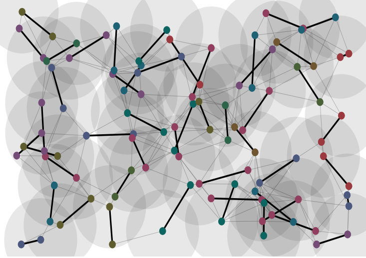
Graph Signal Diffusion Models for Wireless Resource Allocation

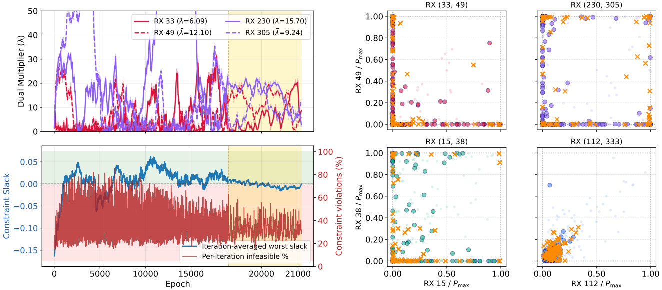
Yiğit Berkay Uslu Samar Hadou Shirin Saeedi Bidokhti Alejandro Ribeiro

Department of Electrical and Systems Engineering, University of Pennsylvania

June 10, 2026

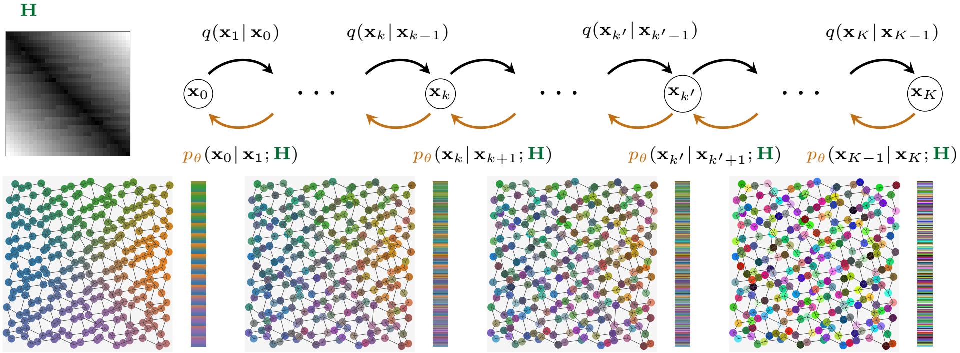
- Optimally allocate network resources to maximize an ergodic utility, subject to ergodic QoS requirements.





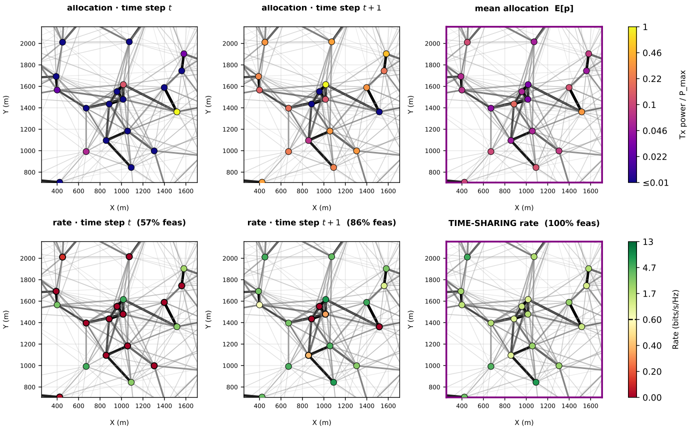
- Optimal (expert) allocations are stochastic (graph) signals. ⇒ We learn to sample them.

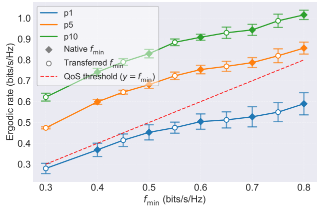
- Train a conditional diffusion model to imitate a primal-dual expert policy over a family of network graphs.

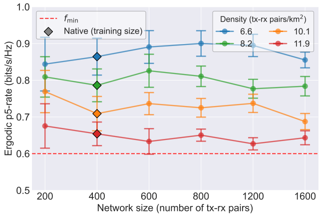


- A U-GNN denoiser generates allocations, conditioned on channel + node states.

- Learned policy-switching achieves near-optimality and ergodic feasibility.







- Generalizes to unseen QoS targets and transfers to larger networks (graphs).

Imitation Learning of Expert Policies

- Find an optimal (stochastic) allocation of network resources for a given network state $\mathbf{H}$:

$$\begin{aligned} \mathrm{P}^*(\mathbf{H}) \;=\; \underset{\mathcal{D}_{\mathbf{x}}(\mathbf{H})}{\text{maximum}} \;\; & \mathbb{E}_{\mathcal{D}_{\mathbf{x}}(\mathbf{H})}\left[f_0\left(\mathbf{x}(\mathbf{H}), \mathbf{H}\right)\right], \\ \text{subject to} \;\; & \mathbb{E}_{\mathcal{D}_{\mathbf{x}}(\mathbf{H})}\left[\mathbf{f}\left(\mathbf{x}(\mathbf{H}), \mathbf{H}\right)\right] \geq \mathbf{0}. \end{aligned}$$

$\Rightarrow \mathcal{D}^*_{\mathbf{x}}(\mathbf{H})$ maximizes an expected utility while satisfying expected requirements.

- Fit a parametric policy to the expert over a channel state distribution:

$$\mathcal{D}^*_{\mathbf{x}}(\mathbf{H};\theta) = \underset{\mathcal{D}_{\mathbf{x}}(\mathbf{H};\theta)}{\text{argmin}} \ \mathbb{E}_{\mathcal{D}_{\mathbf{H}}}\Big[\mathrm{D}_{\mathrm{KL}}\big(\mathcal{D}^*_{\mathbf{x}}(\mathbf{H}) \,\big\|\, \mathcal{D}_{\mathbf{x}}(\mathbf{H};\theta)\big) \Big].$$

- Expert conditionals lack a closed-form, but we can draw samples from them.

- Train a conditional diffusion model policy to generate samples $\mathbf{x}\,\big|\,\mathbf{H}$ drawn from dual descent roll-outs.

A Primal–Dual Expert Algorithm

- Take the deterministic per-network problem and form its Lagrangian:

$$\widetilde{\mathcal{L}}(\mathbf{x}, \lambda) = f_0(\mathbf{x}, \mathbf{H}) + \lambda^\top \mathbf{f}(\mathbf{x}, \mathbf{H})$$

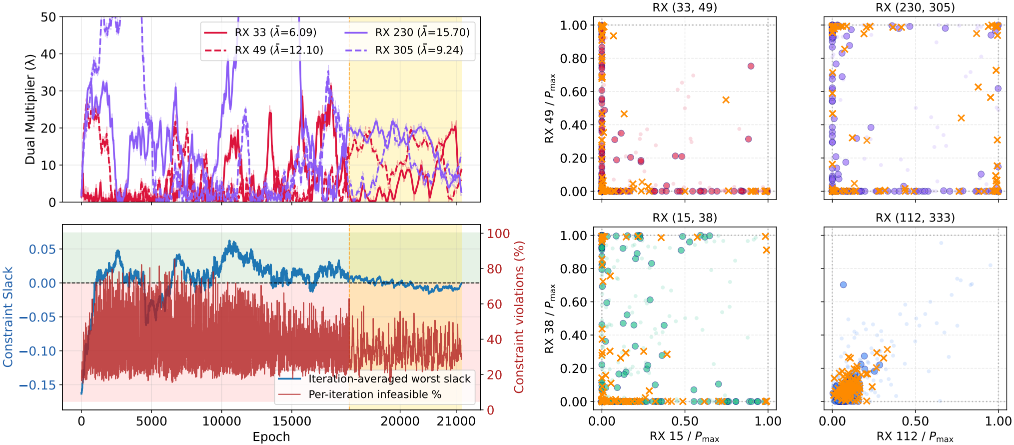
- Run projected dual (sub)gradient descent iterates:

$$\mathbf{x}_k := \mathbf{x}^\dagger(\lambda_k) \in \operatorname*{argmax}_{\mathbf{x} \in \mathcal{X}} \widetilde{\mathcal{L}}(\mathbf{x}, \lambda_k), \qquad \lambda_{k+1} = \Big[\lambda_k - \eta_\lambda \mathbf{f}\big(\mathbf{x}_k\big)\Big]_+ .$$

- Expert dataset = a late-iteration window of the primal–dual trajectory.

Primal–Dual Expert Policies

- Iterates are pointwise infeasible, yet the randomized policy is near-optimal and average-feasible.[1]



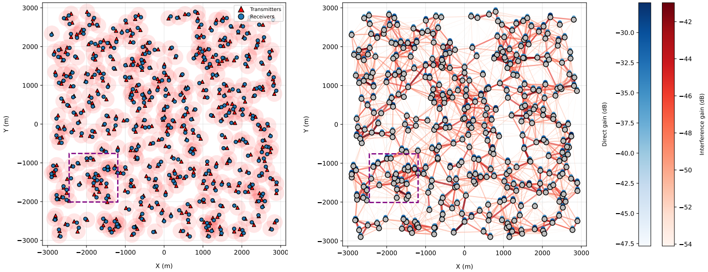
- A long-transient precedes convergence. $\Rightarrow$ Generative solutions bypass it.

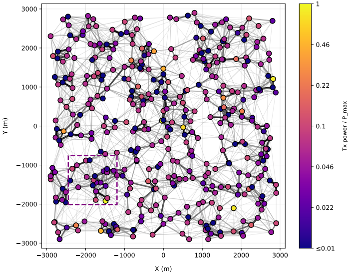
[1] Ribeiro, *Ergodic Stochastic Optimization Algorithms for Wireless Communication and Networking*, IEEE TSP, 2006.

Graph Signal Diffusion Models

- Channel state information (CSI) defines a graph-structure.

  ⇒ Users are nodes; allocations are signals on top.

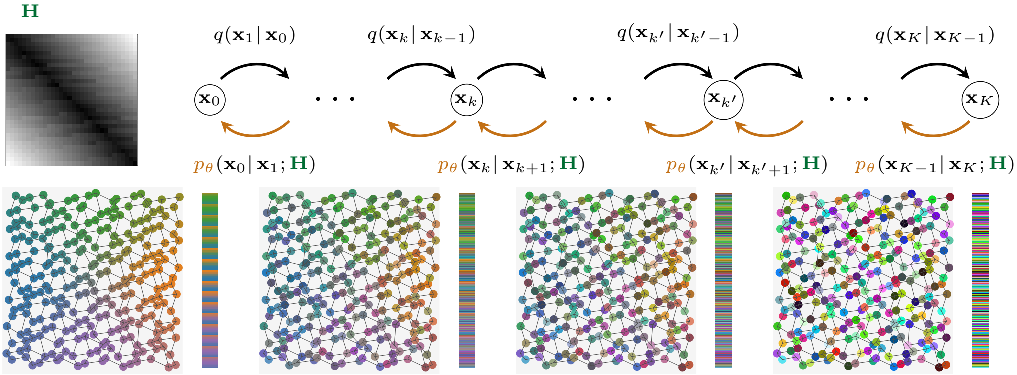




- GNNs are scalable, transferable parametrizations for the generation task.

- **Forward chain** transforms a policy sample $\mathbf{x}_0$ stochastically into a noise sample.

$$\Rightarrow \mathbf{x}_k(\mathbf{x}_0, \boldsymbol{\epsilon}) = \sqrt{\bar{\alpha}_k}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_k}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \Rightarrow q(\mathbf{x}_K) \approx \mathcal{N}(\mathbf{0}, \mathbf{I}).$$



- Learned reverse chain maps noise back into synthetic samples, conditional on $\mathbf{H}$. $\Rightarrow p_\theta(\cdot; \mathbf{H}) \approx \mathcal{D}^*_{\mathbf{x}}(\mathbf{H})$.

- We reparametrize the reverse chain p_θ and train a denoiser $\boldsymbol{\epsilon}_{\theta^*}$:

$$\theta^* \in \operatorname*{argmin}_{\theta} \mathcal{L}(\theta) := \mathbb{E}_{\mathbf{x}_0,\mathbf{H},k,\boldsymbol{\epsilon}} \left\| \boldsymbol{\epsilon}_\theta\left(\mathbf{x}_k(\mathbf{x}_0, \boldsymbol{\epsilon}), k; \mathbf{H}\right) - \boldsymbol{\epsilon} \right\|^2,$$

- We iterate the learned reverse chain:

$$p_{\theta^*}\left(\mathbf{x}_{k-1} \,\middle|\, \mathbf{x}_k; \mathbf{H}\right): \quad \mathbf{x}_{k-1} = \frac{1}{\sqrt{\alpha_k}}\left(\mathbf{x}_k - \frac{\beta_k}{\sqrt{1-\bar{\alpha}_k}} \boldsymbol{\epsilon}_{\theta^*}(\mathbf{x}_k, k; \mathbf{H})\right) + \sigma_k \mathbf{w}, \quad \mathbf{x}_K, \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad k = K, \dots, 1.$$

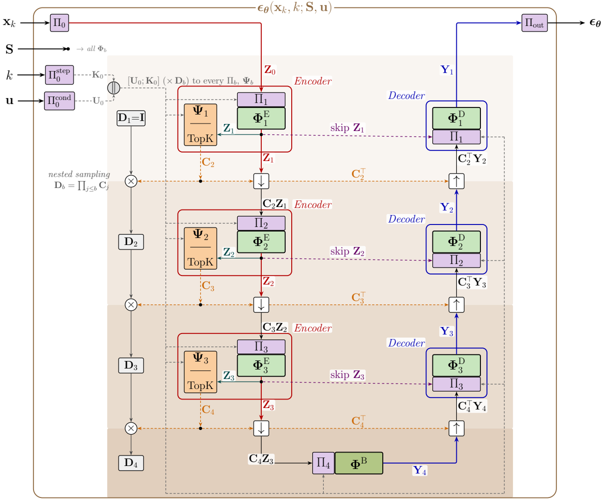
- We generate novel samples $\mathbf{x}_0 \,\middle|\, \mathbf{H} \sim p_{\theta^*}(\cdot; \mathbf{H}) := \mathcal{D}_{\mathbf{x}}(\mathbf{H}; \theta^*) \approx \mathcal{D}^*_{\mathbf{x}}(\mathbf{H})$.

U-GNN Denoiser Backbone

- A U-shaped encoder–decoder cascade of GNN blocks.
- Hierarchy of graph convolutions + learned pooling of nodes for coarsening.[2]

  $\Rightarrow$ Nested down/up-sampling by $\mathbf{D}_b$.

  $\Rightarrow$ Reduced GSOs $\mathbf{S}_b := \mathbf{D}_b \mathbf{S}^{\gamma_b} \mathbf{D}_b^\top$ with pooling and stride.[3]
- Conditioned jointly on channel graph GSO $\mathbf{S} = \mathbf{H}$ and node states $\mathbf{u}$ directly.

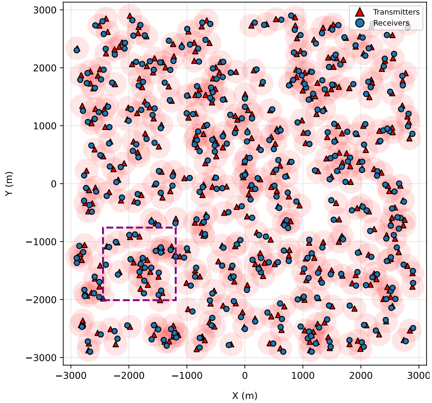


[2] Zhitao et. al., *Hierarchical Graph Representation Learning with Differentiable Pooling*, NeurIPS, 2018.

[3] Gama et. al., *Convolutional Neural Network Architectures for Signals Supported on Graphs*, IEEE TSP, 2018.

Experimental Results: Power Control

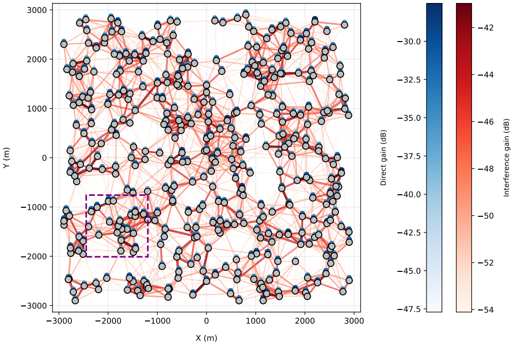
- $N = 400$ tx-rx pairs as nodes; long-term channel gains $\mathbf{H}$ are the GSO.



- Networks drawn from an RGG layout + large-scale fading (path-loss & shadowing).
- Mixed user densities $\nu \in \{6.6, 8.2, 10.1, 11.9\}$ pairs / km^2
- Instantaneous channel gains fluctuate $h_{ij,\tau} \sim \mathcal{D}_{\widetilde{\mathbf{H}}|\mathbf{H}}(\mathbf{H})$:

  $\Longrightarrow$ a small-scale (Rayleigh) fading model.

- Tx i allocates power $x_{i,\tau} \geq 0$ at time τ and causes interference to neighboring tx-rx pairs $j \in \mathcal{N}(i)$.
- Communication rate r determined by SINR at each rx j. $\Rightarrow \mathrm{SINR}_{j,\tau} = \dfrac{h_{jj,\tau} \,.\, x_{j,\tau}}{1 + \sum_{i \in \mathcal{N}(j)} h_{ij,\tau} \,.\, x_{i,\tau}}$.



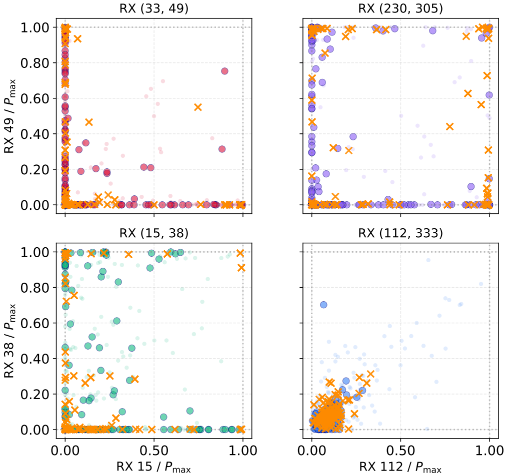
- Given $\mathbf{H}$, allocate transmit powers to maximize sum-rate utility, s.t. min-rate requirements and power budget $x_{\max}$:

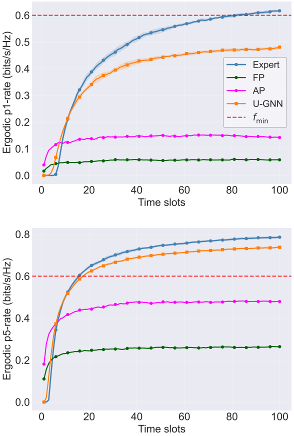
$$\mathrm{P}^*(\mathbf{H}) = \underset{\mathbf{x}_\tau \sim \mathcal{D}_{\mathbf{x}}(\mathbf{H})}{\text{maximum}} \ \frac{1}{T}\sum_{\tau=1}^{T}\sum_{j} r(\mathrm{SINR}_{j,\tau})$$

$$\text{subject to} \ \frac{1}{T}\sum_{\tau=1}^{T} r(\mathrm{SINR}_{j,\tau}) \geq r_{\min}, \ \forall j,$$

$$\text{subject to} \ 0 \leq \mathbf{x}_{i,\tau} \leq x_{\max}, \ \forall i, \tau = 1, \ldots, T.$$

- Optimal power control policies $\mathbf{x}_\tau^* \,\big|\, \mathbf{H} \sim \mathcal{D}_{\mathbf{x}}^*(\mathbf{H})$ are generally multi-modal.

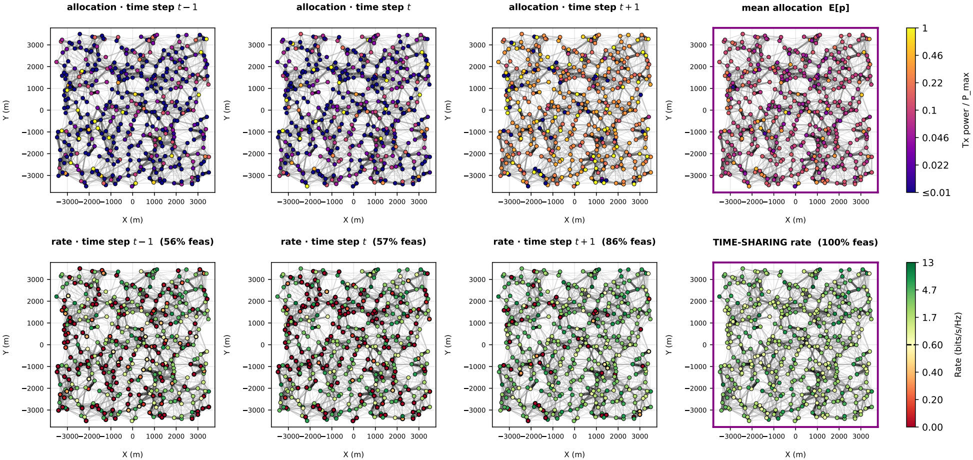




- U-GNN policies closely match the experts. ⟹ Improved percentile-rates over the baselines.

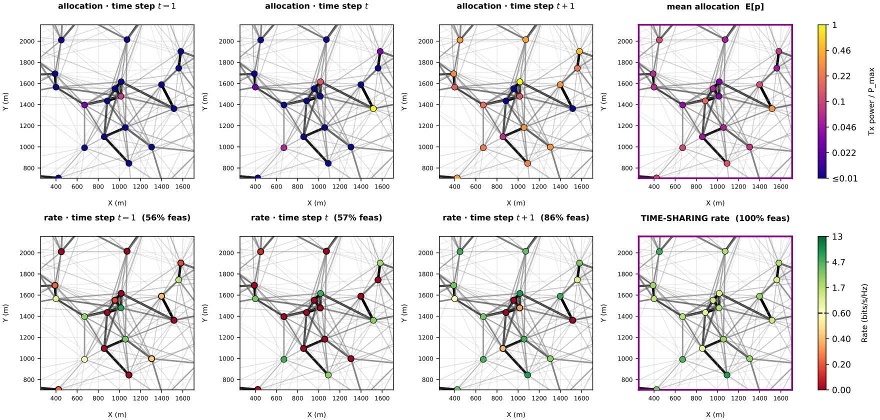
Optimal Time-Sharing of Resources

- Strongly-interfering pairs alternate their transmission slots. $\Rightarrow$ Policy-switching



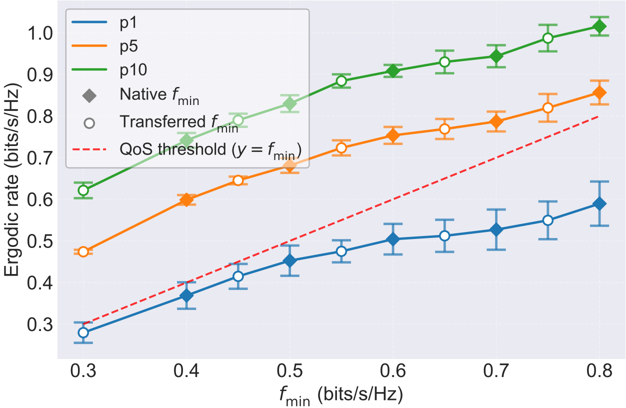
- A learned stochastic policy that is near-optimal and near-feasible in the ergodic sense.

- Strongly-interfering pairs alternate their transmission slots. $\Rightarrow$ Policy-switching



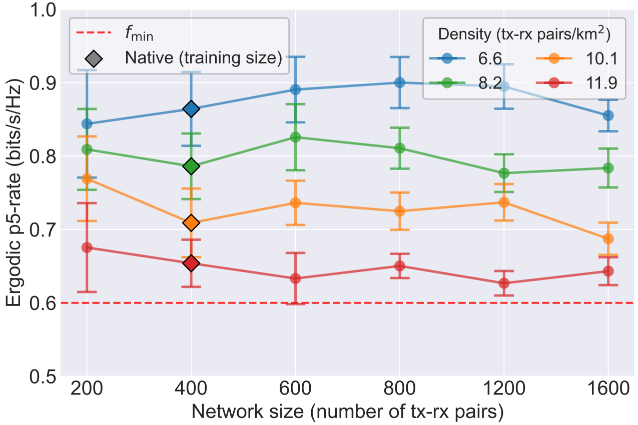
- A learned stochastic policy that is near-optimal and near-feasible in the ergodic sense.

- One U-GNN trained across $f_{\text{min}} \in \{0.4, \ldots, 0.8\}$, evaluated at unseen QoS levels (hollow markers).



- Percentile rates stay above the requirement across $f_{\text{min}} \in [0.3, 0.8]$ $\Rightarrow$ Inheriting the GNN's stability.

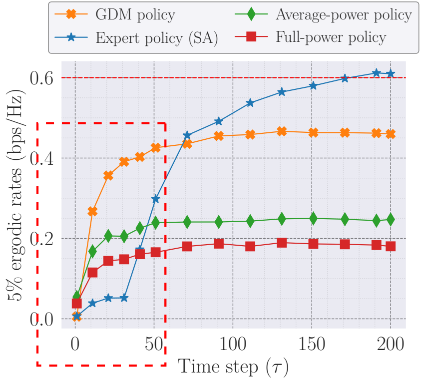
- Trained at $N = 400$ (mixed densities), evaluated on networks of varying size at $f_{\min} = 0.6$.

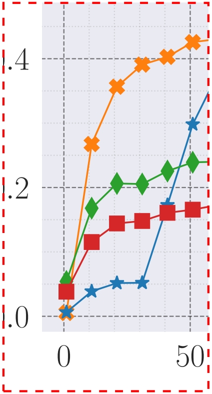


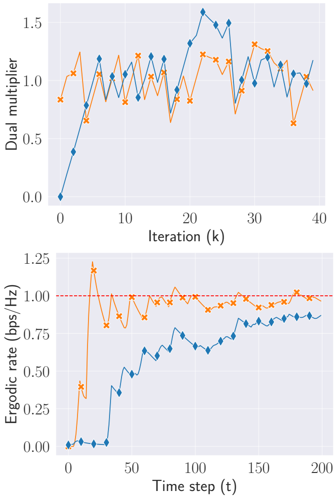
- Tail-rate percentiles stay stable across sizes. $\Rightarrow$ inheriting the GNN's scalability and transferability.

- Iterative dual descent trades off objective optimality for faster dynamics.

  $\Rightarrow$ Dual dynamics are required.







- Learned generative solutions bypass the transient and sample from the stationary dynamics.

- We trained a graph signal diffusion model to

  ⇒ imitate stochastic experts for constrained wireless optimization.

  ⇒ amortize thousands of primal-dual iterations into a single DDIM sampling pass.

  ⇒ generalize across QoS levels and transfers across network sizes and densities.

- Beyond wireless:

  ⇒ Denoising graph signals on known topologies. E.g., financial forecasting, RecSys . . .
  ⇒ Latent graph signal generative VAE/diffusion models.

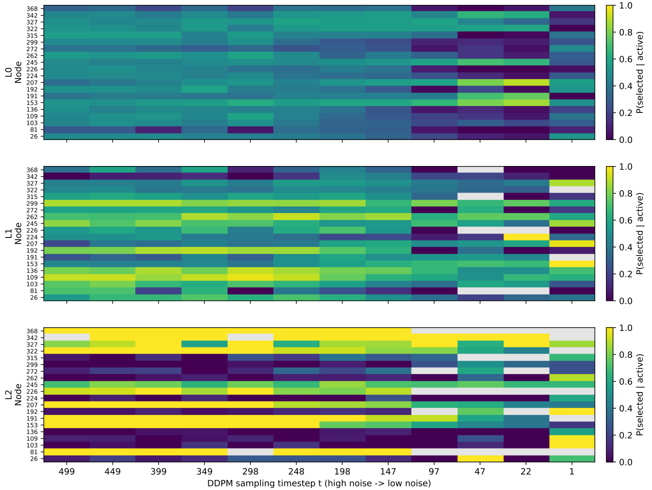
- Beyond imitating experts:

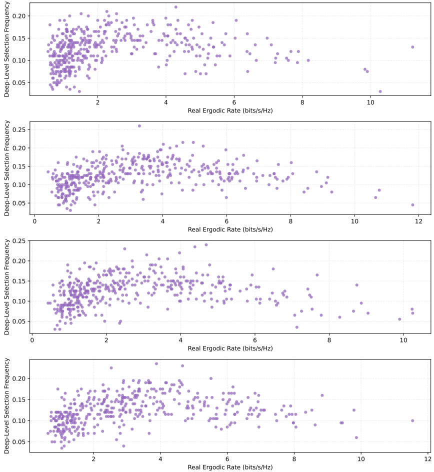
  ⇒ Energy-based models, stochastic control . . .

  ⇒ Inference-time algos. coupling diffusion-sampling and optimization.

Supplementary Slides: Interpretability of Learned Pooling

U-GNN Pooling Patterns in Wireless





U-GNN Pooling Patterns in S&P 500