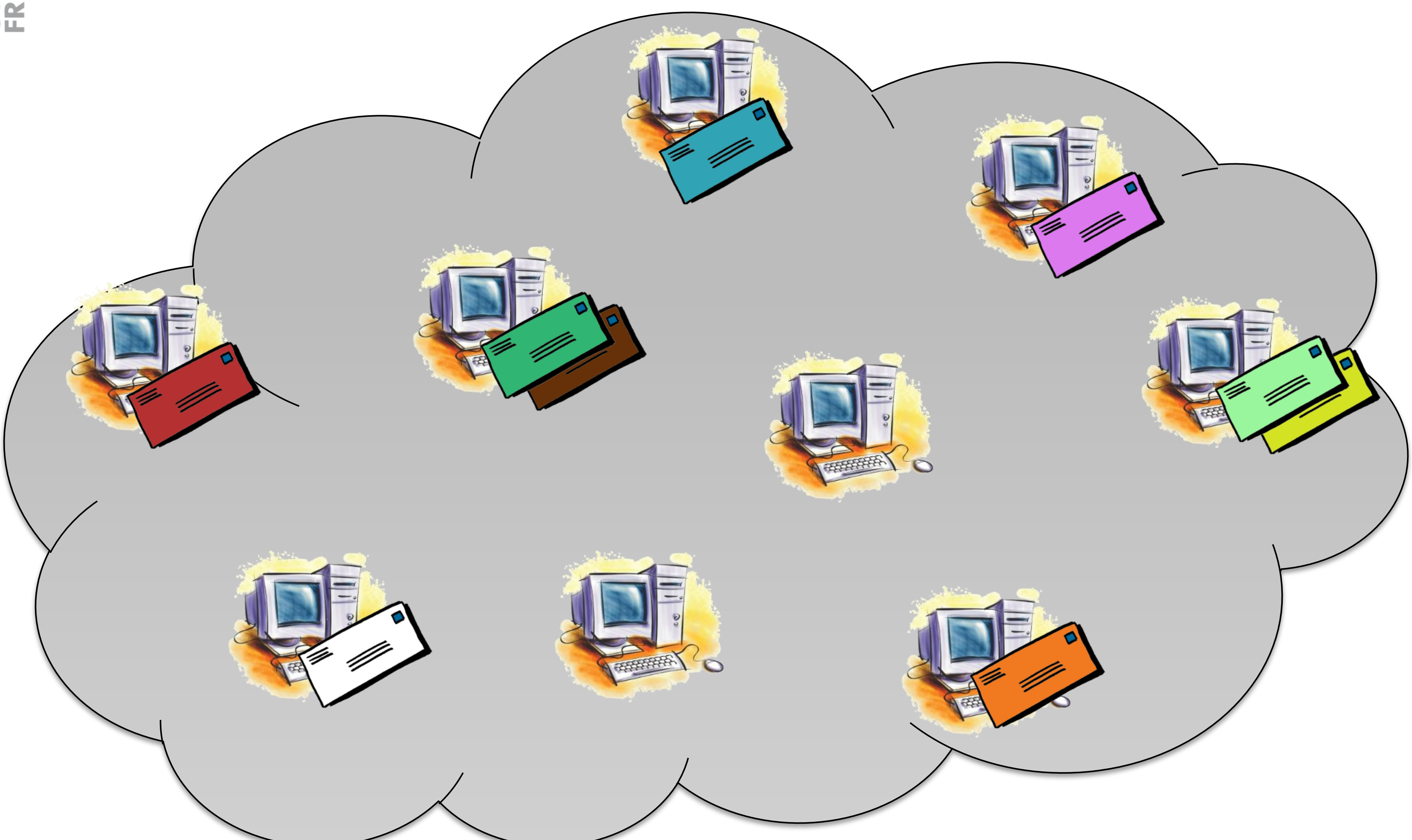
Decomposing Vertex Connectivity
and the
Cost of Multiple Broadcasts

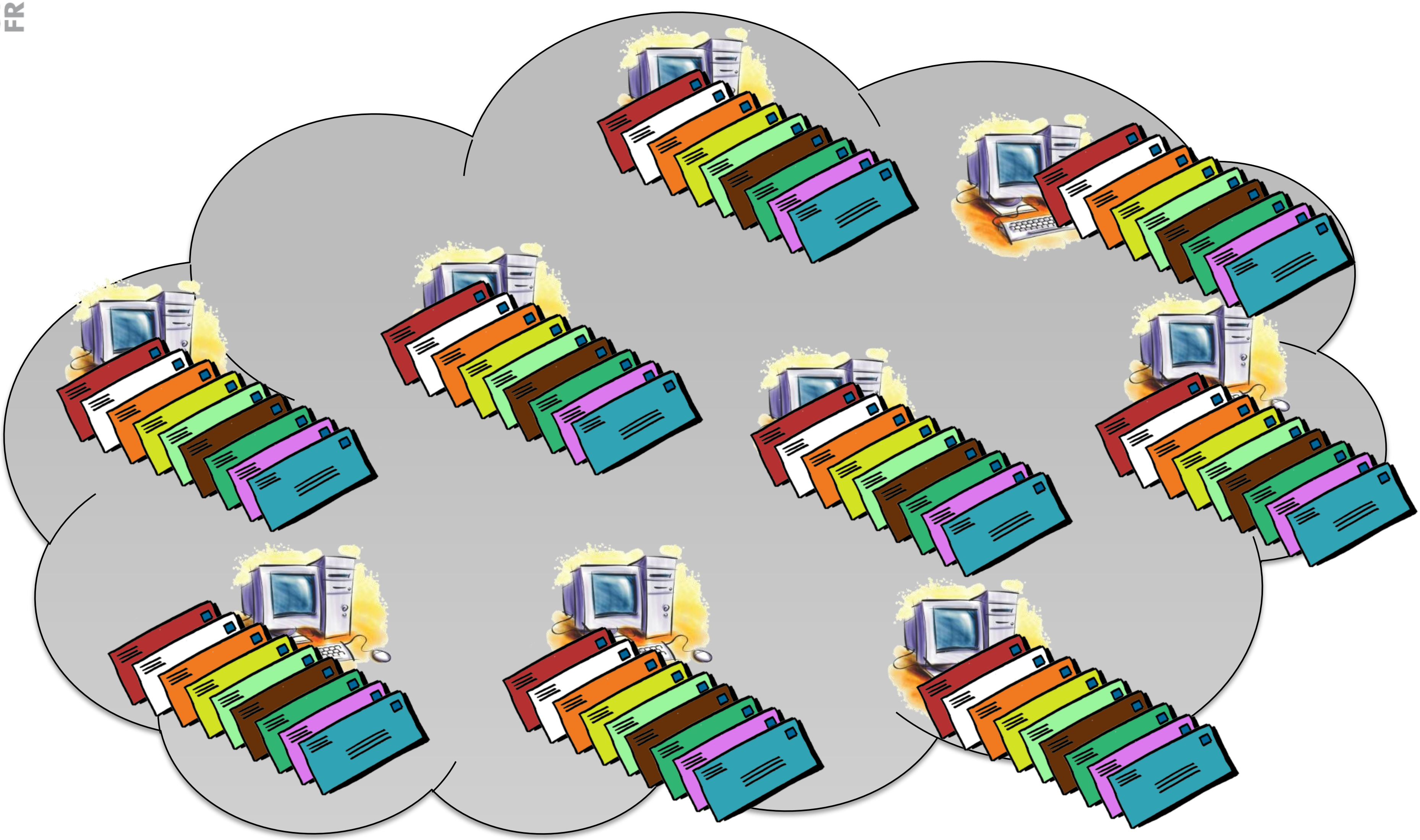
Fabian Kuhn
University of Freiburg, Germany

Based on joint work with
Mohsen Ghaffari (MIT) and Keren Censor-Hillel (Technion)

Multi-Message Broadcast



Multi-Message Broadcast

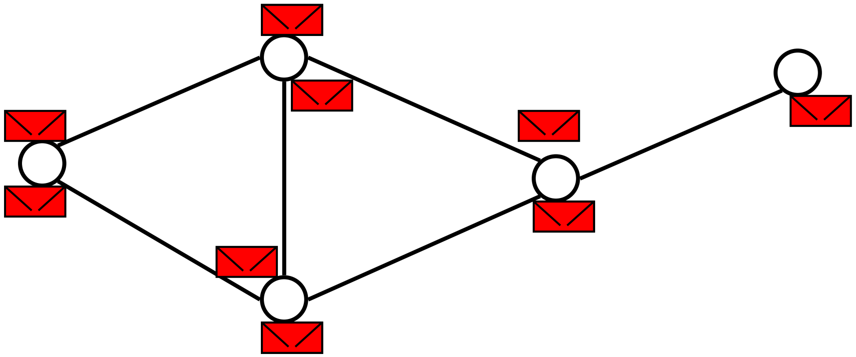


- a.k.a. gossip, token dissemination, …

Multi-Message Broadcast

Communication Assumptions

- For simplicity: synchronous model
- **In each round**:
  Each node can send a message to each neighbor



- **Message size**: $O(\log n)$ bits, $O(1)$ broadcast messages
  – a.k.a. CONGEST model [Peleg 2000]

Broadcasting Multiple Messages

Goal: (Globally) broadcast N messages

> **Strategy:** In each round, each node forwards an "unforwarded" message to its neighbors

Which message should be forwarded to neighbors?

- It doesn't matter…

> **Total time for N broadcasts $\leq D + N$** [Topkis '85]

- D: diameter
- Optimal pipelining on a path of length d gives $O(d + N)$
  - **$D + N$ is asymptotically optimal in general**

- What about networks with better connectivity?

Communication Model

Two natural variants…

Edge-Capacitated Model

- Message size $O(\log n)$
- Nodes can send different messages to different neighbors
- Classic CONGEST model

Node-Capacitated Model

- Message size $O(\log n)$
- Have to send the same message to all neighbors
- Communication by local broadcasts

Multi-Broadcast with Edge Capacities

Basic assumption:

- store-and-forward algorithms

Each message M:

- Edges on which M is forwarded induce a spanning tree!

Throughput (N messages):

- N spanning trees, one for each message
- Optimize throughput:
  - try to use each edge as few times as possible

Packing Spanning Trees

Spanning Tree Packing: set of edge-disjoint spanning trees

Spanning tree packing of size $s \implies$ throughput $\Omega(s)$

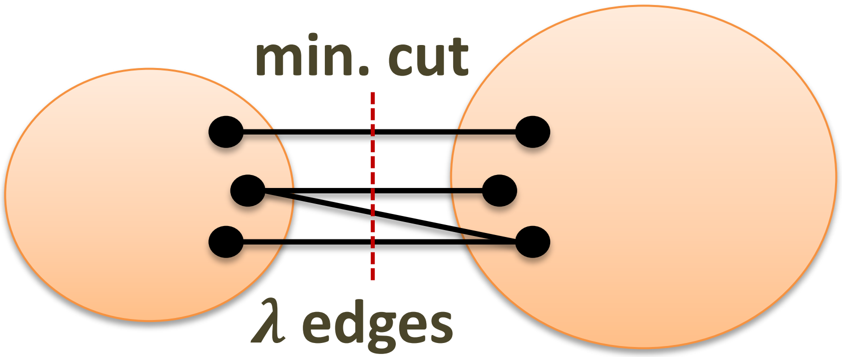
- sp. tree packing of size $s \iff s$ edge-disjoint sp. trees

Proof sketch:

- Each spanning tree gets $\approx N/s$ messages
- Spanning trees don't interfere with each other
- Use pipelining on each spanning tree

Edge Connectivity

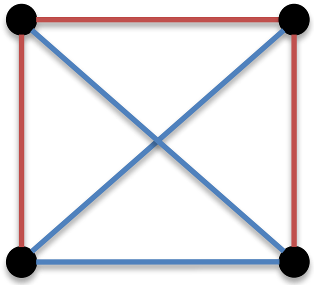
G has edge connectivity λ:



Thm: G has $\leq \lambda$ edge-disjoint spanning trees.

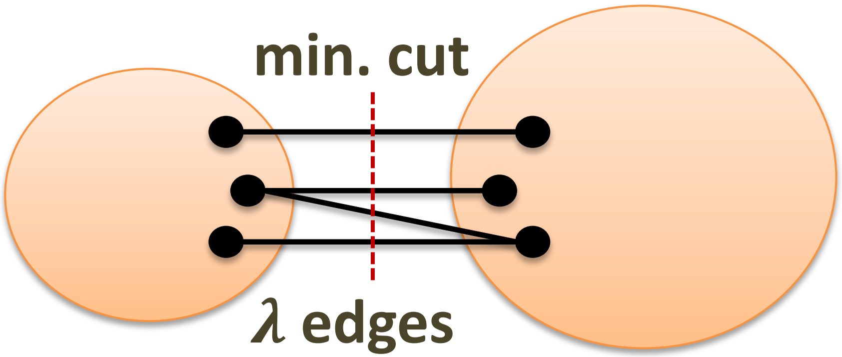
Thm: G has $\geq \lambda/2$ edge-disjoint spanning trees.

[Tutte '61, Nash-Williams '61]



Edge Connectivity

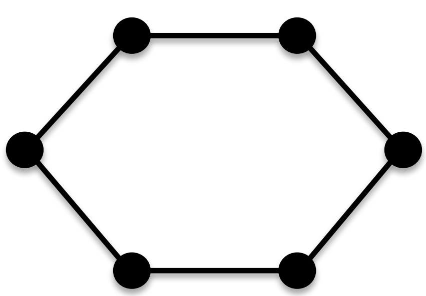
G has edge connectivity λ:



Thm: G has $\leq \lambda$ edge-disjoint spanning trees.

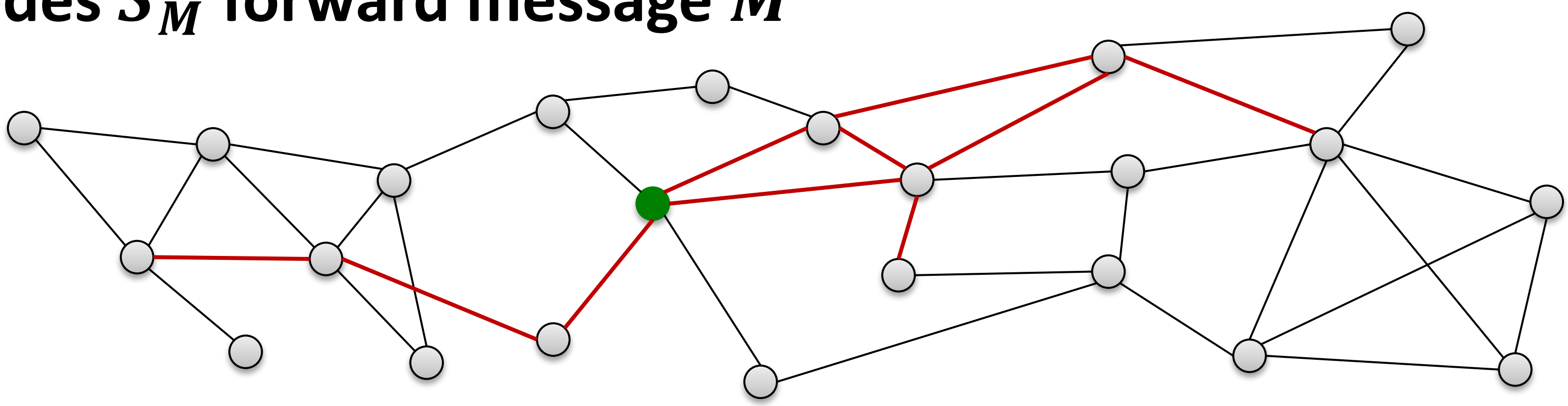
Thm: G has $\geq \lambda/2$ edge-disjoint spanning trees.
[Tutte '61, Nash-Williams '61]

- This is tight:



Vertex-Capacitated Networks?

Nodes S_M forward message M



Every other node needs to get the message:

- S_M **is a** ***dominating set***

One source $\Longrightarrow$ nodes in S_M are connected to each other

- S_M **is a** ***connected dominating set (CDS)***

One CDS for each message M

- Use each node in as few CDSs as possible

Packing Connected Dominating Sets

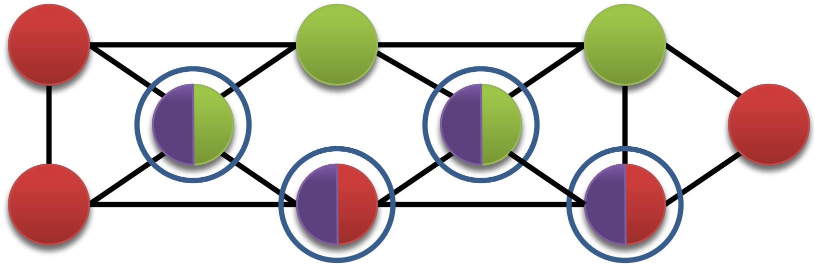
CDS packing of size c

- c vertex-disjoint connected dominating sets

Fractional CDS packing of size c

- CDSs $S_1, \dots, S_t$ and weights $\lambda_1, \dots, \lambda_t$ such that

$$\sum_{i=1}^{t} \lambda_i = c, \qquad \forall v \in V(G): \sum_{i: v \in S_i} \lambda_i \leq 1$$



CDS Packings and Throughput

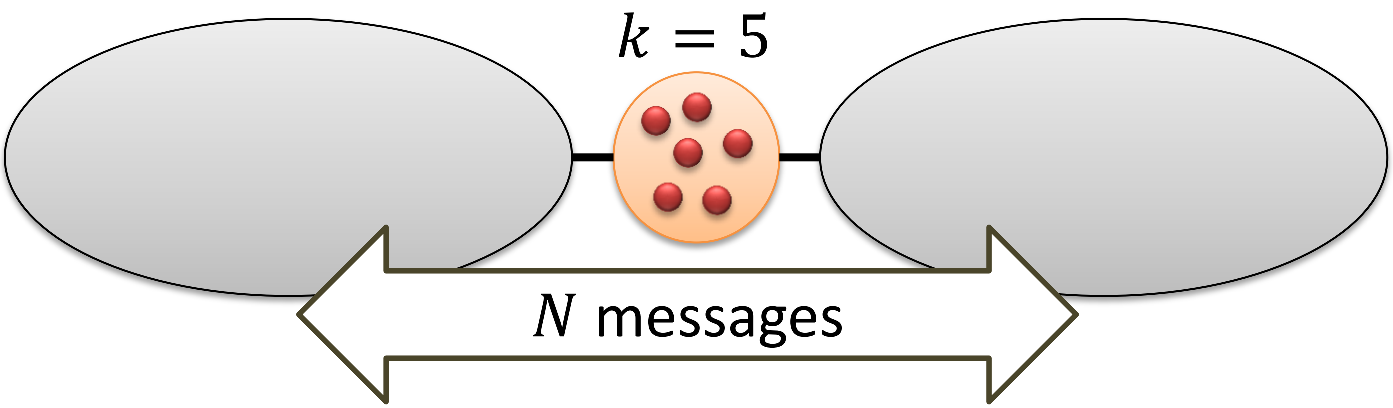
Fractional CDS packing of size c $\Longleftrightarrow$ throughput $\Omega(c)$

~~Proof sketch:~~ Some Intuition

- Distribute msg. among CDSs (according to weight)
  - Time-share between CDSs according to weight
- Use pipelining on each CDS (optimal throughput)
- Throughput $\Omega(c)$:
  - Tracking routes gives CDS for each message
  - Each nodes used at most $O(N/c)$ times
  - CDS S used by ℓ messages $\Longrightarrow$ weight of S is $\Theta(\ell c/N)$

Vertex Connectivity

G has vertex connectivity k:



- Vertex cut $C \subseteq V(G)$
  - Each msg. needs to be forwarded by some node in C

$$\text{throughput} \leq k$$

Thm: Size of largest fractional CDS packing $\leq k$

- Can we find a (fractional) CDS packing of size $\Omega(k)$?

CDS Packing Results

- Joint work with Mohsen Ghaffari and Keren Censor-Hillel

Thm: There is a family of graphs with vertex connectivity k and maximum fractional CDS packing size $O(k/\log n)$.

Thm: Every graph with vertex connectivity $k \geq 1$ has a fractional CDS packing of size $\Omega(1 + k/\log n)$.

Thm: Every graph with vertex connectivity $k \geq 1$ has a CDS packing of size $\Omega(1 + k/\log^5 n)$.

Vertex Sampling Results

CDS results/techniques lead to other interesting results

Thm: If each node of a k-vertex connected graph is indep. sampled with probability p, the vertex connectivity of the induced sub-graph is $\Omega(kp^2/\log^3 n)$.

- Proof idea: Fractional CDS packing construction can also be applied to sampled sub-graph.

Thm: When sampling with prob. $p = \gamma \cdot \log(n)/\sqrt{k}$, the induced sub-graph is connected w.h.p.

- Tight up to factor $O\left(\sqrt{\log n}\right)$.

- No non-trivial results of this kind where known before!

Edge Sampling

Graph G is λ-edge connected:

- $\boldsymbol{H}$: sub-graph induced when independently sampling each edge with probability $p = \Omega(\log(n)/\lambda)$.

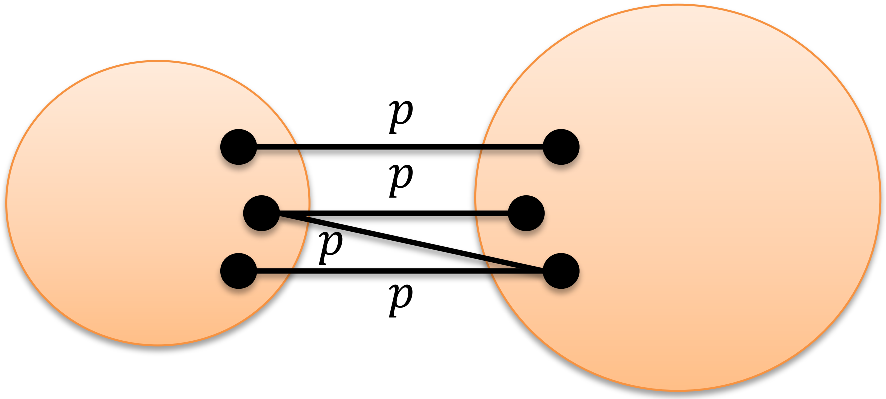
- H is a connected graph, w.h.p. [Lomonosov and Poleskii '71]

- The edge connectivity of H is $\Omega(\lambda p)$, w.h.p. [Karger '94]

- Both results are tight

Cuts After Sampling

- A cut with value $\alpha\lambda$ in G has expected value $p \cdot \alpha\lambda$ in H



- Chernoff bound: the probability that the value is far by a factor $\geq (1+\varepsilon)$ is $e^{-\Theta(\varepsilon^2 p\alpha\lambda)}$

- Union bound: values of **all** cuts are close to expectation

- **Main tool:** number of edge cuts of size $\leq \alpha\lambda$ is $O(n^{2\alpha})$

Number of Cuts

G is λ-edge connected:

- Number of edges cuts of size $\leq \alpha\lambda$ is $O(n^{2\alpha})$ [Karger '94]
- Number of min. edge cuts is $O(n^2)$

G is k-vertex connected:

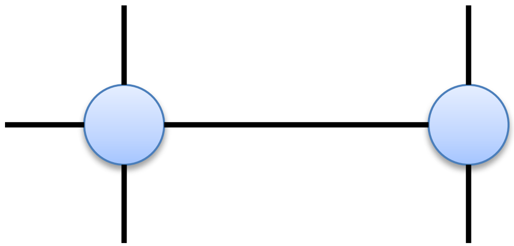
- Number of min. vertex cuts can be $\Theta\left(2^k (n/k)^2\right)$.

Our results: Tools for analyzing vertex connectivity

Vertex Sampling Proof

$H \subseteq G$: Sub-graph with nodes sampled independently with probability $p \geq \beta \log(n)/\sqrt{k} \implies H$ **connected, w.h.p.**

Proof Sketch:



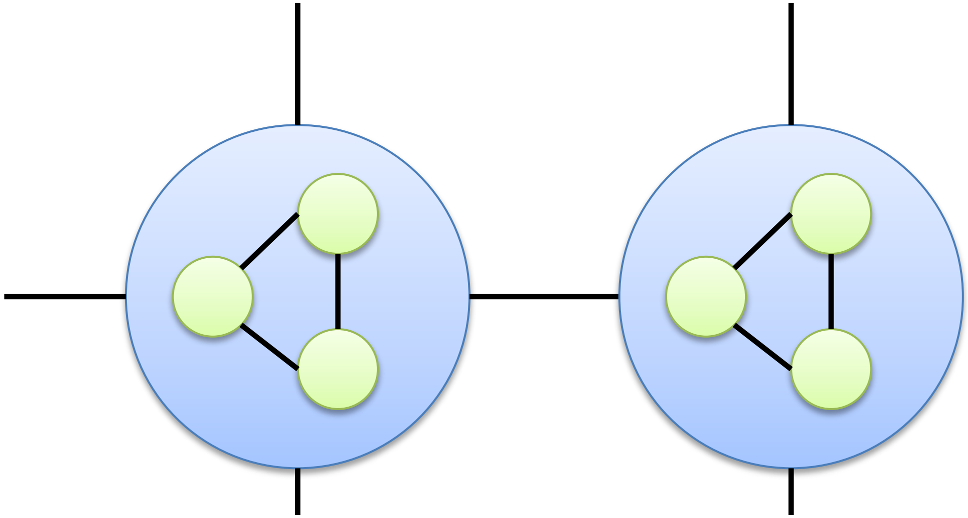
Vertex Sampling Proof

$H \subseteq G$: Sub-graph with nodes sampled independently with probability $p \geq \beta \log(n)/\sqrt{k} \implies H$ connected, w.h.p.

Proof Sketch:

Virtual graph G' with

$L = \Theta(\log n)$ layers



Edge between copies of same node or of neigboring nodes

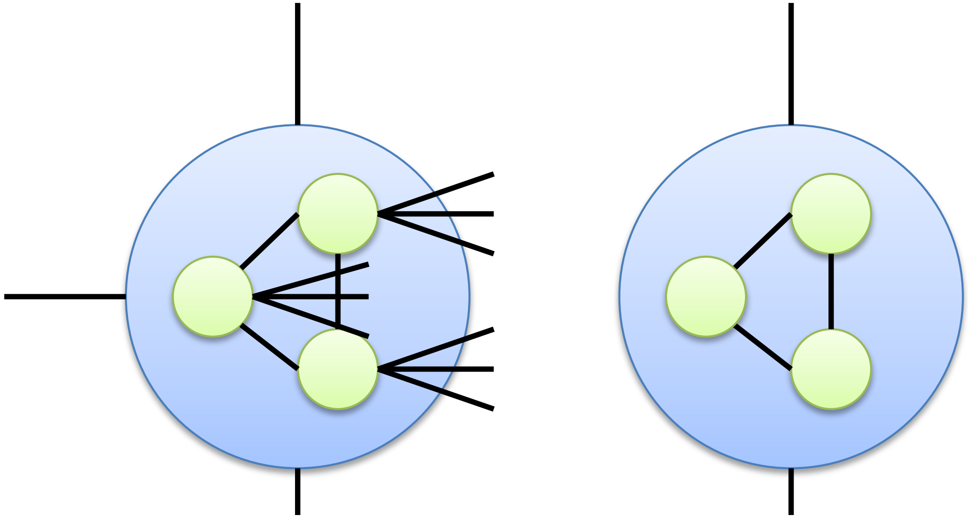
Vertex Sampling Proof

$H \subseteq G$: Sub-graph with nodes sampled independently with probability $p \geq \beta \log(n)/\sqrt{k} \implies H$ **connected, w.h.p.**

Proof Sketch:

Virtual graph G' with

$L = \Theta(\log n)$ layers

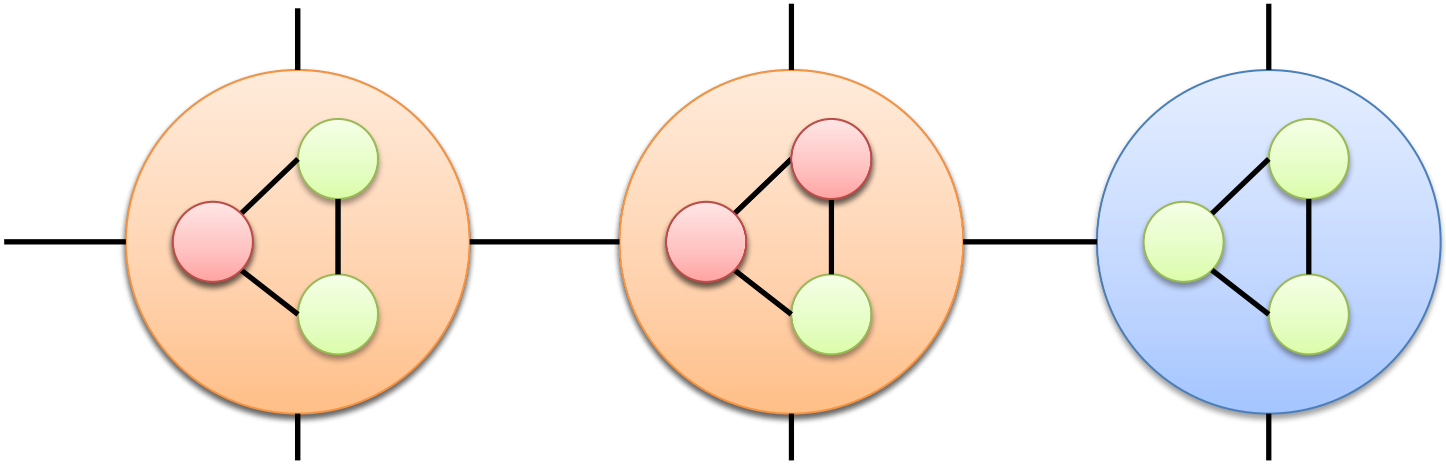


Edge between copies of same node or of neigboring nodes

Virtual Graph

Node set $\boldsymbol{W'} \subseteq \boldsymbol{V'}$ is projected to $\boldsymbol{W} \subseteq \boldsymbol{V}$:

$$w \in W \quad \Longleftrightarrow \quad W' \text{ contains a copy of } w$$



W' connected $\Longleftrightarrow$ W connected

W' dominating $\Longleftrightarrow$ W dominating

Coupling Argument

- Sample virtual nodes with probability

$$\boldsymbol{q = 1 - (1-p)^{1/L} \approx \frac{p}{L}}$$

- Sample real node v iff v is in the projection of the sampled virtual nodes (at least one copy of v sampled in G')

  Happens with **probability $\boldsymbol{1-(1-q)^L = p}$**

- Show that **sampling in $\boldsymbol{G'}$ gives a CDS**

Idea: sample layer by layer and study progress

Domination

Claim: After sampling $L/2$ layers, the sampled nodes form dominating set.

Proof Sketch:

- Sampling probability in G after $L/2 = \Theta(\log n)$ layers is

$$\frac{\Theta(\log n)}{\sqrt{k}}$$

- Domination follows directly because every node in G has degree $\geq k$

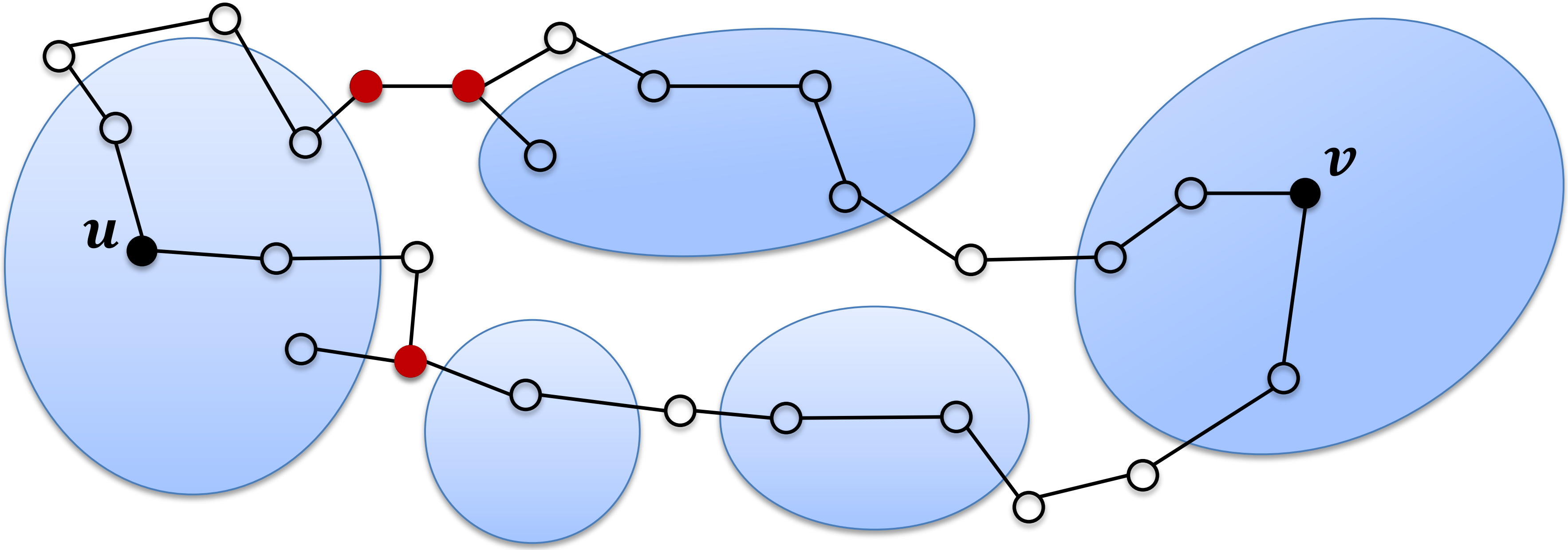
Connectivity

Recall Menger's theorem:

- In a k-vertex connected graph $G = (V, E)$, any two nodes are connected by k internally vertex-disjoint paths

Assume: G is k-vertex connected, $S \subseteq V$ is a dominating set

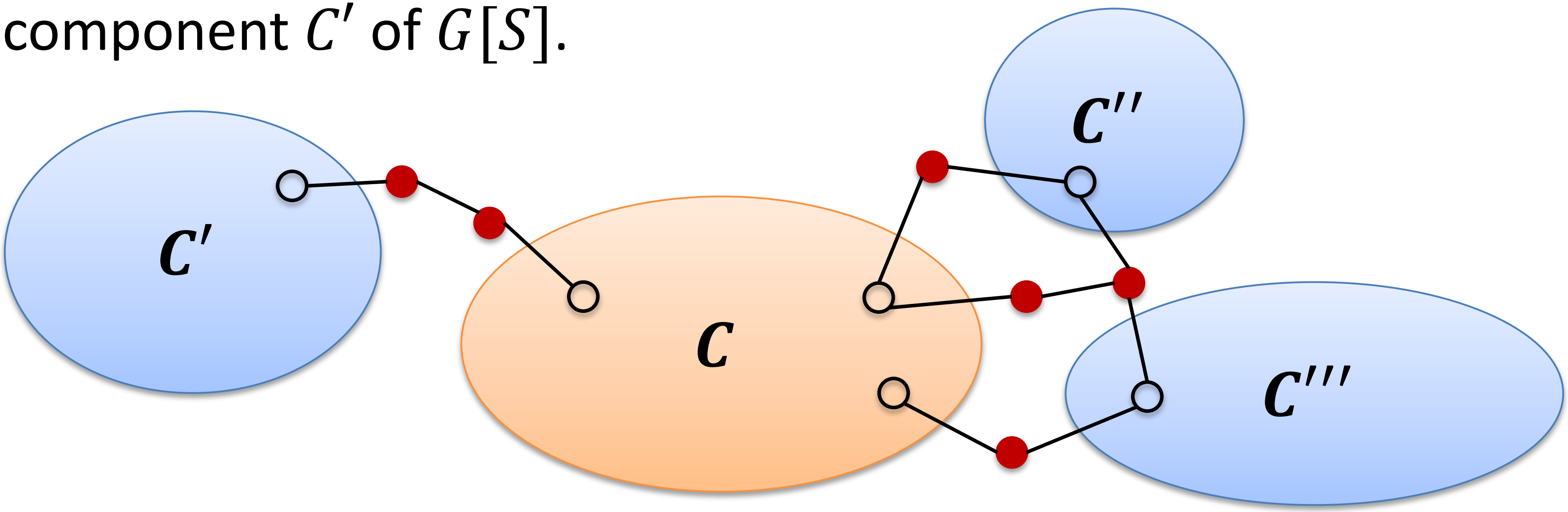
Components of $G[S]$:



Connector Paths

Assume: $G = (V, E)$, $S \subseteq V$ a dominating set

Definition: For a component C of $G[S]$, a *connector path* is a path with ≤ 2 internal nodes connecting C to another component C' of $G[S]$.



Menger & Domination of S:

- G k-vertex connected $\Longrightarrow$ there are $\geq k$ such paths!

Connectivity

Consider a layer $\ell > L/2$

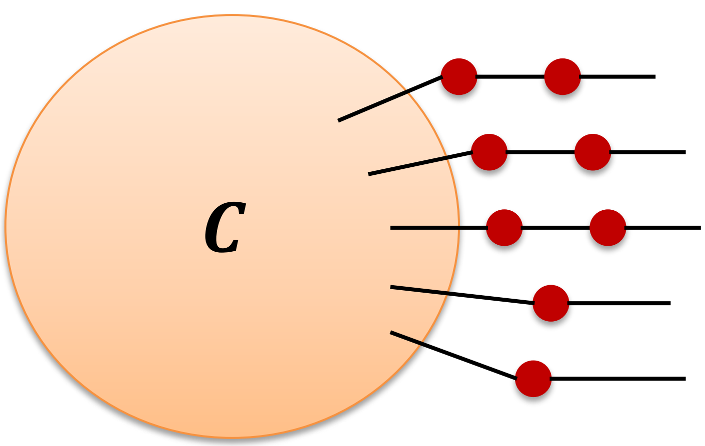
- Nodes $S_{<\ell}$ sampled by layers $< \ell$ form a dominating set

Sampling of layer ℓ:

- Virtual nodes sampled with probability

$$q \approx \frac{p}{L} = \frac{1}{L} \cdot \frac{\beta \log n}{\sqrt{k}} = \Theta\left(\frac{1}{\sqrt{k}}\right)$$

Component of $G[S_{<\ell}]$:



- $\geq k$ connector paths
- each of them sampled with prob $\Omega(1/k)$ in layer ℓ
- C connected to another component with const. prob.

Connectivity

Fast Merging:

- On each layer $\ell > L/2$, each component is connected to at least one other component with at least constant prob.
- With at least constant probability, the number of connected components of the induced sub-graph is reduced by a constant factor
- After $O(\log n)$ layers, we have connectivity, w.h.p.
  - Initially, #components is $O(n)$

(Fractional) CDS Packing

- Sampling directly gives CDS packing of size $\Omega(\sqrt{k}/\log n)$

From size $\tilde{\Omega}(\sqrt{k})$ to $\tilde{\Omega}(k)$...

- also based on virtual graph and layering
- construct all the CDSs at the same time
- carefully choose / assign connector paths
  - make progress for all CDSs (reduce overall # of components)
- Gives **fractional CDS packing**
  - different virtual copies of the same node in G can go to different CDSs
  - Each node is in at most $O(\log n)$ CDSs

- **CDS packing:**
  Use random layers of real nodes instead of virtual nodes

Distributed Construction

(Fractional) CDS packings of the same quality can be computed in a distributed way in time $\tilde{O}(D + \sqrt{n})$.

- Algorithm works in the CONGEST model with
  - Messages of size $O(\log n)$
  - Capacities at nodes (comm. by local broadcast)

Lower bound

- If k is not known, $\tilde{\Omega}\left(D + \sqrt{n/k}\right)$ rounds needed
- Proof based on techniques from [Das Sarma et al. '12]

Network/Graph Decompositions

Classic Locality Decompositions

- Decompose graph into clusters of small diameter
- Preserve locality, cluster graph sparse, low chrom. #, …
  - e.g., [Awerbuch,Goldberg,Luby,Plotkin ’89], [Awerbuch,Peleg ‘90], [Linial,Saks ’93]
  - leads to efficient algorithms in the LOCAL model

(Fractional) CDS and Spanning Tree Packings

- Decompose nodes / edges of a graph G into components
- Components are connected and they “span” G
- Also useful as a generic tool to build distributed alg.?
  - if we want to exploit the inherent parallelism in networks
  - for CONGEST algorithms…

Approximating Vertex Connectivity

Fractional CDS packing construction gives $\boldsymbol{O(\log n)}$-**approximation** of the **vertex connectivity** k of G.

- Time to construct fractional CDS packing: $\tilde{O}(m)$
- Fastest known non-trivial approximation of vertex conn.
- Best known algorithms:
  - compute k exactly [Gabow '00]: $O(n^2k + \min\{nk^{3.5}, n^{1.75}k^2\})$
  - 2-approximation [Henzinger '97]: $O(\min\{n^{2.5}, n^2k\})$

- Distributed algorithm: $\tilde{O}\left(\min\left\{\frac{n}{k}, D + \sqrt{n}\right\}\right)$

Open Problems

- Close the log / polylog gaps!
- CDS packings as a useful primitive (e.g., for distr. alg.)?
- Other applications of the techniques in distr. algorithms?
- Other uses of the layering / virtual graph idea?
  - In particular, when dealing with graph connectivity…
  - Idea also appears in the context of edge sampling in [Alon '95]

Thanks for your attention!

Questions, Comments?