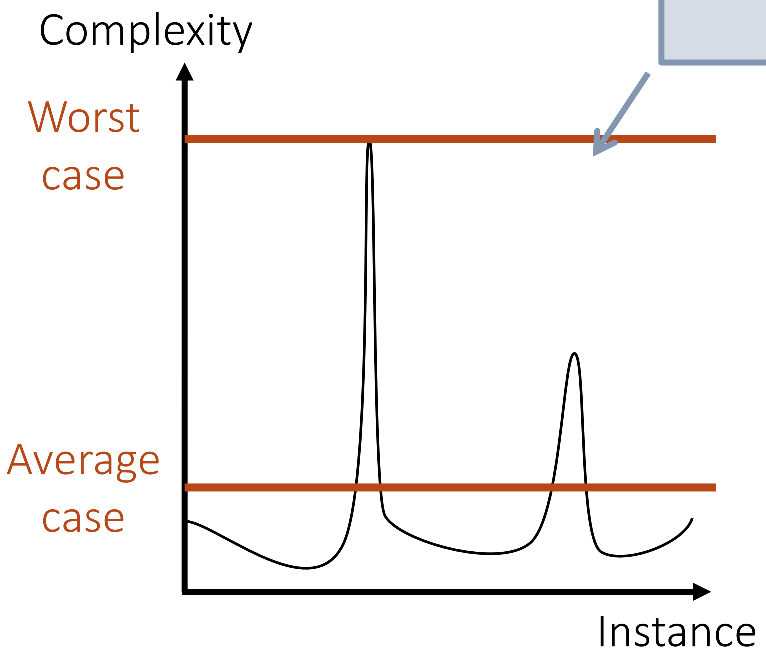
Smoothed Analysis of Dynamic Networks

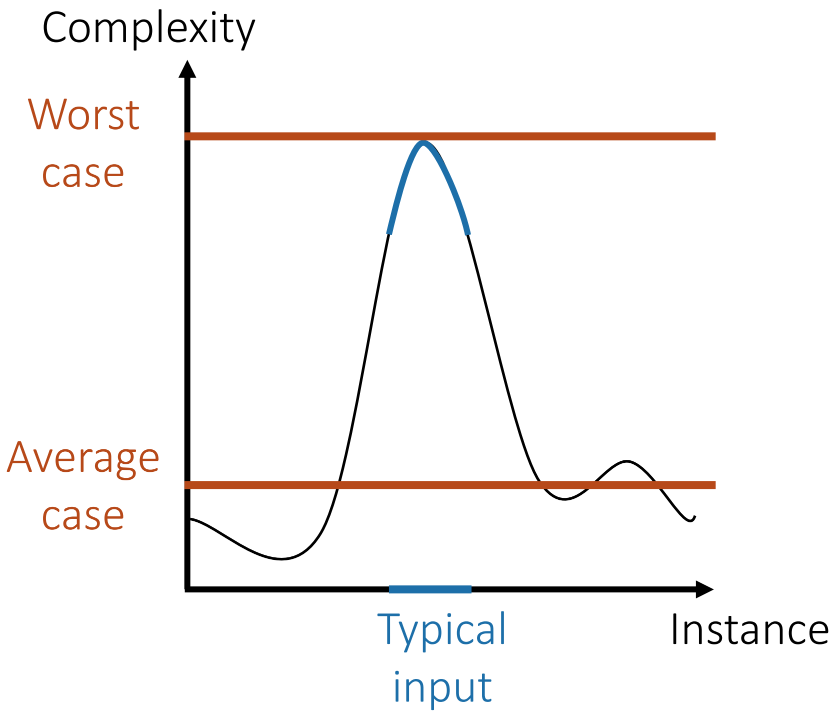
Ami Paz – CNRS

Seth Gilbert – National University of Singapore
Uri Meir – Tel-Aviv University
Gregory Schwartzman – Japan Advanced Institute of Science and Technology

Smoothed Analysis

Spielman and Teng '04:
The simplex algorithm behaves like this





Smoothed Analysis

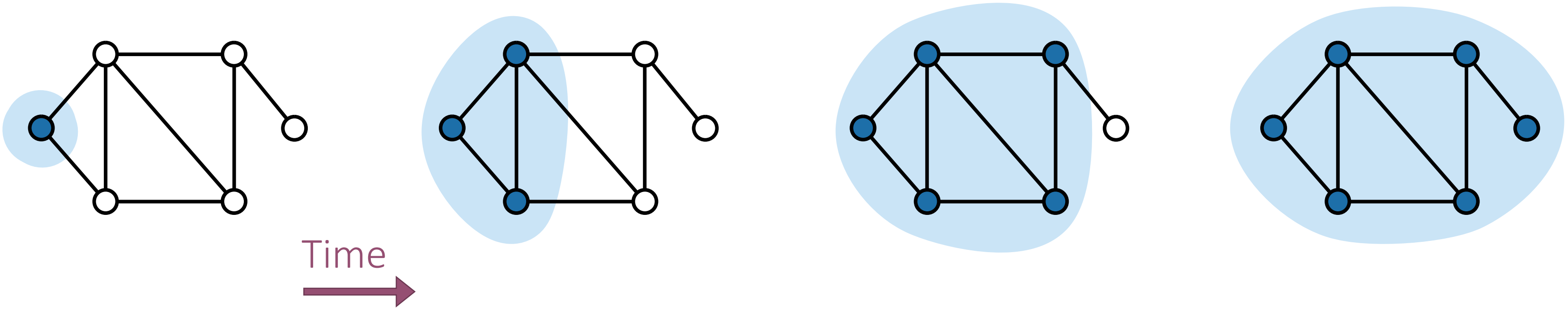
[Spielman and Teng '04]

- A smoothed linear program:
  A linear program + Gaussian noise

Main result
The simplex algorithm on a smoothed linear program takes polynomial time

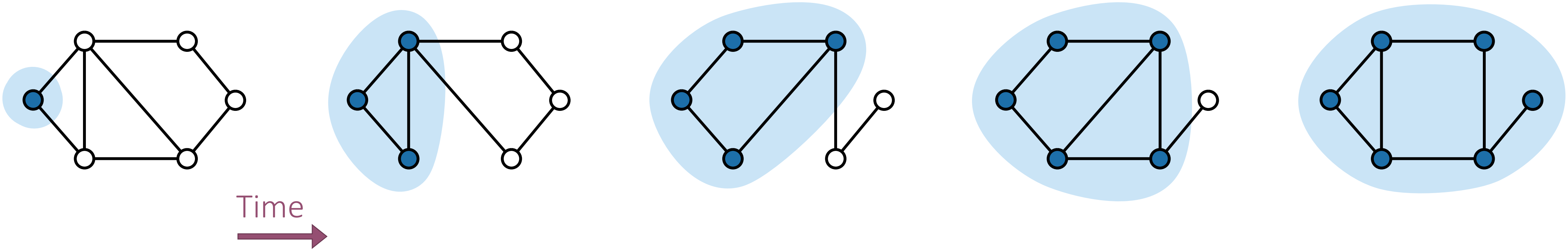
Flooding

- Connected n-node graph (n-unite synchronous network)
- Propagate information to all the network
- Worst-case: $\Theta(D)$ = diameter time

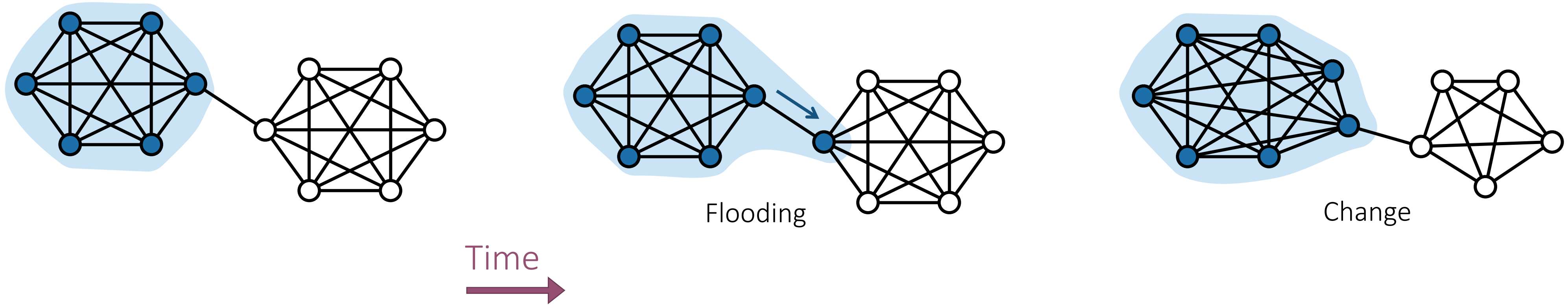


Dynamic Network

- Links change over time
- Worst case: $n - 1$ time
  - even with $D = 3$



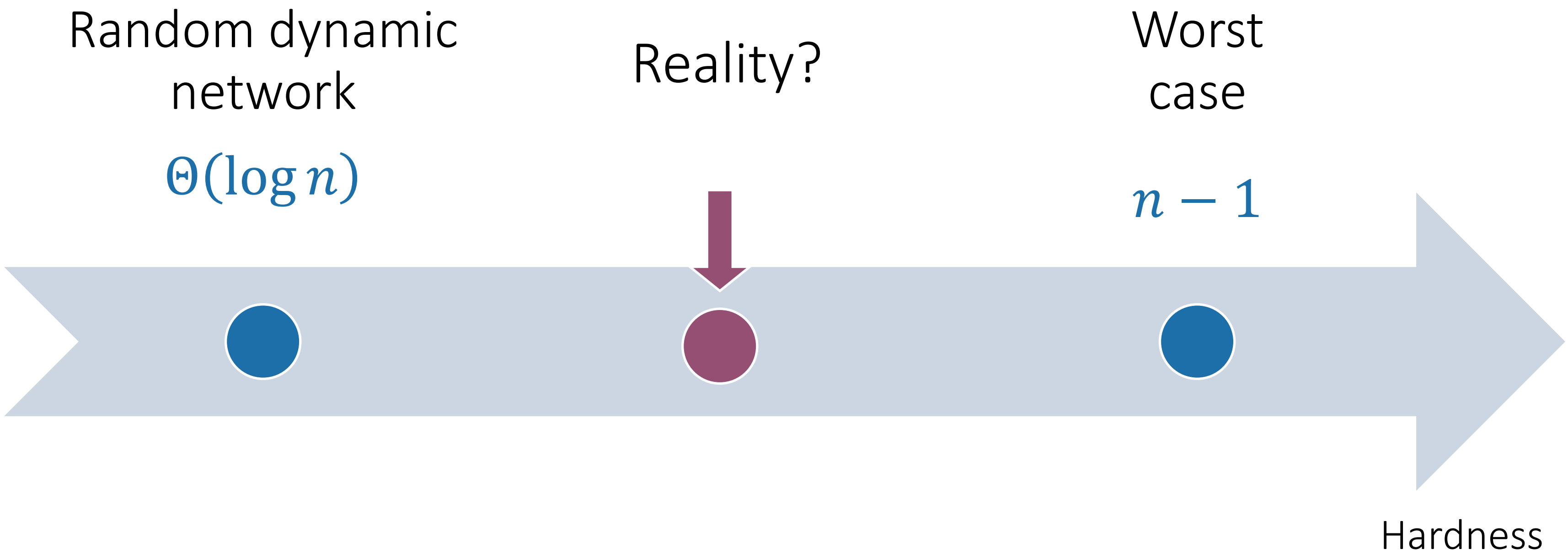
Worst-Case Analysis



Worst case: $n - 1$ time even with $D = 3$

Our goal: Go beyond the worst-case analysis

Flooding Time



Previous Work

- Pivoting rules for the simplex algorithm [Spielman and Teng '04]
- ...
- **Dynamic networks** [Dinitz, Fineman, Gilbert, Newport '18]
- MST in dynamic networks [Chatterjee, Pandurangan, Pham '20]
- Models of Smoothing in Dynamic Networks [Meir, Paz, Schwartzman '20]
- Load Balancing in Dynamic Networks [Gilbert, Meir, Paz, Schwartzman '21]

Smoothed Analysis

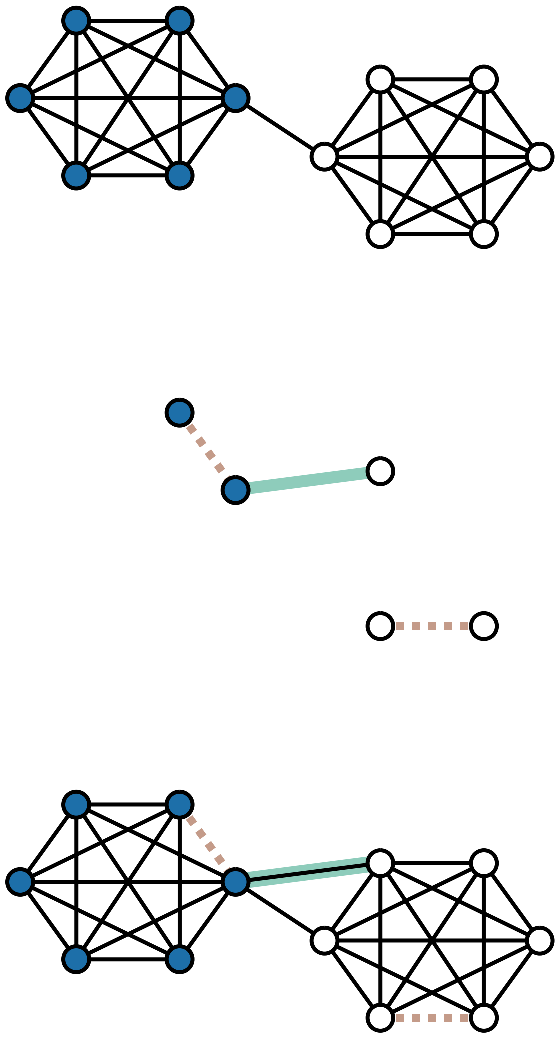
Adversarial input

+

k random edges

=

Smoothed input

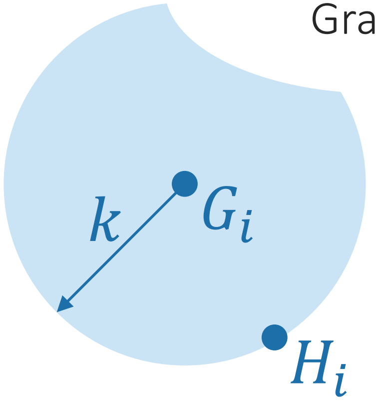


Integer Noise – Oblivious

[DFGN'18]

- Integer Noise: Pick a random graph with Hamming distance $\leq k$
- Adversary: $G_1, G_2, \ldots$
- Smoothed: $H_1, H_2, \ldots$
- $H_i \sim \mathrm{ball}(G_i, k)$
  - Note: Most graphs in $\mathrm{ball}(G_i, k)$ are at distance $\Omega(k)$ from G_i

Disconnected Graphs



Integer Noise – Oblivious

[DFGN’18]

Smoothed edges ≈ k edges

Adversary

Smoothed

Time

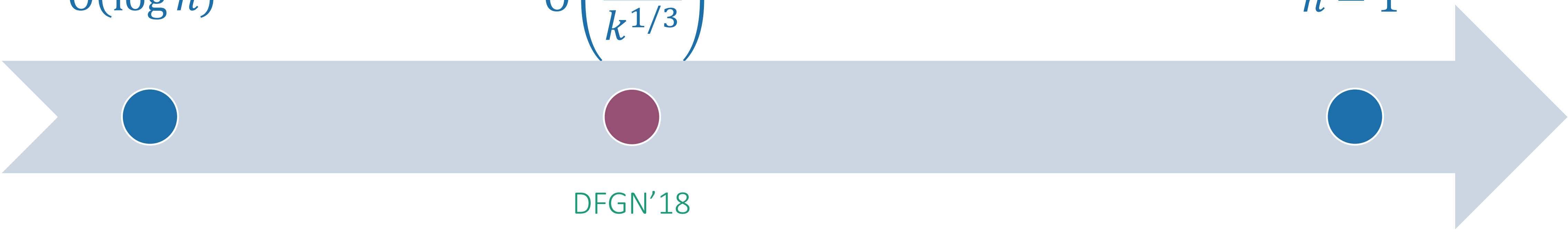
Integer Noise - Results

[DFGN'18]

- Flooding in $\tilde{\Theta}(n^{2/3}/k^{1/3})$ w.h.p.
- Polynomial gap between no noise ($k = 0$) and minimal noise ($k = 1$)
- Questions:
  1. Gap
  2. Adaptive adversary
  3. Responsive noise

Flooding – Some Results

Random Network	Oblivious Adversary	Worst Case
$\Theta(\log n)$	$\tilde{\Theta}\left(\frac{n^{2/3}}{k^{1/3}}\right)$	$n-1$



DFGN'18

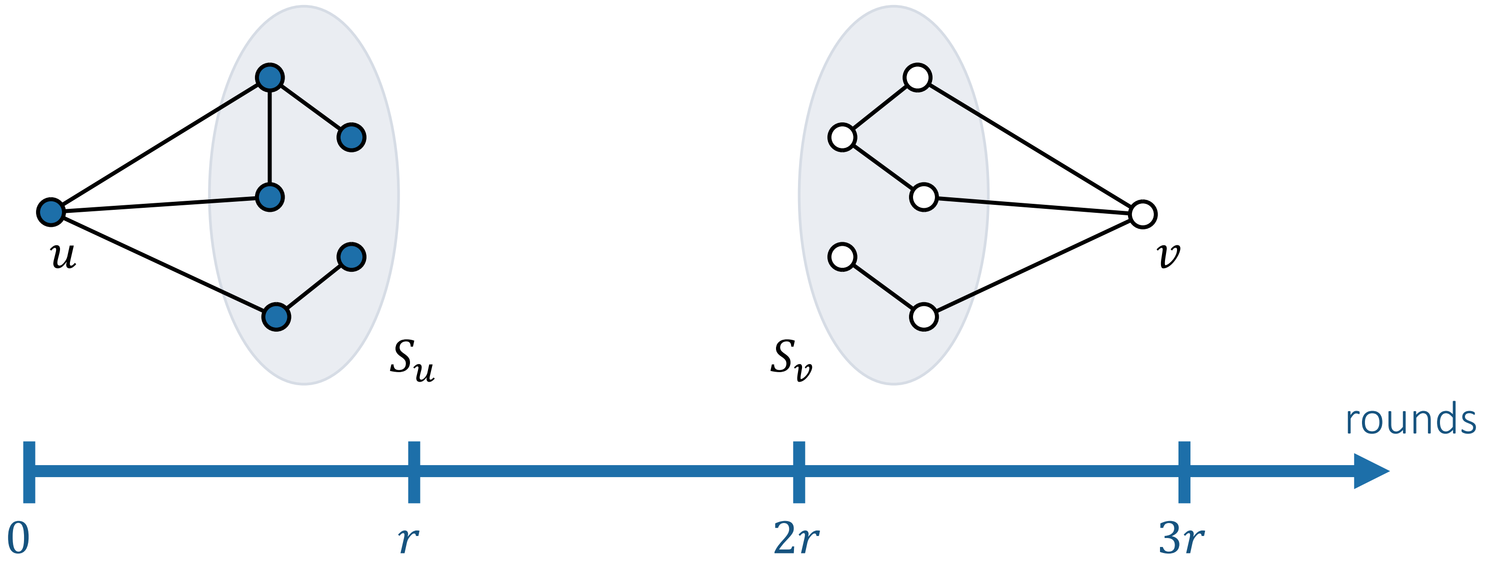
Hardness

Oblivious Adv. - Upper Bound

- Oblivious adversary, $\sim k$ random edges per round
- Fix a source u, arbitrary node v
- Choose $r = \tilde{\Theta}\left(n^{2/3}/k^{1/3}\right)$, analyze $3r$ rounds

Oblivious Adv. - Upper Bound

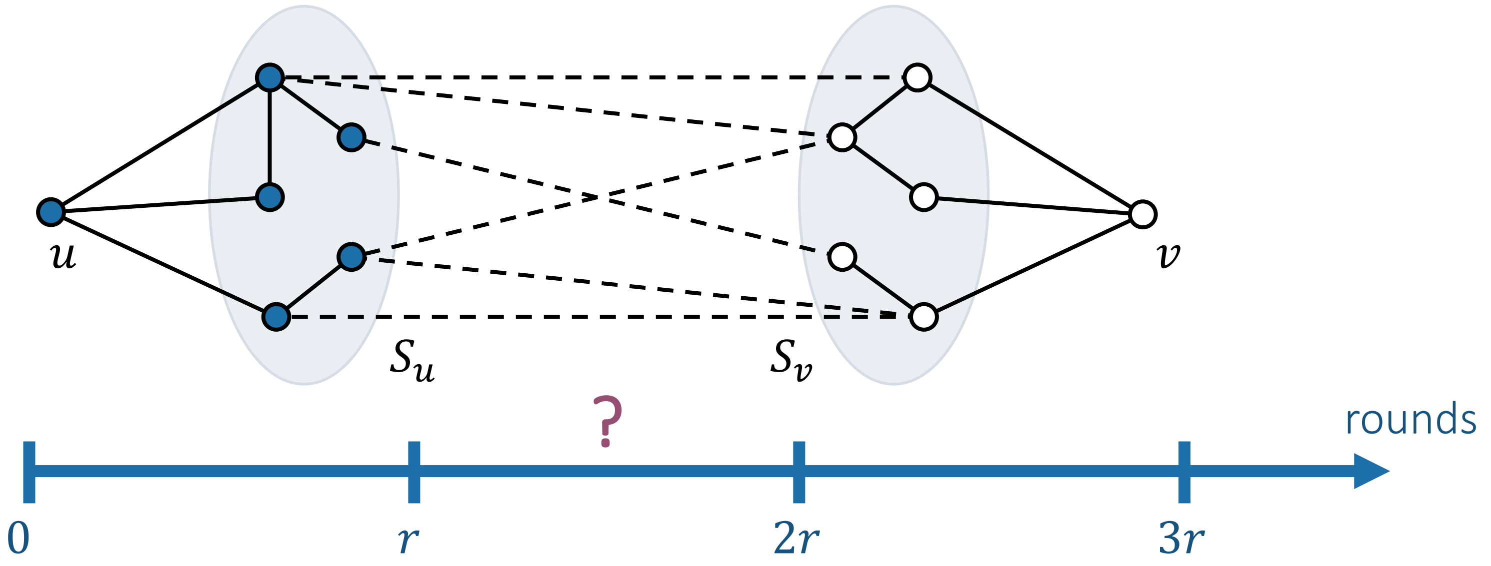
- Let S_u: nodes informed in rounds $1, \dots, r$
  - Each round: at least one new informed node, so $|S_u| \geq r$
- Let S_v: similarly, nodes that will inform v in rounds $2r + 1, \dots, 3r$
  - Again $|S_v| \geq r$
  - Depends on obliviousness



Oblivious Adv. - Upper Bound

$r = \widetilde{\Theta}(n^{2/3}/k^{1/3})$

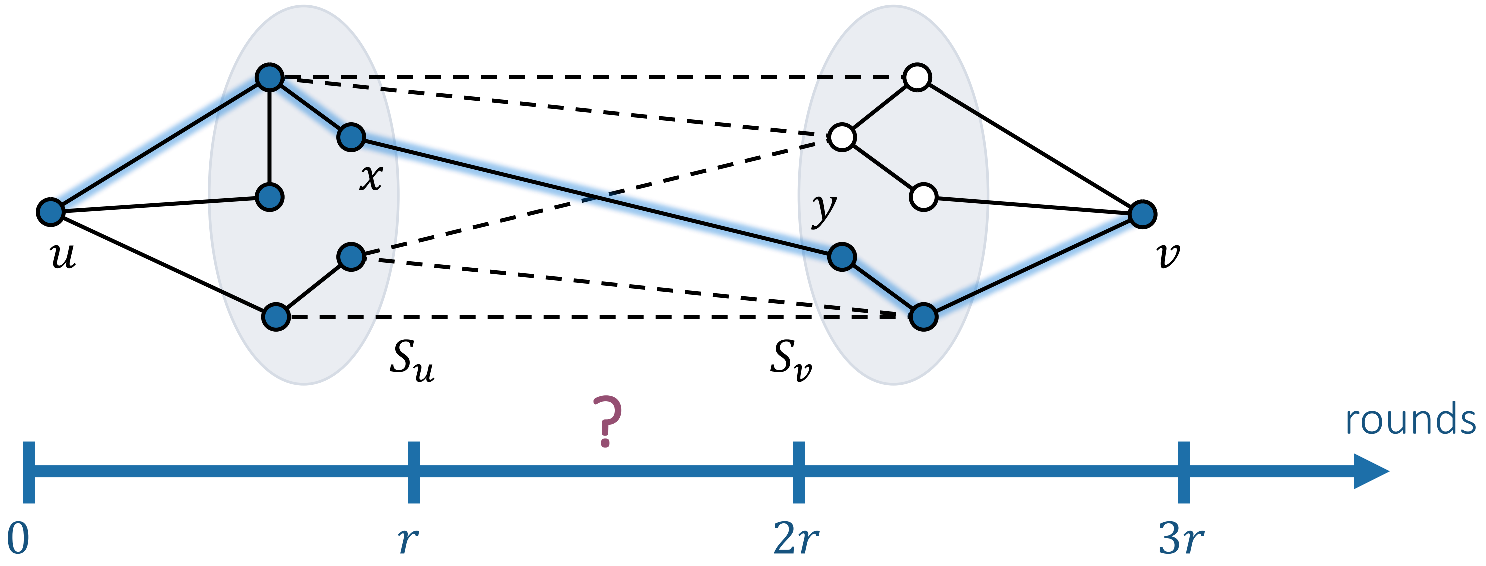
- Rounds $\mathrm{r} + 1, \ldots, 2r$?
- Single round: some edge from $S_u \times S_v$ appears w.p. kr^2/n^2 (lemma)
- r rounds: edge from $S_u \times S_v$ appears w.p. $1 - \left(1 - kr^2/n^2\right)^r \geq 1 - n^{-c}$
  - Also for fractional k



Oblivious Adv. - Upper Bound

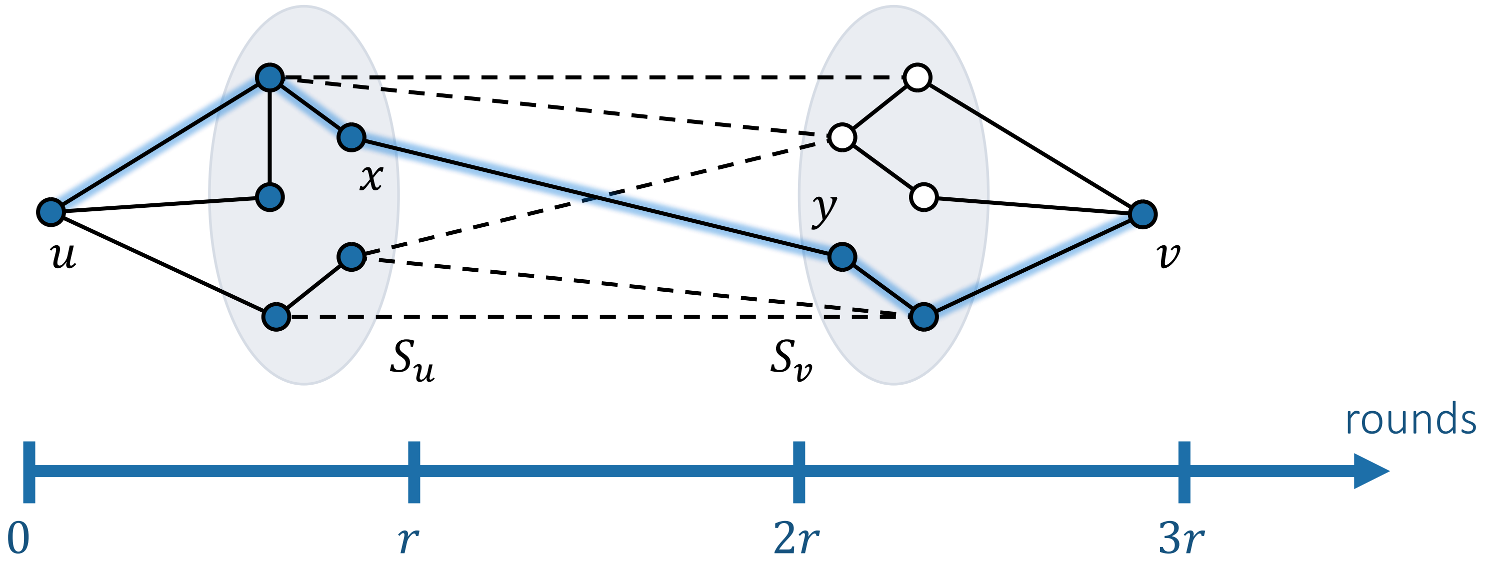
$r = \tilde{\Theta}(n^{2/3}/k^{1/3})$

- Rounds $\mathrm{r}+1, \dots, 2r$?
- Single round: some edge from $S_u \times S_v$ appears w.p. kr^2/n^2 (lemma)
- r rounds: edge from $S_u \times S_v$ appears w.p. $1-(1-kr^2/n^2)^r \geq 1-n^{-c}$
  - Also for fractional k

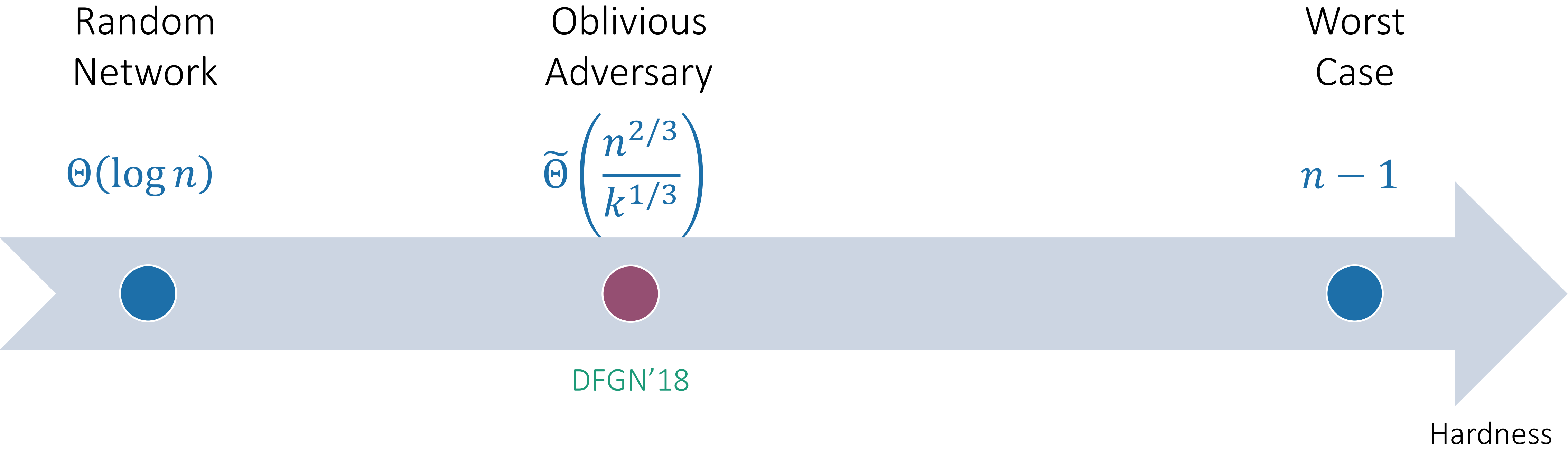


Oblivious Adv. - Upper Bound

- Flooding after $3r = \widetilde{\Theta}(n^{2/3}/k^{1/3})$ rounds w.h.p.
  - By a union bound over all nodes
- Note: highly depends on the obliviousness of the adversary
  - Otherwise S_v cannot be defined



Flooding – Some Results

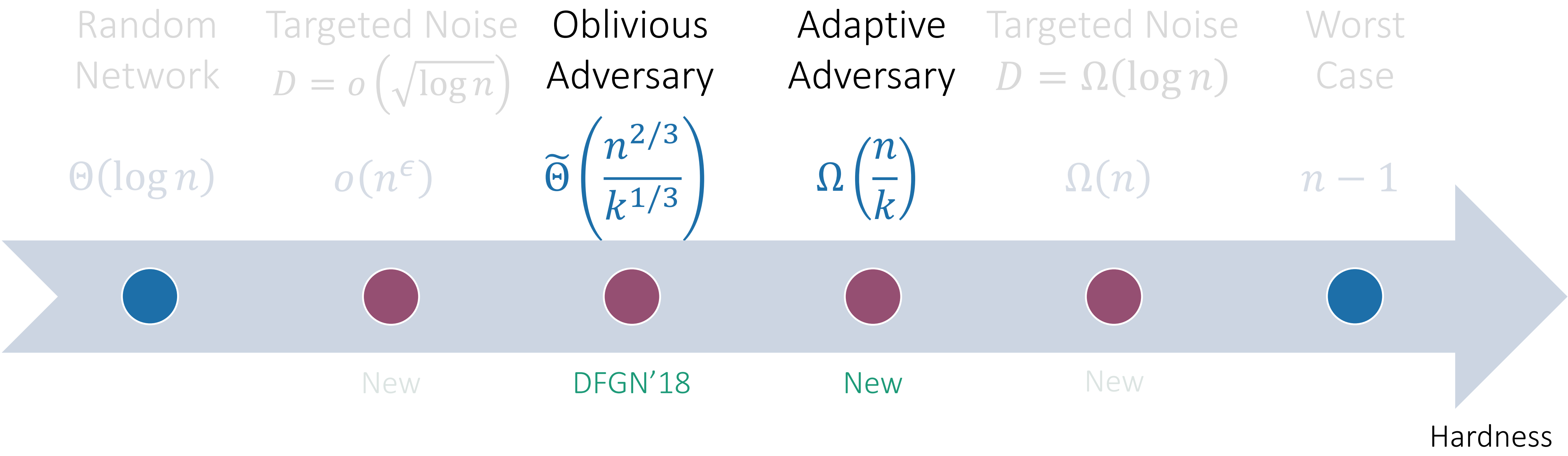


Flooding – Some Results

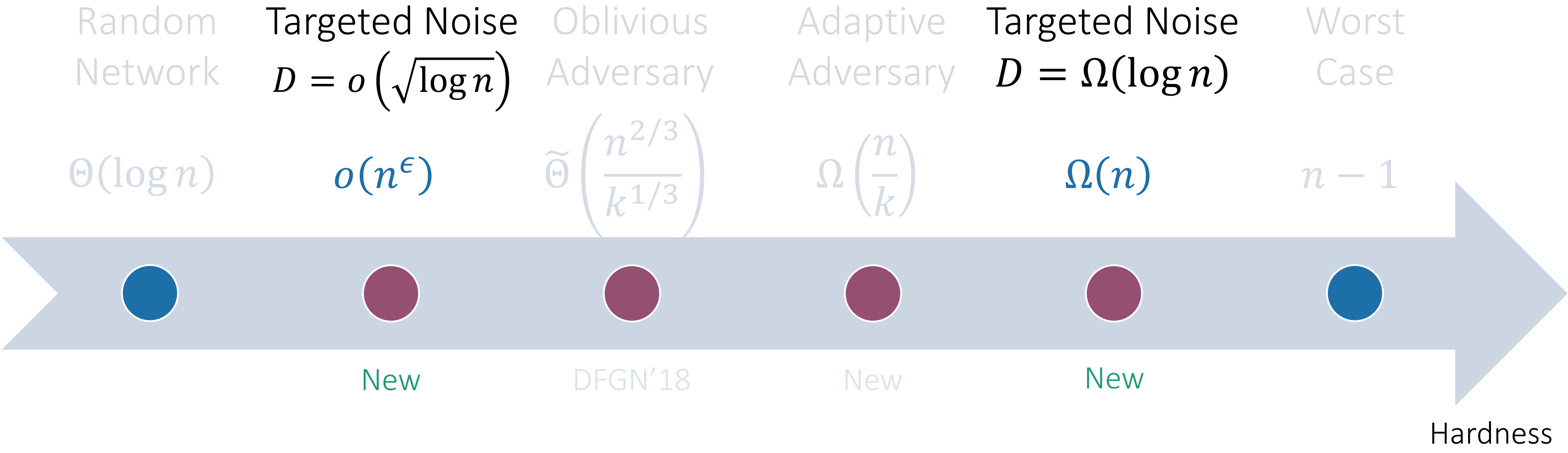
Random Network	Targeted Noise $D = o\left(\sqrt{\log n}\right)$	Oblivious Adversary	Adaptive Adversary	Targeted Noise $D = \Omega(\log n)$	Worst Case
$\Theta(\log n)$	$o(n^{\epsilon})$	$\tilde{\Theta}\left(\frac{n^{2/3}}{k^{1/3}}\right)$	$\Omega\left(\frac{n}{k}\right)$	$\Omega(n)$	$n-1$
	New	DFGN'18	New	New	

Hardness

Flooding – Some Results



Flooding – Some Results



Flooding – Some Results

Random Network	Targeted Noise $D = o\left(\sqrt{\log n}\right)$	Oblivious Adversary	Adaptive Adversary	Targeted Noise $D = \Omega(\log n)$	Worst Case
$\Theta(\log n)$	$o(n^{\epsilon})$	$\widetilde{\Theta}\left(\frac{n^{2/3}}{k^{1/3}}\right)$	$\Omega\left(\frac{n}{k}\right)$	$\Omega(n)$	$n-1$
	New	DFGN'18	New	New	

Hardness

Landscape of smoothing models and complexities

MPS'20

Adaptive Adv. - Upper Bound

[MPS'20]

- Adaptive adversary:
  - Picks a graph
  - $\sim k$ edges perturbed at random
  - Sees the perturbed edges

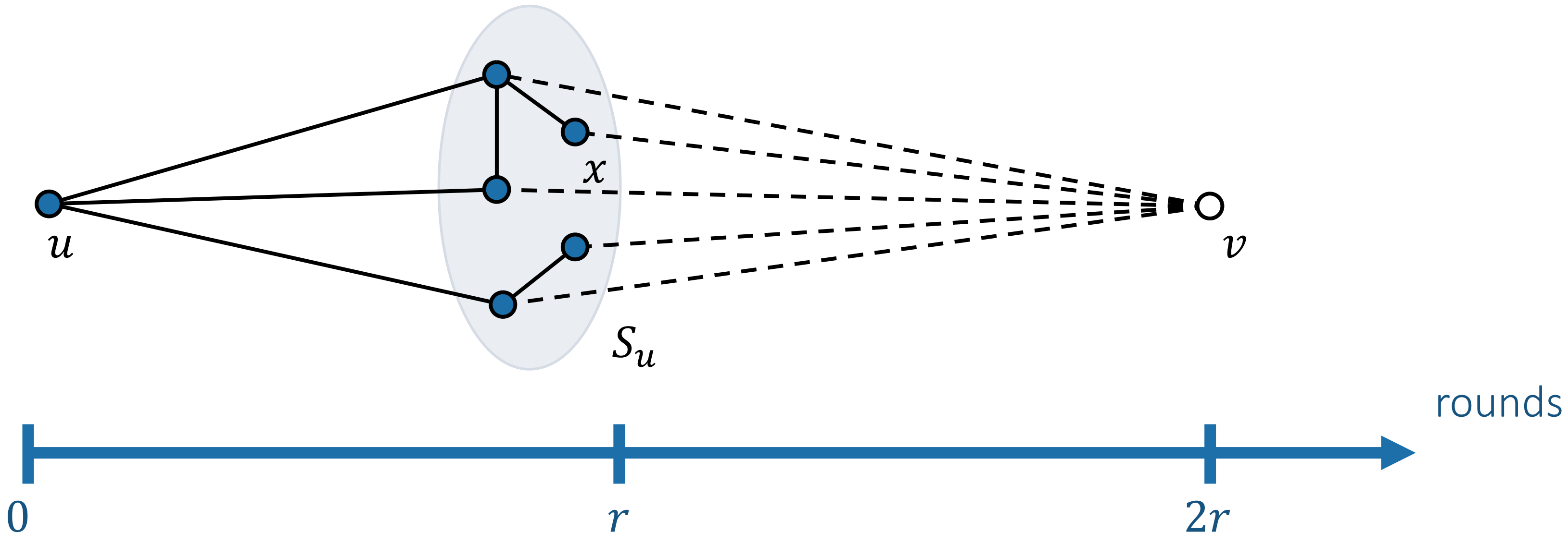
Adversary

Smoothed

Time

Adaptive Adv. - Upper Bound

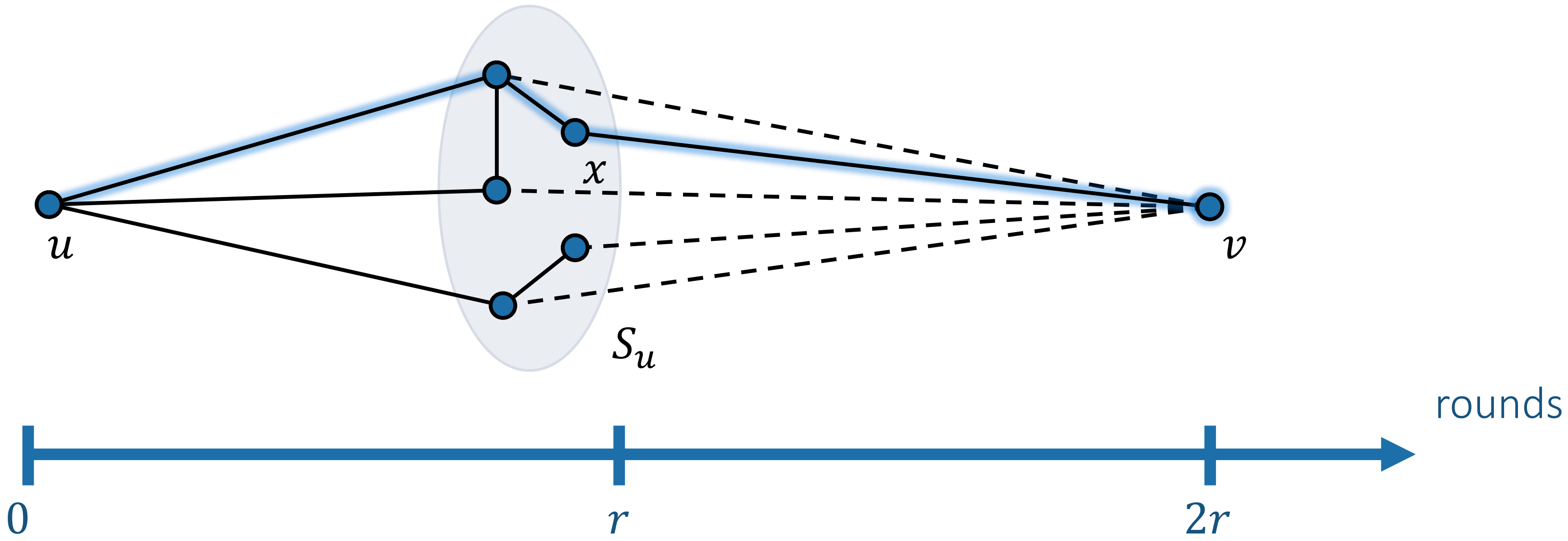
- Choose $r = \widetilde{\Theta}\left(n/\sqrt{k}\right)$, analyze $2r$ rounds
- Let S_u: nodes informed in rounds $1, \ldots, r$; $|S_u| \geq r$



Adaptive Adv. - Upper Bound

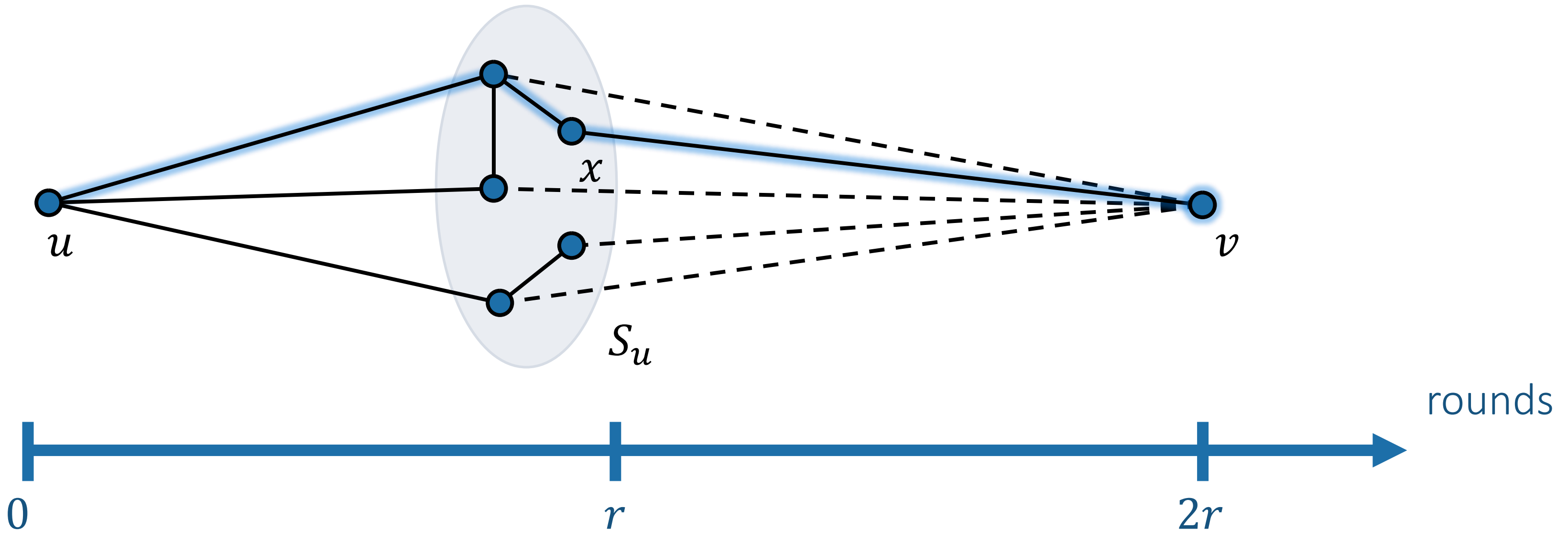
- Rounds $r+1,\dots,2r$,
- Single round: edge from S_u to v w.p. kr/n^2 (lemma)
- r rounds: edge from S_u to v w.p. $1-(1-kr/n^2)^r \geq 1-n^{-c}$

$r = \tilde{\Theta}\left(n/\sqrt{k}\right)$



Adaptive Adv. - Upper Bound

- Flooding after $2r = \tilde{\Theta}\left(n/\sqrt{k}\right)$ rounds w.h.p.
- Exists lower bound: $\tilde{\Omega}(n/k)$
  - Cannot improve the dependence on n



Targeted Noise

[MPS'20]

- Targeted Noise:
  - Adaptive/oblivious adversary
  - Each change happens w.p. $1 - \epsilon$

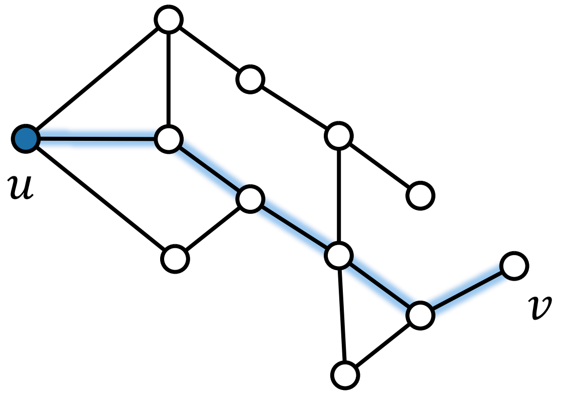
Adversary

Smoothed

Time

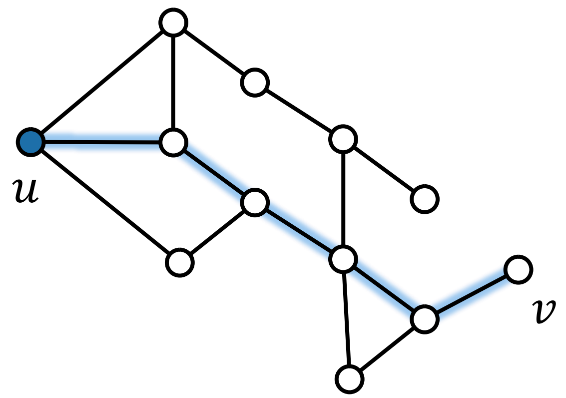
Targeted Noise - Upper Bound

- Small diameter
- Consider a shortest (u, v)-path P_{uv}
- $|P_{uv}| \leq D$



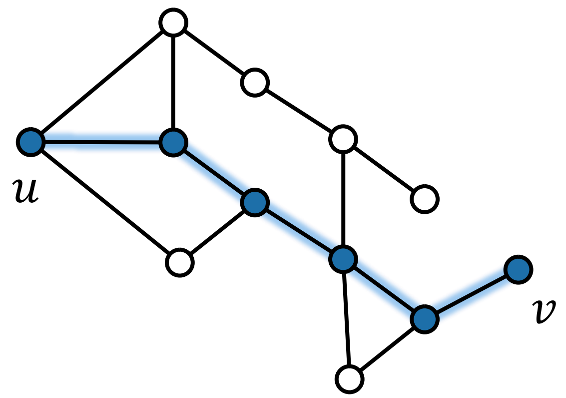
Targeted Noise - Upper Bound

- In D rounds
  - Each $e \in P_{uv}$ exists in all D rounds w.p. $\Omega(\epsilon^D)$
  - P_{uv} exists in all D rounds w.p. $\Omega\left(\epsilon^{D^2}\right)$
  - In which case v is informed

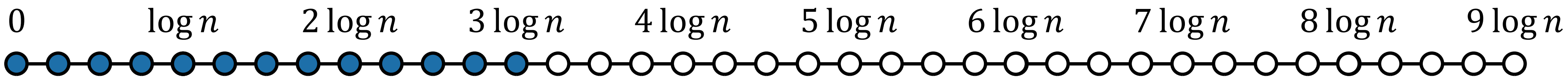


Targeted Noise - Upper Bound

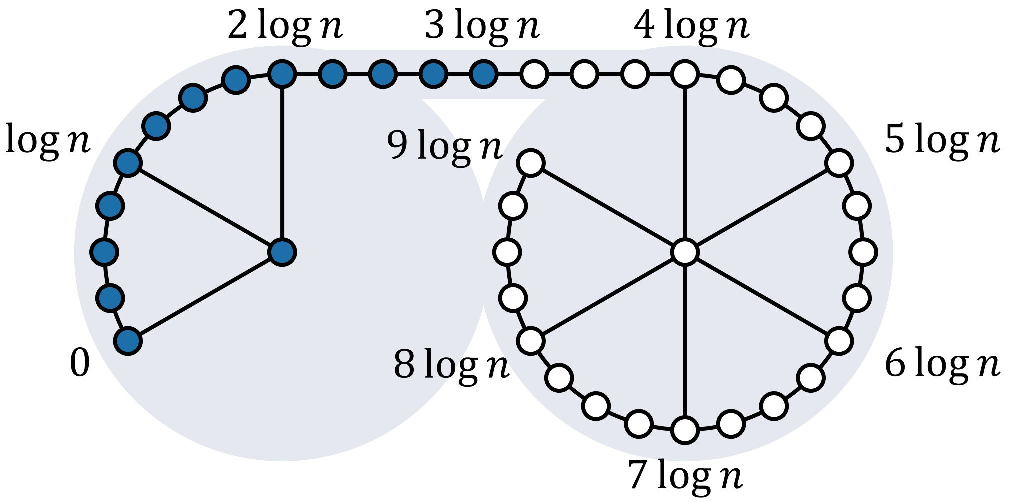
- After tD rounds
  - A node v is uninformed w.p. $O\left(\left(1-\epsilon^{D^2}\right)^t\right)$
- Set $t = \Theta\left(\epsilon^{-D^2}\log n\right)$
  - All nodes informed in tD rounds w.h.p.
- For $D = o\left(\sqrt{\log n}\right)$, $tD = o\left(n^{\delta}\right)$ for any constant δ



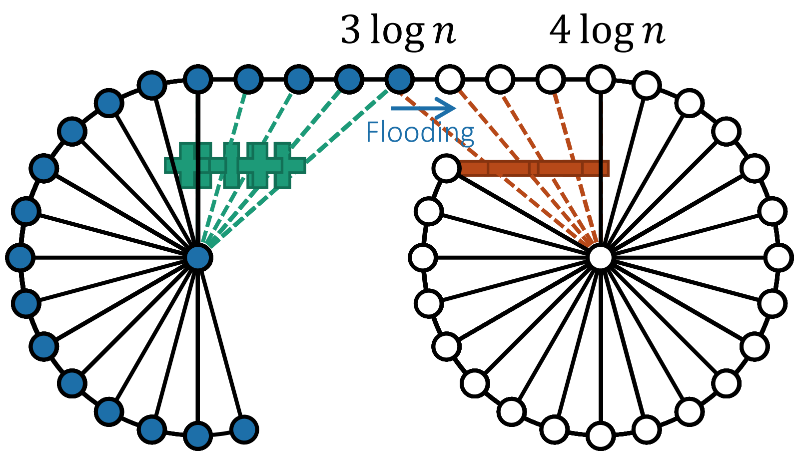
Targeted Noise - Lower Bound



Targeted Noise - Lower Bound



Targeted Noise - Lower Bound



Flooding takes $n-1$ rounds w.h.p.

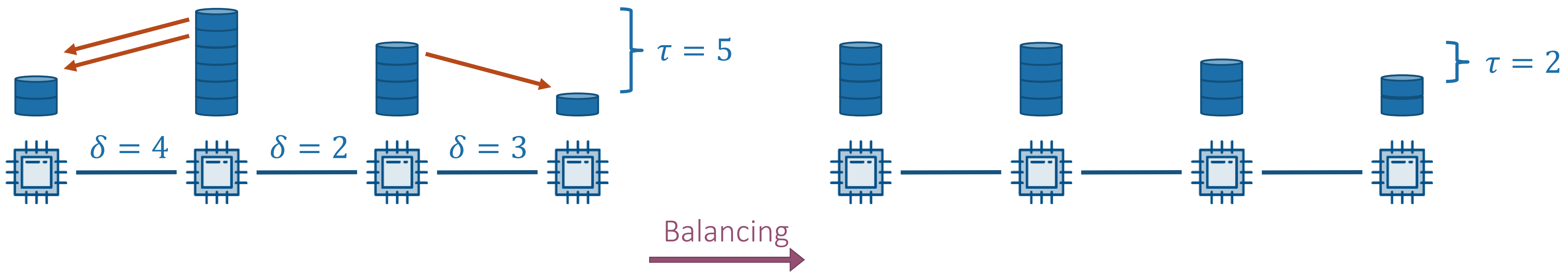
Bounds on Flooding Time

Model	Upper Bound	Lower Bound	Ref.
Non-Responsive Noise Oblivious Adversary	$\tilde{O}\left(n^{2/3}/k^{1/3}\right)$	$\Omega\left(\min\left\{n/k, n^{2/3}/k^{1/3}\right\}\right)$	[Dinitz et al. '18] + NEW
Non-Responsive Noise Adaptive Adversary	$\tilde{O}\left(n/k^{1/2}\right)$	$\tilde{\Omega}(n/k)$	NEW
Proportional Noise Oblivious Adversary	$\tilde{O}\left(n^{2/3}(D/\epsilon)^{1/3}\right)$		NEW
Proportional Noise Adaptive Adversary	$O(n)$	$\Omega(n)$	NEW
Targeted Noise	$O\left(D \log n / \epsilon^{D^2}\right)$	$\Omega(n)$ Even for $D \in \Theta(\log n)$	NEW

Responsive Noise (Proportional Noise, Targeted Noise)

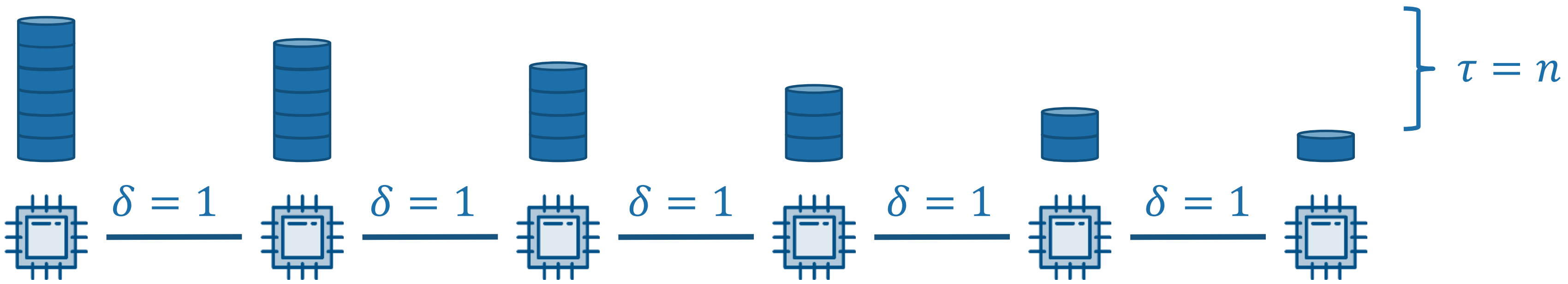
Distributed Load Balancing

[GMPS'21]



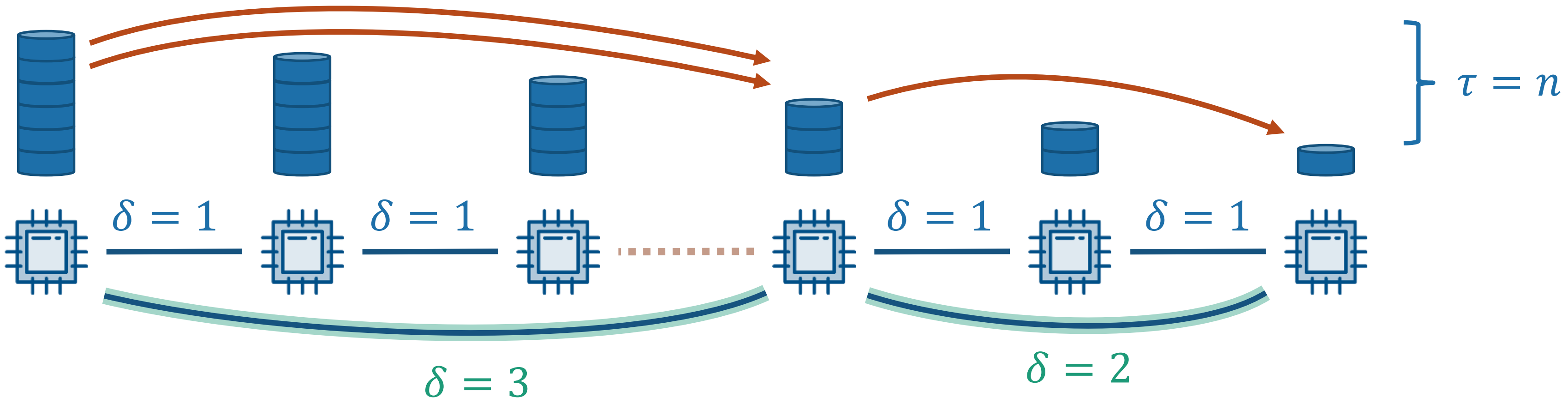
Local Load Balancing

- Dynamic networks: Getting constant τ is impossible!



Local Load Balancing

- Worst case: Getting constant τ is impossible!
- Smoothed dynamic networks: Balancing in $\tilde{O}\left(\frac{n^2}{k}\log\frac{1}{\tau}\right)$



Conclusion & Open Problems

Conclusion

- Many models of smoothing
- Have to choose a model by the concrete system

Open problems

- Beyond flooding and load balancing
- Application-driven models of smoothing

Thank you