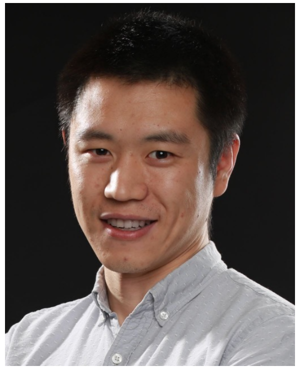
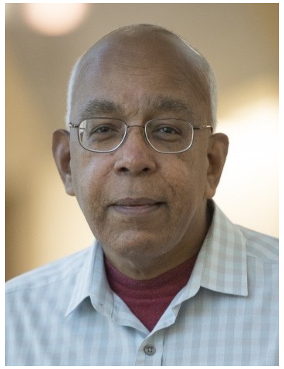
Delay-oriented Distributed Scheduling using Graph Neural Networks






Zhongyuan Zhao*, Gunjan Verma[†], *Ananthram Swami* [†], *Santiago Segarra**

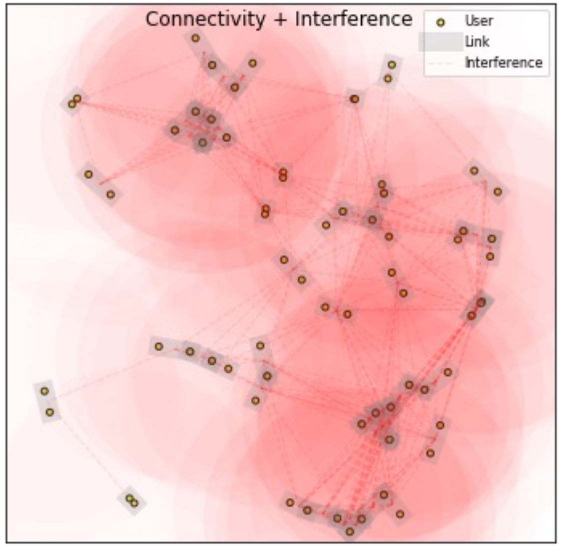
*Rice University, USA

† US Army's DEVCOM Army Research Laboratory, USA

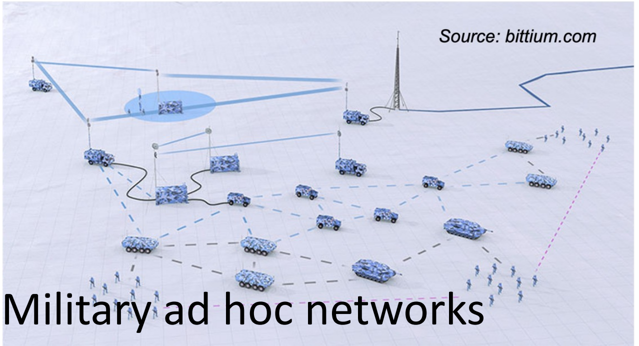
IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

Singapore, 22-27 May 2022

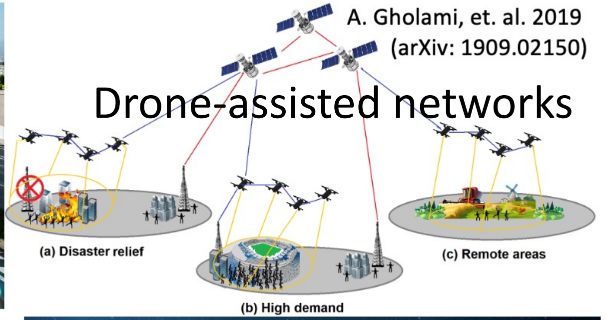
Wireless Multihop Networks

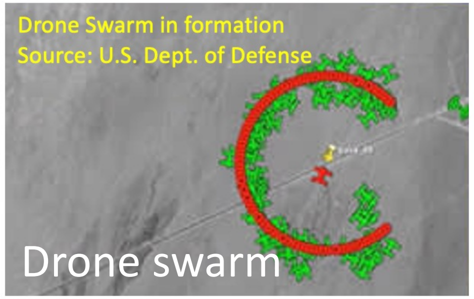


No base-stations!!



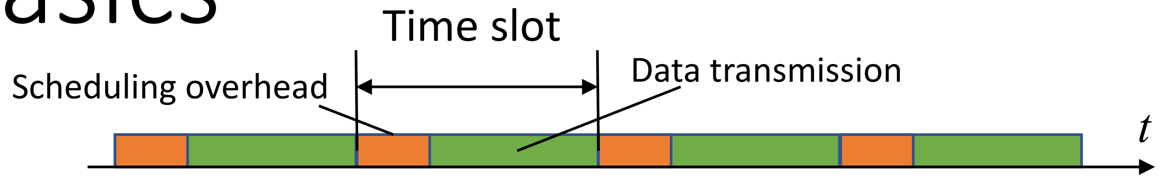
Internet of Things



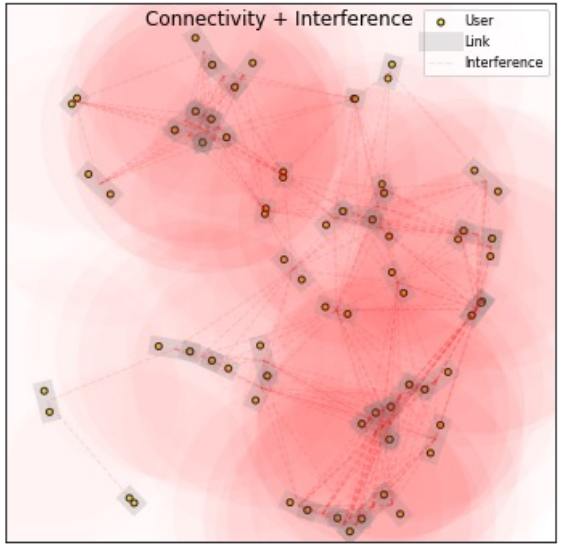


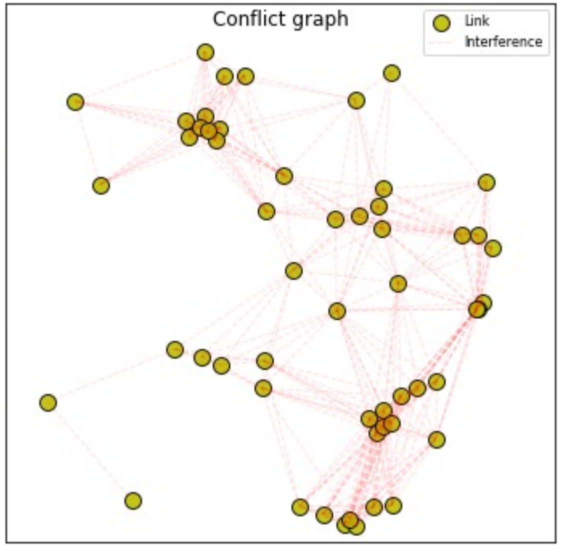
- Autonomous, self-organizing nodes
- Mobile Ad-Hoc Networks
- 5G and beyond
  - Wireless backhaul networks
  - Traffic offloading

Distributed Scheduling Basics



- Time-slotted network
- Conflict graph
  - Interface conflict
  - Potential interference
- Orthogonal access
  - Independent set on conflict graph
- Maximum weighted independent set (MWIS)
  - Node weight: per-link utility function
  - Independent MWIS problems across time slots*
- Optimal scheduler v.s. heuristics
  - Optimal: NP-hard (discrete optimization)
  - Local greedy scheduler (LGS) [Joo 2012]





Throughput-maximization

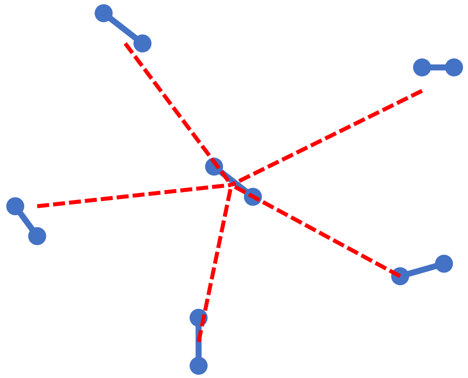
	Synchronized	Random Access
Infrastructure	Cellular, 5G (19.6%)	Wi-Fi (51%)
Ad-hoc	Wireless Ad-hoc networks (military, backhaul, mobile)	Wireless ad-hoc & sensor networks

Self-organizing MAC

* S. Basagni, "Finding a maximal weighted independent set in wireless networks," Telecomm. Systems, vol. 18, no. 1-3, pp. 155–168, 2001.

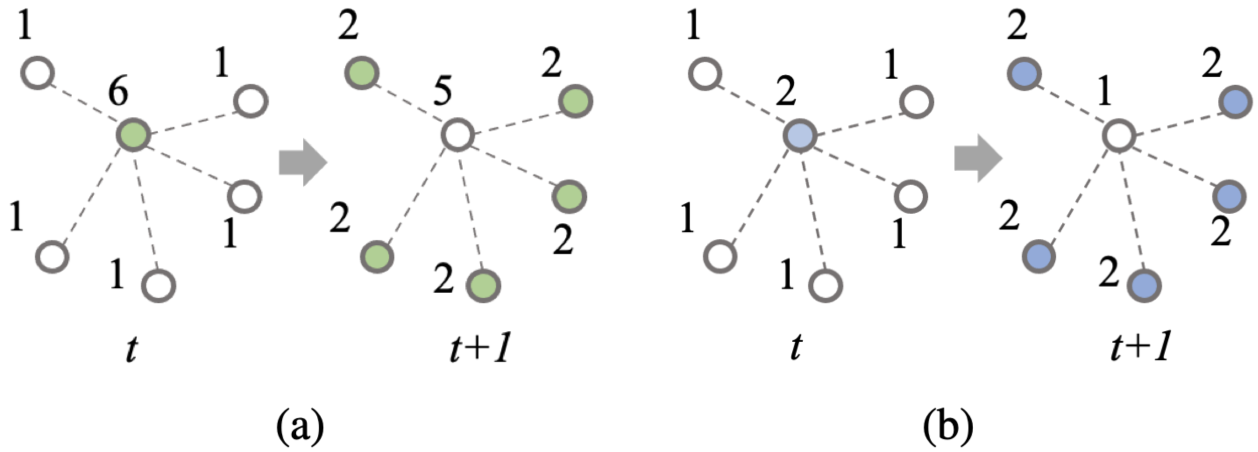
Source: Cisco VNI Global IP Traffic Forecast, 2017–2022

Latency: Optimal v.s. Greedy Schedulers



- Star conflict graph
- Per-link utility: Queue length
- Arrival rate: 1 packet/slot
- Link capacity: 2 packets/slot
- Initial state: all queues empty

Steady queueing state transitions



Optimal MWIS scheduler

Greedy MWIS scheduler

$$\bar{q}_{opt} = 2.17$$

$$\bar{q}_{grd} = 1.5$$

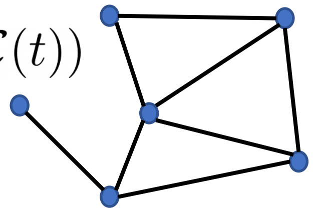
Queue-based per-link utility leads to poor delay performance

Problem formulation

1. MWIS is NP hard
2. Stochastic arrivals and link rates
3. Dependence between time slots

$$\mathcal{G}(t) = (\mathcal{V}(t), \mathcal{E}(t))$$

Conflict graph



Moser-Tardos Algorithm *
(CSMA, LGS[Joo 2012])

Scalability $\mathcal{O}(\log^* |\mathcal{V}|)$

$$\hat{\boldsymbol{v}}(t) = h(\mathcal{G}(t), \mathbf{u}(t))$$

Analytical utility functions
virtual queues of congestion [Xue 2012],
sojourn time [Hai 2018],
age-of-information [Hsu 2017]

Machine Learning generated utility
MLP outputs utility [Gupta 2020]
GCN outputs utility [This work]

Leverages Topology

MLP outputs decisions [Lee 2021]

Average queue length across time and network (links)

Problem 1. *For a time horizon of interest T, we want to solve for the delay-optimal scheduler given by*

$$c^* = \operatorname*{argmin}_{c \in \mathcal{C}} \mathbb{E}\left(\frac{1}{T+1}\sum_{t=0}^{T}\frac{\|\mathbf{q}(t)\|_1}{|\mathcal{V}(t)|}\right) \tag{1a}$$

Schedule Scheduler Network state

$$s.t. \quad \hat{\boldsymbol{v}}(t) = c(\boxed{\mathcal{G}(t), \mathbf{q}(t), \mathbf{r}(t)}), \tag{1b}$$

$$q_v(t+1) = \begin{cases} q_v(t) + a_v(t) & \text{if } v \notin \hat{\boldsymbol{v}}(t), \\ q_v(t) + a_v(t) - \min(r_v(t), q_v(t)) & \text{if } v \in \hat{\boldsymbol{v}}(t), \end{cases} \tag{1c}$$

Queueing state transition

where both constraints hold for every time $t = 0, \ldots, T$ and the second constraint holds for all $v \in \mathcal{V}$.

New arrivals Departures

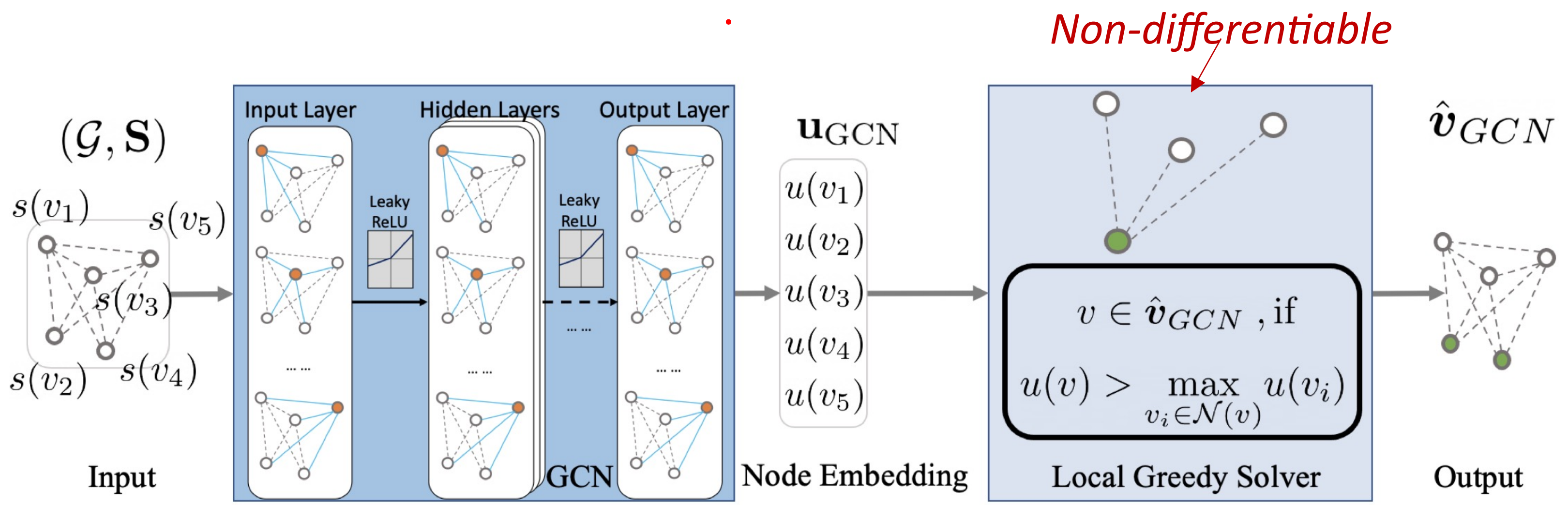
* Robin A Moser and Gábor Tardos. A constructive proof of the general lovász local lemma. *Journal of the ACM (JACM)*, 57(2):1–15, 2010.

Existing approaches to delay-oriented scheduling

Approach + papers	Distributed	Orthogonal Constraint	Scalability	Topology	Heuristic
Constrained optimization Set delay as constraint [Jaramillo 2011, Hou 2010]	No	Hard	Poor	Yes	N.A.
Delay-aware per-link utility functions virtual queues of congestion [Xue 2012], sojourn time [Hai 2018], age-of-information [Hsu 2017]	Yes	Hard	Good	No	Greedy
Machine Learning only MLP outputs binary decisions [Lee 2021]	Yes	Soft	Poor	Explicit	N.A.
Machine Learning + Heuristic MLP outputs utility [Gupta 2020] MLPs (Multiagent) output biases [Gao 2017]	No [Gupta 2020] Yes [Gao 2017]	Hard	Poor	Implicit	Greedy
[Ours] Graph Convolutional Neural Network + Distributed Greedy Heuristic	Yes	Hard	Good	Explicit	Greedy

Downstream Pipeline

[Zhao 2021]



Link features
Queue length,
Sojourn time,
Link rate,
Link type,
…

$$\mathbf{X}^l = \sigma\left(\mathbf{X}^{l-1}\boldsymbol{\Theta}_0^l + \boldsymbol{\mathcal{L}}\mathbf{X}^{l-1}\boldsymbol{\Theta}_1^l\right), l \in \{1, \ldots, L\},$$

$$L = 1, s(v) = q(v)r(v)$$

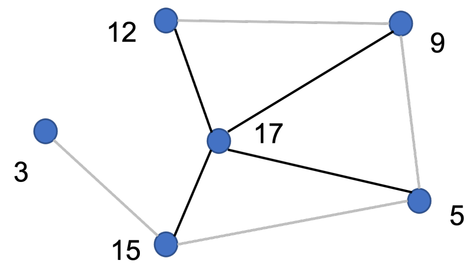
$$u(v) = s(v)\,\theta_0 + \left(s(v) - \sum_{v_i \in \mathcal{N}(v)} \frac{s(v_i)}{\sqrt{d(v)d(v_i)}}\right)\theta_1,$$

[Joo 2012]

$$\hat{\boldsymbol{v}}_{\mathrm{Gr}} \leftarrow \hat{\boldsymbol{v}}_{\mathrm{Gr}} \cup \left\{v \middle| u(v) > \max_{v_i \in \mathcal{N}(v)} u(v_i), \text{ for all } v \in \mathcal{V}'\right\},$$

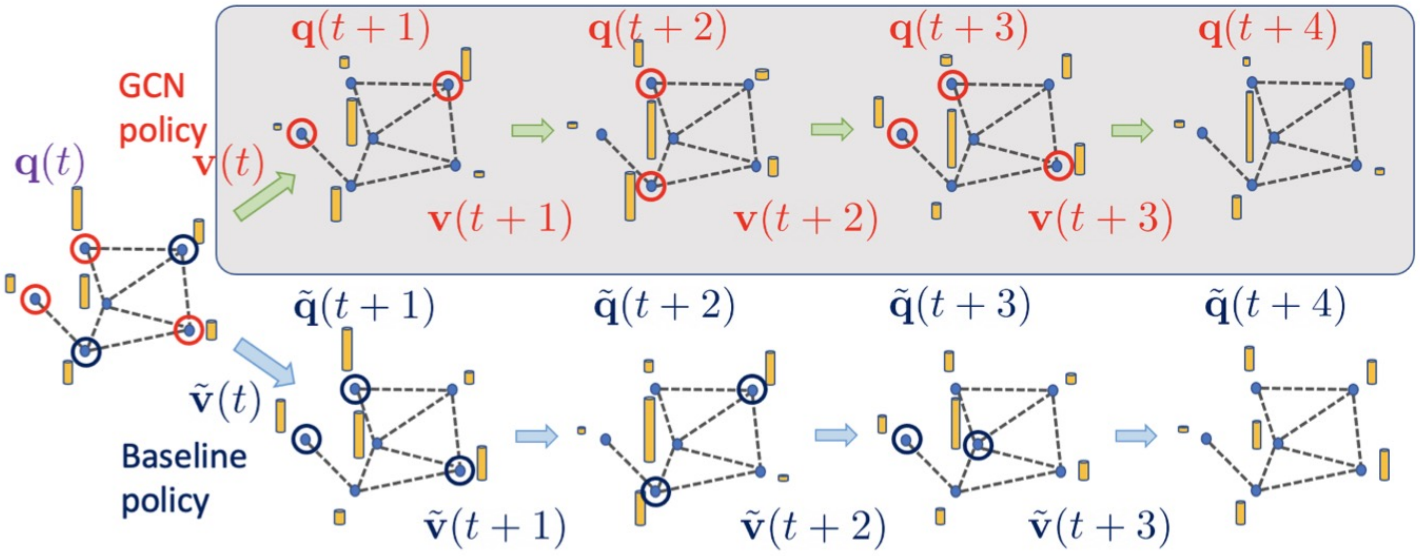
$$\mathcal{G}'(\mathcal{V}', \mathcal{E}') \leftarrow \mathcal{G}'(\mathcal{V}', \mathcal{E}') \setminus (\hat{\boldsymbol{v}}_{\mathrm{Gr}} \cup \mathcal{N}(\hat{\boldsymbol{v}}_{\mathrm{Gr}})),$$

$$\mathcal{O}(L + \log|\mathcal{V}|)$$



Reward by K-step lookahead scheduling

Virtual environment of scheduling



Schedule on each step is different

Stochastic estimation of return

$$\rho(v,t) = \begin{cases} \varphi\left(\frac{\sum_{k=1}^{K}\|\tilde{\mathbf{q}}(t+k)\|_1}{\sum_{k=1}^{K}\|\mathbf{q}(t+k)\|_1}\right), & v \in \hat{\boldsymbol{v}}_{\mathrm{GCN}}(t) \\ u_{\mathrm{GCN}}(v,t), & v \notin \hat{\boldsymbol{v}}_{\mathrm{GCN}}(t) \end{cases}$$

Customized reinforcement learning [Zhao 2021]

$$\ell(\boldsymbol{\omega}; \mathcal{G}(t), \mathbf{S}(t)) = |\mathcal{V}|^{-\frac{1}{2}} \|\mathbf{u}_{\mathrm{GCN}}(t) - \boldsymbol{\rho}(t)\|_2.$$

$$\varphi(x) = x$$

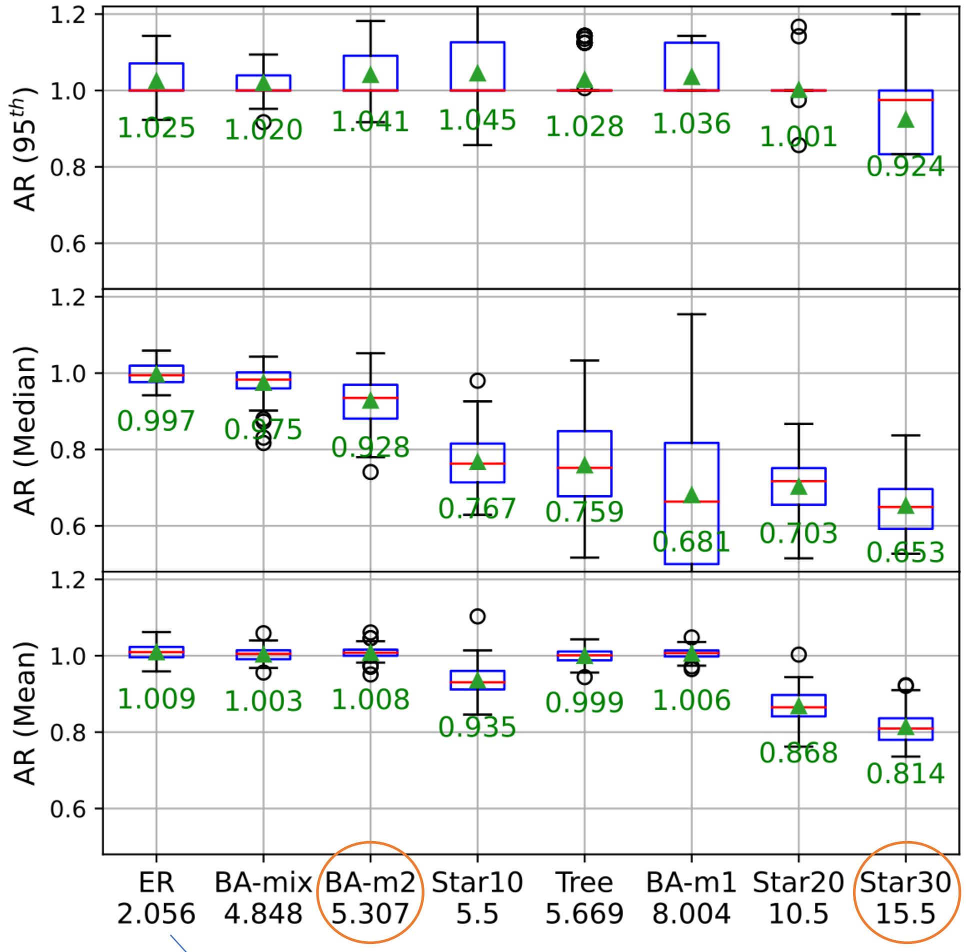
$$\varphi(x) = H(x-1)$$

Zhao, Z., Verma, G., Rao, C., Swami, A. and Segarra, S., 2021. Link scheduling using graph neural networks. *arXiv preprint arXiv:2109.05536*.

Queue lengths across topological distributions

- Scheduler
  - GCN $L = 1, s(v) = q(v)r(v)$
  - Baseline: local greedy solver [Joo 2012]
- Traffic
  - 1-hop link, Poisson arrival, clipped normal link rate
  - Light to moderate traffic load
- Training
  - Conflict graphs: 80% Star30 + 20% BA-m2
- Small-to-median sized networks (50, 70, …, 300 links)

Improved latency on concentrated conflict graphs



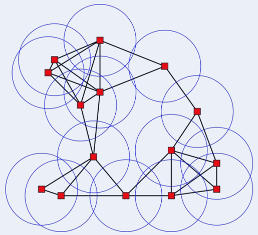
Peak to average node degree

Training graph models

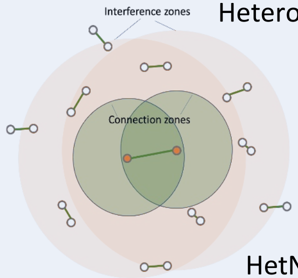
Underlying Network Topologies

Erdős–Rényi (ER)

Homogenous TX power

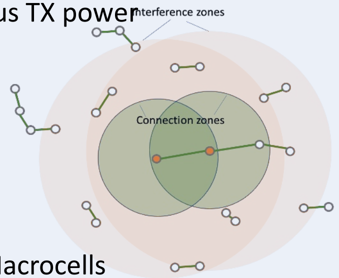


Star



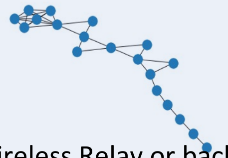
Heterogenous TX power

Barabási–Albert (BA)



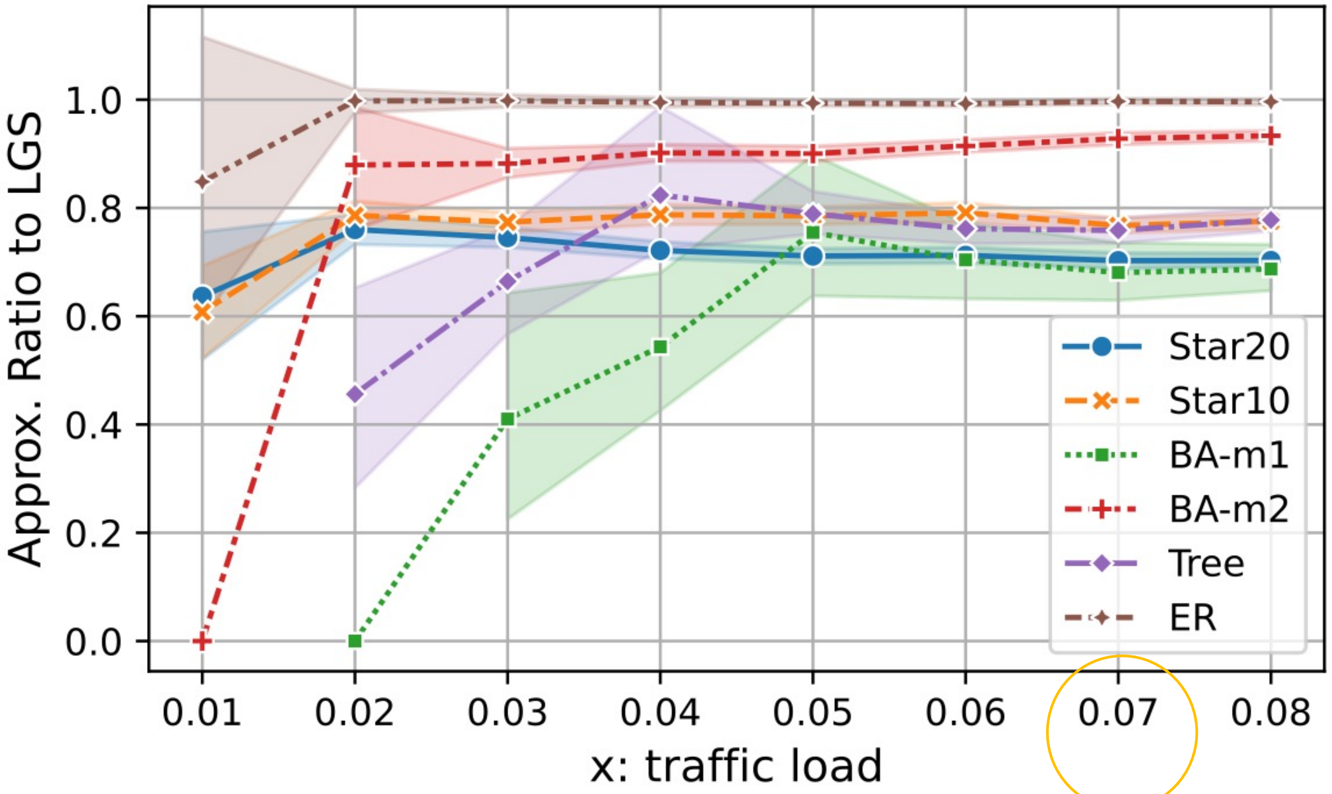
HetNet: Macrocells surrounded by microcells and D2D links

Power-law tree (Tree)



Wireless Relay or backhaul line with a few hot spots

Median backlogs across traffic loads



$$\mathbb{E}_{v \in \mathcal{V}, t \leq T}\left(\frac{a(v,t)}{r(v,t)}\right)$$

Consistent improvement across traffic loads

Conclusion & future work

- GCN-enhanced distributed scheduler
  - *Fully distributed solution*
  - *Hard constraint on orthogonal multiple access*
  - *Constant additional overhead*
  - Exploit topology
  - Generalize across graph distribution, sizes, and traffic loads.
  - Improves latency on networks with heterogenous transmit power
- Integration of delay-aware and topology-aware utility
- Principled learning approach for non-differentiable pipeline

References

- [Jaramillo 2011] J. J. Jaramillo, R. Srikant, and L. Ying, “Scheduling for optimal rate allocation in ad hoc networks with heterogeneous delay constraints,” IEEE J. Sel. Areas Commun., vol. 29, no. 5, pp. 979–987, 2011.
- [Hou 2010] I. H. Hou and P. R. Kumar, “Utility-optimal scheduling in time-varying wireless networks with delay constraints,” in Proc. of the 11th ACM intl. symposium on Mobile ad hoc netw. and comp. (MobiHoc), pp. 31–40, 2010.
- [Xue 2012] D. Xue and E. Ekici, “Delay-guaranteed cross-layer scheduling in multihop wireless networks,” IEEE/ACM Trans. Netw., vol. 21, no. 6, pp. 1696–1707, 2012.
- [Hsu 2017] Y.-P. Hsu, E. Modiano, and L. Duan, “Age of information: Design and analysis of optimal scheduling algorithms,” in IEEE Intl. Symposium on Info. Theory (ISIT), pp. 561–565, 2017.
- [Hai 2018] L. Hai, Q. Gao, J. Wang, H. Zhuang, and P. Wang, “Delay-optimal back-pressure routing algorithm for multihop wireless networks,” IEEE Trans. Vehicular Tech., vol. 67, no. 3, pp. 2617–2630, 2018.
- [Lee 2021] M. Lee, G. Yu, and G. Y. Li, “Graph embedding-based wireless link scheduling with few training samples,” IEEE Trans. Wireless Commun., vol. 20, no. 4, pp. 2282–2294, 2021.
- [Gupta 2020] M. Gupta, A. Rao, E. Visotsky, A. Ghosh, and J. G. Andrews, “Learning link schedules in self-backhauled millimeter wave cellular networks,” IEEE Trans. Wireless Commun., vol. 19, no. 12, pp. 8024–8038, 2020.
- [Gao 2017] J. Gao, Y. Shen, M. Ito, and N. Shiratori, “Multi-agent Q-learning aided backpressure routing algorithm for delay reduction,” arXiv preprint arXiv:1708.06926, 2017.
- [Joo 2012] C. Joo and N. B. Shroff, “Local greedy approximation for scheduling in multihop wireless networks,” IEEE Trans. on Mobile Computing, vol. 11, no. 3, pp. 414–426, 2012.
- [Zhao 2021] Z. Zhao, G. Verma, C. Rao, A. Swami, and S. Segarra, “Distributed scheduling using graph neural networks,” in IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP), pp. 4720–4724, 2021.


Q&A