Iterative rounding approximation algorithms for degree-bounded node-connectivity network design

Takuro Fukunaga
National Institute of Informatics, Japan
JST, ERATO, Kawarabayashi Large Graph Project

Joint work with R. Ravi (CMU), Z. Nutov (Open University of Israel)

July 30, 2013 @ FND workshop

Survivable network design (SND)

Problem

Input:

- an undirected or directed graph $G = (V, E)$
- edge-cost $c : E \to \mathbb{Q}_+$
- terminal set $T \subseteq V$
- connectivity requirements $r : T \times T \to \mathbb{N}$

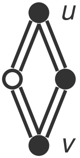
Solution: a minimum cost subgraph of G

Constraints: $\forall u, v \in T$: (connectivity between u and v) $\geq r(u, v)$

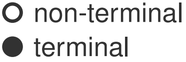
Connectivity

- **edge-connectivity** λ: max # of **edge-disjoint** paths
- **element-connectivity** λ_T:

  max # of paths **disjoint in edges and non-terminals**
- **node-connectivity** κ: max # of paths **disjoint in inner-nodes**



$\lambda(u, v) = 4$
$\lambda_T(u, v) = 3$
$\kappa(u, v) = 2$



Many special cases are defined according to r
(e.g., uniform req., rooted req., subset req.)

Degree-bounded SND

Degree bounds

- Undirected graphs: Given $B \subseteq V$ and $b : B \to \mathbb{N}$,

$$\text{degree of } \forall v \in B \; \leq b(v)$$

- Digraphs: Given $B^-, B^+ \subseteq V$, $b^- : B^- \to \mathbb{N}$ and $b^+ : B^+ \to \mathbb{N}$,

$$\text{in-degree of } \forall v \in B^- \leq b^-(v)$$

$$\text{out-degree of } \forall v \in B^+ \leq b^+(v)$$

Feasible solutions of Degree-bounded SND are Hamiltonian paths

- connectivity requirements: an undirected connected graph
- degree bounds: $B = V$, and $b(v) = 2$ for $\forall v$

➜ **NP-hard to find a feasible solution**

Multi-criteria approximation

Approximation for undirected degree-bounded SND

- $\alpha \in \mathbb{Q}$
- $\beta : \mathbb{N} \to \mathbb{N}$

An algorithm achieves (α, β)**-approximation** if it outputs $F \subseteq E$ such that

- $c(F) \leq \alpha \mathrm{OPT}$ (edge-cost approx)
- degree of $v \leq \beta(b(v))$ for $\forall v \in B$ (degree bounds approx)

for each instance that has a feasible solution.

Key idea: iterative rounding

Iterative rounding is a powerful tool

- Jain '01: 2-approx algorithm for edge-connectivity SND
- Fleischer, Jain, Williamson '01: Extended [Jain '01] to element-connectivity SND and node-connectivity SND w/ $k \leq 2$ ($k := \max_{u,v} r(u, v)$)
- Breakthrough around '07: Applied to degree-bounded spanning tree and degree-bounded SND w/ edge-connectivity req.

Key idea: iterative rounding

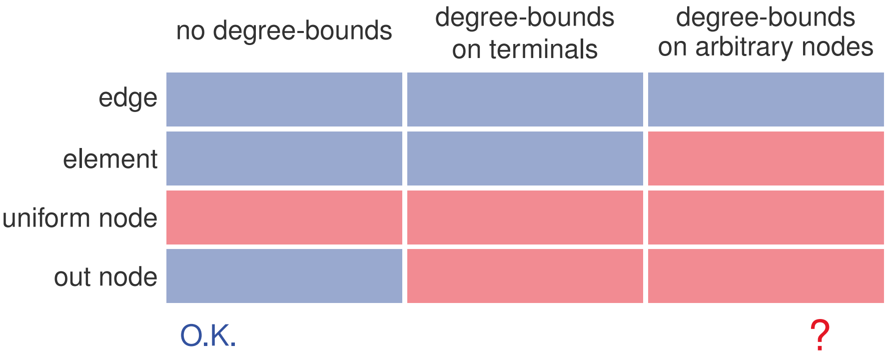
Iterative rounding is a powerful tool

- Jain '01: 2-approx algorithm for edge-connectivity SND
- Fleischer, Jain, Williamson '01: Extended [Jain '01] to element-connectivity SND and node-connectivity SND w/ $k \leq 2$ ($k := \max_{u,v} r(u, v)$)
- Breakthrough around '07: Applied to degree-bounded spanning tree and degree-bounded SND w/ edge-connectivity req.

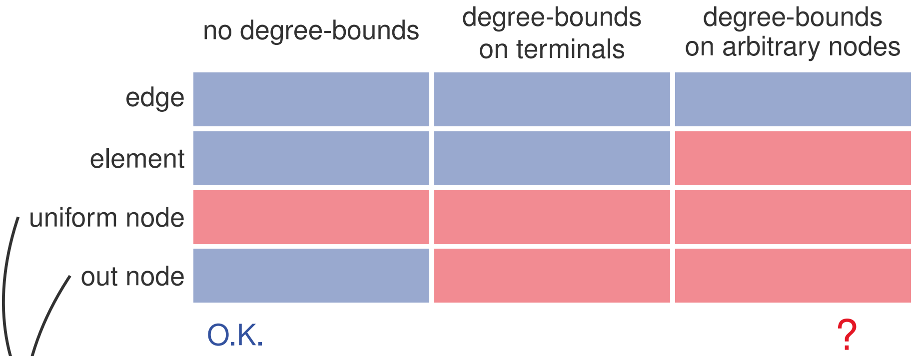
But, **iterative rounding did NOT work well** for

- element-connectivity SND w/ degree-bounds on arbitrary nodes
- node-connectivity SND even w/o degree-bounds if $k \geq 3$

Situation


no degree-bounds
degree-bounds on terminals
degree-bounds on arbitrary nodes
edge
element
uniform node
out node
O.K.
?

Situation

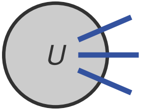


- uniform node-conn. req.: undirected graph, $r(u, v) = k, \forall u, v \in V$
- out node-conne. req.: directed graph, root $s \in V$,

$$r(u, v) = \begin{cases} k & \text{if } u = s, \\ 0 & \text{otherwise.} \end{cases}$$

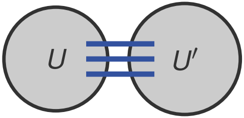
Why were they difficult?

- edge-connectivity SND ➜ covering set functions by edges



$$\geq R(U) := \max_{u \in U, v \notin U} r(u, v)$$

- node-connectivity SND ➜ covering set-pair functions by edges



$$\geq R(U, U') := \max_{u \in U, v \in U'} r(u, v)$$

There was no good analysis of iterative rounding for covering set-pair functions except a few restricted cases.

What did we do?

- We gave two definitions of laminarity for set-pairs
  - Laminarity of set-pairs
  - Strongly laminarity of set-pairs

What did we do?

- We gave two definitions of laminarity for set-pairs
  - Laminarity of set-pairs
  - Strongly laminarity of set-pairs

- We characterized structure on tight set-pair families of element-connectivity and node-connectivity SND
  - Iterative rounding was known to work

    ➜ strongly laminar family (undirected graphs)

    or laminar family, one direction (directed graphs)
  - Iterative rounding was NOT known to work

    ➜ laminar family (undirected graphs)

    or laminar family, both directions (directed graphs)

What did we do?

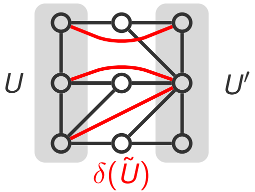
- We gave a new analysis for
  - laminar families (both in undirected and in directed graphs)
  - strongly laminar families w/ degree-bounds
  - no edge-cost case

Our ideas

1. New token counting method for laminar family of set-pairs
2. Using two different counting methods according to # of tight set-pairs v.s. # of tight degree nodes

Set-pair

- set-pair (= biset): ordered pair $\tilde{U} = (U, U')$ of disjoint node sets
- $U :=$ tail, $U' :=$ head
- $\delta(\tilde{U}) := \{uv \in E : u \in U, v \in U'\}$
- $\Gamma(\tilde{U}) := V \setminus (U \cup U')$ (boundary)



LP relaxation

- $R(\tilde{U}) := \max_{u \in U, v \in U'} r(u, v) - |\Gamma(\tilde{U})|$
- $R(\tilde{U}) > 0 \Rightarrow |\Gamma(\tilde{U})| < k$
- $\mathcal{F}$: a family of set-pairs defined depending on the connectivity

Set-pair relaxation for undirected graphs

$$\begin{aligned} \min \quad & c^T x & \\ \text{s.t.} \quad & x(\delta(\tilde{U})) \geq R(\tilde{U}) & \forall \tilde{U} \in \mathcal{F} \\ & x(\delta(v)) \leq b(v) & \forall v \in B \\ & 0 \leq x(e) \leq 1 & \forall e \in E \end{aligned}$$

Laminarity of set-pairs

Laminar family of set-pairs

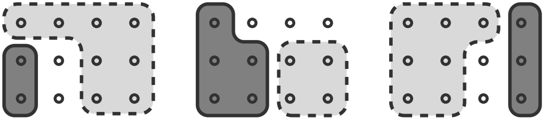
$\mathcal{L}$ is a laminar family of set-pairs if

- $\{U : (U, U') \in \mathcal{L}\}$ is a laminar set family,
- $\forall (U, U'), (W, W') \in \mathcal{L} : U \subseteq W \Rightarrow W' \subseteq U'$.

Strongly laminar family of set-pairs

$\mathcal{L}$ is a strongly laminar family of set-pairs if

- $\mathcal{L}$ is a laminar family of set-pairs,
- $\forall \tilde{U} = (U, U'), \tilde{W} = (W, W') \in \mathcal{L}$:
  $U \cap W = \emptyset \Rightarrow U \cap \Gamma(\tilde{W}) = \emptyset, \Gamma(\tilde{U}) \cap W = \emptyset$.



Laminar
NOT strongly laminar

Results via structure of tight constraints

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary nodes
element			
uniform node			
out node			

Results via structure of tight constraints

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary nodes
element	← strongly	laminar →	laminar
uniform node	←	laminar, if $n > 3k - 3$	→
out node	laminar, only entering arcs	← laminar	→

Results via structure of tight constraints

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary nodes
element	← strongly laminar →		laminar
uniform node	← laminar, if $n > 3k - 3$ →		
out node	laminar, only entering arcs	← laminar →	

1

1. Laminar, undirected ➜ $(O(k), O(k) \cdot b(v))$-approx

Results via structure of tight constraints

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary nodes	
element	⟵ strongly	laminar ⟶	laminar	
uniform node	⟵	laminar, if $n > 3k-3$	⟶	
out node	laminar, only entering arcs	⟵	laminar ⟶	2

1. Laminar, undirected ➜ $(O(k), O(k) \cdot b(v))$-approx
2. Laminar, directed ➜ $(2, k, 2b^{+}(v) + O(k))$-approx

Results via structure of tight constraints

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary nodes	
element	⟵ strongly laminar ⟶		laminar	3
uniform node	⟵ laminar, if $n > 3k - 3$ ⟶			
out node	laminar, only entering arcs	⟵ laminar ⟶		

1. Laminar, undirected ➜ $(O(k), O(k) \cdot b(v))$-approx
2. Laminar, directed ➜ $(2, k, 2b^{+}(v) + O(k))$-approx
3. Strongly laminar w/ degree-bounds, undirected ➜ $(4, 4b(v) + O(k))$-approx

Approximation factors: SND w/o degree bounds

node-connectivity

$k \leq 2$	2-approx	[Fleischer et al. 06]	**iterative rounding**
general	$O(k^3 \log n)$-approx	[Chuzhoy, Khanna 09]	**decomposition**
rooted	$O(k \log k)$-approx	[Nutov 09]	**decomposition**
subset	$O(k^2)$-approx	[Nutov 09]	**decomposition**
uniform	$O(\log^2 k)$-approx	[Fakcharoenphol, Laekhanukit 08]	
		[Nutov 09]	**decomposition**
	$O(\sqrt{n/\epsilon})$-approx	[Cheriyan et al. 06]	**iterative rounding**
	$\Omega(\sqrt{k})$-fractionality	[Aazami et al. 10]	**iterative rounding**
uniform ($n > 3k-3$)	**$O(k)$-approx**	**This talk**	**iterative rounding**

Approximation factors: Edge- and element-connectivity SND w/ degree-bounds

edge-connectivity

	edge-cost	degree	
spanning tree	1	$b(v) + 1$	[Singh, Lau 07]
general	2	$b(v) + O(k)$	[Lau et al. 07]

element-connectivity

edge-cost	deg terminals	deg non-terminals	
2	$b(v) + O(k)$	$+\infty$	[Lau et al. 07]
$O(\log k)$	$O(\log k \cdot b(v) + k)$	$O(2^k) \cdot b(v)$	[Nutov 12]
4	$\mathbf{4b(v) + O(k)}$	$\mathbf{4b(v) + O(k)}$	**This talk**

Approximation factors: Node-connectivity SND w/ degree-bounds

node-connectivity, undirected graphs

	edge-cost	degree	
general	$O(k^3 \log k \log \vert T \vert)$	$O(2^k k^3 \log \vert T \vert) \cdot b(v)$	[Nutov 12]
	$\boldsymbol{O(k^3 \log \vert T \vert)}$	$\boldsymbol{O(k^3 \log \vert T \vert) \cdot b(v)}$	**This talk**
rooted	$O(k^2 \log k \log \vert T \vert)$	$O(2^k k^2 \log \vert T \vert) \cdot b(v)$	[Nutov 12]
	$\boldsymbol{O(k \log k)}$	$\boldsymbol{O(k \log k) \cdot b(v)}$	**This talk**
subset	$O(k^2 \log k \log \vert T \vert)$	$O(2^k k^2 \log \vert T \vert) \cdot b(v)$	[Nutov 12]
$\vert T \vert = O(k)$	$O(k^2)$	$O(k^2)$	trivial
$\boldsymbol{\vert T \vert = \omega(k)}$	$\boldsymbol{O(k \log k)}$	$\boldsymbol{O(k \log k) \cdot b(v)}$	**This talk**

Note: $(+\infty, 2^{\log^{1-\epsilon} n} b(v))$-approx hardness is known for subset node-connectivity SND when k is large [Lau et al. 09]

Approximation factors: Degree-bounded SND for digraphs

node-connectivity, digraphs

	edge-cost	in-degree	out-degree	
out-conn	$O(\log k)$	$+\infty$	$O(2^k) \cdot b^+(v)$	[Nutov 12]
	2	$\boldsymbol{k}$	$\boldsymbol{2b^+(v) + O(k)}$	**This talk**
uniform	$O(k)$	$+\infty$	$O(2^k) \cdot b^+(v)$	[Nutov 12]
	$\boldsymbol{O(k)}$	$\boldsymbol{O(k\sqrt{k})}$	$\boldsymbol{2b^+(v) + O(k\sqrt{k})}$	**This talk**

implications for undirected graphs

	edge-cost	degree	
out-conn	$O(\log k)$	$O(2^k) \cdot b(v)$	[Nutov 12]
	4	$\boldsymbol{2b(v) + O(k)}$	**This talk**
uniform	$O(k)$	$O(2^k) \cdot b(v)$	[Nutov 12]
	$\boldsymbol{O(k)}$	$\boldsymbol{2b(v) + O(k\sqrt{k})}$	**This talk**

Result 1

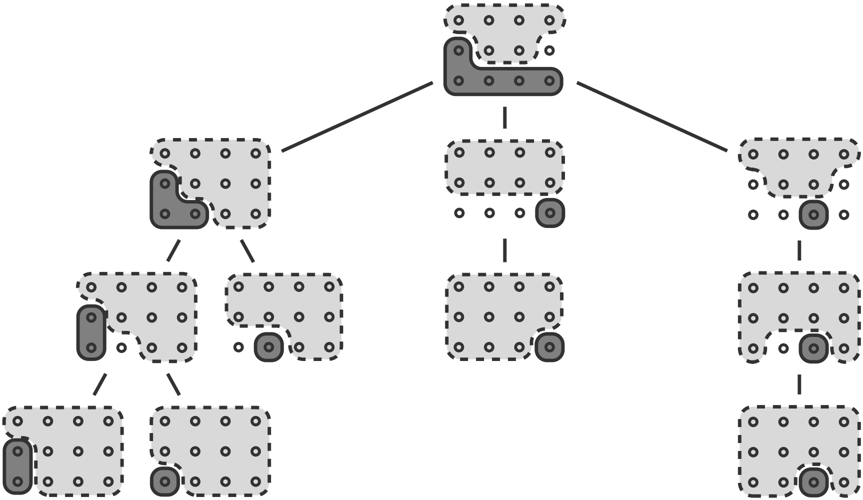
	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary terminals
element	← strongly laminar →		laminar
uniform node	← laminar, if $n > 3k - 3$ →		
out node	laminar, only entering arcs	← laminar →	

1

1. Laminar, undirected ➜ $(O(k), O(k) \cdot b(v))$-approx

Laminar family of set-pairs defines a forest

(U, U') is the parent of (W, W') if $W \subset U$ or if $W = U$ and $U' \subset W'$.



Token distribution

We prove

Theorem

If x^* is uniquely defined from the laminar family of tight set-pairs, one of the following holds:

- $\exists e \in E : x^*(e) = 0$
- $\exists e \in E : x^*(e) \geq 1/(4k-1)$
- $\exists v \in B : |\delta(v)| \leq 4k-1$

Assume $0 < x^*(e) < 1/(4k-1)$ for $\forall e \in E$, and $|\delta(v)| \geq 4k$ for $\forall v \in B$.

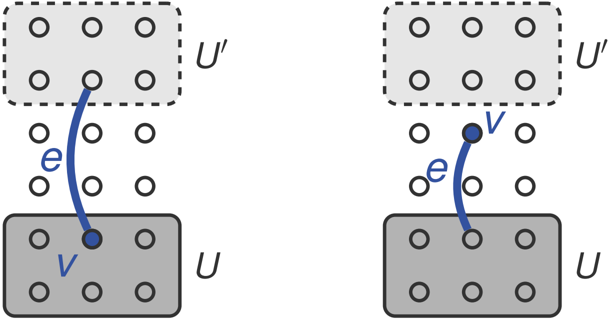
1. We make each edge distributes at most 2 tokens to set-pairs.
2. We show each set-pair receives ≥ 2 tokens, and the root receives ≥ 4 tokens.

Initial distribution

Token distribution rule

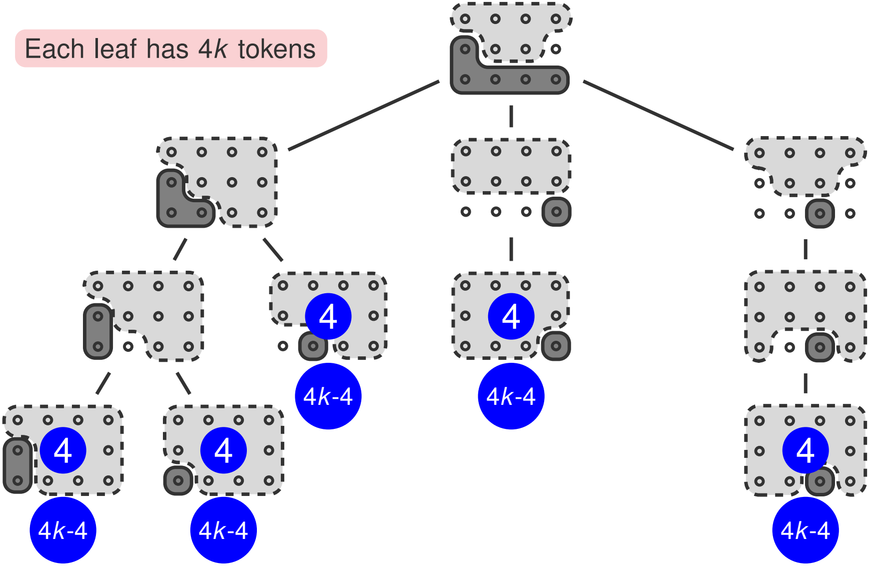
For each $e = uv$ and its end-node v, e gives a token to

1. minimal (U, U') s.t. $e \in \delta(U, U')$ and $v \in U$ if it exists,
2. minimal (U, U') s.t. $e \notin \delta(U, U')$ and $u \in U$ otherwise.



Inductive distribution

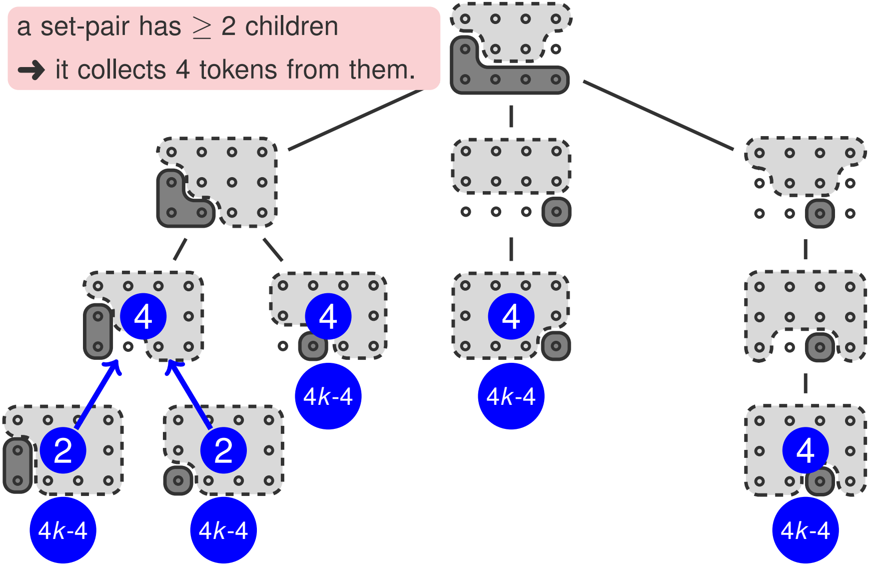
Each leaf has $4k$ tokens



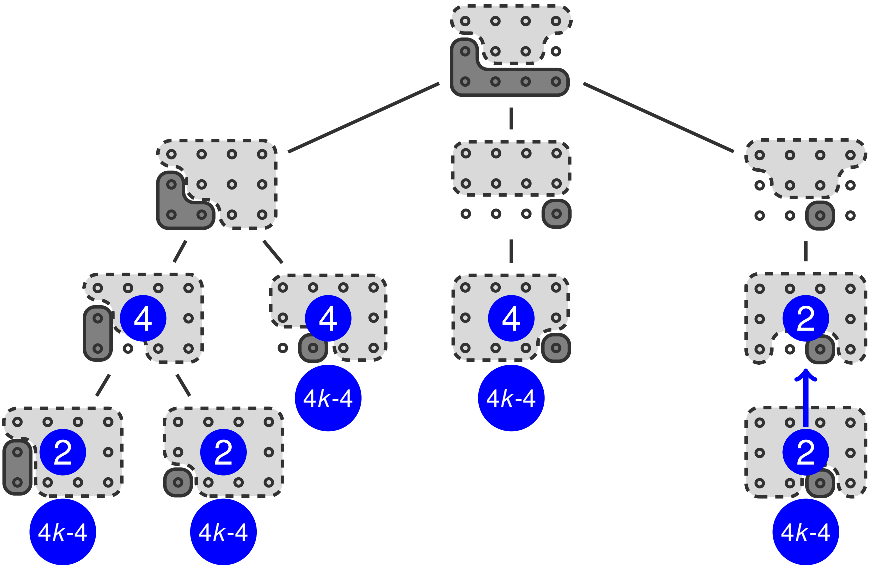
Inductive distribution

a set-pair has $\geq$ 2 children

➜ it collects 4 tokens from them.

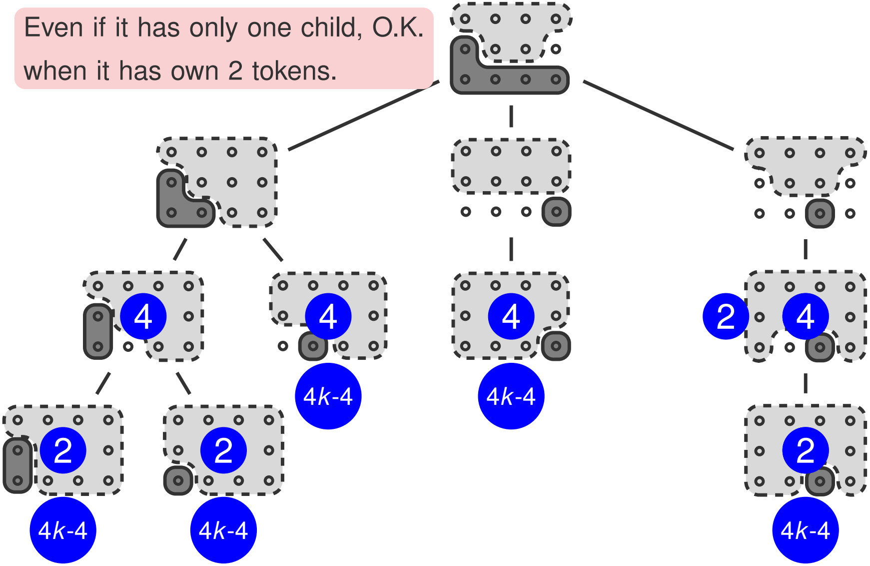


Inductive distribution

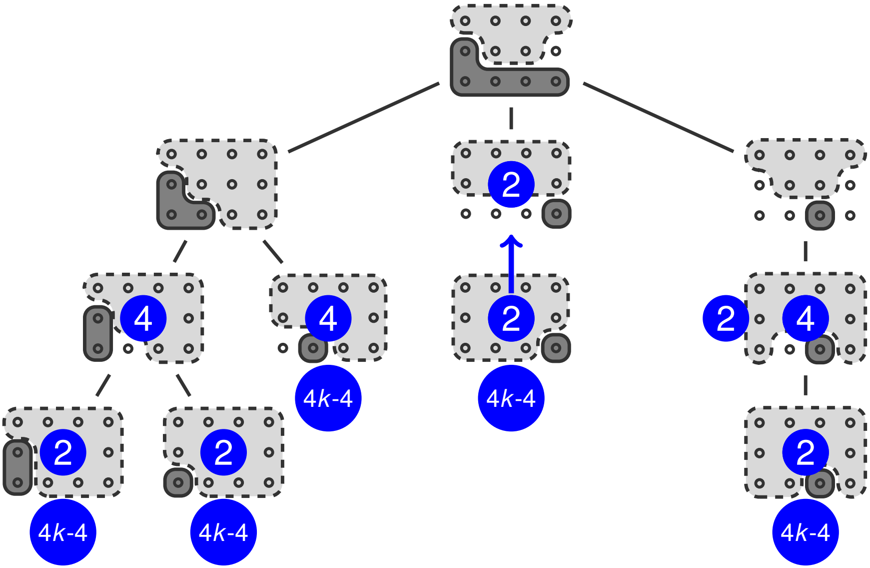


Inductive distribution

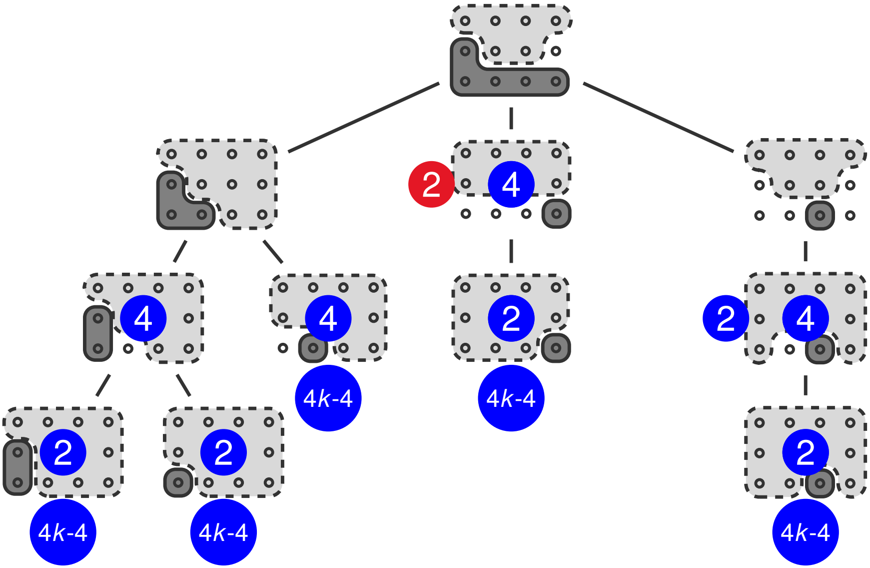
Even if it has only one child, O.K. when it has own 2 tokens.



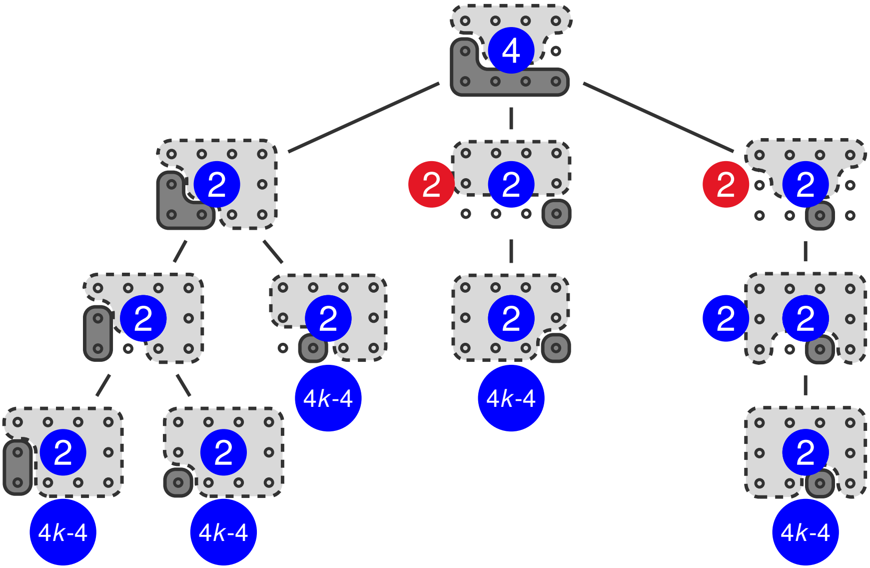
Inductive distribution



Inductive distribution



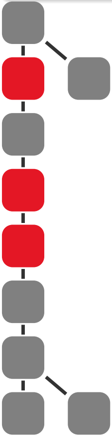
Inductive distribution



How many red tokens?

Theorem

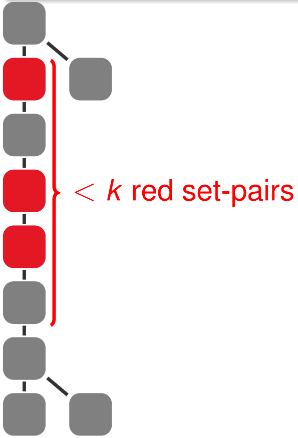
# of red tokens in the forest $< 4(k-1)\times$ (# of leaves)



How many red tokens?

Theorem

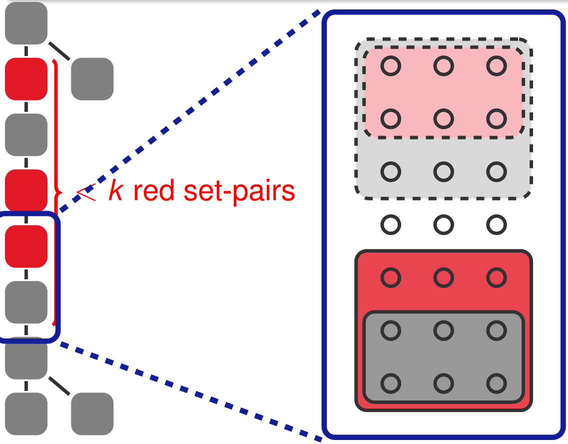
# of red tokens in the forest $< 4(k-1)\times$ (# of leaves)



How many red tokens?

Theorem

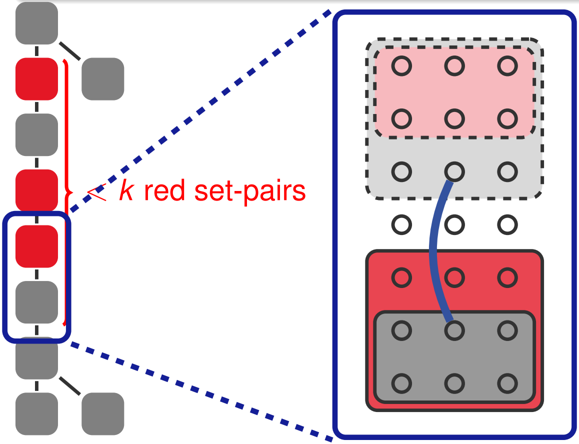
# of red tokens in the forest $< 4(k-1) \times$ (# of leaves)



How many red tokens?

Theorem

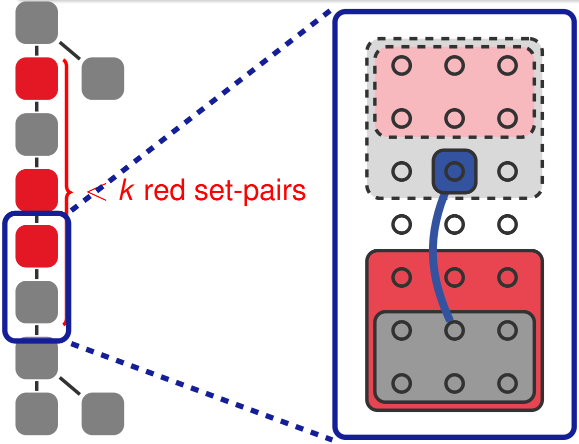
$\#$ of red tokens in the forest $< 4(k-1)\times$ (# of leaves)



How many red tokens?

Theorem

# of red tokens in the forest $< 4(k-1) \times$ (# of leaves)



blue set remains in the boundary of the last set-pair on the path

↓

#red set-pairs on the path $\leq k-1$

↓

#red set-pairs in the forest $< 2(k-1) \times$ #leaves

Result 3

	no degree-bounds	degree-bounds on terminals	degree-bounds on arbitrary terminals
element	⟵ strongly laminar ⟶		laminar
uniform node	⟵ laminar, if $n > 3k - 3$ ⟶		
out node	laminar, only entering arcs	⟵ laminar ⟶	

3

3. Strongly laminar w/ degree-bounds, undirected ➜ $(4, 4b(v) + O(k))$-approx

Strongly laminar family of set-pairs with degree-bounds

Theorem

If x^* is defined from a strongly laminar family of tight set-pairs and tight degree-bounds, then one of the following holds:

- $\exists e \in E : x^*(e) = 0$
- $\exists e \in E : x^*(e) \geq 1/4$
- $\exists v \in B : |\delta(v)| < 2.5k + 6.25$

Idea 2: Using two different counting methods

$\mathcal{L}$:= strongly laminar family of set-pairs
C := set of tight degree-bounded nodes

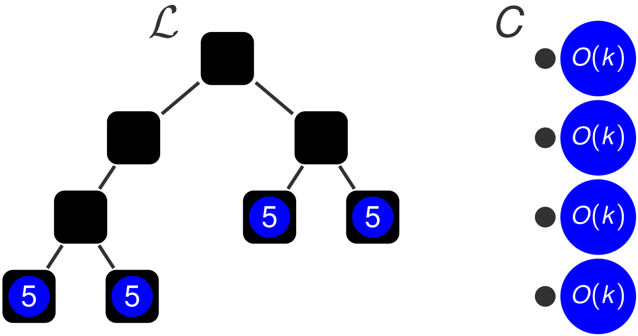
Case (i): C is small (i.e. $2|C| \leq$ # leaves)

A leaf gives tokens to nodes in C, and follow the 2-approx proof without C

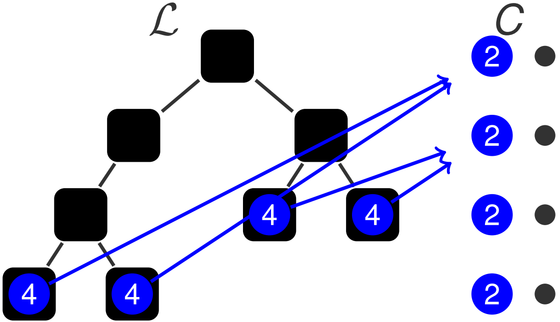
Case (ii): C is large (i.e. $2|C| >$ # leaves in $\mathcal{L}$)

Nodes in C give tokens to leaves, and follow the proof for the laminar set-pair family in undirected graphs

Token distribution



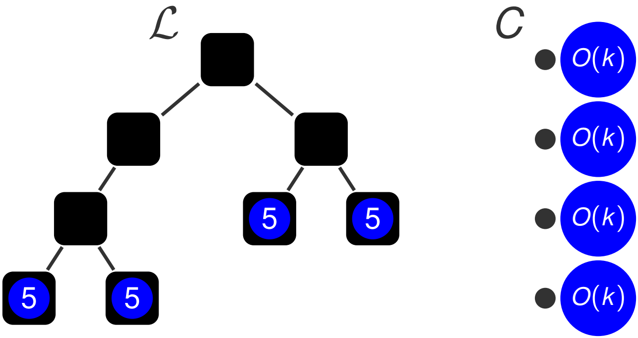
Token distribution



When $2|C| \leq$ # leaves of $\mathcal{L}$

- each leaf gives 2 tokens to a node in C
- nodes in C release their tokens

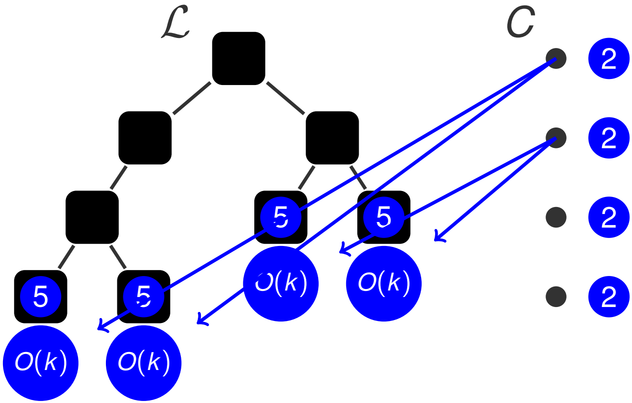
Token distribution



When $2|C| \geq$ # leaves of $\mathcal{L}$

- each node in C keeps 2 tokens
- nodes in C give the other tokens to leaves of $\mathcal{L}$

Token distribution



When $2|C| \geq$ # leaves of $\mathcal{L}$

- each node in C keeps 2 tokens
- nodes in C give the other tokens to leaves of $\mathcal{L}$

Conclusion

- Laminar, undirected ➜ $(O(k), O(k) \cdot b(v))$-approx
- Laminar, directed ➜ $(2, k, 2b^{+}(v) + O(k))$-approx
- Strongly laminar w/ degree-bounds, undirected ➜ $(4, 4b(v) + O(k))$-approx
- Laminar, undirected ➜ $(+\infty, 6b(v) + O(k^2))$-approx
- Strongly laminar w/ degree-bounds, undirected ➜ $(+\infty, 2b(v) + O(k^2))$-approx

Future works

- Narrow the gap between $O(k)$ and $\Omega(\sqrt{k})$ for uniform node-connectivity req. by iterative rounding
- Iterative rounding for other cases of node-connectivity