Streaming Algorithms for Submodular Function Maximization

Chandra Chekuri   Shalmoli Gupta
Kent Quanrud

University of Illinois at Urbana-Champaign

October 6, 2015

Submodular functions

$f : 2^{\mathcal{N}} \to \mathbb{R}$

if $S \subseteq T \subseteq \mathcal{N}$, and $e \in \mathcal{N} \setminus T$, then

$$f(S+e) - f(S) \geq f(T+e) - f(e)$$

we will abbreviate $f_S(e) \stackrel{\text{def}}{=} f(S+e) - f(S)$

Types of submodular f

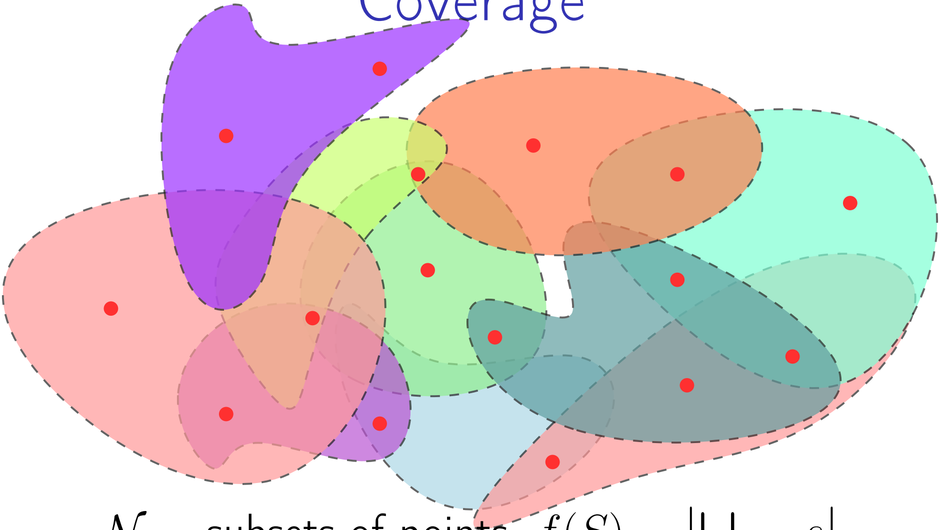
Monotone

$$S \subseteq T \Rightarrow f(S) \leq f(T)$$

Nonnegative

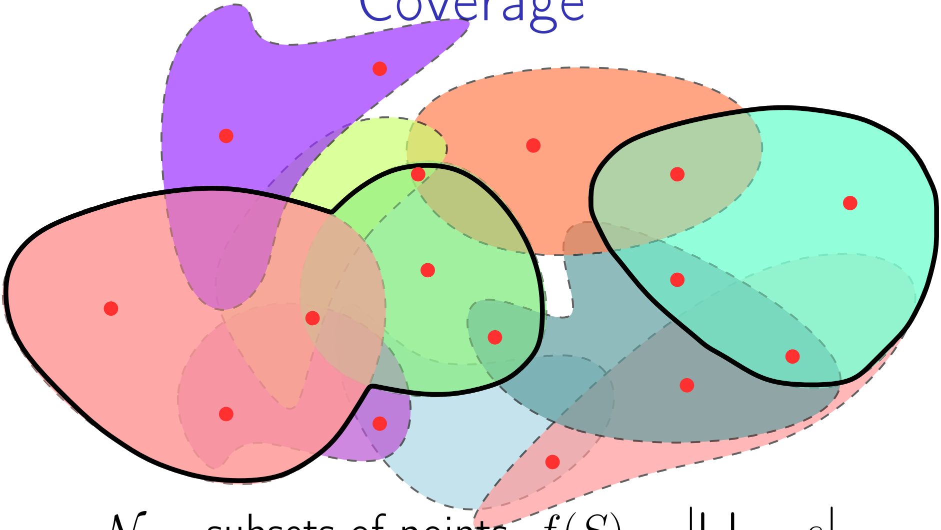
$$f(S) \geq 0$$

Coverage



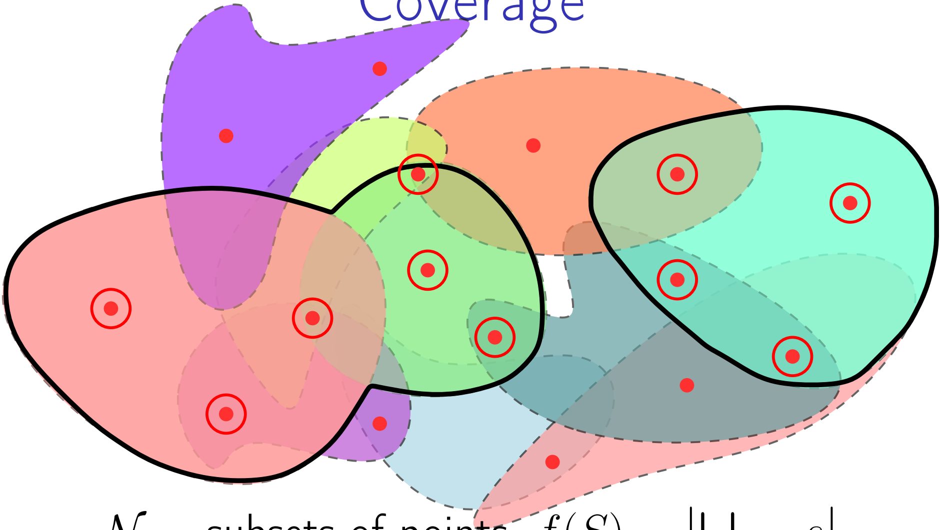
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



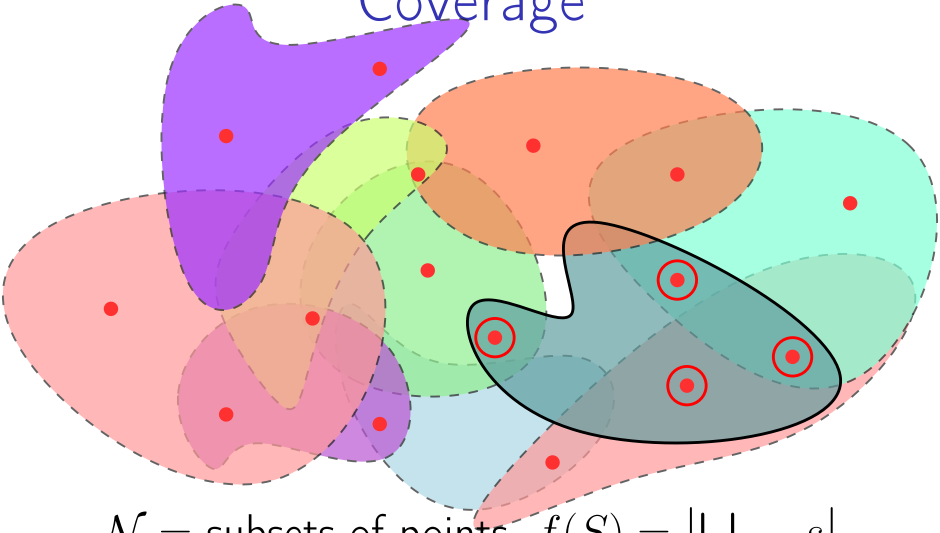
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



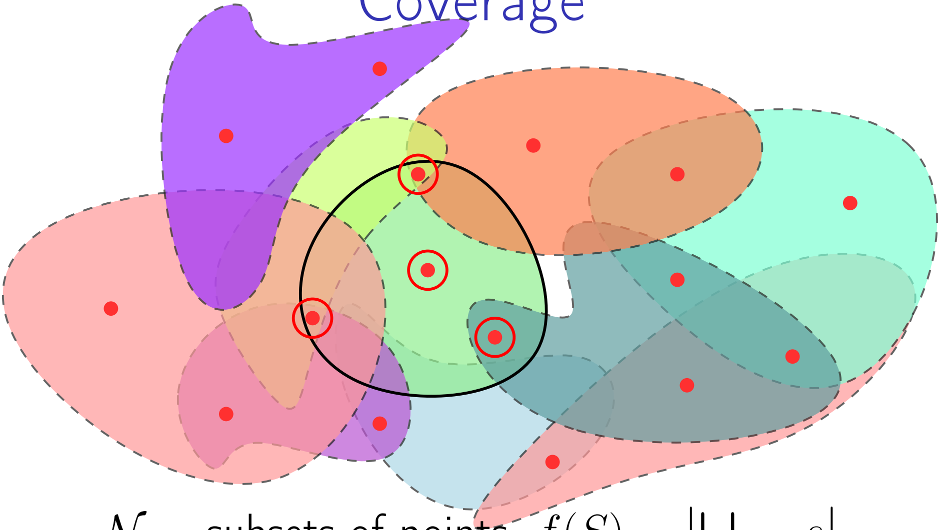
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



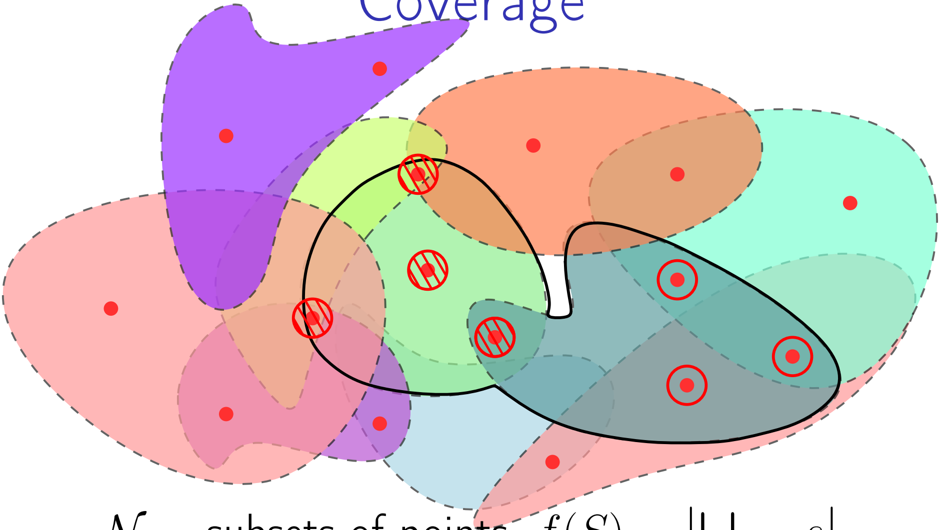
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s\in S} s\right|$

Coverage



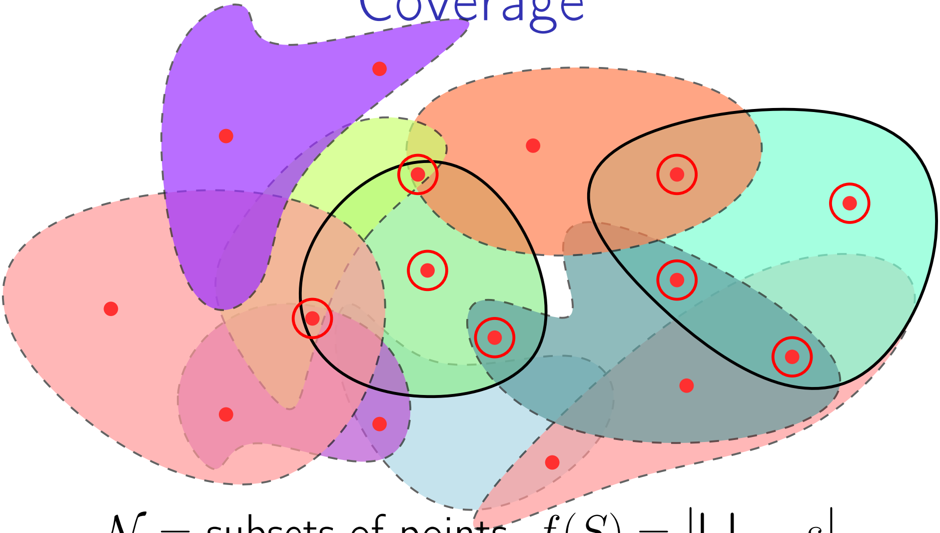
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



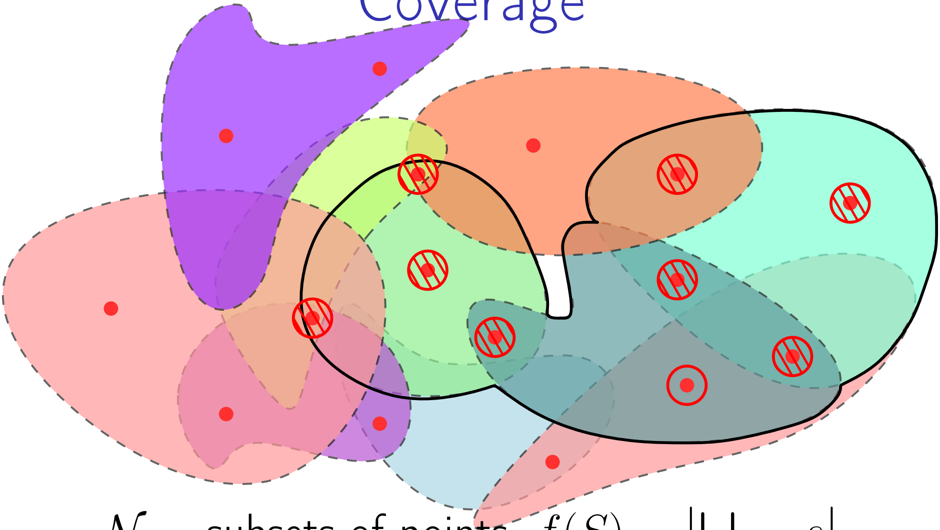
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



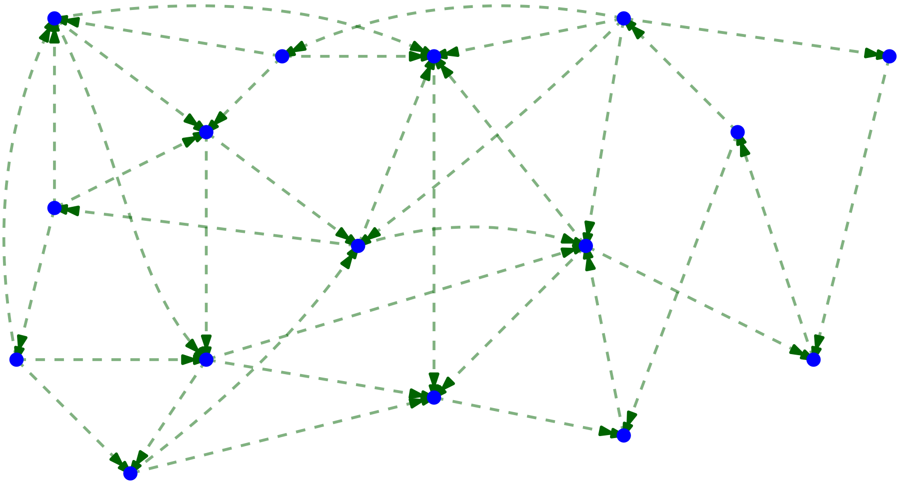
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Coverage



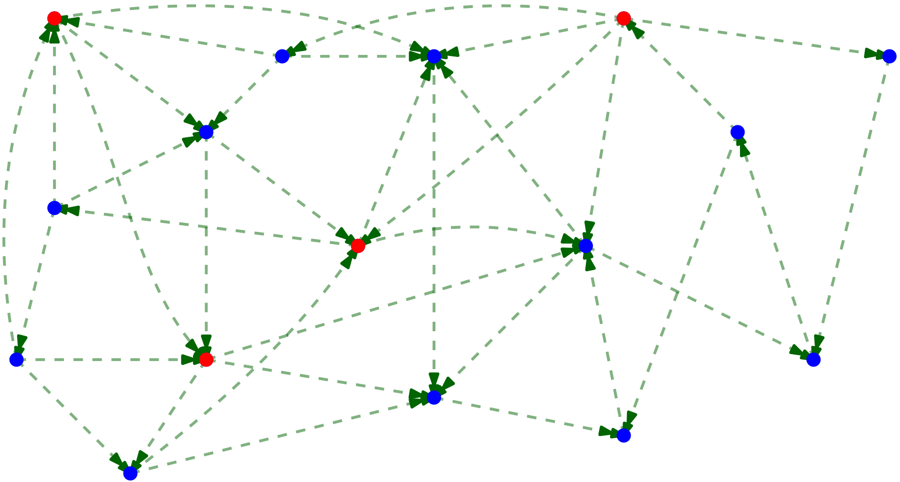
$\mathcal{N}$ = subsets of points, $f(S) = \left|\bigcup_{s \in S} s\right|$

Directed edge cuts



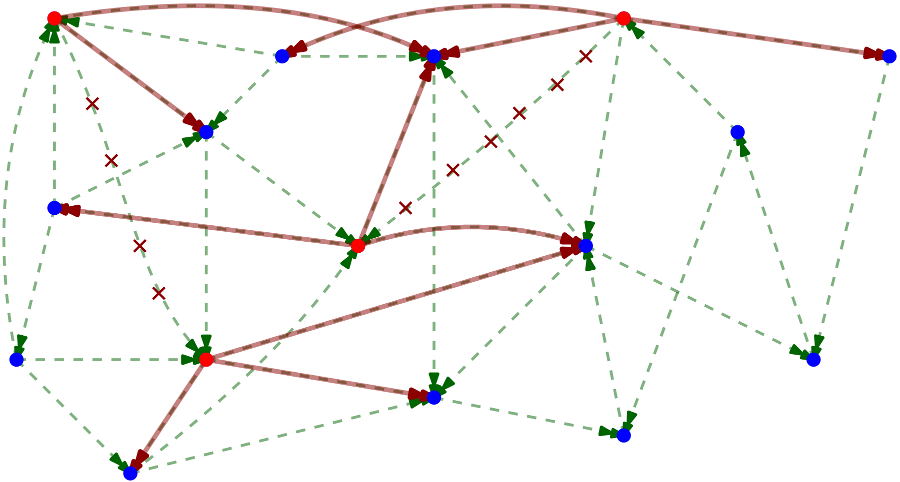
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



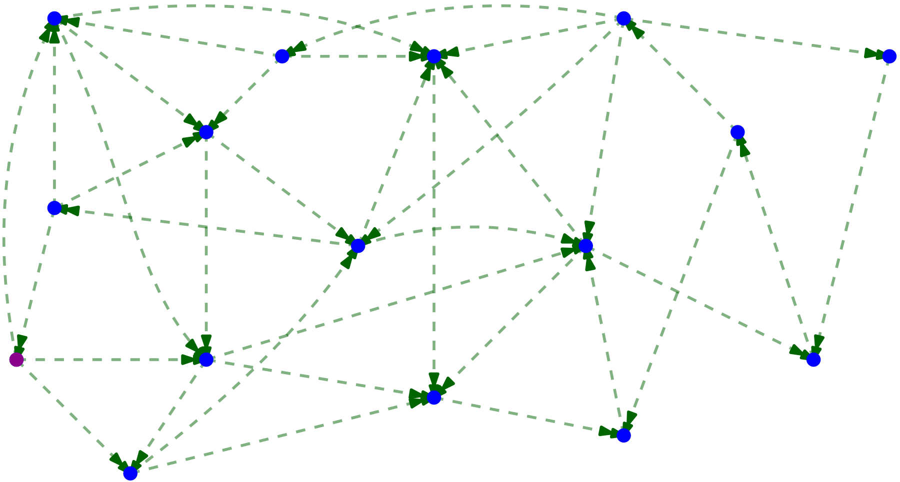
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



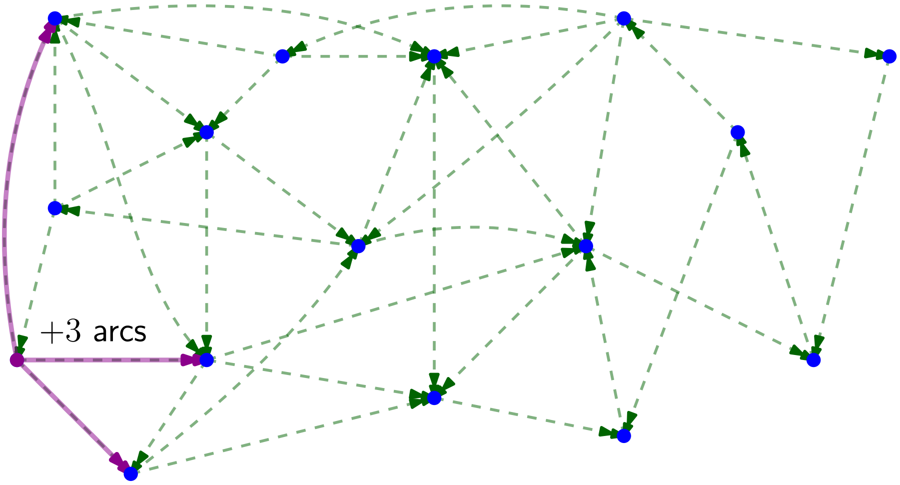
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



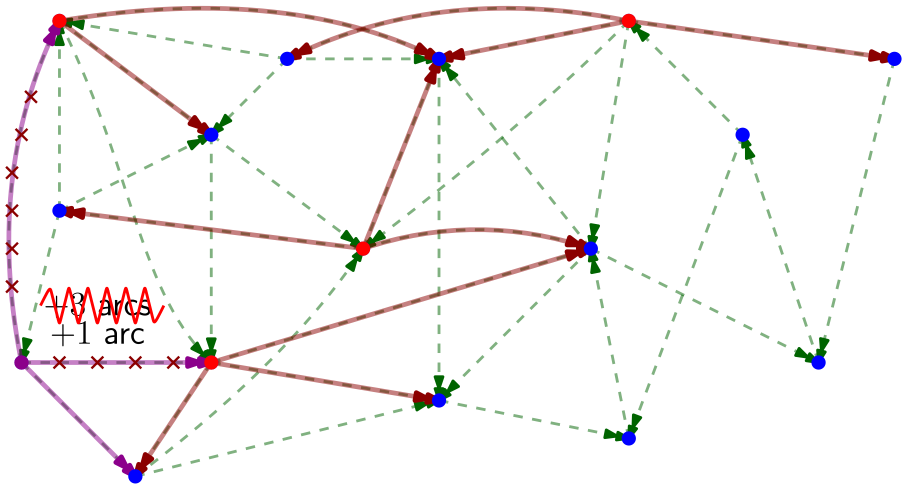
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



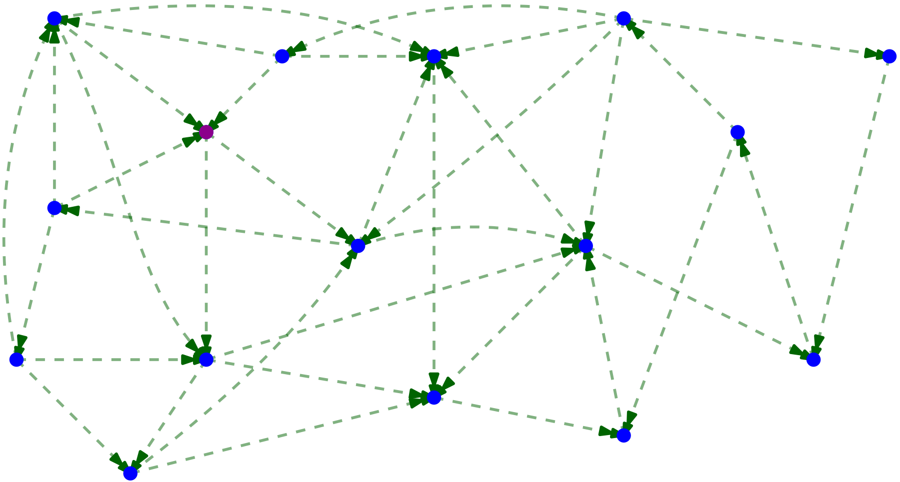
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



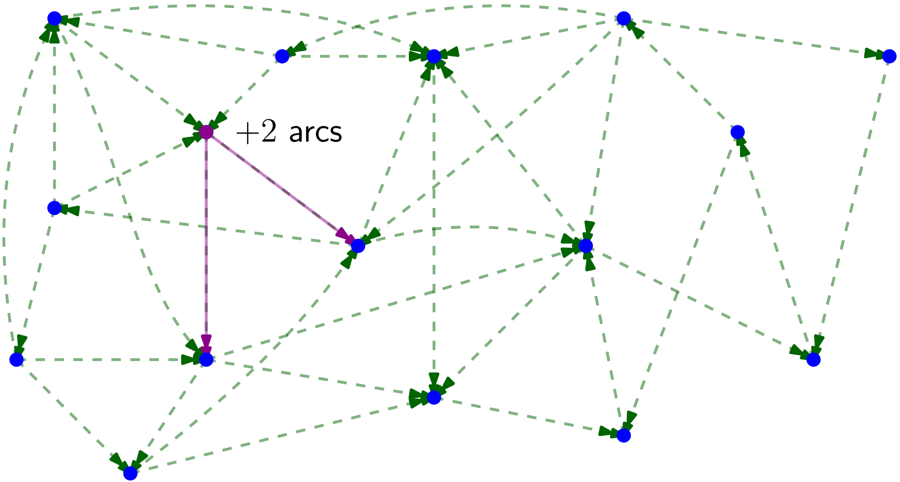
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



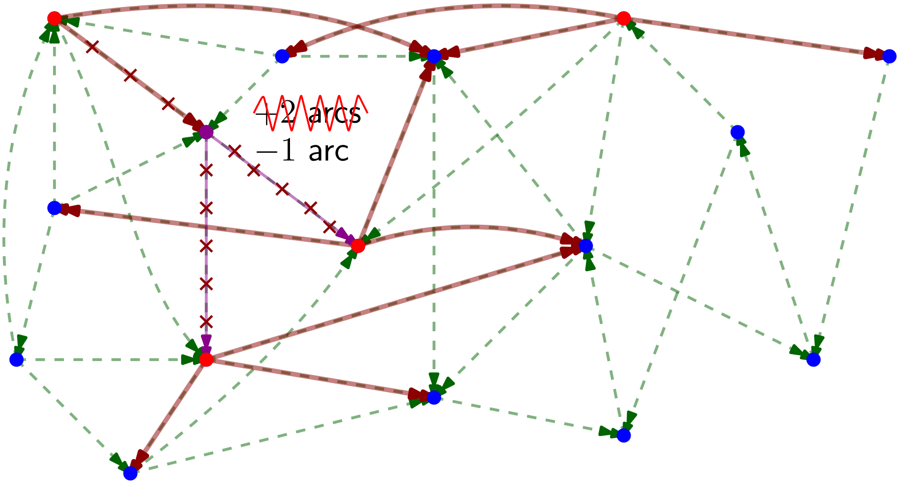
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Directed edge cuts



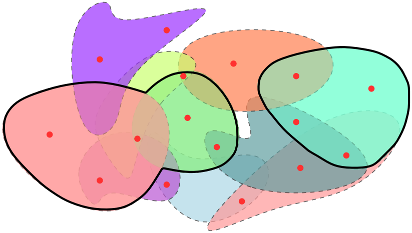
$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

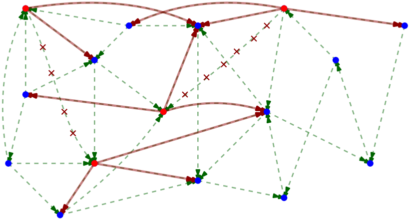
Directed edge cuts



$$\mathcal{N} = \mathcal{V},\ f(S) = |\mathcal{E}(S, \mathcal{V} \setminus S)|$$

Canonical problem





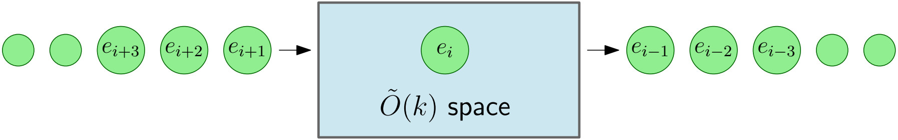
Pick (up to) k elements $e_1, \ldots, e_k \in \mathcal{N}$ maximizing $f(\{e_1, \ldots, e_k\})$

Streaming model

$\mathcal{N} = \{e_1, e_2, \ldots\}$ presented one at a time

Arbitrary order

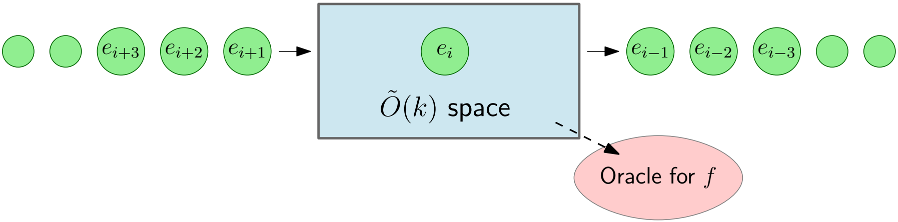
Our main constraint is **space** (ideally, $\tilde{O}(k)$)



Oracle model

Black box access for:

(a) Evaluating $f(S)$



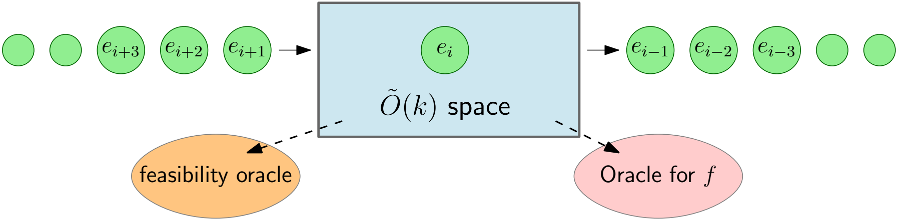
Oracle model

Black box access for:

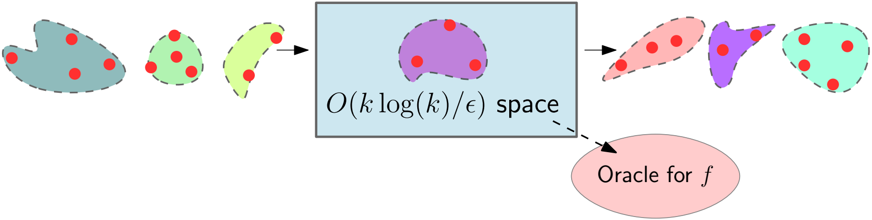
(a) Evaluating $f(S)$

(b) Checking if S is feasible

(for combinatorial constraints)



Monotone f in streams



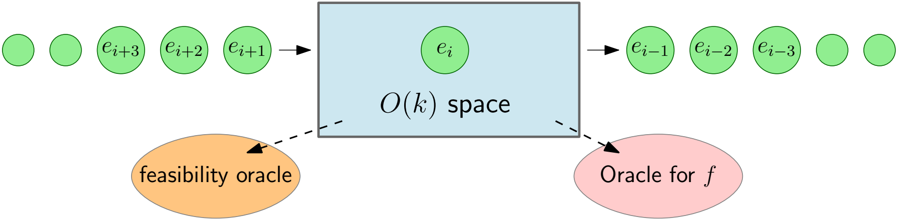
Constraint: cardinality

Approximation ratio: $\frac{1}{2} - \epsilon$

Badanidiyuru, Mirzasoleiman, Karbasi, and Krause

KDD 2014

Monotone f in streams

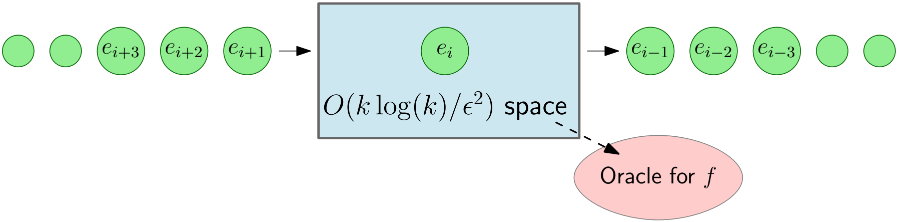


Constraint: Matroids, matchings, matroid intersection

Approximation ratio: $\frac{1}{4p}$ for p matroids

Chakrabarti and Kale IPCO 2014

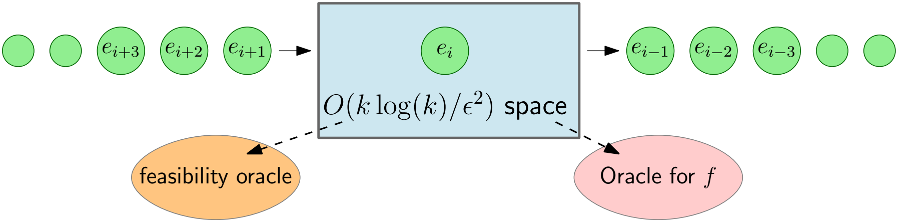
Nonnegative f in streams (our result)



Constraint: Cardinality

Approximation ratio: $\frac{1-\epsilon}{2+e}$

Nonnegative f in streams (our result)

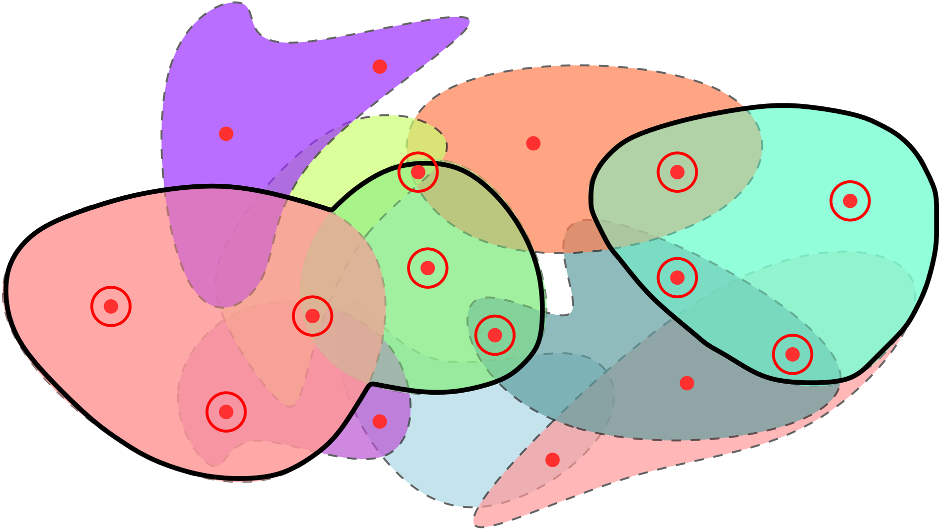


Constraint: p-matchoid $\mathcal{M} = (\mathcal{N}, \mathcal{I})$

Approximation ratio: $\Omega\left(\frac{1}{p}\right)$

	Monotone	*Nonnegative*
Cardinality	$\frac{1-\epsilon}{2}$	$\frac{1-\epsilon}{2+\epsilon}$
p *matroids*	$\frac{1}{4p}$	$\frac{(1-\epsilon)(p-1)}{5p^2-4p}$

Monotone submodular maximization



Nemhauser, Wolsey, Fisher 1978

- `greedy`
- $1 - 1/e$ approximation for cardinality constraint

greedy

$S \leftarrow \emptyset$

`for` $i = 1, \ldots, k$

$\quad e_i \leftarrow \arg\max_{e \in \mathcal{N}} f_S(e)$

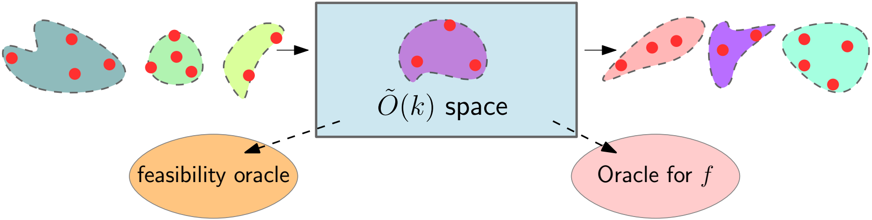
$\quad S \leftarrow S + e_i$

$\quad \mathcal{N} \leftarrow \mathcal{N} - e_i$

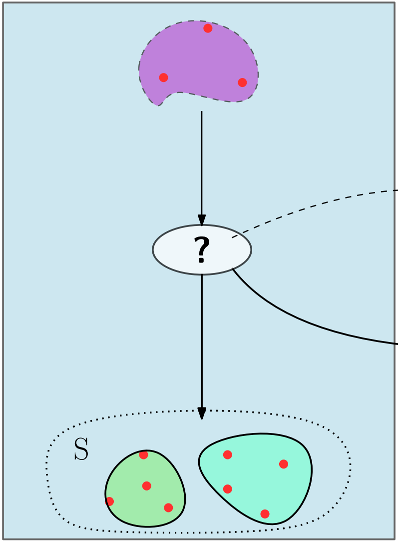
`return` S

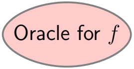
(recall $f_S(e) = f(S + e) - f(S)$)

Monotone submodular maximization *in streams*



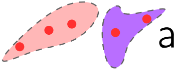
Setup





You have a running solution S. $(|S| \leq k)$

The stream gives you



an element e.

Should you add e to S?

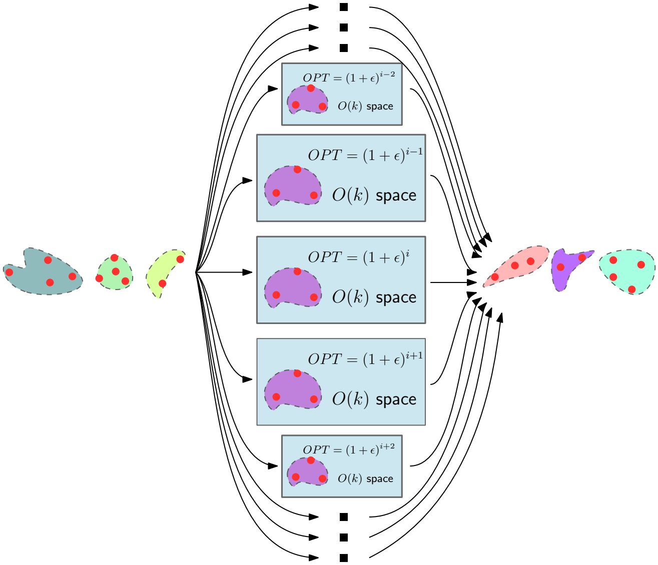
Thresholding

Badanidiyuru, Mirzasoleiman, Karbasi, and Krause

```
if |S| < k
    if f_S(e) ≥ OPT/2k
        S ← S + e
```

$$\texttt{if}\ |S| < k$$
$$\quad \texttt{if}\ f_S(e) \geq \mathrm{OPT}/2k$$
$$\qquad S \leftarrow S + e$$

Guessing OPT



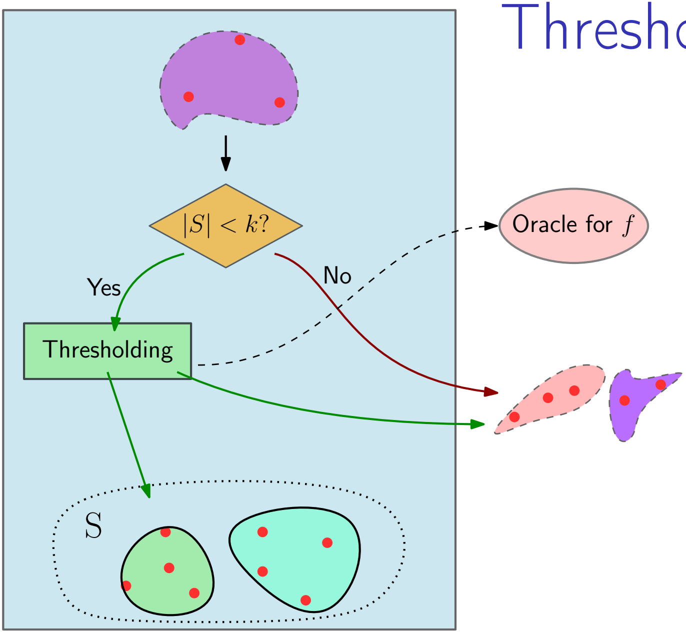
Thresholding

Badanidiyuru, Mirzasoleiman, Karbasi, and Krause

```
if |S| < k
    if f_S(e) ≥ OPT/2k
        S ← S + e
```

$$\texttt{if}\ \ |S| < k$$
$$\quad \texttt{if}\ \ f_S(e) \geq \mathrm{OPT}/2k$$
$$\qquad S \leftarrow S + e$$

Thresholding

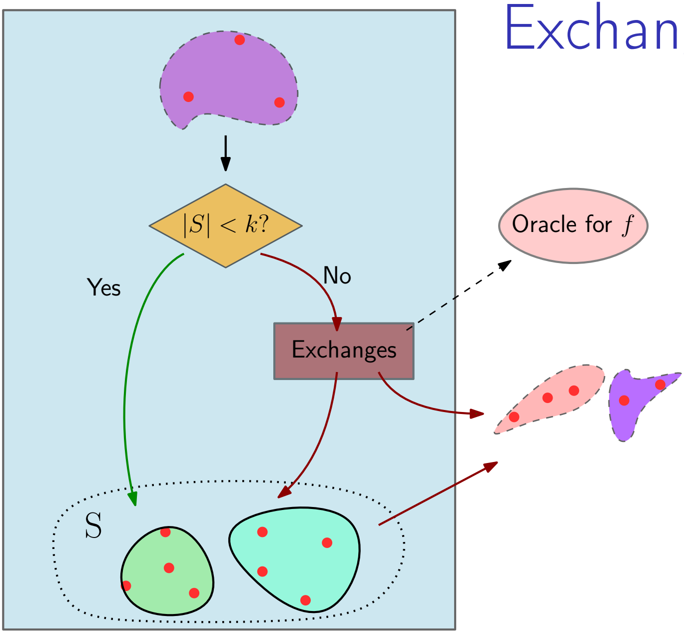


Exchange-based algorithm

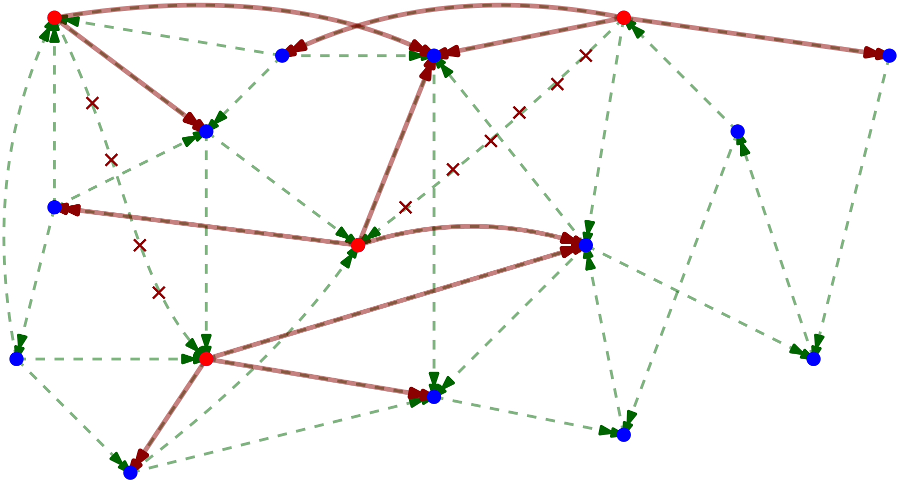
Chakrabarti and Kale

`if` $|S| < k$ `then` $S \leftarrow S + e$
`else`
  `if` $\exists d \in S$ `s.t. exchanging` d `for` e
    `is a ‘‘good enough’’ exchange`
  `then` $S \leftarrow S - d + e$

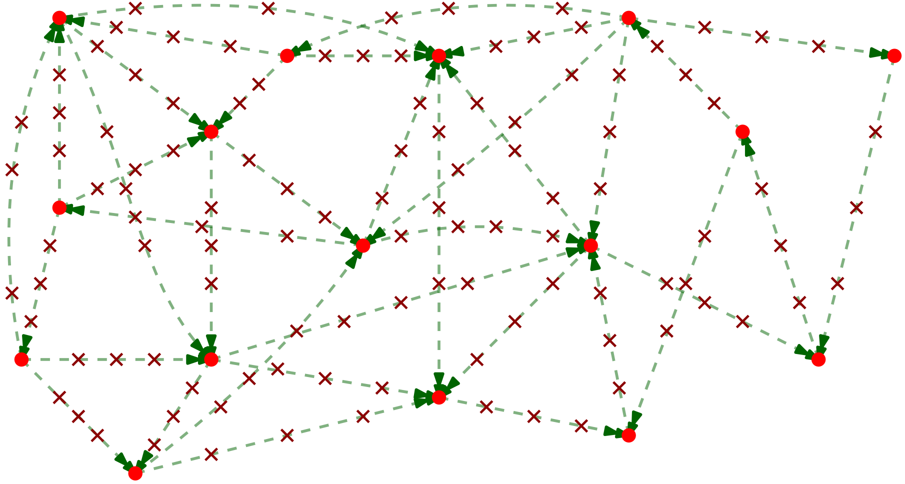
Exchanges



Nonnegative submodular maximization



greedy does not work



Something like greedy works

Gupta, Roth, Schoenebeck, Talwar WINE 2010

- `iterated-greedy`
- $\Omega(1/p)$ approximation for p systems

Buchbinder, Feldman, Naor, Schwartz SODA 2014

- `randomized-greedy`
- $1/e$ approximation for cardinality

`greedy`?

Let S be `greedy` solution, and T an optimal solution

`greedy` gets you

$$f(S) \geq c\, f(S \cup T) \quad \text{for some constant } c$$

without invoking monotonicity

if f is monotone, then

$$f(S \cup T) \geq f(T) = \mathrm{OPT}$$

If f is not monotone, then $f(S \cup T) \geq$ what?

Randomization lemma

if S *is a random set with*

$$\boldsymbol{P}[e \in S] \leq p$$

for all e, *then*

$$\boldsymbol{E}[f(S \cup T)] \geq (1-p) f(T).$$

Buchbinder, Feldman, Naor, Schwartz

randomized-greedy

$S \leftarrow \emptyset$

for $i = 1, \ldots, k$

  let $e_1, \ldots, e_k$ maximize $f_S(e)$

  pick $e_j \in \{e_1, \ldots, e_k\}$ randomly

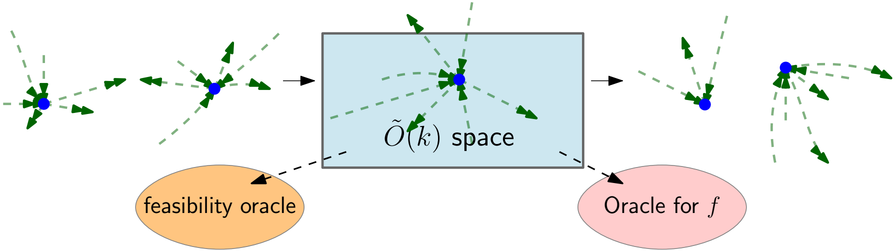
  $S \leftarrow S + e_j$

  $\mathcal{N} \leftarrow \mathcal{N} - e_j$

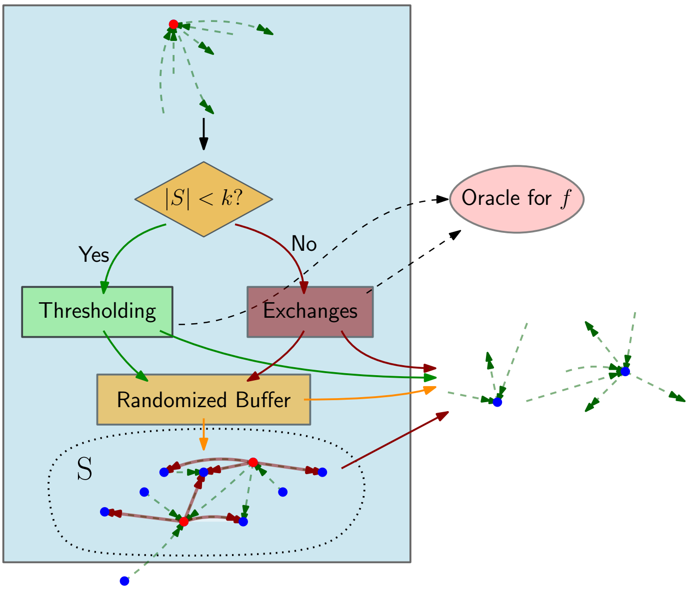
return S

Buchbinder, Feldman, Naor, Schwartz SODA 2014

Nonnegative submodular maximization in streams



Randomized-Streaming-Greedy



Randomized-Streaming-Greedy

```
S ← ∅,  B ← ∅
for each element e in the stream
  if Is-Good(S,e)
    B ← B + e
  if B is full // |B| = Θ(k)
    pick e ∈ B randomly
    add or exchange e into S
    clean up B
S' ← Offline(f,B)
return better of S and S'
```

Is-Good(S, e)

```
if |S| < k
  if f_S(e) ≥ Ω(OPT/k)
  then return ‘‘GOOD’’
else // |S| = k
  if ∃d ∈ S such that
      f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
  then return ‘‘GOOD’’
return ‘‘BAD’’
```

Magic value $\nu(f, S, d)$

```
if ∃d ∈ S such that
   f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
then return ‘‘GOOD’’
```

$\nu(f, S, d)$ should:

Magic value $\nu(f, S, d)$

```
if ∃d ∈ S such that
   f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
then return ‘‘GOOD’’
```

$\nu(f, S, d)$ should:

- Account for the value originally added by d to S.

Magic value $\nu(f, S, d)$

```
if ∃d ∈ S such that
   f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
then return "GOOD"
```

$\nu(f, S, d)$ should:

- Account for the value originally added by d to S.
- Adapt dynamically to changing S.

Magic value $\nu(f, S, d)$

```
if ∃d ∈ S such that
   f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
then return ‘‘GOOD’’
```

$\nu(f, S, d)$ should:

- Account for the value originally added by d to S.
- Adapt dynamically to changing S.
- Ensure that exchanging $S \to S - d + e$ increases $f(S)$ substantially.

Incremental value

Let $S = \{d_1, \ldots, d_k\}$ in order of insertion

The *incremental value* of d_i is defined as

$$\nu(f, S, d_i) \stackrel{\text{def}}{=} f(d_1, \ldots, d_i) - f(d_1, \ldots, d_{i-1}).$$

Incremental value

Let $S = \{d_1, \ldots, d_k\}$ *in order of insertion*

The incremental value of d_i *is defined as*

$$\nu(f, S, d_i) \stackrel{\text{def}}{=} f(d_1, \ldots, d_i) - f(d_1, \ldots, d_{i-1}).$$

Property 1: When we add an element e to the running solution $S - d$,

$$\nu(f, S - d + e, e) = f_{S-d}(e)$$

Incremental value

Let $S = \{d_1, \ldots, d_k\}$ *in order of insertion*

The incremental value of d_i *is defined as*

$$\nu(f, S, d_i) \stackrel{\text{def}}{=} f(d_1, \ldots, d_i) - f(d_1, \ldots, d_{i-1}).$$

Property 2: ν telescopes.

$$\sum_{d \in S} \nu(f, S, d) = f(S)$$

Incremental value

Let $S = \{d_1, \ldots, d_k\}$ *in order of insertion*

The incremental value of d_i *is defined as*

$$\nu(f, S, d_i) \stackrel{\text{def}}{=} f(d_1, \ldots, d_i) - f(d_1, \ldots, d_{i-1}).$$

Property 3: For fixed $d \in S$, its incremental value $\nu(f, S, d)$ only increases over the course of the algorithm.

Is-Good(S,e)

```
if |S| < k
   if f_S(e) ≥ Ω(OPT/k)
   then return ''GOOD''
else // |S| = k
   if ∃d ∈ S such that
      f_S(e) ≥ 2ν(f, S, d) + Ω(OPT/k)
   then return ''GOOD''
return ''BAD''
```

Randomized-Streaming-Greedy

```
S ← ∅,  B ← ∅
for each element e in the stream
  if Is-Good(S,e)
    B ← B + e
  if B is full // |B| = Θ(k)
    pick e ∈ B randomly
    add or exchange e into S
    clean up B
S′ ←Offline(f,B)
return better of S and S′
```

Conclusion

Technical result

Constant factor approximations

Nonnegative submodular maximization

1-pass streams

Broad class of combinatorial constraints

Main techniques

Randomized buffer

Greedy w/r/t incremental value

Post-processing

Open questions

Modeling

Lower bounds

thanks