac{x_S^*}{x_S^t}$$

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1 \, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.
@ time t, element v arrives:
  If v covered, do nothing.
  Else:
    (I) Buy random $R \sim x$.
    (II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.
      Renormalize $x \leftarrow x/\|x\|_1$.
  Buy arbitrary set to cover v.

$$\sum_S x_S^* \log \frac{x_S^*}{x_S^t}$$

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1\, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v uncovered, then $E[\Delta\Phi] \leq -\frac{1}{k}$.

(Recall $k = |OPT|$)

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.
@ time t, element v arrives:
  If v covered, do nothing.
  Else:
    (I) Buy random $R \sim x$.
    (II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.
      Renormalize $x \leftarrow x/\|x\|_1$.
  Buy arbitrary set to cover v.

$\sum_S x_S^* \log \frac{x_S^*}{x_S^t}$

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1\, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v uncovered, then $E[\Delta\Phi] \leq -\frac{1}{k}$.

(Recall $k = |OPT|$)

Bound $E_R[\Delta \log |\mathcal{U}^t|]$ over randomness of R.

Bound $E_v[\Delta KL]$ over randomness of v.

LearnOrCover
(Unit cost)

Init. $x \leftarrow 1/m$.

@ time t, element v arrives:

If v covered, do nothing.

Else:

(I) Buy random $R \sim x$.

(II) $\forall S \ni v$, set $x_S \leftarrow e \cdot x_S$.

Renormalize $x \leftarrow x/\|x\|_1$.

Buy arbitrary set to cover v.

$\sum_S x_S^* \log \frac{x_S^*}{x_S^t}$

Idea! Measure convergence with potential function:

$$\boxed{\Phi(t) = c_1 \, \mathrm{KL}(x^* \,||\, x^t) + c_2 \log |\mathcal{U}^t|}$$

$\mathcal{U}^t$:= uncovered elements @ time t

x^* := uniform distribution on OPT

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v uncovered, then $E[\Delta\Phi] \leq -\frac{1}{k}$.

(Recall $k = |OPT|$)

Bound $E_R[\Delta \log |\mathcal{U}^t|]$ over randomness of R.

Bound $E_v[\Delta KL]$ over randomness of v. $\longleftarrow$ This is where we use RO!

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathscr{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathscr{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

$$E[\Delta\Phi] = E_v[\Delta \mathrm{KL}] + (e-1) \cdot E_R[\Delta \log |\mathscr{U}^t|] \leq -\frac{1}{k}$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

$$E[\Delta\Phi] = E_v[\Delta \mathrm{KL}] + (e-1) \cdot E_R[\Delta \log |\mathcal{U}^t|] \leq -\frac{1}{k}$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

$$E[\Delta\Phi] = E_v[\Delta \mathrm{KL}] + (e-1) \cdot E_R[\Delta \log |\mathcal{U}^t|] \leq -\frac{1}{k}$$

Since $\Phi(0) = O(\log(mn))$, expected total cost is $k \log(mn)$.

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathscr{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\mathbf{KL}(x^* \,||\, x^t) - \mathbf{KL}(x^* \,||\, x^{t-1})$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \le -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \sum_S x_S^* \log \|x\|_1 - \sum_{S \ni v} x_S^* \log e$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \log e$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathscr{U}^t|] \le -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$\color{red}{E_v[\Delta \mathrm{KL}]} \leq (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \underbrace{\cancel{\log e}}_{=1}$$

Claim 2b: If v^t uncovered,

$$\color{blue}{E_R[\Delta \log|\mathcal{U}^t|]} \leq -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \underbrace{\cancel{\log e}}_{=1}$$

$$= \log\left(\sum_S x_S^{t-1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \sum_{S\ni v} x_S^*$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathscr{U}^t|] \le -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \sum_{S\ni v} x_S^*$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathcal{U}^t|] \leq -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\right) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \text{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \underbrace{\cancel{\log e}}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\right) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \underbrace{\cancel{\log e}}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \underbrace{\sum_{S\ni v} x_S^*}_{\ge 1/k}$$

$$\le \log\left(1 + \sum_{S\ni v}(e-1)\cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z)\le z$, take expectation over v, ■.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathcal{U}^t|] \le -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\Bigg(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\Bigg) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \leq z$, take expectation over v, ■.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

$$\log|\mathcal{U}^t| - \log|\mathcal{U}^{t-1}|$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\Bigg(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\Bigg) - \underbrace{\sum_{S \ni v} x_S^*}_{\ge 1/k}$$

$$\le \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \le z$, take expectation over v, ■.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \le -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

$$\log|\mathcal{U}^t| - \log|\mathcal{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathcal{U}^{t-1}| - |\mathcal{U}^t|}{|\mathcal{U}^{t-1}|}\right)$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta\mathrm{KL}] \leq (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \underbrace{\sum_{S\ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S\ni v}(e-1)\cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \leq z$, take expectation over v, ■.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathscr{U}^t|] \leq -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

$$\log|\mathscr{U}^t| - \log|\mathscr{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathscr{U}^{t-1}| - |\mathscr{U}^t|}{|\mathscr{U}^{t-1}|}\right)$$

Use $\log(1-z) \leq -z$.

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \underbrace{\sum_{S\ni v} x_S^*}_{\ge 1/k}$$

$$\le \log\left(1 + \sum_{S\ni v}(e-1)\cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \le z$, take expectation over v, $\blacksquare$.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathcal{U}^t|] \le -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

$$\log|\mathcal{U}^t| - \log|\mathcal{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathcal{U}^{t-1}| - |\mathcal{U}^t|}{|\mathcal{U}^{t-1}|}\right)$$

Use $\log(1-z) \le -z$.

$$\le -\frac{1}{|\mathcal{U}^{t-1}|}\sum_{v\in\mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}.$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\underbrace{\sum_S x_S^*}_{=1}} \log \|x\|_1 - \sum_{S \ni v} x_S^* \cancel{\underbrace{\log e}_{=1}}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\right) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \leq z$, take expectation over v, $\blacksquare$.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

$$\log |\mathcal{U}^t| - \log |\mathcal{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathcal{U}^{t-1}| - |\mathcal{U}^t|}{|\mathcal{U}^{t-1}|}\right)$$

Use $\log(1-z) \leq -z$.

$$\leq -\frac{1}{|\mathcal{U}^{t-1}|} \sum_{v \in \mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}.$$

Take expectation over R.

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \le (e-1)\cdot E_v\left[\sum_{S\ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S\ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S\ni v}(e-1)\cdot x_S^{t-1}\right) - \underbrace{\sum_{S\ni v} x_S^*}_{\ge 1/k}$$

$$\le \log\left(1 + \sum_{S\ni v}(e-1)\cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \le z$, take expectation over v, $\blacksquare$.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log|\mathscr{U}^t|] \le -E_v\left[\sum_{S\ni v} x_S\right].$$

Proof:

$$\log|\mathscr{U}^t| - \log|\mathscr{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathscr{U}^{t-1}| - |\mathscr{U}^t|}{|\mathscr{U}^{t-1}|}\right)$$

Use $\log(1-z) \le -z$.

$$\le -\frac{1}{|\mathscr{U}^{t-1}|}\sum_{v\in\mathscr{U}^{t-1}} \mathbb{1}\{R\ni v\}.$$

Take expectation over R.

$$E_R[\Delta \log|\mathscr{U}^t|] \le -\frac{1}{|\mathscr{U}^{t-1}|}\sum_R x_R \sum_{v\in\mathscr{U}^{t-1}} \mathbb{1}\{R\ni v\}$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \cancel{\sum_S x_S^*}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \cancel{\log e}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\right) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \leq z$, take expectation over v, $\blacksquare$.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

$$\log |\mathcal{U}^t| - \log |\mathcal{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathcal{U}^{t-1}| - |\mathcal{U}^t|}{|\mathcal{U}^{t-1}|}\right)$$

Use $\log(1-z) \leq -z$.

$$\leq -\frac{1}{|\mathcal{U}^{t-1}|} \sum_{v \in \mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}.$$

Take expectation over R.

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -\frac{1}{|\mathcal{U}^{t-1}|} \sum_R x_R \sum_{v \in \mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}$$

$$= -\frac{1}{|\mathcal{U}^{t-1}|} \sum_{v \in \mathcal{U}^{t-1}} \sum_{R \ni v} x_R.$$

Claim 2a: If v^t uncovered,

$$E_v[\Delta \mathrm{KL}] \leq (e-1) \cdot E_v\left[\sum_{S \ni v} x_S\right] - \frac{1}{k}.$$

Proof:

$$\sum_S x_S^* \log\left(\frac{x_S^*}{x_S^t}\right) - \sum_S x_S^* \log\left(\frac{x_S^*}{x_S^{t-1}}\right)$$

$$= \sum_S x_S^* \log\left(\frac{x_S^{t-1}}{x_S^t}\right)$$

$$= \underbrace{\cancel{\sum_S x_S^*}}_{=1} \log \|x\|_1 - \sum_{S \ni v} x_S^* \underbrace{\cancel{\log e}}_{=1}$$

$$= \log\left(\underbrace{\sum_S x_S^{t-1}}_{=1} + \sum_{S \ni v} (e-1) \cdot x_S^{t-1}\right) - \underbrace{\sum_{S \ni v} x_S^*}_{\geq 1/k}$$

$$\leq \log\left(1 + \sum_{S \ni v} (e-1) \cdot x_S\right) - \frac{1}{k}.$$

Use $\log(1+z) \leq z$, take expectation over v, $\blacksquare$.

Claim 2b: If v^t uncovered,

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -E_v\left[\sum_{S \ni v} x_S\right].$$

Proof:

$$\log|\mathcal{U}^t| - \log|\mathcal{U}^{t-1}|$$

$$= \log\left(1 - \frac{|\mathcal{U}^{t-1}| - |\mathcal{U}^t|}{|\mathcal{U}^{t-1}|}\right)$$

Use $\log(1-z) \leq -z$.

$$\leq -\frac{1}{|\mathcal{U}^{t-1}|} \sum_{v \in \mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}.$$

Take expectation over R.

$$E_R[\Delta \log |\mathcal{U}^t|] \leq -\frac{1}{|\mathcal{U}^{t-1}|} \sum_R x_R \sum_{v \in \mathcal{U}^{t-1}} \mathbb{1}\{R \ni v\}$$

$$= -\frac{1}{|\mathcal{U}^{t-1}|} \sum_{v \in \mathcal{U}^{t-1}} \sum_{R \ni v} x_R. \quad \blacksquare$$

Extensions & Lower bounds

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs… Not clear how to handle box constraints.

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs... Not clear how to handle box constraints.

New!

Theorem: $O(\log mn)$ for (non-metric) facility location in random order.

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs... Not clear how to handle box constraints.

New!

Theorem: $O(\log mn)$ for (non-metric) facility location in random order.

We are working on generalizing to Group Steiner Tree!

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs… Not clear how to handle box constraints.

New!

Theorem: $O(\log mn)$ for (non-metric) facility location in random order.

We are working on generalizing to Group Steiner Tree!

Theorem: $\Omega(\log n \log m)$ for “batched” RO set cover.

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs… Not clear how to handle box constraints.

New!

Theorem: $O(\log mn)$ for (non-metric) facility location in random order.

We are working on generalizing to Group Steiner Tree!

Theorem: $\Omega(\log n \log m)$ for "batched" RO set cover.

Corollary: $\Omega(\log m \log f(\mathcal{N}))$ for RO submodular

Extensions & Lower bounds

Theorem: $O(\log mn)$ for pure covering IPs in random order.

Interesting ideas for general costs... Not clear how to handle box constraints.

New!

Theorem: $O(\log mn)$ for (non-metric) facility location in random order.

We are working on generalizing to Group Steiner Tree!

Theorem: $\Omega(\log n \log m)$ for "batched" RO set cover.

Corollary: $\Omega(\log m \log f(\mathcal{N}))$ for RO submodular

Recall, in Part I [Gupta L. 20], we show $O(\log m \log(n \cdot f(\mathcal{N})))$ for adversarial order.

Online Set Cover With-a-Sample

New!

Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

S_1

S_2

S_3

S_4

S_5

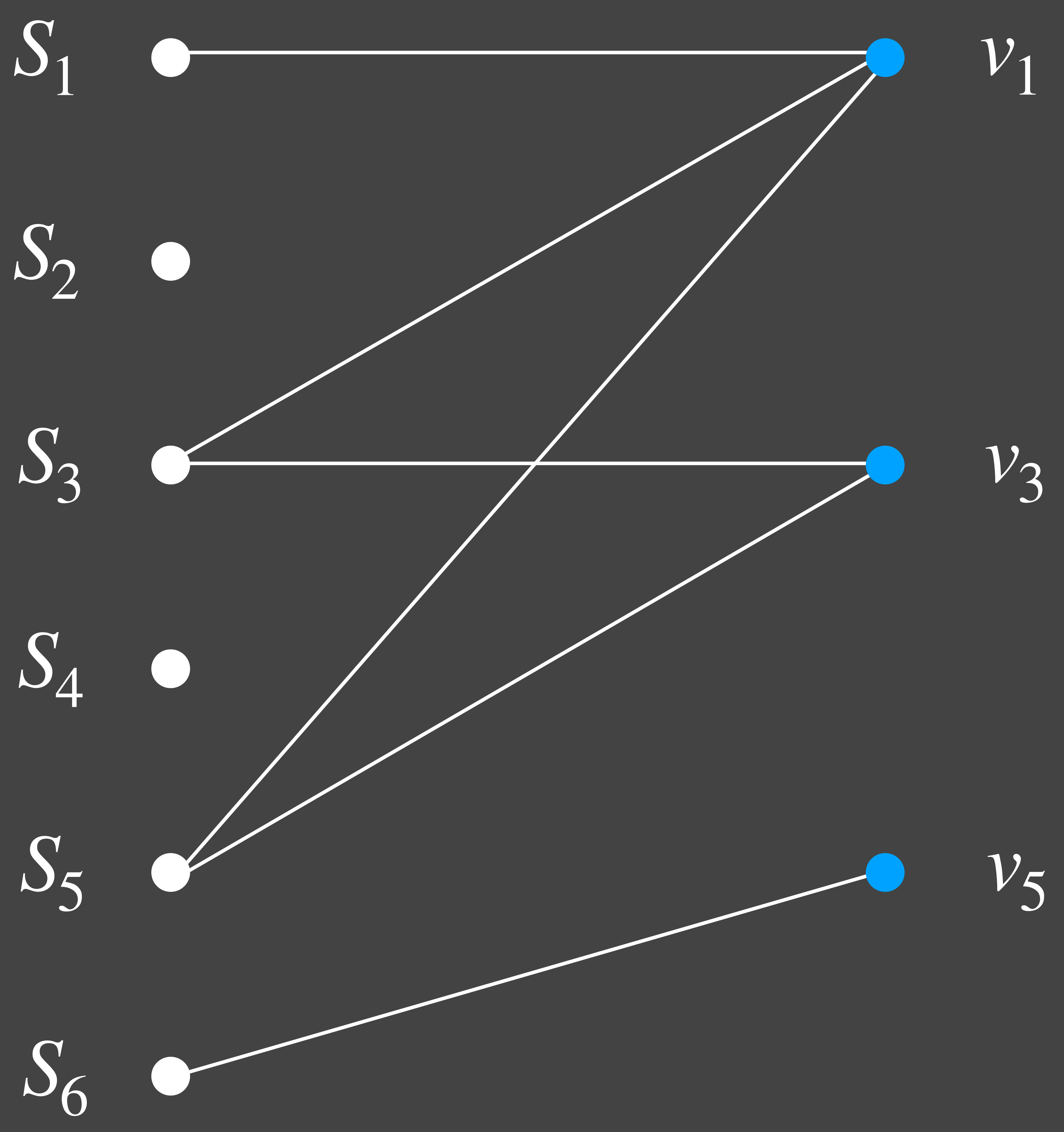
S_6

Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

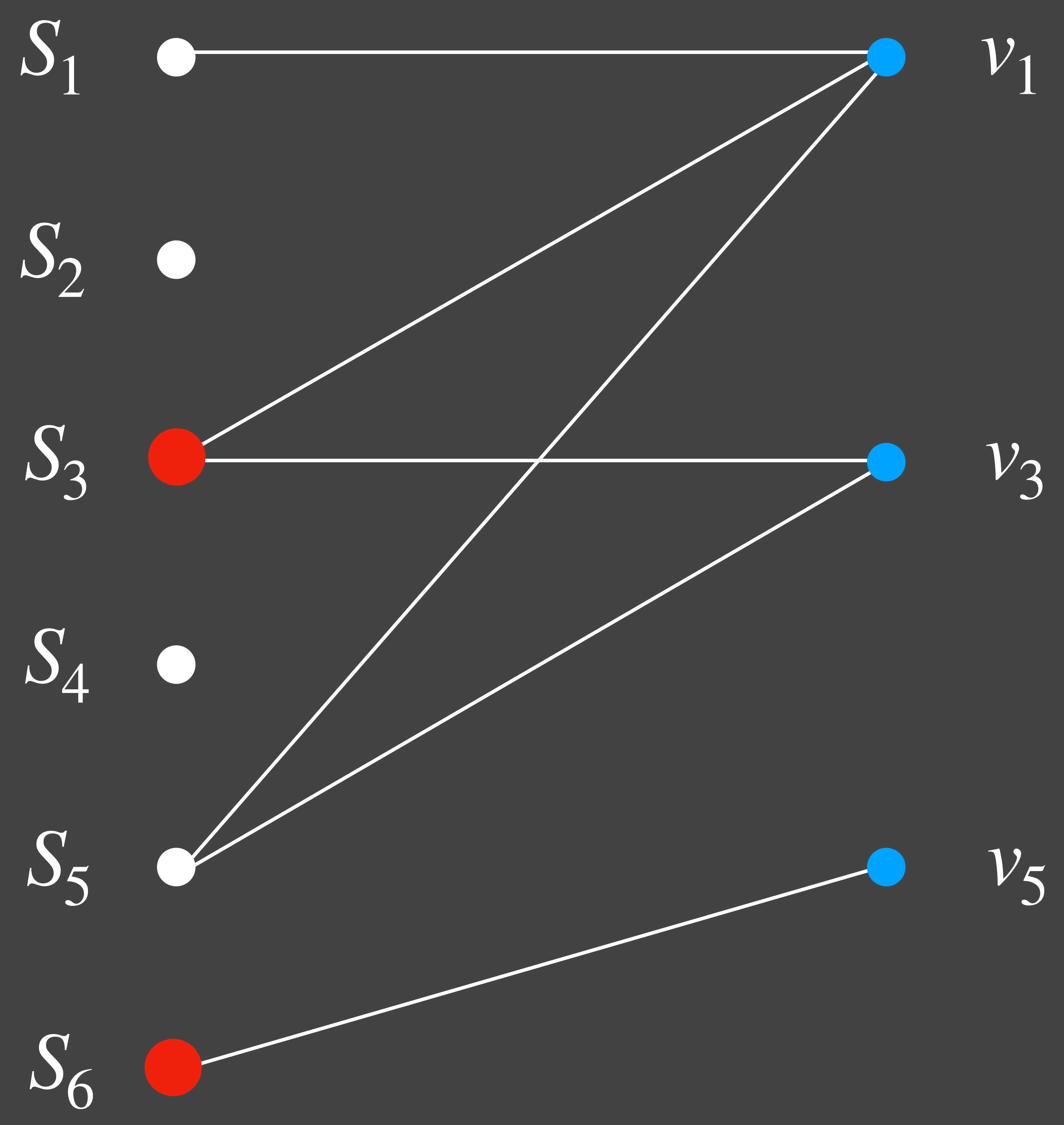


Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in <u>adversarial order.</u>

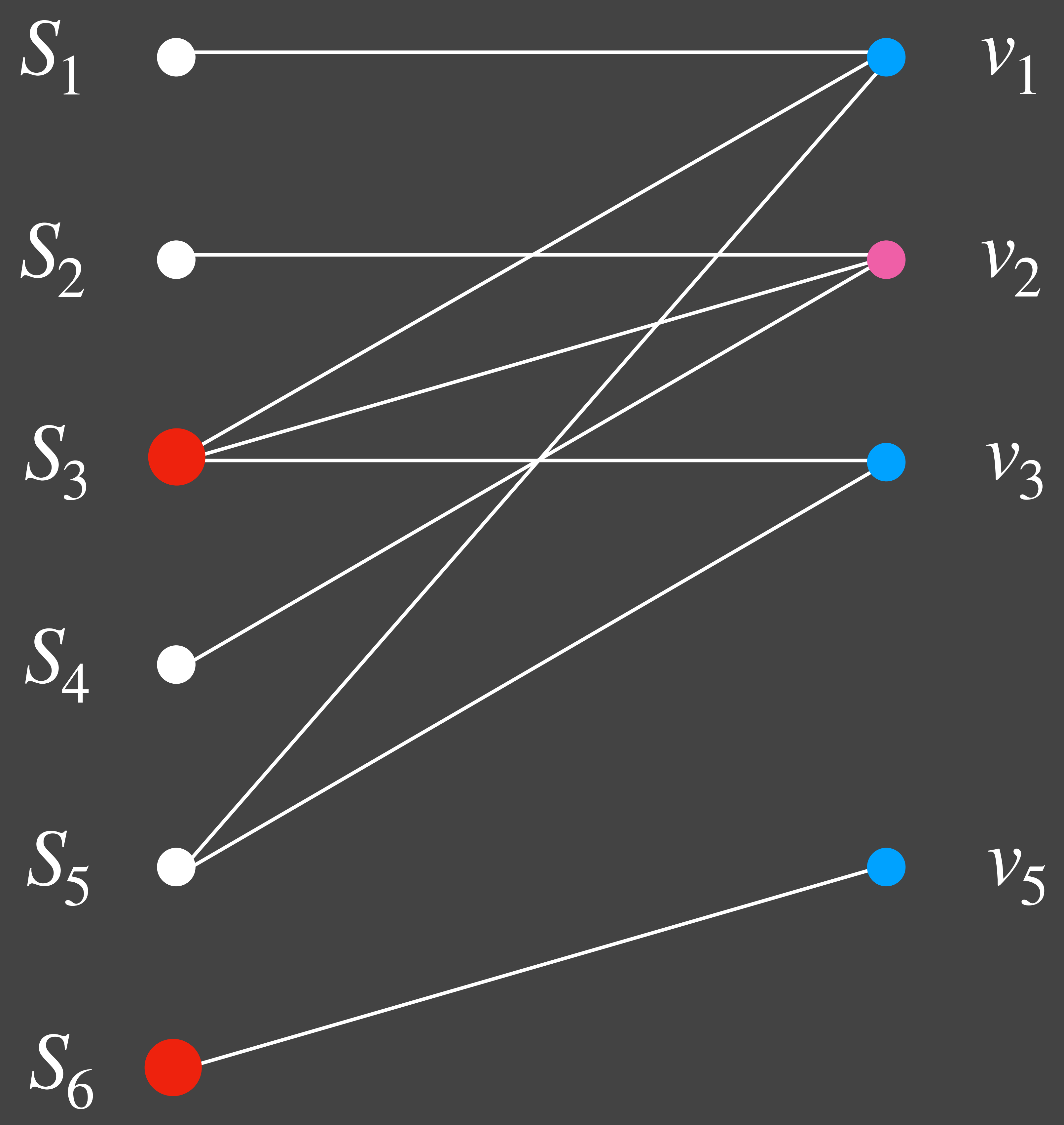


Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

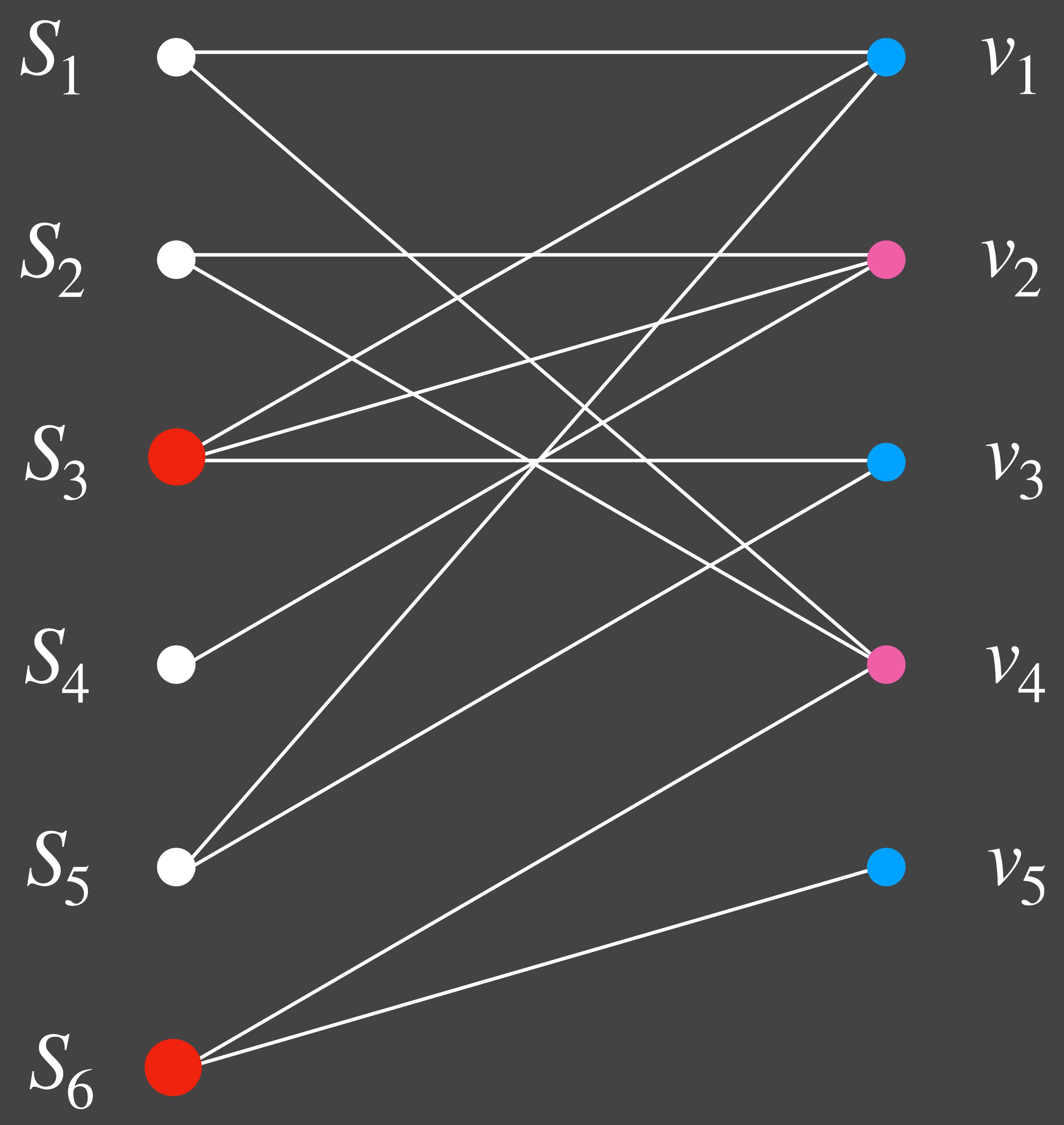


Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

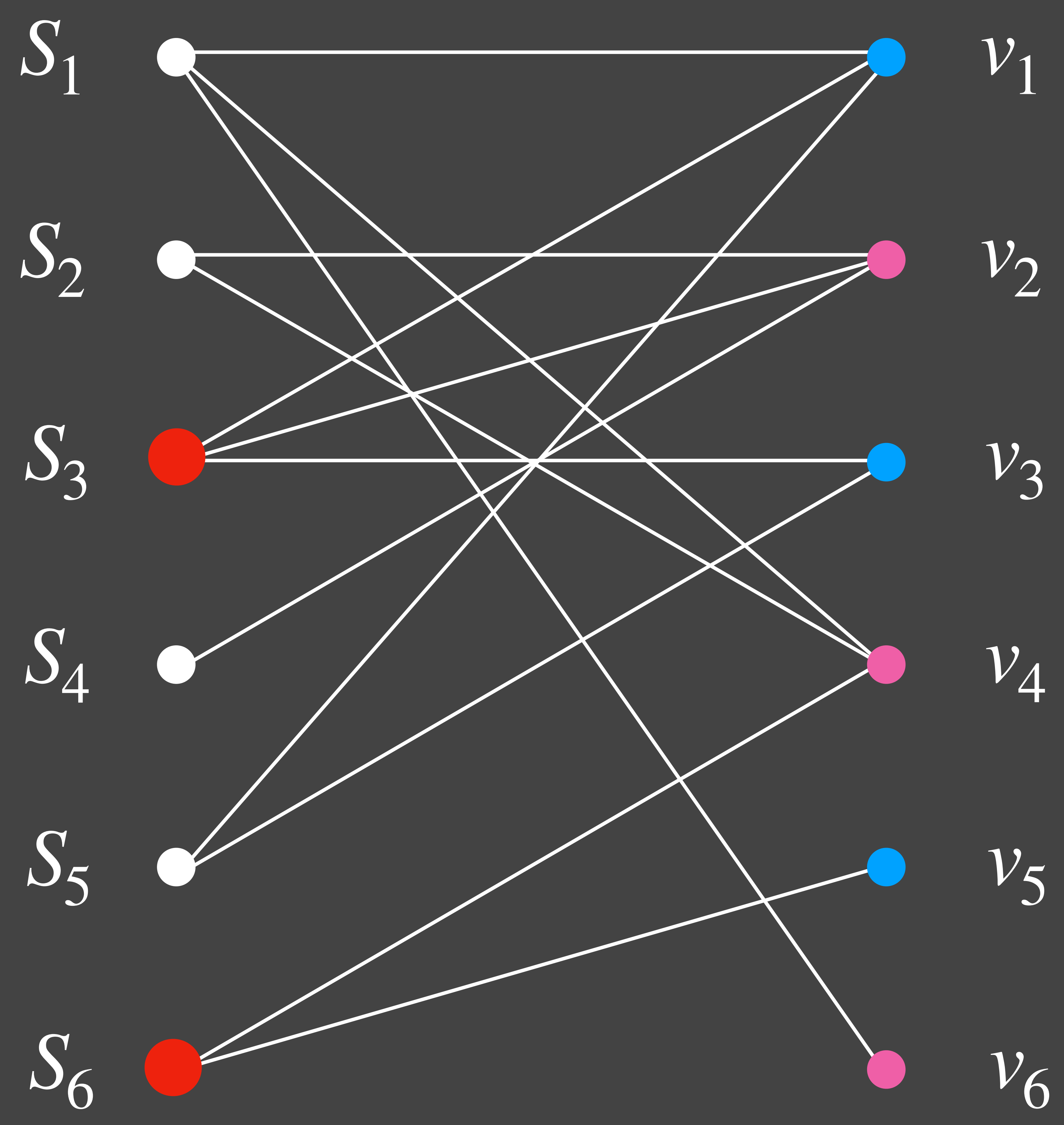


New!

Online Set Cover With-a-Sample

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

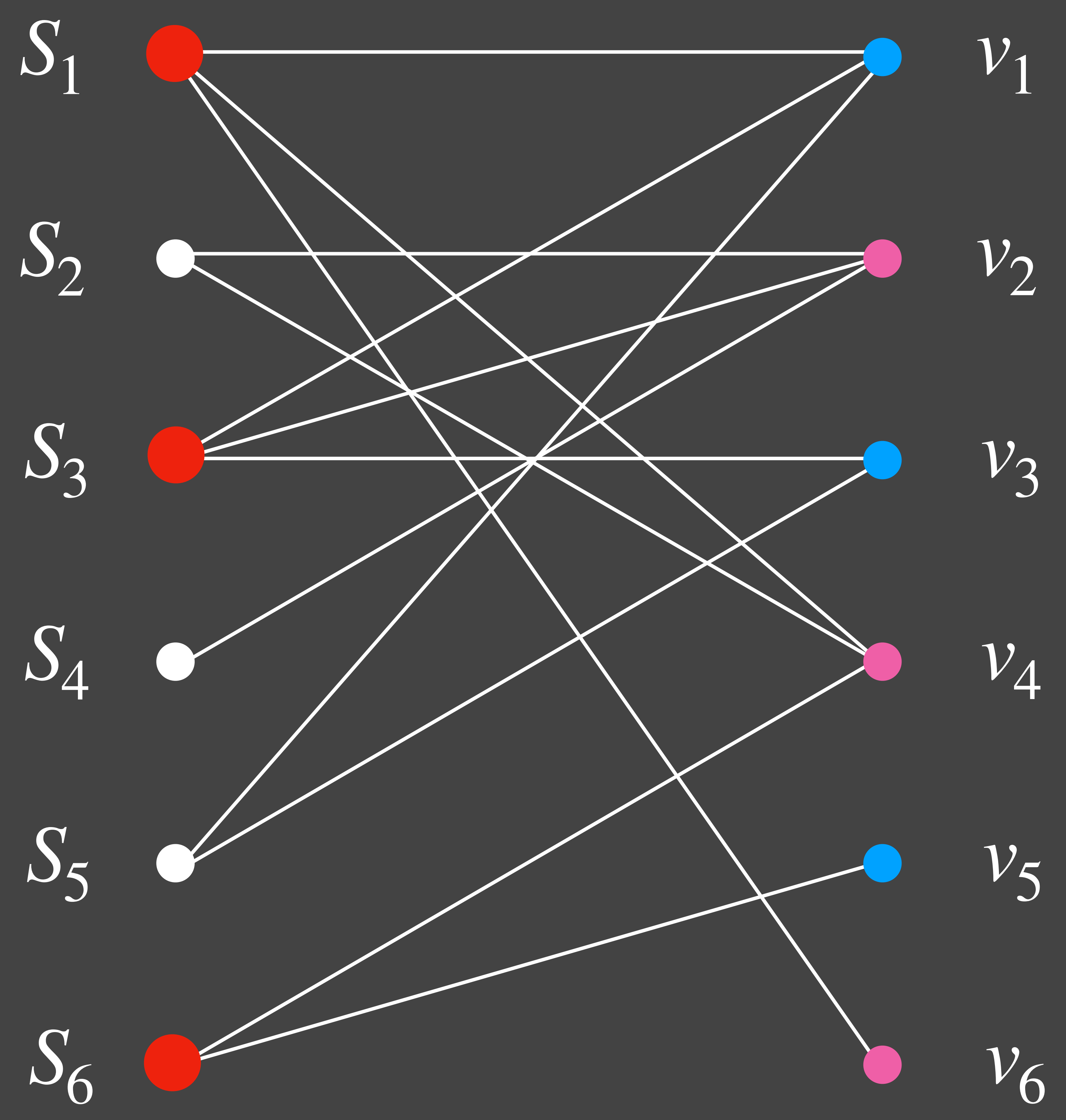


Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.

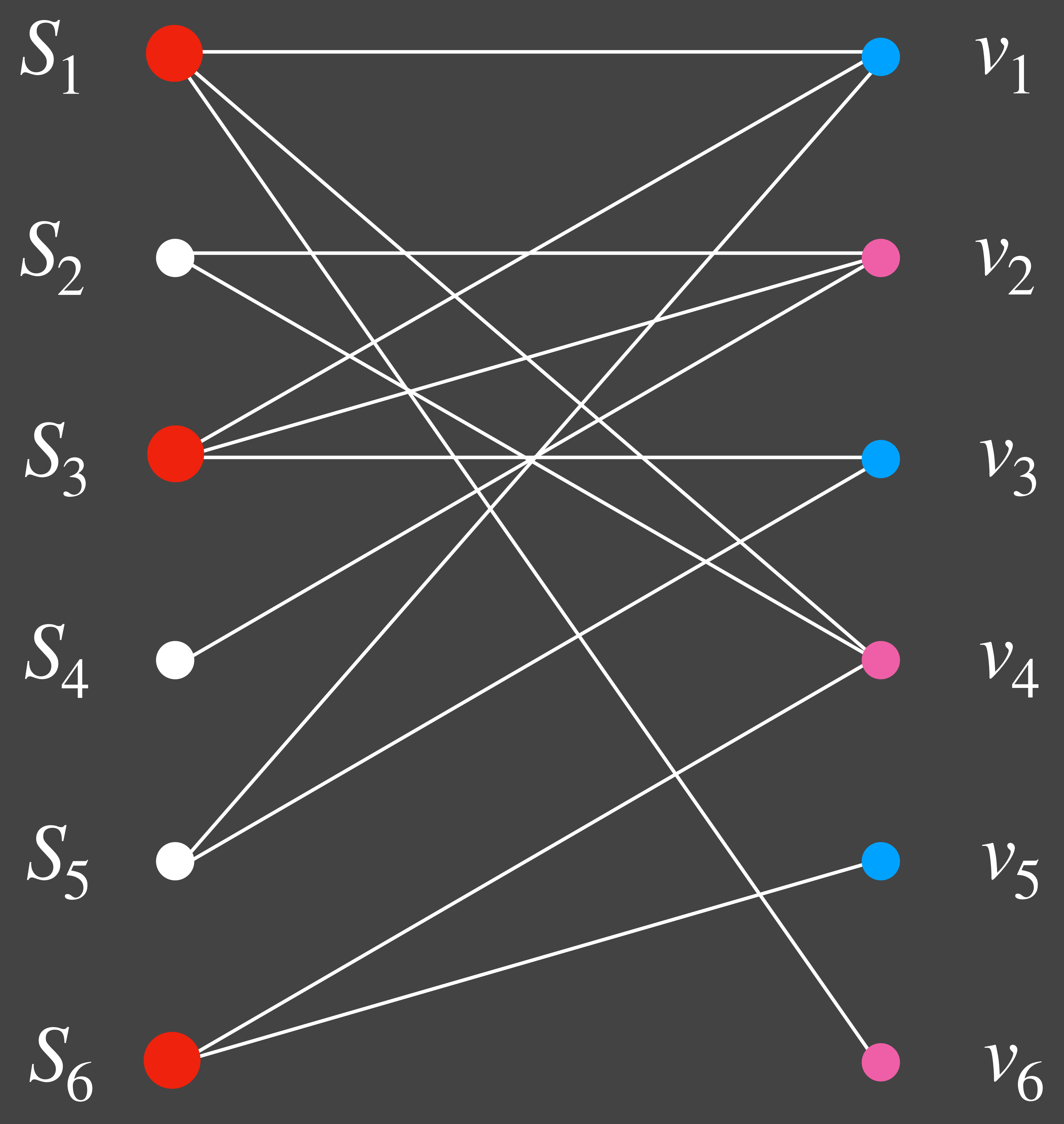


Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in <u>adversarial order.</u>



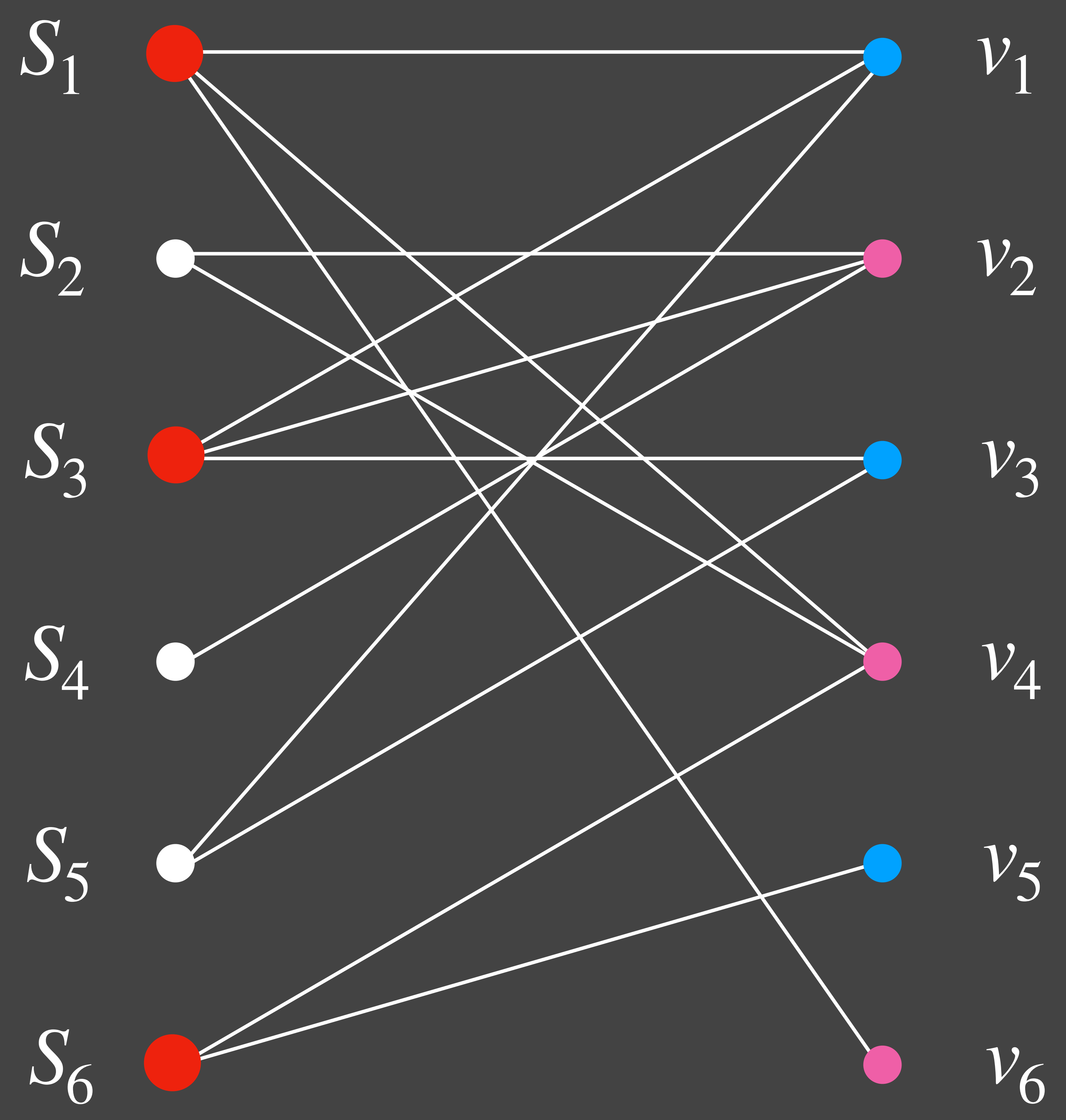
More like RO Set Cover, or adversarial-order Online Set Cover?

Online Set Cover With-a-Sample

New!

Online set cover, but random 1/2 of elements known upfront (see [Kaplan Naori Raz 21]).

Remaining fraction revealed in adversarial order.



More like RO Set Cover, or adversarial-order Online Set Cover?

Theorem:

*There is a **randomized poly time** algorithm for **Online Set** Cover With-a-Sample with competitive ratio* $O(\log(mn))$.

Reduction to LearnOrCover!

S_1 ●

S_2 ●

S_3 ●

S_4 ●

S_5 ●

S_6 ●

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.

S_1 ●

S_2 ●

S_3 ●

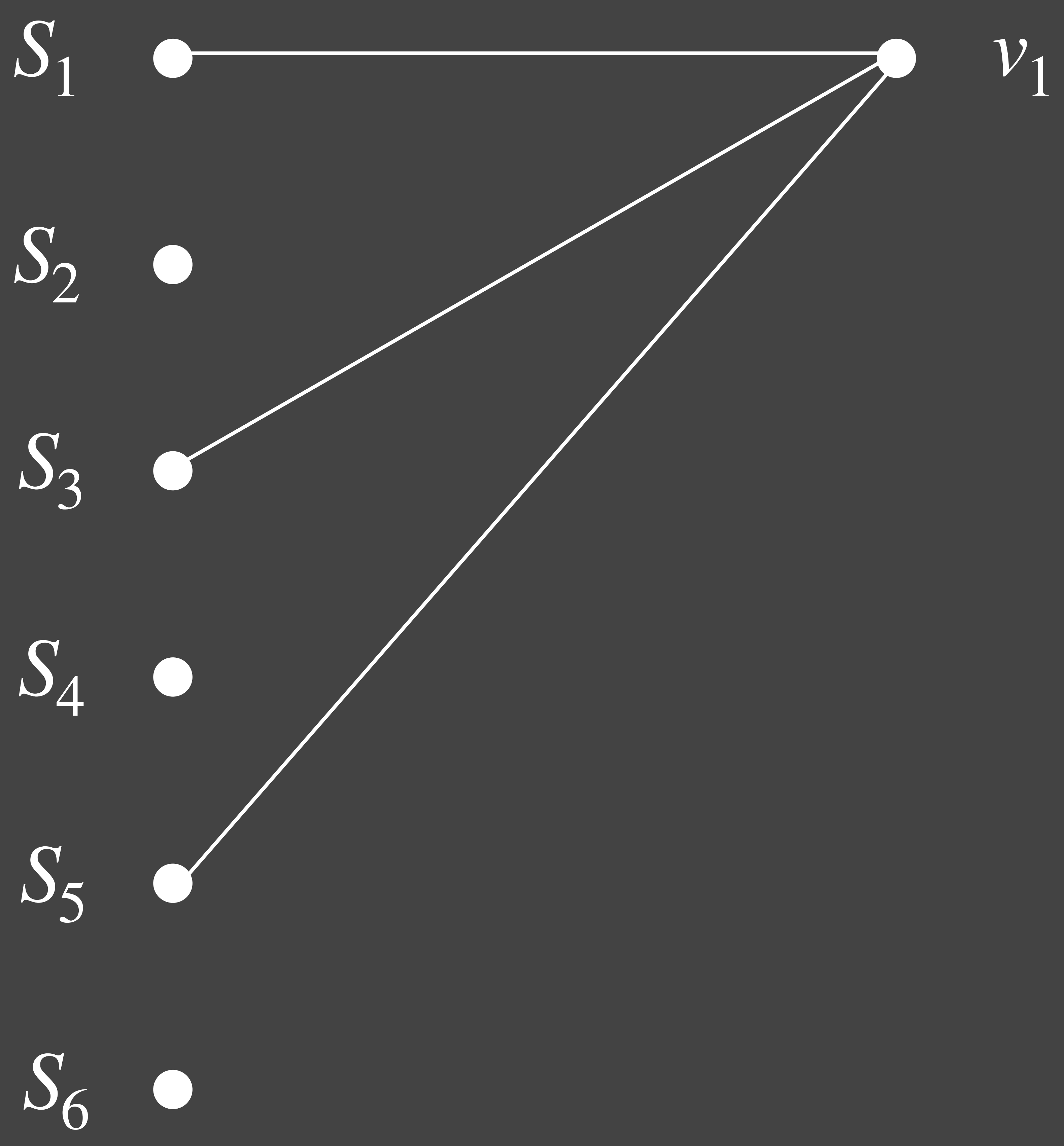
S_4 ●

S_5 ●

S_6 ●

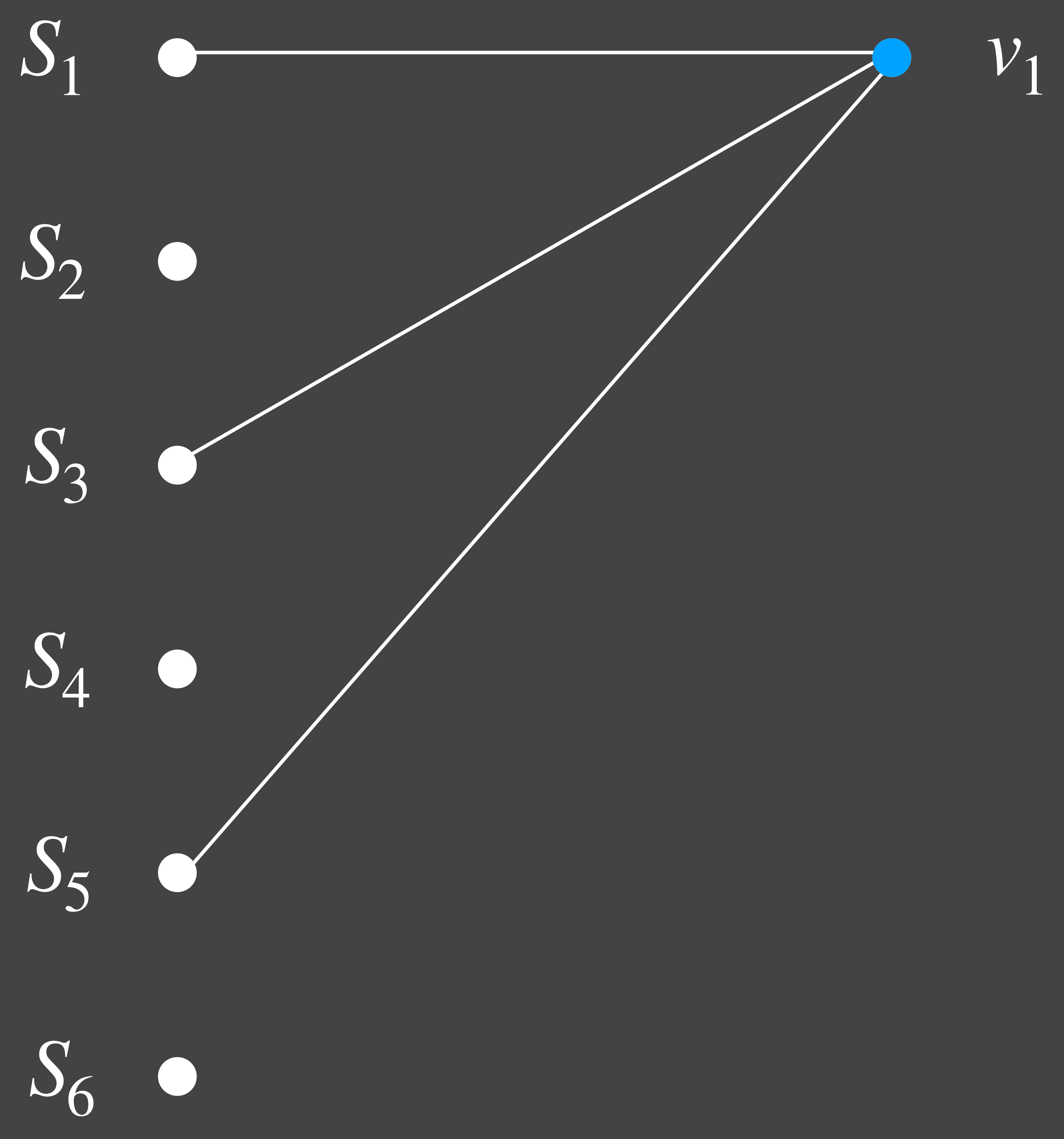
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



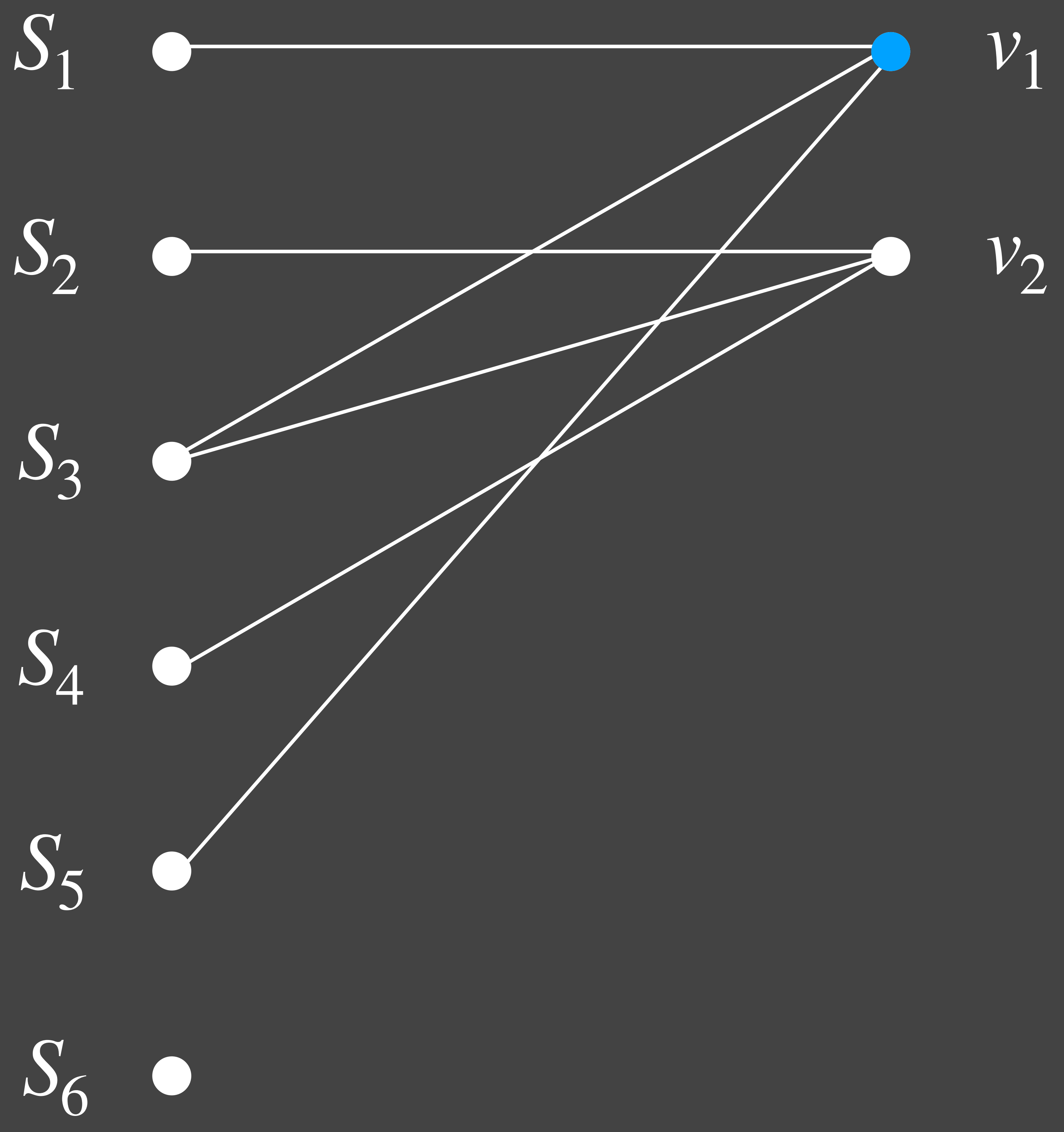
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



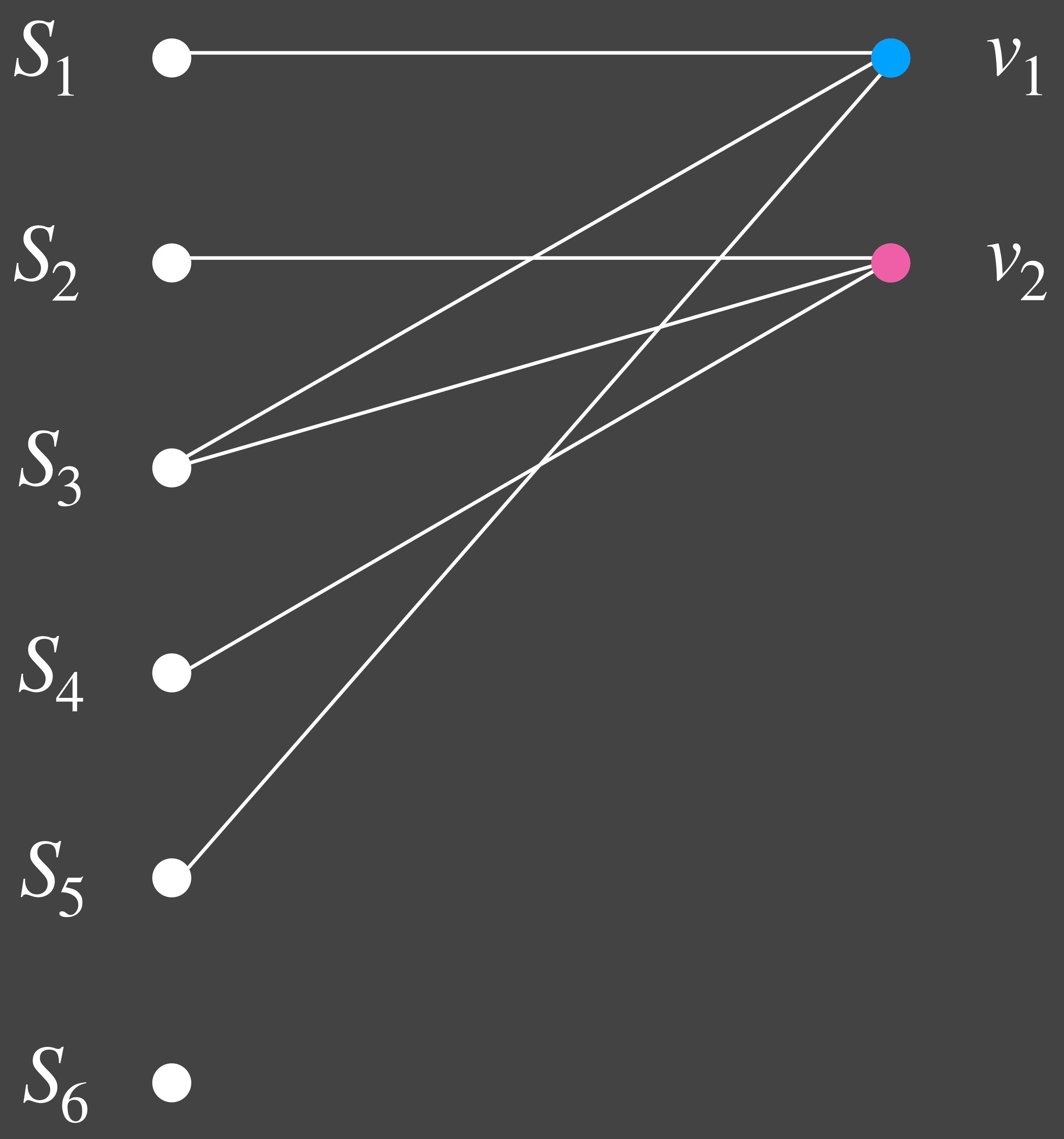
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



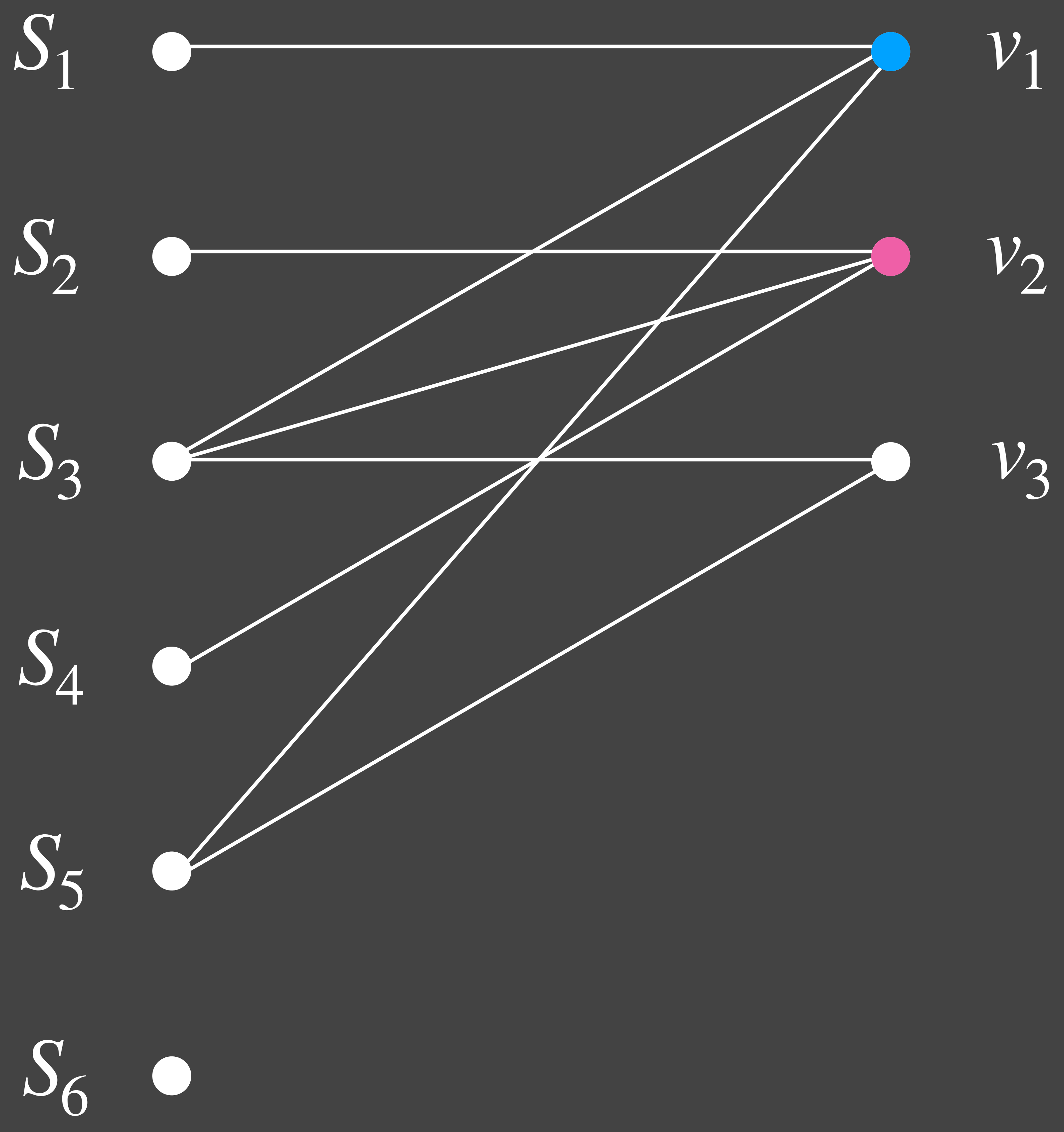
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



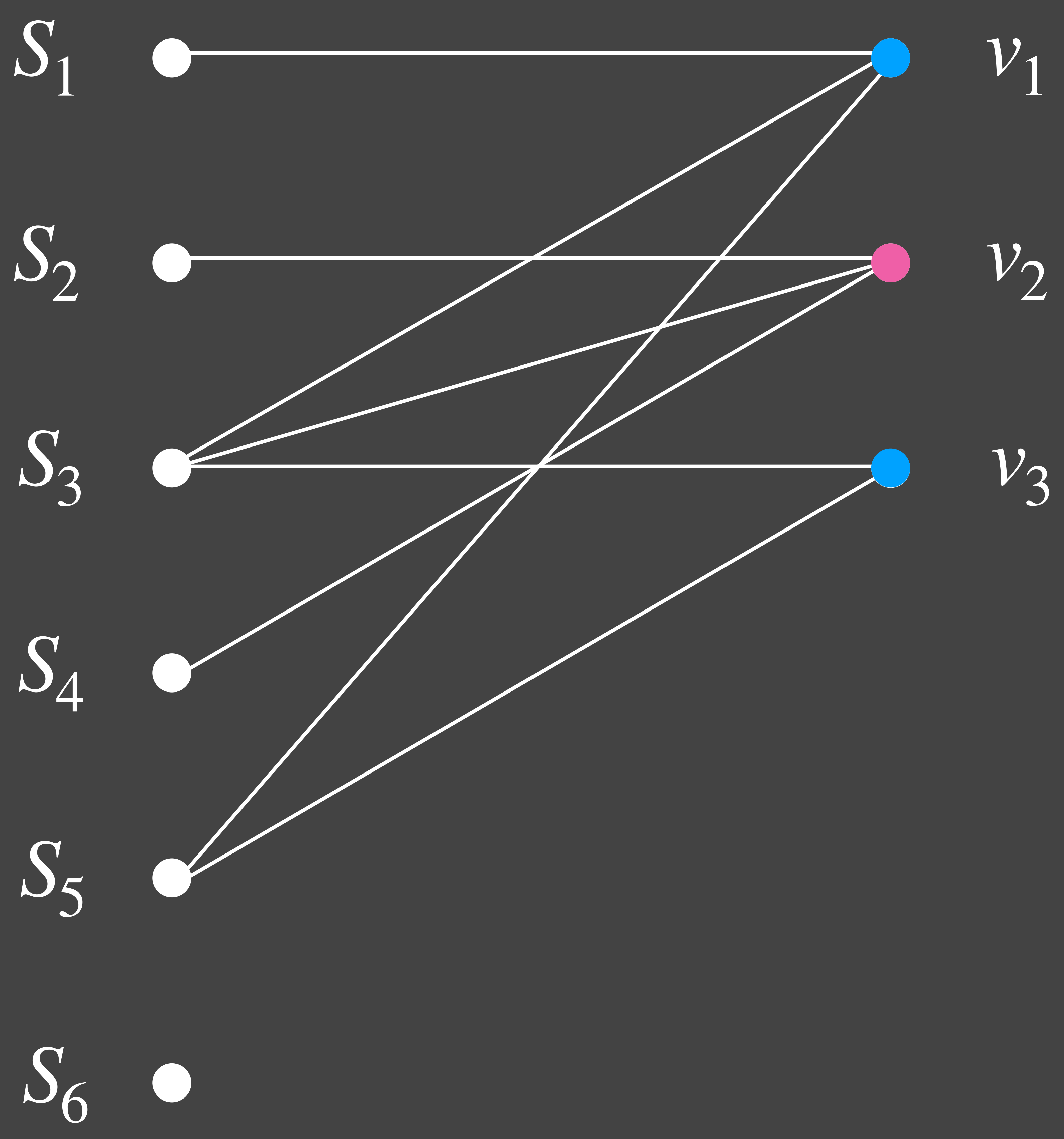
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



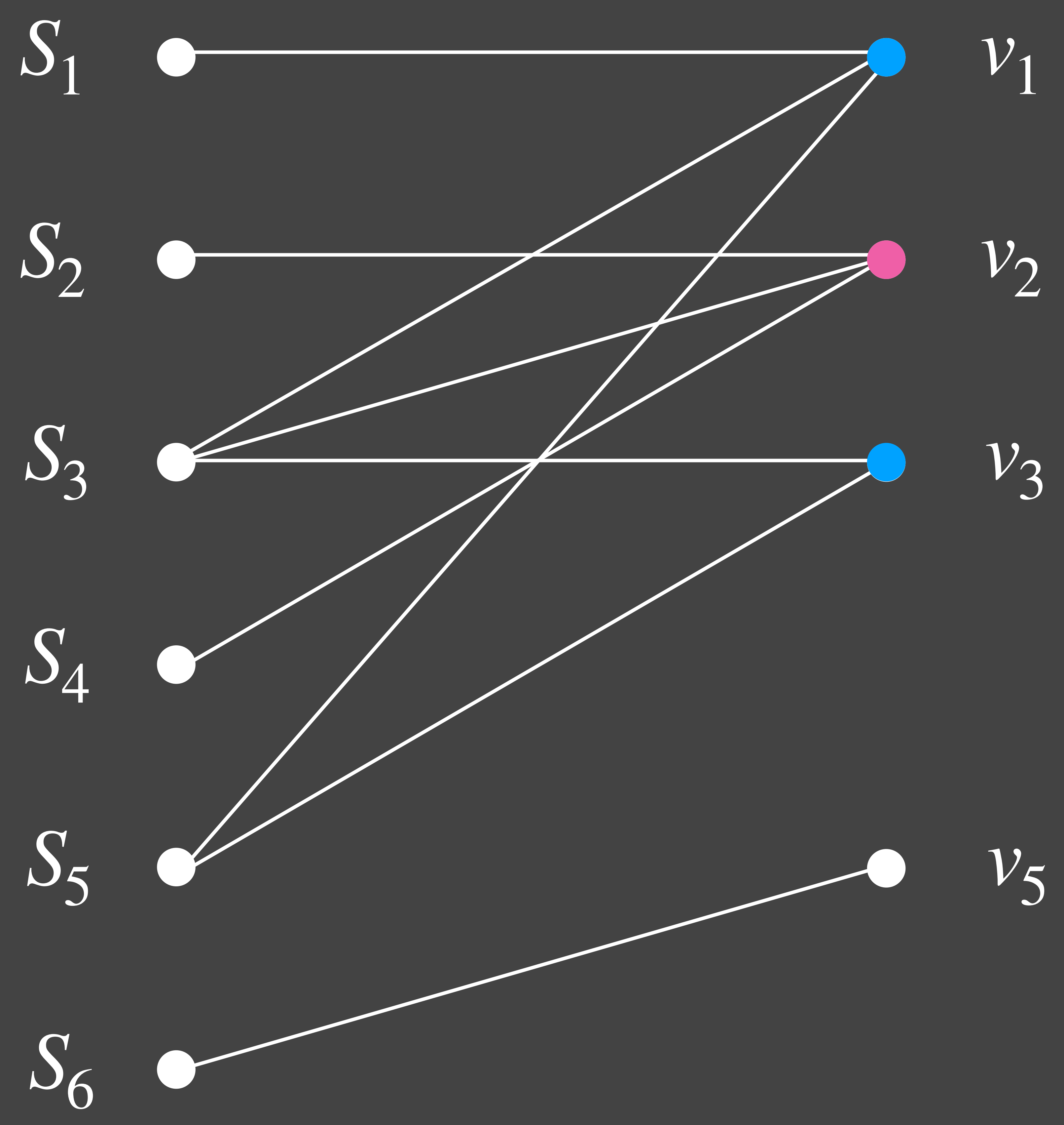
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



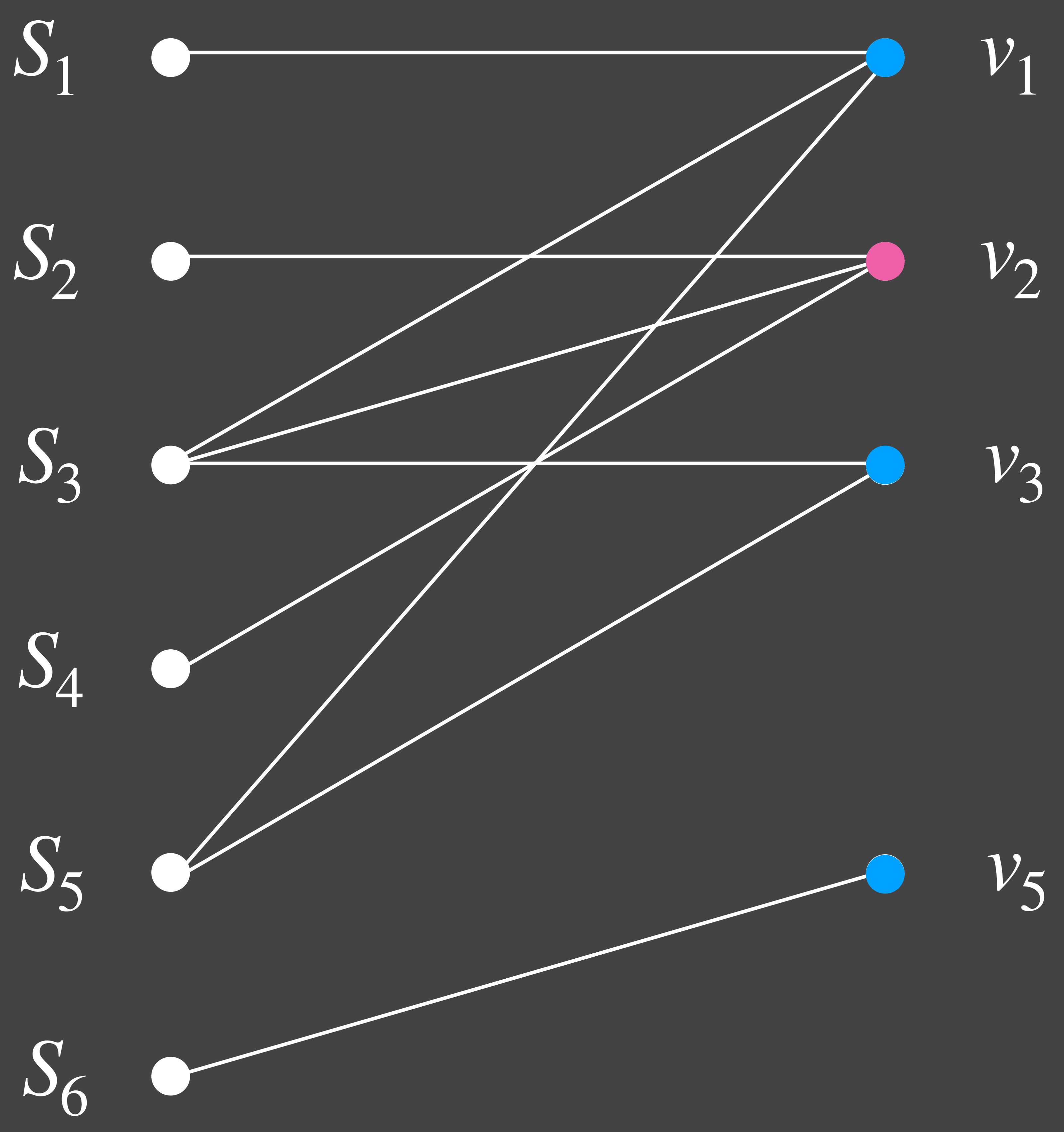
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



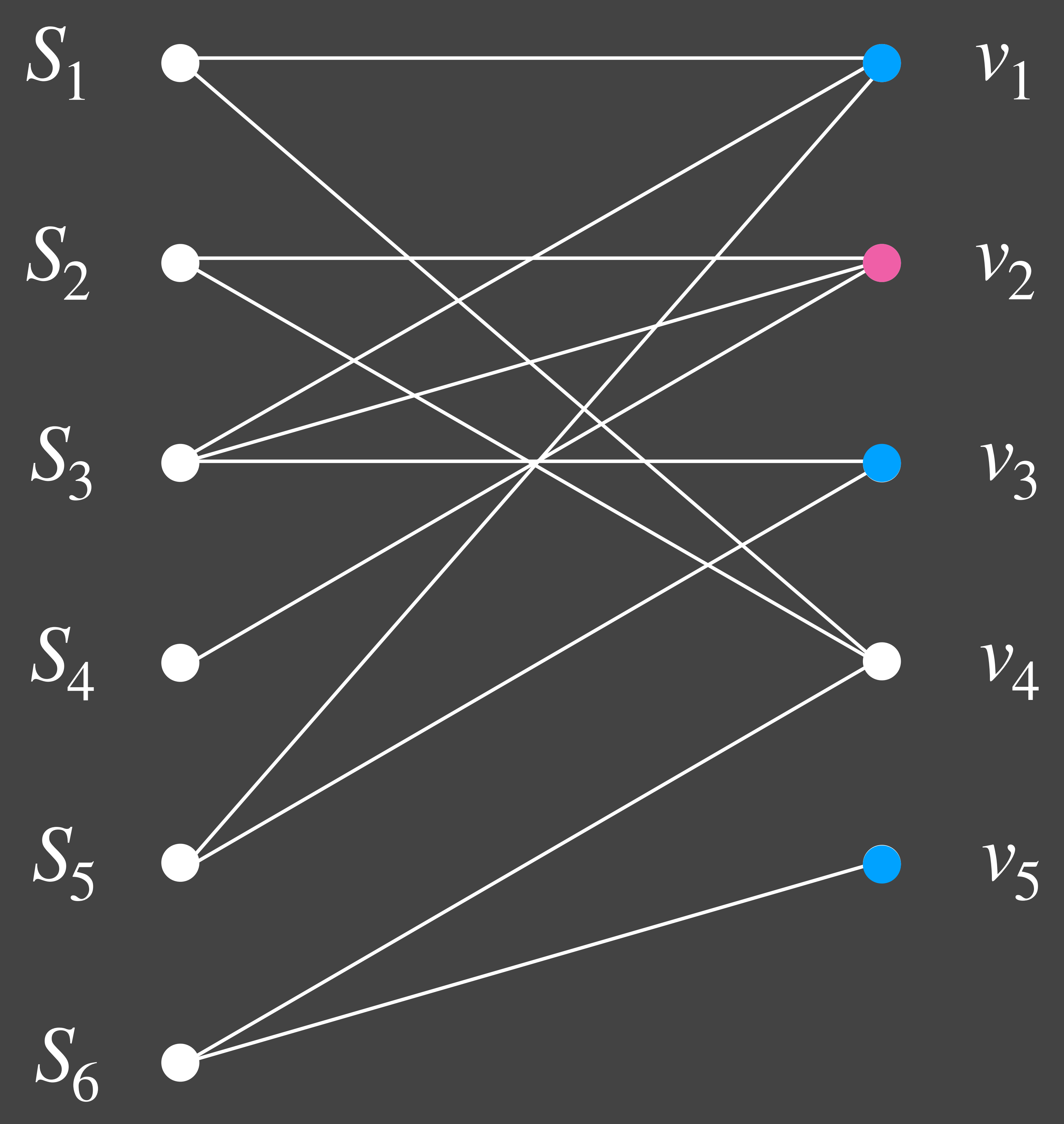
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



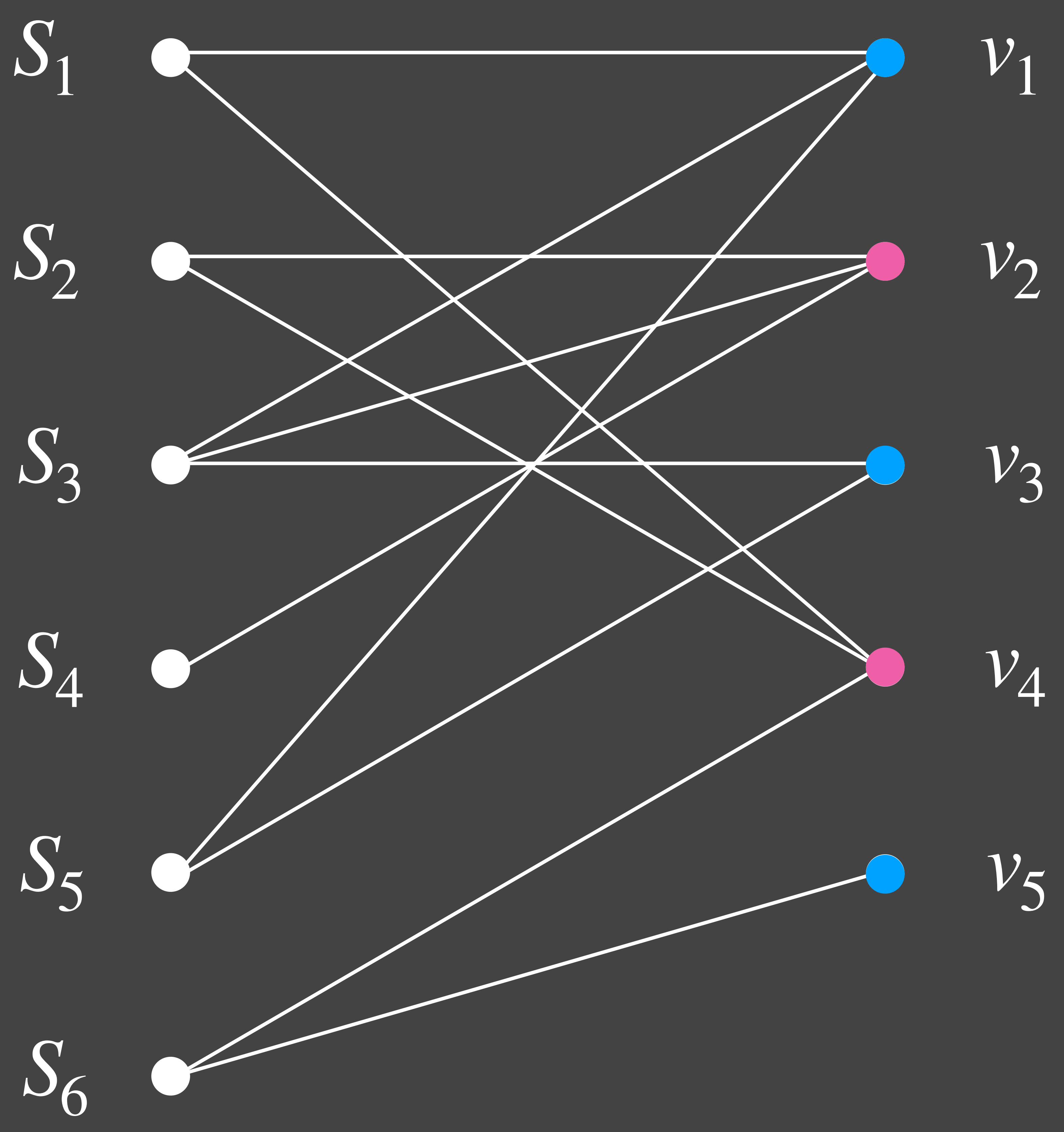
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



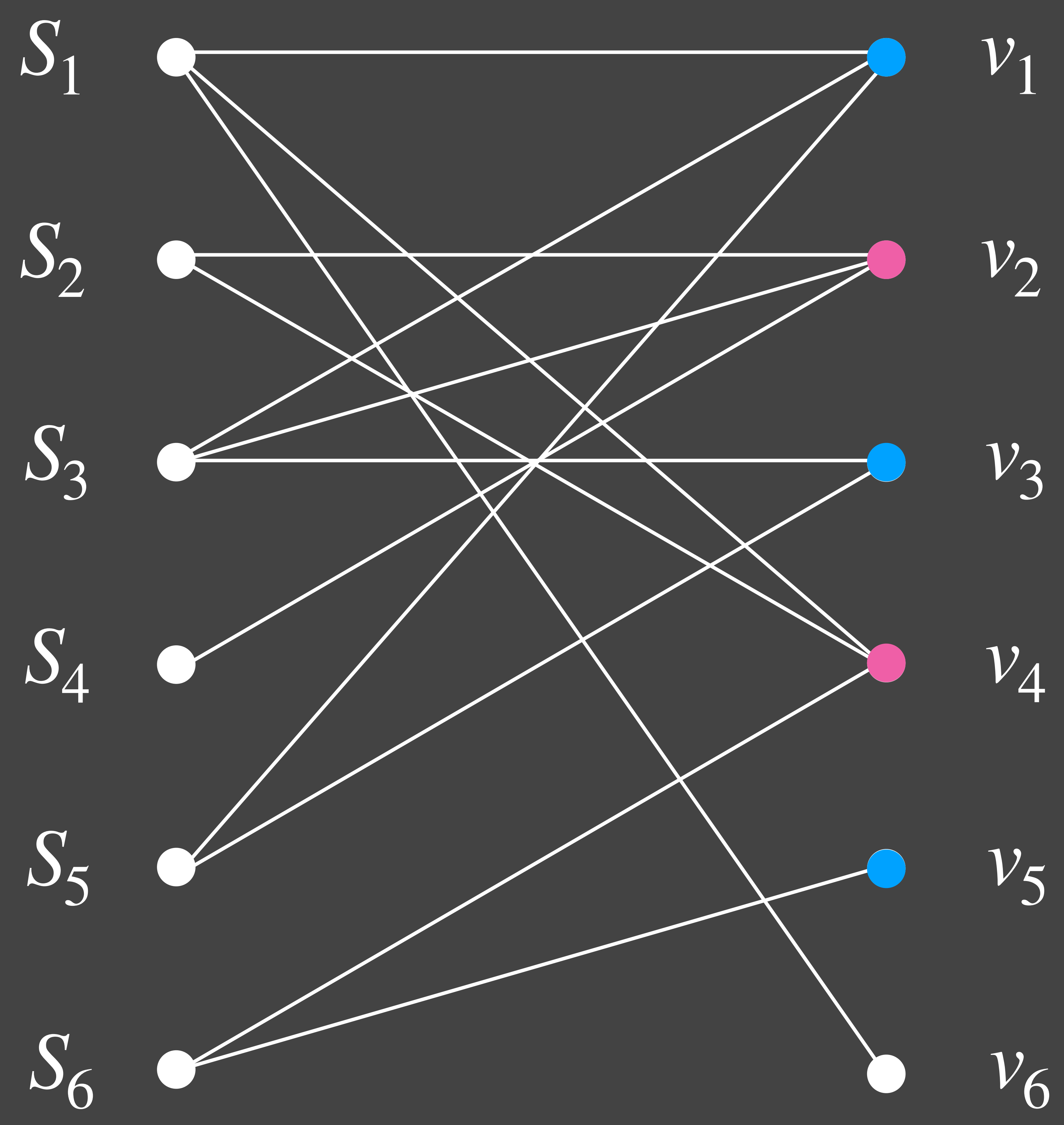
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



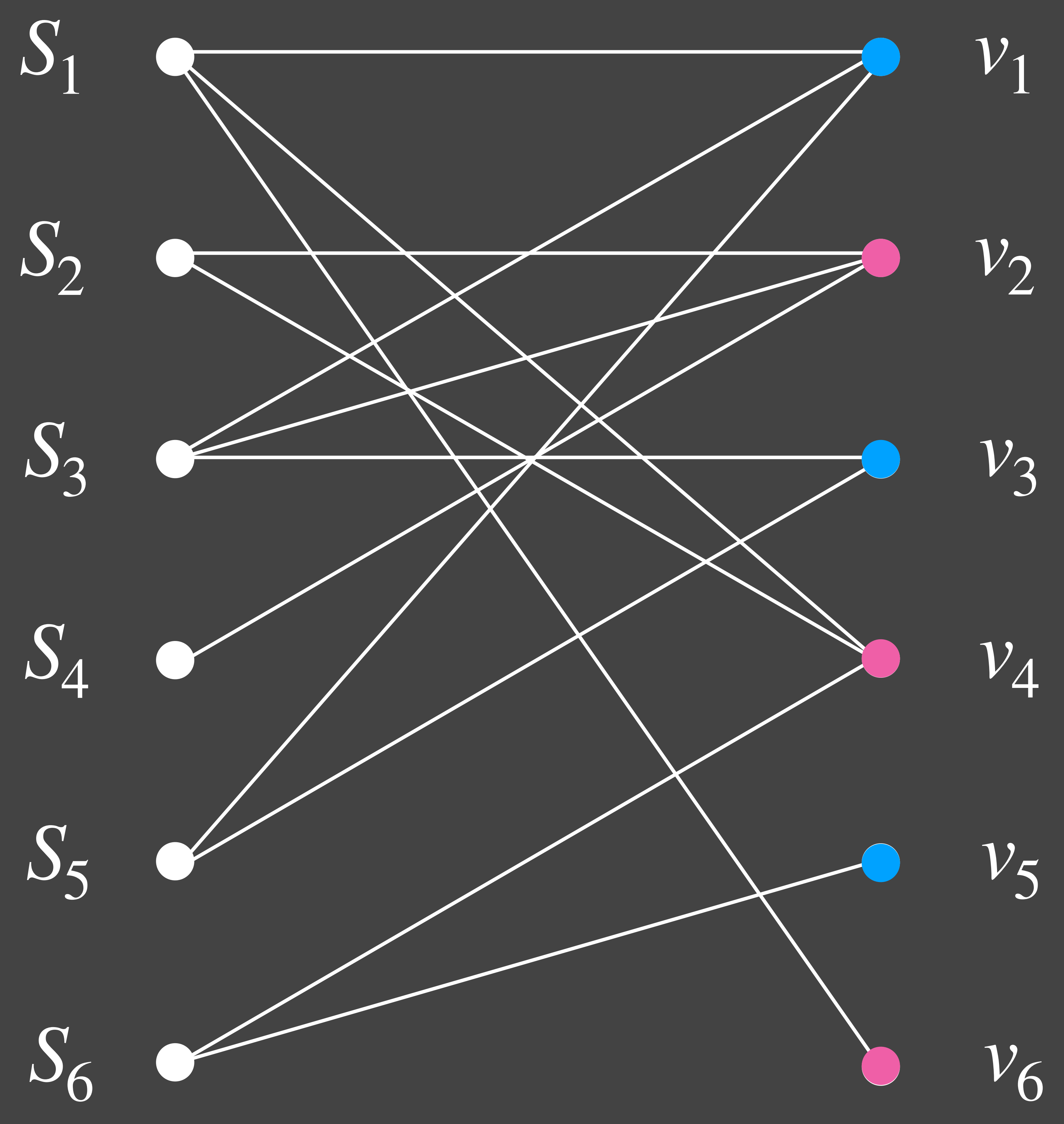
Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

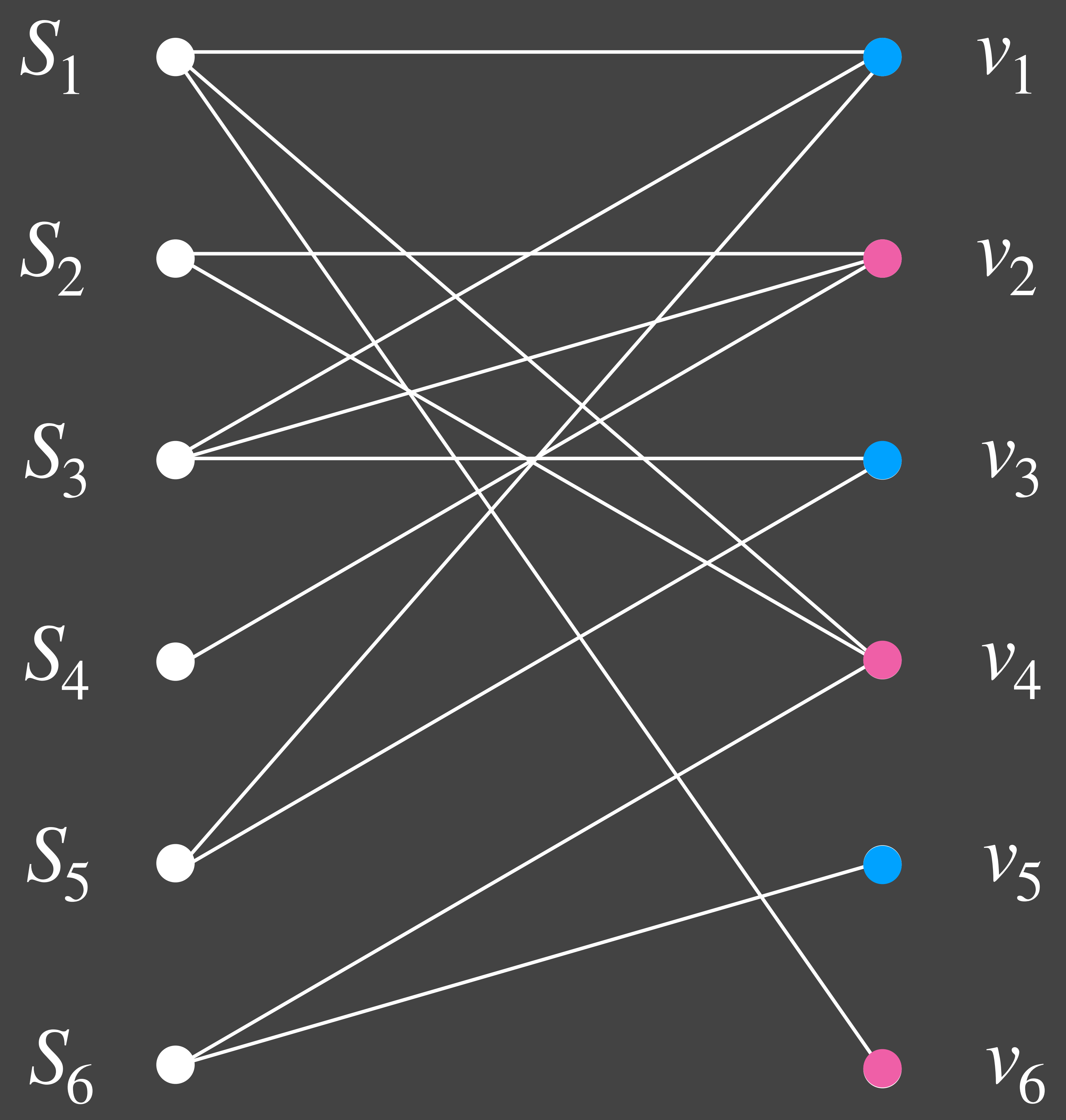
Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

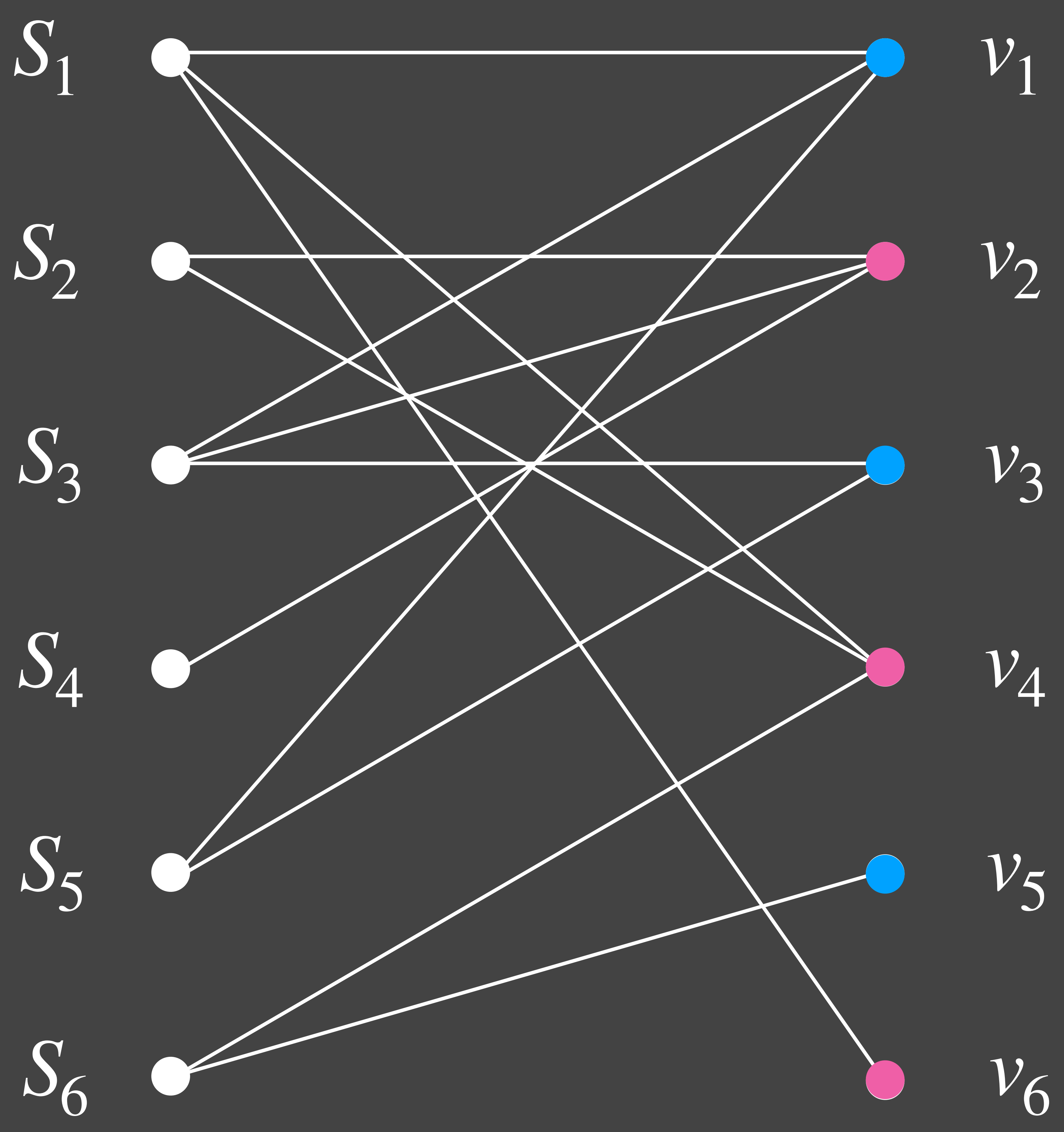
Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.

Reduction to LearnOrCover!



Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.

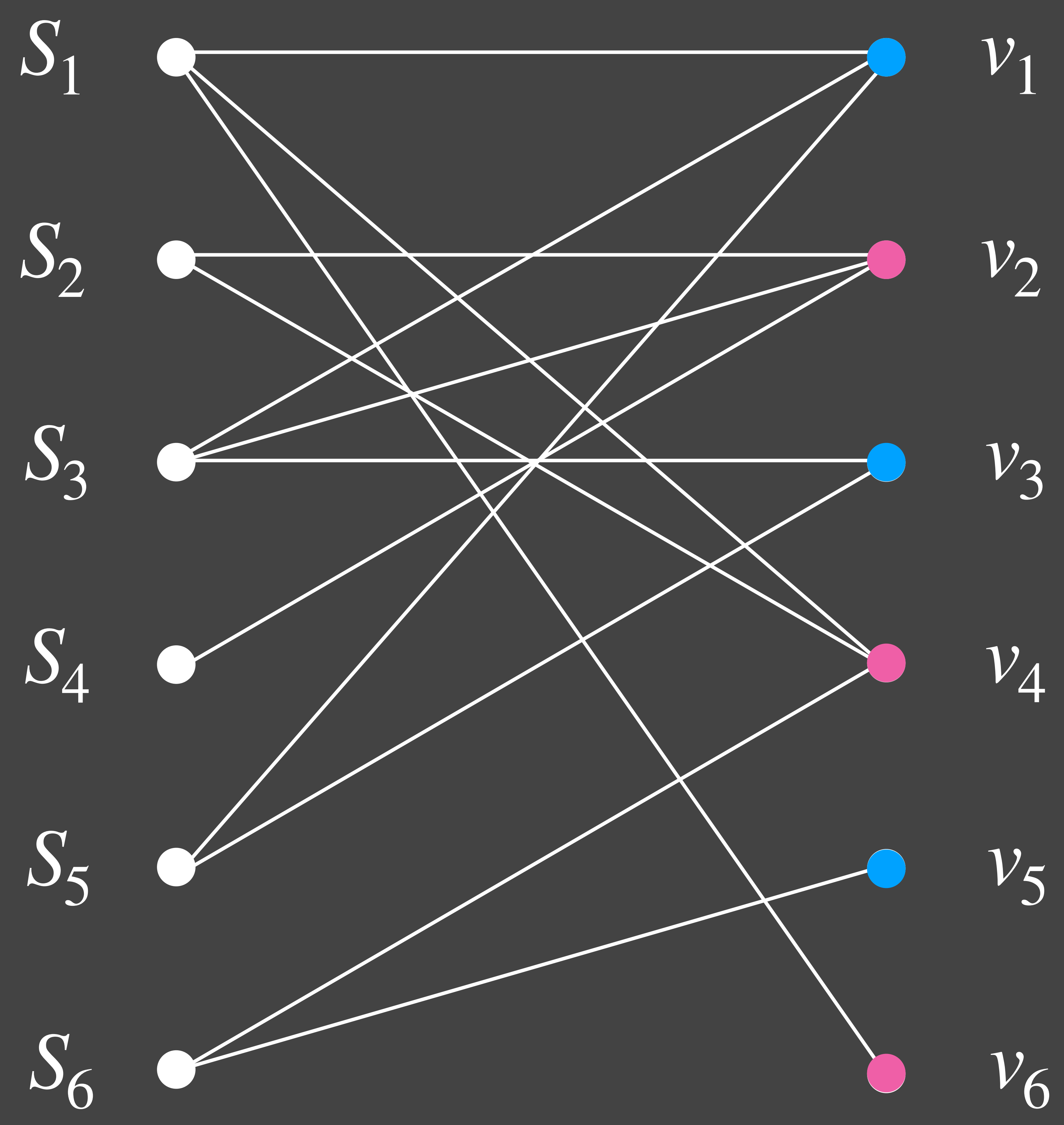


Reduction to LearnOrCover!

@ time t:

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



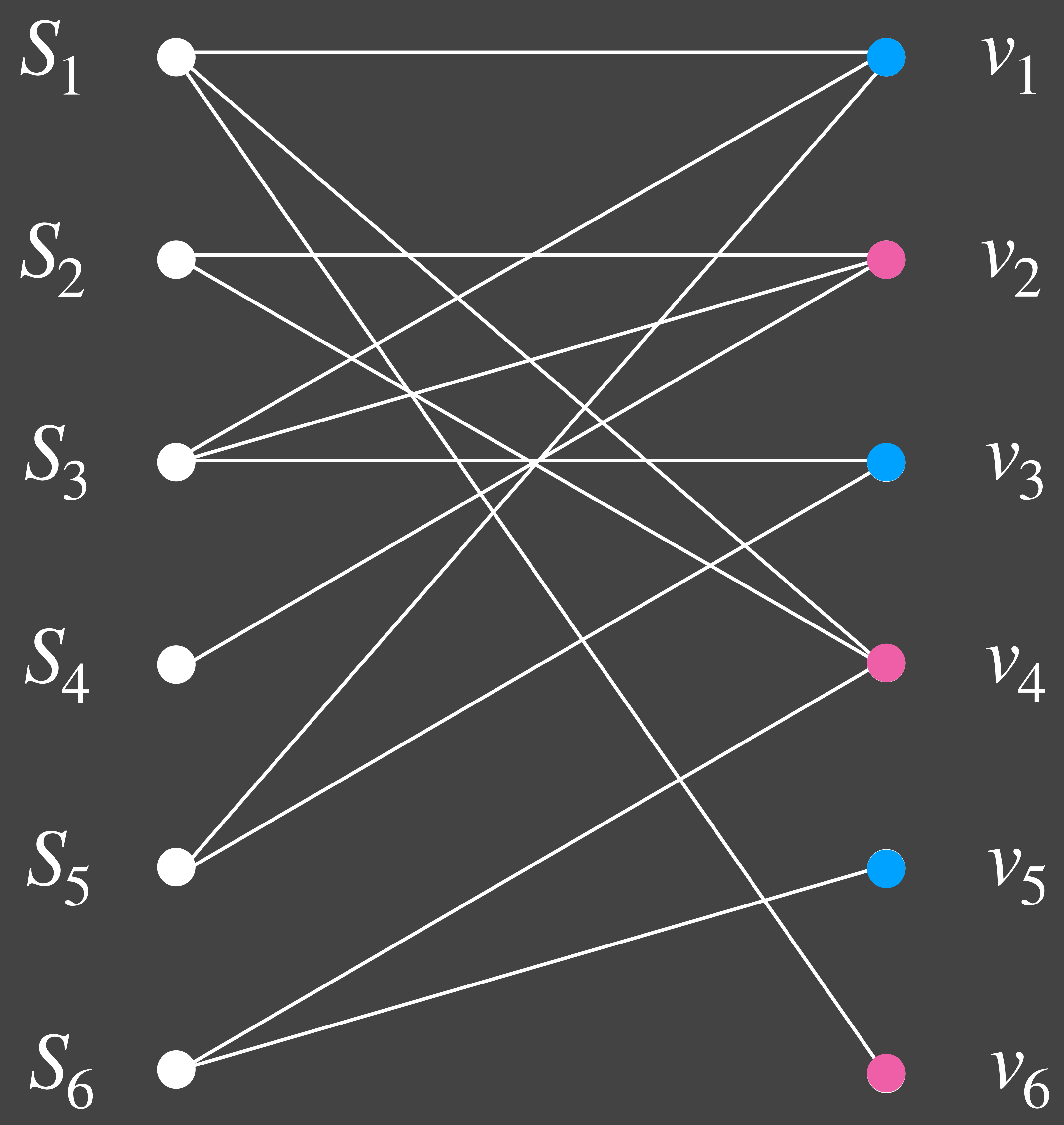
Reduction to LearnOrCover!

@ time t:

If v^t pink, feed to LearnOrCover.

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

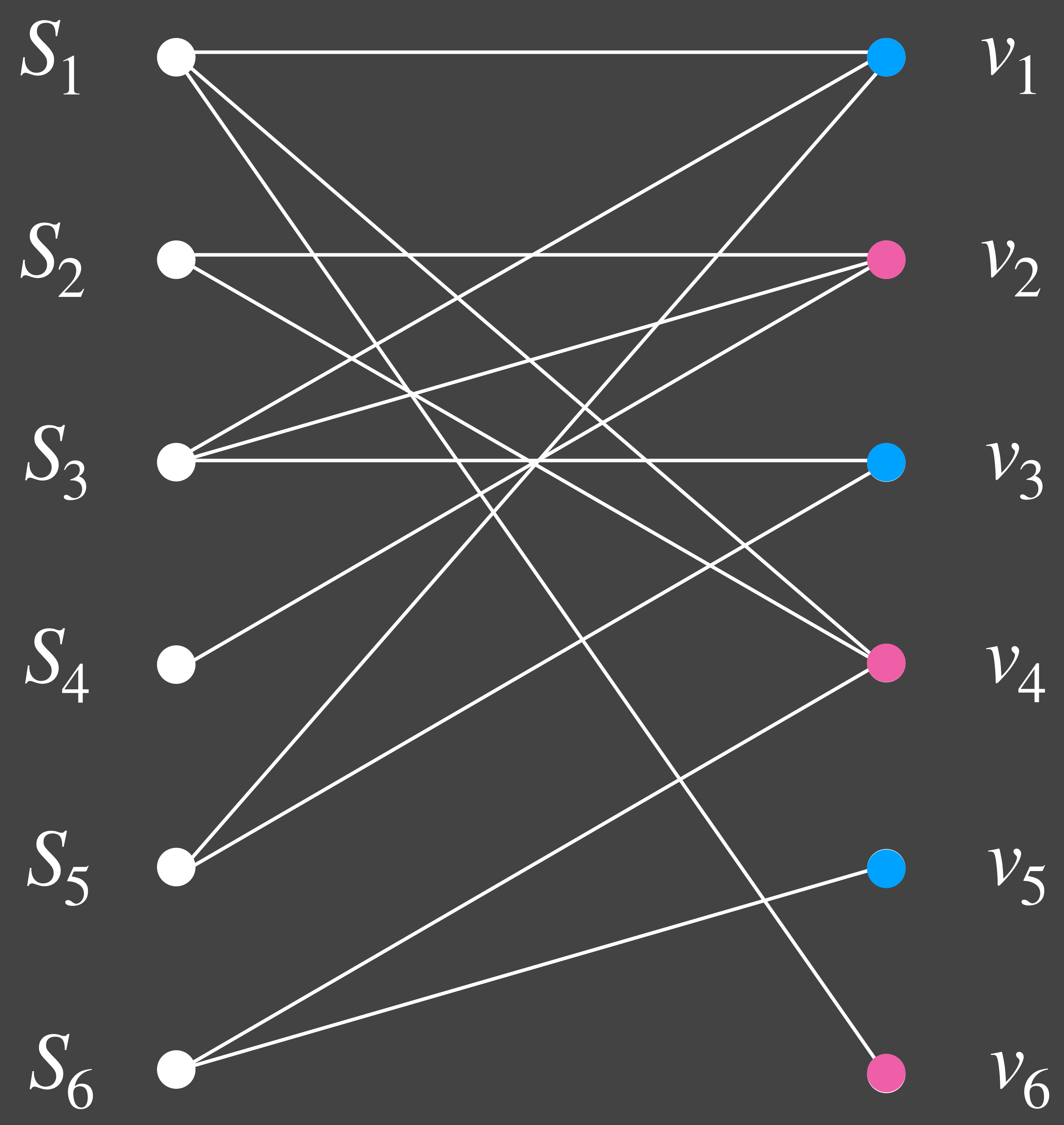
@ time t:

If v^t pink, feed to LearnOrCover.

If v^t blue, buy arbitrary set to cover.

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

@ time t:

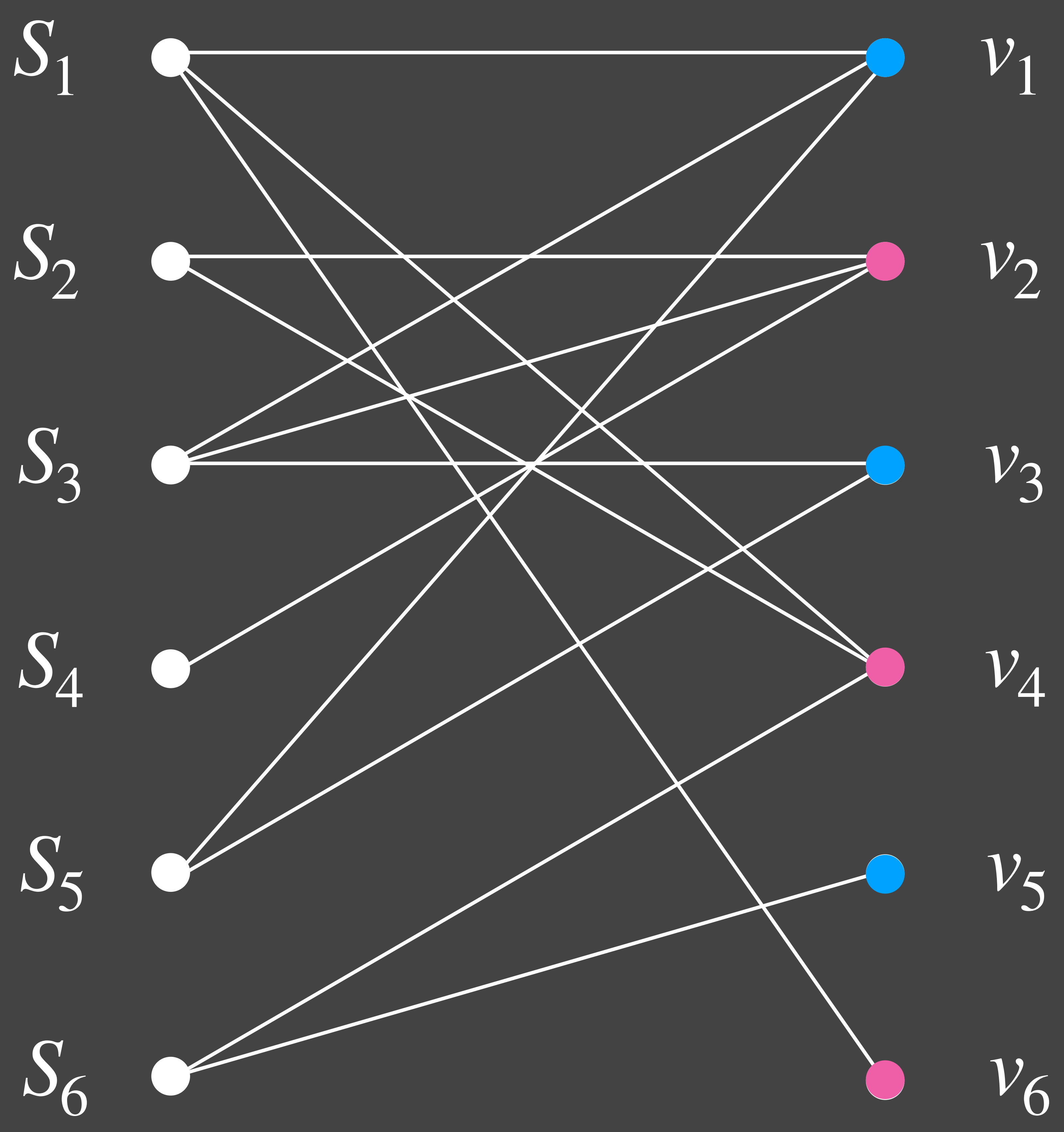
If v^t pink, feed to LearnOrCover.

If v^t blue, buy arbitrary set to cover.

Recall LearnOrCover proof template:

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

@ time t:

If v^t pink, feed to LearnOrCover.

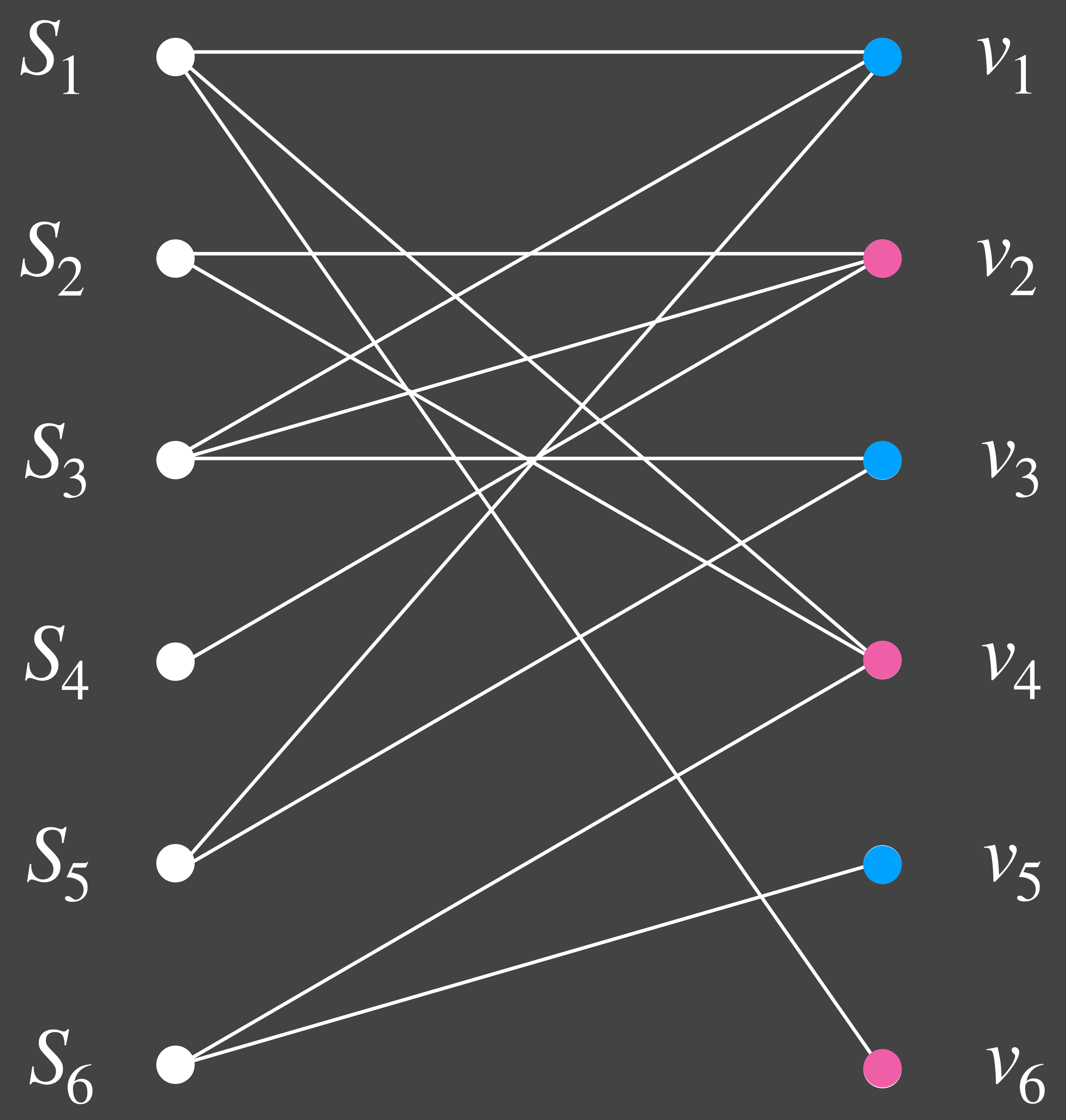
If v^t blue, buy arbitrary set to cover.

Recall LearnOrCover proof template:

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

@ time t:

If v^t pink, feed to LearnOrCover.

If v^t blue, buy arbitrary set to cover.

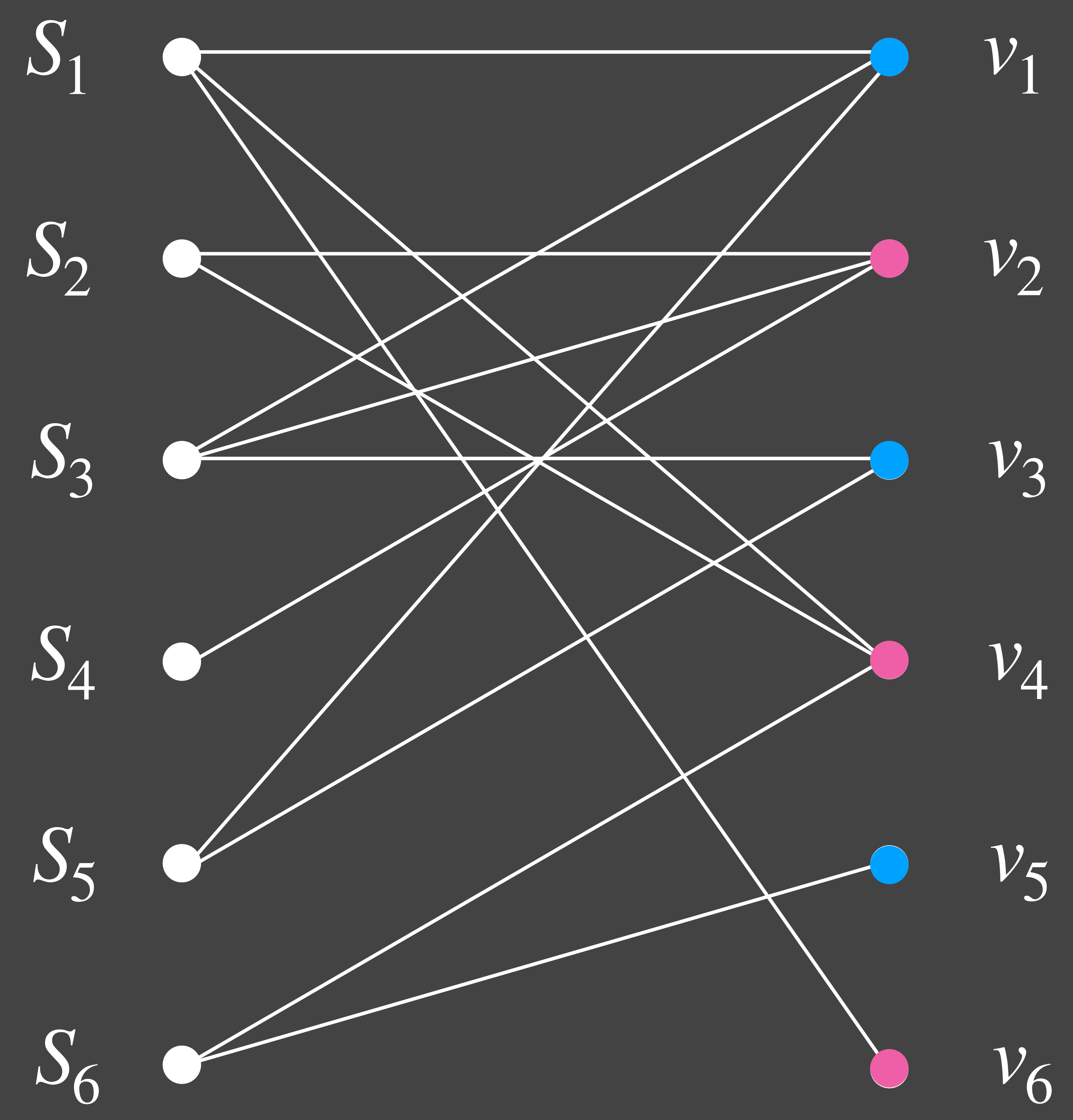
Recall LearnOrCover proof template:

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v^t uncovered, then $E[\Delta\Phi] \leq -\Omega\left(\frac{1}{k}\right)$.

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

@ time t:

If v^t pink, feed to LearnOrCover.

If v^t blue, buy arbitrary set to cover.

Recall LearnOrCover proof template:

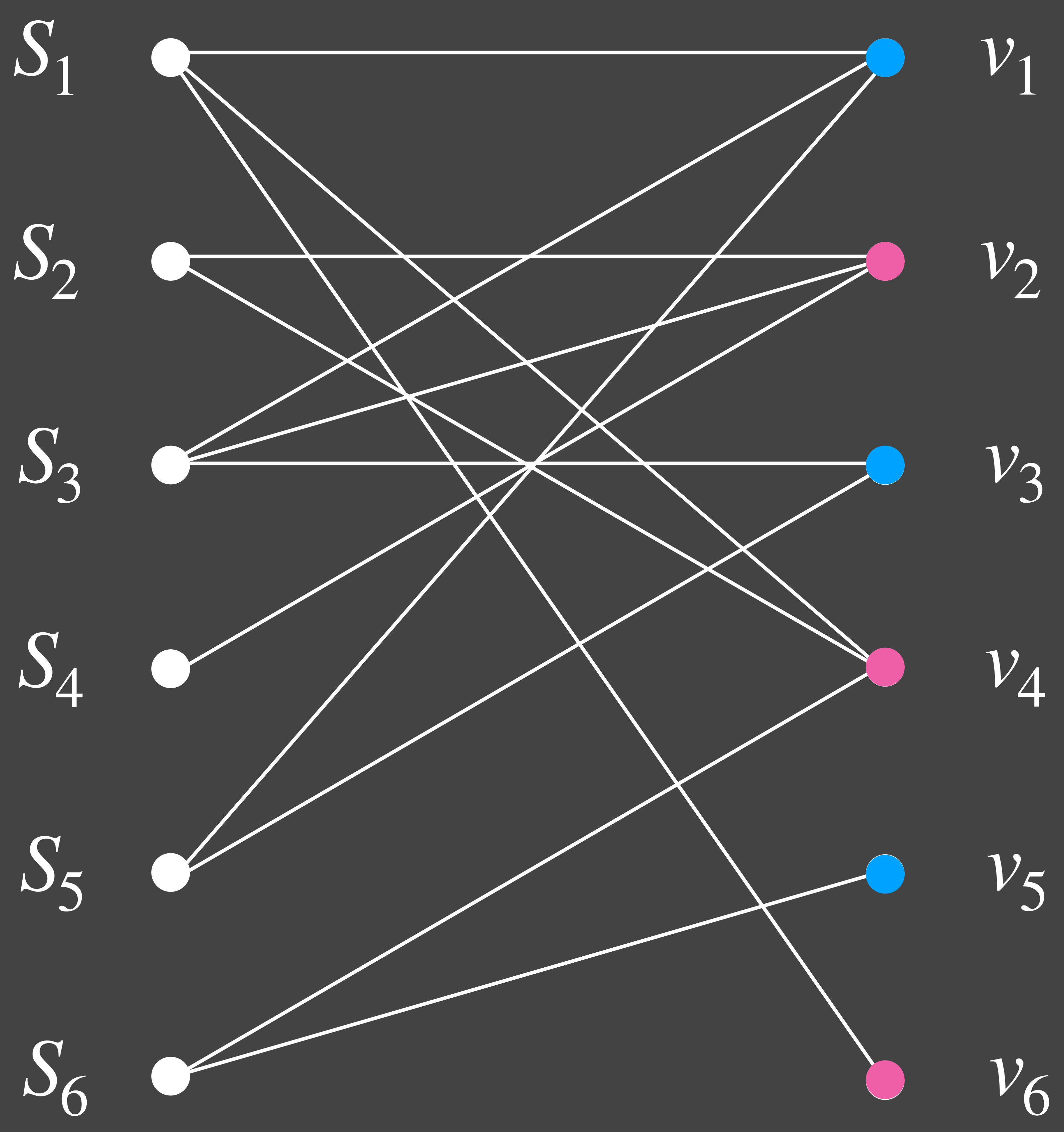
Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v^t uncovered, then $E[\Delta\Phi] \leq -\Omega\left(\frac{1}{k}\right)$.

Φ only deceases during pink steps (so with prob. 1/2), but still $E[\Delta\Phi] \leq -\Omega\left(\frac{1}{k}\right)$.

Reduction to LearnOrCover!

Idea! Pretend colored pink (sampled)/blue (adversarial) on arrival.



Reduction to LearnOrCover!

@ time t:

If v^t pink, feed to LearnOrCover.

If v^t blue, buy arbitrary set to cover.

Recall LearnOrCover proof template:

Claim 1: $\Phi(0) = O(\log mn)$, and $\Phi(t) \geq 0$.

Claim 2: If v^t uncovered, then $E[\Delta\Phi] \leq -\Omega\left(\frac{1}{k}\right)$.

Φ only deceases during pink steps (so with prob. 1/2), but still $E[\Delta\Phi] \leq -\Omega\left(\frac{1}{k}\right)$. 🥳

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

➡ Part III — Random Order Online Set Cover

Conclusion

Talk Outline

Intro

Part I — Online/Dynamic Submodular Cover

Part II — Application: Block-Aware Caching

Part III — Random Order Online Set Cover

➔ Conclusion

My Amazing Collaborators (so far!)

My Family

Thanks!