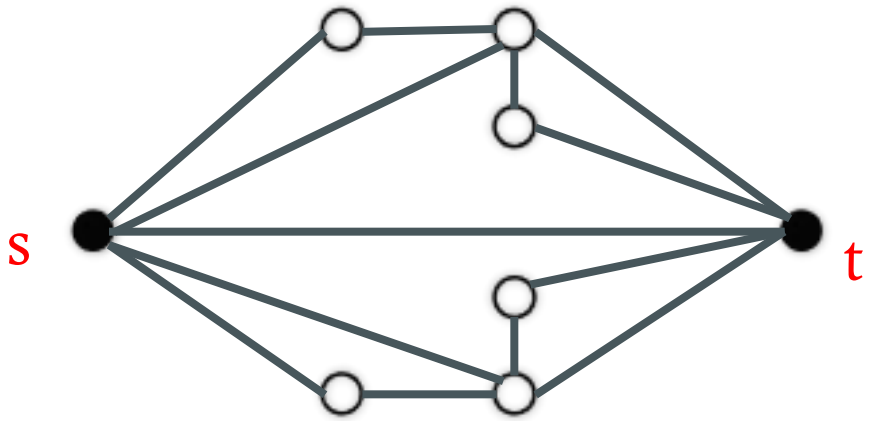
Element Connectivity Preserving Graph Reduction Step

Chandra Chekuri

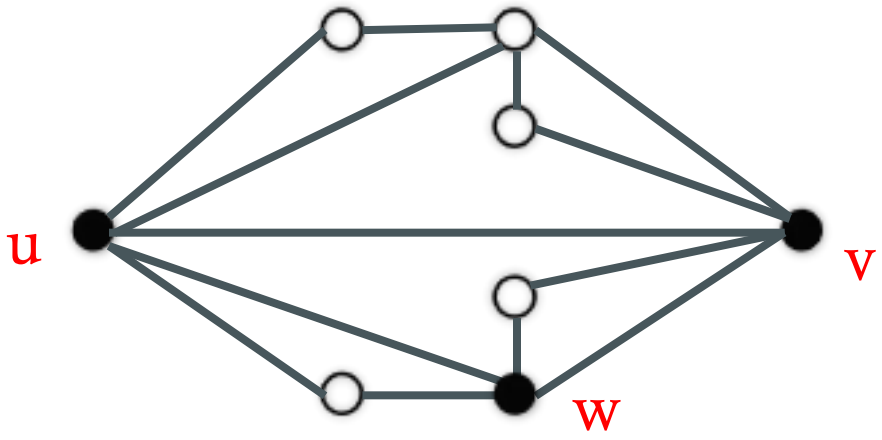
Univ of Illinois, Urbana-Champaign



- Edge connectivity between s and t: max # of edge disjoint paths from s to t. $\lambda_G(s,t)$
- Node connectivity between s and t: max # of internally node disjoint paths from s to t. $\kappa_G(s,t)$

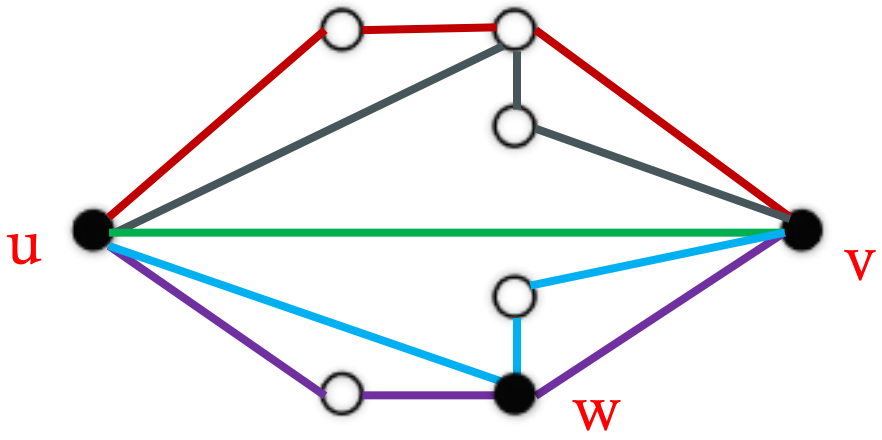
Global connectivity: minimum over all pairs

Element Connectivity



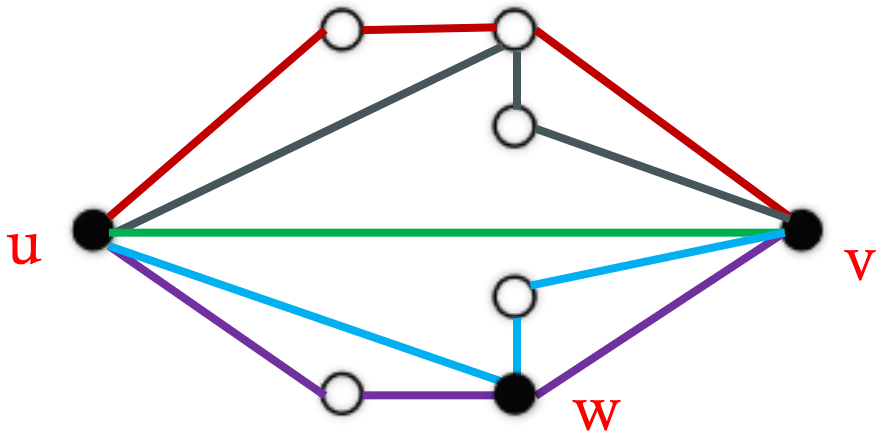
- V(G) partitioned into terminals T and non-terminals V\T
- Terminals (black nodes) and non-terminals (white nodes)
- Elements: edges and white nodes
- Element connectivity between terminals u and v: max # of element disjoint paths from u to v. $\kappa'_G(u,v)$

Element Connectivity



- V(G) partitioned into terminals T and non-terminals V\T
- Terminals (black nodes) and non-terminals (white nodes)
- Elements: edges and white nodes
- Element connectivity between terminals u and v: max # of element disjoint paths from u to v. $\kappa'_G(u,v)$

Element Connectivity



- V(G) partitioned into terminals T and non-terminals V\T
- If T = V(G) element-connectivity coincided with edge-connectivity
- If T = {s, t} $\kappa_G(s,t) = \kappa'_G(s,t)$

Motivation/Applications

[Frank-Ibaraki-Nagamochi'93] first ref for this notion

- Network Design [Jain-Mandoiu-Vazirani-Williamson'99, Jain-Fleischer-Williamson'01, Cheriyan-Vempala-Vetta'01, Chuzhoy-Khanna'09, Nutov'09, …]
- Packing element-disjoint Steiner trees/forests [Hind-Oellerman'96, Cheriyan-Salavatipour'05, Aazami-Cheriyan-Jampani'09, C-Korula'09]
- Node-capacitated routing problems, treewidth etc [C-Ene'13, C-Chuzhoy'14]

Rest of the Talk

- Packing element-disjoint Steiner trees/forests
  - A *graph reduction* step to achieve above
- Algorithmic aspects of element-connectivity and reduction step

Goal: highlight some open problems

Classical Results on Packing in Undirected Graphs

Menger's theorem: max # of edge/internally-node disjoint s-t paths is equal to min s-t edge/node cut

NashWilliams-Tutte theorem: max # of edge disjoint spanning trees in a graph equal to partition connectivity. *Special case of packing bases in a matroid.*

Packing Steiner trees & forests

$G=(V,E)$ undirected graph

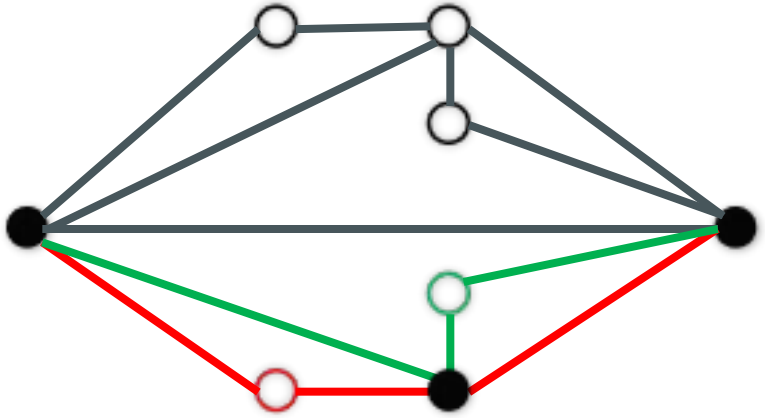
- Terminal set $T \subseteq V$. How many edge/internally node disjoint *Steiner trees* for T in G?
- Terminal sets $T_1, T_2, \ldots, T_h$. How many edge/internally node disjoint *Steiner forests* in G?

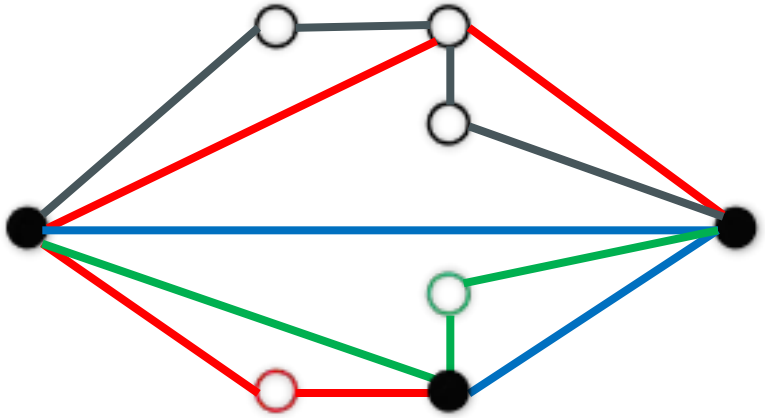
Packing internally node-disjoint Steiner trees

[Hind-Oellerman'96]

Menger's theorem: max # of internally node-disjoint s-t paths is equal to min s-t node-cut = $\kappa_G(s,t)$

What about max # of internally node-disjoint Steiner trees for a terminal set T in G when $|T| > 2$?





Packing internally node-disjoint Steiner trees

[Hind-Oellerman'96]

Menger's theorem: max # of internally node-disjoint s-t paths is equal to min s-t node-cut = $\kappa_G(s,t)$

What about max # of internally node-disjoint Steiner trees for a terminal set T in G when $|T| > 2$?

Upper bound: $k = \min_{u,v \text{ in } T} \kappa'_G(u,v)$ the global element-connectivity of T

Packing internally node-disjoint Steiner trees

Upper bound: $k = \min_{u,v \text{ in } T} \kappa'_G(u,v)$

$t = |T|$

[Hind-Oellerman'96]: Upper bound not tight. For $|T| = 3, 4$ # is $\left\lfloor \frac{1}{t-1} \lceil t\ k/2 \rceil \right\rfloor$

[Cheriyan-Salavatipour'05] In any graph # of element-disjoint trees is $\Omega(k/\log t)$ and this is tight modulo constant

Packing internally node-disjoint Steiner trees

Upper bound: $k = \min_{u,v \text{ in } T} \kappa'_G(u,v)$

$t = |T|$

[Cheriyan-Salavatipour'05] In any graph # of element-disjoint trees is $\Omega(k/\log t)$ and this is tight modulo constant

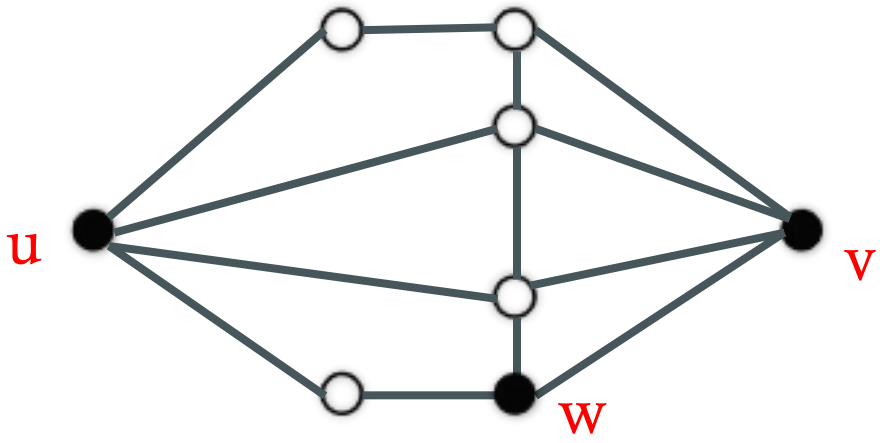
[Aazami-Cheriyan-Jampani'09] For planar graphs $\frac{k}{2} - 1$ and $\Omega(k)$ in an proper minor-closed family of graphs. Based on theorem of [Frank-Kiraly-Kriesell'03] on packing spanning sub-graphs in hypergraphs

Key tool: a graph simplification step

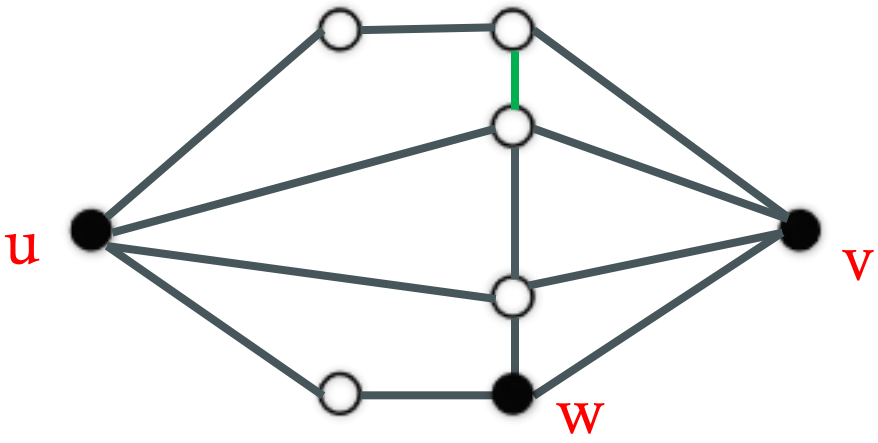
[Hind-Oellerman'96]

Theorem: For any edge pq where p, q are *non-terminals* one can *delete* pq or *contract* pq such that $\kappa'(T)$ is unchanged

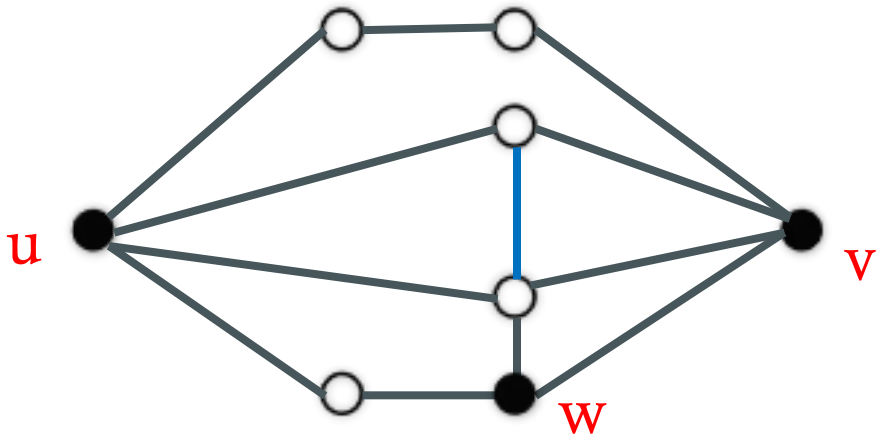
$\kappa'_G(u,v) = 4$
$\kappa'_G(u,w) = 3$
$\kappa'_G(v,w) = 3$



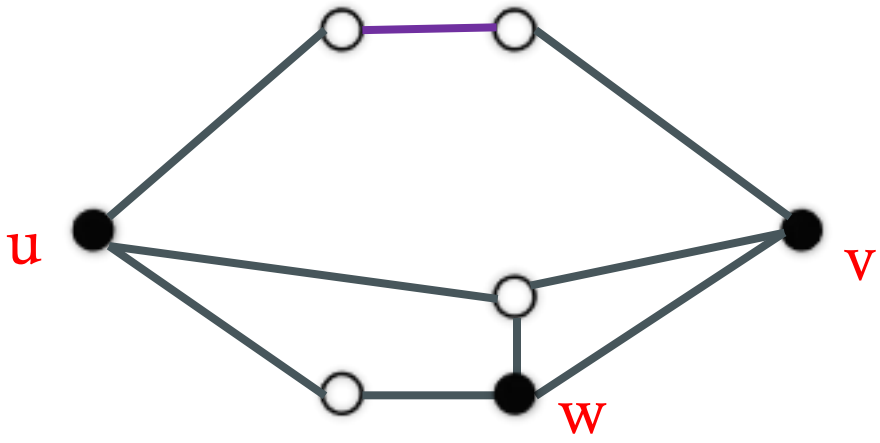
$\kappa'_G(u,v) = 4$
$\kappa'_G(u,w) = 3$
$\kappa'_G(v,w) = 3$



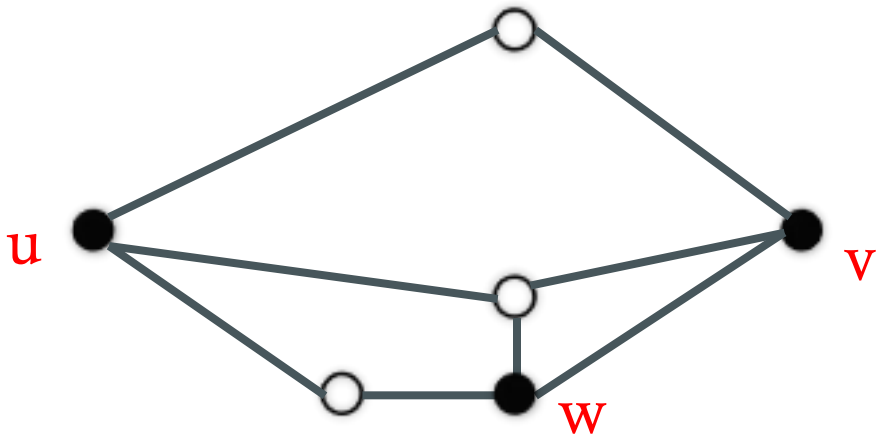
$\kappa'_G(u,v) = 4$
$\kappa'_G(u,w) = 3$
$\kappa'_G(v,w) = 3$



$\kappa'_G(u,v) = 4$
$\kappa'_G(u,w) = 3$
$\kappa'_G(v,w) = 3$



$\kappa'_G(u,v) = 4$
$\kappa'_G(u,w) = 3$
$\kappa'_G(v,w) = 3$



Global element connectivity is preserved but not local element connectivity

Key tool: a graph simplification step

[Hind-Oellerman'96] **Theorem:** For any edge pq where p, q are *non-terminals* one can *delete* pq or *contract* pq such that $\kappa'(T)$ is unchanged

[C-Korula'09] **Theorem:** For any edge pq where p, q are *non-terminals* one can *delete* pq or *contract* pq such that for *all* terminal paris u,v, $\kappa'(u,v)$ is unchanged.

[C-Rukkanchanunt-Xu] proof recast in terms of uncrossing arguments

After reduction

- Start with G end with graph H
- For all terminal pairs u,v, $\kappa'_G(u,v) = \kappa'_H(u,v)$
- H is a *minor* of G
- H is a *hypergraph* on node set T
  - $\kappa'_G(u,v)$ = min u-v hypergraph cut in H
  - κ' is a symmetric submodular function on T
  - Implies that κ' admits a Gomory-Hu tree

Back to packing element-disjoint trees

After reduction:

Given hypergraph $H = (T, \mathcal{E})$ how many hyper-edge-disjoint connected graphs are there?

Cheriyan-Salavatipour Algorithm

- Given G obtain reduced graph $H = (T, \mathcal{E})$
- For each $e \in \mathcal{E}$ random color from 1 to $c\, k/\log |T|$

Claim: each color class is a connected hypergraph with constant probability

Implies we obtain $\Omega(k/\log t)$ trees in expectation

Packing bases in polymatroids

For graphs/matroids

NashWilliams-Tutte theorem:

- $\mathcal{M} = (E, \mathcal{I})$ is a matroid with rank function r
- Max # if disjoint bases is $\left\lfloor \min_{S} \frac{|E \backslash S|}{r(E) - r(S)} \right\rfloor$

[Frank-Kiraly-Kriesell'03] generalize graph case to hypergraphs via partition connectivity

Packing bases in polymatroids

Polymatroid: E is ground set with monotone integer valued submodular function $f : 2^E \to R_+$

$E' \subseteq E$ is a **base** if $f(E') = f(E)$

Max # of disjoint bases is **at most** $k = \left\lfloor \min_S \frac{\sum_{e \in E \setminus S} f_S(e)}{f(E) - f(S)} \right\rfloor$

Packing bases in polymatroids

Polymatroid: E is ground set with monotone integer valued submodular function $f : 2^E \to R_+$

$E' \subseteq E$ is a **base** if $f(E') = f(E)$

Max # of disjoint bases is **at most** $k = \left\lfloor \min_S \frac{\sum_{e \in E \setminus S} f_S(e)}{f(E) - f(S)} \right\rfloor$

[Calinescu-C-Vondrak'09]

Theorem: For any polymatroid there are $\Omega(k/\log f(E))$ disjoint bases. Tight modulo constant.

Algorithm is same

- For each e in E random color from 1 to c k/log f(E)

Claim: each color class is a base with constant probability

Comments:

- Also works by choosing a random permutation. Can be derandomized via min-wise independent premutations.
- Better than k/log f(E) is hard

Open problem

Theorem: For any polymatroid there are $\Omega(k/\log f(E))$ disjoint bases. Tight modulo constant.

Open problem:

Can we replace $\log f(E)$ by $\log (\max_e f(e))$?

Fractional packing possible via [Wolsey]

Packing *edge-disjoint* Steiner trees

Given undirected G=(V,E) and terminals $T \subseteq V$ how many *edge-disjoint* T-trees are there?

Kriesell's conjecture: If T is 2k-edge-connected then there are k edge-disjoint T-trees

[Lau'05]: True if T is 24k-edge-connected

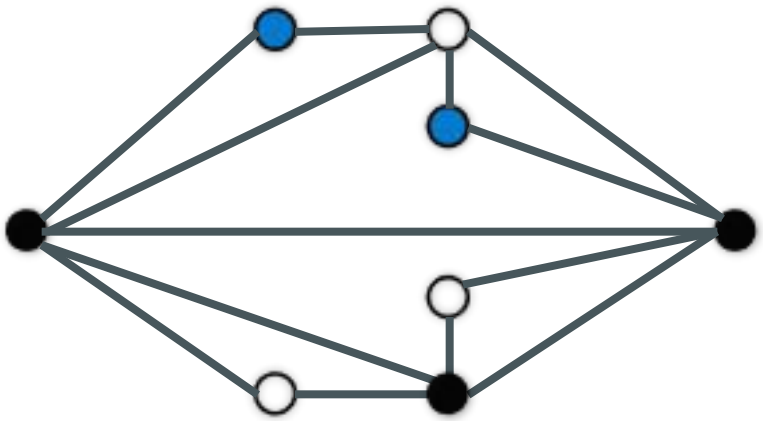
[West-Wu'10] : True if T is 6.5k edge-connected

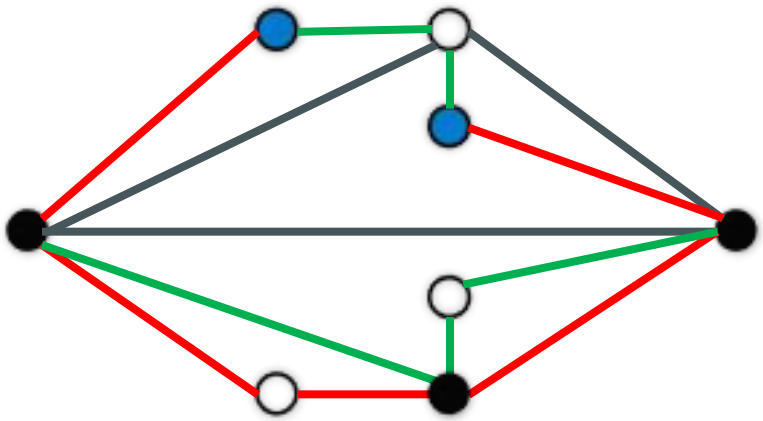
[DeVos-McDonald-Pivotto'13] : (5k+4) edge-connectivity sufficient

For planar graphs 2k+2 sufficient via elem connectivity

Packing Steiner forests

Given G=(V,E) and disjoint terminal sets $T_1, T_2, \ldots, T_h$ how many edge/element disjoint Steiner forests?





Packing Elem-Disjoint Steiner forests

[C-Korula'09]

If each T_i is k-element-connected then there are $\Omega(k/(\log |T| \log h))$ element-disjoint Steiner forests

In *planar* graphs $\frac{k}{5} - 1$ disjoint Steiner forests

In *bounded genus* graphs $\Omega(k)$ disjoint Steiner forests

Different technique from that of [Aazami-Cheriyan-Jampani] and does not rely on [Frank-Kiraly-Kriesell]

Packing Edge-Disjoint Steiner forests

[Lau'05]

If each T_i is k-ege-connected then there are $k/30$ edge-disjoint Steiner forests

Open Problems

	General graphs	Planar	Minor-closed
Edge-disj Steiner trees	$k/5$ $k/2$	$k/2$ $k/2$	$k/5$ $k/2$
Edge-disj Steiner forests	$k/32$ $k/2$	$k/5$ $k/2$	$k/32$ $k/2$
Elem-disj Steiner trees	$k/\log t$	$k/2$	$\Omega(k)$
Elem-disj Steiner forests	$k/(\log t \log h)$ $k/\log t$	$k/5$ $k/2$	$k/(\log t \log h)$ $O(k)$

Capacitated Problems

Results on planar/minor-closed graphs apply only to *uncapacitated case*

Open Problem: extend to capacitated case

Conjectured results hold for *fractional packing* via [Demaine-Hajiaghayi-Klein] on node-weighted network design

Special Case

$G=(V,E)$ is from a proper minor-closed family

T terminal set and $T' \subset T$

T' is k-element connected in G (not all of T)

Question: are there $\Omega(k)$ disjoint T'-Steiner trees ?

Yes in planar graphs and surface embedded graphs [C-Korula]

[Aajami-Cheriyan-Jampani] technique seems to fail

Switching Topics

Algorithmic Aspects

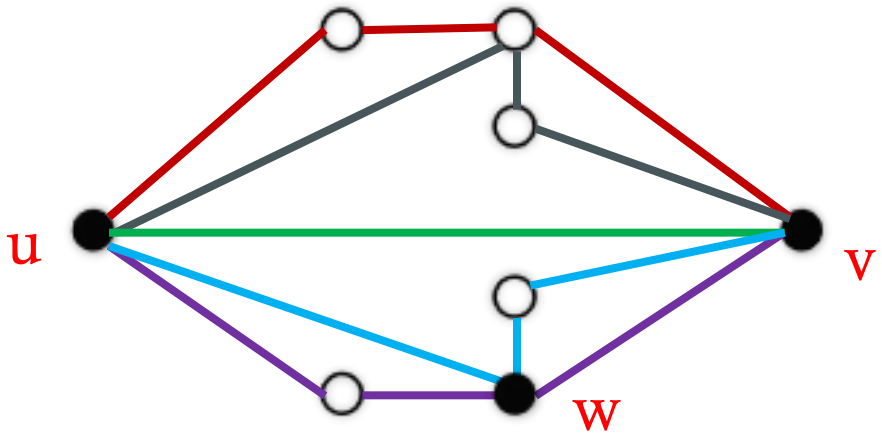
Given $G=(V,E)$ and $T \subseteq V$

$|V| = n$, $|E| = m$, $|T| = t$

- Given s, t *how fast* can we compute $\kappa'_G(s,t)$?
- Global element connectivity? $\kappa'_G(T)$
- All-pair element connectivity?
- Reduction step? Obtaining H from G

[C-Rukkanchanunt-Xu'15]

Single Pair



Can reduce to s-t flow in node-capacitated undir graphs

Can reduce to s-t flow in *unit capacity* directed graph

Single Pair

MF(n,m): max-flow in unit-capacity directed graphs

- $O(m \min(n^{2/3}, \sqrt{m}))$ [Even-Tarjan, Karzanov]
- $\oplus(m^{10/7})$ [Madry]
- $\oplus(m\sqrt{n})$ [Lee-Sidford]

All-Pair Connectivity

Recall that κ' is a symmetric submodular function on T

Construct a Gomory-Hu tree using $|T|-1$ single-pair computations

Hence $O(|T|\ MF(m,n))$ time to compute all-pair and global connectivity

Global Connectivity

Compute all-pair connectivity and take min ☹.

Open problem: O(nm) time algorithm? Can do this for hypergraphs via [Klimmek-Wagner'96].

Open problem: Can O(nm) for global mincut in hypergraphs be improved?

Reduction

Given $G=(V,E)$ and $T \subseteq V$

$|V| = n$, $|E| = m$, $|T| = t$

Want to reduce G to hypergraph H preserving element connectivity of terminals

Naïve Algorithm

While there is an edge $e=pq$ between non-terminals

1. $G_1 = G - e$, $G_2 = G/e$
2. Compute all-pair element-connectivities in G_1, G_2
3. Delete e or contract e accordingly

Run time: $O(m\ t\ MF(n,m))$

A little less naïve algorithm

Idea: deleting edge e or contracting edge e does not change connectivity by more than 1

Compute Gomory-Hu tree R=(T, F) for κ'

For each edge $uv \in F$ maintain a flow achieving κ'(uv)

While there is an edge e=pq between non-terminals

1. $G_1 = G - e$, $G_2 = G/e$
2. For each edge $uv \in F$

   Update flow for uv to see if it changed in O(m) time via residual graph
3. Delete e or contract e accordingly

Run time: $O(t\ MF(n,m) + t\ m^2)$

A less naïve algorithm

Fix a non-terminal node p

In $O(t\ m)$ time figure out if p is deletable or find an edge pq such that pq can be contracted

Run time: $O(t\ MF(n,m) + t\ nm)$

Check if p is deletable

Compute Gomory-Hu tree R=(T, F) for κ'

For each edge $uv \in F$ maintain a flow achieving κ'(uv)

For each edge $uv \in F$

Update flow for uv after removing p in O(m) time

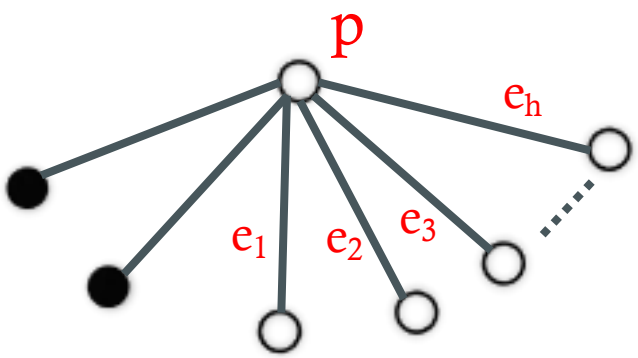
p is deletable iff no flow values change

Finding contractible edge incident to p

i: smallest index such that e_i is not deletable

Can do binary search to find i

Can avoid by being a bit more clever



Run Times

	Edge Connetivity	Element	Node
s-t mincut	MF(n,m), $O(n^{2.375})$	MF(n,m)	MF(n,m), $O(n^{\omega})$
Global min cut	$\oplus$(m)	O(t MF(n,m)) ?	$\oplus$(nm)
All pair connectivity	$\oplus$(nm)	O(t MF(n,m)) ?	
Reduction step	N/A	O(t nm) ?	N/A

Approximations

Recent breakthroughs: $\oplus(m)$ time algorithm for $(1-\epsilon)$ approximation to s-t maxflow in undirected graphs

Open problems:

- $\oplus(m)$ time algorithm for $(1-\epsilon)$ or *constant factor* for s-t node-disjoint paths in unit-cap undirected graphs
- Obtain faster than $O(nm)$ time for *constant factor* approx to global min-cut in hypergraphs. Can obtain $c/\log n$ approx in $O(m \log n)$ time using packing result
- Obtain $O(nm)$ or better for *constant factor* to global element connectivity.

Thank You