

---

# 通信信息智能空口

北京航空航天大学电子信息工程学院

杨晨阳

[cyyang@buaa.edu.cn](mailto:cyyang@buaa.edu.cn)

RISTA前沿大讲堂

北京, 2026-09-07

## 报告提纲

背景/挑战：空口AI的特点、需求与困境

解决 思路：通信“信息”智能空口

核心 技术：无线通信知识 — “信息”

融入通信知识的系统化方法

学习偏置与观测偏置

结构偏置：无线策略的性质与数值算法的结构

效果 验证：可扩展、可泛化、多任务

总结/展望



## 空口AI的特点:

决策变量多维度高 (空时频功率资源、用户、天线、载波、小区维度...)

无线环境动态性强 (信道时变、业务动态)

不同类任务差异大 (信道估计、预编码、资源分配、MIMO 检测...)

无线边缘计算与存储资源受限

## 空口 AI 的需求:

可扩展: 训练代价低 (样本/可训练参数少)、推理速度快 (毫秒/微秒级软/硬实时)

可泛化: 场景 (信道分布/SNR/映射关系), 维度 (小区/用户/天线/载波数)

多任务: 适用于多类空口任务 (映射关系不同)

可解释: 知其然也知其所以然 (设计机理明确)



### 传统通信方法的困境

数值算法：高维/大规模优化算不动，无模型或部分模型问题难建模/性能差

**原因：**针对每个环境参数（如H）迭代计算一次决策变量、环境/系统复杂

⇒ **LTO：**针对每个策略（环境参数到决策变量的映射）设计一个DNN

### 现有学习方法的困境

大多采用通用AI技术：CNN/RNN/Transformer ... → LM

拼数据、拼算力、实验范式 → 即使LM也未必通用、可泛化

- 缺乏可扩展性：天线、带宽等规模不断增长，训练代价急剧上升
- 缺乏泛化能力：难以适应信道、业务（如用户数）的动态变化
- 难以一模多能：优化问题/算法/策略的结构各异
- 缺乏可解释性：主要通过经验设计，产生信任危机

学习效率低

**原因：**没有系统化利用无线领域知识



## 无线通信 — 工程科学

理解机理与限制、建模与理论分析、提出方法、实验验证 → 系统化  
引入合理假设/分析上下界/极限情况 → 优雅的结果、清晰的洞见

## AI for Communications

针对通信问题明确“智能”涵义和需求：大量空口问题需要高效地学习**策略**（映射关系）

针对通信问题分析学习机理 → 设计无线AI算法

自2018年以来，无线AI研究者一直在探索利用通信领域知识的方法 ——

➤ “模型驱动”学习、“模型-数据双驱动”、“数据-知识双驱动” .....

思想：结合数值算法和计算公式等计算/数学模型 → 提高可解释性

只学计算复杂度高或无模型的运算或参数 → 简化待学策略

方法：深度展开、利用最优解的结构（如 MU-MISO 预编码）、进行预处理 .....



其中：深度展开适用于很多优化问题 ——

如 SE/EE 最大预编码、MIMO 检测、信道估计、资源分配 .....

**因为**有系统化的设计思路 ——

保留特定数值算法 (如 WMMSE、OMP...) 对每个变量的迭代计算过程

把迭代过程中的高复杂度运算或无模型参数变为可训练模块或参数

**因为**空间和样本复杂度远低于纯数据驱动方法

**但是**训练和推理复杂度随规模增长快 (难以并行), 针对算法定制 (难以通用)

SE 最大 MU-MISO 预编码的样本复杂度和训练时间

|                  |        | 模型 GNN   | 传统 GNN          | BNN  | 深度展开      |
|------------------|--------|----------|-----------------|------|-----------|
| $N = 8, K = 4$   | 样本复杂度  | 100      | 40,000          | 10   | 10        |
|                  | CPU 时间 | 1 分 2 秒  | —               | —    | 4 分 43 秒  |
|                  | GPU 时间 | 32 秒     | 43 分 16 秒       | 11 秒 | —         |
| $N = 32, K = 16$ | 样本复杂度  | 100      | >100,000        | 10   | 10        |
|                  | CPU 时间 | 4 分 35 秒 | —               | —    | 70 分 21 秒 |
|                  | GPU 时间 | 39 秒     | >6 小时 22 分 48 秒 | 21 秒 | —         |

近 20 倍!



## ➤ 最新趋势：“物理”成为高频词！—— 物理信息机器学习/神经网络 (PIML/PINN)

PIML 的思想：机器学习/神经网络 + 物理定律/原理/约束

PIML 的动机：突破传统物理领域中数值和学习方法的困境

受关注的原因：痛点突破 —— 解决传统数值算法算不动、建模难的问题

解决传统学习方法样本多、泛化弱的问题

顶刊加持 —— Nature/Science/ICLR/NeurIPS 等相关论文爆发

根据 Nature Reviews Physics 综述<sup>[1]</sup>:

| 物理知识嵌入方式 | 相应的 PIML 方法   | 特点              |
|----------|---------------|-----------------|
| 学习偏置     | 损失函数中加入物理约束项  | 最灵活，PINN 采取这种方式 |
| 观测偏置     | 数据本身反映物理规律    | 最简单，但需要大量数据     |
| 结构偏置     | DNN 架构中引入物理知识 | 满足物理约束，但设计比较复杂  |



通信“信息”机器学习 (CIML)与 PIML 理念一致，但独立发展

通信知识：物理 + 数学 + 统计 ➔ 利用知识的方法与 PIML 不同

**理念：**把领域知识融入学习模型的设计、训练或推理过程 ——

在不降低学习性能的前提下降低样本/计算/存储/能量/通信代价、提高泛化能力

把传统 DNN 的黑盒化设计灰盒化

**知识：**静态/动态数学模型、数学性质...

物理：如微分方程、守恒定律（如质量/动量守恒）、对称性（如平移/旋转/标度不变性）

通信：如山农公式、优化问题、最优解/数值算法结构、对称性（如置换等变/不变性）

| 通信知识嵌入方式 | 可利用的通信知识举例        | 相应的 CIML 方法举例 (怎么学)   |
|----------|-------------------|-----------------------|
| 学习偏置     | 优化目标/约束、接收信号模型    | 无监督深度学习               |
| 观测偏置     | 置换性、相位不变性、服务/竞争关系 | 预/后处理、图模型构建           |
| 结构偏置     | 空口策略性质、数值算法结构     | GNN/GFormer、深度展开、算法同构 |



CIML, 除了“怎么学才高效”还有“学什么才好学”——学习方法 vs 待学映射

例如：对于 MU-MISO 线性预编码

纯数据驱动方法的待学策略为  $\mathbf{V} = f(\mathbf{H}) \rightarrow$  难学习、难泛化<sup>[2]</sup>

一种模型驱动方法待学策略为  $\{\mathbf{P}, \Lambda\} = f(\mathbf{H}) \rightarrow \mathbf{V}^* = \left( \mathbf{I}_N + \frac{1}{\sigma^2} \mathbf{H}^H \Lambda \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{P}^{\frac{1}{2}}$

又如：对于两类“端到端”预编码

联合优化信道获取与预编码的待学策略应该为  $\mathbf{V} = f(\mathbf{R}_1, \dots, \mathbf{R}_{T_0})$

即从导频时间序列到预编码的映射。因为信道获取与预编码的优化目标不同  $\rightarrow$

分别学习信道获取策略和预编码策略导致导频较短时 SE 较低

联合优化模拟与基带预编码的待学策略应该为  $\mathbf{V}_{\text{RF}} = f_{\text{RF}}(\mathbf{H})$  与  $\mathbf{V}_{\text{BB}} = f_{\text{BB}}(\mathbf{H}_{\text{eff}})$ <sup>[3]</sup>

即从信道到模拟预编码的映射、以及从等效信道到基带预编码的映射。

因为模拟和基带预编码优化的约束不同，且  $\mathbf{W}_{\text{BB}} = f_{\text{BB}}(\mathbf{H}, f_{\text{RF}}(\mathbf{H})) \rightarrow$

若直接学习  $\{\mathbf{V}_{\text{RF}}, \mathbf{V}_{\text{BB}}\} = f_{\text{HY}}(\mathbf{H})$ ，则  $f_{\text{RF}}(\cdot)$  被重复学习两次、导致学习效率低！



**无线领域知识** — 种类繁多：优化问题、数值算法、关系先验、策略结构...

视角：I-O关系 — 空口策略为高维函数 $Y = f_1(X)$ ，神经网络为高维函数族 $Y = f_2(X, \Theta)$

例如：大量空口策略存在一种尚未被广为认知的数学性质 — 对称性<sup>[4,5,6]</sup>

$$y = x^2$$

机理：集合元素无序  $\rightarrow$  集合上函数的置换 (PE) 性质 — 高维函数的对称性质

集合上的函数： $Y = f(X)$ ，其中  $X$  和  $Y \in \mathbb{R}^{M_1 \times M_2 \times \dots \times M_S}$  是  $M_S$  维张量

集合元素的输入特征  $X \rightarrow$  集合元素的输出特征  $Y$

一维对称函数，如 SISO 多用户功率分配  $\mathbf{p} = f(\mathbf{h})$ ，— 向量  $\rightarrow$  向量，对应用户集合

二维对称函数，如 MU-MISO 预编码  $V = f(H)$  —  $V \in \mathbb{C}^{N \times K} \rightarrow H \in \mathbb{C}^{N \times K}$ ，对应天线和用户集合

高维对称函数，如功率/RB 分配/调度/接入、MIMO 预编码/检测器、信道估计策略 — 张量  $\rightarrow$  张量，对应于天线、RF 链、用户、载波、小区等集合，满足多个集合导致的置换性质

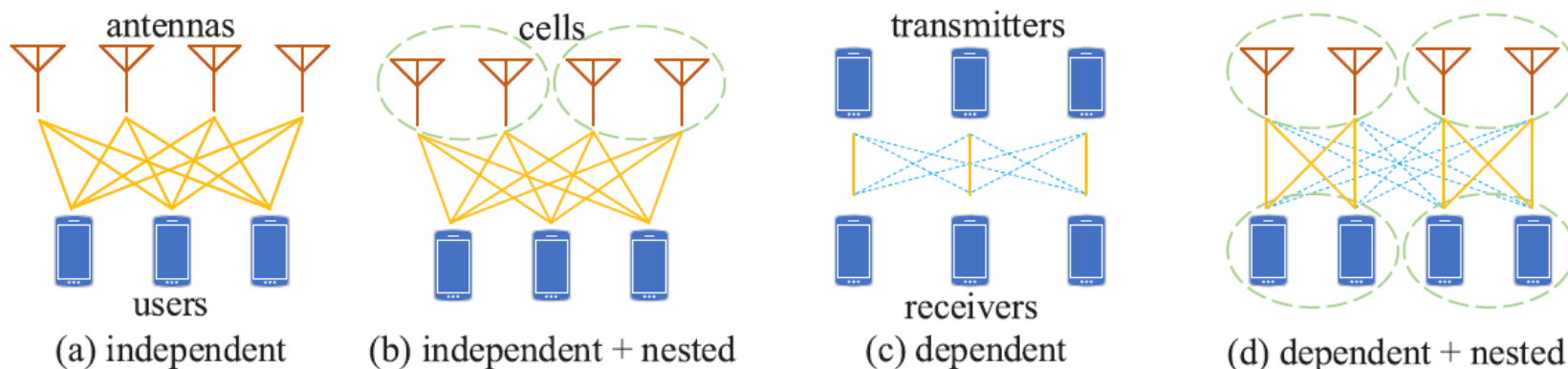


## 空口问题中的集合类型、关系与个数<sup>[6, 7]</sup>

两类集合：常规集合 $\{1, 2, \dots, K\}$ 、嵌套集合 $\{\{1_1, \dots, N_1\}, \{1_2, \dots, N_2\}, \dots, \{1_M, \dots, N_M\}\}$  (两层/多层)

集合关系：独立(independent)、相关(dependent)

集合个数：从单集合到六集合（目前已发现）



两集合问题实例：(a) MU-MISO 预编码, (b) CoMP-JT, (c) D2D 功控, (d) 多小区功率分配

对应的置换性质： $\Pi_1^T \mathbf{W}^T \Pi_2 = f_a(\Pi_1^T \mathbf{H} \Pi_2)$ ,  $\Pi_1^T \mathbf{W}^T \Omega = f_b(\Pi_1^T \mathbf{H} \Omega)$ ,  $\Pi^T \mathbf{p} = f_c(\Pi^T \mathbf{H} \Pi)$ ,  $\Omega^T \mathbf{P} \Omega = f_d(\Omega^T \mathbf{H} \Omega)$

$$\Omega^T \triangleq (\Pi_3^T \otimes \mathbf{I}_{N_t}) \text{diag}(\Pi_{2,1}^T, \dots, \Pi_{2,B}^T)$$



## 集合上函数的结构<sup>[7]</sup>

### ➤ 单集合上函数的结构 (四个基函数)

常规集合 — 两个Arbitrary PE (APE)函数:

$$y_k = f_I(x_k, \sum_{j=1, j \neq k}^K q_I(x_j))$$

I 型 APE/NPE 函数 — 常规处理函数

$$y_k = f_{II}(x_k, \sum_{j=1, j \neq k}^K q_{II}(x_k, x_j))$$

II 型 APE/NPE 函数 — 注意力处理函数

嵌套集合 — 两个Nested PE (NPE)函数:

$$y_{k_m} = f_I\left(x_{k_m}, \sum_{j=1, j \neq k}^K q_{I,1}(x_{j_m}), \sum_{i=1, i \neq m}^M q_{I,2}\left(\sum_{j=1}^K q_{I,3}(x_{j_i})\right)\right)$$

$$y_{k_m} = f_{II}\left(x_{k_m}, \sum_{j=1, j \neq k}^K q_{II,1}(x_{k_m}, x_{j_m}), \sum_{i=1, i \neq m}^M q_{II,2}\left(\sum_{j=1}^K q_{II,3}(x_{k_m}, x_{j_i})\right)\right)$$

### ➤ 多集合上函数的结构(不唯一), 一种按照维度进行分解的结构 ——

维度递归结构: 维度个数 → 集合个数 → 递归次数

各集合类型 + 集合间关系 + 元素间关系 → 各维度上的基函数



又如：混合预编码策略存在相位不变性<sup>[8]</sup>

原因：目标函数与约束都与复数模值有关

$$\begin{aligned} \max_{\mathbf{W}_{RF}, \mathbf{W}_{BB}} R(\mathbf{H}, \mathbf{W}_{RF}, \mathbf{W}_{BB}) &= \sum_{k=1}^K R_k \\ \text{s. t. } \begin{cases} R_k = \log_2 \left( 1 + \frac{|\mathbf{h}_k^H \mathbf{W}_{RF} \mathbf{w}_{BBk}|^2}{\sum_{i=1, i \neq k}^K |\mathbf{h}_k^H \mathbf{W}_{RF} \mathbf{w}_{BBi}|^2 + \sigma^2} \right) \\ \|\mathbf{W}_{RF} \mathbf{W}_{BB}\|_F^2 = P_{max} \\ (\mathbf{W}_{RF})_{jl} = 1, j = 1, \dots, N_t, l = 1, \dots, N_s \end{cases} \end{aligned}$$

对于混合预编码策略  $(\mathbf{W}_{RF}, \mathbf{W}_{BB}) = f(\mathbf{H})$ :

$$(\Phi_2^{-1} \mathbf{W}_{RF}, \mathbf{W}_{BB}) = f(\Phi_1 \mathbf{H} \Phi_2)$$

$$(\mathbf{W}_{RF} \Phi_3, \Phi_3^{-1} \mathbf{W}_{RF} \Phi_4) = f(\mathbf{H})$$

$$\Phi_1 = \begin{pmatrix} e^{j\phi_{11}} & & \\ & \ddots & \\ & & e^{j\phi_{1K}} \end{pmatrix} \in \mathbb{C}^{K \times K}, \phi_{1k} \in [0, 2\pi), k = 1, \dots, K$$

$\Phi_2 \in \mathbb{C}^{N_t \times N_t}, \Phi_3 \in \mathbb{C}^{N_s \times N_s}, \Phi_4 \in \mathbb{C}^{K \times K}$  与  $\Phi_1$  结构相同

再如：信道估计策略存在相对位置依赖性

可以证明：对于空间相关信道， $\mathbf{h}_{UL} = f(\mathbf{r})$  在天线维度不满足置换等变性

单用户信道的协方差矩阵依赖于天线的相对位置 —— 均匀线阵为Toeplitz矩阵

$$\mathbf{C} = \mathbb{E}[\mathbf{a}(\phi) \mathbf{a}^H(\phi)] \triangleq \mathbb{E}[\tilde{\mathbf{C}}(\phi)], \quad \tilde{c}_{n,n'}(\phi) = e^{j2\pi(n-n') \frac{d \sin \phi}{\lambda}}$$

LMMSE估计策略  $f(\mathbf{r}) = \mathbf{W} \mathbf{r}$ ,  $\mathbf{W} = (\mathbf{C} + \frac{\sigma_{UL}^2}{P_{ULQ}} \mathbf{I})^{-1} \mathbf{C}$ , 近似满足相对位置依赖性



## 融入通信知识的系统化方法：

**学习偏置：**例如有/无模型无监督深度学习<sup>[9,10,11]</sup> —

可以利用优化目标/约束函数构成拉格朗日函数满足 QoS 约束

可以利用显式/部分显式的目标/约束函数消除/减少与环境的交互

可以解决联合优化多个模块（如调度+预编码）的端到端学习问题

DRL  
难解决

**观测偏置：**例如样本预处理/后处理、图构建 —

可以利用无线策略的置换性质对样本/决策变量排序从而消除冗余样本<sup>[5]</sup>

可以利用无线关系先验构建图模型从而使 GNN/GFormer 利用这类先验<sup>[6-8,12]</sup>：

从环境参数与决策变量维度、优化目标与约束、服务与竞争关系中分析集合

→ 图中的顶点、边及其类别与相应特征



**结构偏置：**两类面向可扩展、可泛化和多任务等挑战发现并利用通信结构的方法

策略性质：

**思路：**利用空口策略数学性质限制 DNN 结构，如设计与策略置换性质匹配的 DNN<sup>[2,12~15]</sup>

**优点：**训练开销较小、推理复杂度较低、维度可泛化、结构可解释

假设空间较小 → 达到同样性能时降低 DNN 空间/样本复杂度/提高 DRL 收敛速度

某维度可置换 → 该维度有可能扩展和维度泛化

如 MU-MISO 预编码满足 2D-PE，天线和用户置换 →  $N!K!$  个等价解

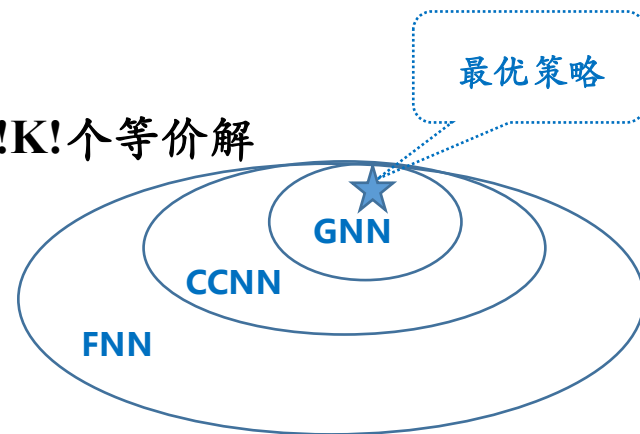
应设计 DNN 只表示 2D-PE 函数族

如果 DNN 与策略置换性质失配会如何？

FNN 最灵活、能表示 2D-PE 函数族 → 效率低

Transformer (有位置编码) 不满足置换等变性 → 性能差

CCNN 能表示平移等变函数族，也包含 2D-PE 函数族 → 效率低



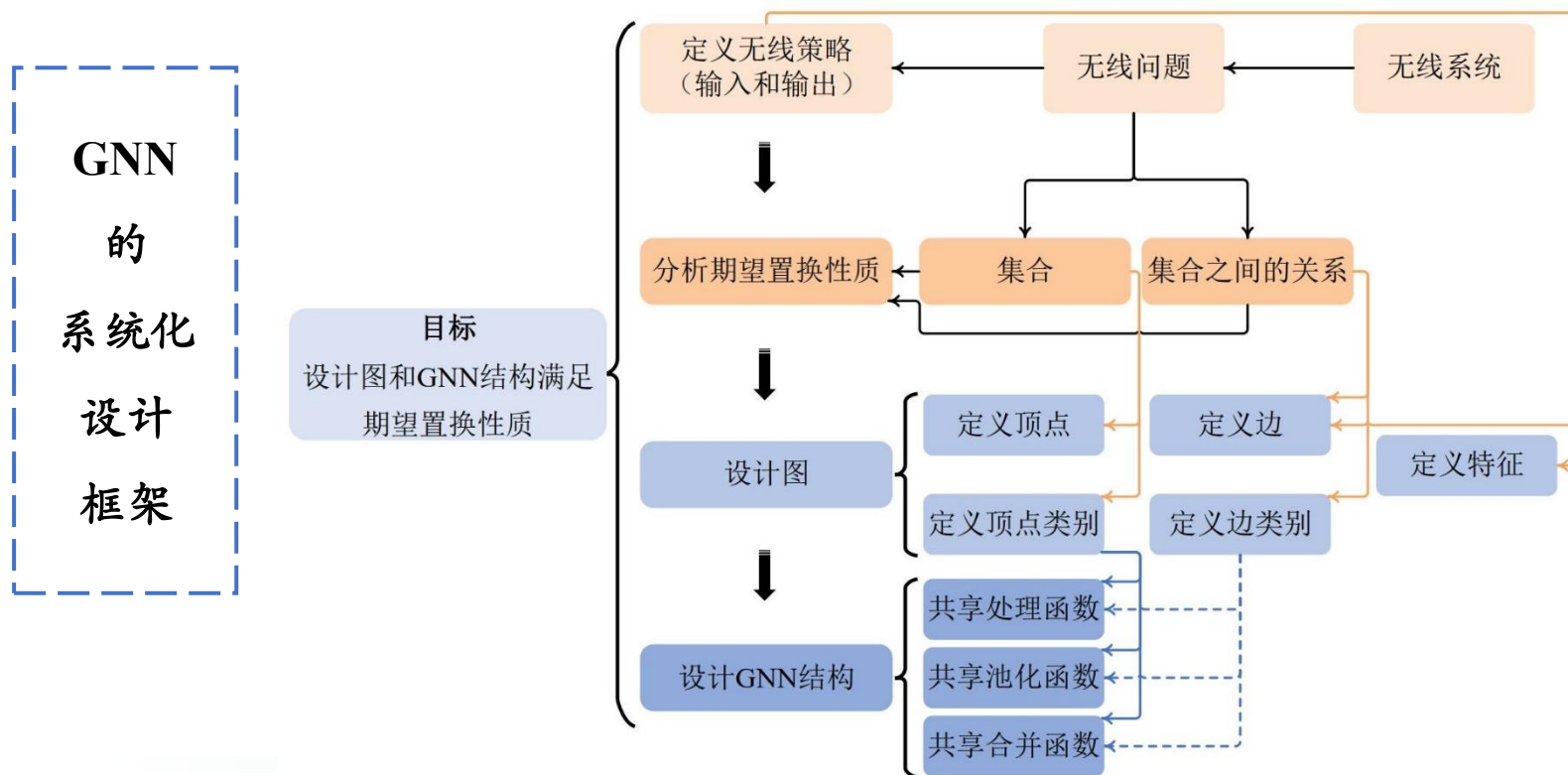
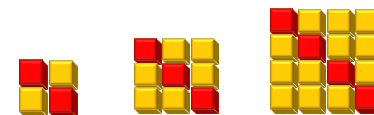
方法：识别无线问题中的集合（个数、类别、关系）→ 确定无线策略的置换性质<sup>[6]</sup>

## ➤ 设计单任务置换网络

FNN — 根据置换性质设计参数共享<sup>[6,13,16,17]</sup>

GNN — 根据集合建模图 + 根据顶点与边的类别设计函数共享<sup>[12]</sup>

Transformer — 根据集合设计词元/注意力 + 根据置换性质设计参数共享<sup>[18]</sup>



## ➤ 设计多任务网络

✧ 学习满足多种置换性质的多个空口策略<sup>[19]</sup> — 基于置换性质间关系进行组合式设计

机理：

1) 满足不同 PE 性质 DNN 的假设空间存在包含关系 ——

例如： $F_{\text{ind.3D-PE}} \subseteq F_{\text{ind.2D-PE}} \subseteq F_{\text{partial-nested 2D-PE}} \subseteq F_{\text{1D-PE}}$

2) DNN的PE性质取决于假设空间最大那一层的PE性质

方法：

设计基本模块满足单+双集合上函数的PE性质 ——

有无参数共享的注意力模块和FFN模块、以及输出模块

➔ 主要考虑  $\leq$  两集合的无线问题，复用7个模块，适配10种PE性质，覆盖>30种无线策略

✧ 学习满足多类不同性质的多个空口策略 — 基于时空频维度分解进行组合式设计



## 算法结构：

思路：把算法“求解路径”中变量间的依赖关系“翻译”为 **DNN** 架构<sup>[7,20]</sup>

优点：训练开销小、维度可泛化、适于有/无置换性质的策略

方法：算法同构<sup>[7]</sup> (置换性相关的共性结构)，算法启发<sup>[20-22]</sup> (置换无关结构)

➤ **单任务DNN-1**：算法同构 — 集合上函数结构 → 重写迭代算法、引入可训练处理/合并器

第 $k$ 个维度可置换策略的两种结构：

$$y_k = f_I(x_k, \sum_{j=1, j \neq k}^K q_I(x_j))$$

$$y_k = f_{II}(x_k, \sum_{j=1, j \neq k}^K q_{II}(x_k, x_j))$$

两个维度：天线、用户

### WMMSE算法

$$\begin{aligned} w_k^{(\ell+1)} &= (\widetilde{H}_j \widetilde{H}_j^H + \mu^* I)^{-1} h_k u_k^{(\ell)} z_k^{(\ell)} \\ u_k^{(\ell+1)} &= \frac{h_k^H w_k^{(\ell)}}{\sum_{j=1}^K |h_k^H w_j^{(\ell)}|^2 + \sigma_0^2} \\ z_k^{(\ell+1)} &= \frac{1}{1 - u_k^{(\ell)} h_k^H w_k^{(\ell)}} \end{aligned}$$

### 重新表述成GNN的更新方程形式

红框内：求和 → GNN的**池化函数**

黄框内：求和函数**内**运算 → GNN**处理器**

绿框内：求和函数**外**运算 → GNN**合并器**

第一次递归：满足**用户**维度置换性质(

$$d_k^{(\ell+1)} = f(d_k^{(\ell)}, \sum_{j=1}^K q(d_k^{(\ell)}, d_j^{(\ell)}))$$

用户维度：内积运算 → 注意力机制

第二次递归：满足**天线**维度置换性质

$$f(\cdot): x \mapsto y: y_k = f_f(x_k, \sum_{i=1}^N q_f(x_i))$$

$$q(\cdot): x \mapsto y: y_k = f_q(x_k, \sum_{i=1}^N q_q(x_i))$$

天线维度：无需注意力

### 谱效最大 MU-MISO 预编码

重新表述 WMMSE 算法：

1. 两次递归，满足两个维度的 PE 性质

2. 仅在产生干扰的维度上计算注意力

— 2<sup>nd</sup> 递归处理器与合并器可训练、天线和用户共享

— 同一问题多个算法可表述成相同形式

— 适于能效最大等其他问题



## ➤ 单任务DNN-2：算法启发 — 分析并借鉴迭代算法的更新方式

举例：学习 MU-MISO 预编码的 DNN

注意力处理函数的具体结构？

GCN(第 $k$ 个边表示)与无位置编码/无解码器 Transformer(第 $k$ 个 token 表示)的更新方程：

$$\mathbf{d}_{k,n}^{(l)} = \sigma \left( \mathbf{S}^{(l)} \mathbf{d}_{k,n}^{(l-1)} + \sum_{i=1}^{N_T} \mathbf{Q}^{(l)} \mathbf{d}_{k,i}^{(l-1)} + \sum_{j=1}^K \mathbf{P}^{(l)} \mathbf{d}_{j,n}^{(l-1)} \right)$$

$$\mathbf{d}_k^{(\ell)} = \sigma \left( \mathbf{W}^F \mathbf{d}_k^{(\ell-1)} + \mathbf{W}^F \sum_{i=1}^K \text{softmax} \left( \left( \mathbf{W}^Q \mathbf{d}_k^{(\ell-1)} \right)^\top \left( \mathbf{W}^K \mathbf{d}_i^{(\ell-1)} \right) \right) \mathbf{W}^V \mathbf{d}_i^{(\ell-1)} \right)$$

谱效/log 谱效最大预编码 GD 算法第  $k$  个用户预编码向量的第  $l$  次迭代方程：

$$\mathbf{v}_k^{(l)} = \mathbf{v}_k^{(l-1)} - \delta \sum_{i=1}^K \rho_{k,i}(\mathbf{h}_i, \mathbf{V}^{(l-1)}) \mathbf{h}_i$$

→ 梯度启发 GNN(第 $k$ 个用户顶点与第 $n$ 个天线顶点间第 $t$ 个边表示)的更新方程<sup>[21]</sup>：

$$d_{k,n,t}^{(l)} = \sigma(s^{(l)} d_{k,n,t}^{(l-1)} + p^{(l)} \sum_{i=1}^K \alpha_{k,i,t}^{(l-1)} h_{i,n})$$

$$\alpha_{k,i,t}^{(l-1)} = \phi(\underbrace{\mathbf{h}_i^H \bar{\mathbf{d}}_{k,t}^{(l-1)}}_{\text{UE}_k \text{ 对 UE}_i \text{ 的干扰}}, \underbrace{\sum_{j=1, j \neq i}^K |\mathbf{h}_i^H \bar{\mathbf{d}}_{j,t}^{(l-1)}|^2}_{\text{UE}_i \text{ 受到的干扰}}, \underbrace{|\mathbf{h}_i^H \bar{\mathbf{d}}_{i,t}^{(l-1)}|^2 \cdot \xi_{i,k}}_{\text{UE}_i \text{ 的接收功率}}), \quad \xi_{i,k} = 0 \text{ if } i = k, \xi_{i,k} = 1 \text{ if } i \neq k$$



## ➤ 多任务DNN

**方法：** 寻找不同问题数值算法结构的共性<sup>[23]</sup>

**举例：** 学习不同系统 (有/无 RIS 的 MIMO/MISO) 多用户预编码的 DNN

$$\max_{\mathcal{X}} \sum_{k=1}^K R_k(\mathcal{X})$$

$$\text{s.t.} \sum_{k=1}^K \|\tilde{\mathbf{V}}_k\|_F^2 \leq P_{\max},$$

$$|(\mathbf{V}_{\text{RF}})_{i,j}| = 1, \forall i, j, \quad \text{if required,}$$

$$|\theta_n| = 1, \forall n, \quad \text{if required,}$$

$$R_k(\mathcal{X}) = \log_2 \det \left( \mathbf{I} + \mathbf{Q}_{k,k} \mathbf{Q}_{k,k}^H \left( \sum_{j=1, j \neq k}^K \mathbf{Q}_{k,j} \mathbf{Q}_{k,j}^H + \sigma_0^2 \mathbf{I} \right)^{-1} \right)$$

不同系统的变量集合、预编码和 Q 的形式及约束不同，如 ——

MU-MISO 混合预编码：  $\mathcal{X} \triangleq \{\mathbf{V}, \mathbf{V}_{\text{RF}}\}$ ,  $\tilde{\mathbf{V}}_k \triangleq \mathbf{V}_{\text{RF}} \mathbf{V}_k$ ,  $\mathbf{Q}_{k,j} \triangleq \mathbf{H}_k \mathbf{V}_{\text{RF}} \mathbf{V}_j$ .

RIS+MU-MISO 混合预编码：  $\mathcal{X} \triangleq \{\mathbf{V}, \mathbf{V}_{\text{RF}}, \boldsymbol{\Theta}\}$ ,  $\tilde{\mathbf{V}}_k \triangleq \mathbf{V}_{\text{RF}} \mathbf{V}_k$ ,  $\mathbf{Q}_{k,j} \triangleq (\mathbf{H}_k^d + \mathbf{G} \boldsymbol{\Theta} \mathbf{H}_k^r) \mathbf{V}_{\text{RF}} \mathbf{V}_j$ .

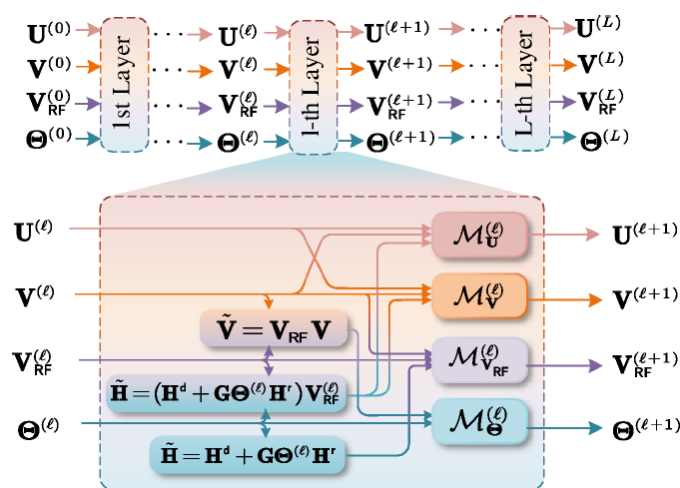
**交替优化多种变量，更新不同系统每种变量的迭代方程结构相同(如更新模拟预编码)：**

$$\mathbf{V}_{\text{RF}} \leftarrow \mathcal{P}_{\text{RF}} \left( \mathbf{V}_{\text{RF}}, \sum_{j=1}^K \tilde{\mathbf{H}}_j^H \mathbf{U}_j^H \mathbf{V}_j^H + \sum_{j=1}^K \sum_{k=1}^K \tilde{\mathbf{H}}_j^H \mathbf{U}_j^H (\mathbf{U}_j \tilde{\mathbf{H}}_j \mathbf{V}_k) \mathbf{V}_k^H \right. \\ \left. + \frac{\sigma_0^2}{P_{\max}} \sum_{j=1}^K \sum_{k=1}^K \mathbf{V}_{\text{RF}} \mathbf{V}_k \mathbf{U}_j \mathbf{U}_j^H \mathbf{V}_k^H \right), \quad (7)$$

- For HP in FAS,  $\tilde{\mathbf{H}}_k = \mathbf{H}_k$ ;
- For HP in RIS systems,  $\tilde{\mathbf{H}}_k = \mathbf{H}_k^d + \mathbf{G} \boldsymbol{\Theta} \mathbf{H}_k^r$ .

$$\mathcal{M}_{\mathbf{V}_{\text{RF}}}^{(\ell)} : \mathbf{V}_{\text{RF}}^{(\ell+1)} = \text{FNN}_{\mathbf{V}_{\text{RF}}, \text{ele}}^{(\ell)} \left( \mathbf{V}_{\text{RF}}^{(\ell)}, \sum_{j=1}^K \tilde{\mathbf{H}}_j^{(\ell)H} \mathbf{U}_j^{(\ell)H} \mathbf{V}_j^{(\ell)H} \right),$$

$$\sum_{j=1}^K \sum_{k=1}^K \tilde{\mathbf{H}}_j^{(\ell)H} \mathbf{U}_j^{(\ell)H} (\mathbf{U}_j^{(\ell)} \tilde{\mathbf{H}}_j^{(\ell)} \mathbf{V}_k^{(\ell)}) \mathbf{V}_k^{(\ell)H}, \\ \frac{\sigma_0^2}{P_{\max}} \sum_{j=1}^K \sum_{k=1}^K \mathbf{V}_{\text{RF}}^{(\ell)} \mathbf{V}_k^{(\ell)} \mathbf{U}_j^{(\ell)} \mathbf{U}_j^{(\ell)H} \mathbf{V}_k^{(\ell)H} \Big), \quad (16)$$



## 可扩展：复杂度~系统规模

MU-MISO 预编码的学习性能和推理时间<sup>[20]</sup>

**Model-GNN**：算法启发的 GNN，有用户间注意力机制

**传统 GNN**：GCN，无注意力机制

**深度展开**：针对总功率约束下 SE 最大问题的 WMMSE 算法

**BNN**：利用 MU-MISO 最优预编码结构，只学习功率分配、再利用公式重构预编码矩阵

各 DNN 在所有场景下都用 1000 个样本训练

Model-GNN 的网络规模几乎不随  $N$  和  $K$  变化，训练时间为秒级，传统 GNN 和深度展开的训练时间随  $N$  和  $K$  快速从分钟级增长到小时级

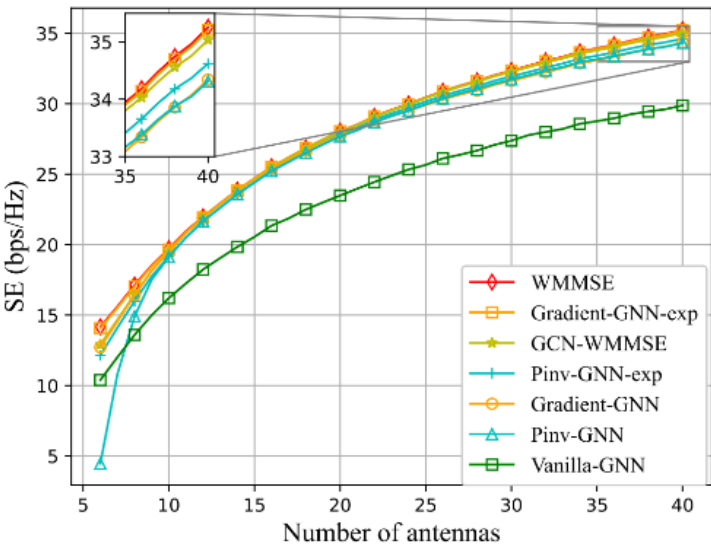
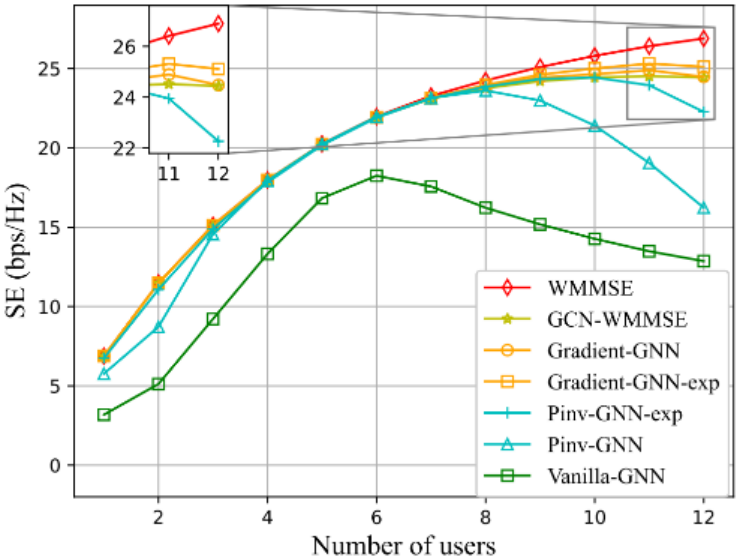
| SNR<br>(dB) | $(N, K)$ | Model-GNN |             |             | 传统 GNN |             | 深度展开 * |             | BNN    |             | WMMSE |             |
|-------------|----------|-----------|-------------|-------------|--------|-------------|--------|-------------|--------|-------------|-------|-------------|
|             |          | 谱效比       | 时间<br>(GPU) | 时间<br>(CPU) | 谱效比    | 时间<br>(GPU) | 谱效比    | 时间<br>(CPU) | 谱效比    | 时间<br>(GPU) | 谱效比   | 时间<br>(CPU) |
| 0           | (8, 4)   | 97.76     | 0.06        | 0.12        | 90.42  | 0.19        | 96.37  | 1.07        | 98.74  | 0.03        | 100   | 0.9         |
|             | (32, 16) | 99.01     | 0.07        | 0.13        | 81.63  | 0.19        | 95.08  | 7.89        | 99.07  | 0.03        | 100   | 6.85        |
|             | (64, 16) | 99.74     | 0.08        | 0.20        | 69.71  | 0.17        | 92.67  | 8.73        | 95.55  | 0.04        | 100   | 39.78       |
|             | (64, 32) | 99.42     | 0.09        | 0.21        | 68.37  | 0.18        | 96.04  | 28.89       | 99.57  | 0.05        | 100   | 51.3        |
| 10          | (8, 4)   | 97.74     | 0.06        | 0.12        | 67.7   | 0.17        | 97.69  | 1.08        | 99.92  | 0.04        | 100   | 1.37        |
|             | (32, 16) | 99.03     | 0.07        | 0.13        | 42.74  | 0.17        | 90.64  | 7.23        | 99.97  | 0.04        | 100   | 12.22       |
|             | (64, 16) | 99.65     | 0.08        | 0.20        | 42.05  | 0.17        | 87.97  | 8.74        | 99.99  | 0.03        | 100   | 122.25      |
|             | (64, 32) | 99.41     | 0.09        | 0.21        | 37.45  | 0.18        | 80.38  | 28.51       | 100.4  | 0.05        | 100   | 93.05       |
| 20          | (8, 4)   | 97.49     | 0.06        | 0.13        | 38.81  | 0.16        | 98.11  | 1.08        | 100.1  | 0.04        | 100   | 5.6         |
|             | (32, 16) | 98.35     | 0.08        | 0.13        | 24.23  | 0.16        | 75.87  | 7.96        | 101.19 | 0.04        | 100   | 43.96       |
|             | (64, 16) | 99.75     | 0.08        | 0.20        | 28.16  | 0.17        | 83.66  | 8.81        | 100.08 | 0.04        | 100   | 289.41      |
|             | (64, 32) | 98.94     | 0.09        | 0.21        | 25.32  | 0.18        | 77.47  | 28.89       | 101.67 | 0.05        | 100   | 276.87      |

\*：对于深度展开网络，为了在可接受的时间范围内进行训练，只训练了 10 个轮次。因此，在  $N$  与  $K$  较大、SNR 较高时，该网络所达到的谱效比较低。



**可泛化：** 无需再训练、在未见过的系统规模(PE+注意力)和分布(鲁棒无监督训练)下推理<sup>[21,24]</sup>

MU-MISO 预编码：  
训练样本的系统配置  
 $N = 12, K = 6$



MU-MISO 模数混合预编码的平均 SE：训练样本服从瑞利分布

|             |     | $\rho_U$ varies ( $\rho_A = 0$ ) |       |              |              | $\rho_A$ varies ( $\rho_U = 0$ ) |        |              |              |
|-------------|-----|----------------------------------|-------|--------------|--------------|----------------------------------|--------|--------------|--------------|
|             |     | 0.2                              | 0.4   | 0.6          | 0.8          | 0.2                              | 0.4    | 0.6          | 0.8          |
| DC (bps/Hz) |     | 18.77                            | 17.62 | 15.65        | 12.36        | 19.13                            | 18.86  | 18.14        | 16.26        |
| RGNN        | PDL | 100.3%                           | 99.2% | 91.3%        | 72.6%        | 100.6%                           | 99.1%  | 95.6%        | 78.3%        |
|             | UAT | 99.9%                            | 98.8% | <b>95.8%</b> | <b>87.4%</b> | 100.4%                           | 100.3% | <b>99.7%</b> | <b>98.6%</b> |
| 3D-GNN      | PDL | 98.7%                            | 97.3% | 93.8%        | 79.2%        | 99.1%                            | 98.5%  | 96.4%        | 84.9%        |
|             | UAT | 100.0%                           | 99.4% | <b>98.1%</b> | <b>93.0%</b> | 100.3%                           | 99.9%  | <b>99.5%</b> | <b>95.5%</b> |
| CNN         | PDL | 82.2%                            | 80.0% | 74.2%        | 57.6%        | 82.6%                            | 81.9%  | 80.2%        | 70.5%        |
|             | UAT | 92.5%                            | 91.3% | 89.2%        | 80.2%        | 92.8%                            | 92.1%  | 91.3%        | 84.0%        |

|             |     | Sparse Channels<br>Number of paths, $L$ |              |              |              | Scenarios in TR 38.901 |               |              |               |
|-------------|-----|-----------------------------------------|--------------|--------------|--------------|------------------------|---------------|--------------|---------------|
|             |     | 5                                       | 4            | 3            | 2            | UMa NLOS               | UMa LOS       | UMi NLOS     | UMi LOS       |
| DC (bps/Hz) |     | 18.28                                   | 18.24        | 17.93        | 17.43        | 9.74                   | 15.44         | 14.89        | 16.84         |
| RGNN        | PDL | 96.6%                                   | 95.0%        | 89.7%        | 80.2%        | 95.0%                  | 88.1%         | 73.8%        | 82.6%         |
|             | UAT | 97.8%                                   | 96.1%        | 92.7%        | 84.6%        | 111.3%                 | 100.5%        | 91.7%        | 96.1%         |
| 3D-GNN      | PDL | 96.4%                                   | 95.7%        | 93.1%        | 86.0%        | 97.0%                  | 91.5%         | 76.4%        | 87.2%         |
|             | UAT | <b>99.1%</b>                            | <b>98.4%</b> | <b>96.6%</b> | <b>90.2%</b> | <b>117.2%</b>          | <b>103.9%</b> | <b>95.1%</b> | <b>100.3%</b> |
| CNN         | PDL | 77.4%                                   | 75.8%        | 70.5%        | 60.8%        | 95.9%                  | 77.8%         | 69.4%        | 74.2%         |
|             | UAT | 87.9%                                   | 86.0%        | 80.8%        | 70.3%        | 102.6%                 | 86.2%         | 77.3%        | 82.7%         |



多策略：多类型、多系统的可置换策略——

预编码 (MIMO/MISO/宽带/CB)、MIMO 检测、信道估计、功率分配/功控<sup>[19]</sup>

每个策略仅需 100 个样本，可泛化到未见过的小区/用户/天线数

总参数量为 11.2k < 9 个单策略模型总和的 1/3、LLM 的十万分之一！

| Wireless tasks                    | Metric   | LMMSE | WMMSE | PE-MoFormer |                                                                                                       | Specific PE-Formers |       | Fine-tuned LLM |                                                                          |
|-----------------------------------|----------|-------|-------|-------------|-------------------------------------------------------------------------------------------------------|---------------------|-------|----------------|--------------------------------------------------------------------------|
|                                   |          | Perf. | Perf. | Perf.       | Space                                                                                                 | Perf.               | Space | Perf.          | Space                                                                    |
| Channel estimation                | MSE      | 0.089 | —     | 0.026       | Pre-processing:<br>0.76k<br>“Backbone”:<br>9.71k<br>Post-processing:<br>0.73k<br><br>Total:<br>11.20k | 0.026               | 1.23k | —              | Encoder:<br>6M<br>GPT-2:<br>124M<br>Decoder:<br>1M<br><br>Total:<br>131M |
| Signal detection                  | SER      | 0.213 | —     | 0.198       |                                                                                                       | 0.196               | 8.48k | 0.207          |                                                                          |
| MU-MISO precoding (data-driven)   | SE Ratio | —     | 100%  | 99.42%      |                                                                                                       | 99.51%              | 3.02k | 66.25%         |                                                                          |
| MU-MISO precoding (model-driven*) | SE Ratio | —     | 100%  | 99.95%      |                                                                                                       | 99.97%              | 1.23k | 99.93%         |                                                                          |
| D2D power control                 | SE Ratio | —     | 100%  | 99.69%      |                                                                                                       | 99.78%              | 1.23k | —              |                                                                          |
| MU-MIMO precoding                 | SE Ratio | —     | 100%  | 99.07%      |                                                                                                       | 99.31%              | 8.48k | —              |                                                                          |
| CB                                | SE Ratio | —     | 100%  | 100.21%     |                                                                                                       | 100.48%             | 8.48k | —              |                                                                          |
| Multi-cell power allocation       | SE Ratio | —     | 100%  | 99.13%      |                                                                                                       | 99.22%              | 1.23k | —              |                                                                          |
| Wideband MU-MISO precoding        | SE Ratio | —     | 100%  | 99.50%      |                                                                                                       | 99.62%              | 3.02k | —              |                                                                          |



## 融合通信知识AI的优势

小数据、小模型 → 强泛化、多任务 —— 并非空想

⇒ 降低大规模/高维问题的训练开销

机理与机制部分可解释

⇒ 预知设计效果、减少盲目尝试

## 融合通信知识AI的挑战

除了置换性质、如何在庞杂的知识中系统化地寻找其他先验？

如何系统化利用多种通信知识设计高效率更通用的智能空口？

机理发现 → 先验提取 → 结构映射：

通信机理驱动的AI是实现6G智能空口的关键路径！



## 参考文献

1. G. Karniadakis, I. Kevrekidis, L. Lu, et al., “Physics-informed machine learning,” Nature Reviews Physics, 2021.
2. B. Zhao, J. Guo, and C. Yang, “Understanding the performance of learning precoding policies with graph and convolutional neural networks,” IEEE Trans. Commun, 2024
3. N. Zhang and C. Yang, “A Decomposed learning framework for hybrid precoding,” IEEE Open J. of the Comm. Society, 2026
4. M. Eisen and A. Ribeiro, “Optimal wireless resource allocation with random edge graph neural networks,” IEEE Trans. Signal Process., 2020.
5. C. Sun, J. Wu, and C. Yang, “Improving learning efficiency for wireless resource allocation with symmetric prior,” IEEE Wireless Commun., 2022.
6. S. Liu, J. Guo, and C. Yang, “Multidimensional graph neural networks for wireless communications,” IEEE Trans. Wireless Commun., 2024.
7. J. Guo and C. Yang, “When attention is beneficial for learning wireless resource allocation efficiently?” [arxiv.org/ 2507.02427](https://arxiv.org/abs/2507.02427)
8. S. Liu, J. Guo, and C. Yang, “Optimizing hybrid precoding for multi-user MIMO systems with graph neural network,” IEEE GLOBECOM 2022
9. C. Sun and C. Yang, “Learning to optimize with unsupervised learning: Training deep neural networks for URLLC,” IEEE PIMRC 2019.
10. D. Liu, C. Sun, C. Yang, et al., “Optimizing wireless systems using unsupervised and reinforced- unsupervised deep learning,” IEEE Network, 2020



11. 吴佳骏、赵剑羽、孙乘坚，杨晨阳，“预测资源分配：马尔科夫决策过程的无监督学习,” 中国科学: 信息科学, 2024
12. J. Zhao, C. Yang, T. Liu, et al., “Designing heterogeneous GNNs with desired permutation properties for wireless resource allocation,” IEEE Open J. of the Commun. Society, 2025.
13. J. Guo, and C. Yang, “Learning power allocation for multi-cell-multi-user systems with heterogeneous graph neural network,” IEEE Trans. Wireless Commun, 2022.
14. J. Wu, C. Sun, and C. Yang, “Size generalization for resource allocation with graph neural networks,” Science of China, 2023.
15. J. Zhao and C. Yang, “Graph reinforcement learning with relational priors for predictive power allocation,” Science of China, 2025.
16. M. Zaheer, S. Kottur, S. Ravanbakhsh et al., “Deep sets,” NeuIPS, 2017.
17. S. Ravanbakhsh, J. Schneider, and B. Póczos, “Equivariance through parameter-sharing,” ICML, 2017.
18. Y. Duan, J. Guo, and C. Yang, “Learning precoding in multi-user multiantenna systems: Transformer or graph Transformer?” IEEE Trans. Wireless Commun, 2026.
19. Y. Duan and C. Yang, “Modular learning for multiple wireless policies with PE properties,” IEEE Open J. of the Commun. Society, 2026.
20. J. Guo and C. Yang, “A model-based GNN for learning precoding,” IEEE Trans. on Wireless Commun., 2024
21. L. Zhang, S. Han, and C. Yang, “Gradient-driven graph neural networks for learning digital and hybrid precoder,” IEEE Trans. on Commun., 2025
22. 张乃天，杨晨阳，“基于图神经网络的 DPC 预编码”，中国科学: 信息科学, to appear
23. J. Guo and C. Yang, “Cross-system neural precoder: exploiting structural consistency for fast adaptation,” IEEE GLOBECOM 2026
24. S. Liu and C. Yang, “Adversarial Training: Enhancing Out-of-Distribution Generalization for Learning Wireless Resource Allocation,” IEEE WCNC 2026



**Comments?**

**Questions?**

