Quotient Sparsification for Submodular Functions

Kent Quanrud, Purdue

Graph (2-)cuts

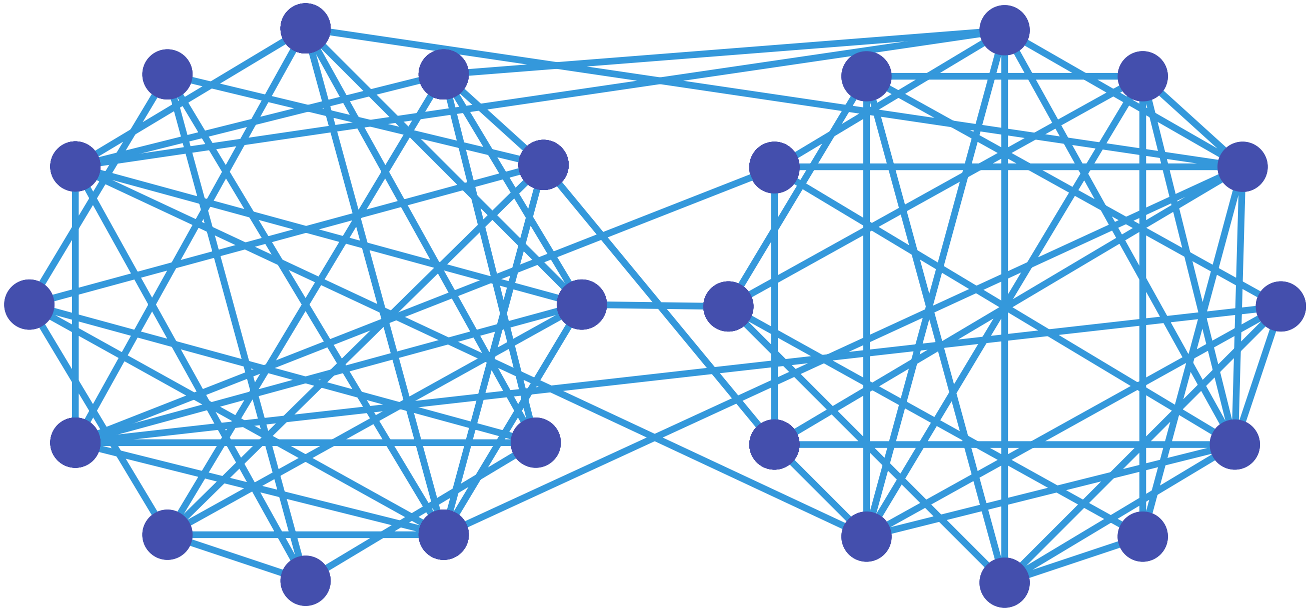
(undirected)

$G=(V,E)$,

$n=|V|$ $m=|E|$

edge weights

$w(e)>0$ for $e\in E$



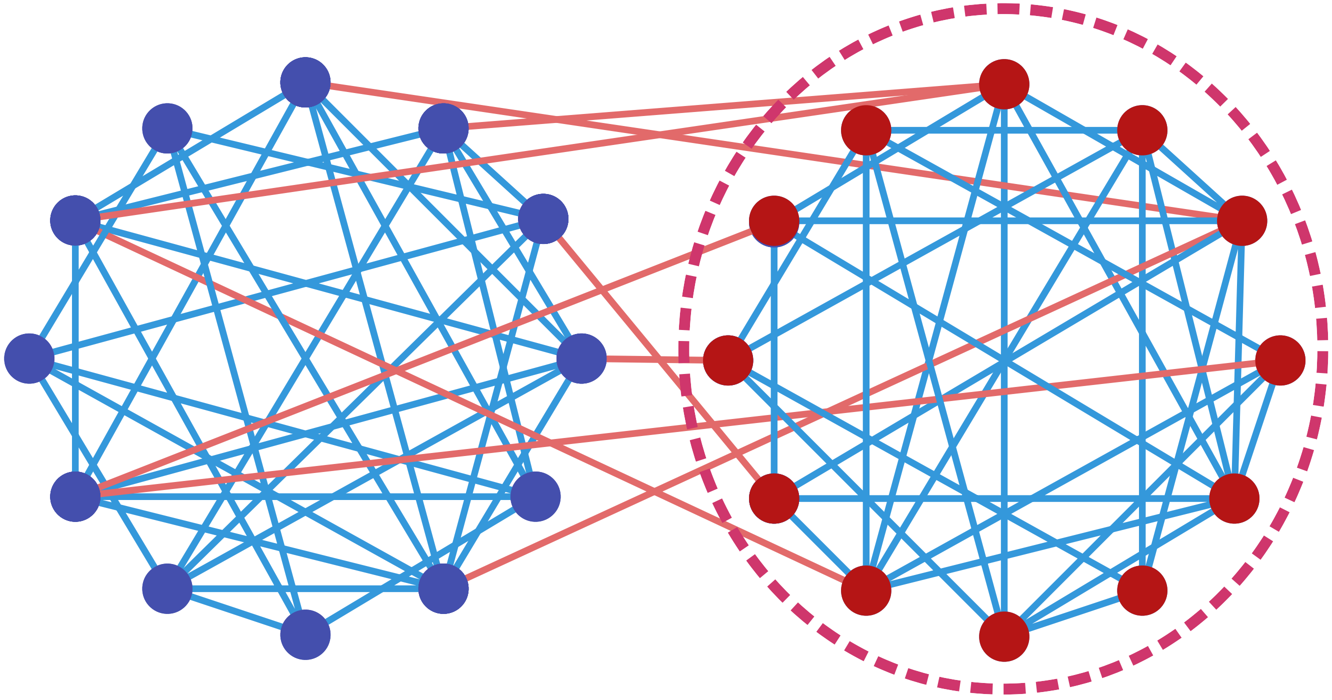
Graph (2-)cuts

(undirected)

$G=(V,E)$, $n=|V|$ $m=|E|$

edge weights

$w(e)>0$ for $e\in E$



$$\partial(S)=\{e=\{u,v\}\in E,\ u\in S,\ v\notin S\}$$

$$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)} w(e)$$

Graph (2-)cuts

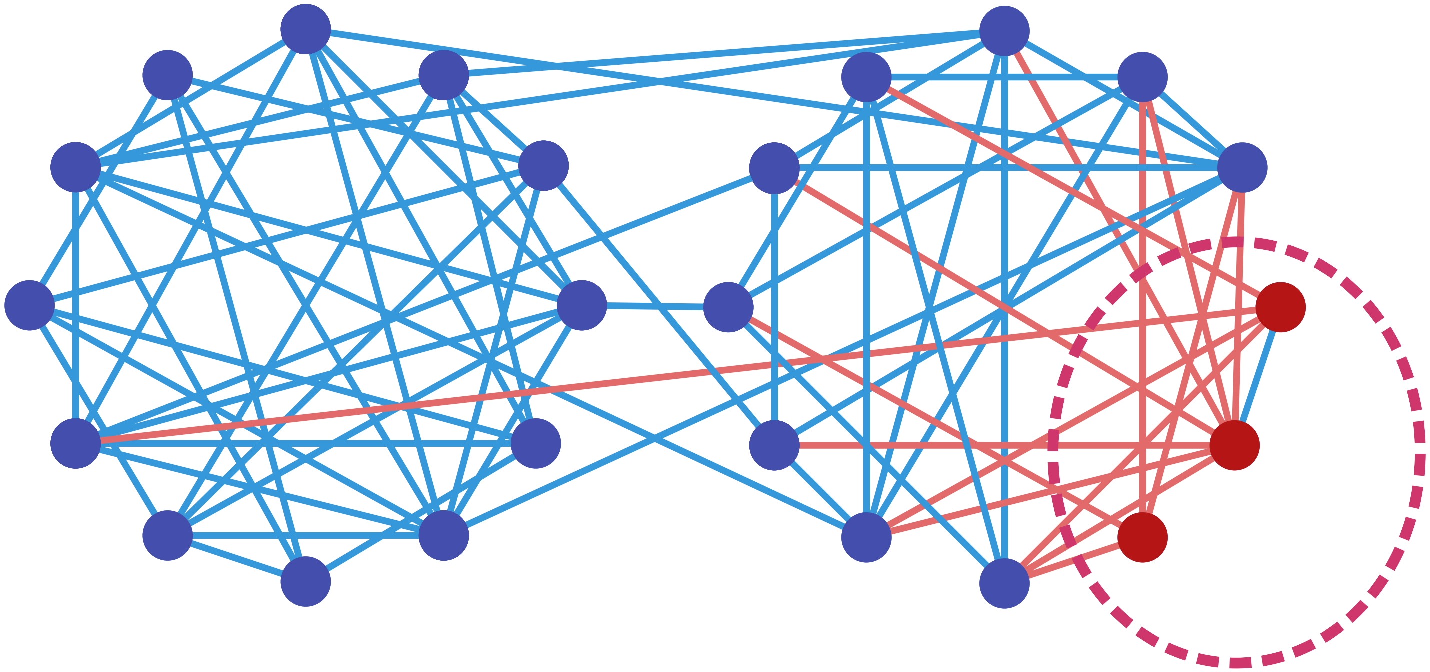
(undirected)

$G=(V,E)$,

$n=|V|$ $m=|E|$

edge weights

$w(e)>0$ for $e\in E$

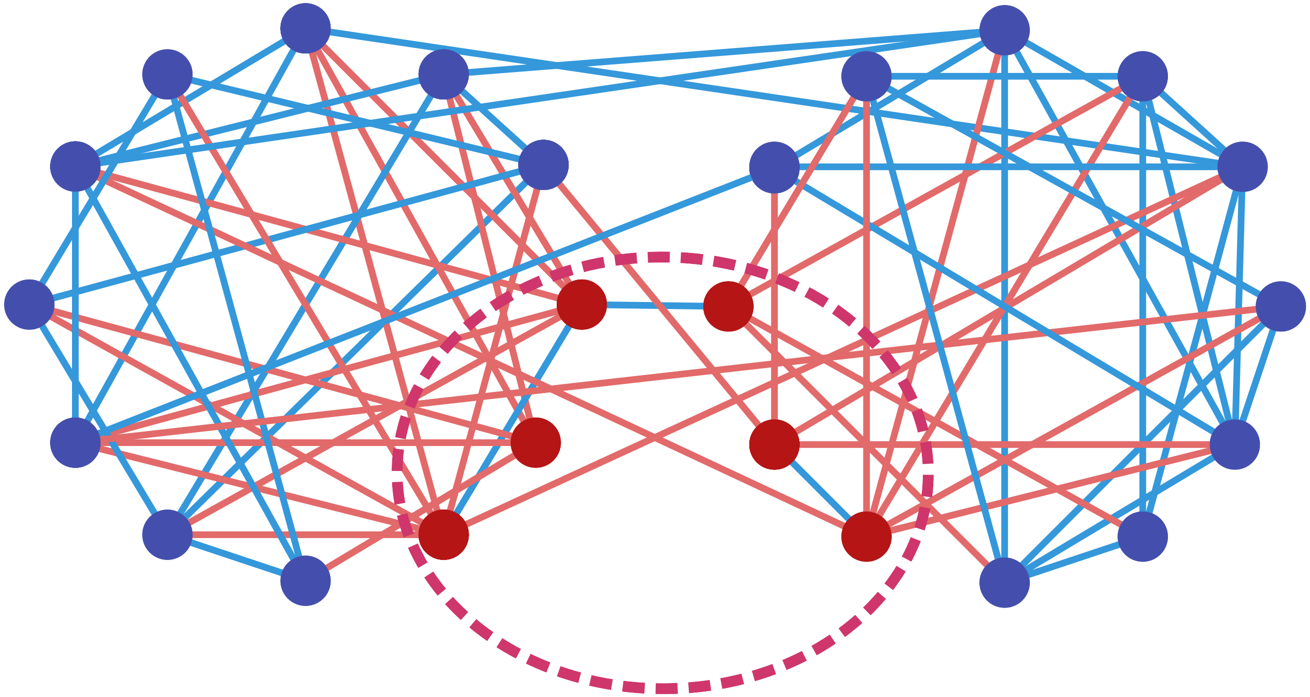


$$\partial(S)=\{e=\{u,v\}\in E,\ u\in S,\ v\notin S\}$$

Graph (2-)cuts

(undirected)
$G=(V,E)$,
$n=|V|$ $m=|E|$

edge weights
$w(e)>0$ for $e\in E$

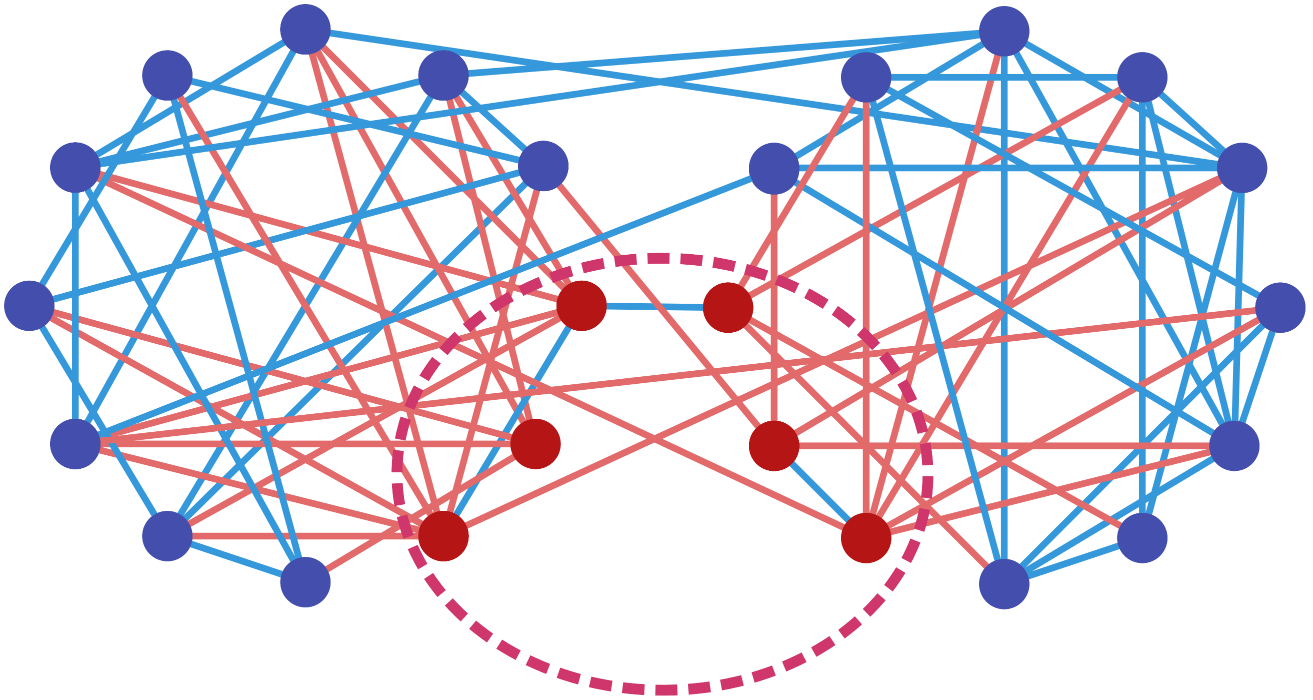


$\partial(S)=\{e=\{u,v\}\in E,\ u\in S,\ v\notin S\}$

Graph (2-)cuts

(undirected)
$G=(V,E)$, $n=|V|$ $m=|E|$

edge weights
$w(e) > 0$ for $e \in E$



$\partial(S) = \{e=\{u,v\} \in E, u \in S, v \notin S\}$

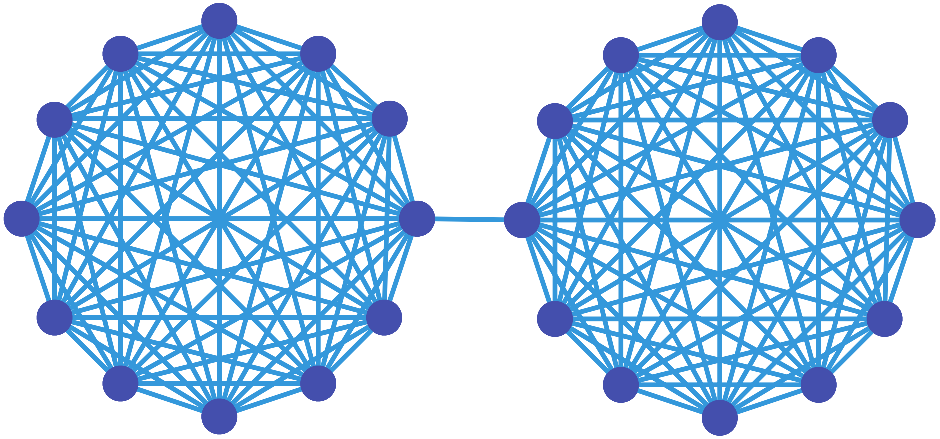
cut values are important: determine max flows, connectivity, balanced separators, expansion, etc.

Graph cut sparsification

Input: $G=(V,E)$ (undirected)

$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$

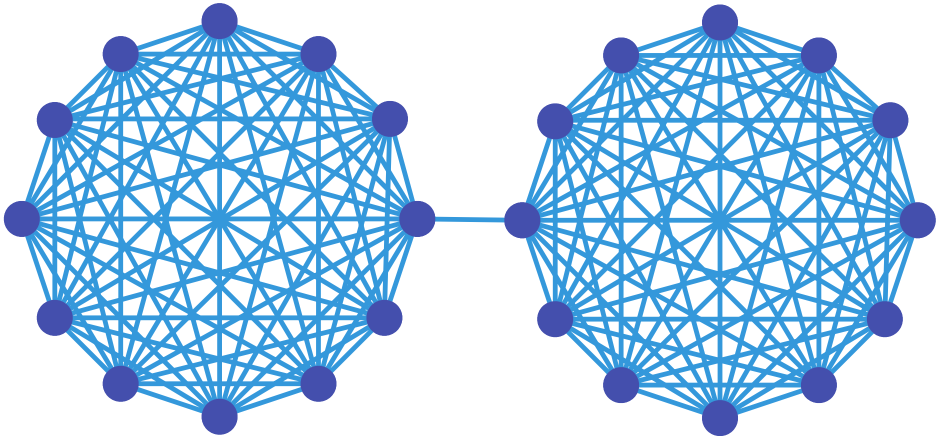


Graph cut sparsification

Input: $G=(V,E)$ (undirected)

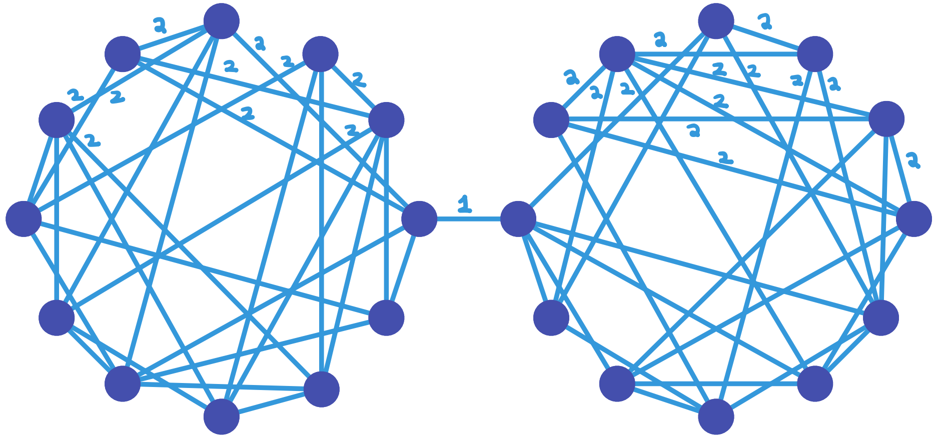
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



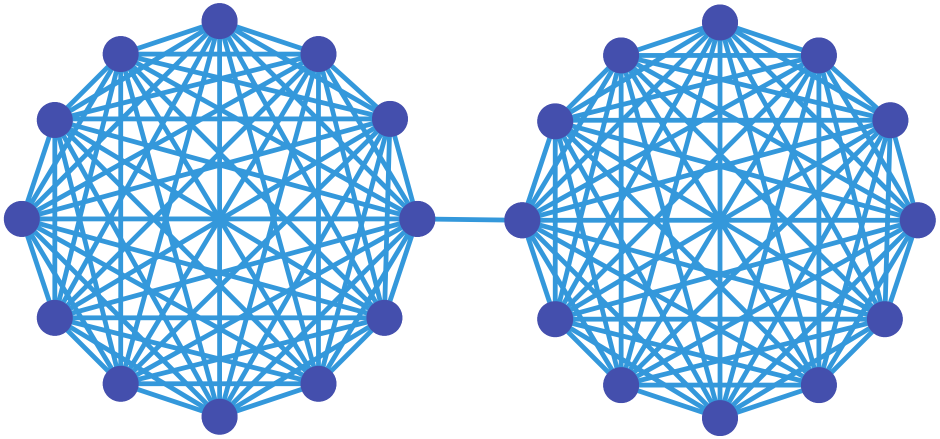
s.t. (a)

(b)

Graph cut sparsification

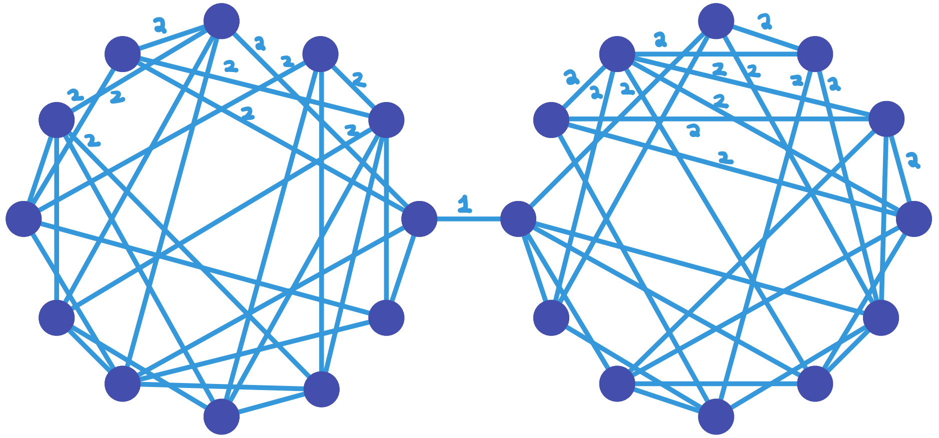
Input: (undirected) $G=(V,E)$ $n=|V|$ $m=|E|$

$w(e)>0$ for $e \in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e \in \tilde{E}$



s.t. (a) $|\tilde{E}|$ small

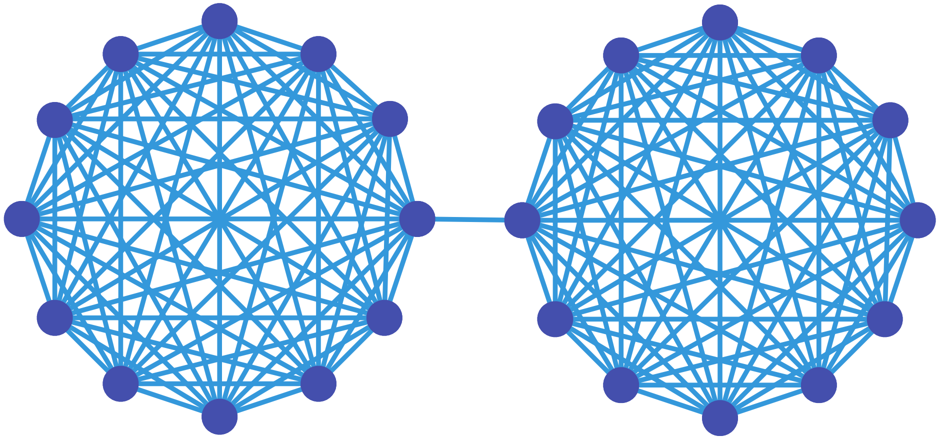
(b)

Graph cut sparsification

Input: (undirected) $G=(V,E)$

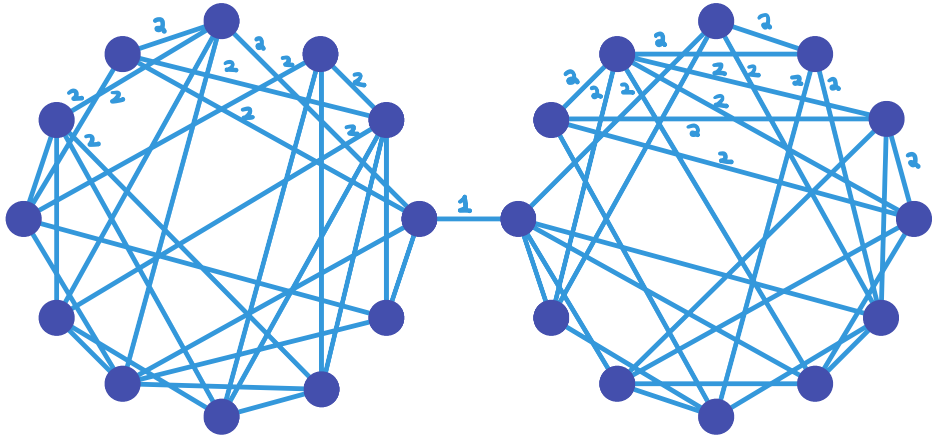
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



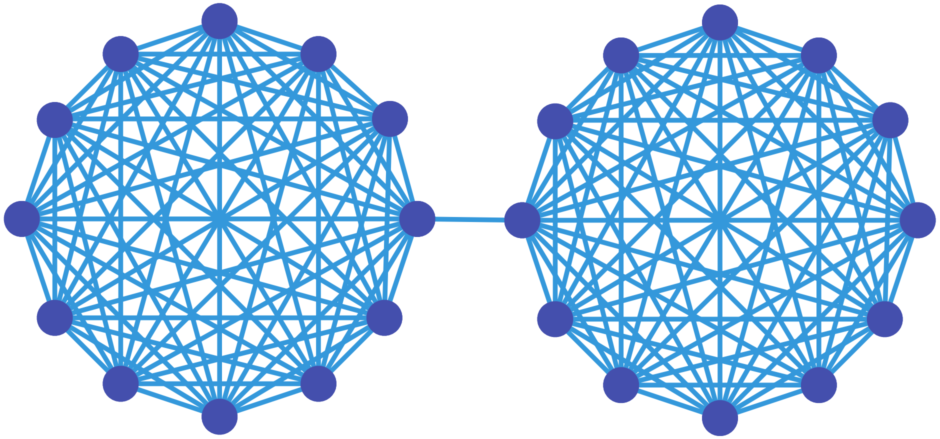
s.t. (a) $|\tilde{E}|$ small

(b) all cuts have similar weight as in G

Graph cut sparsification

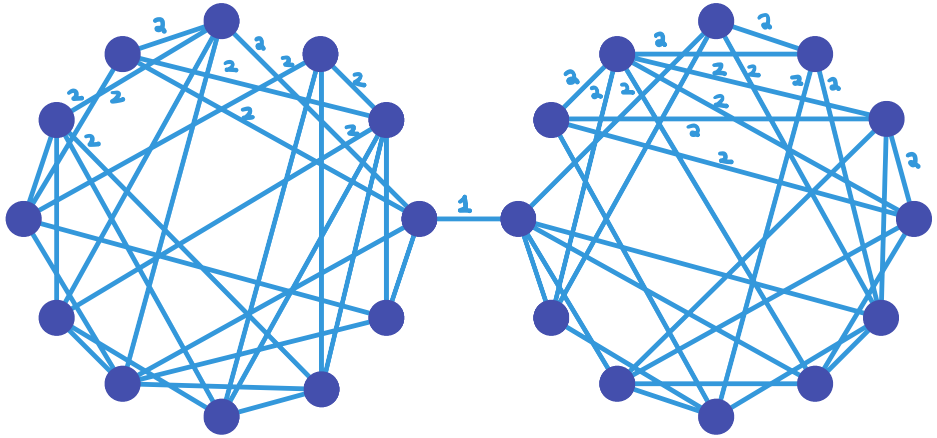
Input: (undirected) $G=(V,E)$ $n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all cuts have similar weight as in G

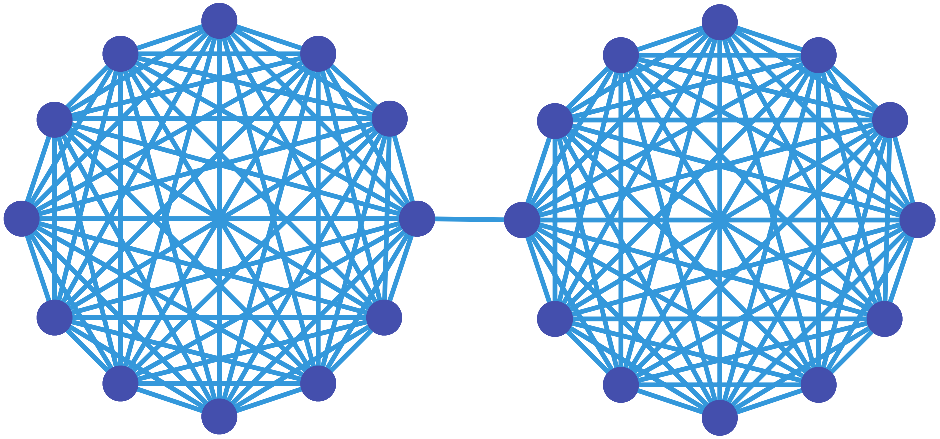
Benczúr, Karger (2002):

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)} \tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)} \tilde{w}(e)$

Graph cut sparsification

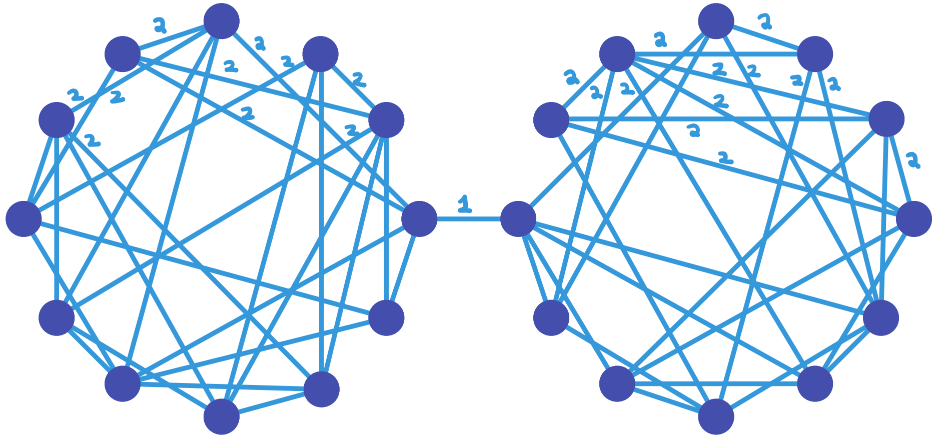
Input: (undirected) $G=(V,E)$ $n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all cuts have similar weight as in G

Benczúr, Karger (2002):

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e)$

Also: spectral [ST04, SS11], $|\hat{E}|=O(n/\varepsilon^2)$ [BSS12], [FHHP11], ~

(edges have many endpoints)

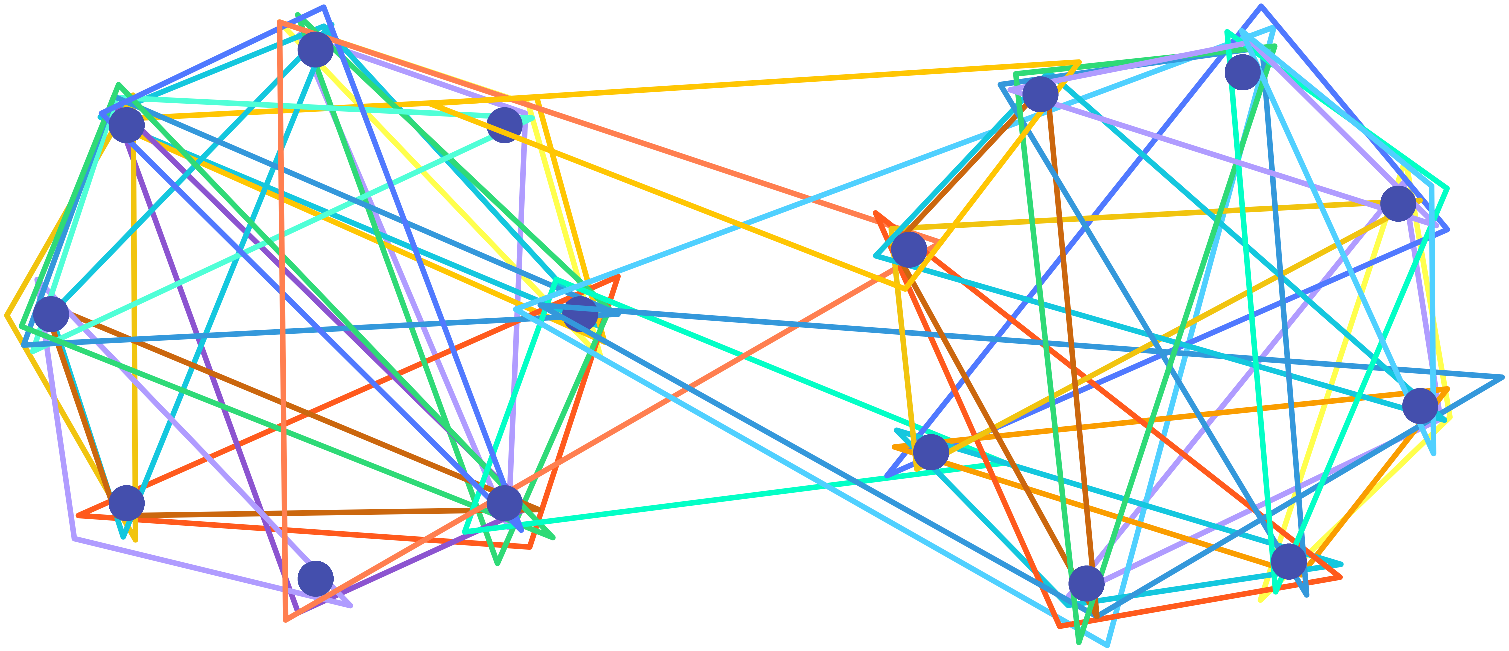
Hypergraph (2-)cuts

edge weights

$G=(V,E)$, $w(e)>0$ for $e\in E$

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$



(edges have many endpoints)

Hypergraph (∂-)cuts

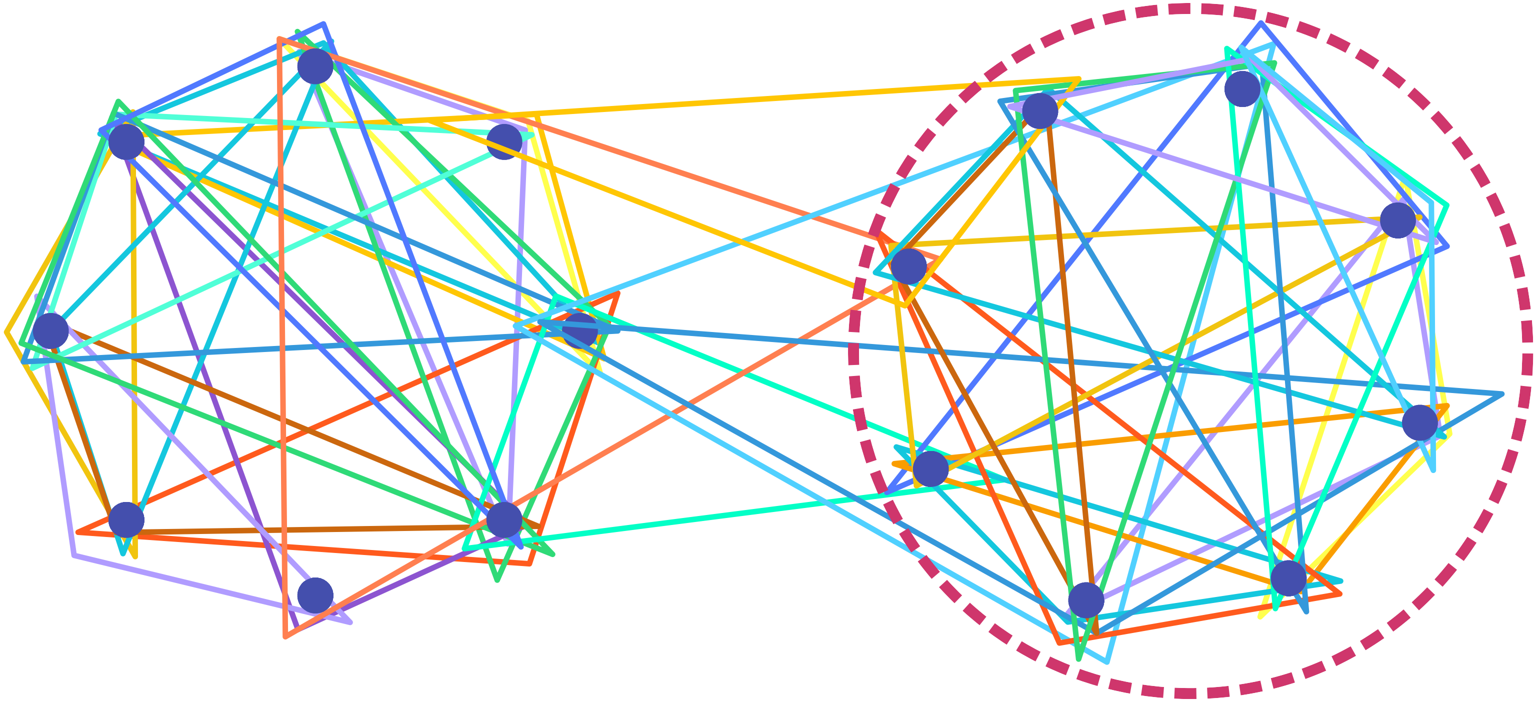
$G=(V,E)$,

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$$\partial(S)=\{e\in E,\ \emptyset\subsetneq e\cap S\subsetneq e\}$$

$$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)}w(e)$$

(edges have many endpoints)

Hypergraph (∂-)cuts

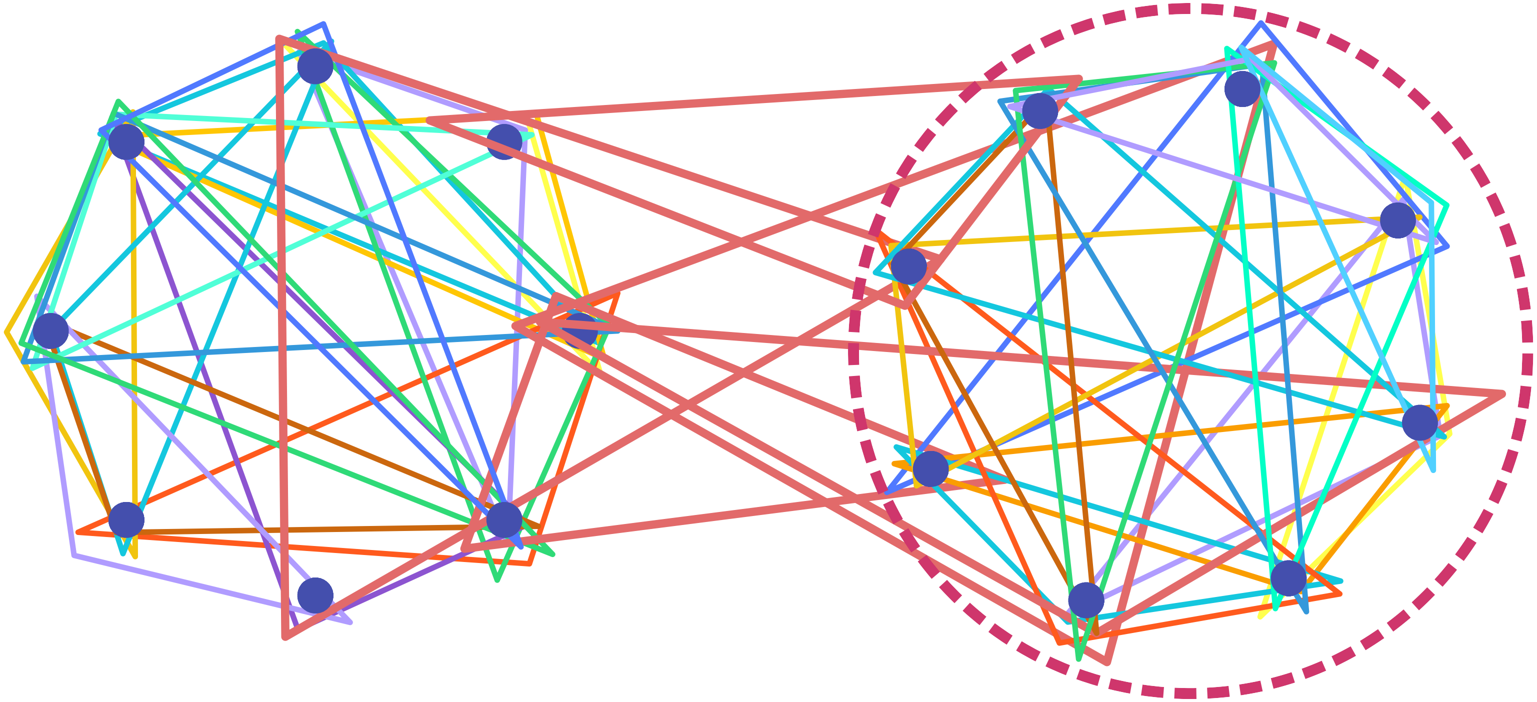
$G=(V,E)$,

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$\partial(S)=\{e\in E,\ \emptyset\subsetneq e\cap S\subsetneq e\}$

$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)}w(e)$

(edges have many endpoints)

Hypergraph (∂-)cuts

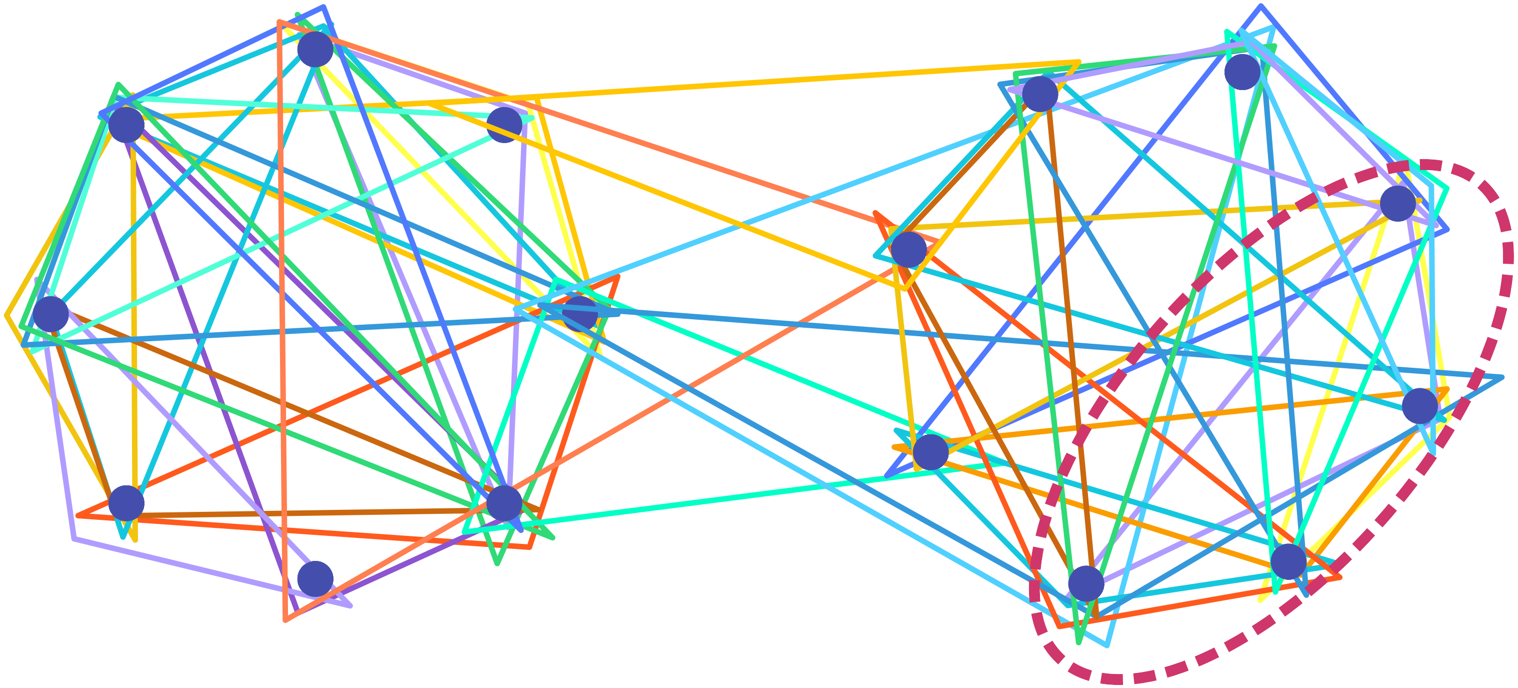
$G=(V,E)$,

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$$\partial(S)=\{e\in E,\ \emptyset\subsetneq e\cap S\subsetneq e\}$$

$$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)}w(e)$$

(edges have many endpoints)

Hypergraph (∂-)cuts

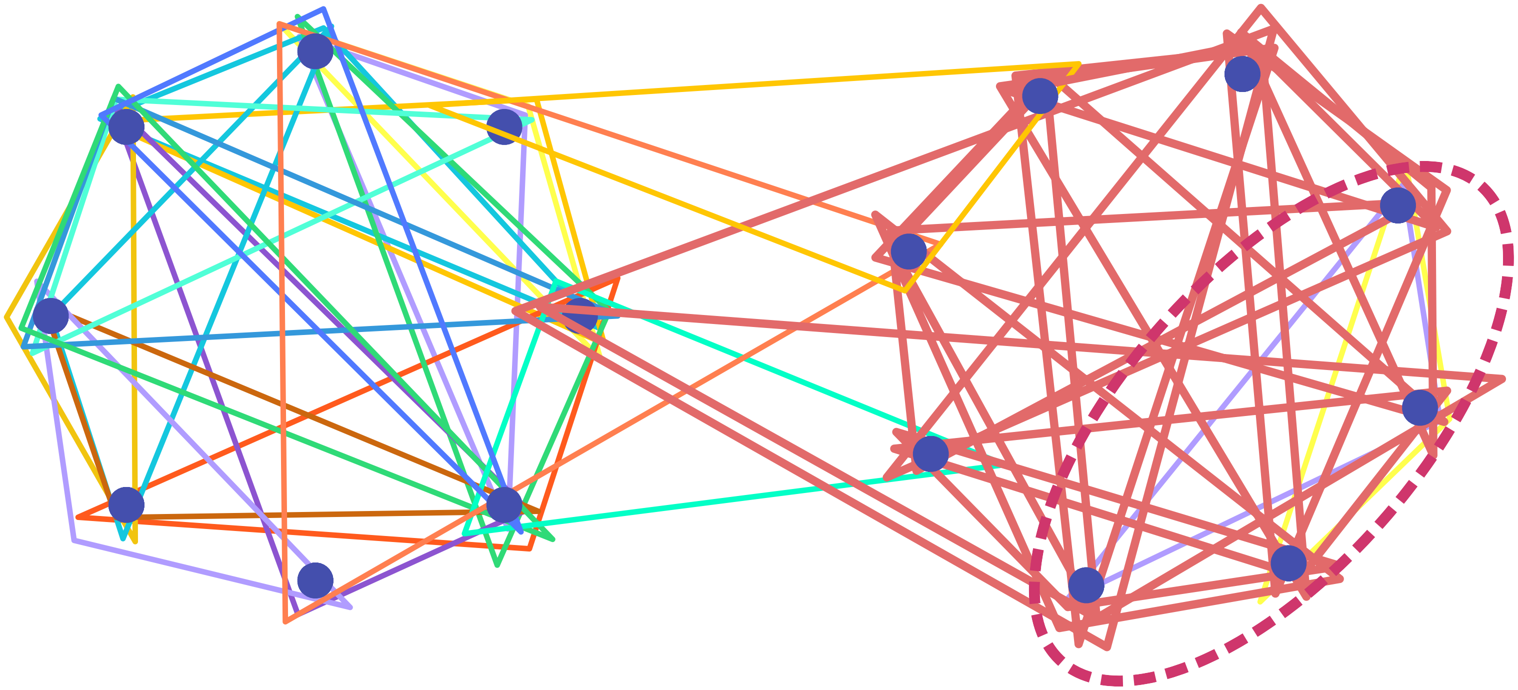
$G=(V,E)$,

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$$\partial(S)=\{e\in E,\ \emptyset\subsetneq e\cap S\subsetneq e\}$$

$$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)}w(e)$$

(edges have many endpoints)

Hypergraph (∂-)cuts

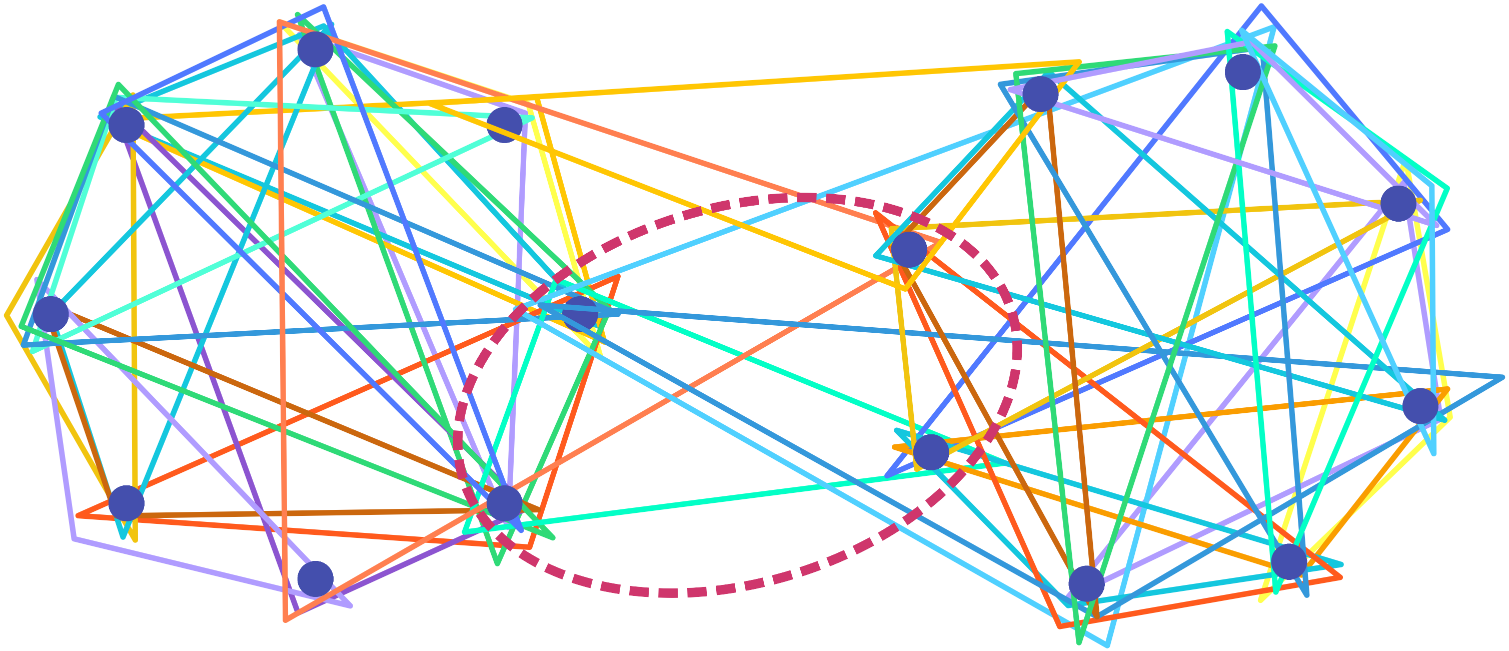
$G=(V,E)$,

$n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$\partial(S)=\{e\in E, \emptyset \subsetneq e\cap S \subsetneq e\}$

$w(\partial(S)) \overset{\text{def}}{=} \sum_{e\in\partial(S)} w(e)$

(edges have many endpoints)

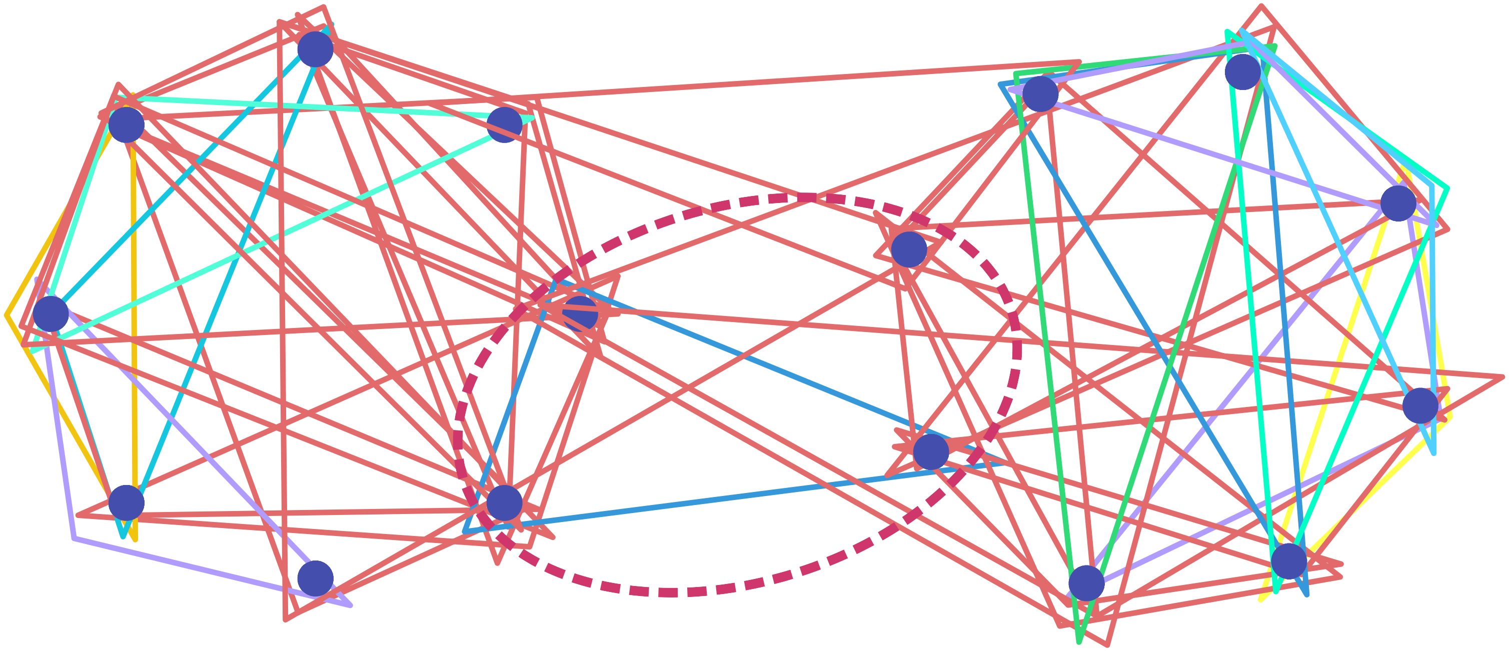
Hypergraph (∂-)cuts

$G=(V,E)$, $n=|V|$ $m=|E|$

$p=$ total size $=\sum_{e\in E}|e|$

edge weights

$w(e)>0$ for $e\in E$



$\partial(S)=\{e\in E,\ \emptyset\subsetneq e\cap S\subsetneq e\}$

$w(\partial(S))\overset{\text{def}}{=}\sum_{e\in\partial(S)}w(e)$

Hypergraph (2-) cut sparsification

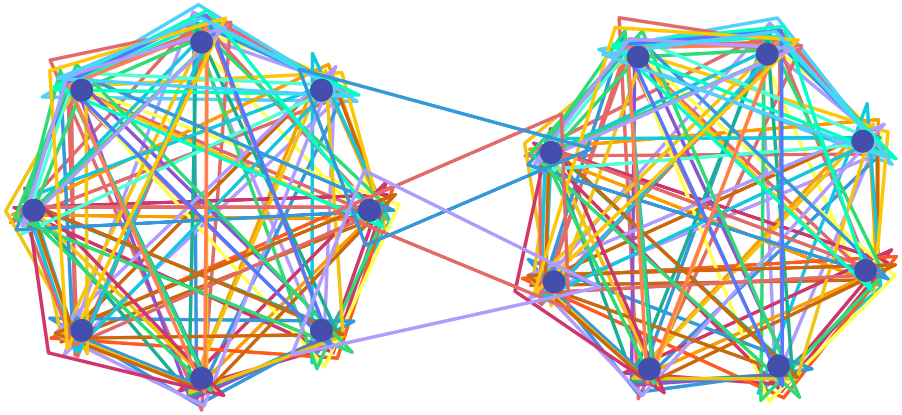
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

$p = \text{total size} = \sum_{e \in E} |e|$

Input: Hypergraph $G = (V, E)$

$n = |V|$ $m = |E|$

$w(e) > 0$ for $e \in E$



Hypergraph (2-) cut sparsification

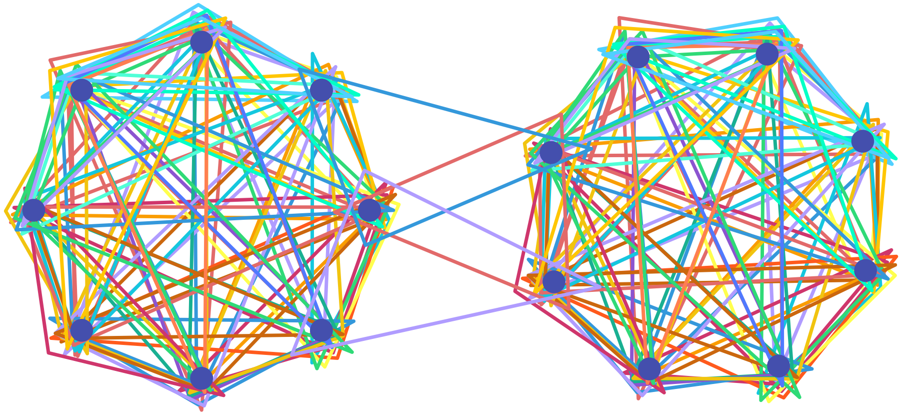
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

$p =$ total size $= \sum_{e \in E} |e|$

Input: Hypergraph $G=(V,E)$

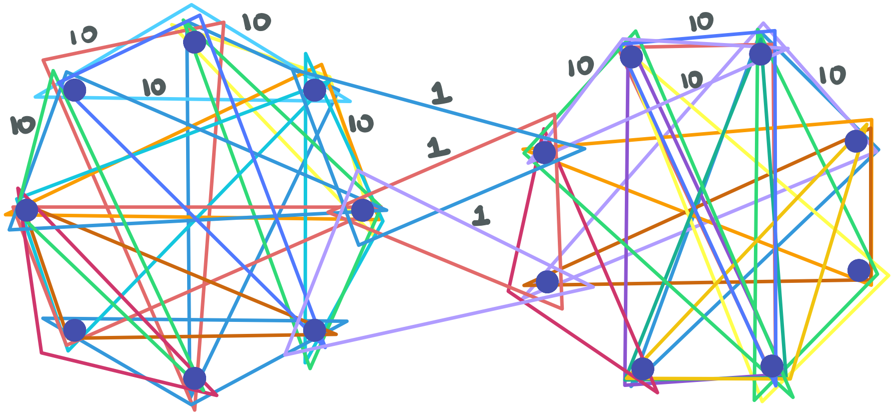
$n=|V|$ $m=|E|$

$w(e)>0$ for $e \in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e \in \tilde{E}$



s.t. (a)

(b)

Hypergraph (2-) cut sparsification

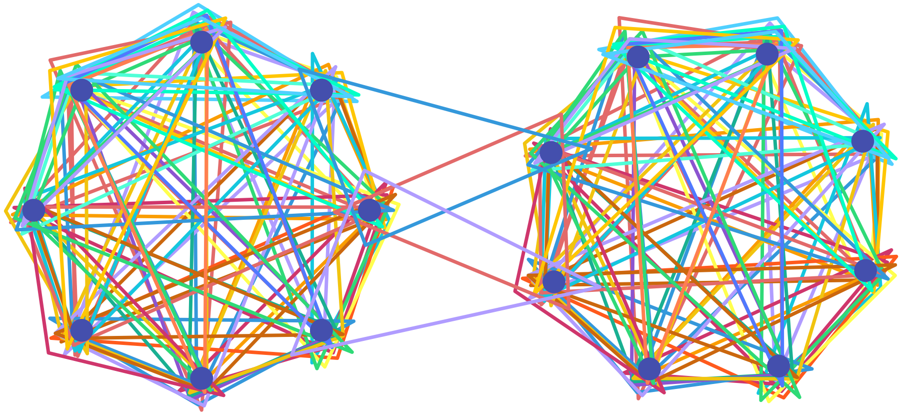
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

$p =$ total size $= \sum_{e \in E} |e|$

Input: Hypergraph $G = (V, E)$

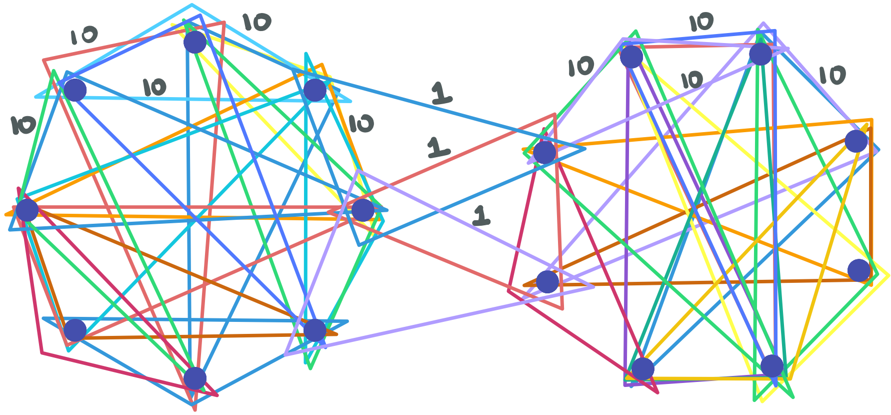
$n = |V|$ $m = |E|$

$w(e) > 0$ for $e \in E$



Goal: subgraph $\tilde{G} = (V, \tilde{E})$

$\tilde{w}(e) > 0$ for $e \in \tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b)

Hypergraph (2-) cut sparsification

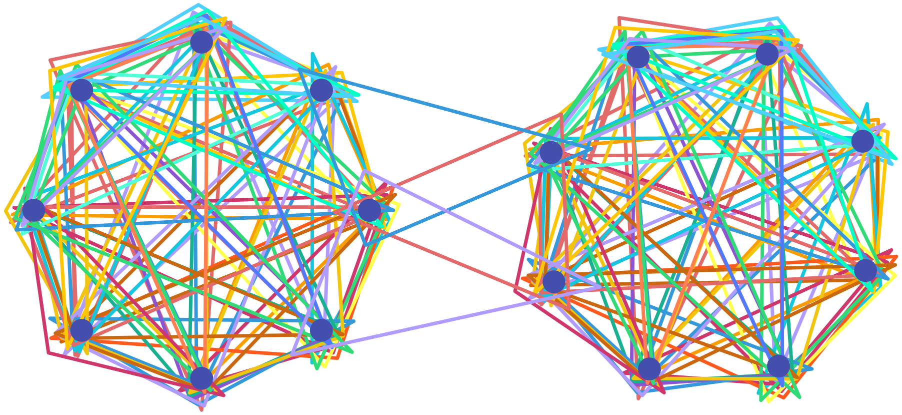
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

p = total size = $\sum_{e\in E} |e|$

Input: Hypergraph $G=(V,E)$

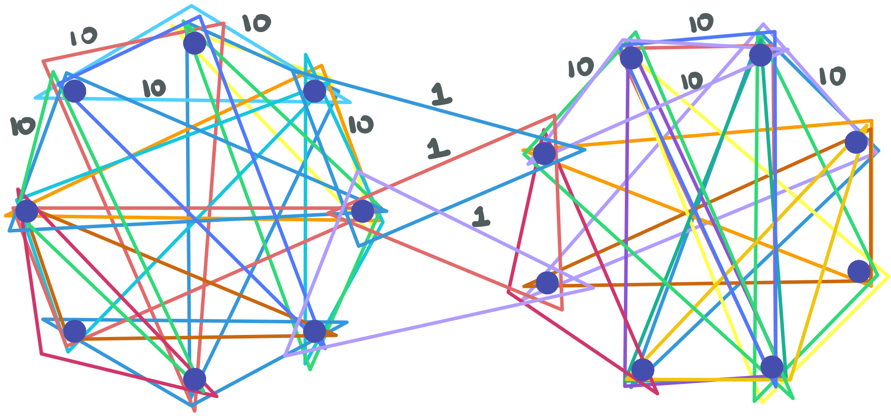
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all 2-cuts have similar weight as in G

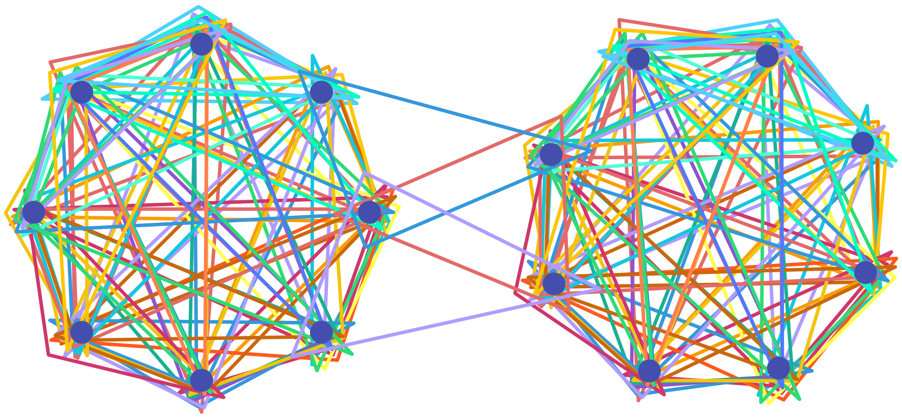
Hypergraph (2-) cut sparsification [KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

$p = \text{total size} = \sum_{e\in E} |e|$

Input: Hypergraph $G=(V,E)$

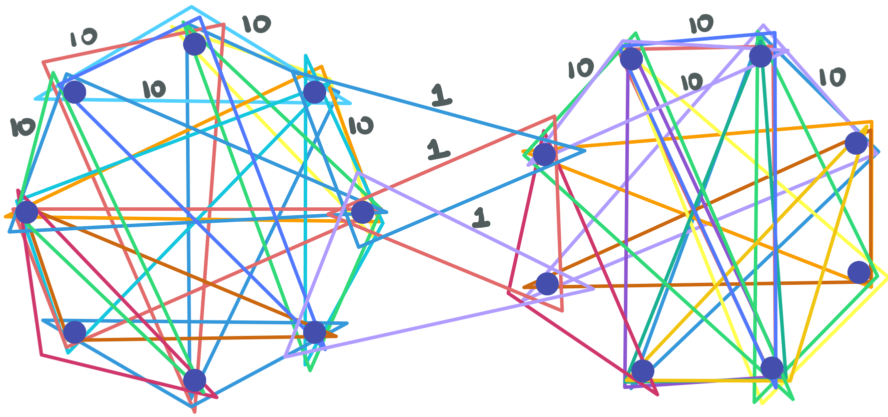
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all 2-cuts have similar weight as in G

Chen, Khanna, Nagda 2020:

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e)$

Hypergraph (2-) cut sparsification

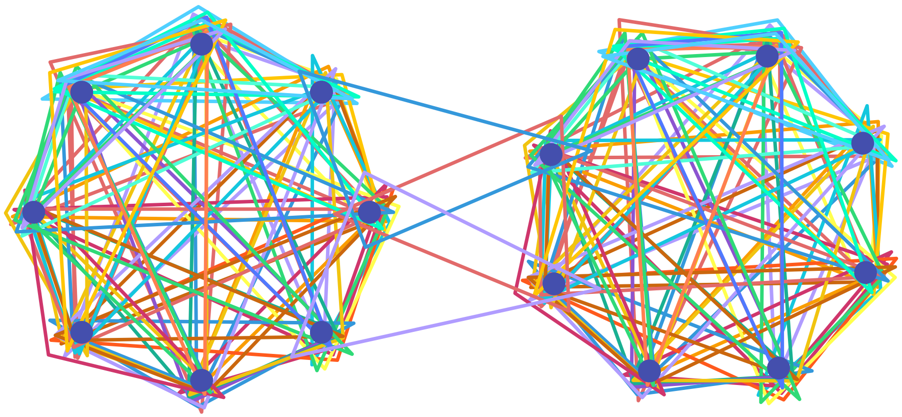
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

p = total size = $\sum_{e\in E} |e|$

Input: Hypergraph $G=(V,E)$

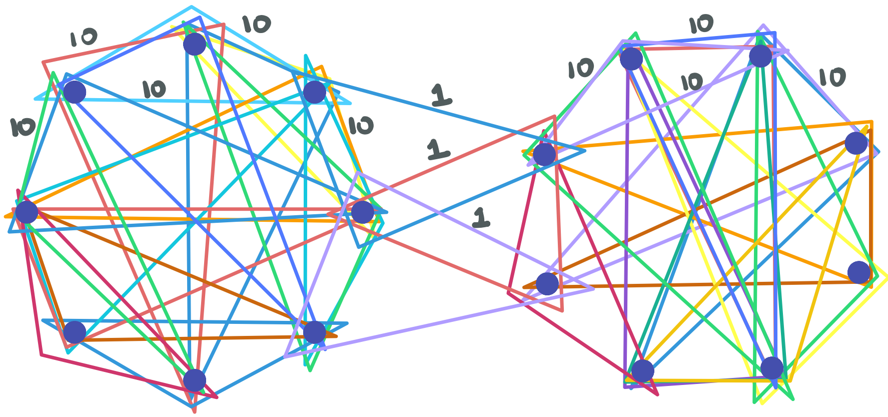
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all 2-cuts have similar weight as in G

Chen, Khanna, Nagda 2020:

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e)$

- Also: "spectral extensions, sums of symm. submodular fun,

(by others)

Matroids and "quotients"

$$M = (V, \mathcal{I})$$

Matroids and “quotients”

$$\mathcal{M} = (\underset{\text{“groundset”}}{\mathcal{N}}, \mathcal{I})$$

Matroids and "quotients"

$$M = (\underset{\text{“groundset”}}{N}, \overset{\subseteq 2^N}{\mathcal{I}})$$

Matroids and "quotients"

$$\mathcal{M} = (\underset{\text{“groundset”}}{\mathcal{N}}, \underset{\text{(feasible) “independent sets”}}{\overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}}})$$

Matroids and "quotients"

$$\mathcal{M} = (\underset{\text{“groundset”}}{\mathcal{N}}, \overset{\subseteq 2^{\mathcal{N}}}{\underset{\text{(feasible) “independent sets”}}{\mathcal{I}}})$$

Independent sets $\mathcal{I}$ satisfy:

1.

2.

3.

Matroids and "quotients"

$$\mathcal{M} = (\underset{\text{“groundset”}}{\mathcal{N}}, \underset{\text{“independent sets”}}{\overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}}})$$

(feasible)

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$

2.

3.

Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset"

(feasible)
"independent sets"

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3.

Matroids and "quotients"

$$\mathcal{M} = (\underset{\text{“groundset”}}{\mathcal{N}}, \underset{\text{“independent sets” (feasible)}}{\overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}}})$$

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

Matroids and "quotients"

$$\mathcal{M} = (\underset{\text{"groundset"}}{\mathcal{N}}, \overset{\subseteq 2^{\mathcal{N}}}{\underset{\text{(feasible) "independent sets"}}{\mathcal{I}}})$$

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

2, 3 $\Rightarrow$ maxim<u>al</u> indep. sets are $\overbrace{\text{maxim}\underline{\text{um}}\text{ indep. sets}}^{\text{"base"}}$

Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \underset{\text{"independent sets"}}{\overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}}})$$

"groundset" → $\mathcal{N}$

(feasible) "independent sets" → $\mathcal{I}$

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

Graphic Matroid

(i.e., forests)

Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset"

(feasible) "independent sets"

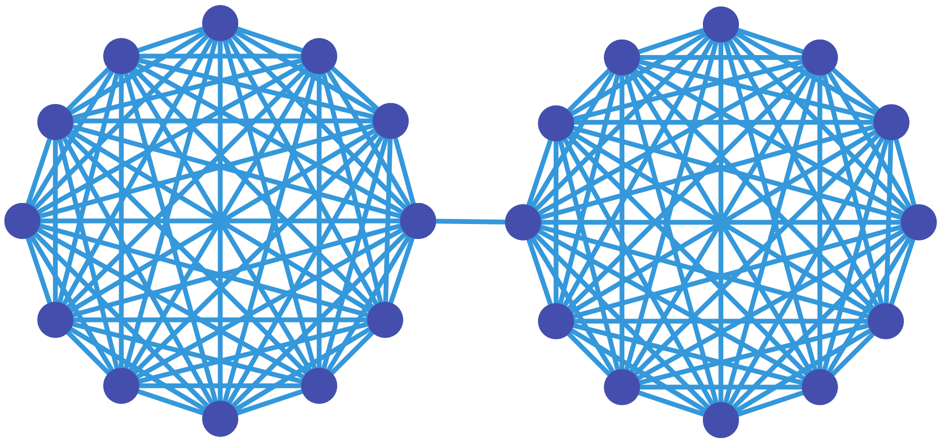
Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)



Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset" (feasible) "independent sets"

Independent sets $\mathcal{I}$ satisfy:

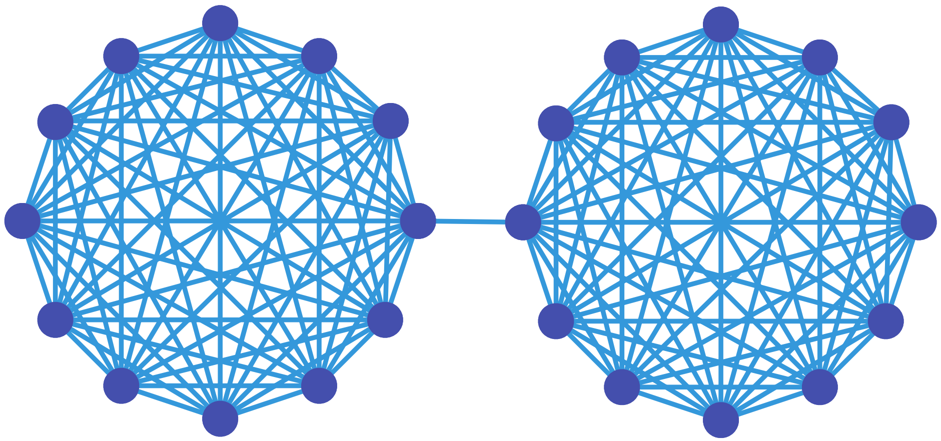
1. $\phi \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$ (groundset)



Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset"

(feasible)
"independent sets"

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

Graphic Matroid

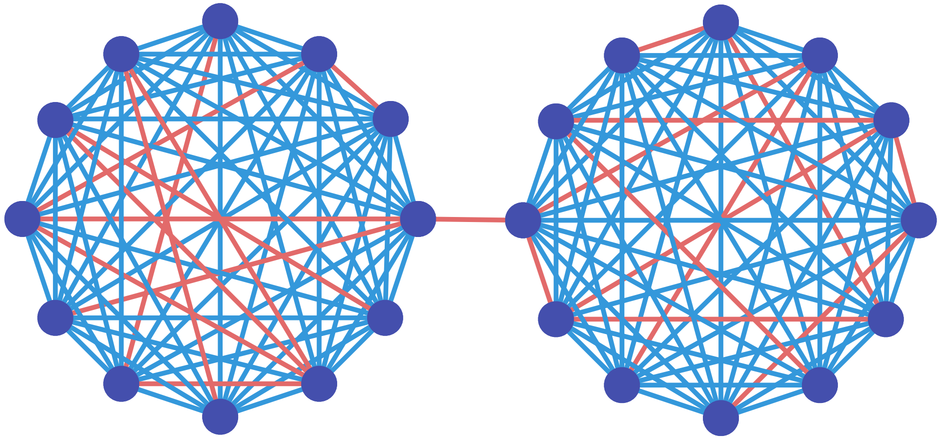
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

(independent sets)



Matroids and "quotients"

$$\mathcal{M} = (\mathcal{N}, \overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset" (feasible) "independent sets"

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

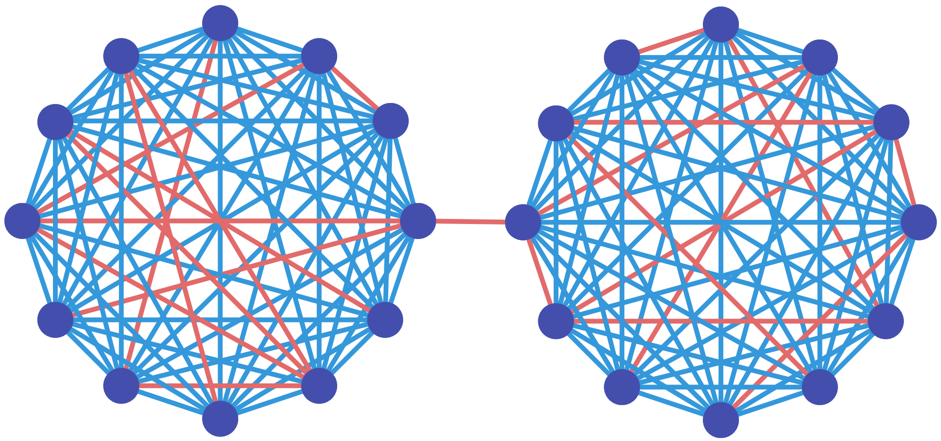
Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$



1. $\emptyset$ is a forest

Matroids and "quotients"

$\subseteq 2^{\mathcal{N}}$

$$\mathcal{M} = (\mathcal{N}, \mathcal{I})$$

"groundset"

(feasible) "independent sets"

Independent sets $\mathcal{I}$ satisfy:

1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S + e \in \mathcal{I}$ for some $e \in T \setminus S$.

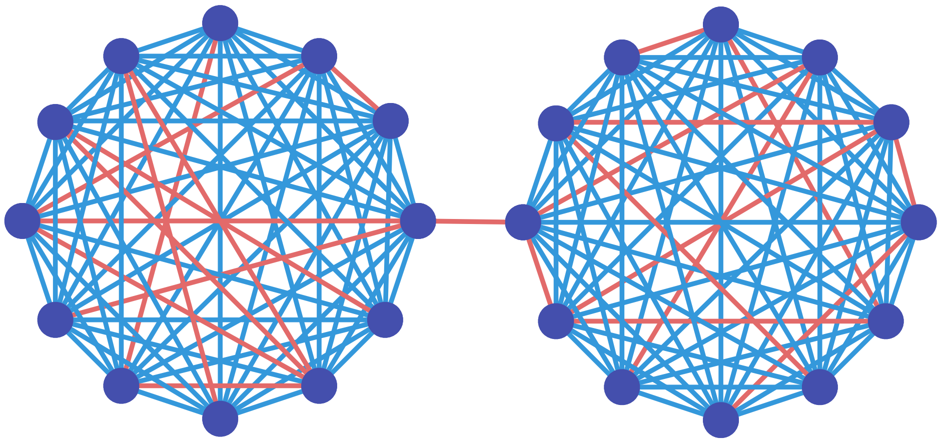
Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$



1. $\emptyset$ is a forest

2. subset of a forest is a forest

Matroids and "quotients"

$$\mathcal{M}=(\mathcal{N},\overset{\subseteq 2^{\mathcal{N}}}{\mathcal{I}})$$

"groundset"

(feasible) "independent sets"

Independent sets $\mathcal{I}$ satisfy:

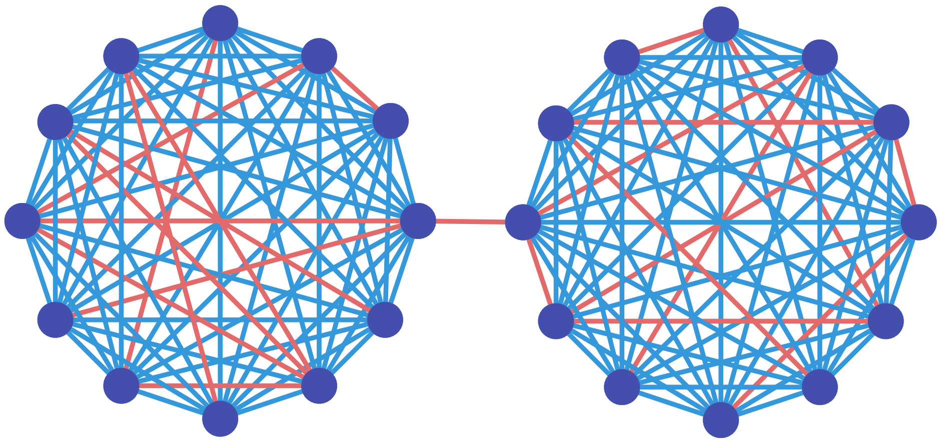
1. $\emptyset \in \mathcal{I}$
2. If $S \subseteq T$, and $T \in \mathcal{I}$, then $S \in \mathcal{I}$
3. If $S, T \in \mathcal{I}$, and $|S| < |T|$, then $S+e \in \mathcal{I}$ for some $e \in T \setminus S$.

Graphic Matroid
(i.e., forests)

fix $G=(V,E)$ (undirected)

$\mathcal{N}=E$

$\mathcal{I}=\{F \subseteq E : F \text{ is a forest}\}$



1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1|<|F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

(Apologies for terminology)

need to define

"rank function", "span", "closed sets",

"quotients"

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F$ is a forest$\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$

2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$

3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$
   $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxima**l** = maxim**um**

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest

2. subset of a forest is a forest

3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

$$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" $=$ $\text{rank}(\mathcal{N})$

e.g. graphic matroid

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maximal = maximum

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

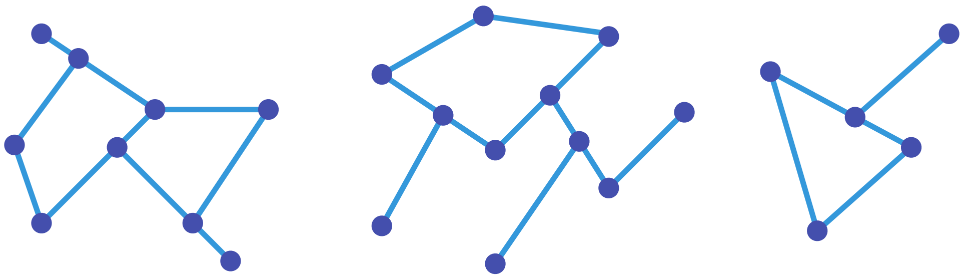
$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = rank($\mathcal{N}$)

e.g. graphic matroid

S

(edges)



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$

2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$

3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$
   $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

$\underline{\underline{\text{maximal}}} = \text{maxi}\underline{\underline{\text{mum}}}$

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest

2. subset of a forest is a forest

3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

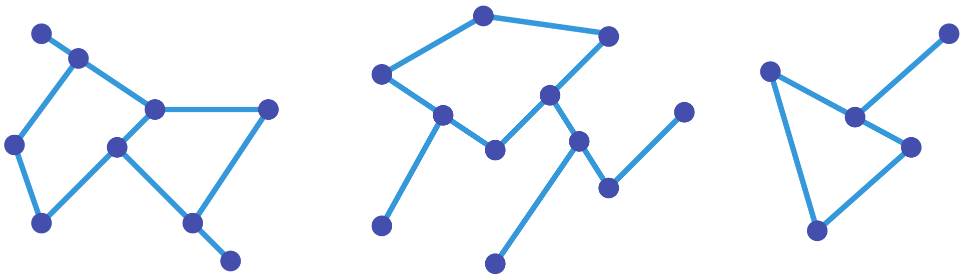
Rank

$$\mathrm{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

e.g. graphic matroid

$\mathrm{rank}(S) =$



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$

2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$

3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$
$e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxima$\underline{\underline{\text{l}}}$ = maxim$\underline{\underline{\text{um}}}$

Graphic Matroid
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest

2. subset of a forest is a forest

3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

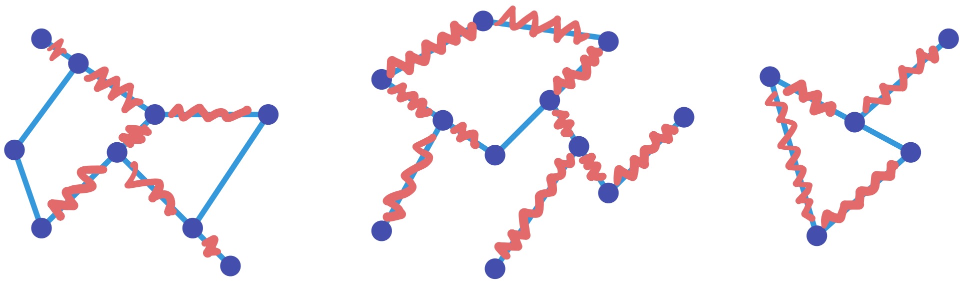
Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

e.g. graphic matroid

$$\text{rank}(S) = \max\{|F| : F \subseteq S,\ F \text{ forest}\}$$



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T,\ T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I},\ |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim**al** = maxim**um**

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

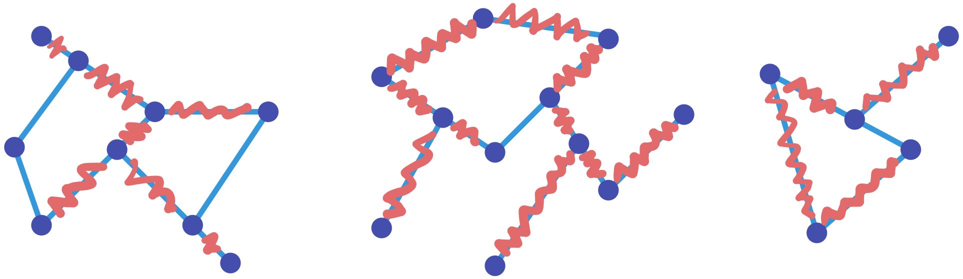
$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

e.g. graphic matroid

$\text{rank}(S) = \max\{|F| : F \subseteq S,\ F \text{ forest}\}$

$= n - (\#\text{ conn. comp. of } S)$



rank of graphic matroid

$= n-1$ if graph is connected

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T,\ T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I},\ |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxima$\underline{\underline{l}}$ = maxi$\underline{\underline{mum}}$

Graphic Matroid

(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

●

●

●

$\mathcal{M} = (\overset{\text{“groundset”}}{\mathcal{N}}, \underset{\text{“independent sets”}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T,\ T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I},\ |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

$\underline{\underline{\text{maximal}}} = \underline{\underline{\text{maximum}}}$

Graphic Matroid

(i.e., forests)

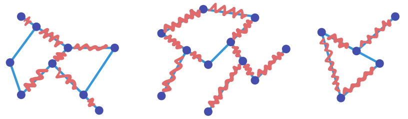
fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

$\text{rank}(S) = n - (\#\text{CC of } S)$



Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = rank($\mathcal{N}$)

Properties of f = rank:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
-
-

$\mathcal{M} = (\mathcal{N}, \mathcal{I})$

"groundset"

"independent sets"

1. $\phi \in \mathcal{I}$
2. $S \subseteq T,\ T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I},\ |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maximal = maximum

Graphic Matroid

(i.e., forests)

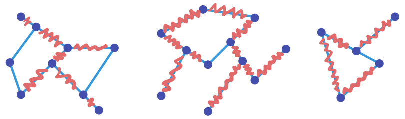
fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. ϕ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

rank(S) = n − (#CC of S)



Rank

$$\text{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

$$f(e|T) \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

"decreasing marginal returns"

-

$$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T,\ T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I},\ |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

Graphic Matroid
(i.e., forests)

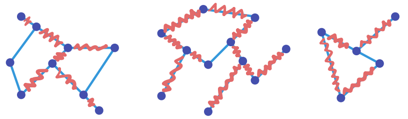
fix $G=(V,E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

$\text{rank}(S) = n - (\#\text{CC of } S)$



Rank

$$\mathrm{rank}(S) = \max\{|I| : I \subseteq S,\ I \in \mathcal{I}\}$$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of $f = \mathrm{rank}$:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

$$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

"decreasing marginal returns"

- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

$$f(e|T) = 0 \text{ or } f(e|T) \geq 1$$

(actually $=1$ for rank function)

$$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

Graphic Matroid

(i.e., forests)

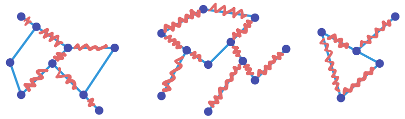
fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

1. $\emptyset$ is a forest
2. subset of a forest is a forest
3. If F_1, F_2 are forests w/ $|F_1| < |F_2|$, some $e \in F_2 \setminus F_1$ connects diff. conn. comp. of F_1

$\mathrm{rank}(S) = n - (\#\text{CC of } S)$



span

$$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\phi \in \mathcal{I}$

2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$

3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maximal = maximum

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

  "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

Graphic Matroid

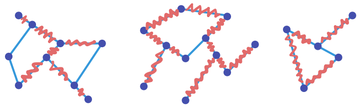
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



span

$$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$$

(including S)

S is "closed" if $S = \text{span}(S)$

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maxima$\underline{\underline{\text{l}}}$ = maxim$\underline{\underline{\text{um}}}$

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

  "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \geq 1$

Graphic Matroid

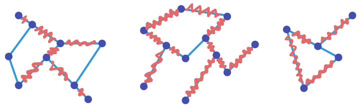
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



span

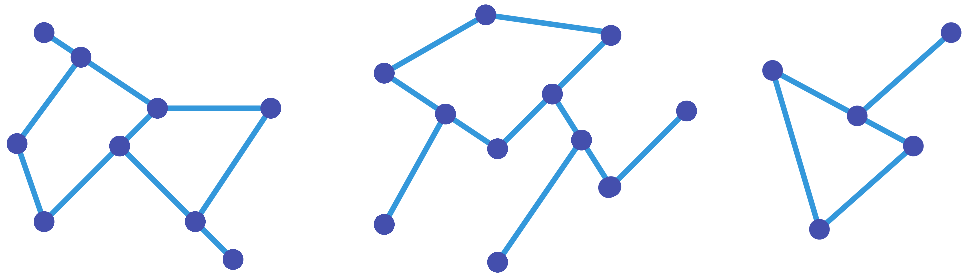
$$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$$

(including S)

S is "closed" if $S = \text{span}(S)$

e.g. graphic matroid

(edges) S



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \mathcal{I})$ "independent sets"

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maxim**al** = maxim**um**

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,
  $$\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$$
  "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \ge 1$

Graphic Matroid

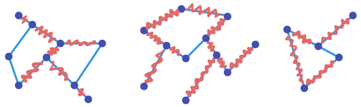
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



span

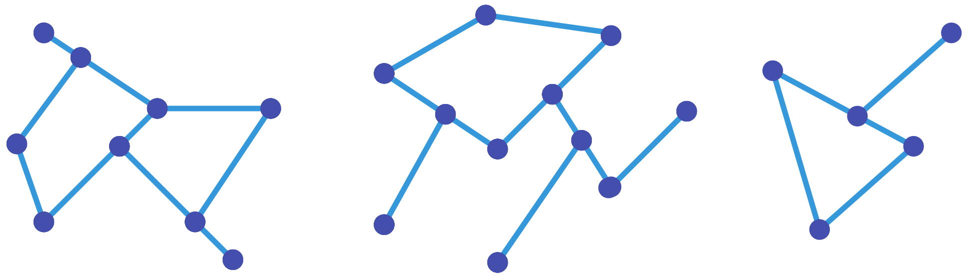
$\text{span}(S) = \{ e \in \mathcal{N} : f(S+e) = f(S) \}$

(including S)

S is "closed" if $S = \text{span}(S)$

e.g. graphic matroid

$\text{span}(S) = \{ \qquad \}$



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\phi \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$
   $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maximal = maximum

$\text{rank}(S) = \max\{ |I| : I \subseteq S, I \in \mathcal{I} \}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \ge 1$

Graphic Matroid

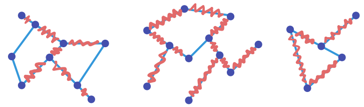
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{ F \subseteq E : F \text{ is a forest} \}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



span

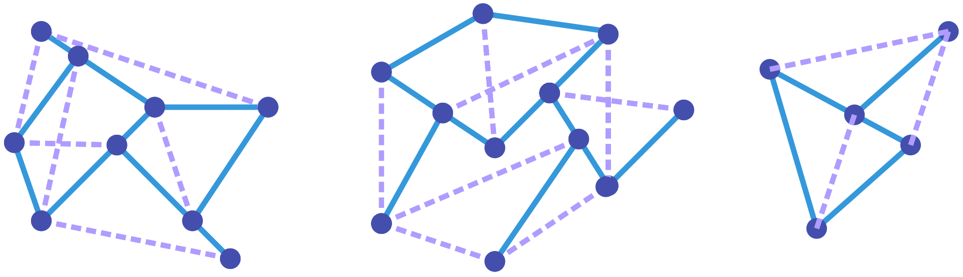
$$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$$

(including S)

S is "closed" if $S = \text{span}(S)$

e.g. graphic matroid

$$\text{span}(S) = \{\text{all edges connected by } S\}$$



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \mathcal{I})$ "independent sets"

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maximal = maximum

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \geq 1$

Graphic Matroid

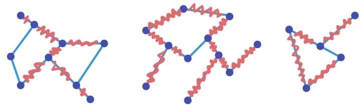
(i.e., forests)

fix $G = (V, E)$ (undirected)

$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



Quotients

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \mathcal{I})$

1. $\emptyset \in \mathcal{I}$ "independent sets"
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maximal = maximum

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

  "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \text{span}(S)$

Graphic Matroid

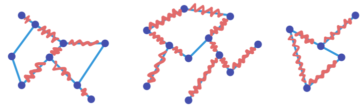
(i.e., forests)

fix $G = (V, E)$ (undirected)

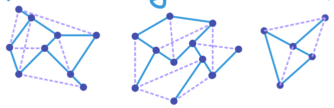
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\text{rank}(S) = n - (\# CC \text{ of } S)$



$\text{span}(S)$ = {edges connected by S}



Quotients

Q is a quotient if $\bar{Q}=\mathcal{N}\setminus Q$ is closed

i.e., if $Q=\mathcal{N}\setminus \text{span}(S)$ for some $S\subseteq\mathcal{N}$.

$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I} \Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}, |S|<|T| \Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maximal = maximum

$\text{rank}(S)=\max\{|I| : I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f=\text{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$,

  $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N} : f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Graphic Matroid

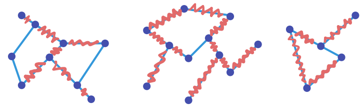
(i.e., forests)

fix $G=(V,E)$ (undirected)

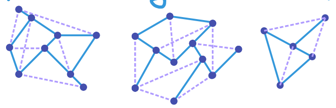
$\mathcal{N}=E$

$\mathcal{I}=\{F\subseteq E : F \text{ is a forest}\}$

$\text{rank}(S)=n-(\#\text{CC of } S)$



span(S) = {edges connected by S}

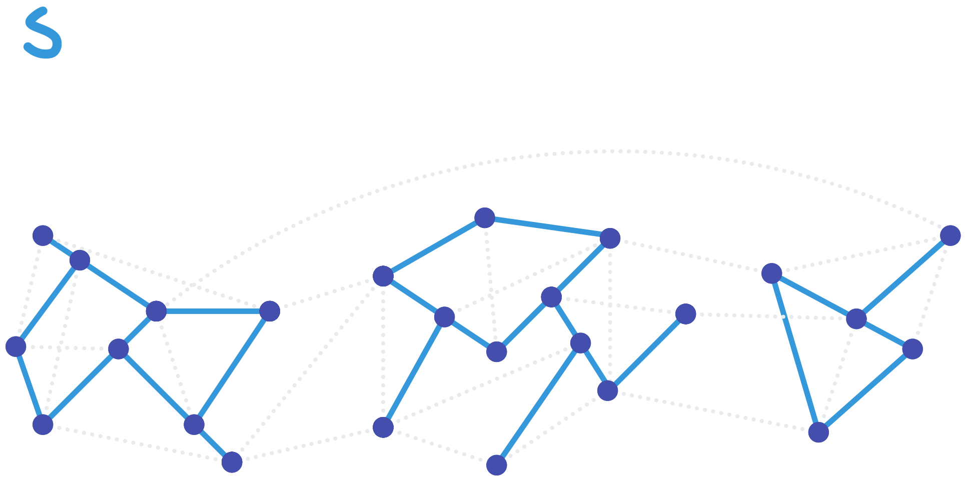


Quotients

Q is a quotient if $\overline{Q} = \mathcal{N} \setminus Q$ is closed

i.e., if $Q = \mathcal{N} \setminus \mathrm{span}(S)$ for some $S \subseteq \mathcal{N}$.

e.g. for graphic matroid:



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maximal = maximum

$\mathrm{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

$$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$ "decreasing marginal returns"

- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

$\mathrm{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \mathrm{span}(S)$

Graphic Matroid

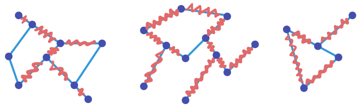
(i.e., forests)

fix $G = (V, E)$ (undirected)

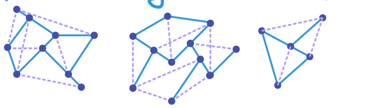
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F \text{ is a forest}\}$

$\mathrm{rank}(S) = n - (\#\text{CC of } S)$



$\mathrm{span}(S) = \{\text{edges connected by } S\}$



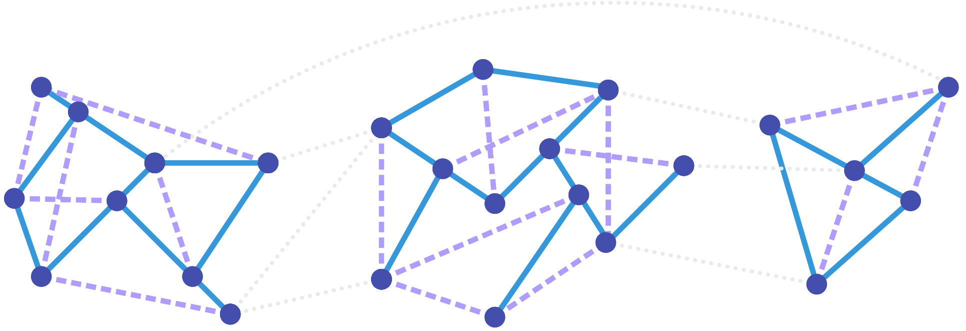
Quotients

Q is a quotient if $\bar{Q} = \mathcal{N} \setminus Q$ is closed

i.e., if $Q = \mathcal{N} \setminus \mathrm{span}(S)$ for some $S \subseteq \mathcal{N}$.

e.g. for graphic matroid:

$\mathrm{span}(S) = \{$all edges connected by $S\}$



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maximal = maximum

$\mathrm{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \ge 1$

$\mathrm{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \mathrm{span}(S)$

Graphic Matroid

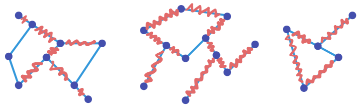
(i.e., forests)

fix $G = (V, E)$ (undirected)

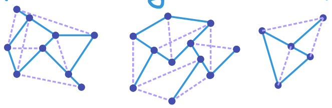
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F$ is a forest$\}$

$\mathrm{rank}(S) = n - (\#\mathrm{CC}$ of $S)$



$\mathrm{span}(S) = \{$edges connected by $S\}$



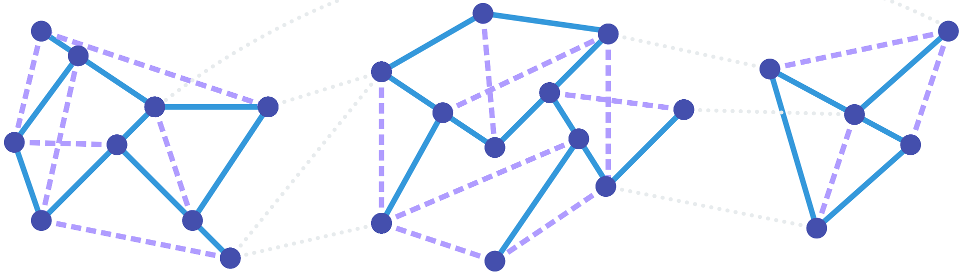
Quotients

Q is a quotient if $\bar{Q} = \mathcal{N} \setminus Q$ is closed

i.e., if $Q = \mathcal{N} \setminus \text{span}(S)$ for some $S \subseteq \mathcal{N}$.

e.g. for graphic matroid:

$\text{span}(S) = \{$all edges connected by $S\}$



$Q = E \setminus \text{span}(S)$

$\mathcal{M} = (\overset{\text{“groundset”}}{\mathcal{N}}, \underset{\text{“independent sets”}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maximal = maximum

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

“rank of $\mathcal{M}$” = $\text{rank}(\mathcal{N})$

Properties of $f = \text{rank}$:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

$$\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

“decreasing marginal returns”

- “normalized”: for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \ge 1$

$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is “closed” if $S = \text{span}(S)$

Graphic Matroid

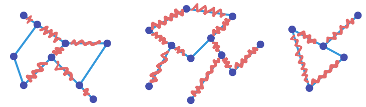
(i.e., forests)

fix $G = (V, E)$ (undirected)

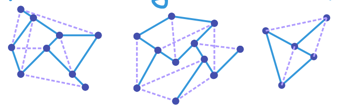
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F$ is a forest$\}$

$\text{rank}(S) = n - (\#CC$ of $S)$



$\text{span}(S) = \{$edges connected by $S\}$



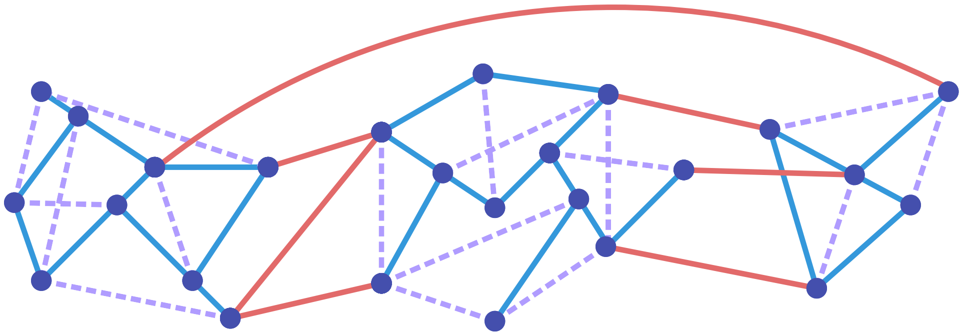
Quotients

Q is a quotient if $\bar{Q} = \mathcal{N} \setminus Q$ is closed

i.e., if $Q = \mathcal{N} \setminus \mathrm{span}(S)$ for some $S \subseteq \mathcal{N}$.

e.g. for graphic matroid:

$\mathrm{span}(S) = \{$all edges connected by $S\}$



$Q = E \setminus \mathrm{span}(S)$

$=$ edges cut by conn. comp. of S

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

$\mathrm{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" $= \mathrm{rank}(\mathcal{N})$

Properties of $f = \mathrm{rank}$:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \ge 1$

$\mathrm{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \mathrm{span}(S)$

Graphic Matroid

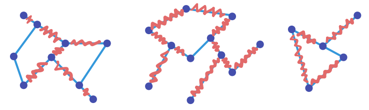
(i.e., forests)

fix $G = (V, E)$ (undirected)

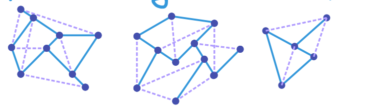
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F$ is a forest$\}$

$\mathrm{rank}(S) = n - (\#\mathrm{CC}$ of $S)$



$\mathrm{span}(S) = \{$edges connected by $S\}$

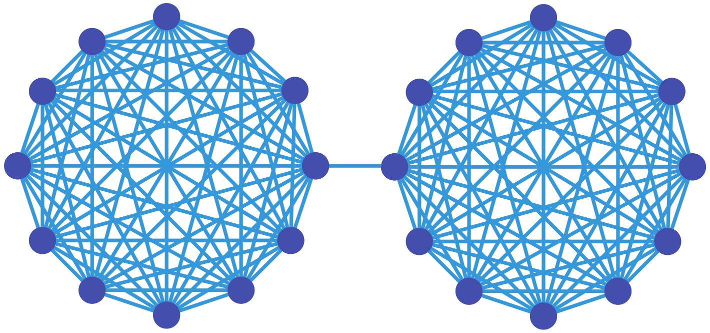


Matroid quotient sparsification

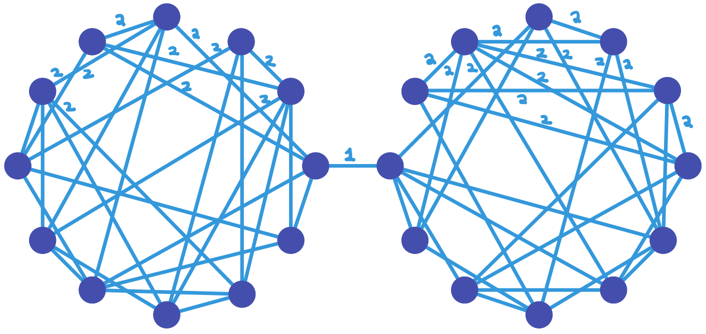
Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

$w(e)>0$ for $e\in\mathcal{N}$



Goal: $\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $\text{support}(\tilde{w})$ small

(b) all quotients have similar weight as $(\mathcal{M},w)$

$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\phi\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxim**al** = maxim**um**

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f=\text{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

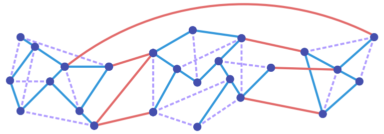
Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests

$\text{rank}(S)=n-(\#\text{CC of } S)$

$\text{span}(S)=\{\text{edges connected by } S\}$

$Q=$ edges cut by conn. comp. of S

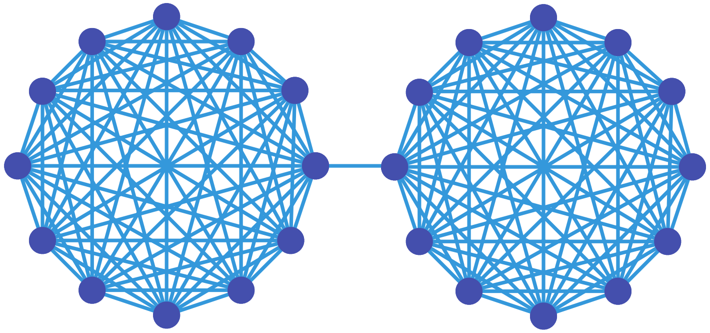


Matroid quotient sparsification

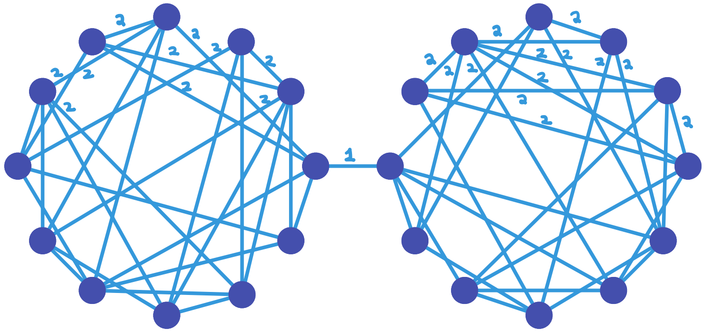
Input: $M=(\mathcal{N},\mathcal{I})$

$n=|\mathcal{N}|$, $r=\mathrm{rank}(\mathcal{N})$

$w(e)>0$ for $e\in\mathcal{N}$



Goal: $\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $\mathrm{support}(\tilde{w})$ small

(b) all quotients have similar weight as (M,w)

Theorem:

$M=(\mathcal{N},\mathcal{I})$ "groundset" "independent sets"

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I} \Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T| \Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxim$\underline{\underline{\text{al}}}$ = maxim$\underline{\underline{\text{um}}}$

$\mathrm{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of M" = $\mathrm{rank}(\mathcal{N})$

Properties of $f=\mathrm{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$,

  $f(e|T)=0$ or $f(e|T)\ge 1$

$\mathrm{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\mathrm{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus \mathrm{span}(S)$

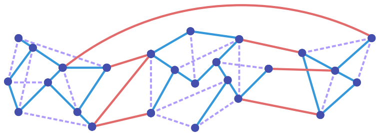
Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests

$\mathrm{rank}(S)=n-(\#\text{CC of } S)$

$\mathrm{span}(S)=\{\text{edges connected by } S\}$

$Q=$ edges cut by conn. comp. of S

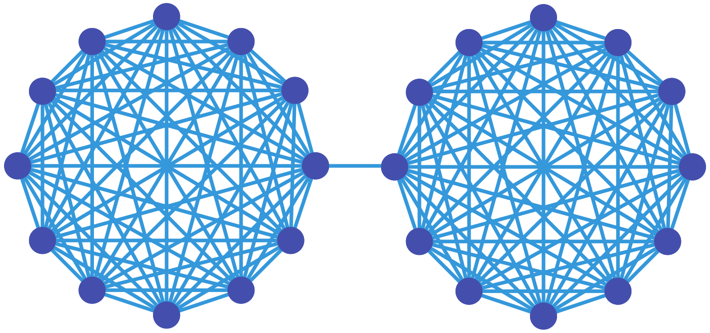


Matroid quotient sparsification

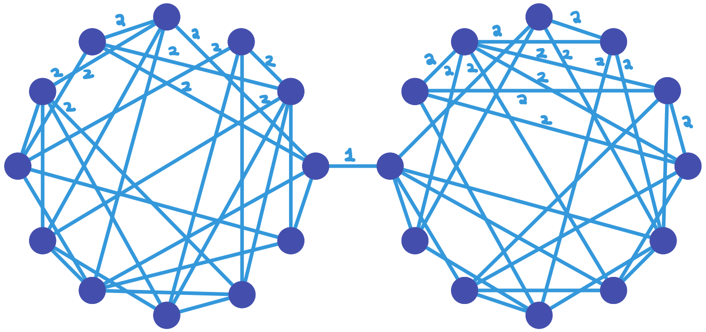
Input: $\mathcal{M} = (\mathcal{N}, \mathcal{I})$

$n = |\mathcal{N}|$, $r = \mathrm{rank}(\mathcal{N})$

$w(e) > 0$ for $e \in \mathcal{N}$



Goal: $\tilde{w}(e) > 0$ for $e \in \tilde{E}$



s.t. (a) $\mathrm{support}(\tilde{w})$ small

(b) all quotients have similar weight as $(\mathcal{M}, w)$

Theorem:

- $|\mathrm{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$

$\mathcal{M} = (\mathcal{N}, \mathcal{I})$

"groundset"

"independent sets"

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxima<u>l</u> = maxim<u>um</u>

$\mathrm{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of $f = \mathrm{rank}$:

- monotone: $f(S) \le f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

$$\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$$

"decreasing marginal returns"

- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \ge 1$

$\mathrm{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$

(including S)

S is "closed" if $S = \mathrm{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q = V \setminus \mathrm{span}(S)$

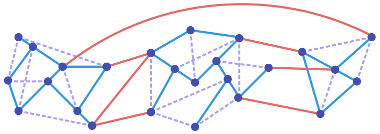
Graphic Matroid

$\mathcal{N} = E$, $\mathcal{I}$ = forests

$\mathrm{rank}(S) = n - (\#\mathrm{CC}$ of $S)$

$\mathrm{span}(S) = \{$edges connected by $S\}$

Q = edges cut by conn. comp. of S

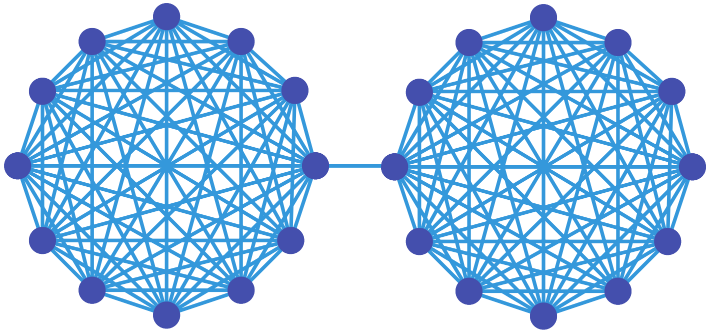


Matroid quotient sparsification

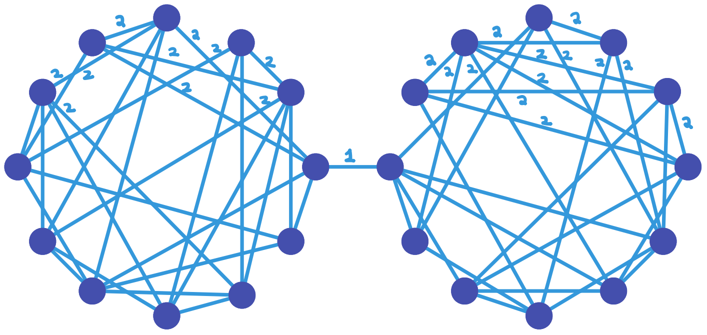
Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$

$n=|\mathcal{N}|,\ r=\text{rank}(\mathcal{N})$

$w(e)>0$ for $e\in\mathcal{N}$



Goal: $\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as $(\mathcal{M},w)$

Theorem:

- $|\text{support}(\tilde{w})| = O(r\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

$$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}, |S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxim**al** = maxim**um**

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = rank($\mathcal{N}$)

Properties of f = rank:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

$$\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$$

"decreasing marginal returns"

- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\overline{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

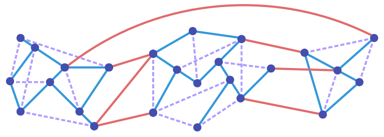
Graphic Matroid

$\mathcal{N}=E,\ \mathcal{I}=$ forests

$\text{rank}(S)=n-(\#\text{CC of } S)$

$\text{span}(S)=\{\text{edges connected by } S\}$

Q = edges cut by conn. comp. of S

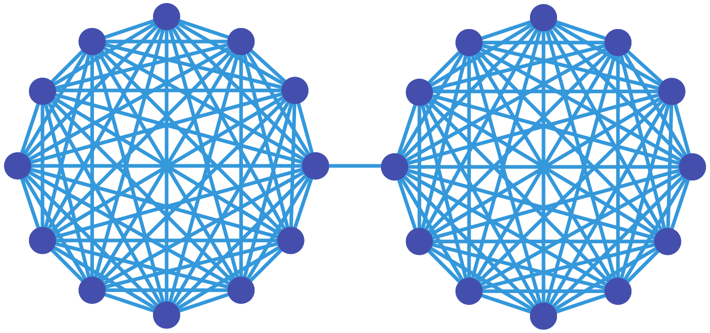


Matroid quotient sparsification

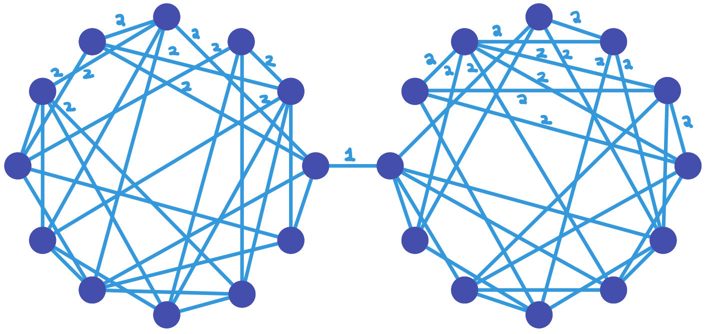
Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

$w(e)>0$ for $e\in\mathcal{N}$



Goal: $\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) support$(\tilde{w})$ small

(b) all quotients have similar weight as $(\mathcal{M},w)$

Theorem:

- $|\text{support}(\tilde{w})| = O(r\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

($\tilde{O}(n)$ rand. time and rank oracle queries)

$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxim<u>al</u> = maxim<u>um</u>

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = rank$(\mathcal{N})$

Properties of f=rank:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

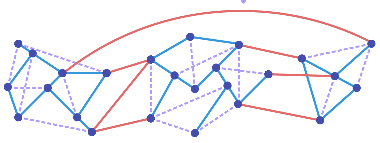
Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}$ = forests

$\text{rank}(S)=n-(\#\text{CC of }S)$

$\text{span}(S)$ = {edges connected by S}

Q = edges cut by conn. comp. of S



Matroid quotient sparsification

Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$ $w(e)>0$ for $e\in\mathcal{N}$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

Theorem:

- $|\text{support}(\tilde{w})| = O(r\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX: all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

$$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\overset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I} \Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T| \Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maximal = maximum

$\text{rank}(S)=\max\{|I| : I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f=\text{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

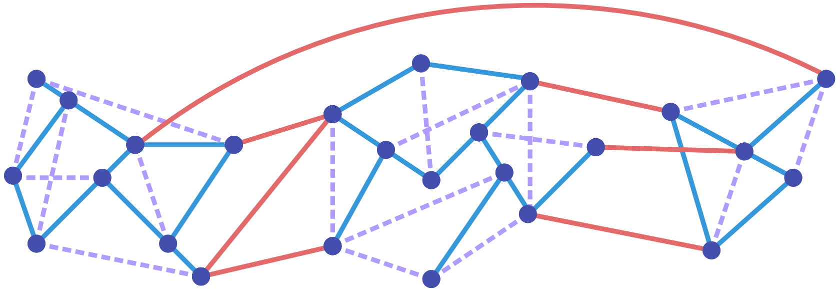
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N} : f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$



Matroid quotient sparsification

Input: $\underset{n=|\mathcal{N}|,\ r=\mathrm{rank}(\mathcal{N})}{\mathcal{M}=(\mathcal{N},\mathcal{I})}$ $w(e)>0$ for $e\in\mathcal{N}$

Theorem:

- $|\mathrm{support}(\tilde{w})| = O(r\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\leq\tilde{w}(Q)\leq(1+\varepsilon)w(Q)$

$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}, |S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxima$\underline{\underline{\text{l}}}$ = maxim$\underline{\underline{\text{um}}}$

$\mathrm{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of $f=\mathrm{rank}$:

- monotone: $f(S)\leq f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)}\leq\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$,

  $f(e|T)=0$ or $f(e|T)\geq 1$

$\mathrm{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

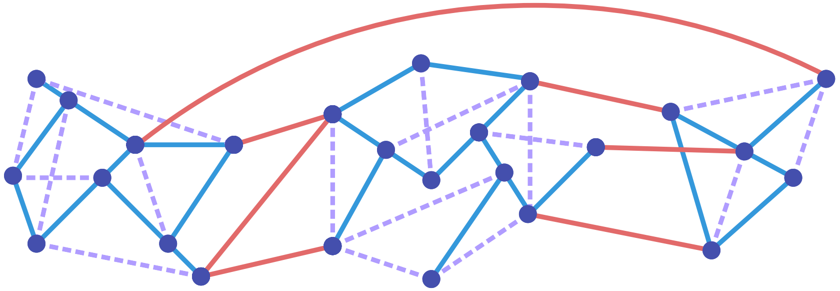
S is "closed" if $S=\mathrm{span}(S)$

Q quotient $\Leftrightarrow\bar{Q}$ closed

i.e., $Q=V\setminus\mathrm{span}(S)$

Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests



Matroid quotient sparsification

Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$ $w(e)>0$ for $e\in\mathcal{N}$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

Theorem:

- $|\text{support}(\tilde{w})| = O(r\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

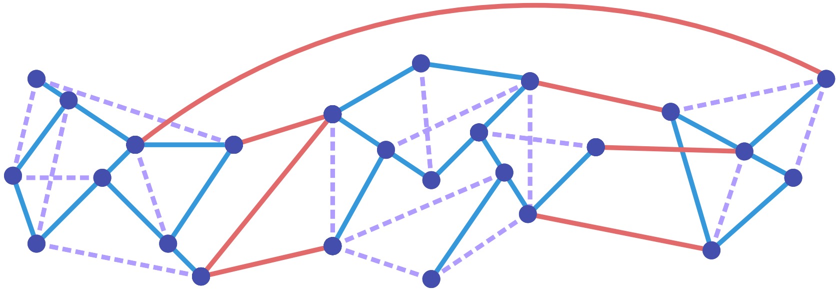
Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests

$\text{rank}(S)=n-(\#\text{CC of } S)$

$\text{span}(S)=$ {edges connected by S}

$Q=$ edges cut by conn. comp. of S



$$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxim**al** = maxim**um**

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" $=\text{rank}(\mathcal{N})$

Properties of $f=\text{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

Matroid quotient sparsification

Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$ $w(e)>0$ for $e\in\mathcal{N}$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

Theorem:

- $|\text{support}(\tilde{w})| = \boxed{O(r\log(n)/\varepsilon^2)}$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

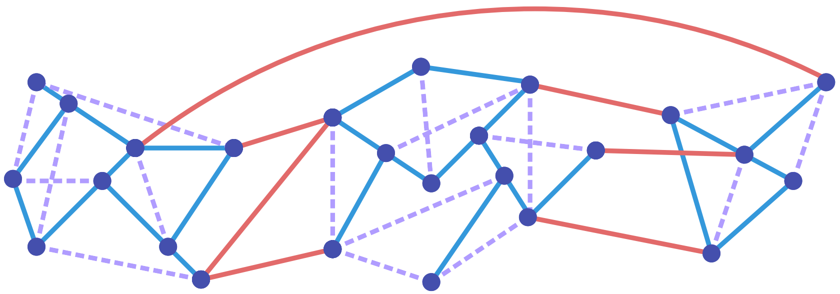
Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests

$\text{rank}(S)=n-(\#\text{CC of } S)$

$\text{span}(S)=$ {edges connected by S}

$Q=$ edges cut by conn. comp. of S



$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T$, $T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxima**l** = maxim**um**

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" $=\text{rank}(\mathcal{N})$

Properties of $f=$ rank:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

Matroid quotient sparsification

Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$ $w(e)>0$ for $e\in\mathcal{N}$

$n=|\mathcal{N}|$, $r=\text{rank}(\mathcal{N})$

Theorem:

- $|\text{support}(\tilde{w})| = \boxed{O(r\log(n)/\varepsilon^2)}$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)$

Graphic Matroid

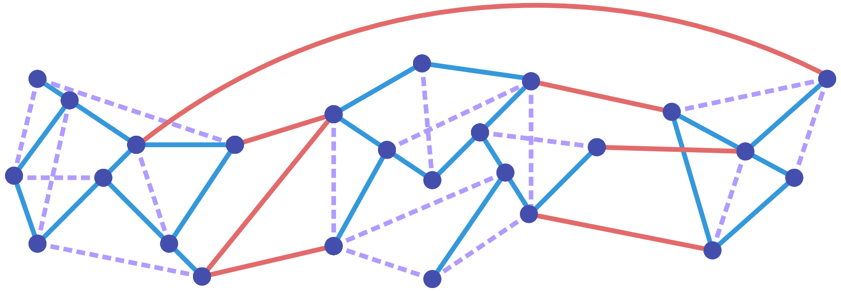
$\mathcal{N}=E$, $\mathcal{I}$= forests

$O(n\log(n)/\varepsilon^2)$ edges

$\text{rank}(S)=n-(\#\text{CC of } S)$

$\text{span}(S)=$ {edges connected by S}

Q = edges cut by conn. comp. of S



$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\mathcal{I})$ "independent sets"

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}, |S|<|T| \Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maxima<u>l</u> = maxim<u>um</u>

$\text{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of $f=\text{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge 1$

$\text{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is "closed" if $S=\text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\text{span}(S)$

Matroid quotient sparsification

Input: $\mathcal{M} = (\mathcal{N}, \mathcal{I})$ $w(e) > 0$ for $e \in \mathcal{N}$

$n = |\mathcal{N}|$, $r = \mathrm{rank}(\mathcal{N})$

Theorem:

- $|\mathrm{support}(\tilde{w})| = \boxed{O(r \log(n)/\varepsilon^2)}$
- $(1+\varepsilon)$-APX all quotients: $\boxed{(1-\varepsilon)\, w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)\, w(Q)}$

Graphic Matroid

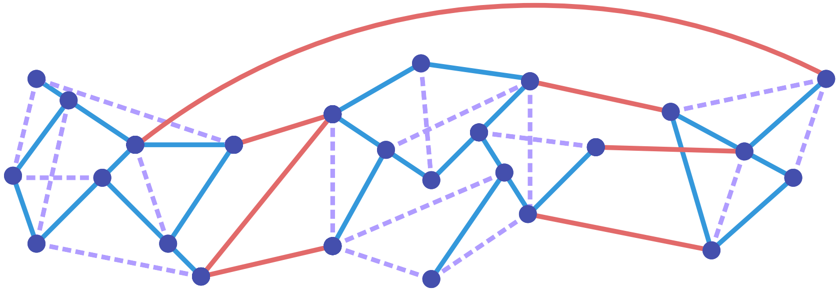
$\mathcal{N} = E$, $\mathcal{I} =$ forests

$O(n \log(n)/\varepsilon^2)$ edges

$\mathrm{rank}(S) = n - (\#\mathrm{CC} \text{ of } S)$

$\mathrm{span}(S) =$ {edges connected by S}

$Q =$ edges cut by conn. comp. of S



$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\phi \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in \mathcal{I}$

maxim$\underline{\underline{\text{al}}}$ = maxim$\underline{\underline{\text{um}}}$

$\mathrm{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\mathrm{rank}(\mathcal{N})$

Properties of $f = \mathrm{rank}$:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \geq 1$

$\mathrm{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \mathrm{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q = V \setminus \mathrm{span}(S)$

Matroid quotient sparsification

Input: $\mathcal{M}=(\mathcal{N},\mathcal{I})$ $w(e)>0$ for $e\in\mathcal{N}$

$n=|\mathcal{N}|$, $r=\mathrm{rank}(\mathcal{N})$

Theorem:

- $|\mathrm{support}(\tilde{w})| = \boxed{O(r\log(n)/\varepsilon^2)}$
- $(1+\varepsilon)$-APX all quotients: $\boxed{(1-\varepsilon)w(Q)\le\tilde{w}(Q)\le(1+\varepsilon)w(Q)}$

Graphic Matroid

$\mathcal{N}=E$, $\mathcal{I}=$ forests

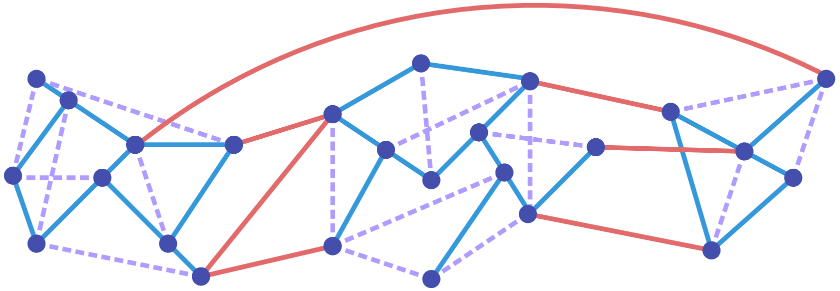
$O(n\log(n)/\varepsilon^2)$ edges

$\mathrm{rank}(S)=n-(\#\mathrm{CC}\text{ of }S)$

$\mathrm{span}(S)=$ {edges connected by S}

$Q=$ edges cut by conn. comp. of S

(1+ε)-APX all cuts



$$\mathcal{M}=(\overset{\text{“groundset”}}{\mathcal{N}},\underset{\text{“independent sets”}}{\mathcal{I}})$$

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I}\Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T|\Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maximal = maximum

$\mathrm{rank}(S)=\max\{|I|: I\subseteq S, I\in\mathcal{I}\}$

“rank of $\mathcal{M}$” = $\mathrm{rank}(\mathcal{N})$

Properties of $f=\mathrm{rank}$:

- monotone: $f(S)\le f(T)$ for $S\subseteq T$
- submodular: if $S\subseteq T$, and $e\in\mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)}\le\underbrace{f(e|S)}_{f(S+e)-f(S)}$ “decreasing marginal returns”
- “normalized”: for $T\subseteq\mathcal{N}$ and $e\in\mathcal{N}$, $f(e|T)=0$ or $f(e|T)\ge1$

$\mathrm{span}(S)=\{e\in\mathcal{N}: f(S+e)=f(S)\}$ (including S)

S is “closed” if $S=\mathrm{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q=V\setminus\mathrm{span}(S)$

"Submodular quotients"

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be: (e.g., rank function)

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$, "decreasing marginal returns" $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

"Submodular quotients"

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be: (e.g., rank function)

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$, $\overset{\text{"decreasing marginal returns"}}{\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}}$
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

- $\text{span}_f(S) = \{\overset{\text{(including } S)}{e \in \mathcal{N}}: f(e|S) = 0\}$

"Submodular quotients"

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be: (e.g., rank function)

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

- $\text{span}_f(S) = \{e \in \mathcal{N} : f(e|S) = 0\}$ (including S)
- S closed if $S = \text{span}_f(S)$

"Submodular quotients"

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be: (e.g., rank function)

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$, $\overset{\text{"decreasing marginal returns"}}{\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}}$
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$

- $\mathrm{span}_f(S) = \{ \overset{\text{(including } S)}{e \in \mathcal{N} :} f(e|S) = 0 \}$
- S closed if $S = \mathrm{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed

  i.e., $Q = \mathcal{N} \setminus \mathrm{span}_f(S)$ for some S

"Submodular quotients"

Let $f: 2^N \to \mathbb{R}_{\geq 0}$ be: (e.g., rank function)

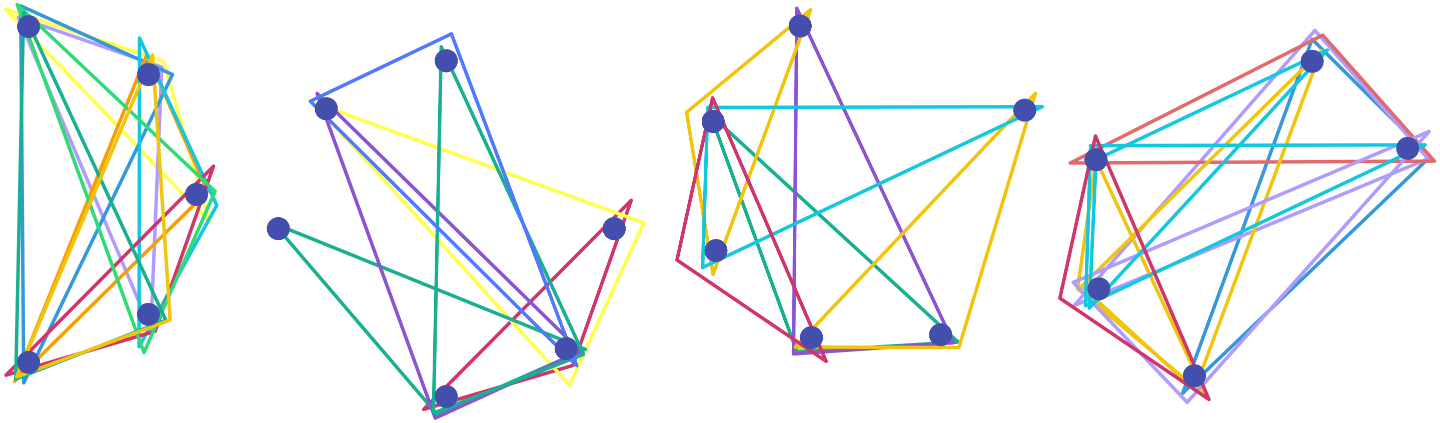
- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in N$, "decreasing marginal returns" $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$
- "normalized": for $T \subseteq N$, $e \in N$, $f(e|T) = 0$ or $f(e|T) \geq 1$

- $\text{span}_f(S) = \{e \in N : f(e|S) = 0\}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = N \setminus Q$ closed

  i.e., $Q = N \setminus \text{span}_f(S)$ for some S
- "rank of f" $= f(N)$

Hypergraphic polymatroid function

fix hypergraph $G=(V,E)$



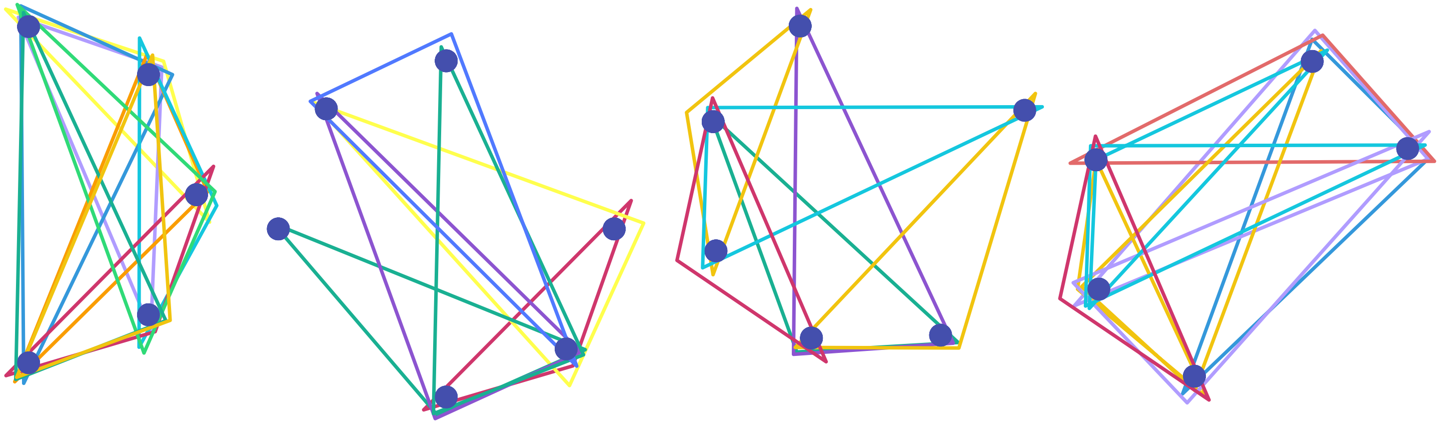
$$f(S) = n - \left(\begin{array}{c}\text{\# connected} \\ \text{components of } S\end{array}\right)$$

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\mathrm{span}_f(S) = \{\overset{\text{(including } S\text{)}}{e \in \mathcal{N}}: f(e|S) = 0\}$
- S closed if $S = \mathrm{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \mathrm{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Hypergraphic polymatroid function

fix hypergraph $G=(V,E)$



$$f(S) = n - \left(\begin{matrix}\#\text{ connected}\\ \text{components of } S\end{matrix}\right)$$

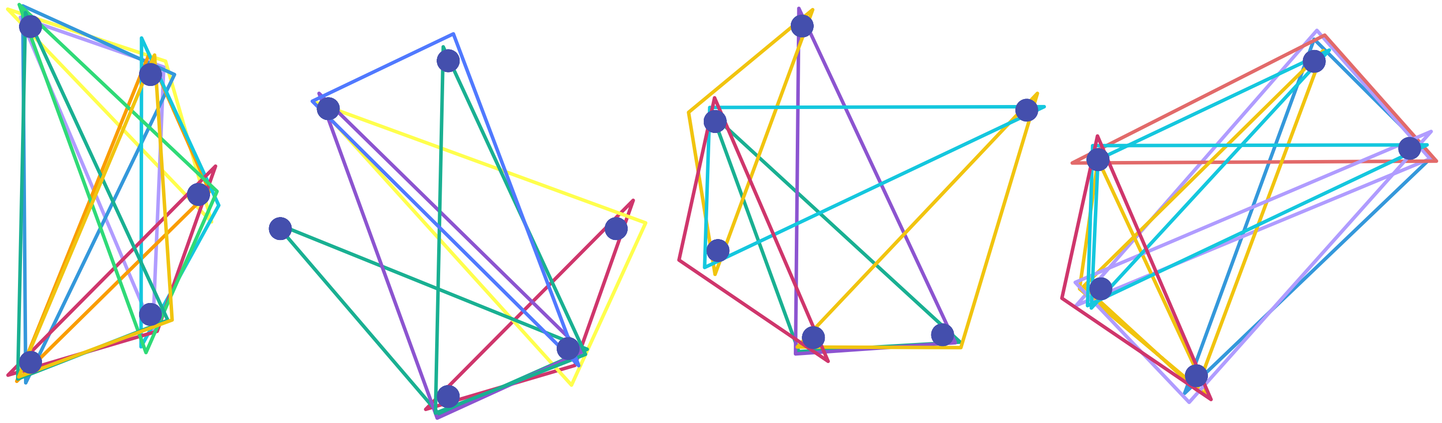
$$\text{span}(S) = \{\text{hyperedges } \overset{\text{(all endpoints)}}{\text{connected}} \text{ by } S\}$$

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{\overset{\text{(including } S)}{e \in \mathcal{N}}: f(e|S) = 0\}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Hypergraphic polymatroid function

fix hypergraph $G=(\mathcal{V},E)$



$$f(S) = n - \begin{pmatrix}\#\text{ connected}\\ \text{components of } S\end{pmatrix}$$

Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{\overset{\text{(including } S)}{e \in \mathcal{N}} : f(e|S) = 0\}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

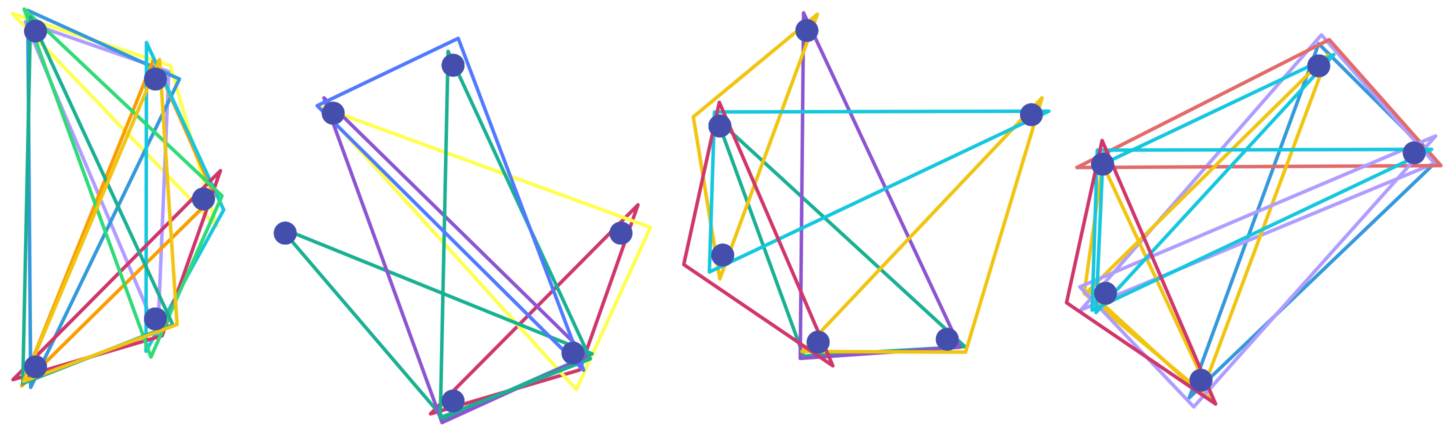
span(S) = {hyperedges $\overset{\text{(all endpoints)}}{\text{connected}}$ by S}

Quotient Q = E \ span(S)

= {edges cut by conn. comp. of S}

Hypergraphic polymatroid function

fix hypergraph $G=(V,E)$



$$f(S) = n - \left(\begin{array}{c}\#\text{ connected}\\ \text{components of } S\end{array}\right)$$

$$\text{span}(S) = \{\text{hyperedges } \overset{\text{(all endpoints)}}{\text{connected}} \text{ by } S\}$$

$$\text{Quotient } Q = E \setminus \text{span}(S)$$

$$= \{\text{edges cut by conn. comp. of } S\}$$

rank of $f = n-1$ if G connected

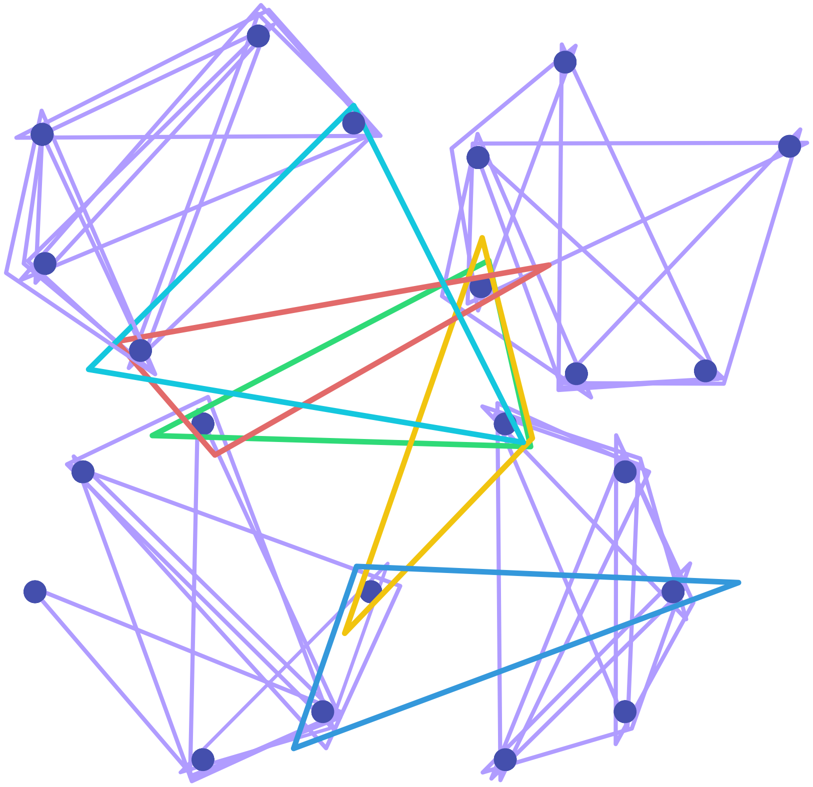
Let $f: 2^N \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in N$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq N$, $e \in N$, $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{\overset{\text{(including } S)}{e \in N}: f(e|S) = 0\}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = N \setminus Q$ closed i.e., $Q = N \setminus \text{span}(S)$ for some S
- "rank of f" $= f(N)$

$f(S) = n - \left(\begin{array}{c} \text{\# connected} \\ \text{components of } S \end{array} \right)$

Hypergraphic polymatroid function

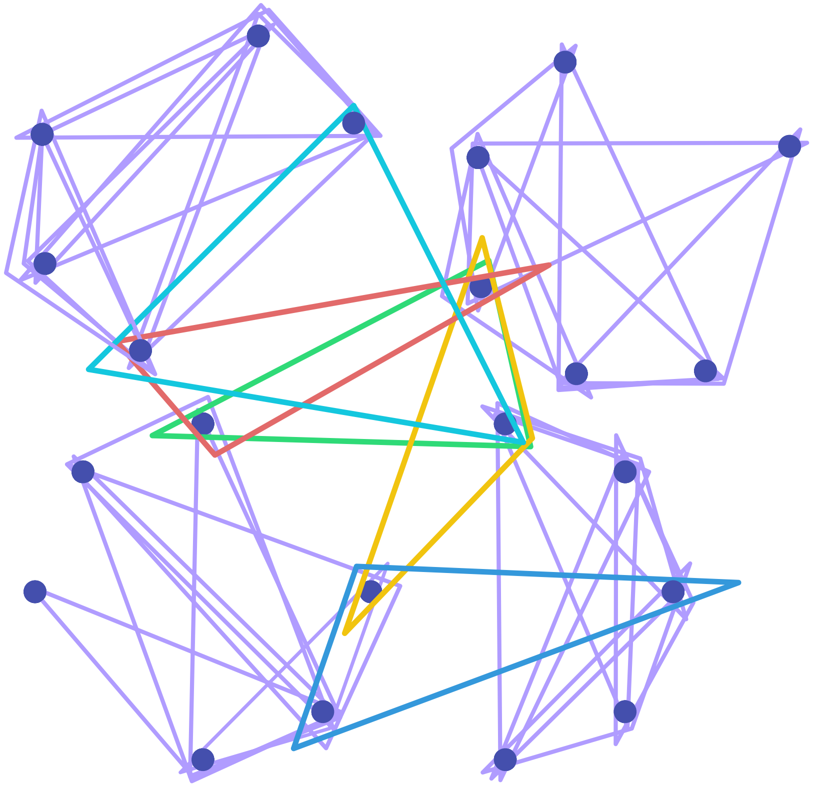
Quotient $Q = E \setminus \mathrm{span}(S) = \left\{ \begin{array}{c} \text{edges cut by} \\ \text{conn. comp. of } S \end{array} \right\}$



$f(S) = n - \left(\begin{matrix} \#\text{ connected} \\ \text{components of } S \end{matrix} \right)$

Hypergraphic polymatroid function

Quotient $Q = E \setminus \text{span}(S) = \left\{ \begin{matrix} \text{edges cut by} \\ \text{conn. comp. of } S \end{matrix} \right\}$

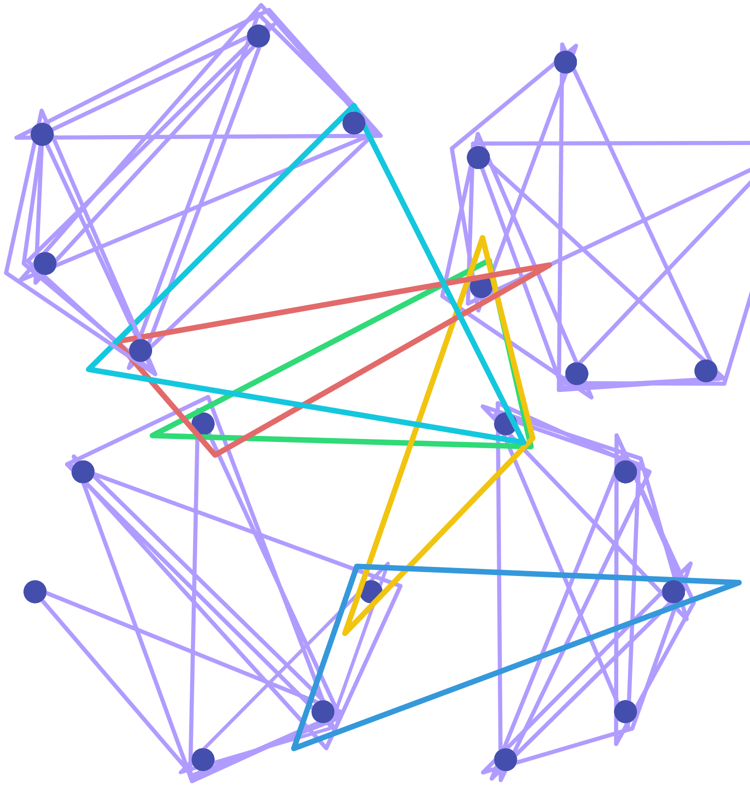


quotients = k-cuts (for varying k)

$f(S) = n - \left(\begin{matrix} \text{\# connected} \\ \text{components of } S \end{matrix} \right)$

Hypergraphic polymatroid function

Quotient $Q = E \setminus \text{span}(S) = \left\{ \begin{matrix} \text{edges cut by} \\ \text{conn. comp. of } S \end{matrix} \right\}$



quotients = k-cuts (for varying k)

k-cuts include 2-cuts

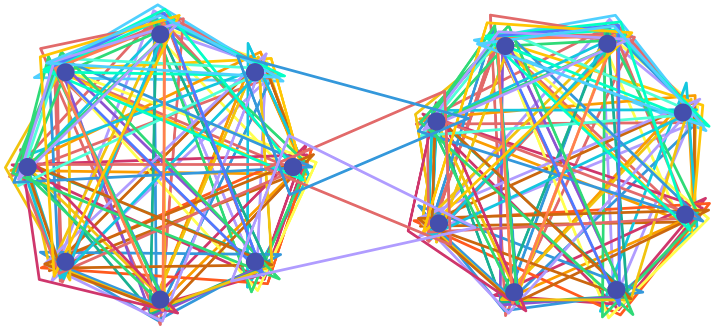
unlike graphs,

k-cut ≠ (half of) sum of 2-cuts over conn. comp.

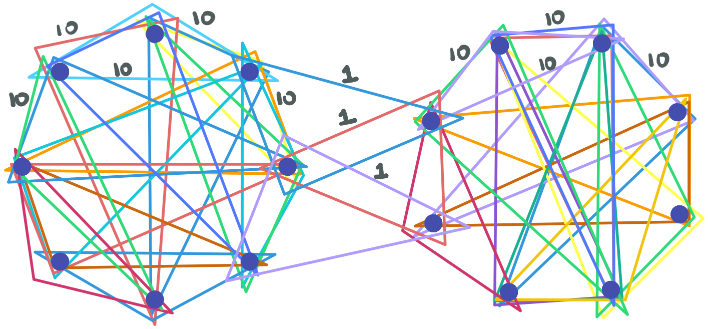
Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,

  $$\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,

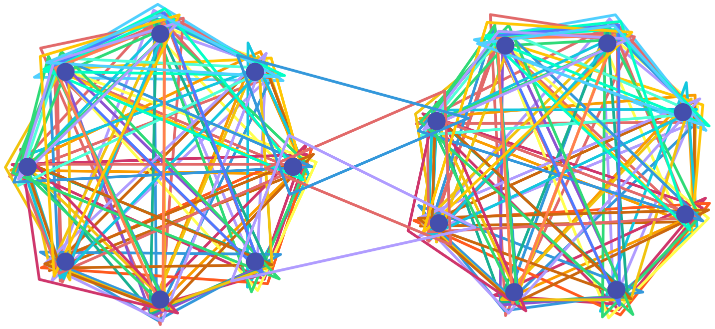
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{\overset{\text{(including } S)}{e \in \mathcal{N}}: f(e|S) = 0\}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed

  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

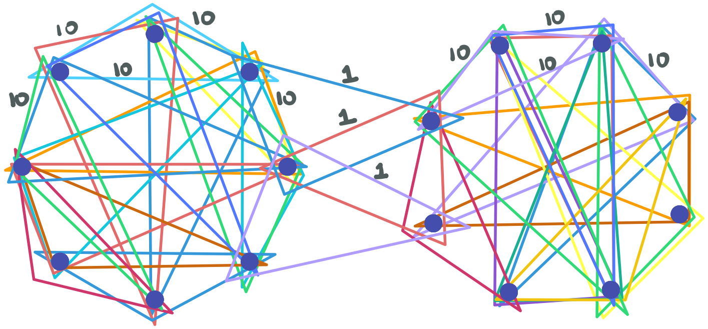
weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Theorem

let $r = f(\mathcal{N})$.

Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

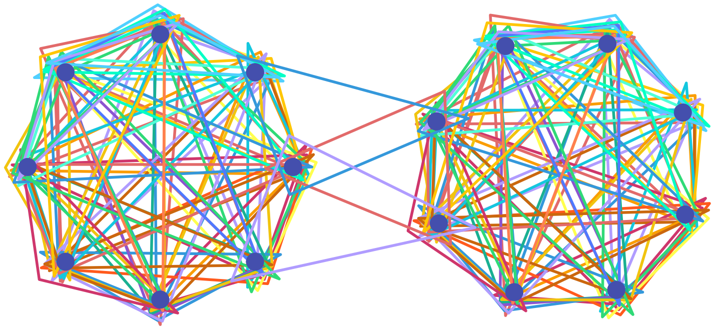
Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ e \in \mathcal{N} : f(e|S) = 0 \}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

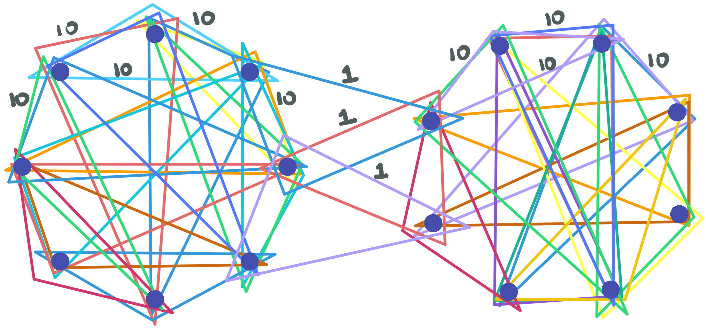
Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) $\text{support}(\tilde{w})$ small

(b) all quotients have similar weight as w/ w.

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ \overset{\text{(including } S)}{e \in \mathcal{N}} : f(e|S) = 0 \}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Theorem

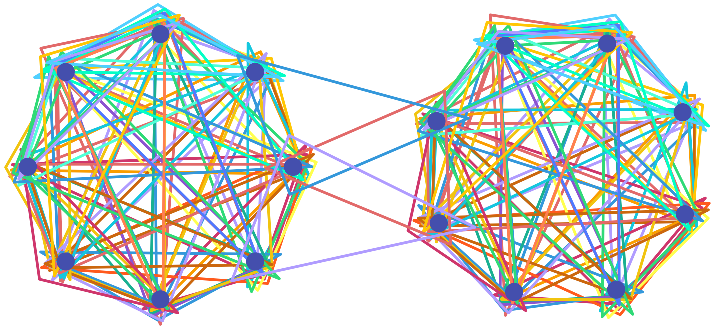
let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(rn)/\varepsilon^2)$
-

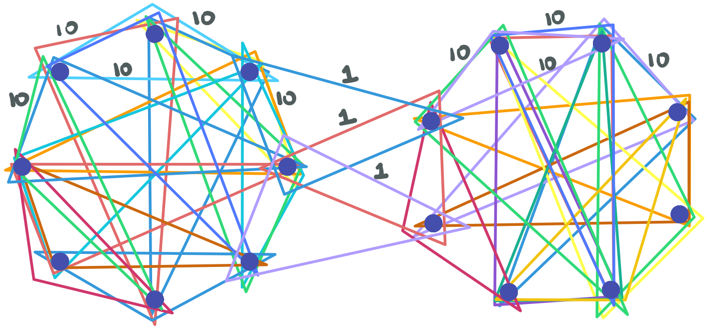
Submodular quotient sparsification

Input: $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ $\left(\begin{smallmatrix}\text{normalized}\\ \text{monotone}\\ \text{submodular}\end{smallmatrix}\right)$

weights $w : \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w} : \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(rn)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)\, w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)\, w(Q)$

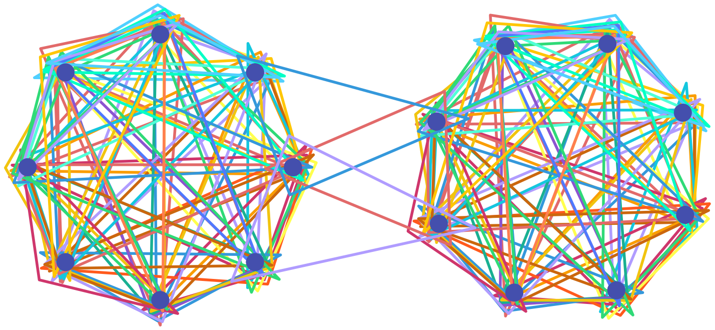
Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ \overset{\text{(including S)}}{e \in \mathcal{N}} : f(e|S) = 0 \}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\bar{Q} = \mathcal{N} \setminus Q$ closed i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

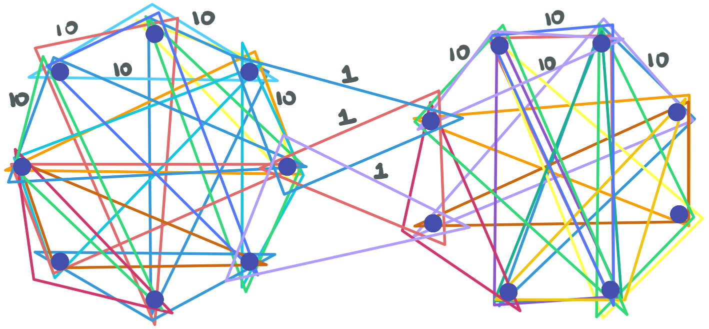
Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ \overset{\text{(including } S)}{e \in \mathcal{N}} : f(e|S) = 0 \}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed

  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(rn)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon) w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon) w(Q)$

(w/ high prob., rand. poly time, w/ oracle access to f)

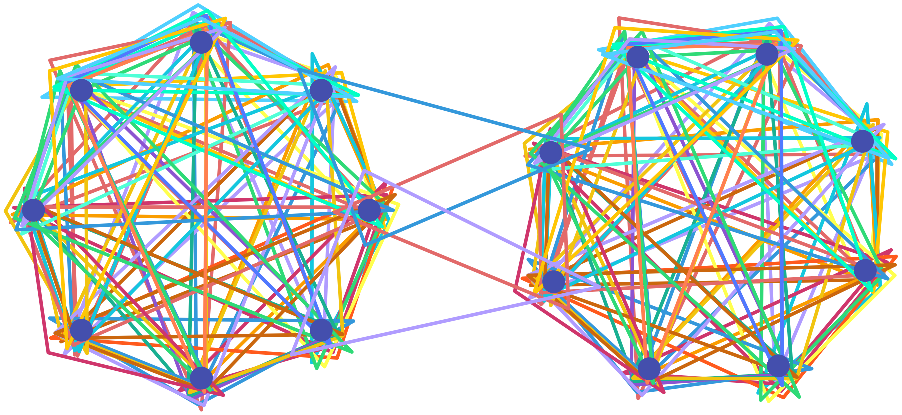
Hypergraph k-cut sparsification

$p =$ total size $= \sum_{e\in E} |e|$

Input: Hypergraph $G=(V,E)$

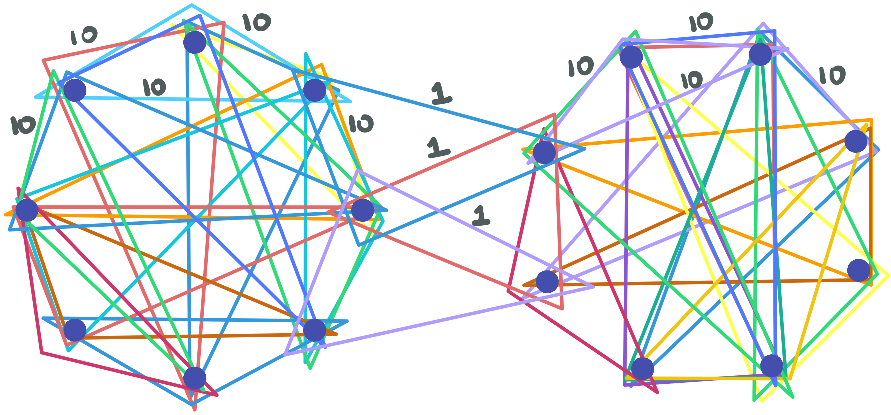
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in \tilde{E}$



s.t. (a) $|\tilde{E}|$ small

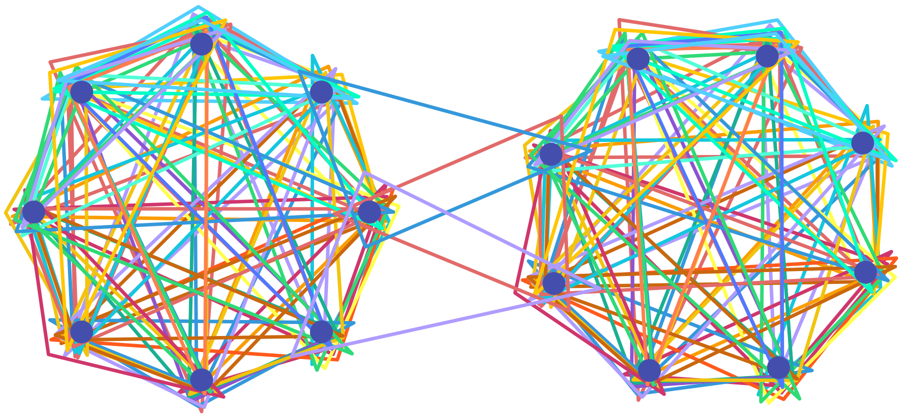
(b) all k-cuts have similar weight as in G

Hypergraph k-cut sparsification

p = total size = $\sum_{e \in E} |e|$

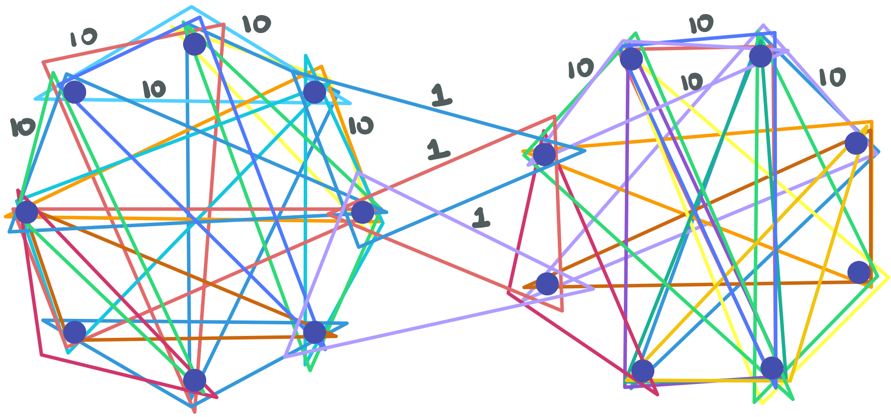
Input: Hypergraph $G=(V,E)$ $n=|V|$ $m=|E|$

$w(e)>0$ for $e \in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e \in \tilde{E}$



s.t. (a) $|\tilde{E}|$ small

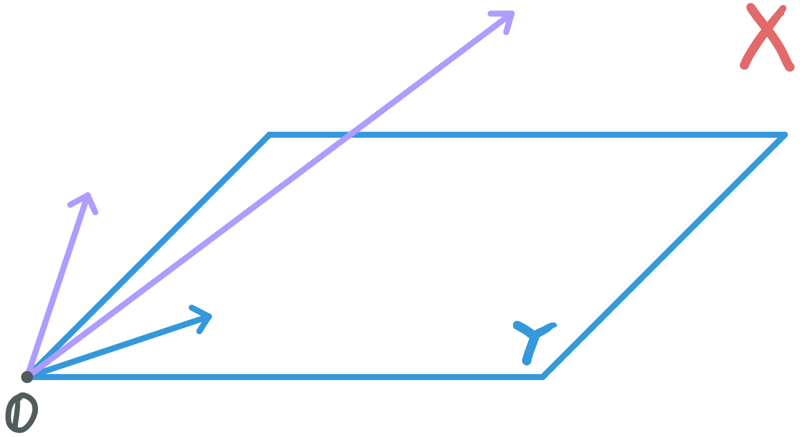
(b) all k-cuts have similar weight as in G

Theorem

- $|\tilde{E}| = O(n \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all k-cuts • $(1-\varepsilon)\sum_{e \in \partial(S_1,\dots,S_k)} \tilde{w}(e) \le \sum_{e \in \partial(S_1,\dots,S_k)} w(e) \le (1+\varepsilon) \sum_{e \in \partial(S_1,\dots,S_k)} \tilde{w}(e)$

(w/ high prob., in randomized $\tilde{O}(p)$ time)

Vector quotient spaces



$f : 2^{N} \to \mathbb{R}_{\geq 0}$ $\begin{pmatrix}\text{normalized} \\ \text{monotone} \\ \text{submodular}\end{pmatrix}$

weights $w : N \to \mathbb{R}_{>0}$

let $r = f(N)$.

Theorem: $\tilde{w}$ s.t.

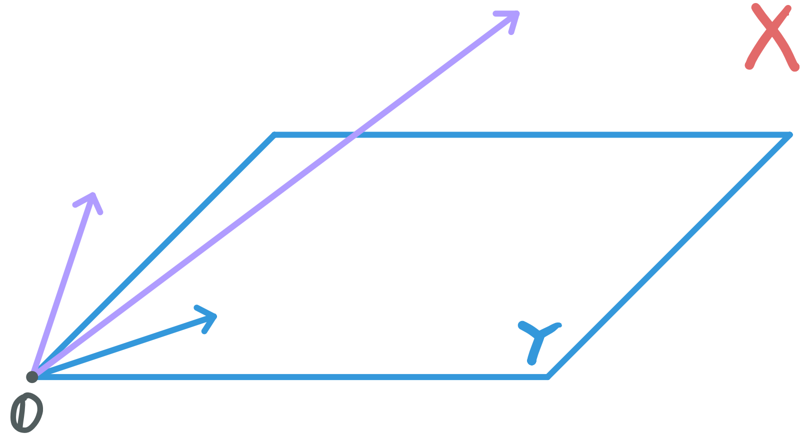
- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^{N} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: N \to \mathbb{R}_{>0}$

let $r = f(N)$.

Theorem: $\tilde{w}$ s.t.

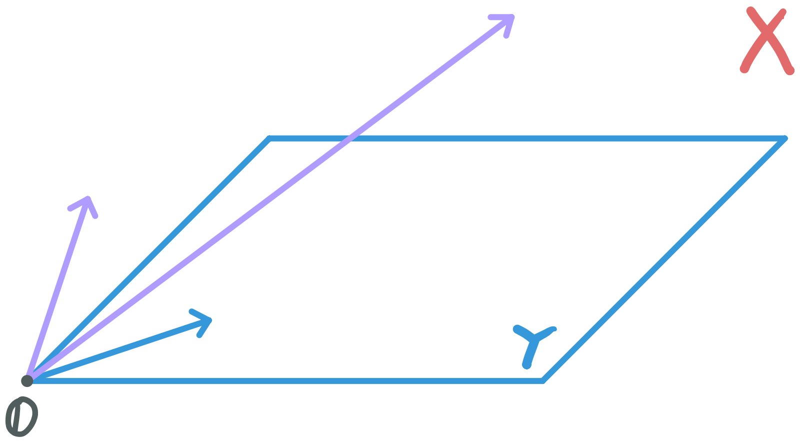
- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^N \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: N \to \mathbb{R}_{>0}$

let $r = f(N)$.

Theorem: $\tilde{w}$ s.t.

- $|\mathrm{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

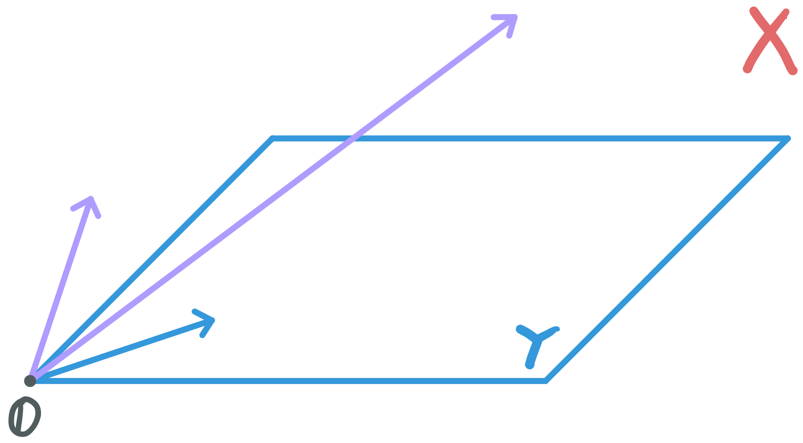
Theorem: Given n vectors $X \subseteq \mathbb{R}^d$, weights $w: X \to \mathbb{R}_{>0}$:

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^N \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: N \to \mathbb{R}_{>0}$

let $r = f(N)$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem: Given n vectors $X \subseteq \mathbb{R}^d$, weights $w: X \to \mathbb{R}_{>0}$:

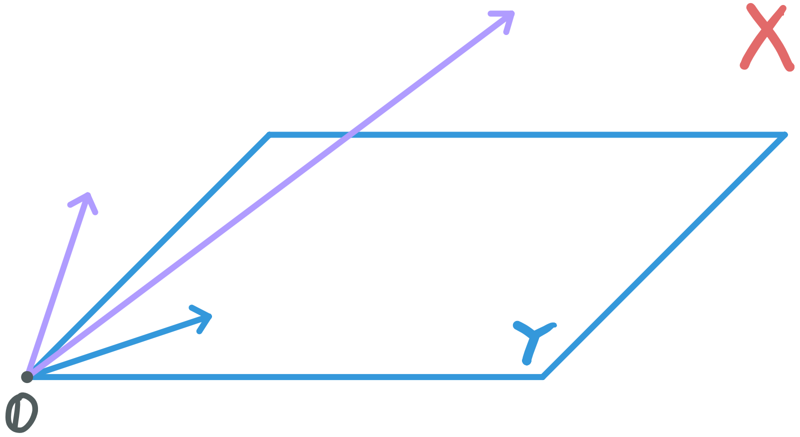
- $\tilde{X} \subseteq X$, $\tilde{w}: \tilde{X} \to \mathbb{R}_{>0}$
- $|\tilde{X}| \leq O(d \ln(n)/\varepsilon^2)$

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^N \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: N \to \mathbb{R}_{>0}$

let $r = f(N)$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem: Given n vectors $X \subseteq \mathbb{R}^d$, weights $w: X \to \mathbb{R}_{>0}$:

- $\tilde{X} \subseteq X$, $\tilde{w}: \tilde{X} \to \mathbb{R}_{>0}$
- $|\tilde{X}| \leq O(d \ln(n)/\varepsilon^2)$
- for all quotient spaces Q,

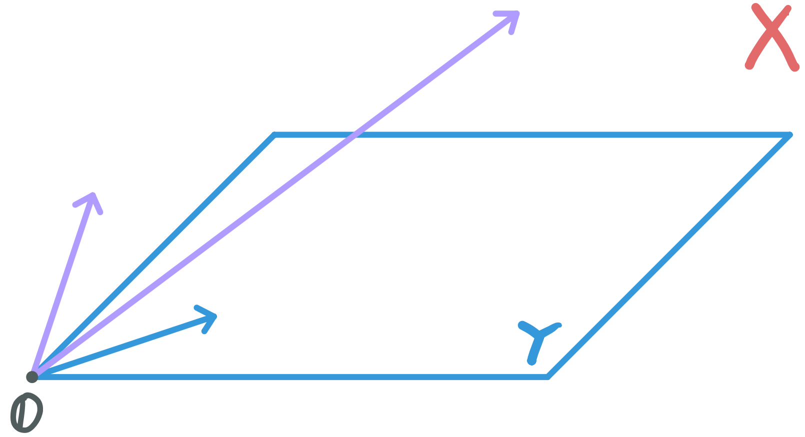
$$(1-\varepsilon)\, w(X \cap Q) \leq \tilde{w}(\tilde{X} \cap Q) \leq (1+\varepsilon)\, w(X \cap Q)$$

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem: Given n vectors $X \subseteq \mathbb{R}^d$, weights $w: X \to \mathbb{R}_{>0}$:

- $\tilde{X} \subseteq X$, $\tilde{w}: \tilde{X} \to \mathbb{R}_{>0}$
- $|\tilde{X}| \leq O(d \ln(n)/\varepsilon^2)$
- for all quotient spaces Q,

$$(1-\varepsilon)\, w(X \cap Q) \leq \tilde{w}(\tilde{X} \cap Q) \leq (1+\varepsilon)\, w(X \cap Q)$$

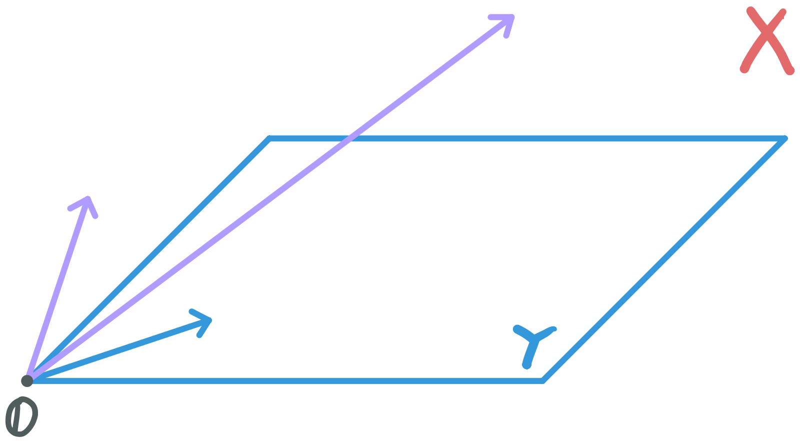
why?

Vector quotient spaces

vector space X

subspace Y

quotient space X/Y



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem: Given n vectors $X \subseteq \mathbb{R}^d$, weights $w: X \to \mathbb{R}_{>0}$:

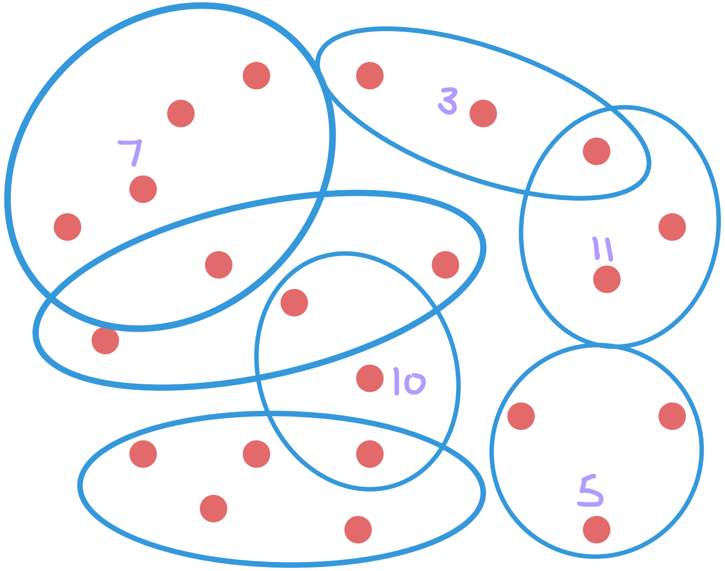
- $\tilde{X} \subseteq X$, $\tilde{w}: \tilde{X} \to \mathbb{R}_{>0}$
- $|\tilde{X}| \leq O(d \ln(n)/\varepsilon^2)$
- for all quotient spaces Q,

$$(1-\varepsilon)\, w(X \cap Q) \leq \tilde{w}(\tilde{X} \cap Q) \leq (1+\varepsilon)\, w(X \cap Q)$$

why?

- linear matroid: $\mathcal{N} = X$, $\mathcal{I} =$ independent sets of vectors
- quotients of $\uparrow$ = quotient subspaces

Weighted coverage



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ $\left(\begin{array}{c}\text{normalized}\\ \text{monotone}\\ \text{submodular}\end{array}\right)$

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

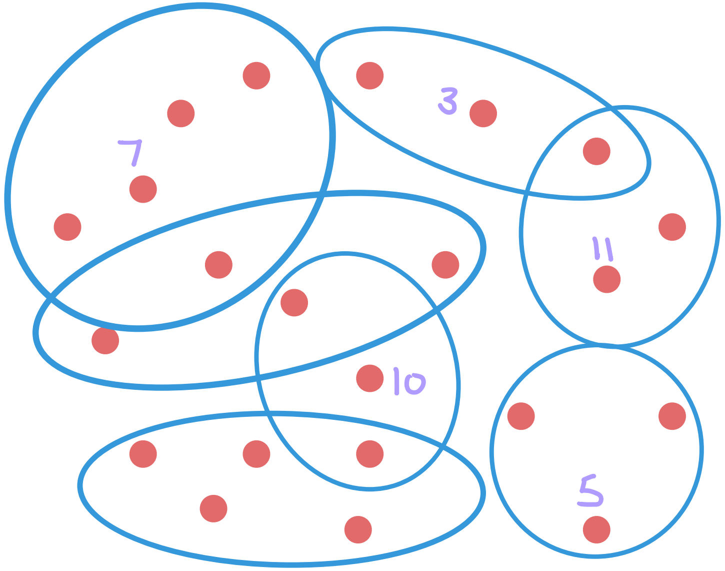
let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Weighted coverage

- n elements $\mathcal{N}$
- m sets $F \leq 2^{\mathcal{N}}$
- $\left(\begin{matrix}\text{elem.}\\ \text{weights}\end{matrix}\right)$ $w: \mathcal{N} \to \mathbb{R}_{>0}$



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ $\left(\begin{matrix}\text{normalized}\\ \text{monotone}\\ \text{submodular}\end{matrix}\right)$

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

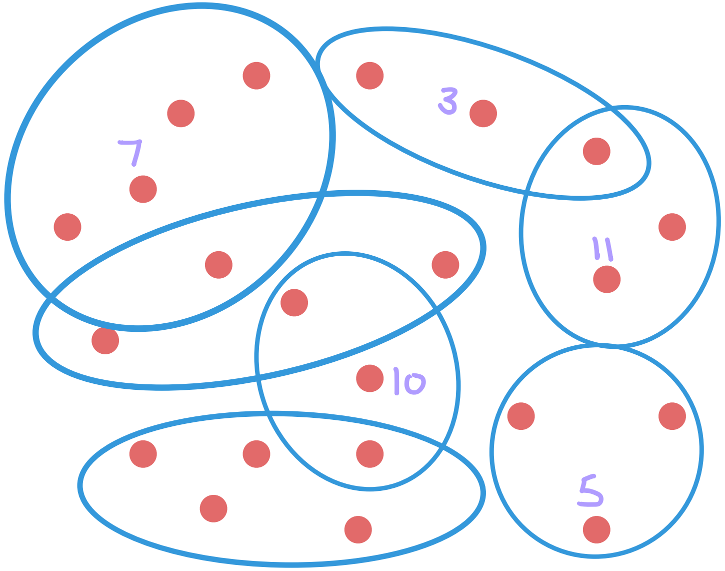
let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Weighted coverage

- n elements $\mathcal{N}$
- m sets $\mathcal{F} \subseteq 2^{\mathcal{N}}$
- $\left(\begin{matrix}\text{elem.}\\ \text{weights}\end{matrix}\right)$ $w: \mathcal{N} \to \mathbb{R}_{>0}$



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ $\left(\begin{matrix}\text{normalized}\\ \text{monotone}\\ \text{submodular}\end{matrix}\right)$

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

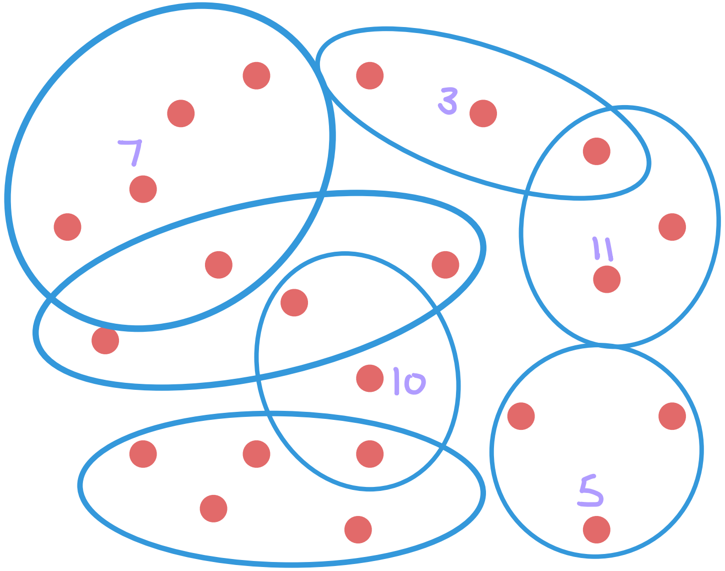
Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem

Weighted coverage

- n elements $\mathcal{N}$
- m sets $\mathcal{F} \subseteq 2^{\mathcal{N}}$
- (elem. weights) $w : \mathcal{N} \to \mathbb{R}_{>0}$



$f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w : \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

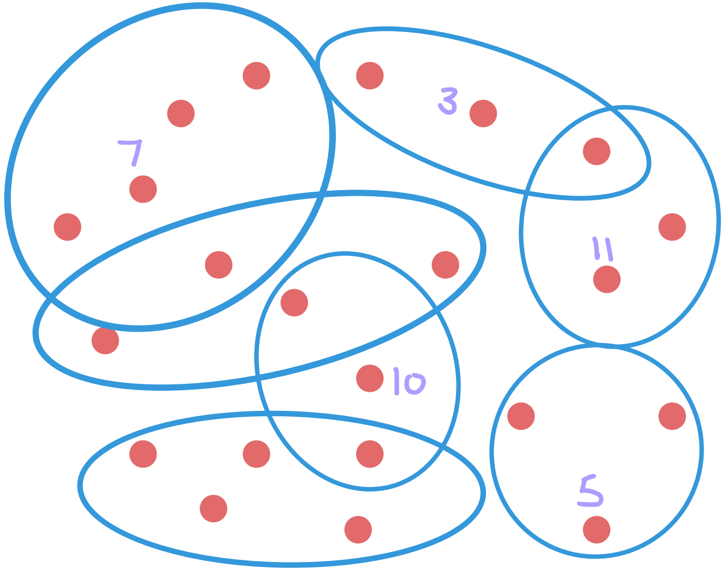
- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem

- $\tilde{\mathcal{N}} \subseteq \mathcal{N}$, $|\tilde{\mathcal{N}}| \leq O(m \log(n)/\varepsilon^2)$

Weighted coverage

- n elements $\mathcal{N}$
- m sets $\mathcal{F} \subseteq 2^{\mathcal{N}}$
- (elem. weights) $w: \mathcal{N} \to \mathbb{R}_{>0}$



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

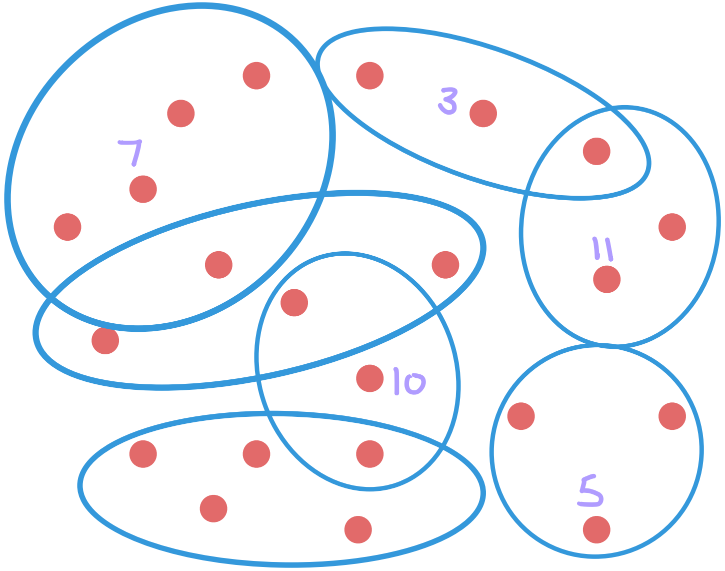
Theorem

- $\tilde{\mathcal{N}} \subseteq \mathcal{N}$, $|\tilde{\mathcal{N}}| \leq O(m \log(n)/\varepsilon^2)$
- for all $S_1, \ldots, S_k \in \mathcal{F}$,

$$(1-\varepsilon)\, w(S_1 \cup \cdots \cup S_k) \leq \tilde{w}((S_1 \cup \cdots \cup S_k) \cap \tilde{\mathcal{N}}) \leq (1+\varepsilon)\, w(S_1 \cup \cdots \cup S_k)$$

Weighted coverage

- n elements $\mathcal{N}$
- m sets $F \subseteq 2^{\mathcal{N}}$
- $\left(\begin{smallmatrix}\text{elem.}\\\text{weights}\end{smallmatrix}\right)$ $w : \mathcal{N} \to \mathbb{R}_{>0}$



$f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w : \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem

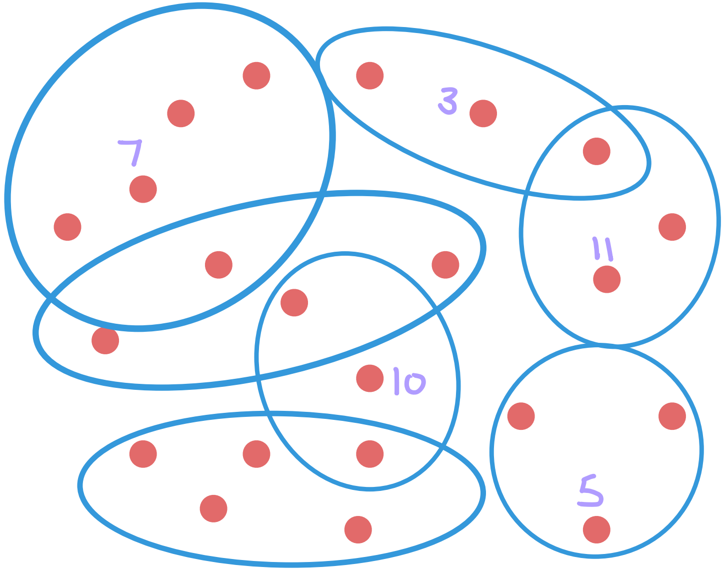
- $\tilde{\mathcal{N}} \subseteq \mathcal{N}$, $|\tilde{\mathcal{N}}| \leq O(m \log(n)/\varepsilon^2)$
- for all $S_1, \dots, S_k \in F$,

$$(1-\varepsilon)\, w(S_1 \cup \cdots \cup S_k) \leq \tilde{w}((S_1 \cup \cdots \cup S_k) \cap \tilde{\mathcal{N}}) \leq (1+\varepsilon)\, w(S_1 \cup \cdots \cup S_k)$$

why?

Weighted coverage

- n elements $\mathcal{N}$
- m sets $F \subseteq 2^{\mathcal{N}}$
- (elem. weights) $w: \mathcal{N} \to \mathbb{R}_{>0}$



$f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

let $r = f(\mathcal{N})$.

Theorem: $\tilde{w}$ s.t.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients

Theorem

- $\tilde{\mathcal{N}} \subseteq \mathcal{N}$, $|\tilde{\mathcal{N}}| \leq O(m \log(n)/\varepsilon^2)$
- for all $S_1, \dots, S_k \in F$,

$$(1-\varepsilon)\, w(S_1 \cup \cdots \cup S_k) \leq \tilde{w}((S_1 \cup \cdots \cup S_k) \cap \tilde{\mathcal{N}}) \leq (1+\varepsilon)\, w(S_1 \cup \cdots \cup S_k)$$

why?

hitting set fn: define $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ by

$$f(X) = |S \in F : S \cap X \neq \emptyset|$$

Quotients of f = unions of F

high-level ideas?

relation to Benczúr-Karger?

how to do it fast?

additional constraint: be concrete

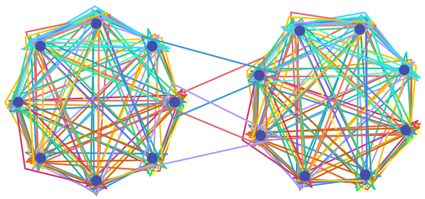
(no more "submodular f")

Focus on graph cuts, graphic matroid

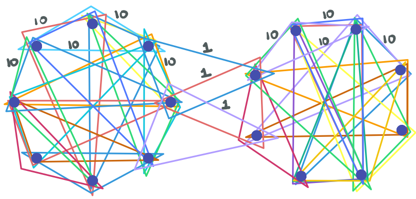
Submodular quotient sparsification

Input: $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w : \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w} : \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ e \in \mathcal{N} : f(e|S) = 0 \}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\overline{Q} = \mathcal{N} \setminus Q$ closed i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

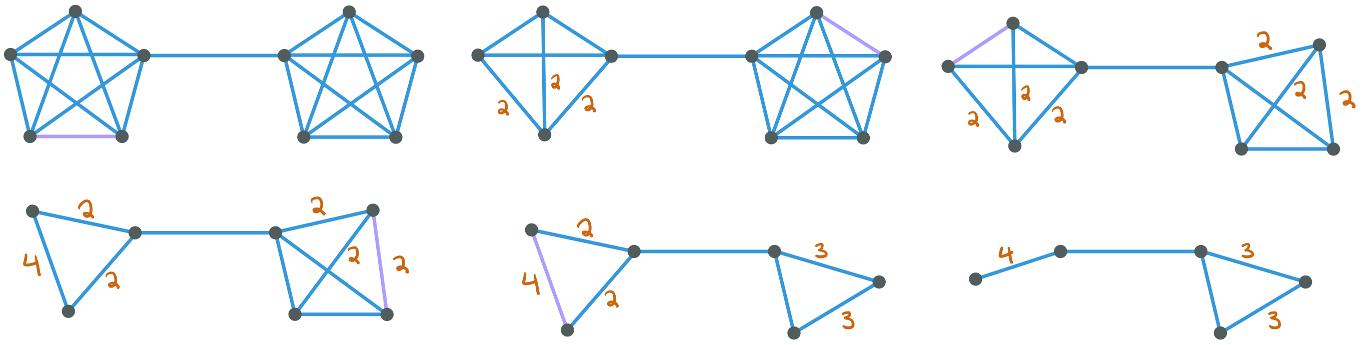
Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)\, w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)\, w(Q)$

(w/ high prob., rand. poly time, w/ oracle access to f)

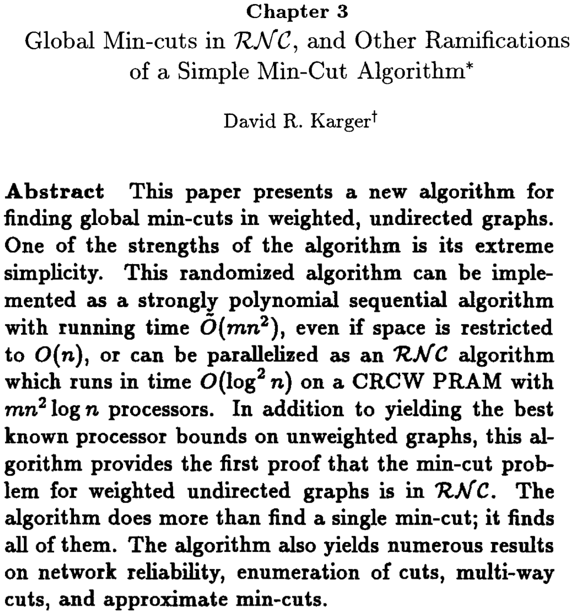
Karger's random contraction alg.



Returns fixed min-cut w/ prob $\Omega(1/n^2)$

$\Rightarrow O(n^2)$ min-cuts!

more generally, # α-APX cuts $\le n^{O(\alpha)}$



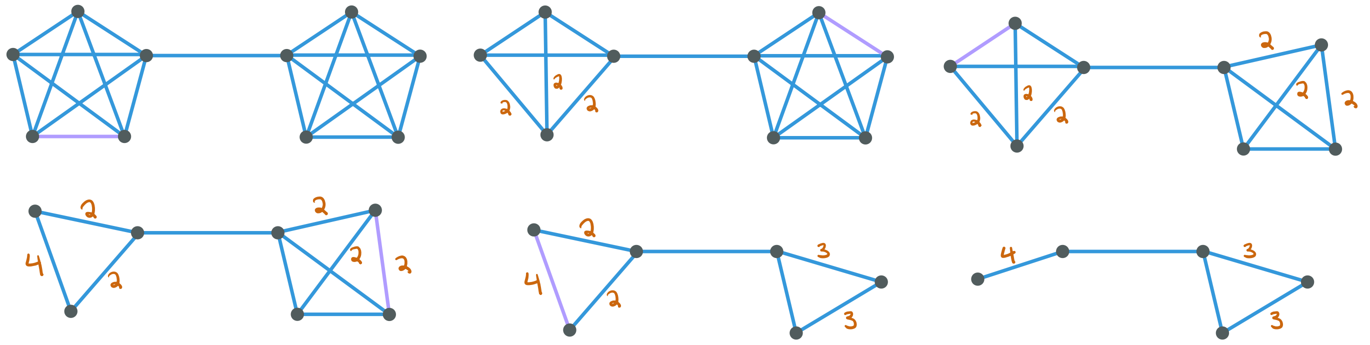
Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger†

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

Karger's random contraction alg.



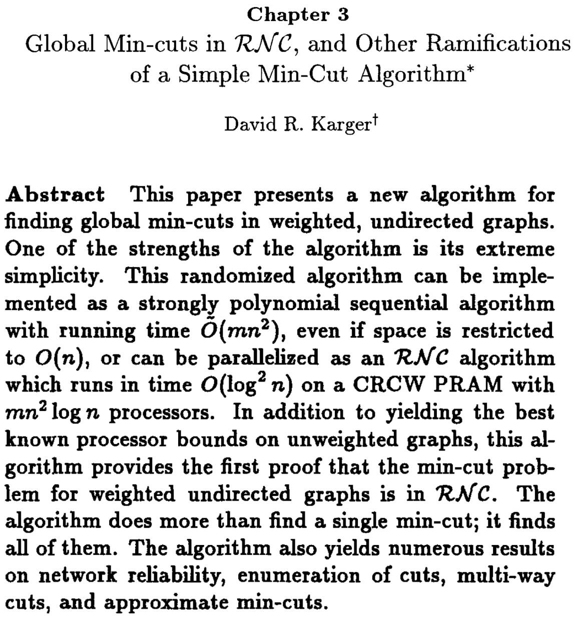
Returns fixed min-cut w/ prob $\Omega(1/n^2)$

$\Rightarrow O(n^2)$ min-cuts!

more generally, # α-APX cuts $\le n^{O(\alpha)}$

$\Rightarrow$ we can union bound over APX-min-cuts

$\Rightarrow$ weaker uniform sparsification that $(1+\varepsilon)$-APX the min-cut

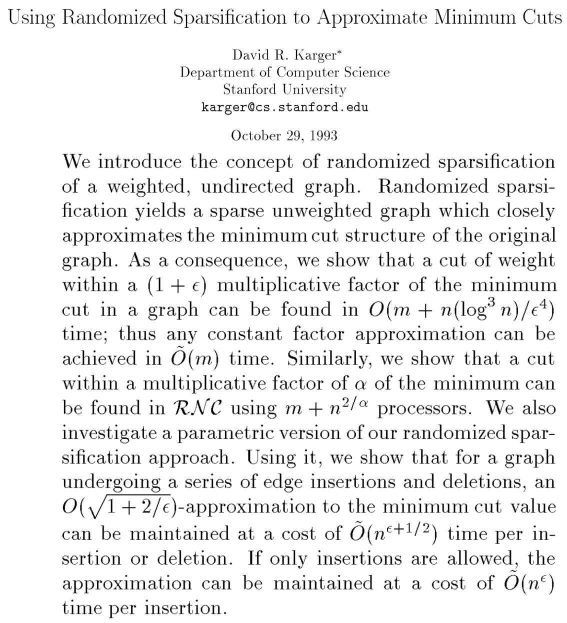


Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger†

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.



Using Randomized Sparsification to Approximate Minimum Cuts

David R. Karger*
Department of Computer Science
Stanford University
karger@cs.stanford.edu

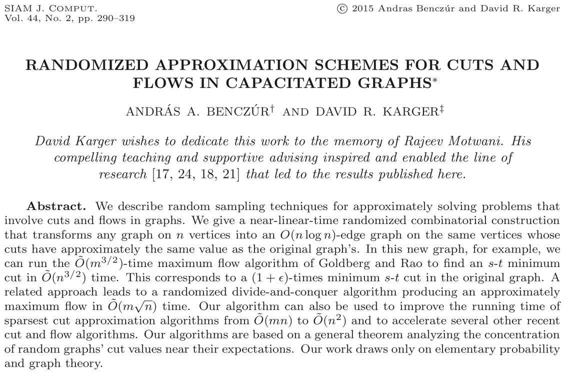
October 29, 1993

We introduce the concept of randomized sparsification of a weighted, undirected graph. Randomized sparsification yields a sparse unweighted graph which closely approximates the minimum cut structure of the original graph. As a consequence, we show that a cut of weight within a $(1+\epsilon)$ multiplicative factor of the minimum cut in a graph can be found in $O(m + n(\log^3 n)/\epsilon^4)$ time; thus any constant factor approximation can be achieved in $\tilde{O}(m)$ time. Similarly, we show that a cut within a multiplicative factor of α of the minimum can be found in $\mathcal{RNC}$ using $m + n^{2/\alpha}$ processors. We also investigate a parametric version of our randomized sparsification approach. Using it, we show that for a graph undergoing a series of edge insertions and deletions, an $O(\sqrt{1+2/\epsilon})$-approximation to the minimum cut value can be maintained at a cost of $\tilde{O}(n^{\epsilon+1/2})$ time per insertion or deletion. If only insertions are allowed, the approximation can be maintained at a cost of $\tilde{O}(n^{\epsilon})$ time per insertion.

Benczúr, Karger 2002

(nonuniformly) samples edges by "strong connectivity"

$e=\{u,v\}$ has strong connectivity λ if it is contained in a subgraph w/ min-cut λ



SIAM J. Comput.
Vol. 44, No. 2, pp. 290–319

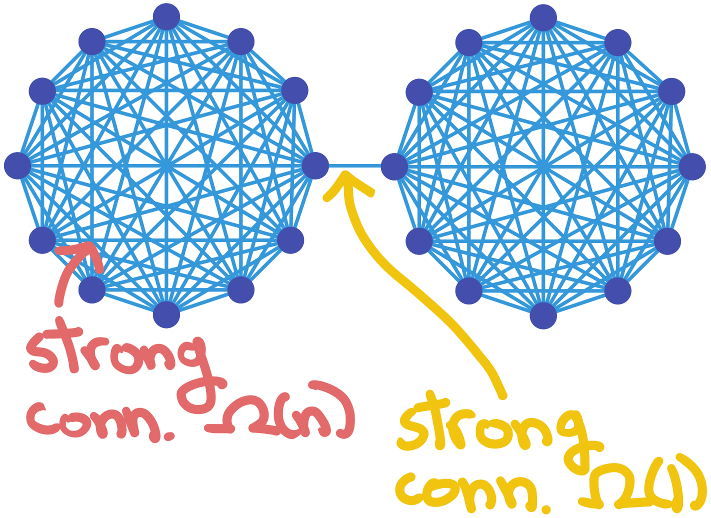
© 2015 Andras Benczúr and David R. Karger

RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

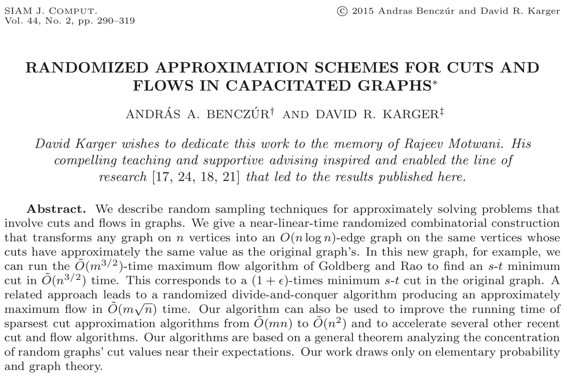
ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.



Benczúr, Karger 2002



SIAM J. COMPUT.
Vol. 44, No. 2, pp. 290–319

© 2015 Andras Benczúr and David R. Karger

RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

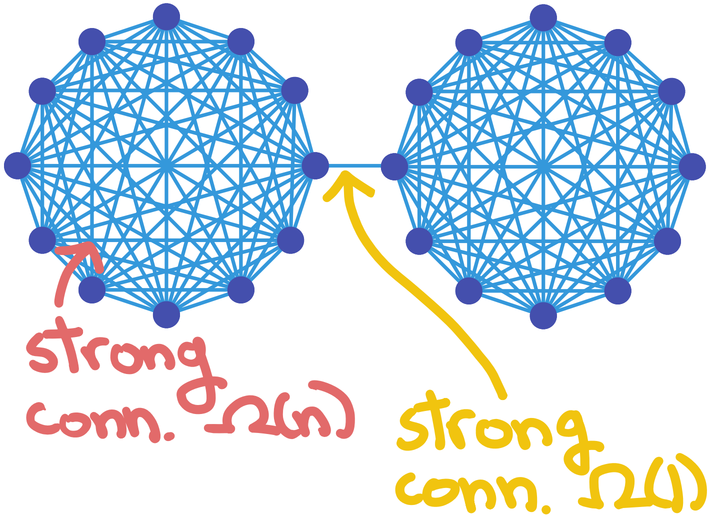
ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

(nonuniformly)
samples edges by "strong connectivity"

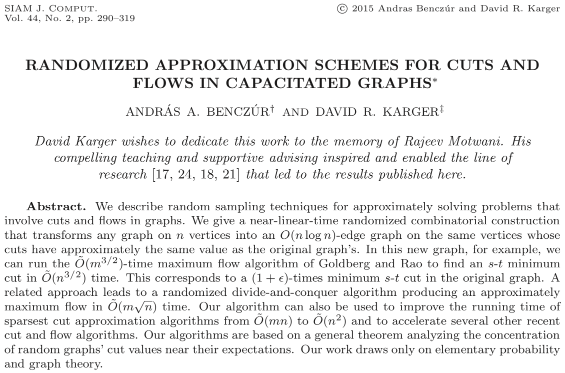
$e = \{u, v\}$ has strong connectivity λ if it is contained in a subgraph w/ min-cut λ



let k_e = strong conn. e, $p_e = \Omega\left(\frac{w_e}{k_e} \frac{\log n}{\varepsilon^2}\right)$

- $\sum_{e \in E} \frac{w_e}{k_e} = O(n)$
- sampling each $e \in E$ w/ prob $p_e \Rightarrow (1 \pm \varepsilon)$-APX all cuts

Benczúr, Karger 2002



SIAM J. COMPUT.
Vol. 44, No. 2, pp. 290–319

© 2015 Andras Benczúr and David R. Karger

RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

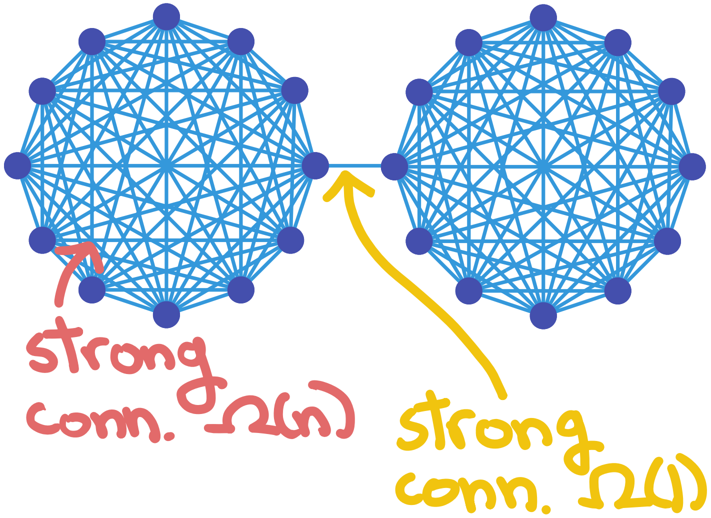
ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

(nonuniformly)
samples edges by "strong connectivity"

$e=\{u,v\}$ has strong connectivity λ if it is contained in a subgraph w/ min-cut λ



let k_e = strong conn. e, $p_e = \Omega\left(\frac{w_e}{k_e}\frac{\log n}{\varepsilon^2}\right)$

- $\sum_{e\in E} \frac{w_e}{k_e} = O(n)$
- sampling each $e \in E$ w/ prob $p_e \Rightarrow (1+\varepsilon)$-APX all cuts

to compute approximations $\tilde{k}_e \lesssim k_e$ (w/ $\sum 1/\tilde{k}_e = O(\sum 1/k_e)$)

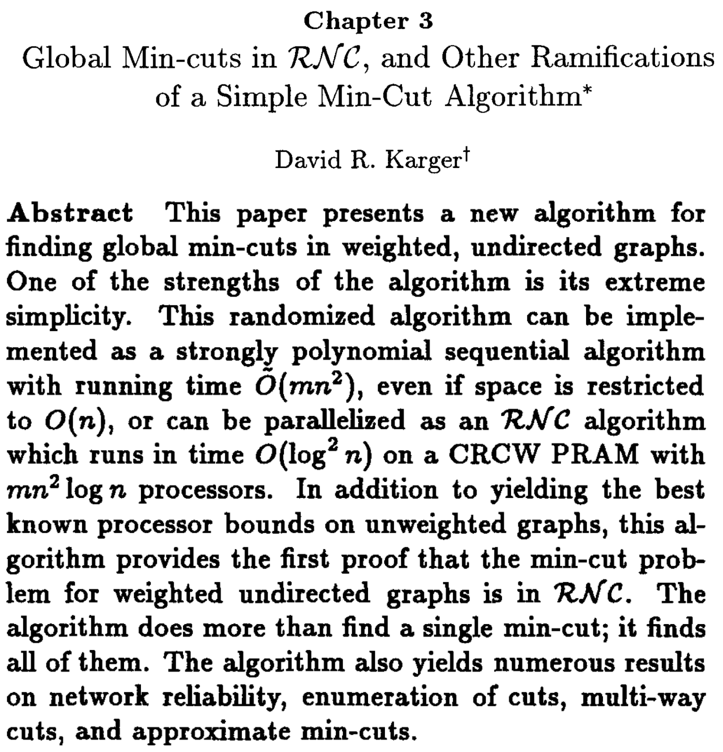
- Nagamochi-Ibaraki greedy tree packing

(breaks graph into components, recurse)

Karger '93

$n^{O(\alpha)}$ α-APX min-cuts

uniform sample preserves min-cut (approximately)



Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm[*]

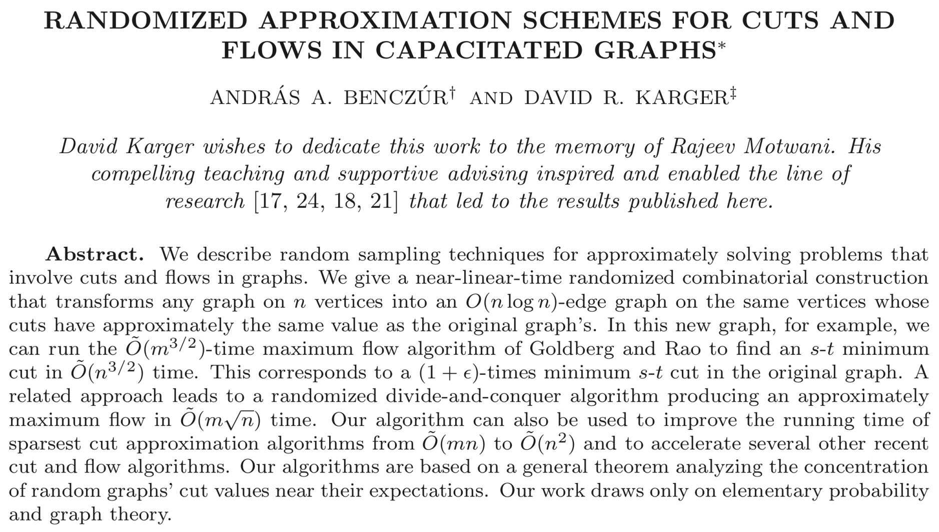
David R. Karger[†]

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

Benczúr, Karger '02

sample inversely proportional to "strong connectivity"

approx. decompose G along greedy tree packings á la Nagamochi-Ibaraki



RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS[*]

ANDRÁS A. BENCZÚR[†] AND DAVID R. KARGER[‡]

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

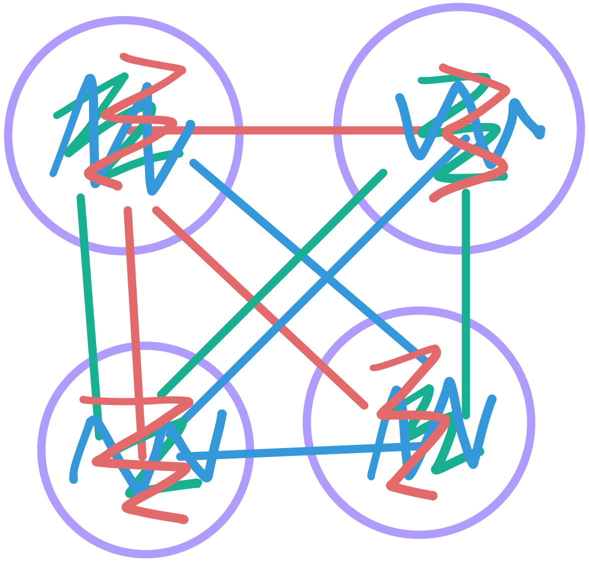
$$\max \sum_{t} x_t \text{ over } x \in \mathbb{R}^{\mathcal{T}}_{\geq 0}$$

$$\sum_{t : e \in t} x_t \leq w_e \;\; \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1,\dots,S_k))}{k-1}$$

over all partitions (any k) $S_1,\dots,S_k$ of V



weak duality: $\sum_{t} x_t \leq \frac{w(\partial(S_1,\dots,S_k))}{k-1}$

(each tree has ≥k-1 edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_{t} x_t = \min \frac{w(\partial(S_1,\dots,S_k))}{k-1}$$

"network strength"

Matroid base packing & covering

matroid $\mathcal{M}=(\mathcal{N},\mathcal{I})$, $w:\mathcal{N}\to\mathbb{R}_{>0}$

$\mathcal{B}=$ bases of $\mathcal{M} \subseteq 2^{\mathcal{N}}$ (max. ind. sets)

(e.g., spanning trees in graphic matroid)

$\mathcal{M}=(\overset{\text{"groundset"}}{\mathcal{N}},\mathcal{I})$ "independent sets"

1. $\emptyset\in\mathcal{I}$
2. $S\subseteq T, T\in\mathcal{I} \Rightarrow S\in\mathcal{I}$
3. $S,T\in\mathcal{I}$, $|S|<|T| \Rightarrow$ $e\in T\setminus S$ s.t. $S+e\in\mathcal{I}$

maximal = maximum

"base" = max ind. set

Max base packing

$$\max \sum_{B} x_B \text{ w/ } x:\mathcal{B}\to\mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{B:e\in B} x_e \leq w_e \quad \forall e\in\mathcal{N}$$

Min ratio quotient

$$\min_{S} \frac{w(\mathcal{N})-w(S)}{\text{rank}(\mathcal{N})-\text{rank}(S)}$$

Weak duality: $\sum_{B} x_B \leq \frac{w(\mathcal{N})-w(S)}{\text{rank}(\mathcal{N})-\text{rank}(S)}$

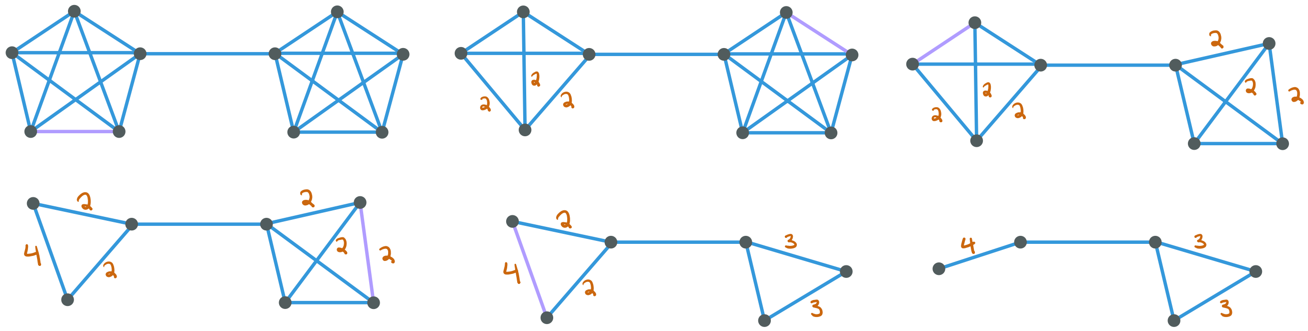
[each base uses rank(N)-rank(S) weight of N\S]

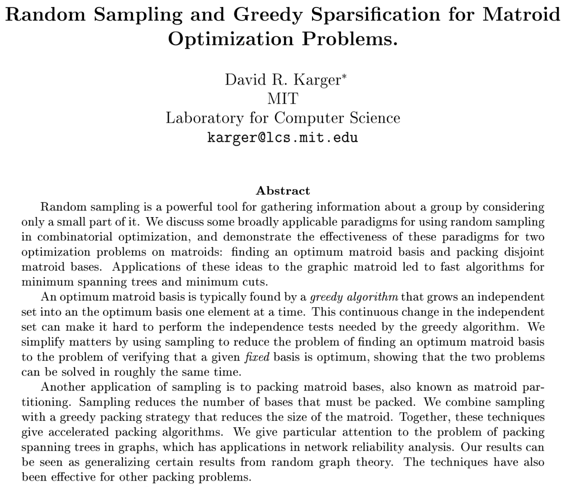
Edmonds '65: $\max \sum_{B} x_B = \min_{S} \frac{w(\mathcal{N})-w(S)}{\text{rank}(\mathcal{N})-\text{rank}(S)}$

↑

"matroid/submodular strength"

Karger '93, '97





Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

analyzed random contractions in (unweighted) matroids

$\Rightarrow$ $(rn)^{O(1)}$ min ratio quotients (k-cuts)

$(rn)^{O(\alpha)}$ α-APX min ratio quotients

$\Rightarrow$ uniform sampling (approximately) preserves min ratio quotient

$M = (\overset{\text{"groundset"}}{N}, \underset{\text{"independent sets"}}{I})$

1. $\emptyset \in I$
2. $S \subseteq T, T \in I \Rightarrow S \in I$
3. $S, T \in I$, $|S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S + e \in I$

maximal = maximum

"base" = max ind. set

$\text{span}(S) = \{e \in N : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \text{span}(S)$

Q quotient $\Leftrightarrow$ $\bar{Q}$ closed

i.e., $Q = V \setminus \text{span}(S)$

Max base packing

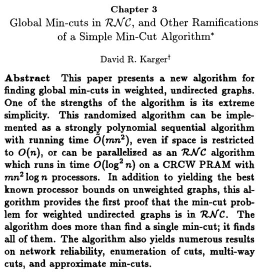
max $\sum_B x_B$ w/ $x : \mathcal{B} \to \mathbb{R}_{\geq 0}$

s.t. $\sum_{B : e \in B} x_e \leq w_e$ $\forall e \in N$

Min ratio quotient

$$\min_S \frac{w(N) - w(S)}{\text{rank}(N) - \text{rank}(S)}$$

Karger '93



Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger†

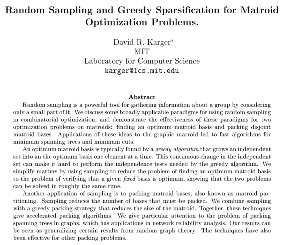
Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

$n^{O(\alpha)}$ α-APX min-cuts

uniform sample preserves min-cut (approximately)

Karger 93, 97

In unweighted matroids:



Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

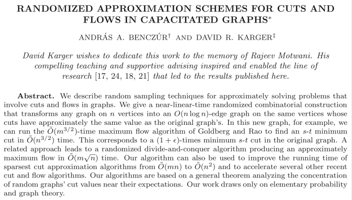
An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

$(rn)^{O(\alpha)}$ α-APX min quotients

uniform sample preserves min quotient (approximately)

Benczúr, Karger '02



RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

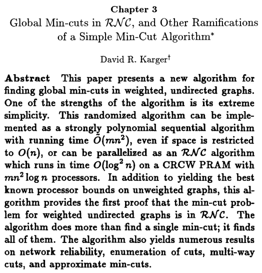
David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

sample inversely proportional to "strong connectivity"

approx. decompose G along greedy tree packings á la Nagamochi-Ibaraki

Karger '93



Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger[†]

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

$n^{O(\alpha)}$ α-APX min-cuts

u es

preserve min-cut

min-cut (approximately)

Karger 93, 97

In unweighted matroids:

Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

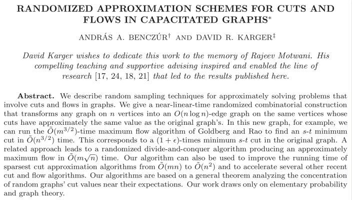
An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

$(rn)^{O(\alpha)}$ α-APX min quotients

preserve min-quotient

Benczúr, Karger '02



RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

ANDRÁS A. BENCZÚR[†] AND DAVID R. KARGER[‡]

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

Sample inversely proportional

to "st ity"

o

preserve all cuts

along greedy tree packings

á la Nagamochi-Ibaraki

"Strength decompositions" of $M = (N, I)$

$N = S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$

"Strength decompositions" of $M=(\mathcal{N},\mathcal{I})$

$$\mathcal{N} = S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$$

(a) $$S_{i+1} = \underset{T \subseteq S_i}{\operatorname{argmin}} \frac{w(S_i) - w(T)}{\operatorname{rank}(S_i) - \operatorname{rank}(T)}$$

submodular min.

$$\min_{T} \lambda^{*} \operatorname{rank}(T) - w(T)$$

w/ $\lambda^{*} = \text{OPT ratio}$

"Strength decompositions" of $M=(\mathcal{N},\mathcal{I})$

$$\mathcal{N} = S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$$

(a) $$S_{i+1} = \underset{T \subseteq S_i}{\operatorname{argmin}} \frac{w(S_i)-w(T)}{\mathrm{rank}(S_i)-\mathrm{rank}(T)}$$

submodular min.

$$\min_{T} \lambda^* \mathrm{rank}(T) - w(T)$$

w/ $\lambda^* =$ OPT ratio

(monotonicity)

(b) $$\frac{w(S_i)-w(S_{i+1})}{\mathrm{rank}(S_i)-\mathrm{rank}(S_{i+1})} \leq \frac{w(S_{i+1})-w(S_{i+2})}{\mathrm{rank}(S_{i+1})-\mathrm{rank}(S_{i+2})}$$

"α-APX Strength decompositions" of $M=(\mathcal{N},\mathcal{I})$

$$\mathcal{N} = S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$$

(a) $$\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \alpha \left[\min_{T \subseteq S_i} \frac{w(S_i)-w(T)}{\text{rank}(S_i)-\text{rank}(T)} \right]$$

e.g. O(1)-APX tree packing & dual

(monotonicity)

(b) $$\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \frac{w(S_{i+1})-w(S_{i+2})}{\text{rank}(S_{i+1})-\text{rank}(S_{i+2})}$$

"α-APX Strength decompositions" of $M=(\mathcal{N},\mathcal{I})$

$\mathcal{N} = S_0 \supseteq S_1 \supseteq \cdots \supseteq \boxed{S_i \supseteq S_{i+1}} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$

(a) $$\boxed{\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \alpha \left[\min_{T \subseteq S_i} \frac{w(S_i)-w(T)}{\text{rank}(S_i)-\text{rank}(T)} \right]}$$

(monotonicity)

(b) $$\boxed{\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \frac{w(S_{i+1})-w(S_{i+2})}{\text{rank}(S_{i+1})-\text{rank}(S_{i+2})}}$$

Sparsification algo

1. compute $O(1)$-APX strength decomp
   $(S_0 = \mathcal{N}) \supseteq S_1 \supseteq \cdots \supseteq S_k$

2. for each element $e \in \mathcal{N}$
   - suppose $e \in S_{i-1} \setminus S_i$

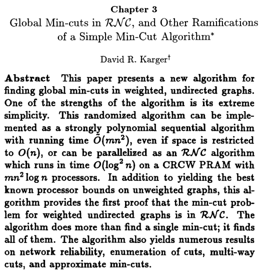
   ratio of quotient containing e

   - let $T_e = \Omega\left(\frac{\varepsilon^2}{\ln(nr)}\right) \boxed{\frac{w(S_{i-1})-w(S_i)}{\text{rank}(S_{i-1})-\text{rank}(S_i)}}$

   randomly round to T_e

   - set $$\tilde{w}(e) = \begin{cases} \left\lceil \frac{w(e)}{T_e} \right\rceil T_e & \text{w/ prob } p_e = \frac{w_e}{T_e} - \left\lfloor \frac{w_e}{T_e} \right\rfloor \\ \left\lfloor \frac{w(e)}{T_e} \right\rfloor T_e & \text{w/ prob } 1 - p_e \end{cases}$$

Karger '93



Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger†

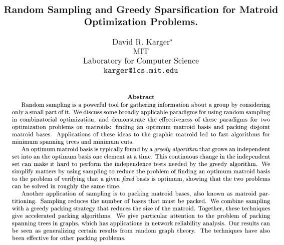
Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

$n^{O(\alpha)}$ α-APX min-cuts

uniform sample preserves min-cut (approximately)

Karger 93, 97

In unweighted matroids:



Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

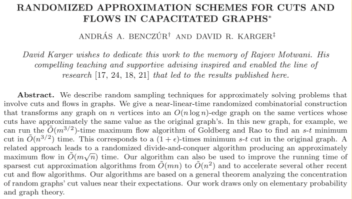
Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

$(rn)^{O(\alpha)}$ α-APX min quotients

uniform sample preserves min quotient (approximately)

(We extend to quotients of submodular f)

Benczúr, Karger '02



RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

Sample inversely proportional to "strong connectivity"

approx. decompose G along greedy tree packings á la Nagamochi-Ibaraki

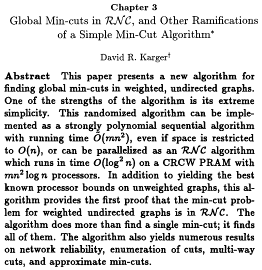
Sample inversely proportional to "matroid/submodular strength"

(network strength for graphs)

decompose $\mathcal{N}$ along O(1)-APX min ratio quotients

(APX-tree packings for graphs)

Karger '93



Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger†

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

$n^{O(\alpha)}$

preserve min-cut

min-cut (approximately)

Karger 93, 97

Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

In unweighted matroids:

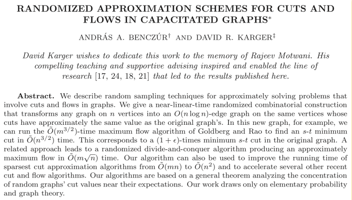
$(rn)^{O(\alpha)}$ α-APX min quotients

preserve min-quotient

min quotient (approximately)

(We extend to quotients of submodular f)

Benczúr, Karger '02



RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

ANDRÁS A. BENCZÚR† AND DAVID R. KARGER‡

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

sample inversely proportional

preserve all cuts

along greedy tree packings

á la Nagamochi-Ibaraki

Sample inversely proportional to "matroid/submodular strength"

preserve all quotients

O(1)-APX min ratio quotients

(APX-tree packings for graphs)

high-level ideas?

relation to Benczúr-Karger?

how to do it fast?

additional constraint:

be concrete

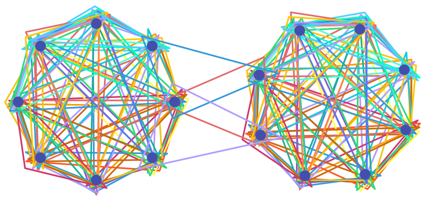
(no more "submodular f")

Focus on graph cuts, graphic matroid

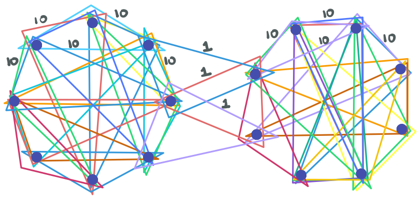
Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)\, w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)\, w(Q)$

(w/ high prob., rand. poly time, w/ oracle access to f)

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{e \in \mathcal{N} : f(e|S) = 0\}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\bar{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

$$\max \sum_t x_t \text{ over } x \in \mathbb{R}^{\mathcal{T}}_{\geq 0}$$

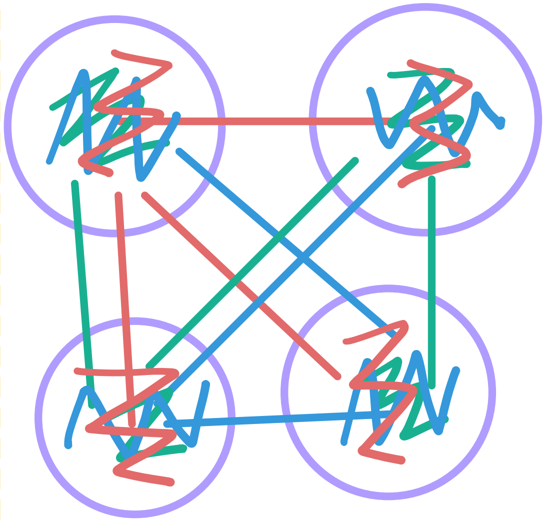
$$\sum_{t: e \in t} x_t \leq w_e \quad \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$$

(any k)

over all partitions $S_1, \ldots, S_k$ of V



weak duality: $\sum_t x_t \leq \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$

(each tree has $\geq k-1$ edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_t x_t = \min \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$$

"network strength"

Sparsification algo

1. compute $O(1)$-APX strength decomp
   $(S_0 = N) \supseteq S_1 \supseteq \cdots \supseteq S_k$

Approximate tree packings & min ratio k-cut (via the dual)

- $\tilde{O}(m)$ time

Worst case:

each k-cut removes 1 vertex

$\Rightarrow \tilde{O}(m) \times n = \tilde{O}(mn)$ time

"α-APX Strength decompositions" of $M=(N,\mathcal{I})$

$N = S_0 \supseteq S_1 \supseteq \cdots \supseteq \boxed{S_i \supseteq S_{i+1}} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$

(a) $$\boxed{\frac{w(S_i)-w(S_{i+1})}{\mathrm{rank}(S_i)-\mathrm{rank}(S_{i+1})} \le \alpha \left[\min_{T \subseteq S_i} \frac{w(S_i)-w(T)}{\mathrm{rank}(S_i)-\mathrm{rank}(T)} \right]}$$

(monotonicity)

(b) $$\boxed{\frac{w(S_i)-w(S_{i+1})}{\mathrm{rank}(S_i)-\mathrm{rank}(S_{i+1})} \le \frac{w(S_{i+1})-w(S_{i+2})}{\mathrm{rank}(S_{i+1})-\mathrm{rank}(S_{i+2})}}$$

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

$$\max \sum_{t} x_t \text{ over } x \in \mathbb{R}^{\mathcal{T}}_{\ge 0}$$

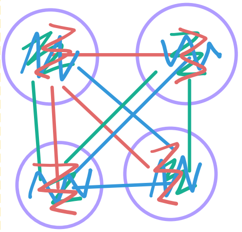
$$\sum_{t:e\in t} x_t \le w_e \ \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1,\ldots,S_k))}{k-1}$$

(any k)

over all partitions $S_1, \cdots, S_k$ of V



weak duality: $\sum_t x_t \le \frac{w(\partial(S_1,\ldots,S_k))}{k-1}$

(each tree has $\ge k-1$ edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_t x_t = \min \frac{w(\partial(S_1,\ldots,S_k))}{k-1}$$

"network strength"

Suppose $\lambda \leq$ network strength

goal: either

- remove k-cut w/ ratio $\leq 4\lambda$
- pack 2λ spanning forests

(then update $\lambda \leftarrow 2\lambda$)

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

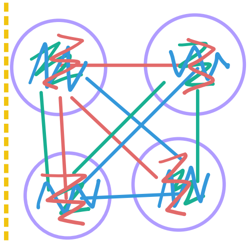
$$\max \sum_{t} x_t \text{ over } x \in \mathbb{R}^{\mathcal{T}}_{\geq 0}$$

$$\sum_{t: e \in t} x_t \leq w_e \; \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1,\dots,S_k))}{k-1}$$

over all partitions $S_1, \dots, S_k$ of V (any k)



weak duality: $\sum_{t} x_t \leq \frac{w(\partial(S_1,\dots,S_k))}{k-1}$

(each tree has ≥ k-1 edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_{t} x_t = \min \frac{w(\partial(S_1,\dots,S_k))}{k-1}$$

"network strength"

Suppose $\lambda \leq$ network strength

goal: either

- remove k-cut w/ ratio $\leq 4\lambda$
- pack 2λ spanning forests

(then update $\lambda \leftarrow 2\lambda$)

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

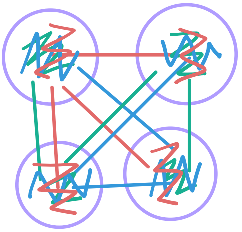
max $\sum_t x_t$ over $x \in \mathbb{R}^{\mathcal{T}}_{\geq 0}$

$$\sum_{t: e \in t} x_t \leq w_e \ \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1, \dots, S_k))}{k-1}$$

over all partitions $S_1, \dots, S_k$ of V (any k)



weak duality: $\sum_t x_t \leq \frac{w(\partial(S_1, \dots, S_k))}{k-1}$

(each tree has $\geq$ k-1 edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_t x_t = \min \frac{w(\partial(S_1, \dots, S_k))}{k-1}$$

"network strength"

1. uniform sparsification relative to λ

   $\Rightarrow$ integer capacities, strength $\tilde{\lambda} = O(\log n)$

Suppose $\lambda \leq$ network strength

goal: either

- remove k-cut w/ ratio $\leq 4\lambda$
- pack 2λ spanning forests

(then update $\lambda \leftarrow 2\lambda$)

Tree packing & covering

(connected) Graph $G=(V,E)$, $w: E \to \mathbb{R}_{>0}$, $\mathcal{T}$ = spanning trees

Max tree packing

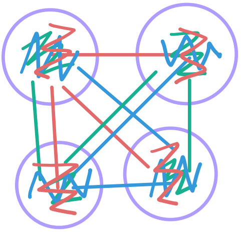
$$\max \sum_{t} x_t \text{ over } x \in \mathbb{R}^{\mathcal{T}}_{\geq 0}$$

$$\sum_{t: e \in t} x_t \leq w_e \ \forall e \in E$$

Min ratio k-cut

$$\min \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$$

over all partitions $S_1, \ldots, S_k$ of V (any k)



weak duality: $\sum_{t} x_t \leq \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$

(each tree has $\geq k-1$ edges from cut)

Tutte, Nash-Williams '61:

$$\max \sum_{t} x_t = \min \frac{w(\partial(S_1, \ldots, S_k))}{k-1}$$

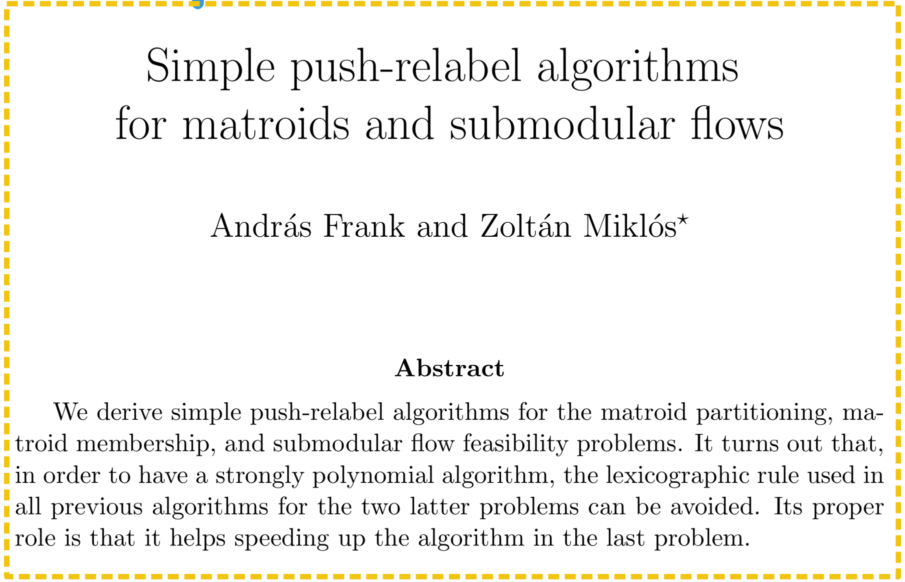
"network strength"

1. uniform sparsification relative to λ

$\Rightarrow$ integer capacities, strength $\tilde{\lambda} = O(\log n)$

2. try to pack $2\tilde{\lambda}$ trees approx. w/ push-relabel
   a. if succeed, done
   b. else find k-cut w/ ratio $4\tilde{\lambda}$ ↑

Key point: when ↑, we can keep push-relabel config of remaining edges and continue. $\tilde{O}(m)$ total.



Simple push-relabel algorithms for matroids and submodular flows

András Frank and Zoltán Miklós*

Abstract

We derive simple push-relabel algorithms for the matroid partitioning, matroid membership, and submodular flow feasibility problems. It turns out that, in order to have a strongly polynomial algorithm, the lexicographic rule used in all previous algorithms for the two latter problems can be avoided. Its proper role is that it helps speeding up the algorithm in the last problem.

Random thoughts

"spectral" version?

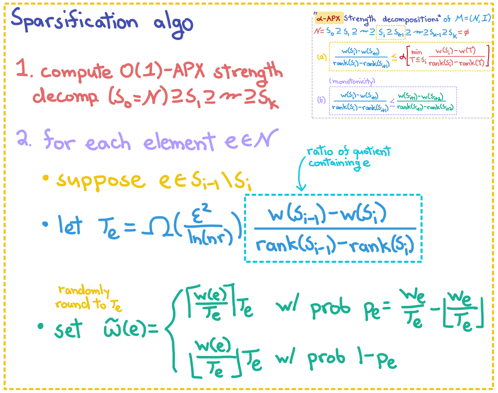
how to model k-cuts?

Submodular sparsification in the wild?

Seems pretty general...

Applications of fast strength decompositions?

Nearly linear time in graphs + hypergraphs.



Sparsification algo

"α-APX strength decompositions" of $M=(\mathcal{N},\mathcal{I})$

$\mathcal{N}=S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$

(a) $\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \alpha \left[\min_{T \subseteq S_i} \frac{w(S_i)-w(T)}{\text{rank}(S_i)-\text{rank}(T)}\right]$

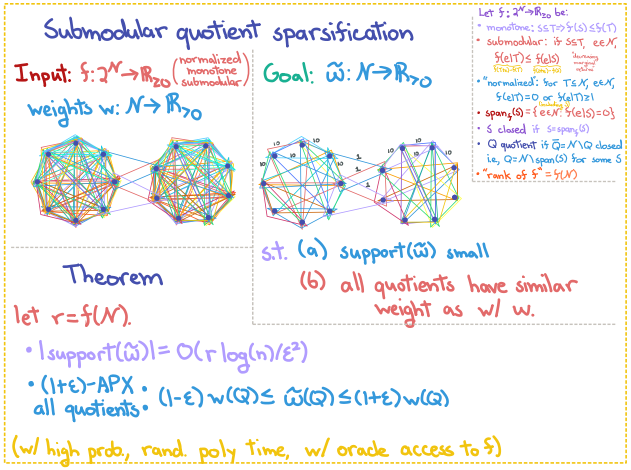
(monotonicity)

(b) $\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \leq \frac{w(S_{i+1})-w(S_{i+2})}{\text{rank}(S_{i+1})-\text{rank}(S_{i+2})}$

1. compute $O(1)$-APX strength decomp $(S_0=\mathcal{N}) \supseteq S_1 \supseteq \cdots \supseteq S_k$
2. for each element $e \in \mathcal{N}$
   - suppose $e \in S_{i-1} \setminus S_i$
   - let $\tau_e = \Omega\left(\frac{\varepsilon^2}{\ln(nr)}\right) \boxed{\frac{w(S_{i-1})-w(S_i)}{\text{rank}(S_{i-1})-\text{rank}(S_i)}}$

     ratio of quotient containing e
   - randomly round to τ_e

     set $\tilde{w}(e) = \begin{cases} \left\lceil \frac{w(e)}{\tau_e} \right\rceil \tau_e & \text{w/ prob } p_e = \frac{w_e}{\tau_e} - \left\lfloor \frac{w_e}{\tau_e} \right\rfloor \\ \left\lfloor \frac{w(e)}{\tau_e} \right\rfloor \tau_e & \text{w/ prob } 1-p_e \end{cases}$



10
10
10
10
10
1
1
1
10
10
10
10

Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$

Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$

s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

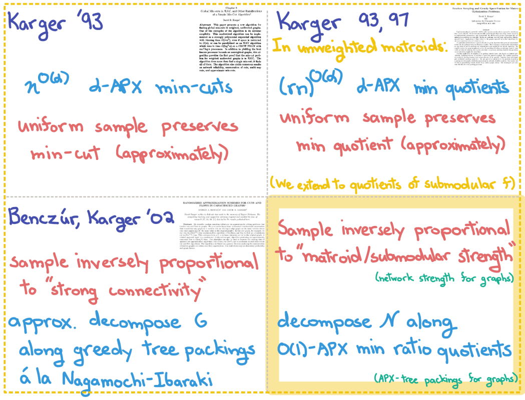
- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$, $f(e|T)=0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{e \in \mathcal{N}: f(e|S)=0\}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\bar{Q} = \mathcal{N} \setminus Q$ closed i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)w(Q)$

(w/ high prob., rand. poly time, w/ oracle access to f)



Karger '93

$n^{O(\alpha)}$ α-APX min-cuts

uniform sample preserves min-cut (approximately)

Karger 93, 97

In unweighted matroids:

$(rn)^{O(\alpha)}$ α-APX min quotients

uniform sample preserves min quotient (approximately)

(We extend to quotients of submodular f)

Benczúr, Karger '02

sample inversely proportional to "strong connectivity"

approx. decompose G along greedy tree packings á la Nagamochi-Ibaraki

Sample inversely proportional to "matroid/submodular strength"

(network strength for graphs)

decompose $\mathcal{N}$ along O(1)-APX min ratio quotients

(APX-tree packings for graphs)

Random thoughts

"spectral" version?

how to model k-cuts?

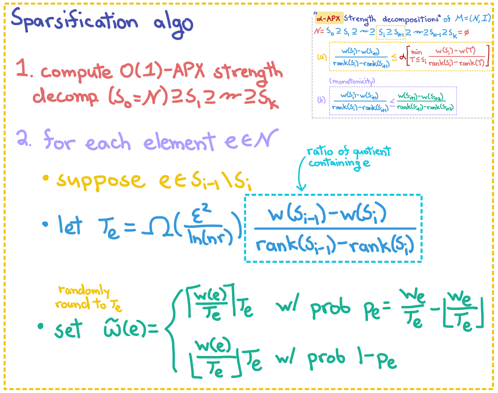
Submodular sparsification in the wild?

Seems pretty general...

Applications of fast strength decompositions?

Nearly linear time in graphs + hypergraphs.

Thanks!



Sparsification algo

"α-APX strength decompositions" of $M=(N,I)$

$N = S_0 \supseteq S_1 \supseteq \cdots \supseteq S_i \supseteq S_{i+1} \supseteq \cdots \supseteq S_{k-1} \supseteq S_k = \emptyset$

(a) $\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \le \alpha \left[\min_{T \subseteq S_i} \frac{w(S_i)-w(T)}{\text{rank}(S_i)-\text{rank}(T)} \right]$

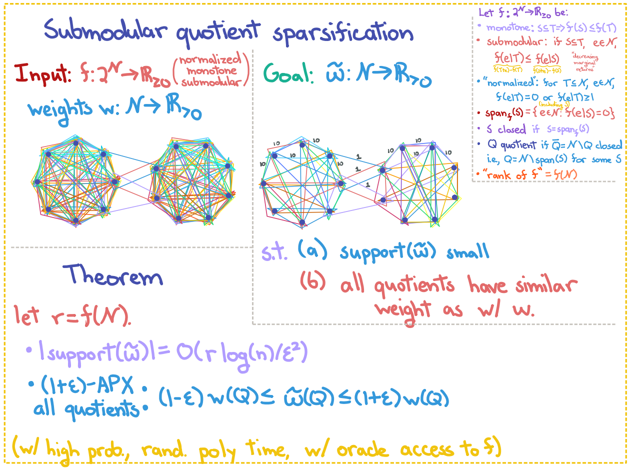
(monotonicity)

(b) $\frac{w(S_i)-w(S_{i+1})}{\text{rank}(S_i)-\text{rank}(S_{i+1})} \le \frac{w(S_{i+1})-w(S_{i+2})}{\text{rank}(S_{i+1})-\text{rank}(S_{i+2})}$

1. compute O(1)-APX strength decomp $(S_0 = N) \supseteq S_1 \supseteq \cdots \supseteq S_k$
2. for each element $e \in N$
   - suppose $e \in S_{i-1} \setminus S_i$
   - let $\tau_e = \Omega\left(\frac{\varepsilon^2}{\ln(nr)}\right) \boxed{\frac{w(S_{i-1})-w(S_i)}{\text{rank}(S_{i-1})-\text{rank}(S_i)}}$

     ratio of quotient containing e
   - set $\tilde{\omega}(e) = \begin{cases} \left\lceil \frac{w(e)}{\tau_e} \right\rceil \tau_e & \text{w/ prob } p_e = \frac{w_e}{\tau_e} - \left\lfloor \frac{w_e}{\tau_e} \right\rfloor \\ \left\lfloor \frac{w(e)}{\tau_e} \right\rfloor \tau_e & \text{w/ prob } 1-p_e \end{cases}$

     randomly round to τ_e



Submodular quotient sparsification

Input: $f: 2^N \to \mathbb{R}_{\ge 0}$ (normalized monotone submodular)

weights $w: N \to \mathbb{R}_{>0}$

Goal: $\tilde{\omega}: N \to \mathbb{R}_{>0}$

s.t. (a) support($\tilde{\omega}$) small

(b) all quotients have similar weight as w/ w.

10
10
10
10
10
10
10
10
10
1
1
1

Let $f: 2^N \to \mathbb{R}_{\ge 0}$ be:

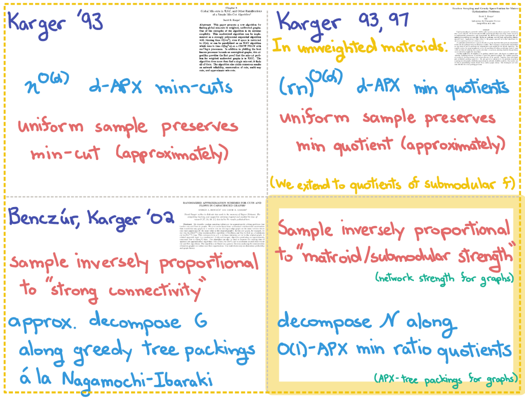
- monotone: $S \subseteq T \Rightarrow f(S) \le f(T)$
- submodular: if $S \subseteq T$, $e \in N$, $\underbrace{f(e|T)}_{f(T+e)-f(T)} \le \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq N$, $e \in N$, $f(e|T)=0$ or $f(e|T) \ge 1$
- $\text{span}_f(S) = \{ e \in N : f(e|S) = 0 \}$ (including S)
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\bar{Q} = N \setminus Q$ closed i.e., $Q = N \setminus \text{span}(S)$ for some S
- "rank of f" $= f(N)$

Theorem

let $r = f(N)$.

- $|\text{support}(\tilde{\omega})| = O(r \log(n)/\varepsilon^2)$
- (1+ε)-APX all quotients: $(1-\varepsilon)w(Q) \le \tilde{\omega}(Q) \le (1+\varepsilon)w(Q)$

(w/ high prob., rand. poly time, w/ oracle access to f)



Karger '93

Chapter 3
Global Min-cuts in RNC, and Other Ramifications of a Simple Min-Cut Algorithm
David R. Karger

$n^{O(\alpha)}$ α-APX min-cuts

uniform sample preserves min-cut (approximately)

Karger 93, 97

Random Sampling and Greedy Sparsification for Matroid Optimization Problems
David R. Karger

In unweighted matroids:

$(rn)^{O(\alpha)}$ α-APX min quotients

uniform sample preserves min quotient (approximately)

(We extend to quotients of submodular f)

Benczúr, Karger '02

RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS
ANDRÁS A. BENCZÚR AND DAVID R. KARGER

sample inversely proportional to "strong connectivity"

approx. decompose G along greedy tree packings á la Nagamochi-Ibaraki

Sample inversely proportional to "matroid/submodular strength"

(network strength for graphs)

decompose N along O(1)-APX min ratio quotients

(APX-tree packings for graphs)

Chapter 3

Global Min-cuts in $\mathcal{RNC}$, and Other Ramifications of a Simple Min-Cut Algorithm*

David R. Karger[†]

Abstract This paper presents a new algorithm for finding global min-cuts in weighted, undirected graphs. One of the strengths of the algorithm is its extreme simplicity. This randomized algorithm can be implemented as a strongly polynomial sequential algorithm with running time $\tilde{O}(mn^2)$, even if space is restricted to $O(n)$, or can be parallelized as an $\mathcal{RNC}$ algorithm which runs in time $O(\log^2 n)$ on a CRCW PRAM with $mn^2 \log n$ processors. In addition to yielding the best known processor bounds on unweighted graphs, this algorithm provides the first proof that the min-cut problem for weighted undirected graphs is in $\mathcal{RNC}$. The algorithm does more than find a single min-cut; it finds all of them. The algorithm also yields numerous results on network reliability, enumeration of cuts, multi-way cuts, and approximate min-cuts.

1 Introduction

This paper studies the min-cut problem. Given a graph with n vertices and m (possibly weighted) edges, we wish to partition the vertices into two non-empty sets S and T so as to minimize the number of edges crossing from S to T (if the graph is weighted, we wish to minimize the total weight of crossing edges). Throughout this paper, the graph is assumed to be connected, since otherwise the problem is trivial. The problem actually comes in two flavors: in the *s-t min-cut* problem, we require that the two specific vertices s and t be on opposite sides of the cut; in what will be called the *min-cut* problem, or for emphasis the *global min-cut* problem, there is no such restriction.

1.1 Previous Work.

The oldest known way to compute min-cuts is to use their well known duality with max-flows [FF56, FF62]. Computation of an s-t max-flow allows the immediate determination of an s-t min-cut. The best presently known sequential time bound for max-flow is $O(mn\log(n^2/m))$, found by Goldberg and Tarjan [GT88]. Global min-cuts can be computed by minimizing over s-t max-flows; Hao and Orlin [HO92] show how the max-flow computations can be pipelined so that together they take no more time than a single max-flow computation; thus the global min-cut problem can be solved in the same $\tilde{O}(mn)$ running time.[1]

Recently, progress has been made in special cases of the min-cut problem. On unweighted graphs, the min-cut problem is often known as the edge-connectivity problem. Gabow [Gab91] shows how to find the edge-connectivity c of a graph in time $O(cn\log(n^2/m))$. On weighted, undirected graphs, the algorithm of Nagamochi and Ibaraki [NI92] computes the min-cut in time $O(mn+n^2\log n)$. These algorithms make no use of max-flow computations.

Work has also been done on parallel solutions to the min-cut problem. Goldschlager, Shaw, and Staples [GSS82] showed that the s-t min-cut problem on weighted directed graphs is P-complete. This is also true for the global min-cut problem (see section 4.2). In the special case of unweighted directed or undirected graphs, the matching algorithm of Karp, Upfal and Wigderson [KUW86], together with a reduction described by Mulmeley, Vazirani and Vazirani [MVV87], can be used to find s-t max-flows and min-cuts in $O(\log^2 n)$ time using $mn^{3.5}$ processors. An alternative approach of Galil and Pan [GP88] uses $n^2M(n)$ processors, where $M(n)$ is the processor cost for multiplying two matrices (presently about $n^{2.37}$). In undirected graphs, fixing a vertex s and finding s-t min-cuts for all vertices t identifies a min-cut; this requires performing n min-cut computations in parallel at a total cost of $mn^{4.5}$ or $n^2M(n)$ processors. Either algorithm can be extended to weighted graphs by treating an edge of weight w as a collection of w unweighted edges. How-

*Supported by a National Science Foundation Graduate Fellowship

[†]Department of Computer Science, Stanford University. karger@cs.stanford.edu

[1] The notation $O(f)$ denotes $O(f \text{ polylog } f)$

Using Randomized Sparsification to Approximate Minimum Cuts

David R. Karger*
Department of Computer Science
Stanford University
karger@cs.stanford.edu

October 29, 1993

Abstract

We introduce the concept of randomized sparsification of a weighted, undirected graph. Randomized sparsification yields a sparse unweighted graph which closely approximates the minimum cut structure of the original graph. As a consequence, we show that a cut of weight within a $(1+\epsilon)$ multiplicative factor of the minimum cut in a graph can be found in $O(m + n(\log^3 n)/\epsilon^4)$ time; thus any constant factor approximation can be achieved in $\tilde{O}(m)$ time. Similarly, we show that a cut within a multiplicative factor of α of the minimum can be found in $\mathcal{RNC}$ using $m + n^{2/\alpha}$ processors. We also investigate a parametric version of our randomized sparsification approach. Using it, we show that for a graph undergoing a series of edge insertions and deletions, an $O(\sqrt{1+2/\epsilon})$-approximation to the minimum cut value can be maintained at a cost of $\tilde{O}(n^{\epsilon+1/2})$ time per insertion or deletion. If only insertions are allowed, the approximation can be maintained at a cost of $\tilde{O}(n^{\epsilon})$ time per insertion.

1 Introduction

1.1 Minimum Cuts.

This paper studies the min-cut problem. Given a graph with n vertices and m (possibly weighted) edges, we wish to partition the vertices into two non-empty sets so as to minimize the number or total weight of edges crossing between them. Throughout this paper, the graph is assumed to be connected because otherwise the problem is trivial. We also require that all edge weights be non-negative, because otherwise the problem is $\mathcal{NP}$-complete by a trivial transformation from the maximum-cut problem [GJ79, page 210]. The problem actually has two variants: in the *s-t min-cut problem* we require that two specified vertices s and t be on opposite sides of the cut; in what we call the *min-cut problem* there is no such restriction.

Particularly on unweighted graphs, solving the min-cut problem is sometimes referred to as finding the *connectivity* of a graph; that is, determining the minimum number of edges (or minimum total edge weight) that must be removed to disconnect the graph.

Throughout this paper, we will focus attention on an n vertex, m edge graph with minimum cut value c. The fastest presently known algorithm for finding minimum cuts in weighted undirected graphs is the Recursive Contraction Algorithm (RCA) of Karger and Stein [KS93]; it runs in $O(n^2 \log^3 n)$ time. An algorithm by Gabow [Gab91] finds the minimum cut in an unweighted graph in time $O(m + c^2 n \log(n/c))$, where c is the value of the minimum cut. It is thus faster than the RCA on unweighted graphs with small minimum cuts ($c < \sqrt{n}$).

1.2 New Results.

This paper studies algorithms for *approximating* the minimum cut. To this end, we make the following definition:

DEFINITION 1.1. *An* α-approximation to the minimum cut, *or more concisely an* α-minimal cut, *is a cut whose weight is within a multiplicative factor of* α *of the minimum cut. An* α-approximation algorithm *is one which finds an* α*-minimal cut in every input graph.*

In this paper, we give a collection of minimum cut approximation algorithms. They are based on a randomized algorithm for taking a weighted undirected graph and constructing a sparse unweighted graph, or *skeleton*, which closely approximates the minimum cut information of the original graph. Finding a minimum cut in the skeleton gives information about the minimum cut in the original graph. Because the skeleton is sparse and unweighted, fast specialized minimum cut algorithms (such as Gabow's) can be applied.

We use graph skeletons in a new sequential approximation algorithm for minimum cuts. For any $\epsilon > 0$, the algorithm finds a $(1+\epsilon)$-minimal cut in $O(m + n(\log^3 n)/\epsilon^4)$; thus, in particular, if ϵ is any constant it finds a $(1+\epsilon)$-approximation in $O(m + n\log^3 n)$

*Supported by a National Science Foundation Graduate Fellowship, by NSF Grant CCR-9010517, and grants from Mitsubishi and OTL.

SIAM J. COMPUT.
Vol. 44, No. 2, pp. 290–319

RANDOMIZED APPROXIMATION SCHEMES FOR CUTS AND FLOWS IN CAPACITATED GRAPHS*

ANDRÁS A. BENCZÚR[†] AND DAVID R. KARGER[‡]

David Karger wishes to dedicate this work to the memory of Rajeev Motwani. His compelling teaching and supportive advising inspired and enabled the line of research [17, 24, 18, 21] *that led to the results published here.*

Abstract. We describe random sampling techniques for approximately solving problems that involve cuts and flows in graphs. We give a near-linear-time randomized combinatorial construction that transforms any graph on n vertices into an $O(n \log n)$-edge graph on the same vertices whose cuts have approximately the same value as the original graph's. In this new graph, for example, we can run the $\tilde{O}(m^{3/2})$-time maximum flow algorithm of Goldberg and Rao to find an s-t minimum cut in $\tilde{O}(n^{3/2})$ time. This corresponds to a $(1+\epsilon)$-times minimum s-t cut in the original graph. A related approach leads to a randomized divide-and-conquer algorithm producing an approximately maximum flow in $\tilde{O}(m\sqrt{n})$ time. Our algorithm can also be used to improve the running time of sparsest cut approximation algorithms from $\tilde{O}(mn)$ to $\tilde{O}(n^2)$ and to accelerate several other recent cut and flow algorithms. Our algorithms are based on a general theorem analyzing the concentration of random graphs' cut values near their expectations. Our work draws only on elementary probability and graph theory.

Key words. minimum cut, maximum flow random graph, random sampling, connectivity, cut enumeration, network reliability

AMS subject classifications. 05C21, 05C40, 05C80, 68W25, 68W40, 68Q25, 05C85

DOI. 10.1137/070705970

1. Introduction.

This paper gives results on random sampling methods for reducing the number of edges in any undirected graph while approximately preserving the values of its cuts and consequently its flows. It then demonstrates how these techniques can be used in faster algorithms to approximate the values of minimum cuts and maximum flows in such graphs. We give an $\tilde{O}(m)$-time[1] *compression* algorithm to reduce the number of edges in any n-vertex graph to $O(n \log n)$ with only a small perturbation in cut values and then use that compression method to find approximate minimum cuts in $\tilde{O}(n^2)$ time and approximate maximum flows in $\tilde{O}(m\sqrt{n})$ time.

1.1. Background.

Previous work [19, 18, 22] has shown that random sampling is an effective tool for problems involving cuts in graphs. A *cut* is a partition of a graph's vertices into two groups; its *value* is the number, or in weighted graphs the total weight, of edges with one endpoint on each side of the cut. Many problems depend only on cut values. The maximum flow that can be routed from s to t is the minimum value of any cut separating s and t [10]. A minimum bisection is the smallest cut that splits the graph into two equal-sized pieces. The *connectivity* or *minimum*

*Received by the editors October 21, 2007; accepted for publication (in revised form) December 13, 2013; published electronically March 31, 2015.
http://www.siam.org/journals/sicomp/44-2/70597.html

[†] Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, Hungary (benczur@sztaki.hu, http://www.sztaki.hu/∼benczur).

[‡] M.I.T. Computer Science and Artificial Intelligence Laboratory, Cambridge MA 02139 (karger@mit.edu, http://people.csail.mit.edu/∼karger). The work of this author was supported in part by a grant from the National Science Foundation.

[1] The notation $\tilde{O}(f)$ denotes $O(f \text{ polylog } n)$, where n is the input problem size.

Random Sampling and Greedy Sparsification for Matroid Optimization Problems.

David R. Karger*
MIT
Laboratory for Computer Science
karger@lcs.mit.edu

Abstract

Random sampling is a powerful tool for gathering information about a group by considering only a small part of it. We discuss some broadly applicable paradigms for using random sampling in combinatorial optimization, and demonstrate the effectiveness of these paradigms for two optimization problems on matroids: finding an optimum matroid basis and packing disjoint matroid bases. Applications of these ideas to the graphic matroid led to fast algorithms for minimum spanning trees and minimum cuts.

An optimum matroid basis is typically found by a *greedy algorithm* that grows an independent set into an the optimum basis one element at a time. This continuous change in the independent set can make it hard to perform the independence tests needed by the greedy algorithm. We simplify matters by using sampling to reduce the problem of finding an optimum matroid basis to the problem of verifying that a given *fixed* basis is optimum, showing that the two problems can be solved in roughly the same time.

Another application of sampling is to packing matroid bases, also known as matroid partitioning. Sampling reduces the number of bases that must be packed. We combine sampling with a greedy packing strategy that reduces the size of the matroid. Together, these techniques give accelerated packing algorithms. We give particular attention to the problem of packing spanning trees in graphs, which has applications in network reliability analysis. Our results can be seen as generalizing certain results from random graph theory. The techniques have also been effective for other packing problems.

1 Introduction

Arguably the central concept of statistics is that of a representative sample. It is often possible to gather a great deal of information about a large population by examining a small sample randomly drawn from it. This has obvious advantages in reducing the investigator's work, both in gathering and in analyzing the data.

We apply the concept of a representative sample to combinatorial optimization. Given an optimization problem, it may be possible to generate a small representative subproblem by random sampling. Intuitively, such a subproblem may form a microcosm of the larger problem. In particular, an optimum solution to the subproblem may be a nearly optimum solution to the problem as a whole. In some situations, such an approximation might be sufficient. In other situations, it may be relatively easy to improve this good solution to a truly optimum solution.

*Some of this work done at Stanford University, supported by National Science Foundation and Hertz Foundation Graduate Fellowships, and NSF Young Investigator Award CCR-9357849, with matching funds from IBM, Schlumberger Foundation, Shell Foundation and Xerox Corporation. Research supported in part by ARPA contract N00014-95-1-1246 and NSF contract CCR-9624239

Simple push-relabel algorithms for matroids and submodular flows

András Frank and Zoltán Miklós*

Abstract

We derive simple push-relabel algorithms for the matroid partitioning, matroid membership, and submodular flow feasibility problems. It turns out that, in order to have a strongly polynomial algorithm, the lexicographic rule used in all previous algorithms for the two latter problems can be avoided. Its proper role is that it helps speeding up the algorithm in the last problem.

1 Introduction

Push-relabel algorithms (see, for example, the first one of Goldberg and Tarjan, [16]), unlike augmenting path type algorithms, use only small, local steps. In order to make progess, in selecting the current element where the next local step is to be performed, they use a control parameter $\Theta : S \to \{0, 1, 2, \ldots\}$ called a **level** (or distance) **function**. Here S can be the node-set of a directed graph or the ground-set of a matroid. In the present work the range of the level functions is $\{0, 1, 2, \ldots, n\}$ where $n = |S|$ while the original algorithm of Goldberg and Tarjan for maximum flows must have allowed $\{0, 1, 2, \ldots, 2n - 1\}$ for the range of Θ.

The goal of the present paper is to develop simple push-relabel algorithms in submodular optimization. We exhibit versions for matroid partition, for membership in a matroid polytope, and for submodular flow feasibility. All the previous algorithms relied on a selection rule based on a consistent ordering of the elements. This rule can be considered as a counterpart of the lexicographic rule of Schönsleben [19] applied to augmenting path type algorithms. The new push-relabel algorithms do not use the consistency rule and the proof of strong polynomiality becomes much simpler. The true role of the consistency rule is that, though not needed for strong polynomiality, it improves the complexity of the algorithm by one order of magnitude.

For a given level function Θ, the sets $L_i = \{v : \Theta(v) = i\}$ $(i = 0, \ldots, n)$ are called the **level sets** of Θ. For an element s with $\Theta(s) = j$, we say that the level of s is j or that s is in L_j. For a subset $X \subseteq S$, let $\Theta_{\min}(X) := \min\{\Theta(v) : v \in X\}$. One of the local steps during the algorithm is **lifting** an element s of S with $\Theta(s) \leq n - 1$

*MTA-ELTE Egerváry Research Group, Department of Operations Research, Eötvös University, Pázmány P. s. 1/c, Budapest, Hungary, H-1117. e-mail: `frank@cs.elte.hu` and `miklosz@cs.elte.hu`