Doctoral Thesis in Computer Science

Generalizable Representation for Wireless Networks Optimization through Native Graph Topology

YIFEI JIN

KTH ROYAL INSTITUTE OF TECHNOLOGY

Generalizable Representation for Wireless Networks Optimization through Native Graph Topology

YIFEI JIN

Academic Dissertation which, with due permission of the KTH Royal Institute of Technology, is submitted for public defence for the Degree of Doctor of Philosophy on Tuesday the 16th December 2025 at 1:15 pm. in Kollegiesalen, Brinellvägen 8, Stockholm.

Doctoral Thesis in Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden 2025

TRITA-EECS-AVL-2025:88
ISBN 978-91-8106-418-6

Printed by: Universitetsservice US-AB, Sweden 2025

Abstract

Graph representation learning has become a powerful paradigm for modeling structured data, enabling machine learning systems to reason over relationships, spatial dependencies, and topological patterns. However, its potential in wireless networks remains underexplored, particularly in learning native representations of complex and dynamic wireless environments. This thesis addresses the challenge of applying graph representation learning (such as graph neural networks and transformer architectures) to wireless systems, where topology, domain heuristics, and physical constraints critically impact optimization performance and generalization.

The core problem investigated is how to construct and exploit graph representations that faithfully encode the native structure of wireless networks to enable scalable, topology-aware optimization. This includes coverage relations, interference patterns, and environment-specific propagation effects. Existing solutions in wireless machine learning often overlook these structural priors, resulting in brittle models that generalize poorly across different network deployments.

This thesis introduces a graph-centric methodology to bridge this gap. By representing wireless elements—such as base stations, links, and coverage zones as nodes and their interactions as graph edges, we develop learning architectures that integrate attention mechanisms, domain-aware features, and physics-inspired constraints. Four studies demonstrate the approach across key tasks: routing latency prediction, antenna tilt optimization, real-time radio coverage estimation, and neural ray tracing for link-level modeling.

Our results suggest that these graph-based models significantly outperform traditional baselines, achieving near-simulator accuracy with improved generalization across unseen topologies and user scenarios. They also uncover a correspondence between engineering practices and graph spectral properties, offering a new lens for understanding network design. The proposed methods reduce supervision needs and support scalable deployment across variable network configurations.

Overall, this thesis establishes graph representation learning as a foundational tool for wireless intelligence, enabling structure-informed, optimization-driven modeling across diverse network conditions. These advances pave the way towards future wireless foundation models capable of supporting a wide range of optimization, sensing, and decision-making tasks with minimal retraining.

Keywords

Graph neural network, Wireless network, Representation learning, 5G & B5G, Digital twins

Sammanfattning

Inlärning av grafrepresentation har blivit ett kraftfullt paradigm för modellering av strukturerad data, vilket gör det möjligt för maskininlärningssystem att resonera kring relationer, rumsliga beroenden och topologiska mönster. Dess potential inom trådlösa nätverk är dock fortfarande underutforskad, särskilt när det gäller att lära sig nativa representationer av komplexa och dynamiska trådlösa miljöer. Denna avhandling tar upp utmaningen att tillämpa inlärning av grafrepresentation – såsom grafiska neurala nätverk och transformatorarkitekturer – på trådlösa system, där topologi, domänheuristik och fysiska begränsningar kritiskt påverkar optimeringsprestanda och generalisering.

Det centrala problemet som undersöks är hur man konstruerar och utnyttjar grafrepresentationer som troget kodar den nativa strukturen i trådlösa nätverk för att möjliggöra skalbar, topologimedveten optimering. Detta inkluderar optimering av täckningsrelationer, interferensmönster och miljöspecifika utbredningseffekter. Befintliga lösningar inom trådlös maskininlärning förbiser ofta dessa strukturella förutsättningar, vilket resulterar i sköra modeller som generaliserar dåligt över implementeringar och erbjuder begränsad återanvändbarhet.

Denna avhandling introducerar en grafcentrerad metod för att överbrygga detta gap. Genom att representera trådlösa element – såsom basstationer, länkar och täckningszoner – som noder, och deras interaktioner som grafkanter, utvecklar vi inlärningsarkitekturer som integrerar uppmärksamhetsmekanismer, domänmedvetna funktioner och fysikinspirerade begränsningar. Fyra studier demonstrerar denna metod för viktiga trådlösa optimeringsuppgifter: förutsägelse av routinglatens, antennlutningskonfiguration, realtidsuppskattning av radiotäckning och neural strålspårning för modellering på länknivå.

Våra resultat visar att dessa grafbaserade modeller avsevärt överträffar traditionella baslinjer och uppnår nästan simulatornoggrannhet med förbättrad generalisering över osynliga topologier och användarscenarier. De avslöjar också korrespondenser mellan tekniska designmönster och grafspektralegenskaper, vilket erbjuder en ny lins för att förstå och optimera nätverksbeteende. De föreslagna metoderna minskar övervakningsbehovet och stöder skalbar distribution över variabla nätverkskonfigurationer.

Sammantaget etablerar denna avhandling grafrepresentationsinlärning som ett grundläggande verktyg för trådlös intelligens – vilket möjliggör strukturinformerad, optimeringsdriven modellering över olika nätverksförhållanden. Dessa framsteg banar väg för framtida trådlösa grundmodeller som kan stödja ett brett spektrum av uppgifter inom optimering, avkänning och beslutsfattande med minimal omskolning.

Nyckelord

Grafiskt neuralt nätverk, Trådlöst nätverk, Representationsinlärning, 5G & B5G, Digitala tvillingar

Acknowledgments

As I conclude this academic journey, I feel a sense of gratitude for the work that all my knowledgeable supervisors, talented co-author, and supportive peers and families have done, which has been more than necessary, both for the journey we have co-conquered and for the irreplaceable role you all have played. We have explored the representation learning with graph modality, which has been distinct yet under-investigated, in combination with wireless network systems. Such a case will not prevail, thanks to every individual who helped, contributed, and paved the way together throughout the process.

First and foremost, I would like to express my deepest gratitude to my main supervisor, Prof. Aristides Gionis. He was one of the earliest and most supportive advocates of this ambitious endeavor. The fundamental insight—that effective learning in networked systems begins with a well-defined network representation—has guided and shaped all the work presented in this thesis. Aris has consistently demonstrated patience, professionalism, and scholarly rigor, providing thoughtful and constructive guidance throughout my research journey. Beyond technical mentorship, Aris consistently encouraged academic collaboration and supported my research visits. His guidance fostered connections that greatly helped me grow as an independent researcher.

I would also like to express my sincere gratitude to my assistant supervisor, Prof. Šarūnas Girdzijauskas, who consistently offered thoughtful, constructive, and critical feedback throughout my PhD journey. His sharp perspective challenged my assumptions, deepened the maturity of my research questions, and significantly strengthened the clarity and rigor of my work. Šarūnas was always generous with his time, especially during intense rounds of paper revisions and submissions. Beyond research, he actively encouraged me to engage with the wider academic community through public talks and outreach activities, helping me grow in confidence and gain visibility.

I am deeply grateful to Dr. Marios Daoutis, who served as my industrial supervisor from 2020 to 2024. His unwavering dedication and tireless support across numerous projects played a pivotal role in shaping this dissertation. Marios consistently provided detailed technical feedback, strategic guidance, and mentorship, both academically and beyond. Many of the contributions in this thesis would not have been possible without his involvement. Although he had to step away from this role before my graduation due to health reasons, his impact on my work and growth remains profound and will always be remembered.

Dr. Aneta Vulgarakis Feljan and Dr. Athanasios Karapantelakis formally stepped into supervisory roles in 2024. Still, their influence on my research extends back to 2017 through our collaboration at Ericsson Research AI. Their guidance has been instrumental not only in shaping the core contributions of this thesis but also in driving research visibility, fostering application beyond innovation, and navigating the patenting process. I am also grateful to Dr. Konstantinos Vandikas, Dr. Pedro Batista, Dr. Maxime Bouton, Dr. Jaeseong Jeong from Ericsson Research; Mr. Wenfeng Hu, Dr. Francesco Davide Calabrese, and Mrs. Françoise Cuvillier from Ericsson AB, who have each contributed meaningfully through idea brainstorming, data pipeline design, and manuscript review. In various phases of the PhD journey, I received valuable support and encouragement from Mrs. Xiaoyu Lan, Dr. Marin Orlic, Dr. Selim Ickin, Dr. Lackis Eleftheriadis, Dr. Andrii Berezovskyi, Dr. Alessandro Previti, Dr. Xin Tao, and Dr. Maxim Teslenko, whose advice and assistance I sincerely appreciate.

I am deeply grateful to my PhD fellows: Mr. Filip Cornell, Mr. Martin Isaksson, Mr. Ahmad Terra, Mrs. Carmen Lee, Dr. Tianze Wang, Dr. Zekarias Tilahun Kefato, Dr. Filippo Vannella, and Assistant Prof. Guangyi Zhang, each of whom has generously shared their knowledge and wisdom throughout countless hours of co-authoring, collaborative work, and daily brainstorming. Your support, both within and beyond academia, has been essential to the completion of this journey. I would also like to thank those who have accompanied and inspired me along the way: Assistant Prof. Ece Calikus and Assistant Prof. Sebastian Dalleiger; Dr. Simon Lindståhl, Dr. Han Xiao, Dr. Ilie Sarpe, Dr. Ruo-Chun Tzeng, Dr. Sijing Tu, and Dr. Debaditya Roy; Mr. Tianyi Zhou, Mr. Fabian Schmidt, Mr. Sofiane Ennadir, Mr. Vangjush Komini, Mr. Adam Orucu, and Mr. Gehua Dong; and Mrs. Honglian Wang. Your camaraderie, encouragement, and collaboration made this journey intellectually vibrant and personally rewarding.

I have also been fortunate to receive support and inspiration from many talented colleagues and esteemed scholars beyond my home institutions. I would like to thank Prof. Shugong Xu from Xi'an Jiaotong-Liverpool University, Assistant Prof. Rex Ying and Prof. Leandros Tassiulas from Yale University, Dr. Chang Liu from Luleå University of Technology, and Dr. Jakob Hoydis from NVIDIA for their intellectual encouragement and engagement. During my research visit to Yale, I was privileged to collaborate and exchange ideas with Prof. Menglin Yang, Dr. Ce Zhang, Dr. Ali Maatouk, Mr. Weikang Qiu, Mr. Tinglin Huang, Mr. Chen Liu, and Mrs. Jialin Chen, whose thoughtful discussions and insights were invaluable to the development of this work. The sponsorship from the Wallenberg AI, Autonomous Systems

and Software Program has supported me through research activities.
Finally, I owe my deepest gratitude to my family, whose caring, love, and belief in me have been the foundation of this journey. To my wife, Dr. Huqiao Luo, who has walked this path with me while undertaking her own PhD journey, thank you for your patience, encouragement, and quiet strength. Your understanding through countless late nights and deadlines has made all the difference. This dissertation is as much yours as it is mine.

Stockholm, November 2025
Yifei Jin

Contents

List of Figures

List of Tables

List of acronyms and abbreviations

2D Two-Dimensional

3D Three-Dimensional

3GPP 3rd Generation Partnership Project

6G Sixth generation of cellular network technology

AI Artificial Intelligence

B5G Fifth generation of cellular network technology & Beyond

Core Core Network Infrastructure

gNB next generation Node B

GNN Graph Neural Network

IoT Internet of Things

ITU International Telecommunication Union

MDP Markov Decision Process

NLP Natural Language Processing

RAN Radio Access Network

RRC Radio Resource Control

RSSI Received Signal Strength Indicator

SINR Signal to Interference & Noise Ratio

UE User Equipment

URLLC Ultra-Reliable Low-Latency Communication

Chapter 1

Introduction

To provide reliable, multi-service wireless connectivity, cellular networks have been continuously improved to support user requirements. Radio Access Network (RAN) refers to the critical component of wireless communication systems that connect end-user devices, also known as User Equipment (UE) (for example, smartphones, Internet of Things (IoT) devices) to the core network infrastructure. It includes radio base stations (noted as next generation Node Bs (gNBs)) and associated technologies that transmit and receive wireless signals. As is the case with the Fifth generation of cellular network technology & Beyond (B5G) and Sixth generation of cellular network technology (6G) infrastructures [1, 2], RAN has been evolving in terms of both hardware support and software, including algorithms for modeling the radio environment.

Managing the Complexity of Wireless Networks. The complexity of modern wireless networks[1] presents significant challenges for optimizing configurations and ensuring robust performance across diverse scenarios. Conventional approaches, such as statistical channel modeling, queuing-theory-based network analysis, and maximum-likelihood estimation for user localization, attempt to handle network uncertainties using per-site statistical observations and hard-coded heuristics. Such heuristic models struggle to adapt to the dynamic and open-world environments typical of next-generation networks. They are often hindered by computational costs and a lack of generalization capability [3].

[1] Beyond 3GPP-standardized cellular networks, radio-frequency-based sensing and topology modeling extend beyond cellular concepts, encompassing broader wireless networks such as Mobile Ad-Hoc Network, Vehicle-to-everything Network, and device-free Radio Frequency systems. In the following text, we unify the context as a wireless network,

To overcome these limitations, simulation tools (for example, InfoVista Planet [4], NS-3 [5]) and digital twin frameworks have been introduced in the pre-rollout phase of wireless network design [6]. These tools enable controlled evaluations and "what-if" analysis, but rely on simplified and parameterized assumptions. In practice, these assumptions often fail to capture the complexities of real-world scenarios [7, 8]. Incomplete measurements, evolving environments, and modeling inaccuracies lead to what we define as a *Weakly-Informed Environment*: a setting where available heuristic logic, statistical models, and data provide only partial, outdated, or noisy representations of the actual network state. This introduces challenges in calibration, drift detection, and robust deployment for accurate modeling under real-world uncertainty.

From Simulation to Learning-Driven Network Modeling. Building on these simulation-based efforts, recent methods have explored learning-enhanced frameworks that adapt to data availability and network dynamics [6]. As wireless environments become more complex, such assumptions fail to capture the dynamic, weakly-informed nature of real-world networks, leading to performance degradation and reduced inter-operator transferability.

Early applications of Artificial Intelligence (AI) in wireless networks have shown promise in site-specific tasks with learning-based models [9]. Replacing statistical and handcrafted assumptions, these methods trigger *learning-driven digital twins*, where estimating network states is partially learnt with observation from smaller-scale simulations or real-world measurements to eliminate uncertainties. Such learning often relies on isolated per gNB or per-service inputs and lacks structured representations of the network. We refer to these approaches as *Representation-less AI*: models that process observations without capturing spatial or relational dependencies between gNB, UE, and the radio environment. As a result, such methods struggle to generalize across sites, especially in weakly-informed environments.

Representation-less AI's deficiencies can be mitigated by leveraging the native topology in organizing wireless networks through Graph Representation Learning, which enables structured representations that capture the interactions among nodes, spatial dependencies, and topological constraints. This thesis employs Graph Neural Networks (GNNs) and transformer-based architectures to develop generalizable representations of wireless network behavior, facilitating each node's prediction within the network. These representations serve as a foundation for scalable, topology-aware modeling of next-generation wireless systems. In addition, this thesis

focusing on the wireless cellular system.

aims to estimate real-time network performance by bridging the theoretical advancements in GNNs and practical applications in wireless networks. A comparison of the three modeling paradigms is presented in Table 1.1.

Table 1.1: Comparison of Modeling Paradigms in Wireless Systems

Network Component	**Simulation-Based Modeling**	**Representation-less Learning**	**Graph Representation Learning**
Radio Site Configuration	Heuristic-based parameter tuning (e.g., statistical path loss models)	Site-specific prediction (e.g., deep neural network mapping geolocation to channel response)	Learn network-scale embeddings incorporating configuration priors (e.g., inter-site antenna azimuth relations)
Radio Environment	Fixed assumptions on environment (e.g., wall penetration loss ~ 3 dB [10])	Surface-specific predictions (e.g., ray angle estimation via deep neural network)	Leverage visibility graphs or geometry-aware topology to capture radio frequency multi-path structure
User Measurement	Predefined mobility profiles or static traces	Model individual user's trajectory and velocity	Incorporate road network, inter-user proximity, or mobility graphs

Finally, we observe that incorporating the aforementioned heuristics can provide efficient learning signals when aligned with graph-based representations [11, 12, 13, 14]. These heuristics can be integrated into a learning framework as supervision, inductive priors, or structural features. We define such models as *heuristic-informed neural surrogates*, a class of models that unify the strengths of simulation-based assumptions, data-driven adaptability, and topology-aware graph learning.

Thesis Statement. Critical optimization tasks in wireless networks, such as radio environment modeling, quality of service optimization, and Joint Communication and Sensing, require scalable learning frameworks that can adapt to complex and weakly-informed environments. This thesis introduces heuristic-informed neural surrogates based on graph representation learning and transformer architectures, enabling generalizable, topology-aware optimization across diverse wireless scenarios, with applications to digital twins and autonomous network management.

1.1 Research Motivation

A **wireless network** is fundamental to modern communication systems, enabling data exchange and connectivity without relying on physical cables. By using electromagnetic waves, such as radio frequency or Visible-Light, wireless networks provide wide-area coverage and ample capacity, forming the backbone of various applications, from personal devices to industrial automation and smart infrastructure. Their role in the ongoing digitization of society is important, supporting the increased demand for high-speed, low-latency, and reliable communication. Wireless networks are structured around a well-defined architecture comprising key components, such as:

- gNB and UE: gNB refers to the radio base station that connects UE to the network. UE includes mobile and stationary devices such as smartphones, IoT sensors, wearables, and vehicle transceivers. While gNB handles radio transmission and connectivity, UE acts as the endpoint for communication and data exchange.
- Core Network Infrastructure (Core): Core is responsible for managing end-to-end services across the entire wireless infrastructure, including mobility management, security, and data routing. It connects all gNBs within the service area and interfaces with the mobile service provider's backbone network.
- Wireless Medium: The wireless medium refers to the physical space, such as air or vacuum, through which electromagnetic signals propagate. It is subject to gNB and UE to optimize efficient spectrum usage and minimize interference.

Driven by user demands for higher data rates, lower latency, and more connected devices, wireless networks have evolved significantly across generations. With the advent of B5G and 6G, wireless networks are being designed to integrate diverse features, such as joint communication and sensing, Ultra-Reliable Low-Latency Communication (URLLC), and programmable network architectures. These advancements require breakthroughs not only in hardware and infrastructure but also in the optimization of intelligent network control mechanisms and radio environment modeling. However, realizing such optimization in next-generation wireless networks introduces several fundamental challenges.

- **Configuration Complexity**: The substantial growth in network nodes and heterogeneous devices makes manual optimization on each device's connection impractical. (For example, Table 1.2 on RRC[1] parameter scale growth).
- **Weakly-Informed Environments** [15, 16]: Wireless networks operate in dynamic, open-world scenarios where mobility, interference, and environmental obstructions impact performance. Weakly-informed environments arise due to the lack of adequate user service quality measurements, which are expensive to collect, and the uncertainty of the radio propagation environment, often caused by complex electromagnetic fields.
- **Generalization and Adaptability** [17, 18]: In addition to the weakly-informed nature of wireless environments, the increasing configuration complexity and deployment density further exacerbate the difficulty of generalizing from limited supervision for AI models. Heuristic-based models and representation-less AI methods, which are often preconditioned on fixed assumptions or overfit to site-specific tuning, struggle to adapt to diverse network deployments. This lack of adaptability limits their generalization performance in real-world deployments.

Recent advancements in deep learning and graph representation learning can address the aforementioned challenges. The advantages of using an AI-driven approach are as follows:

- **Proactive Network Optimization**: Given the learning and eventually predictive capability of AI algorithms such as GNNs algorithms, wireless network communication can be optimized proactively before issues such as bottlenecks [23], packet drops [17], etc., occur. Different applications include wireless coverage optimization [11], routing estimation [13], and interference mitigation [24]. Such optimization may encounter *domain adaptation* challenge as the training environment may be simplified and a simulated real world scenario. Thus, fine-formulated graph data representation can reduce the global optima into local optima that can be easier to achieve from training polices.
- **Wireless Neural Surrogates**: Similar to aforementioned simulation-based modeling, representation-less AI has been widely studied

[1] RRC is a protocol in mobile networks that manages the connection between UE and the radio base station (for example, gNB). As newer mobile network generations introduced additional functionality, such as different modes of connectivity, RRC has grown more complex with more configurable elements (parameters).

Table 1.2: Growth in Wireless Network Complexity Across Generations Regarding 3GPP-Standardized Technology: We consider 3 aspects, Neighboring Cell Number Per Cell, Parameter Complexity (exemplified by RRC parameter number), Mobility Management (exemplified by handover event from RSSI measurement), Transmitter Complexity (exemplified by antenna scheme), and Scheduling Granularity (exemplified by per radio resource granularity).

Aspect	**1G–3G**	**4G**	**5G**
Neighboring Cells	6–10	6–20	10–40+
RRC Parameter Complexity	100–500 [19]	$\sim$ 1000 [20]	$>$ 10000 [21]
Mobility Management	Basic macro level mobility	Macro mobility with per-cell offset and adjustment	Beam-level tracking, AI-based predictive user modeling
Transmitter Complexity	Single antenna at gNB and UE	Usually 8 antennas at gNB	Up to 64 antennas at gNB
Scheduling Granularity	10ms per data frame	1ms per data sub-frame	100 $\sim$ 125µs per mini-slot [22]

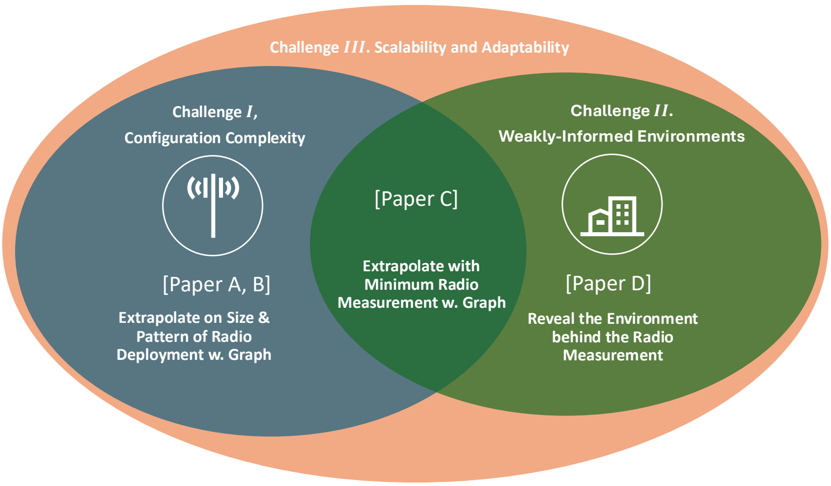


Figure 1.1: Challenges and Focus Areas of the Thesis

for learning the heuristic parameters for accelerated inference [25, 26]. Recent work [27] has revealed that such methods lack generalization when estimating unseenly larger, drifted, or configured wireless networks. Specifically, we exemplify the advantage of graph representation learning over the representation-less method on two use-cases: 1. **Wireless Ray Tracing:** Inspired by high-precision graphical rendering, similar methodologies (e.g., ray marching, path tracing) have been reused by understanding wireless signal as visible light. Unlike graphical rendering, wireless ray tracing discretizes each user location's signal with countable rays that contribute to a portion of the signal strength, phase spectrum, and latency spectrum [28, 29]. While being a strong baseline for high-fidelity channel modeling, wireless ray tracing suffers from complexity constraints, because as the ray originates from gNB, its exploring space grows exponentially when interacting with the environment, until it is received by the UE. The recent AI solution has been working on producing a neural surrogate for the ray tracing engine to reduce computation overhead. 2. **Wireless Network Modeling:** Wireless networks are naturally structured as graphs, where vertices represent network nodes, such as gNB and UE, and edges represent connectivity properties, such as interference or routing paths [30]. Graphs can be effectively modeled using AI algorithms, such as GNNs, to effectively learn the network behavior while being more precise than simulators towards real-world networks [31].

1.2 Research Objectives

The contribution of this thesis lies in the intersection of graph representation learning with B5G and 6G wireless networks, aiming to address core tasks such as estimation, prediction, and optimization at scale. By leveraging structured graph representations, we enable scalable learning and generalization across diverse wireless scenarios and topologies. We address several objectives and research questions.

O1 Develop heuristic-informed neural surrogates using graph representation learning models to jointly capture the native network topology and its domain-specific attributes. These representations are evaluated for their ability to enhance learning performance in four wireless network tasks: coverage estimation, coverage optimization, routing prediction, and wireless ray tracing.

O2 Facilitate learning through representation in B5G and 6G networks by integrating GNNs, transformer architectures, and Self-Supervised Learning. The proposed methods are applied to tasks such as coverage optimization, routing estimation, and interference mitigation under diverse network conditions.

O3 Validate the proposed graph representation learning-driven solutions using real-world network configuration data and benchmark them against commercial-grade simulation tools. The evaluation focuses on the generalization performance under uncertainty in network topology and radio environments.

1.3 Research Methodology

This section outlines the overall methodology adopted in this thesis. The methodological choices are motivated by the need to address the challenges identified in section 1.1, particularly the generalization issues in weakly-informed environments and configuration complexity. The research follows a three-stage process:

1. **Problem Definition and Theoretical Foundation**: The aforementioned challenges are exemplified through four use cases presented in the thesis, which collectively represent weakly-informed environments and configuration complexity. Despite these challenges, both categories have recently been addressed within a graph-based representation [30]: wireless entities and their interactions can be abstracted as nodes and edges, and snapshots of network states can be simulated on a smaller scale using heuristic models. This motivates the use of graph representation learning, GNNs, and transformer architectures for capturing structural priors and enabling generalization across network scenarios.

2. **Model Development and Data Processing**: We design neural surrogate models that incorporate heuristic-informed learning strategies to improve the efficiency and generalization of wireless network simulations. These models are trained on a combination of real-world and synthetic [32, 33] datasets, including radio propagation maps and network configuration logs, to capture both realistic deployment conditions and controlled variations. To ensure adaptability across diverse scenarios, we employ self-supervised learning and domain adaptation techniques, which help mitigate performance degradation

due to distribution shifts or limited labeled data in weakly-informed environments.

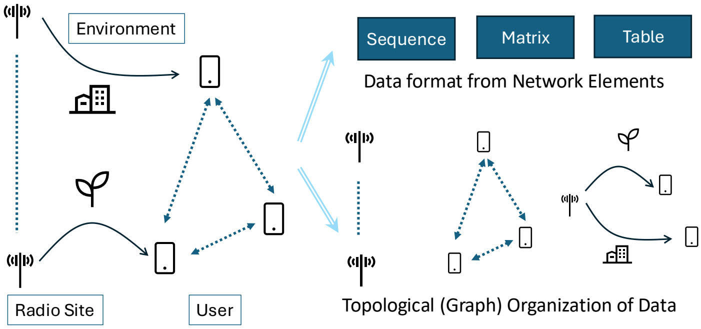
3. **Experimental Evaluation and Benchmarking:** We compare the performance of the proposed graph representation learning-based approaches with traditional analytical heuristics and conventional deep learning models. The evaluation focused on three main criteria: prediction accuracy, computational efficiency, and real-time adaptability. We further benchmarked our models using publicly available datasets and 3GPP-compliant simulation tools to ensure relevance and reproducibility in real-world scenarios.

1.4 List of Included Papers

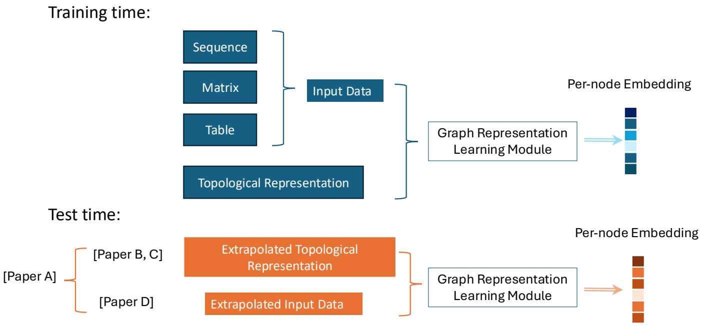
In addition to the challenges and focus areas outlined in Fig. 1.1, the contributions of this thesis are positioned within the broader graph representation learning framework, as illustrated in Fig. 1.2. As shown in Fig. 1.2a, we begin by extracting graph representations from large-scale wireless network topologies, where nodes are attributed with features informed by domain-specific heuristics. Subsequently, as shown in Fig. 1.2b, the issues of generalization and domain adaptation are addressed by learning both the graph topology and its associated features.

Paper A Y. Jin, M. Daoutis, S. Girdzijauskas, and A. Gionis. "Open World Learning Graph Convolution for Latency Estimation in Routing Networks". In: *International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2022

Author Contribution:. The author proposed the **method** of using GNN for modeling latency estimation in dynamic and scalable routing networks. Beyond previous work modeling the problem as sequence learning on per-user network flow, the author proposed a novel **problem formulation** taking routing topology as additional input. Such reformulation not only introduces the ability to generalize to unseen large network topologies, but also scales down the learning from the traffic flow sequence to the per-hop network node status. The author also executed **experiments**, showing the proposed solution is at least 4 times more accurate and 3–7 times faster against state-of-the-art methods. Finally, the author led the **manuscript preparation** to summarize the outcomes and significance. The solution is also



(a) Extracting (Graph) Representation that Proposed Graph Representation Learning Solution: Unleashing the Native Graph Topology that Organizes the Network through Learning.



(b) Position of Included Papers in Regards to the Learning Framework, with Extracted Representation

Figure 1.2: Scope of Included Papers: Perspectives from Representation (a) and Learning Framework (b)

submitted to *2021 ITU Artificial Intelligence/Machine Learning in 5G Challenge [33]* as one of the award-winning solutions.

Paper B Y. Jin, F. Vannella, M. Bouton, J. Jeong, and E. Al Hakim. "A Graph Attention Learning Approach to Antenna Tilt Optimization". In: *1st International Conference on 6G Networking (6GNet)*. IEEE. 2022

Author Contribution. The author initiated and led the **method** of using GNN for Reinforcement Learning for antenna tilt optimization in 6G networks. The author contributed the **problem formulation** on framing cellular coverage geo-proximity relations as graph structure. Such data representation scales the input space from all tilts in the scenario, down to only topological neighboring tilts. Such re-framing brings not only deployment applicability, but also maintains its performance with much less data. The author also contributed to data collection through a simulated environment and led the manuscript writing, ensuring the clarity and technical depth of the research contributions.

Paper C Y. Jin, M. Daoutis, S. Girdzijauskas, and A. Gionis. “Learning Cellular Coverage from Real Network Configurations using GNNs”. In: *IEEE Vehicular Technology Conference (IEEE VTC-Spring).* Florence, Italy, 2023

Author Contribution. Continued from **Paper B**, the author proposed the **method** of leveraging GNN for wireless coverage estimation using limited labeled data. The author designed the **problem formulation** with graph data, integrated self-supervised learning, and Few-Shot Learning to generalize from minimal field measurements.Due to the similarity in cellular configuration across different geo-locations, this graph formulation can scale down learning on wireless coverage to a single cell and the sum of the impact of its neighboring cells. The author also executed **experiments**, data collection, and benchmarking model performance with baseline models, as well as against logic-based simulation. Finally, the author led the **manuscript preparation** to summarize the research outcomes and significance.

Paper D Y. Jin, A. Maatouk, S. Girdzijauskas, S. Xu, L. Tassiulas, and R. Ying. “SANDWICH: Towards an Offline, Differentiable, Fully-Trainable Wireless Neural Ray-Tracing Surrogate”. In: *2025 IEEE International Conference on Machine Learning for Communication and Networking (ICMLCN) (2025 ICMLCN).* Barcelona, Spain

Author Contribution. The author initiated the **method** of GPU-trainable neural ray tracing in wireless environments. Existing solutions require real-time interaction with the radio environment during training, which is both costly and incompatible with GPU-based processing. In response, we propose a novel **problem formulation** that redefines ray trajectory generation as a *sequential decision-making* problem. Such formulation is not only enforcing end-to-end learning on radio ray geometries, but also being more accurate even without online supervision from interacting with the environment. The author also formulated data pipelines, prepared baseline models, and compared them against the proposed method in **experiments**. Finally, the author led the **manuscript preparation** to summarize the significance.

1.5 List of Papers not Included in the Thesis

Paper E F. Cornell, Y. Jin, J. Karlgren, and S. Girdzijauskas. "Unsupervised Ontology- and Taxonomy Construction through Hyperbolic Relational Domains and Ranges". In: *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - Workshop and Tutorial Track*. Springer. 2023

Paper F F. Cornell, Y. Jin, J. Karlgren, and S. Girdzijauskas. "Are We Wasting Time? A Fast, Accurate Performance Evaluation Framework for Knowledge Graph Link Predictors". In: *Proceedings of the IEEE International Conference on Data Mining*. 2025

Paper G Y. Jin, D. Koutlis, H. Bandala, and M. Daoutis. "Estimating Voltage Drop: Models, Features and Data Representation Towards a Neural Surrogate". In: *(Under Review) IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2024)

Paper H Y. Gao, G. Pan, Z. Zhong, Z. Jin, Y. Hu, Y. Jin, and S. Xu. "Sidelink Positioning: Standardization Advancements, Opportunities and Challenges". In: *(Under Review) IEEE Communications Magazine* (2024)

1.6 Outline

The rest of the thesis is organized as follows: In Chapter 2, we provide background knowledge relevant to the thesis. In Chapter 3, we elaborate on the method mentioned in Chapter 2 to formulate our process in more detail and summarize each published contribution. Chapter 4 describes the open questions and limitations and summarizes our findings.

Chapter 2

Background

This chapter briefly introduces the key concepts that are central to this thesis. First, we discuss GNNs and their application to wireless networks. Then, we generalize the context of GNNs to transformer models on data representation. [39] Finally, we introduce reinforcement learning for graph representation.

2.1 Graph Representation Learning

Graph data is a structured representation where entities (nodes) are interconnected through relationships (edges), making it suitable for modeling complex systems. Representation graph learning has been a foundational approach for acquiring informative vector representations from graph data. Such low-dimensional vector representations of structured data, such as nodes, edges, or entire graphs, are referred to as *embedding*. An informative embedding should be *expressive*—that is, capable of distinguishing graph components with distinct roles, topologies, or attributes on structure, ensuring good performance on downstream tasks such as node classification, link prediction, or graph-level regression. Formally, given a graph $G(V, E)$, the node embedding function is defined as:

$$f : V \to \mathbb{R}^d \quad , \tag{2.1}$$

where $f(v)$ encodes node v into a d-dimensional, real-valued vector. Embedding is constructed to preserve relevant graph properties, such as local connectivity, shared neighborhood structure, or node attributes. Similarly, the edge embeddings are defined as follows:

$$g : E \to \mathbb{R}^d \quad , \tag{2.2}$$

which encodes pairwise relational information between connected nodes. At the subgraph level, we define:

$$\hbar : G_0 \to \mathbb{R}^d \quad , \tag{2.3}$$

where $\hbar(G_0(V_0, E_0))$ maps a subgraph $G_0 \subseteq G$, with vertex set $V_0 \subseteq V$ and edge set $E_0 \subseteq E$, to d-dimensional representation. Unlike node or edge embeddings, subgraph embeddings must capture higher-order structural patterns, including motifs, community structures, and multi-hop dependencies within G_0.

Learning-free methods, such as *Random Walk* [40], *Node2Vec, Graph2Vec* [41, 42], and *NetMF* [43], rely on handcrafted heuristics and matrix factorization techniques to capture structural patterns; however, they often lack scalability and generalization capabilities. To address these limitations, neural network-driven methods such as *DeepWalk* [44] have emerged, using random walk-generated sequences as input to a skip-gram model, similar to word embeddings in Natural Language Processing (NLP). *DeepWalk* marks the beginning of deep learning approaches for graph representation learning, enabling more efficient and flexible encoding of node relationships. These advancements have led to the development of more sophisticated architectures in graph representation learning, which integrate both graph topology and node attributes into end-to-end trainable models.

In recent years, we have witnessed significant development in graph representation learning, based on prior research on Graph Mining and Spectral Graph Theory [45]. Typical applications of graph representation learning include health (for example, protein-protein interaction), physics, chemistry, and road networks modeled in graph data [46].

Wireless networks naturally exhibit a graph-like structure, where the gNB and UE form nodes, and the communication links and interference patterns define edges. This graph representation makes wireless networks ideal for the application of graph representation learning techniques that can capture spatial and temporal dependencies. One primary application of graph representation learning is understanding wireless network topologies [47, 48].

2.1.1 Graph Neural Networks

Message Passing Neural Networks. One of the foundations of modern GNNs is the message-passing neural network. Gilmer et al. [49] first integrated the *belief propagation* algorithm [50] into graph-based molecular modeling. Hamilton, Ying, and Leskovec [51] extended *belief propagation* by introducing an inductive framework that builds node embeddings through local neighborhood sampling and aggregation, demonstrating strong generalization to unseen data. Subsequent improvements include multi-head attention [52], enhanced message aggregation [53, 54], and deeper cascades of aggregation layers [55, 56]. A universal receipt of message passing graph neural network can be formulated as a neural function f_θ that updates node representations $h_v^{(l)}$:

$$h_v^{(l+1)} = \sigma\left(\mathrm{AGG}_{u\in\mathcal{N}(v)}\phi(h_u^{(l)}, h_v^{(l)}, e_{uv})\right) \quad , \tag{2.4}$$

where $h_v^{(l)}$ is the representation of node v in layer l, with a set of neighboring nodes $\mathcal{N}(v)$. AGG denotes the neural message aggregation function that fuses neighboring nodes' message embedding. ϕ is a learnable message function that takes the input of each neighboring node's embedding $h_u^{(l)}$, and its connecting edge embedding e_{uv}.

As GNNs become more expressive, concerns arise regarding their robustness to attribute noise and structural perturbations, which can enable adversarial attacks. This tradeoff between expressiveness and robustness has been explored through information-theoretic aggregation methods, such as Graph Information Bottleneck [57] and Deep Graph Infomax [55]. Another fundamental challenge is *over-squashing* [58], where distant node information becomes compressed and underrepresented, limiting effective signal propagation. To address the *over-squashing* challenge, techniques such as graph rewiring [59], hyper-nodes [60], and hyperbolic embeddings [61] have been proposed, although a comprehensive solution remains an open problem.

Spectral Graph Filtering. Beyond message-passing neural networks, spectral graph filtering forms another cornerstone of GNNs. Defferrard, Bresson, and Vandergheynst [62] utilized spectral graph theory to design robust filters, a concept later extended by Kipf and Welling [63] in the development of Graph Convolutional Network, which combines spectral filtering with learnable message aggregation. Although graph convolutional networks have remained a robust baseline for nearly a decade [64], their spectral nature also reveals limits in generalization; Balcilar et al. [65] demonstrates that all message-passing GNNs essentially operate as cascaded spectral filters. Furthermore, Ma et al. [66] characterized GNNs as performing a graph denoising process under a smoothness assumption, which weakens in deeper architectures owing to converging node features (i.e., *over-smoothing*) [67, 68]. To address the *over-smoothing* challenge, techniques such as residual connections [69], graph sparsification [70], and ordered encoding [71] have been explored. However, devising a complete solution remains an open problem.

GNN beyond Message Passing Neural Network – Graph Transformers. Graph Transformers represent an emerging paradigm that extends the conventional Transformer architecture [72] to graph-structured data. Unlike message-passing neural networks that rely on local neighborhood aggregation, Graph Transformers use a global self-attention mechanism to capture both local and long-range dependencies, thereby addressing challenges such as *over-smoothing* and *over-squashing*. Early works on message-passing neural networks, such as the Graph Attention Network and Spectral Attention Network, have extended attention to all nodes. In addition, spatial and positional [73] encoding has been introduced as the foundation of more sophisticated GNN architectures. Graphormer [74, 75], for instance, integrates centrality and spatial encoding, derived from node degrees and shortest-path distances, into attention computation. Graph transformers typically yield a fully connected attention mechanism that captures both fine-grained local patterns and global interactions.

Despite their capabilities, Graph Transformers also face challenges in computational scalability due to the quadratic complexity of global attention concerning the number of nodes [76]. Moreover, designing effective positional encodings that balance local and global structural information [77] remains a non-trivial problem. Addressing these issues is critical for fully realizing the potential of Graph Transformers in complex graph-based tasks.

In this thesis, Paper A, **Paper B**, **Paper C** and **Paper D** cover graph representation learning applied to communication networks, while **Paper A**, **Paper B** and **Paper C** specifically focus on GNN, emphasizing its generalization with respect to both graph structure and attributes. **Paper A** uses *Queuing Theory and Flow Control*, which is based on previous research on the *MaxFlow Problem* [78] in network flow control domain. **Paper B** and **Paper C** trace their origins to *Radio Coloring*, which itself evolved from *Graph Coloring Problem* within wireless network context. **Paper D** formulates *Visibility Graphs*, inspired from computational geometry and the *Path Planning Problem*.

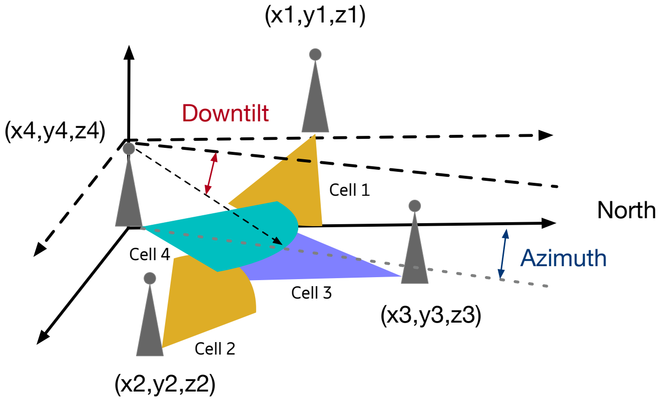


Figure 2.1: Schematic Representation on Euclidean Features in Wireless Coverages for Representing Network Geometries: Azimuth, Down tilt, and Geo-coordinates.

2.1.2 GNN on Wireless Networks

Thanks to recent progress in GNNs, wireless networks as a native structure of network nodes have been extensively studied in the state-of-the-art [79, 17, 80]. Beyond the conventional challenges posed by GNN, novel research gaps have been identified (see Section 1.2). In line with the concepts in graph representation learning, we address the challenge in connection with the aforementioned contributions.

Euclidean Attributes Expressivity on Graph. *Expressivity* in graph representations has been extensively studied across various application domains. In the context of radio frequency, graph data inherently combines Euclidean features (such as antenna azimuth, down tilt, and geo-coordinates, as illustrated in Fig. 2.1) with non-Euclidean attributes (such as inter-cell relation, interference relation, and inter-cellular user mobility), posing challenges for joint representation learning. Prior work by Joshi et al. [81] and Satorras, Hoogeboom, and Welling [82] explored the capability of learning Euclidean node features on graph topology, initially applying such techniques to protein-protein interaction prediction with a Three-Dimensional (3D) structure. Specifically for wireless network topologies, our work [14] first addresses the embedding of network geometrical features in graph representation, highlighting that the expressivity of Euclidean node features plays a critical role in learning wireless network topologies, particularly contributing to their generalization capability.

In wireless networks, the impact of wireless signal decays with a power law in the Euclidean distance between nodes. Therefore, capturing the Euclidean relationships in signal models is essential. Shen et al. [24] and Sun et al. [83] incorporated distance-aware weighting in graph representations for UE-UE communication, whereas Bouton et al. [84] extended Shen et al. [24]'s approach by embedding geographical coordinates in Two-Dimensional (2D) space to model gNB-gNB relationships. Additionally, Zhang et al. [85] addressed gNB–UE interactions by integrating heterogeneous gNB and configuration management data as graph attributes. As summarized by Shen et al. [30], effectively modeling different wireless relationships, whether UE-UE, gNB-UE, or gNB-gNB, within a unified graph representation remains an open research challenge.

Generalization in Weakly-Informed Graphs. Due to the high barrier in acquiring real-world measurement data [86] and gNB configuration parameters, the formulated graph representation is often **weakly-informed**. **Weakly-informed** nature manifests in graph representation learning models through few-shot learning, self-supervised learning, or Open-World Learning, depending on the specific learning scheme employed. While recent advancements in graph self-supervision [87] have improved representation learning, Jin et al. [88] introduced the use of *auxiliary properties* as pre-training labels. *Auxiliary properties* are intended to encode domain knowledge about the graph structure—such as the geometric proximity between a node of interest and its neighbors in a spatial scene graph—to guide the GNN to generalize to unseen or partially observed graph topologies. Hu et al. [89] used a graph structure, incorporating domain insights as pre-training signals. Handling weakly-informed graphs in wireless settings remains underexplored. Lee, Yu, and Li [90] studied UE-UE formulated graphs but focused on unsupervised loss functions rather than pre-training. Ye et al. [91] investigated extrapolating radio frequency signatures in near-neighborhoods using graph representations with limited measurement samples. Given that this formulation aligns with the concept of *Channel Charting* [92], an implicit assumption is that each location's radio frequency signature can be expressed as a combination of its neighboring locations, effectively serving as an *auxiliary property*.

To further contextualize the generalization challenges discussed above, we distinguish between two core learning paradigms in graph-based inference tasks: *transductive* and *inductive* learning.

Transductive vs. Inductive Generalization. In the context of generalization, *transductive learning* assumes that the entire graph $G = (V, E)$ is available during training, where all nodes V and edges E are observed, but only a subset of nodes (or edges) $V_{\text{train}} \subseteq V$ has known labels $Y(V_{\text{train}})$. The goal is to learn a function $f : V \to Y$ such that:

$$f(v_i) \approx y_i, \quad \text{for all } v_i \in V_{\text{test}} = V \setminus V_{\text{train}}, \tag{2.5}$$

where $Y \in \mathcal{Y}$, $\mathcal{Y}$ denotes the label space. This setting applies to fixed wireless topologies where predictions are made on unlabeled nodes within a known deployment scenario.

In contrast, *inductive learning* requires generalization to nodes or graphs that are *not* observed during training. Given a training set of graphs $\{G_i\} = \{G_1, G_2, \ldots, G_n\}$, each with its own node set V_i. Define an input space of graph(s) $\mathcal{G} \supseteq \{G_i\}$, the objective is to learn a function as follows:

$$f : \mathcal{G} \to \mathcal{Y}, \tag{2.6}$$

that performs well on a previously unseen graph $G_{n+1} \notin \{G_i\}$ or on new nodes $v \notin \bigcup_i V_i$. This setting aligns with real-world wireless scenarios, such as extrapolating to newly deployed base stations, evolving user distributions, or generalizing across cities or network configurations.

Building on the concepts of inductive and transductive learning, we now shift focus toward extrapolation. In this even more challenging setting, the graph structure, size, or underlying data distribution deviates significantly from training conditions.

Extrapolation on Novel Data. Similar to the motivation behind self-supervised learning, graph representation learning models are often trained on biased scenarios, while being expected to generalize to more complex, drifted feature-label distributions and extrapolated graph sizes during inference. The challenge of extrapolation is typically formulated within the framework of open-world learning, initially introduced for node classification by Wu, Pan, and Zhu [93].

Let $\mathcal{D}_{\text{train}} = \{(G_i, y_i)\}_{i=1}^{n}$ denote the training distribution over graph instances G_i and associated targets y_i. In extrapolation and open-world learning, the test distribution $\mathcal{D}_{\text{test}}$ differs from $\mathcal{D}_{\text{train}}$ in one or more of the following aspects:

$$\mathcal{D}_{\text{test}} \neq \mathcal{D}_{\text{train}} \quad \text{(distribution shift)}. \tag{2.7}$$

The goal is to learn a function $f : \mathcal{G} \to \mathcal{Y}$ such that:

$$\mathbb{E}_{(G,y)\sim\mathcal{D}_{\text{test}}}[\ell(f(G), y)] \quad \text{is minimized,} \tag{2.8}$$

where ℓ denotes a suitable loss function, here, $\mathcal{D}_{\text{test}}$ may include unseen graph structures, node/edge attributes, or entirely new domains (cities, time-frames, device types), reflecting the open-world or extrapolation setting.

Maintaining predictive performance on out-of-distribution data is the core difficulty. Recent research has investigated different types of extrapolation patterns: arithmetic extrapolation [94, 95, 96], graph pattern generalization [97], and graph size generalization [98]. Although significant progress has been made in physics-inspired graph models [99], the application of open-world learning to wireless-network-inspired graphs remains relatively unexplored, despite being a key challenge for real-world generalization [33]. In addition to structural representation, physics-informed neural networks [100, 101] offer a complementary approach by incorporating domain knowledge through physics-driven loss functions. While physics-informed neural networks focus on enforcing constraints from known governing equations, open-world learning in wireless graphs requires learning complex interactions beyond predefined formulations, presenting an open problem in robust generalization for real-world wireless systems.

In this thesis, Paper A and **Paper B** prove their generalization inductively on novel data distributions, with different patterns of out-of-distribution data. **Paper C** is considered a weakly-informed graph, where generalization transductively on the untrained portion of the graph takes most of the dataset.

2.2 Reinforcement Learning

Reinforcement learning is a machine learning approach that models sequential decision-making problems where an agent interacts with an environment $\mathcal{E}$ to maximize cumulative rewards. Unlike supervised learning, where labeled data guides model training, reinforcement learning enables autonomous learning through exploration and feedback, making it well-suited for dynamic and complex problem settings.

In reinforcement learning, an agent observes a triplet of (s_t, a_t, r_t): the *state* of the environment s_t, performs an *action* a_t, and receives a *reward* r_t based on the reward function $R(s_t, a_t)$. The objective is to learn an optimal *policy* $\pi^*(a|s)$ that maximizes the expected cumulative reward

$$J(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right] \quad , \tag{2.9}$$

where γ is the discount factor that balances short- and long-term rewards.

Online vs. Offline Reinforcement Learning. Reinforcement learning can be categorized based on whether the agent interacts with the environment $\mathcal{E}$ during training:

- **Online Reinforcement Learning:** The agent learns in real time by continuously interacting with $\mathcal{E}$ and updating its policy based on newly observed transitions. In online reinforcement learning, the agent iteratively interacts with the environment to collect transitions and trajectories from $[0, T]$:

  $$\tau = [(s_t, a_t, r_t, s_{t+1})]_{t=0}^{T} \quad , \tag{2.10}$$

  and uses trajectories $\{\tau\}$ as experience to update the policy $\pi(a|s)$ after each episode or time step. The data distribution is on-policy or near on-policy, and evolves as the agent's behavior changes. The learning objective is to find an optimal policy π^*:

  $$\pi^* = \arg\max_{\pi} \mathbb{E}_{\tau\sim\pi|\mathcal{E}}\left[\sum_{t=0}^{T} \gamma^t r_t\right] \quad , \tag{2.11}$$

  where $\tau \sim \pi|\mathcal{E}$ indicates that the trajectories depend on the current policy π being updated online by interaction with environment $\mathcal{E}$. Online reinforcement learning is suitable for dynamic and non-stationary environments, such as adaptive resource management in wireless networks [102].

- **Offline Reinforcement Learning:** The agent learns from a fixed dataset of pre-collected experiences without further environment interaction. In offline reinforcement learning, the agent learns from a static dataset (size N) of past transitions as follows:

  $$\mathcal{D} = \{[(s_t, a_t, r_t, s_{t+1})]_{t=0}^{T}\}_{i=1}^{N}, (s_t, a_t) \in \mathcal{E} \quad , \tag{2.12}$$

  Which are typically collected by a behavior policy π_β that is different from the current policy π being learned. The objective is still to find:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{D}} \left[\sum_{t=0}^{T} \gamma^t r_t \right] \quad , \tag{2.13}$$

but without additional interaction with the environment. This creates a distributional shift between π and the behavior-derived policy $\pi_\beta \sim \mathcal{D}$. Such drift is known as the *off-policy learning problem*, often addressed through techniques such as *importance sampling* [103]. Offline reinforcement learning is beneficial when real-time interaction is expensive or risky, such as learning from historical wireless traffic data [104].

Model-Free vs. Model-Based Reinforcement Learning. Another way to categorize reinforcement learning methods is based on whether the agent explicitly models the environment's transition dynamics.

- **Model-Free Reinforcement Learning:** The agent learns a policy directly from interaction data without constructing an explicit environment model. Instead, it directly optimizes a value function or policy from sampled trajectories. Examples include:
  - **Deep-Q-Network [105]:** Using deep neural network to approximate the action-value function $Q(s, a)$. Such methods have been widely used to use Q as a measurement of "contribution" of an action a in a given state s:

    $$Q(s, a) = \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s, a_0 = a \right] \quad . \tag{2.14}$$

    Thus, one can formulate the following policy:

    $$\pi = \arg\max_{a} Q(s, a) \quad . \tag{2.15}$$

    While Deep-Q-Network is effective in discrete action spaces, it struggles in environments with continuous action spaces and often suffers from instability due to value overestimation.
  - **Policy Gradient Methods:** To address Deep-Q-Network's limitations, policy gradient methods directly parameterize the policy $\pi_\theta(a|s)$ and optimize it by ascending the gradient of expected return. These methods are well-suited to problems

with large or continuous action spaces and allow for stochastic policy modeling, like REINFORCE [106] or Proximal Policy Optimization [107].

- **Model-Based Reinforcement Learning:** The agent constructs an explicit model of the environment's transition function $T(s_{t+1}|s_t, a_t)$ and reward function $R(s_t, a_t)$, using it for planning and decision-making. These methods are particularly useful when the environmental dynamics are known or can be estimated. Examples include:
  - **Planning-Based Approaches:** Such approaches use learned models for forward simulation and decision-making, such as Monte Carlo Tree Search in AlphaGo [108].
  - **Model Predictive Control [109]:** An approach that optimizes future trajectories by iteratively solving finite-horizon planning problems [110] using learned dynamics and reward models.
  - **World Models [111]:** Constructs an embedding of the environment using variational autoencoders and recurrent networks, allowing the agent to plan in latent space.

2.2.1 Graph Reinforcement Learning

Graph Reinforcement Learning. Graph representation learning has increasingly served as a foundational component in reinforcement learning, motivated by its ability to efficiently encode the state space through graph representations or their substructures. By capturing relational dependencies and structural priors, graph representation learning enhances the learning efficiency and generalization in complex environments. You et al. [112] and You et al. [113] explored the use of graph representations as a backbone for both online and offline reinforcement learning settings, particularly in molecular domains, where the task involves generating graph structures through sequential decision-making. In parallel, Sanchez-Gonzalez et al. [114] introduced the notion of modeling physical systems as dynamic graphs, where the environment evolves under physical laws that can be learned through graph representation learning architectures that act as differentiable simulators (i.e., digital twins). This synergy between graph representation learning and reinforcement learning has been further motivated by benefits such as the enhanced interpretability of learned policies [115] and improved inference speed in simulation-heavy tasks [116], enabling more scalable and explainable real-world systems.

Graph-based Reinforcement Learning in Wireless Networks. Despite its promise, graph-based reinforcement learning faces several key challenges when applied to wireless networks:

1. **Scalability:** Wireless networks may have thousands of nodes, making graph-based reinforcement learning computationally expensive. Therefore, efficient aggregation techniques and sparsification methods are required.

2. **Generalization to Unseen Network Scenarios:** Wireless networks are inherently dynamic, with shifting user loads, mobility, and interference. Self-supervised learning and few-shot learning approaches are required to improve policy adaptation.

3. **Exploration-Exploitation Tradeoff :** Wireless resource allocation requires balancing immediate rewards (for example, throughput maximization) with long-term objectives (for example, fairness, energy efficiency). Hierarchical reinforcement learning and multi-agent reinforcement learning approaches can help address this challenge.

There has been early work on using graph-based reinforcement learning to derive intelligent policies for load balancing [117], resource allocation [118], and inference management [14].
In this thesis, Paper B and **Paper D** apply reinforcement learning with GNN backbone. **Paper B** focuses on building an online reinforcement learning solution with real-time feedback in a network control scenario. **Paper D** discusses a wireless ray tracing surrogate based on offline reinforcement learning, which performs better than the state-of-the-art online reinforcement learning solutions. Both studies discuss the scalability of the proposed solution while being superior to its conventional deep learning counterparts.

2.2.2 Reinforcement Learning for Generative AI

Transformer Architecture. Joshi [39] bridged the gap between sparse attention mechanisms in GNN and the Transformer architecture [72], leading to a growing trend in co-designing GNNs with Transformer backbones. In the context of structured physical systems, transformers have proven effective in modeling *long-range interactions* within graphs [119], where such dependencies can be interpreted as sparse supervisory signals, akin to delayed or sparse rewards in reinforcement learning.

This capability makes transformers a compelling backbone for learning from sparse rewards over long horizons. [120] Addressing the longstanding challenge of *exploration versus exploitation*, Chen et al. [121] proposed the Decision Transformer framework, which re-frames sequential decision-making as a conditional sequence modeling task. Instead of relying on value functions or Temporal Difference updates, decision transformers model agents' action-state sequences as an auto-regressive process of (Return-to-Go, state, action) triplets. Return-to-go serves as a goal-conditioned signal, guiding action generation without explicit reward propagation or policy iteration.

Reinforcement Learning with Human Feedback. Thanks to well-defined domain heuristics from 3GPP and International Telecommunication Union (ITU) standards, it is often feasible to estimate simplified wireless system states or radio environments. While physics-informed neural networks incorporate such heuristics into physics-inspired loss terms, reinforcement learning with human feedback leverages them to define heuristic-based rewards or fitness functions that guide policy learning, particularly in scenarios in which explicit rewards are sparse or delayed. Reinforcement learning with human feedback has been widely applied in recommendation systems and robotics, where reward signals are often trajectory-dependent and difficult to specify manually.

Unlike traditional offline reinforcement learning approaches that indiscriminately utilize the entire dataset, reinforcement learning with human feedback emphasizes *sampling efficiency*, thereby favoring high-quality expert trajectories guided by the domain knowledge. This perspective shifts the learning paradigm from *"more data"* to *"better data."* Kim et al. [122] showed that incorporating diverse yet high-quality expert demonstrations is key to generalization, especially when modeling across dynamic environments. Furuta, Matsuo, and Gu [123] further established that carefully sampling policy-guided trajectories aligned with future outcomes significantly enhances stability and training efficiency in models like *decision transformer*. These findings underline that diversity and fidelity in trajectory sampling are critical for robust policy learning in sparse-reward or heuristic-driven regimes.

To optimize policies under such feedback, reinforcement learning with human feedback is commonly integrated with model-free algorithms, such as proximal policy optimization [107]. Proximal policy optimization offers stable, constrained policy updates, making it particularly suitable for learning from noisy, reward-model-driven supervision. In this context, proximal policy optimization acts as a reliable optimizer for reward-aligned policy improvement, effectively bridging domain heuristics with sample efficiency reinforcement learning.

In this thesis, Paper D proposes an efficient heuristic-informed sampling strategy on wireless rays. Paired with a domain-informed reward, the proposed model outperforms its online reinforcement learning counterpart.

Chapter 3

Summary of Appended Papers

This Chapter summarizes the publications and contributions of the studies mentioned in Section 1.4. We first discuss the common aspects in all papers, focusing on methodology in Section 3.1 and their data selection in Section 3.2. We then describe the contributions of each paper, addressing such commonalities across all papers.

3.1 Methodology

In addition to addressing different downstream tasks, a common research methodology has been used across all papers:

- **Representation Preprocessing:** Model representations of data in graphs: In Chapter 2, recent works [30, 92] have addressed that (graph) representations help with the model learning process, given that they intrinsically provide information on the structure of a wireless network. However, Bechler-Speicher et al. [124] and Zheng et al. [125] proved that GNN can overfit the provided representation structure even when this structure is irrelevant to the task. Incorrect graphs do not offer any performance gains and can even be detrimental to performance at times. Thus, a correct graph representation is key to producing a generalizable graph representation learning solution. Such insight has been concluded empirically in the form of self-supervised learning, few-shot learning, and reinforcement learning.

- **Graph Representation Learning:** Enforcing joint learning of the data attributes and the representation structure: Even though graph representation learning naturally over-fits the graph structure [124], the design space lies in the expressivity of graph representation learning on different levels of the graph's substructure (for example, node/edge, subgraph, or path/walk). According to domain heuristics, enforcing graph representation learning on the selected level of the structure must be bound with the downstream task to facilitate generalization.

- **Test/Validation Setup:** Transferring the learning outcome toward weakly-informed test data through inductive bias in representation. By the aforementioned representation and learning model, different forms of generalization (e.g., weakly-informed graph, arithmetic extrapolation, or novel data distribution, as discussed in Section 2.1.2) can be generalized on a similarly preprocessed representation. To prove the generalization performance of our approaches, we compare the proposed solutions against widely used baselines, including the logic-heuristics method and conventional deep learning.

3.2 Dataset selection

In this thesis, we consider a variety of datasets and data sources that resemble real-world distributions and behaviors. This selection is motivated by the fragmented ownership of network components across different vendors, which can hinder data publicity and standardization. To mitigate the heterogeneity introduced by vendor-specific configurations, we adopt a "configuration equivalence" approach. Specifically, when a real network is available, we extract its Configuration Management parameters and Performance Management parameters to construct a digital replica that matches it as closely as possible, enabling consistent and uniform representation across diverse network settings.

Thus, our choice of data includes a mix of real-world configuration management parameters and performance management parameters, high-fidelity simulations derived from real-world configuration parameters, and industrial baseline simulation tools [29, 28, 4]. The choice of data has also been backed by industrial partners [32, 4] and academic peers [33, 126].

3.3 Presentation of Contributions

3.3.1 Routing Network Surrogate with Open World Learning (Paper A)

This study addresses the challenge of accurate latency estimation in routing networks under *open-world* conditions, where the network attributes and topologies encountered during testing may differ significantly from those observed during training. Such scenarios are common in software-defined networks, for example, after network expansion or reconfiguration. To enable generalization across varying network sizes and data traffic distributions, we integrate *queuing theory* with GNN-based modeling, creating a heuristic-informed neural surrogate model. The model demonstrates robust performance in unseen network sizes, changed network topologies, and new user data traffic patterns.

Queuing Theory and Tele-traffic Modeling. Rooted in the foundational work of Erlang [127], queuing theory has historically been essential to traffic modeling, in which traffic is treated as a stochastic process constrained by limited network resources. This modeling approach resembles the classical *MaxFlow* problem in graph theory, where the goal is to determine the maximum feasible flows from a source to a sink under capacity constraints. However, in communication networks, traffic flows are dynamic, time-varying, and impacted by queuing delays, making the problem significantly more complex. Queuing theory extends *MaxFlow* by incorporating temporal and probabilistic factors, enabling the modeling of congestion, latency, and Quality of Service metrics under realistic load conditions.

While these principles have been extensively studied theoretically, their integration into neural architecture is a manual process involving handcrafted features [33]. In **Paper A**, we revisit queuing theory as a fundamental inductive bias and embed it within a learnable graph representation, forming a queuing-informed neural surrogate for latency estimation in communication networks.

Challenges. Modeling queuing behavior in communication networks is known to be EXPTIME-complete [128], thus having significant computational requirements. A practical surrogate model must be both computationally efficient and capable of generalizing across unforeseen test conditions. In particular, training data often lack coverage of the following: *(i)* several network features (possibly used in model training) become greatly imbalanced; *(ii)* unseen attribute values are expected to be

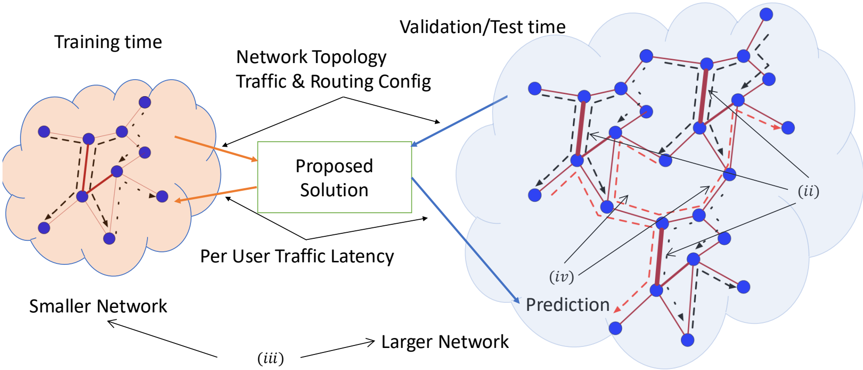


Figure 3.1: Schematic Representation of Software-Defined Network State Estimation Problem [13]. The directed line denotes network traffic flows with different throughput in the given topology, where different distributions of line types exist between training and validation/test time, as mentioned in (i). Undirected lines denote the link connecting pairs of network elements, with the line's strength denoting the link's capacity level.

out-of-distribution; *(iii)* varying, often larger, network sizes are encountered; and *(iv)* relational dependencies expand among longer node chains. These challenges are illustrated in Figure 3.1. The above requires a learning framework capable of transferring insights from smaller-scale simulations to real-world-scale networks, with generalized performance.

Methodology. Specifically, this study focuses on predicting the end-to-end user latency across routing paths. Following the methodology described in Section 3.1, the problem is reformulated from a path-level regression task into a link-level prediction task. This is motivated by the idea that the total latency along a routing path can be decomposed into the sum of the latencies incurred at each hop. Every latency is affected by the *occupancy* and congestion state of the individual communication links. By shifting the modeling granularity from path to link, the approach enables a finer resolution of user traffic demand and helps generalize to new routing paths of different lengths. This approach uses graph-theoretic and queuing-theoretic information to enrich the model's inductive bias.

- **Representation Preprocessing:** Latency estimation is reformulated from a path-level attributes prediction into a link-level prediction task. Since latency occurs on the communication link, which is represented as an edge in the routing network, we use a *line-graph* [129] representation of the original graph topology. This representation captures the sequential structure of routing paths and enables the decomposition of path-level latency into a sum over per-link *occupancy* estimates, which is reduced to a node-level label on the *line-graph* representation. Domain heuristics (such as traffic load, link capacity, and theoretical queuing behavior) are embedded into the graph. A queuing-informed *occupancy* feature is introduced and subsequently refined via GNN estimation, allowing the model to account for complex topological and traffic-dependent interactions.

- **Graph Representation Learning:** A hybrid GNN architecture is developed, combining Directed-Graph Convolutional Network, Role-Centric-Graph Attention Network, and Neural Arithmetic Logic Units components. This design supports both localized message passing and long-range dependency modeling, while also enabling numerical extrapolation. It captures queuing effects and dynamic traffic conditions that influence latency across diverse routing configurations.
- **Test/Validation Setup:** The model is evaluated on Software Defined Network simulation datasets featuring network topologies up to 6$\times$ larger than those used in training, with variations in bandwidth, traffic intensity, and routing paths. The proposed method demonstrates generalization under probability value distribution shifts, achieving at least 75% reduction in Mean Absolute Percentage Error, 3$\times$ to 7$\times$ faster inference, and 6 $\sim$ 25% lower memory usage compared to state-of-the-art GNN baselines.

Summary. The paper proposes an open-world-learning-oriented GNN framework for real-time latency estimation in communication networks. The model achieves strong extrapolation capabilities by embedding classical queuing theory into a graph-based learning pipeline, while maintaining efficiency. The results underscore the importance of representation design and domain-specific heuristics in building scalable and generalizable neural surrogates for latency-sensitive software-defined network applications. **Paper A** aligns with the research objective of O1 by producing a GNN surrogate of the routing network, while jointly capturing the routing topology and the node-level attributes formulated in queuing theory.

3.3.2 Optimizing Antenna Tilt with Graph Attention (Paper B)

Antenna tilt control is a key mechanism for optimizing the coverage and capacity of wireless networks. As wireless networks evolve toward increasingly dense and heterogeneous topologies, traditional heuristic-based tilt adjustment becomes suboptimal for adapting to dynamic interference and traffic patterns. In this paper, we present a graph representation learning-based framework to optimize antenna tilt angles using Graph Attention Q-learning, which is capable of capturing spatial inter-dependencies and generalizing across network scales.

Antenna Remote Electrical Tilt Optimization. Antenna Remote Electrical Tilt control enables dynamic adjustment of the vertical tilt angle of gNB antennas, offering a cost-efficient means to improve signal coverage and interference management. Classical optimization methods rely on handcrafted Key Performance Indicators or offline models that cannot adapt to network dynamics. Reinforcement learning provides an adaptive control mechanism by continuously interacting with the network environment. However, prior reinforcement learning approaches typically consider each cell in isolation or uniformly aggregate neighbor information, failing to capture heterogeneous inter-cell interactions, especially in dense 6G scenarios.

Challenges. Tilt optimization exhibits characteristics of multi-agent reinforcement learning, where each cell, considered as an agent, must optimize its policy while accounting for the behavior of its neighboring cells. This creates two main challenges. 1. The agent's observation space must encode spatial dependencies and interference patterns. While constrained by real-world limitations, such observation space shall not span across all sites for each agent. 2. The learned policy must scale to varying network sizes and topological densities without retraining. Conventional reinforcement learning frameworks (e.g., deep-Q-network, QMIX [130]) fall short in addressing these aspects, either due to the limitations of flat observation design or the explosion of state-action combinations. To overcome these limitations, we propose incorporating attention-based graph representation into the state space to associate neighboring cells, while avoiding global state fusion, which can be unrealistic.

Methodology. In **Paper B**, we formulate antenna tilt optimization as a Markov Decision Process (MDP), where each cell acts as an agent, with an action space being the antenna down-tilt angle, in a graph-structured environment to associate neighboring cells.

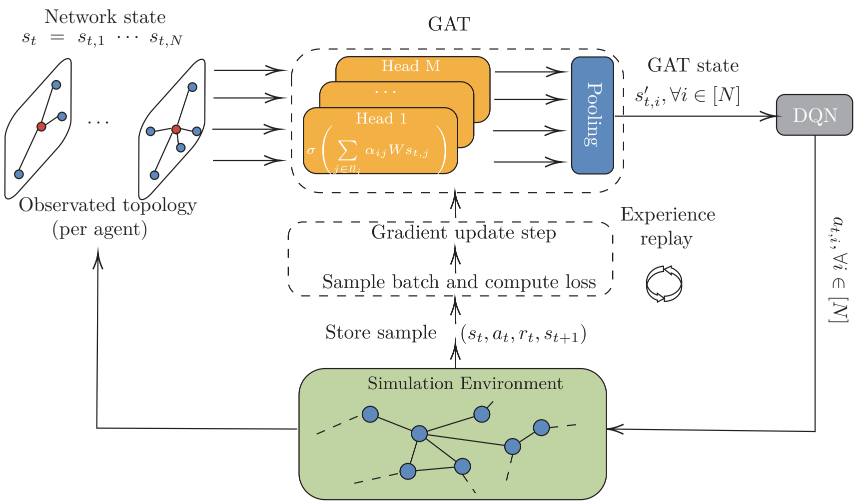


Figure 3.2: Schematic Representation of the Graph Attention Q-learning algorithm: the target agent is depicted as the red node in the graph, and its observation space is represented as the sub-graph of neighbor nodes. The Graph Attention module produces agents' state embedding from per-agent observation fed to a deep-Q-network, which is trained using experience replay.

- **Representation Preprocessing:** The network is represented as a graph, where nodes correspond to individual cells and edges capture spatial adjacency and interference influence, which are derived from geographical proximity and handover patterns. The graph attention network encoder augments each cell's state with selectively weighted features from its neighbors, guided by learned attention coefficients. The feature set includes antenna location, current tilt, antenna azimuth, and Signal to Interference & Noise Ratio (SINR) statistics, specifically the 5th, 10th, and 50th percentiles, which construct a domain-informed and spatially-aware graph structure.

- **Graph Representation Learning:** The Graph Attention Q-learning integrates a graph attention-based encoder with a deep-Q-network policy learner. The graph attention network module learns task-specific importance of neighboring cells, enabling dynamic context modeling. The learned node embeddings form the state representation used by the deep-Q-network, which is trained via prioritized experience replay and ϵ-greedy exploration. This joint architecture balances scalability with fine-grained local coordination, enabling both learning efficiency and interpretability via attention maps.

- **Test/Validation Setup:** Experiments are conducted in a realistic urban macro network simulator, with varying inter-site distances and UE distributions. Graph attention Q-learning is trained on networks with 5 neighbors per cell and tested on a densified topology with 10 and 20 neighbors, demonstrating strong generalization. It consistently outperforms the baselines, including standalone deep-Q-network, neighbor-deep-Q-network, and QMIX, in terms of both the average SINR improvement and reward convergence. Moreover, graph attention heatmaps reveal that the model learns interpretable and spatially coherent patterns that align with domain-specific heuristics.

Summary. Paper B introduces graph attention Q-learning, a graph attention-based reinforcement learning approach for antenna remote electrical tilt optimization. By encoding structured spatial dependencies and learning adaptive neighbor influence, graph attention Q-learning delivers superior performance and scalability compared to traditional reinforcement learning and neural baselines. The method generalizes across network densities and preserves interpretability, marking it as a robust candidate for self-optimizing networks in 6G deployments. **Paper B** aligns with the research objective O2, by using graph representation as prior to facilitate antenna tilt control in a multi-agent reinforcement learning formulation, with generalization capability.

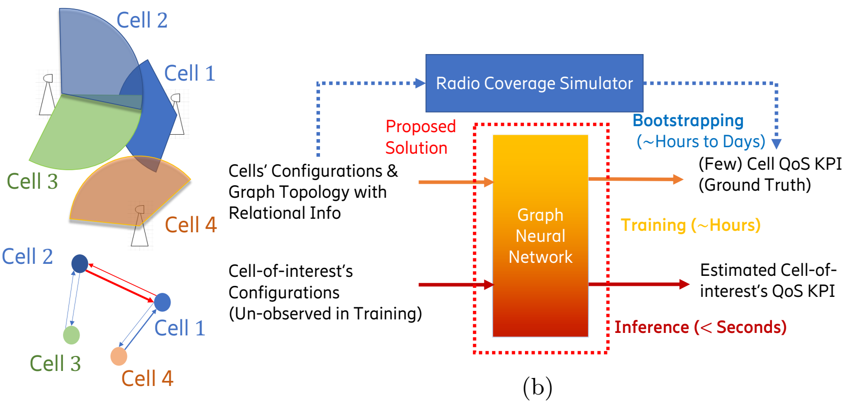
3.3.3 Learning Cellular Coverage from Real Network Configurations using GNNs (Paper C)

This study addresses the challenge of real-time coverage estimation in complex and heterogeneous radio environments. While **Paper B** focuses on a single tunable parameter (i.e., antenna tilt), this study expands the scope to a high-dimensional set of real-world radio configuration parameters. These parameters are drawn from actual live deployments and introduce heterogeneity in both feature and structural spaces, challenging the expressivity and generalization capabilities of conventional GNNs.

Cellular Coverage Estimation. Accurate estimation of cellular coverage is essential for network planning, troubleshooting, and self-optimization. Traditional propagation models or simulation-based approaches often fail to scale to the dynamic and noisy nature of real-world deployments. In contrast, data-driven models trained directly on configuration and measurement data offer promising alternatives. However, learning from production-grade configuration datasets introduces several new challenges, such as noisy inputs, heterogeneous parameter distributions, and domain constraints on feasibility. In this study, we consider a GNN-based data-driven approach that learns a sophisticated propagation model with complex scenario-specific heuristic logic. This formulation is illustrated in Figure 3.3b.

Challenges. Compared to **Paper B**, which operated under clean, simulator-defined environments, the learning task in this study is complicated by the following factors: 1. **Heterogeneous features:** Configuration parameters differ in type (continuous, categorical, binary, and directional), scale, and meaning, requiring advanced encoding strategies to avoid model collapse or overfitting. 2. **Partial observability and weak supervision:** Due to costly measurement, many cells lack ground-truth labels (i.e., coverage measurements), making the setting transductive, where generalization is needed on the untrained part of a few, partially labeled graphs. 3. **Graph noise and edge heterogeneity:** The underlying graph derived from real deployments is incomplete and noisy; edge weights may reflect coverage overlaps, interference relationships, or handover patterns, without heterogeneous semantics.

Methodology. Besides comparing against baselines in **Paper B** and other radio frequency interference management baselines at small scale [24], we formulate the problem in a heterogeneous graph representation and tackle the



(a) (b)

Figure 3.3: (*a*) Example of a graph representation of 'inter-cell' relations [14]. Each vertex represents a cellular coverage, color-coded by its carrier frequency (blue, green, or orange). Each directed edge denotes an interfering (red) or complementing (blue) impact posed to the destination vertex, where edge strength denotes the strength of the impact. (*b*) A schematic overview of the proposed solution [14]. Bootstrapping time ranges from hours to days, training time requires several hours to complete, while test and validation time is in the range of a few seconds.

challenge in a self-supervised learning framework to tackle the aforementioned challenges.

- **Representation Preprocessing:** Cells are modeled as nodes in a graph where edges represent inter-cell relations extracted from overlapping coverage, handover data, or proximity. The node features consist of 18 configuration fields from live real network data (e.g., frequency, azimuth, tilt, transmitter power, and antenna model). Raw features undergo normalization and encoding strategies (e.g., embeddings for categorical values and log-scaling for power-related features) to enhance numerical stability. Such preprocessing is illustrated in Figure 3.3a.

- **Graph Representation Learning:** To handle heterogeneous and high-dimensional inputs, we developed a deep GNN framework with residual skip connections and multiple propagation layers. The architecture incorporates message passing augmented by positional encoding and optional attention modules. The network is first trained on an edge-level pseudo-label that facilitates learning the coverage overlap between cells. Then, the network is fine-tuned using a masked supervision loss over partially labeled nodes on the node level, leveraging spatial coherence as inductive bias.

- **Test/Validation Setup:** The model is trained and validated on real-world data collected from a live commercial network covering urban and suburban areas. We perform spatial holdout and temporal holdout validation. Performance is measured via Root Mean Squared Error for SINR and service quality estimation. The model is benchmarked against other cellular coverage estimation baselines as well as fully connected multilayer perceptrons, and basic GNN variants. The results demonstrate strong generalization even on unseen geographic clusters, highlighting the GNN's capability to transfer across deployment zones through the graph prior.

Summary. **Paper C** demonstrates the feasibility and effectiveness of using GNNs to learn real-world cellular coverage directly from live network configurations. By encoding noisy and heterogeneous configuration parameters into graph representations and training in a transductive learning setup, the proposed model offers a robust tool for network diagnostics and auto-configuration. This study extends beyond simulator-based studies by addressing the challenges of real-world production data and contributes to the realization of data-driven, self-optimizing networks in practical settings. Similar to **Paper B**, **Paper C** also aligns with the research objective O2 by providing generalization performance with a graph representation prior. However, **Paper C** elevates the graph formulation with heterogeneous configuration parameters, resulting in a more realistic and complex representation and solving using a self-supervised learning framework. **Paper C** also addresses objective O3 by using real-world network's configuration data and comparing against commercially-used simulator baselines.

3.3.4 Wireless Neural Ray Tracing Surrogate (Paper D)

While *system-level* simulations typically abstract wireless environments into large-scale network behaviors, such as coverage and interference management, *link-level modeling* offers finer granularity by focusing on radio signal interactions between transmitters and receivers. In contrast to the cell-level measurements and system-level scenarios investigated in **Paper B** and **Paper C**, this study addresses the link-level regime using neural ray tracing as a surrogate modeling approach.

With the emergence of high-frequency B5G systems and site-specific propagation requirements, wireless ray tracing has gained attention as a tool for high-fidelity, scene-aware channel modeling. While traditional ray tracing engines [28, 29] can approximate the behavior of electromagnetic fields through stochastic simulation, they suffer from computational bottlenecks, GPU incompatibility, and limited real-time capability. These constraints hinder their applicability in use cases such as beam management, mobility prediction, and joint communication and sensing applications.

Computational Electromagnetic Theory. In modern wireless network modeling, the wireless medium is no longer abstracted as a static logic link (for example, **Paper A**), but rather as a dynamic and spatially aware propagation field. Wireless ray tracing aims to capture this complexity by simulating how radio signals interact with their environment, that is, by reflecting, diffracting, and penetrating surfaces, resulting in what is known as a Shoot-and-Bounce Sequence. This sequence approximates the physical trajectory of a ray from the transmitter to the receiver, under the influence of the 3D environment.

Traditional methods [28] generate shoot-and-bounce sequences using Monte Carlo-style sampling across potential ray paths [131], following physical heuristics rooted in electromagnetic theory. While accurate, these methods are computationally expensive and heavily rely on proprietary simulators, which limits their scalability and adaptability. In contrast, this paper proposes learning a neural surrogate that generates shoot-and-bounce sequences efficiently while maintaining high fidelity to the underlying propagation phenomena.

In **Paper D**, we investigate the neural process of such heuristics. We consider using reinforcement learning with human feedback to facilitate sampling efficiency in wireless ray tracing heuristics. Additionally, we use the decision transformer to model the multi-hop interaction between the radio frequency and radio environment into a sequential decision problem: each wireless ray from the transmitter, which imposes a contribution to the UE, is rewarded as a reinforcement learning policy integrating electromagnetic field heuristics.

Challenges. Prior neural approaches, such as WINERT [32], treat wireless ray tracing as a greedy next-hop prediction task using temporal-difference learning. However, these methods face fundamental limitations: 1. **Long-range dependencies** are neglected due to step-wise policy learning; 2. Training relies on **real-time supervision** from a ray tracing engine, which is non-differentiable and computationally intensive; 3. **Sparse supervision** across

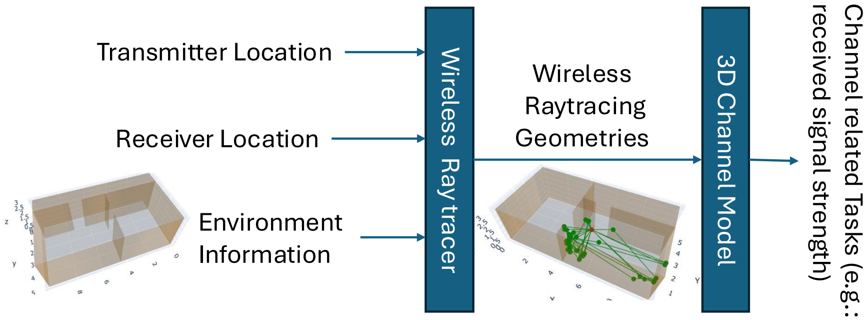


Figure 3.4: Workflow of Wireless Raytracer: Wireless Raytracer takes the input of the transmitter, receiver location, and radio environment information, and outputs a shoot-and-bounce sequence to be further utilized on wireless channel-related tasks.

shoot-and-bounce sequence trajectories limits the scalability of end-to-end training. To address these limitations, we reformulated wireless ray tracing as a sequential decision-making problem that allows offline, vectorized, and fully differentiable training using transformer models.

Methodology. In **Paper D**, we model wireless ray propagation as a sequential decision process using data-driven learning and apply a unified representation-learning-validation pipeline adapted to 3D spatial radio environments:

- **Representation Preprocessing:** Unlike traditional graph-based tasks, wireless ray tracing involves structured sequences of radio frequency interactions in 3D space (i.e., shoot-and-bounce sequence). To learn such sequences, we reformulate each shoot-and-bounce sequence as a state-action-interaction trajectory, capturing the directional flow and interaction type (for example, reflection, diffraction, and penetration) of each ray segment. A Fibonacci-sphere-based augmentation scheme is applied to enrich the shoot-and-bounce sequence space and provide robust coverage of the angular domain. This preprocessing step creates a sequentially structured representation that is compatible with transformer-based models.

- **Graph Representation Learning:** Although no explicit graph is used, sequential ray interactions can be interpreted as walk-based substructures within a spatial graph of the environment. We apply a decision transformer as the backbone model, which is enhanced with state supervision and trajectory-based reward shaping. The decision transformer learns to generate a shoot-and-bounce sequence by modeling long-range dependencies across the ray steps, thereby aligning with the graph representation learning principle of jointly optimizing the representation and downstream objectives, specifically spatial ray fidelity. The augmented shoot-and-bounce sequence and structured reward (for example, angular deviation, terminal error, and geometric consistency) serve as inductive biases guiding generalizations to unseen transmitter-receiver pairs and spatial configurations.

- **Test/Validation Setup:** The model is trained entirely offline and evaluated on three out-of-distribution 3D layouts with novel gNB and UE placements. Validation focuses on both surrogate fidelity (angular Mean Absolute Error and Kullback–Leibler divergence) and downstream usability in the RSSI estimation. The learned surrogate achieves 10^{-2} rad angular accuracy, within 0.5 dB of ground-truth ray tracing in RSSI estimation, and outperforms previous baselines. Such results demonstrate the effectiveness of the surrogate representation in capturing the essential physics-based behaviors for wireless propagation.

Summary. **Paper D** introduces a novel wireless ray tracing neural surrogate that is scene-aware, differentiable, and offline-trainable. By framing ray propagation as a sequential decision-making process, the model learns physically consistent radio frequency behaviors without relying on expensive online supervision. This study marks a step toward scalable and deployable neural surrogates for high-fidelity wireless channel modeling, with direct applications in Joint Communication and Sensing, beamforming, and 3D-aware system-level optimization. **Paper D** aligns with objective O1 by combining optical and wireless domain knowledge, respectively, into ray augmentation and fitting reward to facilitate learning an optical-compliant, wireless-informed ray tracing neural surrogate. **Paper D** also addressed objective O3 by using real-world network's configuration data and comparing against commercially-used simulator baselines.

Chapter 4

Conclusions and Future work

4.1 Conclusions

Despite recent advances in machine learning in wireless networks, the lack of high-quality data, efficient neural architectures, and principled integration of network topology has hindered progress in learning native representations for wireless network optimization and modeling. Graph representation learning, especially with the rise of GNNs, has emerged as the missing piece that enables data-driven wireless network optimization by embedding structural domain knowledge. By incorporating graph-based priors, graph representation learning offers a path toward learning models that generalize across dynamic, heterogeneous, and weakly-informed wireless environments. This dissertation contributes to this emerging field by proposing methods and offering insights into how graph representation learning can be effectively applied to wireless systems, bringing learning-based wireless optimization closer to real-world deployment readiness.

First, we demonstrate that graph representations offer a powerful structural prior that enhances learning outcomes in wireless topologies. Both GNNs and structure-aware transformer architectures, whether explicitly or implicitly [39], leverage graph structure to improve generalization. Our empirical results across **Paper A**, **Paper B**, **Paper C**, and **Paper D** show that these graph-based methods outperform traditional baselines and approach the accuracy of commercial-grade simulators in predicting wireless performance and guiding optimization.

Second, we argue that domain-informed graph representations can serve as task-agnostic embeddings within the wireless domain. Through techniques such as self-supervised learning, multi-task learning, and reinforcement learning with human feedback, **Paper C** and **Paper D** show that: 1. Strong graph representations augmented with domain heuristics can yield useful embeddings with minimal supervision. 2. These embeddings can be optimized in a discretized latent space using rewards derived from domain knowledge, enabling effective policy learning with limited observations.

Last, our work reveals that many well-established engineering practices in wireless networking, both successful and suboptimal, can be interpreted through the lens of graph spectral theory. For instance, **Paper A** attributes strong generalization in routing tasks to consistent graph Laplacian spectra across networks of varying sizes, aligning with engineers' intuition to reinforce primary communication paths during network expansion. Conversely, **Paper C** reveals the difficulty of achieving inductive generalization across cities due to divergence in the Laplacian spectra of their coverage graphs.

In summary, this dissertation validates the role of graph representation learning as a foundational tool for building generalizable, interpretable, and domain-aligned models for wireless networks. Our contributions pave the way for bridging wireless engineering intuition with machine learning rigor, setting the stage for robust, real-world optimization and automation in future generations of networks.

4.2 Limitations

Domain Barrier. Besides the contributions above, constructing graph representations enriched with domain heuristics inevitably requires substantial domain knowledge and understanding of wireless network principles. While such representations significantly improve model performance and generalization, they introduce a dependence on expert-crafted feature engineering and topology design, which may limit scalability to unfamiliar domains or fully automated pipelines.

Structural Dependency. Moreover, reliance on accurate metadata—such as wireless ray tracing geometries, neighbor coverage relations, and user statistics—can be a significant limitation, particularly in scenarios where structural information is incomplete, noisy, or inconsistent. Recent works [124] have demonstrated that while GNNs generalize well over attribute-based features, they are prone to overfitting to the structural aspects of graphs. In **Paper A**, this issue is mitigated through the use of expert-crafted graph representations. In **Paper C**, we adopt a transductive learning setting and observe that the model's performance deteriorates significantly when attempting to generalize to unseen cities—i.e., under inductive conditions. These challenges underscore the need for future research into automated structure discovery and adaptive representations that can reduce reliance on manually designed graph structures without compromising model performance.

High-Order Interaction Inefficiency. Beyond relational and Euclidean features, wireless network representations often involve higher-order interactions across node clusters. Conventional pairwise models do not easily capture these complex dependencies. In **Paper A**, successful generalization hinges not only on handling variable graph sizes but also on accurately predicting path label lengths previously unseen during training. This is addressed by decomposing the prediction into a sequence of node-level label aggregations along each path. Similarly, **Paper D** faces an analogous challenge in generalizing to unseen ray trajectory lengths. To address this, the problem is reformulated as a sequence generation task to accommodate the required inductive capability. Despite these task-specific workarounds, effectively modeling high-order and large-scale interactions across nodes remains a critical barrier for enabling full-scale optimization across multi-hop and multi-agent wireless settings.

4.3 Sustainability and Ethics

Sustainability. Recent advances in Green Machine Learning have highlighted the urgent need for more sustainable training methods and energy-efficient neural architectures. The computational demands of large-scale neural networks, particularly transformers, have driven research toward sparsification techniques that reduce both training and inference costs. GNNs provide a natural solution to this challenge by leveraging structural sparsity through explicit graph topology, effectively implementing learned attention masks that focus computation on relevant connections rather than the dense, 'all-to-all' interactions typical of transformer architectures.
This computational efficiency is particularly relevant as the telecom industry moves toward network softwarization. As demonstrated in this thesis, GNN-based approaches achieve comparable or superior performance to traditional simulation-based methods while requiring significantly less computational resources. Our results in **Paper A** and **Paper C** show 3–7 times faster inference times and a reduction to 6–25% memory usage compared to conventional baselines. This efficiency translates directly to reduced energy consumption during both model training and deployment, making GNNs

particularly valuable for large-scale wireless network optimization where computational sustainability is crucial for practical implementation. The structural inductive biases inherent in graph representations thus serve dual purposes: improving model generalization while simultaneously reducing environmental impact through more efficient computation.

Ethical Consideration. In this thesis, we utilize real-world network configuration data (**Paper C**) and performance measurements (**Paper A**) from operational cellular networks, raising important questions about data stewardship and user privacy. While the network configuration parameters themselves may appear technical and anonymized, they can reveal patterns about user density, mobility, and service demands in specific geographical areas or service groups. We have implemented strict data anonymization protocols that align with General Data Protection Regulation and worked closely with industry partners to ensure compliance with telecommunications data protection regulations. All location-specific identifiers have been removed or generalized, and the research focuses on network-level optimization rather than individual user tracking, which GNN has proven to be capable of as well. [132]
In **Paper D**, we proposed a method that involves precise environment perception and user behavior modeling, which could potentially be repurposed for surveillance or monitoring applications beyond operating network optimization. We emphasize that this research, similar to all research contributions in the Joint Communication and Sensing community and 3GPP standardization progress, is intended solely for improving network performance and efficiency, and we encourage responsible deployment practices that respect user privacy and civil liberties.

4.4 Future work

Wireless Foundation Model. Due to the recent hype around foundation models, the pursuit of learning **task-agnostic wireless embedding** for wireless networks: **jointly capturing both network behavior and environmental characteristics**, has begun to attract attention. This direction holds the potential to become the 'large language model moment' for machine learning in wireless network optimization and intelligent control. The ideal embedding should be expressive of network dynamics, informed by network topology, and compliant with domain-specific knowledge. While previous methods, such as physics-inspired neural networks and graph representation learning, have partially addressed these challenges, a unified solution remains elusive.

In particular, the development of a universal loss function for task-agnostic wireless embeddings—whether grounded in structural properties, physical laws, or engineering heuristics- has yet to be established. We consider the graph representation learning approach, as proposed in this dissertation, a promising foundation for this endeavor. It offers a structured framework that naturally integrates topological information, supports multi-task supervision, and aligns well with real-world wireless configurations. Such a development would pave the way for foundation models for real-time optimization, control, and learning wireless networks, capable of powering diverse downstream applications with minimal retraining and maximum generalizability.

Graph Learning Over-the-Air. Beyond utilizing graph representations to model wireless networks, emerging research [133] suggests that graphs, due to their inherent structural robustness, can naturally mitigate the impact of attribute noise. This is particularly relevant in wireless networks, where channel noise and data corruption are typically addressed through coding overhead and redundancy. Graph-based learning, by contrast, offers an alternative strategy by leveraging topological context to preserve meaningful signal relationships. While these properties have been demonstrated in other types of topologies [134], their applicability to noise-perturbed wireless graphs remains largely unexplored. Key advancements in the broader GNN community, such as graph rewiring [135], positional encoding [73], and graph transformers [136], have yet to be systematically evaluated under realistic wireless noise conditions.

Scalable Multi-Objective Optimization. Many wireless applications involve conflicting goals, such as coverage versus interference, latency versus energy, or sensing versus throughput. Future work should investigate how graph-based models can support multi-objective optimization at scale, balancing such trade-offs using principled frameworks, such as Pareto fronts or constrained reinforcement learning, while ensuring real-time decision-making performance in dynamic environments.

Bibliography

[1] M. A. Uusitalo et al. "Hexa-X the European 6G flagship project". In: *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE. 2021, pp. 580–585.

[2] P. Nagaraj. *Demystifying IMT-2030 aka 6G-Capabilities, Usage Scenarios, and Candidate Technologies*. 2024.

[3] P. Idd. "Channel estimation in mobile wireless systems". In: *M. Tech Thesis, Royal Institute of Technology Stockholm, Sweden* (2012).

[4] Infovista. *Infovista: 5G Network Planning, Testing & Assurance Automation*. Accessed: 2025-04-14. 2025. URL: https://www.infovista.com/.

[5] I. Stojmenovic. "Simulations in wireless sensor and ad hoc networks: matching and advancing models, metrics, and solutions". In: *IEEE Communications Magazine* 46.12 (2008), pp. 102–107.

[6] L. U. Khan et al. "Digital twin of wireless systems: Overview, taxonomy, challenges, and opportunities". In: *IEEE Communications Surveys & Tutorials* 24.4 (2022), pp. 2230–2254.

[7] Y. Jin. *Feasibility study of vehicular teleoperation over cellular network in urban scenario*. 2017.

[8] J. Lessmann et al. "Comparative study of wireless network simulators". In: *Seventh International Conference on Networking (icn 2008)*. IEEE. 2008, pp. 517–523.

[9] J. Wang et al. "Artificial intelligence and wireless communications". In: *Frontiers of Information Technology & Electronic Engineering* 21.10 (2020), pp. 1413–1425.

[10] International Telecommunication Union Radiocommunication Sector (ITU-R). *Prediction of building entry loss.* Recommendation P.2109-0. ITU-R Recommendation P.2109-0. International Telecommunication Union, Sept. 2017. URL: https://www.itu.int/rec/R-REC-P.2109.

[11] Y. Jin et al. "A Graph Attention Learning Approach to Antenna Tilt Optimization". In: *1st International Conference on 6G Networking (6GNet)*. IEEE. 2022.

[12] Y. Jin et al. "SANDWICH: Towards an offline, differentiable, fully-trainable wireless neural ray-tracing surrogate". In: *2025 IEEE International Conference on Machine Learning for Communication and Networking (ICMLCN)*. IEEE. 2025, pp. 1–7.

[13] Y. Jin et al. "Open World Learning Graph Convolution for Latency Estimation in Routing Networks". In: *International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2022.

[14] Y. Jin et al. "Learning Cellular Coverage from Real Network Configurations using GNNs". In: *IEEE Vehicular Technology Conference (IEEE VTC-Spring)*. Florence, Italy, 2023.

[15] J. Hoydis et al. "Learning radio environments by differentiable ray tracing". In: *IEEE Transactions on Machine Learning in Communications and Networking* (2024).

[16] L. Zhang et al. "WiSegRT: Dataset for site-specific indoor radio propagation modeling with 3D segmentation and differentiable ray-tracing". In: *2024 International Conference on Computing, Networking and Communications (ICNC)*. IEEE. 2024, pp. 744–748.

[17] J. Suárez-Varela et al. "Graph neural networks for communication networks: Context, use cases and opportunities". In: *IEEE network* 37.3 (2022), pp. 146–153.

[18] M. Ferriol-Galmés et al. "Scaling graph-based deep learning models to larger networks". In: *arXiv preprint arXiv:2110.01261* (2021).

[19] 3GPP. *Radio Resource Control (RRC); Protocol specification.* Technical Specification (TS) 25.331. 3rd Generation Partnership Project (3GPP), 2011. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=1180.

[20] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification*. Technical Specification (TS) 36.331. 3rd Generation Partnership Project (3GPP), 2017. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2440.

[21] 3GPP. *NR; Radio Resource Control (RRC); Protocol specification*. Technical Specification (TS) 38.331. 3rd Generation Partnership Project (3GPP), 2021. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3197.

[22] 3GPP. *NR; Radio Resource Control (RRC); Protocol specification*. Technical Specification (TS) 38.331. 3rd Generation Partnership Project (3GPP), 2021. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3197.

[23] H. Wang et al. "A graph neural network-based digital twin for network slicing management". In: *IEEE Transactions on Industrial Informatics* 18.2 (2020), pp. 1367–1376.

[24] Y. Shen et al. "Graph neural networks for scalable radio resource management: Architecture design and theoretical analysis". In: *IEEE Journal on Selected Areas in Communications* 39.1 (2020), pp. 101–115.

[25] A. Mozo et al. "B5GEMINI: AI-driven network digital twin". In: *Sensors* 22.11 (2022), p. 4106.

[26] X. Lin et al. "6G digital twin networks: From theory to practice". In: *IEEE Communications Magazine* 61.11 (2023), pp. 72–78.

[27] Z. Tao et al. "Wireless network digital twin for 6g: Generative ai as a key enabler". In: *IEEE Wireless Communications* 31.4 (2024), pp. 24–31.

[28] J. Hoydis et al. "Sionna RT: Differentiable ray tracing for radio propagation modeling". In: *2023 IEEE Globecom Workshops (GC Wkshps)*. IEEE, pp. 317–321.

[29] N. Amiot, M. Laaraiedh, and B. Uguen. "Pylayers: An open source dynamic simulator for indoor propagation and localization". In: *2013 IEEE International Conference on Communications Workshops (ICC)*. IEEE, pp. 84–88.

[30] Y. Shen et al. “Graph neural networks for wireless communications: From theory to practice”. In: *IEEE Transactions on Wireless Communications* 22.5 (2022), pp. 3554–3569.

[31] S. Jiang et al. *Learnable Wireless Digital Twins: Reconstructing Electromagnetic Field with Neural Representations*. arXiv:2409.02564 [cs]. Sept. 2024. DOI: 10.48550/arXiv.2409.02564. URL: http://arxiv.org/abs/2409.02564 (visited on 02/21/2025).

[32] T. Orekondy et al. “Winert: Towards neural ray tracing for wireless channel modelling and differentiable simulations”. In: *The Eleventh International Conference on Learning Representations*. 2023.

[33] J. Suárez-Varela et al. “The graph neural networking challenge: a worldwide competition for education in AI/ML for networks”. In: *ACM SIGCOMM Computer Communication Review* 51.3 (2021), pp. 9–16.

[34] Y. Jin et al. “SANDWICH: Towards an Offline, Differentiable, Fully-Trainable Wireless Neural Ray-Tracing Surrogate”. In: *2025 IEEE International Conference on Machine Learning for Communication and Networking (ICMLCN) (2025 ICMLCN)*. Barcelona, Spain.

[35] F. Cornell et al. “Unsupervised Ontology- and Taxonomy Construction through Hyperbolic Relational Domains and Ranges”. In: *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - Workshop and Tutorial Track*. Springer. 2023.

[36] F. Cornell et al. “Are We Wasting Time? A Fast, Accurate Performance Evaluation Framework for Knowledge Graph Link Predictors”. In: *Proceedings of the IEEE International Conference on Data Mining*. 2025.

[37] Y. Jin et al. “Estimating Voltage Drop: Models, Features and Data Representation Towards a Neural Surrogate”. In: *(Under Review) IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2024).

[38] Y. Gao et al. “Sidelink Positioning: Standardization Advancements, Opportunities and Challenges”. In: *(Under Review) IEEE Communications Magazine* (2024).

[39] C. K. Joshi. “Transformers are graph neural networks”. In: *arXiv preprint arXiv:2506.22084* (2025).

[40] K. Pearson. “The problem of the random walk”. In: *Nature* 72.1867 (1905), pp. 342–342.

[41] A. Grover and J. Leskovec. "node2vec: Scalable feature learning for networks". In: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining.* 2016, pp. 855–864.

[42] R. V. L. C. Y. L. Annamalai Narayanan Mahinthan Chandramohan and S. Jaiswal. "graph2vec: Learning Distributed Representations of Graphs". In: *Proceedings of the 13th International Workshop on Mining and Learning with Graphs (MLG).* 2017.

[43] J. Qiu et al. "Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec". In: *Proceedings of the eleventh ACM international conference on web search and data mining.* 2018, pp. 459–467.

[44] B. Perozzi, R. Al-Rfou, and S. Skiena. "Deepwalk: Online learning of social representations". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2014, pp. 701–710.

[45] D. A. Spielman. "Spectral Graph Theory and its Applications". In: *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07).* 2007, pp. 29–38. DOI: 10.1109/FOCS.2007.56.

[46] W. L. Hamilton. *Graph representation learning.* Morgan & Claypool Publishers, 2020.

[47] H. Tamura et al. "On applications of graph/network theory to problems in communication systems". In: *ECTI Transactions on Computer and Information Technology (ECTI-CIT)* 5.1 (2011), pp. 15–21.

[48] M. A. Hassan and A. Chickadel. "A review of interference reduction in wireless networks using graph coloring methods". In: *arXiv preprint arXiv:1103.5791* (2011).

[49] J. Gilmer et al. "Neural message passing for quantum chemistry". In: *International conference on machine learning.* PMLR. 2017, pp. 1263–1272.

[50] J. S. Yedidia, W. Freeman, and Y. Weiss. "Generalized belief propagation". In: *Advances in neural information processing systems* 13 (2000).

[51] W. Hamilton, Z. Ying, and J. Leskovec. "Inductive representation learning on large graphs". In: *Advances in neural information processing systems* 30 (2017).

[52] P. Veličković et al. “Graph attention networks”. In: *International Conference on Learning Representations*. 2018.

[53] G. Corso et al. “Principal neighbourhood aggregation for graph nets”. In: *Advances in neural information processing systems* 33 (2020), pp. 13260–13271.

[54] K. Xu et al. “How Powerful are Graph Neural Networks?” In: *International Conference on Learning Representations*. 2019.

[55] P. Veličković et al. “Deep Graph Infomax”. In: *International Conference on Learning Representations*. 2019.

[56] K. Xu et al. “Optimization of graph neural networks: Implicit acceleration by skip connections and more depth”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 11592–11602.

[57] T. Wu et al. “Graph information bottleneck”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 20437–20448.

[58] F. Di Giovanni et al. “On over-squashing in message passing neural networks: The impact of width, depth, and topology”. In: *International conference on machine learning*. PMLR. 2023, pp. 7865–7885.

[59] H. Attali, D. Buscaldi, and N. Pernelle. “Rewiring Techniques to Mitigate Oversquashing and Oversmoothing in GNNs: A Survey”. In: *arXiv preprint arXiv:2411.17429* (2024).

[60] U. Alon and E. Yahav. “On the Bottleneck of Graph Neural Networks and its Practical Implications”. In: *International Conference on Learning Representations*. 2021.

[61] Q. Liu, M. Nickel, and D. Kiela. “Hyperbolic graph neural networks”. In: *Advances in neural information processing systems* 32 (2019).

[62] M. Defferrard, X. Bresson, and P. Vandergheynst. “Convolutional neural networks on graphs with fast localized spectral filtering”. In: *Advances in neural information processing systems* 29 (2016).

[63] T. N. Kipf and M. Welling. “Semi-supervised classification with graph convolutional networks”. In: *International Conference on Learning Representations*. 2017.

[64] Y. Luo, L. Shi, and X.-M. Wu. “Classic gnns are strong baselines: Reassessing gnns for node classification”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 97650–97669.

[65] M. Balcilar et al. “Analyzing the expressive power of graph neural networks in a spectral perspective”. In: *International Conference on Learning Representations*. 2021.

[66] Y. Ma et al. “A unified view on graph neural networks as graph signal denoising”. In: *Proceedings of the 30th ACM international conference on information & knowledge management*. 2021, pp. 1202–1211.

[67] T. K. Rusch, M. M. Bronstein, and S. Mishra. “A Survey on Oversmoothing in Graph Neural Networks”. In: *SAM Research Report* 2023 (2023).

[68] K. Oono and T. Suzuki. “Graph Neural Networks Exponentially Lose Expressive Power for Node Classification”. In: *International Conference on Learning Representations*. 2020.

[69] M. Chen et al. “Simple and deep graph convolutional networks”. In: *International conference on machine learning*. PMLR. 2020, pp. 1725–1735.

[70] C. Zheng et al. “Robust graph representation learning via neural sparsification”. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 11458–11468.

[71] Y. Koishekenov. “Reducing over-smoothing in graph neural networks using relational embeddings”. In: *The Ninth International Workshop on Deep Learning on Graphs: Method and Applications (DLG-AAAI’23)*. 2023.

[72] A. Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[73] J. You, R. Ying, and J. Leskovec. “Position-aware graph neural networks”. In: *International conference on machine learning*. PMLR. 2019, pp. 7134–7143.

[74] C. Ying et al. “Do transformers really perform badly for graph representation?” In: *Advances in neural information processing systems* 34 (2021), pp. 28877–28888.

[75] L. Rampášek et al. “Recipe for a general, powerful, scalable graph transformer”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 14501–14515.

[76] Q. Wu et al. “SGFormer: Single-Layer Graph Transformers with Approximation-Free Linear Complexity”. In: *arXiv preprint arXiv:2409.09007* (2024).

[77] M. Black et al. “Comparing graph transformers via positional encodings”. In: *Proceedings of the 41st International Conference on Machine Learning*. 2024, pp. 4103–4139.

[78] A. Schrijver. “On the history of the transportation and maximum flow problems”. In: *Mathematical programming* 91 (2002), pp. 437–445.

[79] S. He et al. “An overview on the application of graph neural networks in wireless networks”. In: *IEEE Open Journal of the Communications Society* 2 (2021), pp. 2547–2565.

[80] P. Tam et al. “Graph neural networks for intelligent modelling in network management and orchestration: a survey on communications”. In: *Electronics* 11.20 (2022), p. 3371.

[81] C. K. Joshi et al. “On the expressive power of geometric graph neural networks”. In: *International conference on machine learning*. PMLR. 2023, pp. 15330–15355.

[82] V. G. Satorras, E. Hoogeboom, and M. Welling. “E (n) equivariant graph neural networks”. In: *International conference on machine learning*. PMLR. 2021, pp. 9323–9332.

[83] Q. Sun et al. “Edge Feature Empowered Graph Attention Network for Sum Rate Maximization in Heterogeneous D2D Communication System”. In: *Neurocomputing* 616 (2025), p. 128883.

[84] M. Bouton et al. “Multi-agent reinforcement learning with graph q-networks for antenna tuning”. In: *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium*. IEEE. 2023, pp. 1–7.

[85] X. Zhang et al. “Scalable Power Control/Beamforming in Heterogeneous Wireless Networks with Graph Neural Networks”. In: *IEEE GLOBECOM* (2021).

[86] A. Narayanan et al. “Lumos5G: Mapping and predicting commercial mmWave 5G throughput”. In: *Proceedings of the ACM internet measurement conference*. 2020, pp. 176–193.

[87] W Hu et al. “Strategies For Pre-training Graph Neural Networks”. In: *International Conference on Learning Representations*. 2020.

[88] W. Jin et al. “Node similarity preserving graph convolutional networks”. In: *ACM WSDM*. 2021, pp. 148–156.

[89] Z. Hu et al. “GPT-GNN: Generative Pre-Training of Graph Neural Networks”. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD ’20. Virtual Event, CA, USA: Association for Computing Machinery, 2020, 1857–1867. ISBN: 9781450379984. DOI: 10.1145/3394486.3403237. URL: https://doi.org/10.1145/3394486.3403237.

[90] M. Lee, G. Yu, and G. Y. Li. “Graph embedding-based wireless link scheduling with few training samples”. In: *IEEE Transactions on Wireless Communications* 20.4 (2020), pp. 2282–2294.

[91] W. Ye et al. “5GNN: extrapolating 5G measurements through GNNs”. In: *Proceedings of the 1st International Workshop on Graph Neural Networking*. 2022, pp. 36–41.

[92] P. Ferrand et al. “Wireless channel charting: Theory, practice, and applications”. In: *IEEE Communications Magazine* 61.6 (2023), pp. 124–130.

[93] M. Wu, S. Pan, and X. Zhu. “OpenWGL: Open-World Graph Learning”. In: *2020 IEEE International Conference on Data Mining (ICDM)*. 2020, pp. 681–690. DOI: 10.1109/ICDM50108.2020.00077.

[94] A. Trask et al. “Neural arithmetic logic units”. In: *Advances in neural information processing systems* 31 (2018).

[95] A. Madsen and A. R. Johansen. “Measuring arithmetic extrapolation performance”. In: *arXiv preprint arXiv:1910.01888* (2019).

[96] A. Madsen and A. R. Johansen. “Neural arithmetic units”. In: *International Conference on Learning Representations*. 2020.

[97] K. Xu et al. “How neural networks extrapolate: from feedforward to graph neural networks”. In: *International Conference on Learning Representations (ICLR)*. 2021.

[98] G. Yehudai et al. “From local structures to size generalization in graph neural networks”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 11975–11986.

[99] A. Sanchez-Gonzalez et al. “Learning to simulate complex physics with graph networks”. In: *International conference on machine learning*. PMLR. 2020, pp. 8459–8468.

[100] F. Jiang et al. “Physics-informed neural networks for path loss estimation by solving electromagnetic integral equations”. In: *IEEE Transactions on Wireless Communications* (2024).

[101] S. Limmer, A. M. Alba, and N. Michailow. "Physics-informed neural networks for pathloss prediction". In: *2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE. 2023, pp. 1–5.

[102] Y. Li et al. "GASTO: A fast adaptive graph learning framework for edge computing empowered task offloading". In: *IEEE Transactions on Network and Service Management* 20.2 (2023), pp. 932–944.

[103] V. Elvira and L. Martino. "Advances in Importance Sampling". In: *Wiley StatsRef: Statistics Reference Online* (2014), pp. 1–14.

[104] K. Li et al. "Deep-graph-based reinforcement learning for joint cruise control and task offloading for aerial edge Internet of Things (EdgeIoT)". In: *IEEE Internet of Things Journal* 9.21 (2022), pp. 21676–21686.

[105] V. Mnih et al. "Playing Atari with Deep Reinforcement Learning". In: *NIPS Deep Learning Workshop 2013*. 2013.

[106] R. J. Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine learning* 8 (1992), pp. 229–256.

[107] J. Schulman et al. "Proximal policy optimization algorithms". In: *arXiv preprint arXiv:1707.06347* (2017).

[108] D. Silver et al. "Mastering the game of Go with deep neural networks and tree search". In: *nature* 529.7587 (2016), pp. 484–489.

[109] M. Arnold and G. Andersson. "Model predictive control of energy storage including uncertain forecasts". In: *Power systems computation conference (PSCC), Stockholm, Sweden*. Vol. 23. Citeseer. 2011, pp. 24–29.

[110] J. L. Barr and A. S. Manne. "Numerical experiments with a finite horizon planning model". In: *Indian Economic Review* 2.1 (1967), pp. 1–31.

[111] D. Ha and J. Schmidhuber. "World models". In: *arXiv preprint arXiv:1803.10122* (2018).

[112] J. You et al. "GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 5708–5717. URL: https://proceedings.mlr.press/v80/you18a.html.

[113] J. You et al. “Graph convolutional policy network for goal-directed molecular graph generation”. In: *Advances in neural information processing systems* 31 (2018).

[114] A. Sanchez-Gonzalez et al. “Graph networks as learnable physics engines for inference and control”. In: *International conference on machine learning*. PMLR. 2018, pp. 4470–4479.

[115] C. Shan et al. “Reinforcement learning enhanced explainer for graph neural networks”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 22523–22533.

[116] X. Jia et al. “Graph-based reinforcement learning for active learning in real time: An application in modeling river networks”. In: *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*. SIAM. 2021, pp. 621–629.

[117] J. Shen et al. “Content Caching-Assisted Vehicular Edge Computing Using Multi-Agent Graph Attention Reinforcement Learning”. In: *IEEE Transactions on Vehicular Technology* (2024).

[118] M. Ji et al. “Graph Neural Networks and Deep Reinforcement Learning-Based Resource Allocation for V2X Communications”. In: *IEEE Internet of Things Journal* (2024).

[119] V. P. Dwivedi et al. “Long range graph benchmark”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 22326–22340.

[120] P. Agarwal et al. “Transformers in reinforcement learning: a survey”. In: *arXiv preprint arXiv:2307.05979* (2023).

[121] L. Chen et al. “Decision transformer: Reinforcement learning via sequence modeling”. In: *Advances in neural information processing systems* 34 (2021), pp. 15084–15097.

[122] C. Kim et al. “Dynamics-Augmented Decision Transformer for Offline Dynamics Generalization”. In: *3rd Offline RL Workshop: Offline RL as a”Launchpad”*.

[123] H. Furuta, Y. Matsuo, and S. S. Gu. “Generalized decision transformer for offline hindsight information matching”. In: *International Conference on Learning Representations*.

[124] M. Bechler-Speicher et al. “Graph neural networks use graphs when they shouldn’t”. In: *Forty-first International Conference on Machine Learning*. 2024.

[125] Y. Zheng et al. “Rethinking Structure Learning For Graph Neural Networks”. In: *arXiv preprint arXiv:2411.07672* (2024).

[126] K. Gupta et al. “LayoutTransformer: Layout Generation and Completion with Self-attention”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 1004–1014.

[127] A. K. Erlang. “The theory of probabilities and telephone conversations”. In: *Nyt. Tidsskr. Mat. Ser. B* 20 (1909), pp. 33–39.

[128] D. Gamarnik. *Algorithmic challenges in the theory of queueing networks*. https://www.mit.edu/~gamarnik/Papers/YEQT-Tutorial.pdf. Tutorial notes from Young European Queueing Theorists (YEQT) workshop. Notes prepared by Shubham Gupta. 2011.

[129] F. Harary and R. Z. Norman. “Some properties of line digraphs”. In: *Rendiconti del circolo matematico di palermo* 9 (1960), pp. 161–168.

[130] T. Rashid et al. “Monotonic value function factorisation for deep multi-agent reinforcement learning”. In: *Journal of Machine Learning Research* 21.178 (2020), pp. 1–51.

[131] M. Disney, P Lewis, and P. North. “Monte Carlo ray tracing in optical canopy reflectance modelling”. In: *Remote Sensing Reviews* 18.2-4 (2000), pp. 163–196.

[132] G. Mateos et al. “Connecting the dots: Identifying network structure via graph signal processing”. In: *IEEE Signal Processing Magazine* 36.3 (2019), pp. 16–43.

[133] Z. Gao and D. Gündüz. “Graph Neural Networks over the Air for Decentralized Tasks in Wireless Networks”. In: *IEEE Transactions on Signal Processing* (2025).

[134] S. Fiorellino, C. Battiloro, and P. Di Lorenzo. “Topological neural networks over the air”. In: *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2024, pp. 12986–12990.

[135] A. Arnaiz-Rodrıguez et al. “DiffWire: Inductive Graph Rewiring via the Lovász Bound”. In: *Learning on Graphs Conference*. PMLR. 2022, pp. 15–1.

[136] S. Yun et al. “Graph transformer networks”. In: *Advances in neural information processing systems* 32 (2019).

Appendix A

Included papers