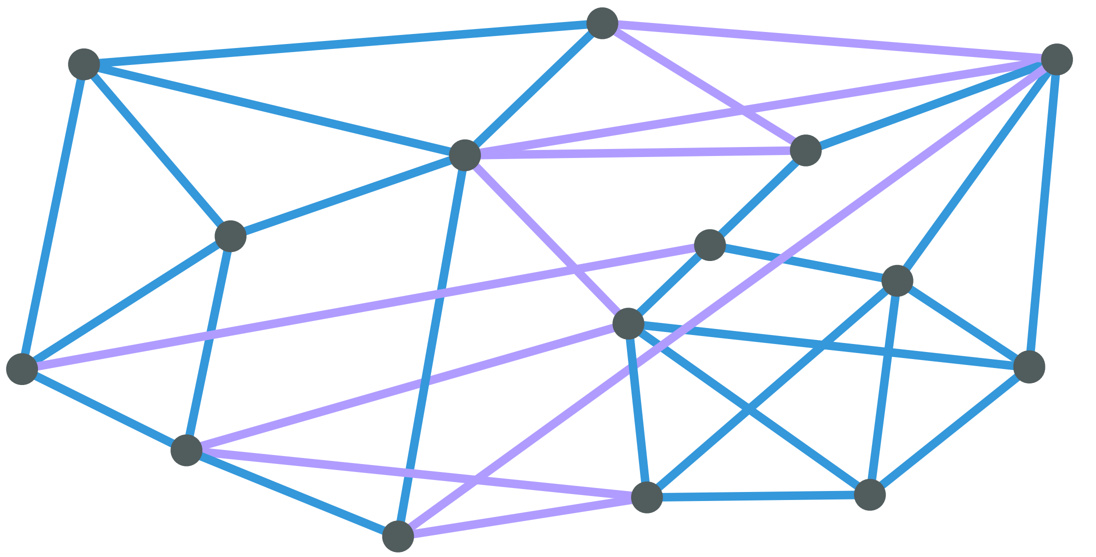
FAST and Deterministic Approximations for K-cut

Kent Quanrud, Purdue

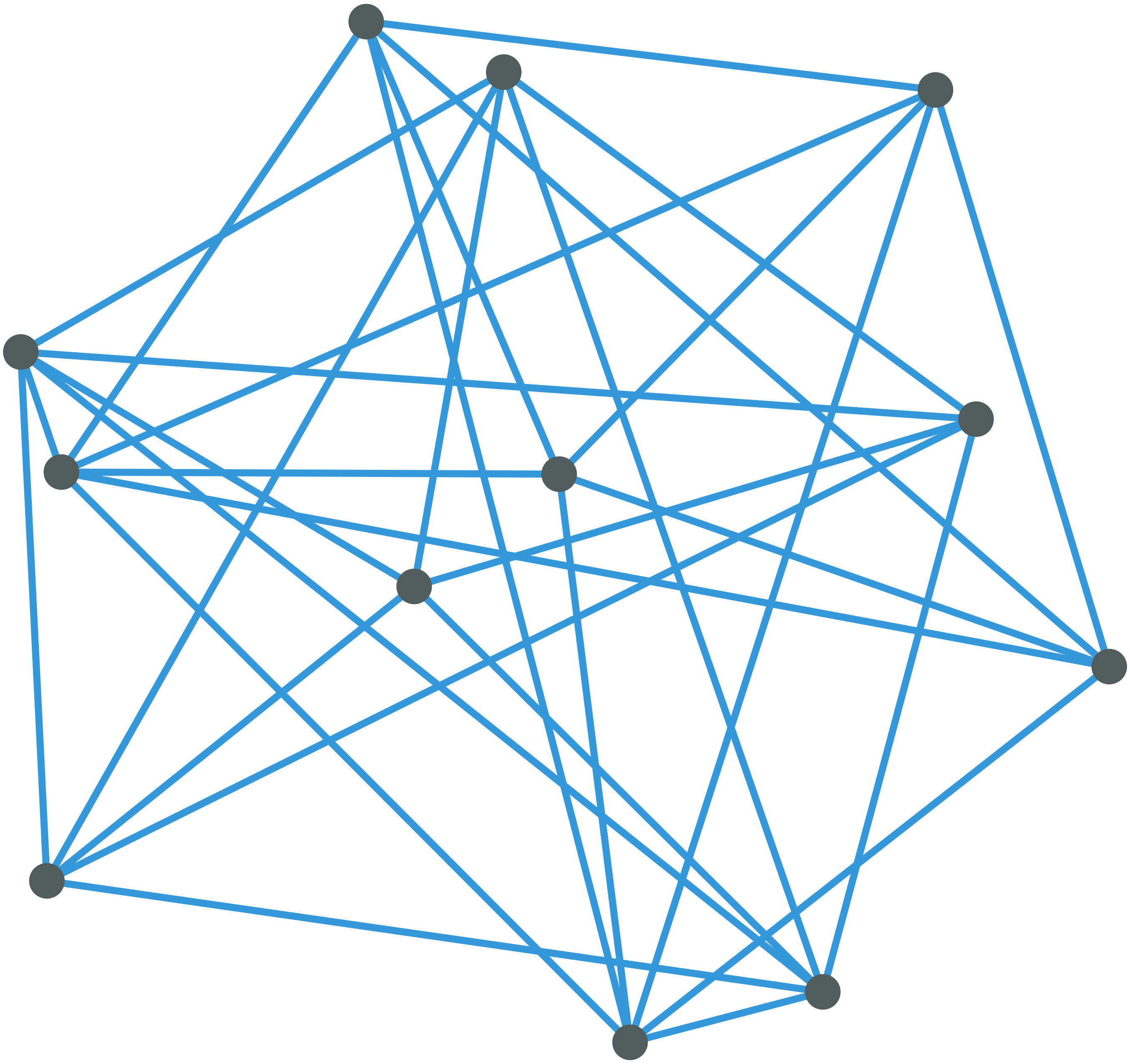
Minimum cut

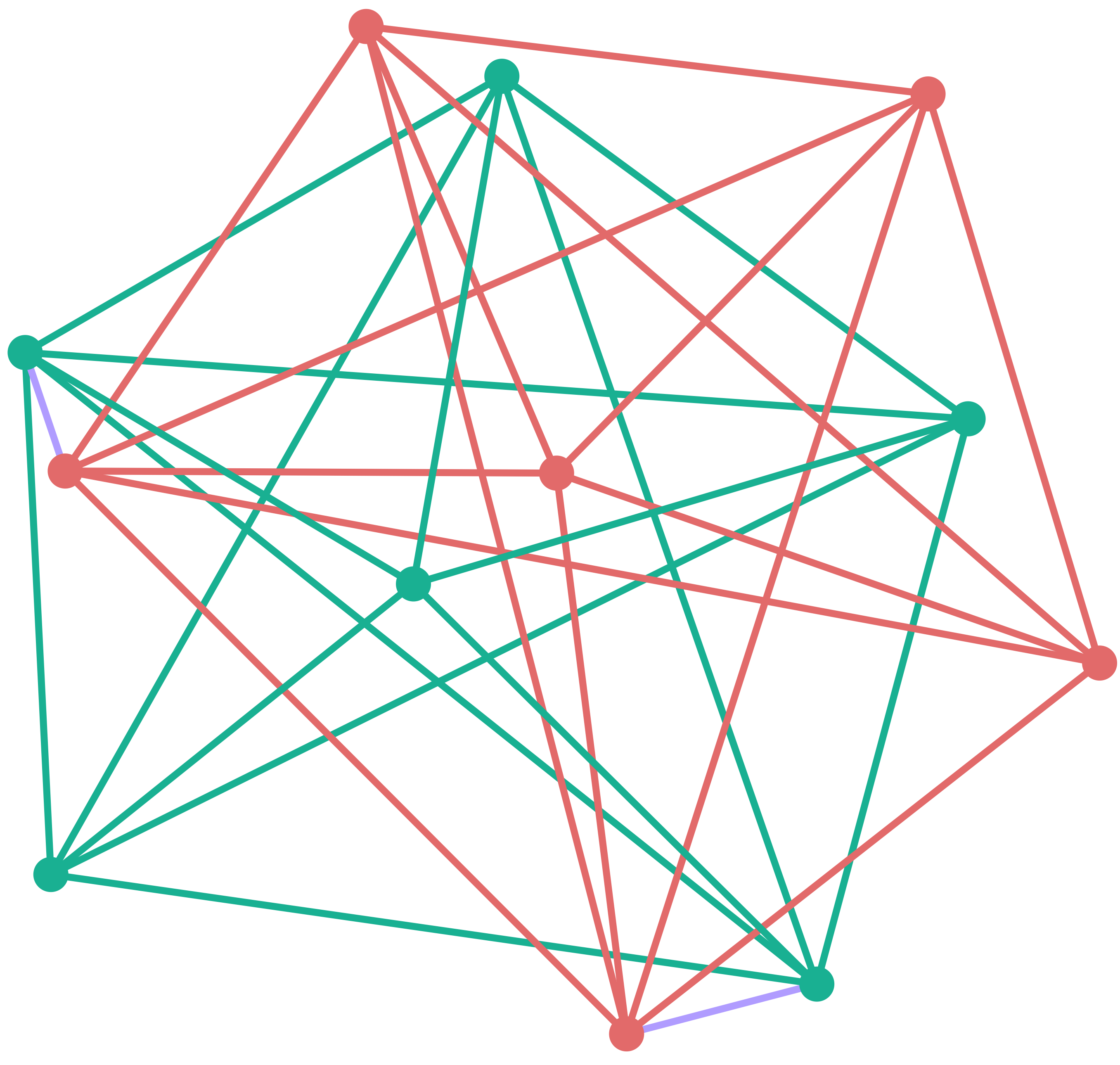
Input: weighted undirected graph



Goal: Find minimum weight <u>cut</u>

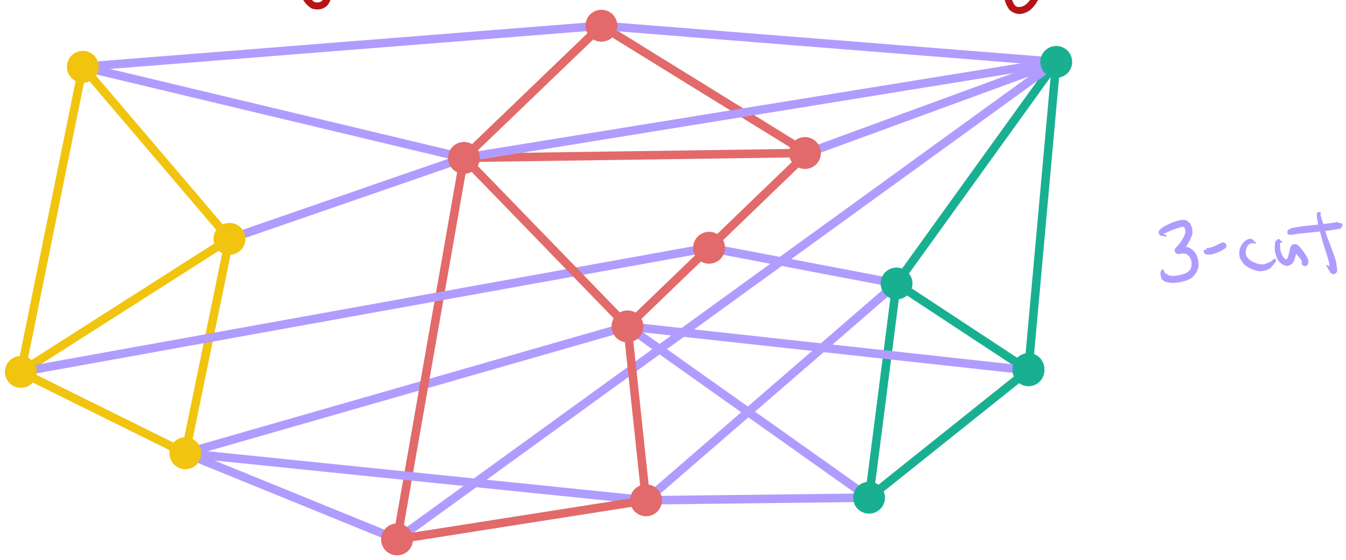
where a <u>cut</u> is a set of edges whose removal disconnects the graph





Minimum k-Cut (k=2,3,4...)

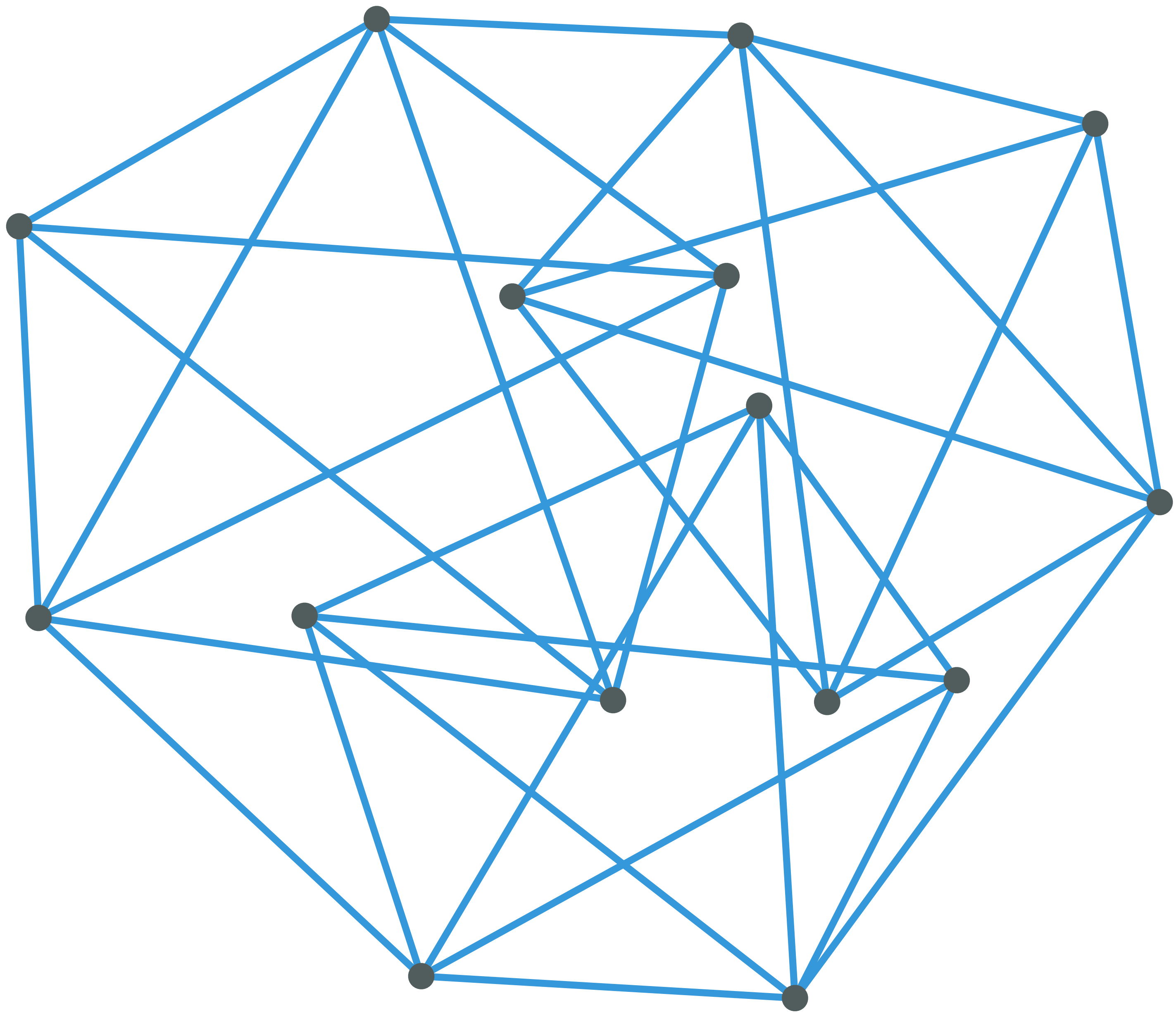
Input: weighted undirected graph



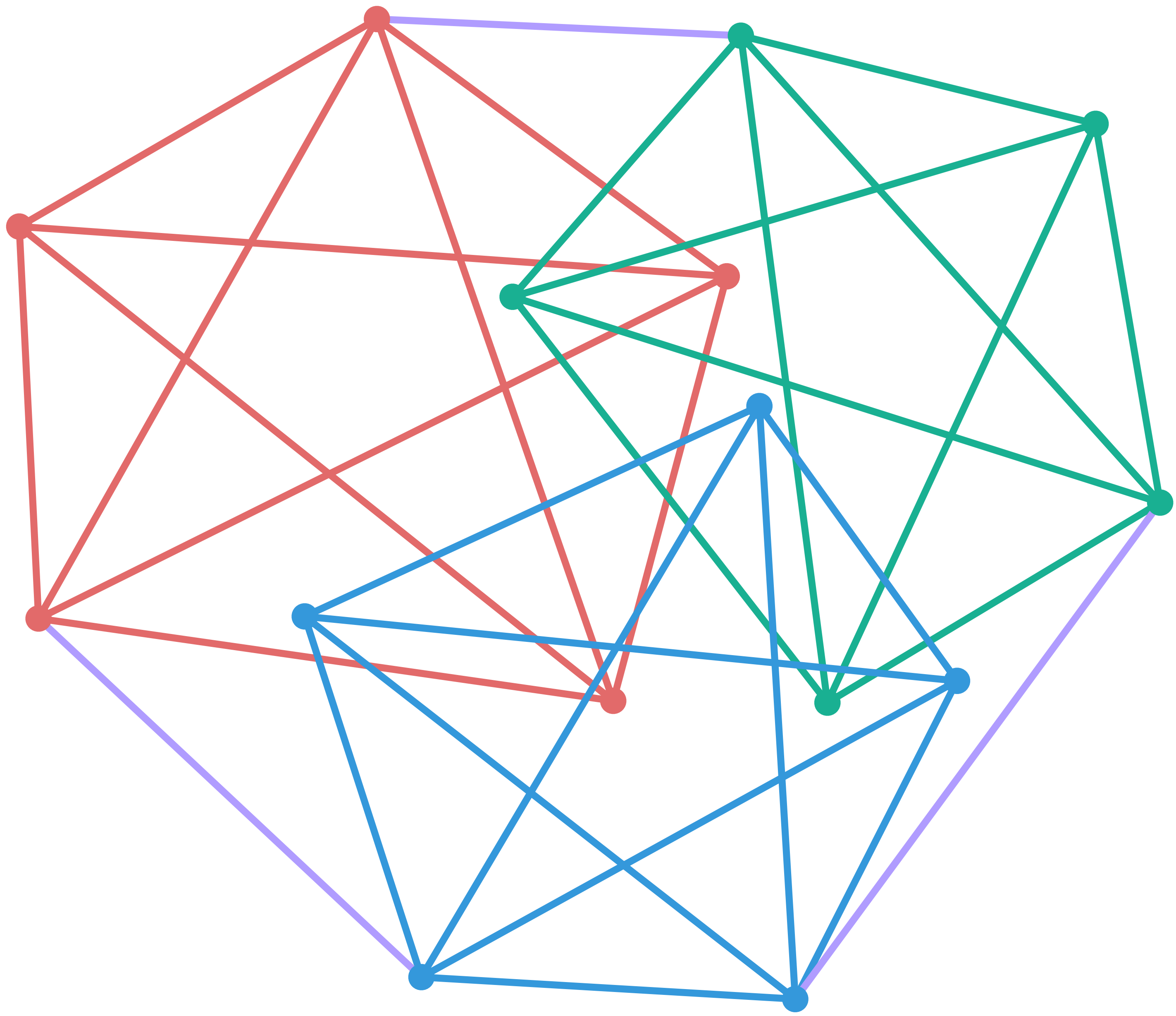
Goal: Find minimum weight k-cut

where a k-cut is a set of edges whose removal disconnects the graph into (at least) k connected components

Minimum 3-cut?



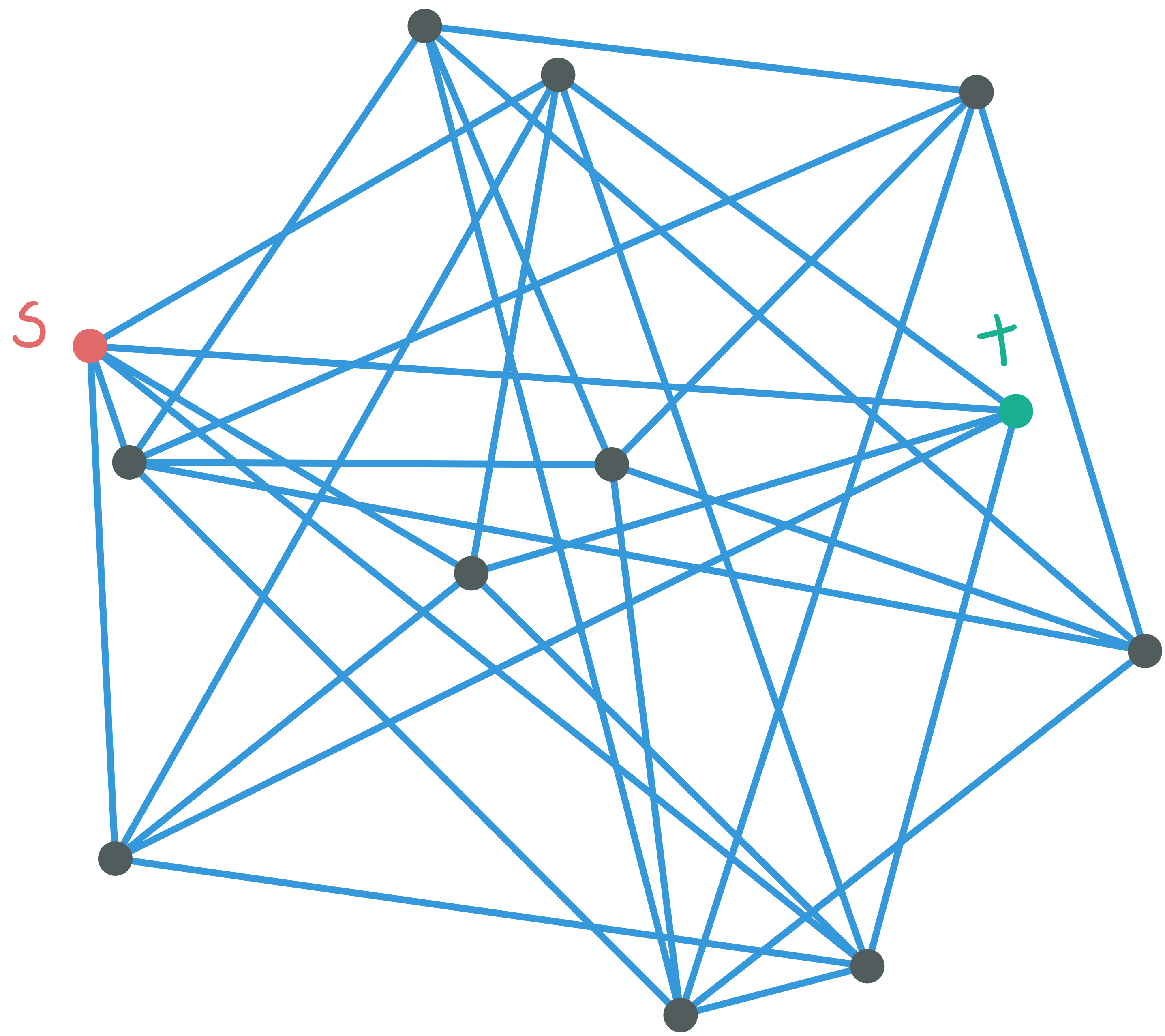
Minimum 3-cut?

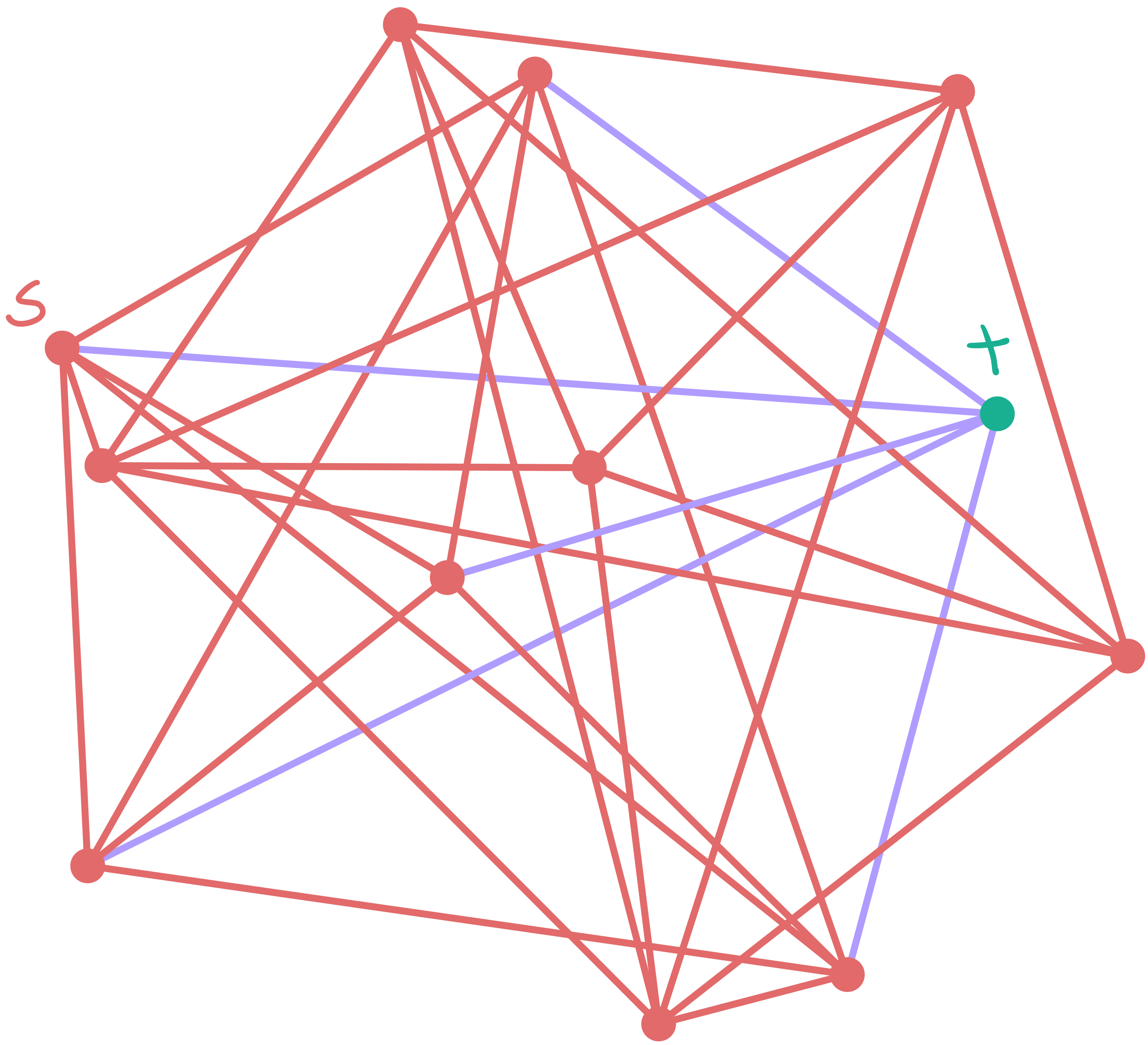


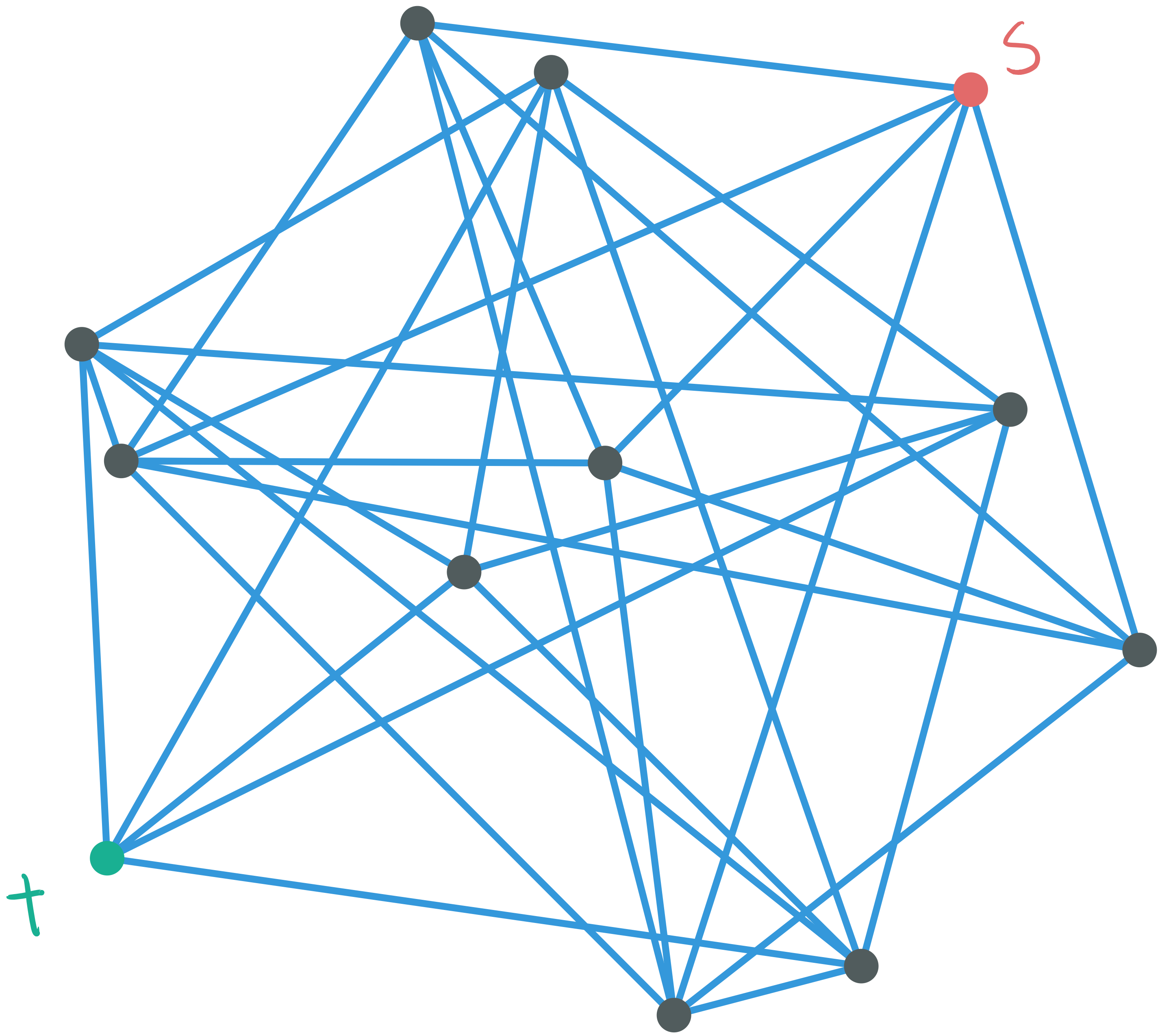
Min-cut

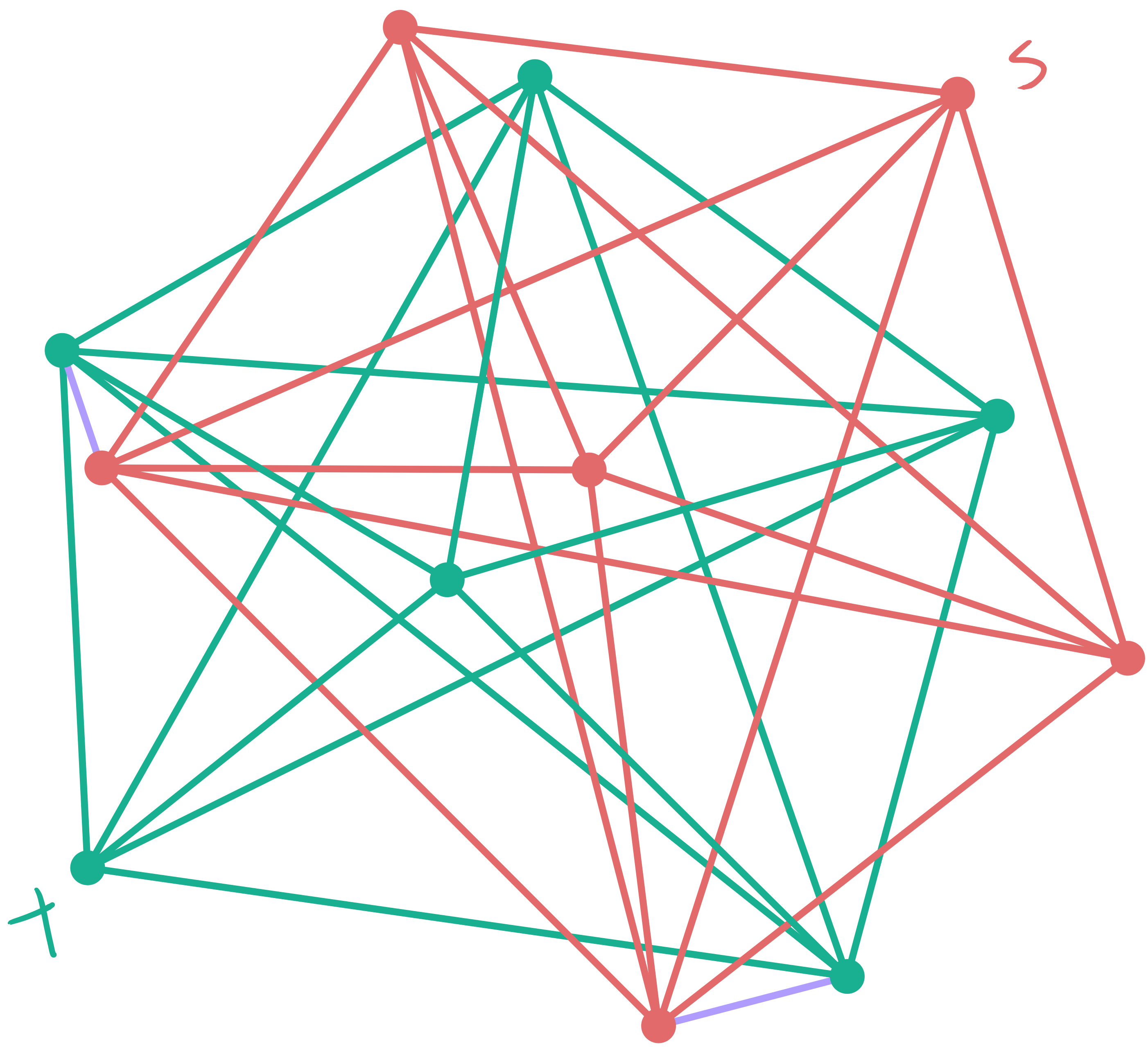
Min-cut is poly-time solvable

min of min s-t cut over all s,t




s
t


s
t


s
t

Min-cut is poly-time solvable

min of min s-t cut over all ~~s~~,t

Running time: $n^{\cancel{2}} \cdot MF(m,n)$

$MF(m,n)$ = time for max flow on m edges, n vertices

$MF(m,n)$	Deterministic	Strong-Poly	
$O(mn)$	✓	✓	[Orlin]
$\tilde{O}(mn^{2/3})$	✓		[Goldberg Rao]
$\hat{O}(m\sqrt{n})$			[Lee Sidford]
$\tilde{O}(m^{10/7})$			[Madry]

Better than $n \cdot$ max-flow?

Nagamochi and Ibaraki ← $\tilde{O}(mn)$ → Hao and Orlin

☆ Karger (random-contract) $\tilde{O}(mn^2)$, parallel

☆ Karger and Stein $\tilde{O}(n^2)$

☆ Karger (tree packing) $\tilde{O}(m)$

☆ randomized

2^n-dollar question:

$\tilde{O}(m)$ deterministic min-cut??

Recent breakthrough:

$\tilde{O}(m)$ deterministic for unweighted

Kawarabayashi, Thorup

k-cut, $k \geq 3$

max-flow is not enough

hard?

Goldschmidt and Hochbaum, 1994

k-cut is NP-Hard when
k is part of the input

i.e. NP-Hard for poly(m, n, k) time

$n^{O(k)}$ exact algo possible

best exponential dependence
k is an active research problem

Approximating k-cut

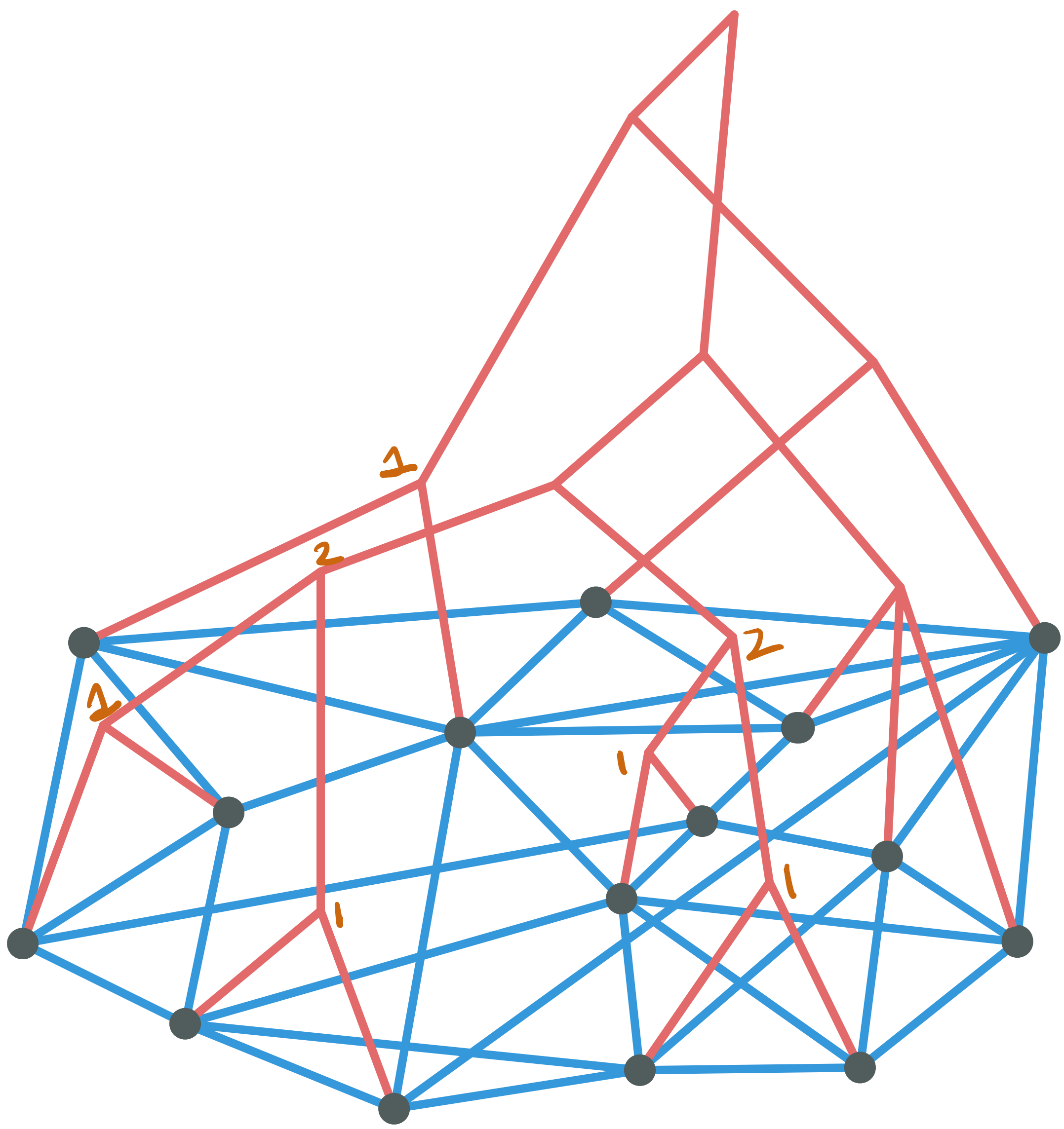
Saran & Vazirani 1995

2-APX in poly(m,n,k)

Two different algos

Algo 1 Build Gomory-Hu tree, take k lightest cuts

(Gomory-Hu tree encodes all s-t min-cut values in an auxiliary tree)

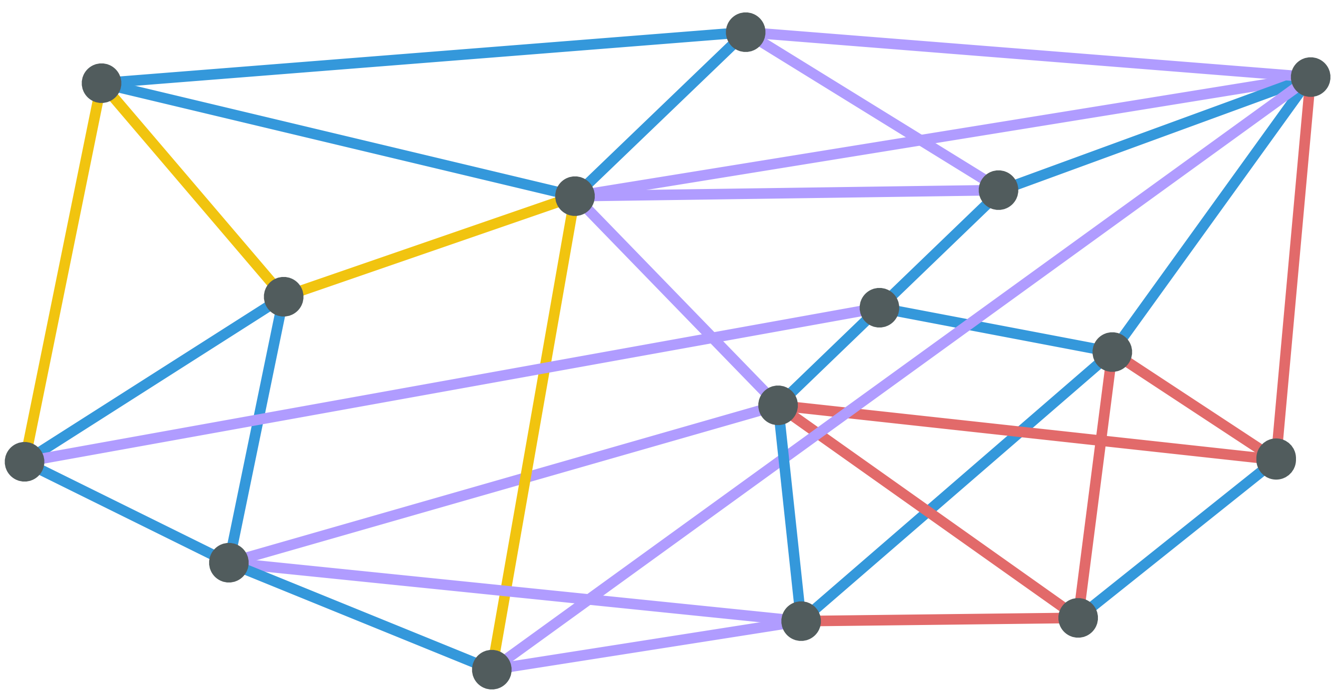


Algo 2 Recursive min-cuts

1. while < k connected components

   a. find min-cut in each component

   b. add smallest cut



Running times

Gomory-Hu tree takes $n \cdot MF(m,n)$ time

Recursive min-cuts takes $k \cdot$(min-cut)

$\tilde{O}(mk)$ randomized time

$\tilde{O}(mnk)$ deterministic

Better deterministic algo

$\tilde{O}(mn)$ Nagamochi & Kamidoi 2007

Is 2 the right APX?

Probably. (Manurangsi 2017)

ETH-Hard to do better than 2

Min-cut vs 2-APX k-cut

	min-cut	2-APX k-cut
Deterministic	$\tilde{O}(mn)$	$\hat{O}(mn)$
Randomized	$\tilde{O}(m)$	$\tilde{O}(mk)$

$\boxed{\text{Q1}}$ deterministic 2-APX k-cut as fast as randomized?

$\boxed{\text{Q2}}$ 2-APX k-cut as fast as min-cut? $\tilde{O}(m)$ randomized?

Our Results

$(2+\varepsilon)$-APX min k-cut in

$O(m \log^3(n)/\varepsilon^2)$ deterministic time

$\forall \varepsilon > 0$

Min-cut vs 2-APX k-cut

	min-cut	2-APX k-cut	$(2+\varepsilon)$-APX k-cut
Deterministic	$\tilde{O}(mn)$	$\hat{O}(mn)$	$O\left(\frac{m \log^3(n)}{\varepsilon^2}\right)$
Randomized	$\tilde{O}(m)$	$\tilde{O}(mk)$	

$\boxed{Q1}$ deterministic 2-APX k-cut as fast as randomized?

really close

$\boxed{Q2}$ 2-APX k-cut as fast as min-cut? $\tilde{O}(m)$ randomized?

Breaking down "$(2+\varepsilon)$-APX in $\tilde{O}(m/\varepsilon^2)$"

① $(1+\varepsilon)$-APX to the LP in $\tilde{O}(m/\varepsilon^2)$ time

② Round the LP (losing factor 2) in $\tilde{O}(m)$

└ Goemans-Williamson primal dual

k-cut LP

(edge costs $c_e > 0$)

$$\min \sum_{e \in E} c_e x_e \quad \text{over } x : E \to \mathbb{R}$$

$$\text{s.t.} \sum_{e \in T} x_e \geq k-1 \quad \forall \text{ spanning trees } T$$

$$0 \leq x_e \leq 1 \quad \forall \text{ edges } e$$

k-cut LP for k=2 (min-cut)

$$\min \sum_{e\in E} c_e x_e \quad \text{over } x: E\to\mathbb{R}$$

$$\text{s.t.} \quad \sum_{e\in T} x_e \ge \cancel{k-1}\ 1 \quad \forall \text{ spanning trees } T$$

$$0 \le x_e \cancel{\le 1} \quad \forall \text{ edges } e$$

Pure covering LP

Dual of min-cut LP

$$\max \sum_T y_T \quad \text{over } y : \left\{ \begin{matrix}\text{spanning} \\ \text{trees}\end{matrix} \right\} \to \mathbb{R}$$

$$\star \quad \sum_{T \ni e} y_T \le c_e \qquad \forall \text{ edges } e$$

$$y_T \ge 0 \qquad \forall \text{ spanning trees } T$$

Pure packing LP

$(1+\varepsilon)$-APX in $\tilde{O}(m/\varepsilon^2)$ deterministic [Chekuri-Q]

k-cut LP

$$\min \sum_{e \in E} c_e x_e \quad \text{over } x : E \to \mathbb{R}$$

$$\text{s.t.} \sum_{e \in T} x_e \geq k-1 \quad \forall \text{ spanning trees } T$$

$$\underline{0 \leq x_e \leq 1} \quad \forall \text{ edges } e$$

Dual of k-cut LP

$$\max \quad (k-1)\sum_T y_T - \underline{\sum_{e\in E} z_e}$$

$$\text{over} \quad \sum_{T\ni e} y_T \le c_e + \underline{z_e} \qquad \forall \text{ edges } e$$

$$y_T \ge 0 \qquad \forall \text{ span. trees } T$$

$$z_e \ge 0 \qquad \forall \text{ edges } e$$

difficult to handle

k-cut LP

$$\min \sum_{e\in E} c_e x_e \quad \text{over } x: E \to \mathbb{R}$$

$$\text{s.t.} \quad \sum_{e\in T} x_e \geq k-1 \quad \forall \text{ spanning trees } T$$

$$\underbrace{0 \leq x_e \leq 1}_{\text{extra packing constraints}} \quad \forall \text{ edges } e$$

Knapsack Covering Constraints [CFLP]

Given covering integer program (CIP)
w/ multiplicity constraints

e.g.

$$\begin{aligned} \min \quad & \sum_j c_j x_j \quad \text{over } x \in \mathbb{Z}^n \\ \text{s.t.} \quad & Ax \geq b \\ & 0 \leq x \leq 1 \leftarrow \end{aligned}$$

where $A \in \mathbb{R}_{\geq 0}^{m \times n}$
$c \in \mathbb{R}_{\geq 0}^{n}$
$b \in \mathbb{R}_{\geq 0}^{m}$

$\forall\ S \subseteq [n]$: write down the residual LP after maxing out all x_j w/ $j \in S$

$$\underbrace{A}_{\text{columns } S \text{ crossed out}} x \geq b - A\mathbb{1}_S$$

Then one can drop multiplicity constraints!

Applying KC inequalities to k-cut

$$\sum_{e \in T} x_e \geq k-1 \qquad \forall \text{ spanning trees } T$$

$\forall\ S \subseteq E$

$$|S \cap T| + \sum_{e \in T \setminus S} x_e \geq k-1 \qquad \forall \text{ spanning trees } T$$

$\Updownarrow$ equivalent to

$$\sum_{e \in F} x_e \geq k-1-(n-1-|F|) \qquad \forall \text{ forests } F$$

$$= |F|-(n-k)$$

k-cut LP w/ KC inequalities

$$\min \sum_{e \in E} c_e x_e \quad \text{over } x : E \to \mathbb{R}$$

$$\text{s.t.} \quad \sum_{e \in F} x_e \geq |F| - (n-k) \quad \forall \text{ forests } F$$

$$0 \leq x_e \cancel{\leq 1} \quad \forall \text{ edges } e$$

Pure covering LP!

Dual LP

$$\max \sum_F (|F|-n-k)\, y_F \quad \text{over } y: \{\text{forests}\} \to \mathbb{R}$$

$$\text{s.t. } \sum_{F \ni e} y_F \le c_e \quad \forall \text{ edges } e$$

$$y_F \ge 0 \quad \forall \text{ forests } F$$

packing forests into graph w/ profits-per-forest depending on $|F|$

We now apply MWU to pack forests

similar to packing trees, but trickier

Tree packings reduce to dynamic MST

Good data structures known for

Here, we need to maintain

$$\max_{\text{forests } F} \frac{|F| - (n-k)}{\sum_{e \in F} w_e}$$

over dynamically increasing edge weights
w_e

Second technical issue - updating edge weights along forests of varying size

w/ some effort we make it work

Min-cut vs 2-APX k-cut

	min-cut	2-APX k-cut	(2+ε)-APX k-cut
Deterministic	$\tilde{O}(mn)$	$\tilde{O}(mn)$	$O\left(\frac{m\log^3(n)}{\varepsilon^2}\right)$
Randomized	$\tilde{O}(m)$	$\tilde{O}(mk)$	

$\boxed{Q1}$ deterministic 2-APX k-cut as fast as randomized?

$\boxed{Q2}$ 2-APX k-cut as fast as min-cut? $\tilde{O}(m)$ randomized?

Very very 2^{very} (very)! nearly

Question:

Can we get 2-APX k-cut in nearly-linear time?

Randomized? Deterministic?

Thank you