5GNN: Extrapolating 5G Measurements through GNNs

Wei Ye[†], Xinyue Hu[†], Tian Liu[†], Ruoyu Sun[†], Yanhua Li[‡], Zhi-Li Zhang[†]
[†]University of Minnesota Twin Cities [‡]Worcester Polytechnic Institute
{ye000094, zhang089}@umn.edu yli15@wpi.edu

ABSTRACT

The advent of 5G networks has attracted a flurry of measurement studies to understand their performance in various settings. Unfortunately, carrying out an in-depth measurement study of 5G is both laborious and costly. The measurement samples cover only limited points in a (potentially large) coverage area of one or more 5G towers/base stations. In this paper, we tackle the following basic question: given a collection of 5G "signal" measurements collected in limited locations in a target 5G coverage area, can we infer or extrapolate 5G "signals" at other locations within the area that we do not have samples? We propose a novel learning paradigm based on graph neural networks (GNNs), dubbed *5GNN*, which captures both the "local" and "global" patterns of the underlying spatial correlation of 5G signals based on the measured data points. This paradigm is guided by insights from the physical characteristics of 5G networks. We conduct comprehensive experiments and evaluations using both synthetic and real-world datasets, which are collected and processed by ourselves with professional tools. Compared with baseline models using existing GNNs, *5GNN* is superior and can reduce the estimation errors for the signal imputation task and channel quality regression task by up to 12.8% and 9.2%, respectively.

CCS CONCEPTS

• **Networks** → **Network measurement**; **Wireless access points, base stations and infrastructure**; • **Computing methodologies** → **Learning paradigms**; **Learning settings**; **Machine learning algorithms**; **Neural networks**; • **Applied computing** → **Earth and atmospheric sciences**.

KEYWORDS

5G Network Measurement, Geospatial Imputation, Graph Convolutional Network

ACM Reference Format:
Wei Ye[†], Xinyue Hu[†], Tian Liu[†], Ruoyu Sun[†], Yanhua Li[‡], Zhi-Li Zhang[†]. 2022. *5GNN*: Extrapolating 5G Measurements through GNNs. In *Graph Neural Networking Workshop (GNNet '22), December 9, 2022, Roma, Italy.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3565473.3569186

1 INTRODUCTION

Emerging 5G networks are expected to usher in a plethora of new applications such as augmented/virtual/mixed reality, autonomous driving, Internet of Things (IoT), and digital twins that require either ultra-high bandwidth, ultra-low latency, or both. Unfortunately, 5G radio signals, especially, mmWave high band radio signals, are known to be highly sensitive to various environmental factors, as having been shown in recent measurement studies of commercial 5G networks [14, 15]. Understanding and predicting 5G performance dynamics are therefore crucial not only to manage and improve 5G networks, but also enable applications to better leverage the potentials of 5G services.

Measurement Challenges. An important way to understand 5G network performance is through detailed "in-the-field" testing and measurements. While commercial 5G operators may conduct "proprietary" measurement studies, the data collected is often kept private. Recently, a number of academic research teams have embarked on large-scale measurement studies of commercial 5G networks with publicly released datasets, notable examples include [4, 6, 8, 15, 17]. Besides gaining a deeper understanding of emerging 5G network performance, an important goal of such studies is to provide datasets to enable the development of machine learning algorithms for predicting 5G network performance (see, e.g., [8, 14]). While such "in-the-field" measurement studies are vitally crucial, they face a number of challenges. (1) The measurement process is labor-intensive and costly. Collecting one hour of valuable data requires more hours of data collection efforts. (2) Due to the inherent nature of measurement studies, data can only be collected in limited numbers of locations in a target (potentially large) geographical coverage area. (3) While the datasets thus collected are of enormous value to both the research community and industry, training machine learning models using the collected datasets alone can introduce biases, due to the limited coverage.

5G Measurement Extrapolation Problem and Our Solution – *5GNN*. We apply machine learning to tackle the following *5G measurement extrapolation* problem: Given a collection of 5G "signal" measurements collected in limited locations in a target 5G coverage area, can we infer or extrapolate 5G "signals" at other locations within the area that we do not have samples? Here we use the term "signals" as a generic term to refer to any 5G network metrics of interest (not merely *actual* radio signals *per se*). We formulate this problem formally in §2. We propose a novel paradigm based on graph neural networks (GNNs), dubbed *5GNN – extrapolating 5G measurements through GNN*. While our paradigm shares the basic spirit of GNNs in that it takes advantage of spatial correlations, *5GNN* augments existing GNN models by explicitly taking into account *local* and *global* spatial patterns of 5G signals. This design is based on insights from the physical characteristics of 5G networks. Take mmWave high band 5G radio as an example. There are strong "local" factors such as local signal waveform variations, noise, and

GNNet '22, December 9, 2022, Roma, Italy
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9933-3/22/12…$15.00
https://doi.org/10.1145/3565473.3569186

Table 1: Key Statistics of Real-World Datasets

Scenario	Public Square
Total area covered	8,000 m^2
Technologies	4G-lowBand/midBand; 5G-highBand
Data samples	Total 200k+ with 100ms sampling rate
Tasks	Signal strength imputation Channel quality regression

interference; there are also strong "global" phenomena such as radio signal propagation, attenuation, fading, and path losses often vary drastically along different paths, with manifest shapes and patterns (cf. Fig.4 & Fig.5).

We conduct extensive evaluations on both synthetic datasets using DeepMIMO [19] and real-world datasets via our measurement campaigns of commercial 5G networks. Table 1 lists the key statistics of the collected datasets (see more details in §4.1 and §4.2). The evaluation results show that *5GNN* consistently outperforms existing GNN-based models (see §4.4).

We summarize our key contributions below:

• To the best of our knowledge, we are the first to explicitly address the 5G measurement extrapolation problem, and argue for the need to account for both local and global dependencies in 5G signal and feature maps.

• We propose a new learning paradigm, *5GNN*, which captures both the local and global spatial patterns. The comprehensive experiments involve six other state-of-art methods and are conducted on both synthetic and real-world datasets, where our *5GNN* reduces the error rates (up to) 12.8% and 9.2% on the signal imputation task and channel quality regression task, respectively.

• We conduct field experiments to collect the commercial 5G network data for this study. *We make our code and the unique 5G dataset publicly available to contribute to the research community* [1].

2 OVERVIEW

We first formulate the 5G measurement extrapolation problem. We then briefly discuss the limitations of existing GNN models when applied to 5G measurements.

Consider a 5G coverage area of interest. Let y denote the 5G signal, e.g., 5G signal strength or CQI (channel quality index), that we are interested in capturing and predicting, which is a function Φ of a set of "factors" or feature vectors X that we can collect or measure, namely, $y = \Phi(X)$. Geometrically, we can view the 5G signals y as a (hypersurface) function Φ defined on a manifold $\mathcal{M}$ (see Fig. 1). We want to learn Φ by sampling N points in the target 5G coverage area. The samples dataset is denoted by $\mathcal{D}^{obs} = \{p_i^{obs} := (c_i, x_i, y_i) \in \mathcal{M}\}_{i=0}^{N}$, where c_i is the geographical coordinates of location i, and x_i and y_i correspond to the feature vector and 5G signal at location i. We aim to approximate the 5G signal function Φ defined on the manifold $\mathcal{M}$ via a neural network f_θ using the sampled dataset. We refer to this learning problem as the *5G measurement extrapolation problem*: Once we have learned Φ, we can extrapolate and predict the 5G signal y_j at any other location j.

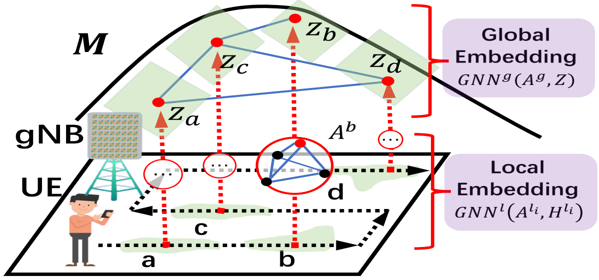

M
z_b
z_c
z_d
z_a
gNB
A^b
UE
d
c
a
b
Global Embedding $GNN^g(A^g, Z)$
Local Embedding $GNN^l(A^{l_i}, H^{l_i})$

Figure 1: Overview of 5GNN.

Limitations of Existing Graph Neural Networks (GNNs). As pointed out in the introduction, 5G signals have strong (yet likely irregular) spatial dependencies. Hence it is natural to consider GNNs. "Standard" GNNs assume that an underlying graph (or an ensemble of graphs) is given, and each node in the graph is associated with a feature vector and a node signal ("label"). Message-passing is invoked at each layer to train a GNN model for either a *node classification/regression* task or *graph classification/regression* task. In our 5G measurement extrapolation problem, while each sample location is a point in an underlying geographical area, the "edges" are not explicitly given. The conventional approach in dealing with this issue is to either directly use the geographical proximity or use the node feature vectors to construct a k-nearest-neighbor (kNN) graph. For example, the authors in [2] first employ a kernel function to construct a sequence of "local" kNN graphs based on the coordinate distance and then combine GNN and kriging to perform graph regression to learn and predict node labels. In contrast, the authors in [12] first iteratively and randomly sample a set of data points in the training dataset to construct a "global" kNN graph and then apply a GNN model for node label prediction as a node regression task. These approaches suffer the limitations that they cannot effectively learn either the global dependence or local patterns that are inherent in our 5G measurement extrapolation problem.

Our Solution. We borrow ideas from differential geometry, where a manifold is defined and constructed by "patching together" a collection of local (coordinate) charts. Therefore, we learn the 5G signal manifold by first (i) constructing "local charts" that best approximate the 5G signals in each local neighborhood $Nbr(i)$ of the sampled data points by adapting to the local variations, patterns, and "smoothness" properties in the 5G feature vectors and signals; and then (ii) patching together the "local charts" into a global manifold by taking into account the global dependencies and shapes in the 5G signal manifold, see Fig. 1 for an illustration. A sequence of local graphs and a global graph (defined on the local charts) are learned and constructed separately but trained jointly, as expounded in the next section.

3 5GNN: 5G GRAPH NEURAL NETWORK

We now present our proposed **5G Graph Neural Network (*5GNN*)** paradigm for the 5G measurement extrapolation problem, which is schematically depicted in Fig. 1. It operates in three stages: 1) *local embedding*, 2) *global embedding* and 3) *joint graph neural network training*, as will be discussed in more details below. In both stage 1) and stage 2), we apply a GNN model such as GCN [11], GIN [25], GraphSAGE [7] for local and global embedding. Such a GNN model

[1] The datasets and codes are hosted on https://github.com/StrongWeiUMN/5GNN

is referred to as a baseline. In other words, our paradigm can work with any existing GNN model, but augment it by *separately constructing but jointly learning local and global embeddings.*

Stage 1: Local embedding. In the first stage we aim to learn a local ("smooth") embedding of data points lying within a local neighborhood by capturing the local variation and patterns in the local 5G feature map and signal. Formally, for a target location p_i referred to as a center node, we sample the k-nearest neighboring locations/nodes from the training dataset D^{obs}, i.e., $\{p_j\}^K_{j \in Nbr(i)}$. We further convert the local set into a (local) graph using the following kernel function:

$$A^l_{jk} = exp(-\frac{d(c_j, c_k)}{2\sigma^2}) \tag{1}$$

$$H^{l_i} = \begin{bmatrix} c_i & 0 & 0 \\ c_j & X_j & y_j \\ ... & ... & ... \end{bmatrix} \begin{matrix} \leftarrow \quad \text{center node} \\ \leftarrow \quad \text{neighbor node} \\ \end{matrix} \tag{2}$$

where $A^{l_i} \in R^{(K+1)\times(K+1)}$ and H^{l_i} are the adjacency matrix and feature matrix corresponding to the center node p_i, respectively. The distance function, $d(\cdot, \cdot)$, calculates the distance of each node pair, and σ is the length scale of the kernel. We apply a GNN model (denoted as GNN^l) on the local set corresponding to the center node p_i to learn a local embedding (local chart):

$$z_i = F(GNN^l(A^{l_i}, H^{l_i})) \tag{3}$$

where the $F(\cdot)$ denotes the flatten function, which flattens the matrix into a vector. This process is repeated for each center node.

Stage 2: Global embedding. Given the local embeddings (local charts) constructed in Stage 1, we aim to approximate the (global) 5G signal manifold by patching together the local charts in an appropriate manner. This is done by taking into account the global dependencies and shapes of the 5G signal manifold, e.g., radio signals propagate and attenuate along certain directions or sectors in the space that are shaped by where 5G towers concentrate their transmitting power as well as the confluent effects of the objects in the environment that reflect, refract or absorb radio waves. More specifically, we build a global kNN graph A^g over all the local charts, and apply a GNN model (denoted as GNN^g) to learn the global embedding:

$$\hat{Z} = GNN^g(A^g, Z) \tag{4}$$

$$\boldsymbol{y} = MLP([\hat{Z}||Z]) \tag{5}$$

where $Z = [z_0, z_1, ..., z_N]^T$ is the feature matrix of the global graph, and GNN is supposed to learn the correlation among local charts. The operator "||" means concatenating, which represents the skip connection of neural networks.

Stage 3: Joint Local and Global GNN Training. We jointly train the local and global GNN models constructed in Stage 1 and Stage 2 to learn the local and global embedding simultaneously in an iterative fashion. The training process is summarized in Algorithm 1. For the inference, we will use all the data points, i,e, set batch size $B = N_{obs} + N_{unknown}$. The backbone GNN can be any off-the-shelf model. Moreover, the neighbor sampling and graph construction can be done in the data pre-processing stage or pre-fetched by the CPU, parallel with the GPU's calculation.

Algorithm 1 *5GNN*

```
Input: Rounds $r = \{1, \ldots, R\}$, Batch size $B$, Learning rates $\eta$,
  Number of neighbors $K$, Measured dataset $\mathcal{D}^{obs}$
Initialize: Parameters $\theta$
for $r = 1$ to $R$ do
    Sample a batch of points $\{p_i; p_i \sim \mathcal{D}^{obs}\}^B_{i=1}$
    for each $p_i$ in the batch do
        Sample the $k$ neighbors of $p_i$ from $\mathcal{D}^{obs}$
        Construct local graph $A^{l_i}$ and $H^{l_i}$          (3-4)
        Compute the local embedding $z_i$                       (5)
    end for
    Construct kNN graph over the batch $\{p_i\}^B_{i=1}$
    Predict the label $\boldsymbol{y}_{pred} = [y_1, ..., y_B]$          (6-7)
    Compute the loss gradient $\nabla\mathcal{L}(\boldsymbol{y}_{pred}, \boldsymbol{y}_{true})$          (1)
    Update the parameters: $\theta \leftarrow \theta - \eta * \nabla\mathcal{L}$          (2)
end for
Return: $\theta$
```

4 EXPERIMENTS AND EVALUATIONS

We first provide more details about our measurement campaigns and the datasets in this section. Then we describe the baseline models, introduce the experiment setup, and finally discuss results.

4.1 Measurement Campaigns

We describe our measurement campaigns through the measurement location, operators, and methodology as below.

Location and Operators: We conduct comprehensive measurement campaigns to collect the signal and radio channel dataset in a public square near a large football stadium in downtown Minneapolis, as shown in Fig. 2. Due to the potential high demand for mobile networking, the mobile operator Verizon deploys diverse radio bands and multiple base stations in this area, including 5G-highBand (band-n261-28GHz) [2], 4G-midBand (band-n66-2.1GHz), and 4G-lowBand (band-n13-0.7GHz).

Methodology: The setup of measurement tools is shown in Fig. 3. We take the leading flagship smartphones (Samsung Galaxy S21 Ultra 5G) as our user-equipments (UE) and tether them to the laptop by USB cables. To access the Qualcomm chipset Diag and physical layer information, we adopt a professional software tool named Accuver XCAL [1], running on the laptop and UEs. We also set up an external GPS to retrieve precise geo-locations and a power bank for sustainable power supply. We equip those devices when we are walking to collect radio information.

To ensure the UEs (especially their 5G data plane) are always in the activated mode for the accurate and continuous data collection, we let UEs send the User Datagram Protocol (UDP) data packages to our university server at 5 Mbps via the commercial cellular network.

4.2 Datasets

Following the above descriptions, we collect the commercial 5G signal/channel dataset. Besides, we simulate the ideal signal data through DeepMIMO [19] to complement the real-world one. The details of datasets are described as follows.

[2] The 5G high band is also known as mmWave in literature. For simplicity and consistency, we use the former one below.

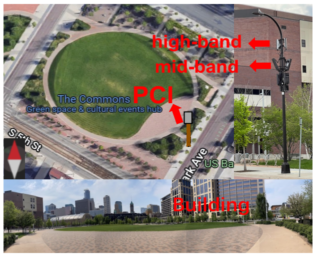


Figure 2: Meas. campaign.

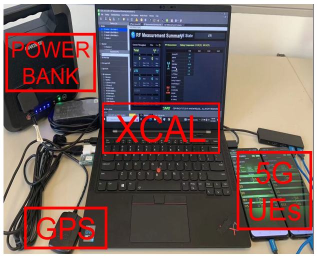


Figure 3: Meas. tools.

Table 2: Fields of signal strength imputation task.

c	Longitude; Latitude Angle: the orientation angle of UE
y	BRSRP: filtered beam reference signal received power [dBm]

Table 3: Fields of channel quality regression task.

c	Longitude; Latitude Angle: the orientation angle of UE
X	ss-RSRP: synchronize signal reference signal received power [dBm] ss-RSRQ: synchronize signal reference signal received quality [dB] csi-RSRP [SSBRI]: CSI reference signal received power [dBm] csi-RSRP [CRI]: CSI reference signal received power [dBm] SINR: signal to interference & noise ratio [dB] Pathloss: reduction in power density as signal propagates [dB] BLER: block error rate [%]
y	CQI: channel quality indicator

Real-world datasets: We extract raw data from the XCAL database with a 100ms sampling rate, filter out the outliers, and aggregate the different routes based on the coordinates. Finally, we get a total of 200k+ valid data records, including three different bands mentioned above. For the deep learning tasks (i.e., signal strength imputation task and channel quality regression task), the features and label are summarized in the Table 2 and Table 3, respectively.

Simulated datasets: We use DeepMIMO simulator [19], which adopts the predefined channel model by 3GPP, to generate the ideal signal/channel dataset. Specifically, we generate datasets under the configuration of a similar environment to our public square (i.e., outdoor scenario) with different frequencies (i.e., 3.4GHz and 28GHz) and name them as DeepMIMO_Mid and DeepMIMO_High.

4.3 Evaluation Setup

We introduce the baseline models, implementation, and hyper-parameter settings below.

Model Comparison: We evaluate and compare our proposed *5GNN* paradigm with several state-of-art machine learning models. For the baseline GNN models, we consider three representative GNN models (GCN [11], GraphSAGE [7], and GIN [25]) that have been widely used for graph learning tasks. As stated in Section 2, unlike conventional graph learning problems, the "graphs" in the 5G measurement extrapolation problem are not explicitly specified, but must be "inferred" or constructed by mining the spatial correlations in the underlying datasets. For this, we consider two state-of-the-art approaches that are developed for "geography inference" problems and discussed in Section 2: 1) the KCN (local) graph regression approach developed in [2], and 2) the PEGNN node regression approach developed in [12]. Similar to our *5GNN* paradigm, these two approaches can also be combined with various GNN models. We denote them as P1 ("Paradigm 1") and P2 ("Paradigm 2") respectively in Table 2 and Table 3.

In addition to above (augmented) GNN-based models, we also consider the Universal Kriging (UK), a widely-used classical statistics-based method, also known as the Gaussian process [20, 21].

Hyperparameter settings: For all experiments, we normalize the dataset using the min-max scaler and split it into the training-validation-test set with a $0.25 : 0.25 : 0.50$ ratio. The models are trained using the Adam optimizer [10] with its default configuration and a batch size of 128. We set 2 hidden layers with 128 hidden units and set the neighbor size as 10 for all the baselines. We also adopt the normalizing layers to avoid the data distribution shift and speed up convergence. We ensure each model is sufficiently trained by setting a big epoch number of 250 and saving the model with the best performance on the validation set. Lastly, we repeat all experiments three times and report the best results.

Implementation: The whole project involves about 5,000 lines of Python codes, including data processing and machine learning model modules. The UK model uses PyKrige library [13], and all of the GNN-based models are built based on PyTorch Geometry library [5]. All data processing and numerical experiments are run on our workstation, equipping AMD Ryzen Threadripper PRO 3995WX CPU and three NVIDIA RTX A6000 GPUs.

4.4 Imputation and Regression Tasks

Table 4 and Table 5 report the results of signal strength imputation task and channel quality regression task, respectively.

Compared with the other graph-based learning paradigms, *5GNN* is consistently superior and reduces errors up to 12.8% on the imputation task and 9.2% on the regression task. The improvements on the commercial 5G dataset with GCN and GIN models are the most significant. It may be because (1) 5G high band has salient local and global spatial characteristics, as discussed in Sec. 2. And our *5GNN* can explicitly consider them. (2) GraphSAGE's training strategy involves some extra built-in sampling steps, which may increase the difficulty for *5GNN* to construct the underlying global manifold. Due to the page limitation, we leave the question of how the learning paradigm affects different baselines to future work.

Meanwhile, *5GNN* also outperforms the Kriging methods on the imputation task, a widely believed hard task since only coordinates are involved for prediction [2, 12]. It is worth noting that *5GNN* reduces up to 25.3% error rates on the 5G high band, while the other graph-based learning methods fail to consistently beat the statistical-based Kriging methods.

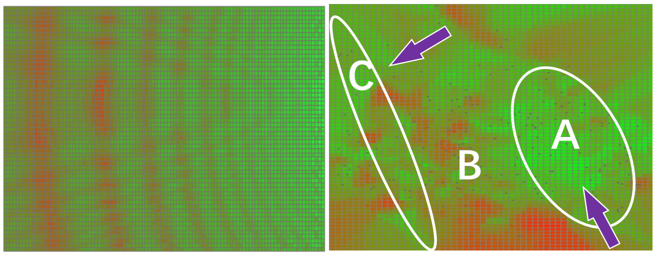
Overall, those results validate that *5GNN* is a better choice than other state-of-art methods in signal imputation and channel quality regression tasks by efficiently capturing the local and global spatial correlations (e.g., interference, signal shape, and attenuation) of the radio signals.

Table 4: Results of signal imputation task. We use RMSE and MAE metrics for evaluations. For each GNN baseline, We compare the three paradigms and report the results in a tuple (P1, P2, *5GNN*). In this table, as well as in Table 3, for each experimental setting/dataset, the best results obtained using a specific GNN baseline are underlined. The best results across all GNN baseline models and paradigms are emphasized in bold.

Datasets	Metrics	UK	GCN			GraphSAGE			GIN		
			P1	P2	5GNN	P1	P2	5GNN	P1	P2	5GNN
DeepMIMO_Mid	RMSE	0.0465	0.0584	0.0451	0.0440	0.0559	0.0438	0.0437	0.0531	0.0458	**0.0436**
	MAE	0.0342	0.0444	0.0334	0.0320	0.0423	**0.0315**	**0.0315**	0.0410	0.0344	0.0316
DeepMIMO_High	RMSE	0.0767	0.0722	0.0720	0.0701	0.0723	0.0696	**0.0690**	0.0722	0.0719	0.0703
	MAE	0.0584	0.0552	0.0550	0.0535	0.0552	0.0524	**0.0519**	0.0555	0.0545	0.0537
4G_Signal_Low	RMSE	0.1250	0.1216	0.1129	0.1076	0.1039	**0.1018**	0.1020	0.1223	0.1130	0.1071
	MAE	0.0915	0.0955	0.0864	0.0784	0.0771	0.0758	**0.0745**	0.0964	0.0874	0.0787
4G_Signal_Mid	RMSE	0.0899	0.0943	0.0849	0.0806	0.0887	0.0796	**0.0795**	0.0939	0.0851	0.0816
	MAE	0.0684	0.0740	0.0662	0.0616	0.0688	**0.0603**	**0.0603**	0.0735	0.0663	0.0622
5G_Signal_High	RMSE	0.1588	0.1598	0.1453	0.1366	0.1218	0.1194	**0.1187**	0.1574	0.1457	0.1361
	MAE	0.1201	0.1297	0.1156	0.1015	0.0889	0.0863	**0.0855**	0.1260	0.1157	0.1009

Table 5: Results of channel quality regression task.

Datasets	Metrics	UK	GCN			GraphSAGE			GIN		
			P1	P2	5GNN	P1	P2	5GNN	P1	P2	5GNN
4G_CQI_Low	RMSE	0.1840	0.1813	0.1710	0.1611	0.1754	**0.1594**	0.1605	0.1809	0.1712	0.1602
	MAE	0.1430	0.1460	0.1365	0.1259	0.1416	**0.1234**	0.1252	0.1472	0.1370	0.1247
4G_CQI_Mid	RMSE	0.1435	0.1336	0.1328	0.1328	0.1310	0.1278	**0.1276**	0.1345	0.1364	0.1329
	MAE	0.1092	0.1042	0.1053	0.1017	0.1029	0.0982	**0.0972**	0.1045	0.1076	0.1015
5G_CQI_High	RMSE	0.1926	0.1730	0.1724	0.1643	0.1751	0.1645	**0.1629**	0.1748	0.1726	0.1638
	MAE	0.1516	0.1455	0.1417	0.1287	0.1469	**0.1279**	**0.1279**	0.1493	0.1422	0.1292



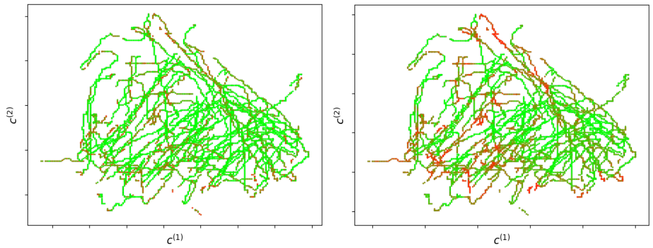
(a) DeepMIMO_Mid. **(b) 5G mmWave.**

Figure 4: Visualized results of the signal imputation.

4.5 Visualization of Radio Maps

To better understand *5GNN*, we visualize the results of the above two tasks, as shown in Fig. 4 and Fig. 5.

Fig. 4a shows the reconstructed DeepMIMO scenario, where we can see that *5GNN* learns the local and global spatial patterns. Fig. 4b is the result of signal strength imputation for the entire public square based on the measured data (small black dots in the figure), where zone A and some parts of zone C have good signal coverage. According to the environment description in Fig. 2, it is because zone A is the line of sight area, and those parts of zone C enjoy signal reflection by the buildings. Fig. 5 shows the result of CQI regression. We can see the predicted results are close to the ground truth.



(a) Ground truth. **(b) Predicted results.**

Figure 5: Visualized results of CQI regression.

In summary, we can conclude that the proposed method can generate the radio map efficiently based on the few measured data points, thus assisting future measurements.

4.6 Discussions on Running Time

Now we discuss the algorithm's running time to demonstrate the feasibility of applying *5GNN* to assist in real-time measurement. For simplicity, we take the 5G_Signal_High dataset and GCN baseline as an example. For comparison fairness [3], we use 64 vCPU cores for UK, but 8 vCPU cores with 1 GPU for other GCN-based methods. The results are reported in Table 6, where we mark three learning paradigms as GCN-P1, GCN-P2, and GCN-5GNN as above.

[3] We consider the computing resources that can be purchased from AWS [3] with almost the same amount of money.

Table 6: Results of running time. For GCN-based methods, we report (1) average running time of one epoch and (2) the number of epochs for training to convergence.

	Training Time		Inference Time
UK	182.4 s		0.135 ms/sample
GCN-P1	0.39 s/epoch	45 epochs	0.006 ms/sample
GCN-P2	0.18 s/epoch	42 epochs	0.178 ms/sample
GCN-5GNN	0.29 s/epoch	46 epochs	0.183 ms/sample

We can observe that different methods have different running time characteristics. The *5GNN* is competitive and can satisfy real-time demands, considering each measurement run usually takes tens of minutes. We leave further algorithm considerations (e.g., incremental learning, transfer learning, etc) and integrate *5GNN* (as an API) into the measurement system to future work.

5 RELATED WORK

We briefly discuss and contrast our work with related works.

Radio Propagation Modeling: There has been a large literature on applying machine learning to radio channel propagation modeling (see, e.g., [23, 24]). While they often take into account the specific physical characteristics of radio bands and channels (which provide value insights to our work), the goal of such studies differ from ours. They also make various assumptions. e.g., knowledge of tower location and transmission power, which are often not available when conducting measurements. In contrast, *5GNN* is a tower-information-free, data-driven, and learning-based approach.

Spatial Imputation and Graph Neural Networks: Machine learning methods, especially, GNNs, have been applied to various "geography-related" problems [2, 12, 22] such as weather forecasting, road traffic prediction where "spatial imputation" is employed [2, 16]. As discussed in §2, they often do not explicitly take into account local vs. global factors and patterns inherent in the problem domain and data.

In addition to the above papers, we also recommend two surveys [9, 18], which elaborate on other opportunities for applying GNN to wireless communication networks.

6 CONCLUSION

To the best of our knowledge, this is the first work that explicitly addresses the 5G measurement extrapolation problem to improve measurement efficiency. We advocate an AI-assisted approach for scaling 5G network measurements and propose *5GNN*. Inspired by ideas from differential geometry, 5GNN augments GNNs by explicitly accounting for local and global factors and patterns in 5G signal maps. Evaluation results using both synthetic and real-world 5G measurement datasets show 5GNN outperforms existing state-of-the-art models. We believe that the general paradigm of 5GNN is also applicable to many other problem domains.

ACKNOWLEDGMENT

This research was in part supported by NSF under Grants CNS-1901103, CNS-2128489, CCF-2212318, CNS-222029, IIS-1942680 (CAREER), CNS-1952085, CMMI1831140, and DGE-2021871 as well as an unrestricted InterDigital Gift.

REFERENCES

[1] Accuver. 2021. XCAL. https://www.accuver.com/sub/products/view.php?idx=6&ckattempt=2. Accessed: 2022-05-09.

[2] Gabriel Appleby, Linfeng Liu, and Li-Ping Liu. 2020. Kriging convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 3187–3194.

[3] AWS. 2022. AWS price. https://aws.amazon.com/ec2/pricing/on-demand/. Accessed: 2022-10-25.

[4] Phuc Dinh, Moinak Ghoshal, Dimitrios Koutsonikolas, and Joerg Widmer. 2022. Demystifying Resource Allocation Policies in Operational 5G mmWave Networks. In *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 1–10.

[5] Matthias Fey and Jan E. Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

[6] Moinak Ghoshal, Z Jonny Kong, Qiang Xu, Zixiao Lu, Shivang Aggarwal, Imran Khan, Yuanjie Li, Y Charlie Hu, and Dimitrios Koutsonikolas. 2022. An in-depth study of uplink performance of 5G mmWave networks. In *Proceedings of the ACM SIGCOMM Workshop on 5G and Beyond Network Measurements, Modeling, and Use Cases*. 29–35.

[7] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[8] Ahmad Hassan, Arvind Narayanan, Anlan Zhang, Wei Ye, Ruiyang Zhu, Shuowei Jin, Jason Carpenter, Z Morley Mao, Feng Qian, and Zhi-Li Zhang. 2022. Vivisecting mobility management in 5G cellular networks. In *Proceedings of the ACM SIGCOMM 2022 Conference*. 86–100.

[9] Shiwen He, Shaowen Xiong, Yeyu Ou, Jian Zhang, Jiaheng Wang, Yongming Huang, and Yaoxue Zhang. 2021. An overview on the application of graph neural networks in wireless networks. *IEEE Open Journal of the Communications Society* (2021).

[10] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[11] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[12] Konstantin Klemmer, Nathan Safir, and Daniel B Neill. 2021. Positional Encoder Graph Neural Networks for Geographic Data. *arXiv preprint arXiv:2111.10144* (2021).

[13] Benjamin S Murphy. 2014. PyKrige: development of a kriging toolkit for Python. In *AGU fall meeting abstracts*, Vol. 2014. H51K–0753.

[14] Arvind Narayanan, Eman Ramadan, Rishabh Mehta, Xinyue Hu, Qingxu Liu, Rostand AK Fezeu, Udhaya Kumar Dayalan, Saurabh Verma, Peiqi Ji, Tao Li, et al. 2020. Lumos5G: Mapping and predicting commercial mmWave 5G throughput. In *Proceedings of the ACM Internet Measurement Conference*. 176–193.

[15] Arvind Narayanan, Muhammad Iqbal Rochman, Ahmad Hassan, et al. 2022. A Comparative Measurement Study of Commercial 5G mmWave Deployments. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 800–809.

[16] Yun Qin, Guoyu Ren, Panfeng Zhang, Lixiu Wu, and Kangmin Wen. 2021. An imputation method for the climatic data with strong seasonality and spatial correlation. *Theoretical and Applied Climatology* 144 (04 2021). https://doi.org/10.1007/s00704-021-03537-9

[17] Darijo Raca, Dylan Leahy, and others J. 2020. Beyond throughput, the next generation: a 5g dataset with channel and context metrics. In *Proceedings of the 11th ACM multimedia systems conference*. 303–308.

[18] José Suárez-Varela, Paul Almasan, Miquel Ferriol-Galmés, Krzysztof Rusek, Fabien Geyer, Xiangle Cheng, Xiang Shi, Shihan Xiao, Franco Scarselli, Albert Cabellos-Aparicio, et al. 2022. Graph Neural Networks for Communication Networks: Context, Use Cases and Opportunities. *IEEE Network* (2022).

[19] Ahmed Alkhateeb Umut Demirhan, Abdelrahman Taha. 2021. DeepMIMO. https://deepmimo.net. Accessed: 2022-05-09.

[20] Wikipedia. 2022. Kriging. https://en.wikipedia.org/wiki/Kriging. Accessed: 2022-10-25.

[21] Wikipedia. 2022. Regression Kriging. https://en.wikipedia.org/wiki/Regression-kriging. Accessed: 2022-10-25.

[22] Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. 2020. Inductive Graph Neural Networks for Spatiotemporal Kriging. https://doi.org/10.48550/ARXIV.2006.07527

[23] William Xia, Sundeep Rangan, Marco Mezzavilla, Angel Lozano, Giovanni Geraci, Vasilii Semkin, and Giuseppe Loianno. 2020. Millimeter wave channel modeling via generative neural networks. In *2020 IEEE Globecom Workshops (GC Wkshps*. IEEE, 1–6.

[24] Yunchou Xing and Theodore S Rappaport. 2021. Millimeter Wave and Terahertz Urban Microcell Propagation Measurements and Models. *IEEE Communications Letters* 25, 12 (2021), 3755–3759.

[25] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826* (2018).