Journal of Trends in Computer Science and Smart Technology (ISSN: 2582-4104)
www.irojournals.com/tcsst/

Context-Aware Multi-Modal Graph Attention Fusion Network for Adaptive Resource Allocation in Wireless Networks

Anoop Mohanakumar[1], Uma Shankari Srinivasan[2], Judy Simon[3], Nellore Kapileswar[4], Sutha K.[5*]

[1]Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, SIMATS, Chennai, India.

[2,5]Department of Computer Science and Applications, Faculty of Science and Humanities, SRM Institute of Science and Technology, Ramapuram, Chennai, India.

[3,4]Department of Electronics and Communication Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, SIMATS, Chennai, India

Email: [1]anoopm.sse@saveetha.com, [2]umabalajees@gmail.com, [3]judysimon.sse@saveetha.com, [4]kapileswarn.sse@saveetha.com, [5*]ksutha1986@gmail.com

Abstract

Resource allocation in wireless networks is an important factor as it defines e the utilization of spectrum usage, resource distribution, and quality of services. The evolution of mobile communication brings additional challenges in allocating the resources due to high user mobility, heterogenous traffic demands, and dynamic topologies. Conventional techniques lag in performance due to their static optimization procedures and limited spatial-temporal awareness. To overcome this, a Spatio-Attentive Graph Mixture Network (SAGMNet) is proposed in this research work for enhanced resource management. The proposed model incorporates graph-based learning with a multi-modal attention mechanism for feature processing and scheduling decisions. The experimental analysis of the proposed model utilizes benchmark vehicular wireless scheduling dataset and evaluates the model's performance with different metrics like spectrum utilization, throughput, and latency. The proposed model exhibits superior performance in terms of 93.6% spectrum utilization efficiency, 29.1 Mbps average throughput, 0.087 interference index, 3.26 Mbps/Watt energy efficiency, 0.961 scheduling fairness, 5.9ms allocation latency, 0.928 mobility robustness score, and 3.2 ms

DOI: https://doi.org/10.36548/jtcsst.2025.2.008
Received: 28.05.2025, received in revised form: 26.06.2025, accepted: 10.07.2025, published: 19.07.2025

inference time, which is better than conventional DNN, GCN, LSTM, ST-GCN, and Transformer-GAT models.

Keywords: Context-Aware Scheduling, Spatio-Temporal Graph Learning, Adaptive Resource Management, Intelligent Wireless Networks, Dynamic Topology Optimization, Mobility-Robust Allocation.

1. Introduction

Adaptive resource allocation plays an important role in modern wireless communication systems due to the demand for low latency and high-speed reliable connectivity. The increased usage of mobile devices, adaptation of IoT systems, and advancements in vehicular communication require dynamic adaptive procedures to handle real-time network variations [1]. The adaptive resource allocation process provides intelligent distribution of resources like bandwidth and power based on user mobility, channel conditions, and interference levels [2]. Additionally, the adaptive resource allocation procedure optimizes spectral efficiency and energy utilization in dynamic environments. Specifically, adaptive resource allocation ensures network robustness and effectiveness in networks that experience frequency topology changes [3]. Moreover, adaptive resource allocation is essential for the evolving 5G and 6G systems, which process real-time data and advanced decision models to provide consistent quality of service to users [4].

Apart from its importance, incorporating adaptive resource allocation in wireless networks brings numerous challenges. One significant challenge is the accurate modeling of the dynamic nature of wireless environments. Since users in dynamic environments move randomly which leads to reduced signal quality due to interference, congestion, and multipath fading [5]. Another challenge in adapting resource allocation in latency-sensitive applications is the requirement for instantaneous decisions making, r especially when deployed in edge or distributed systems. Most of the existing approaches developed so far face limitations in balancing the multiple objectives such as throughput, fairness, and energy efficiency [6-7]. Moreover, the computational complexity of real-time optimization limits the model's practical adaptability and scalability. These limitations highlight the need for an intelligent, lightweight model with the ability to adapt to high mobility and resource-constrained environments. Existing optimization models based on the Markov decision process [8] and game theory [9] assume an ideal environment for analysis. They also require complete knowledge about

network status and user distribution, which is not feasible in real-time environments. Better interpretability and accommodation of dynamic changes in large-scale scenarios are necessary. In some cases, rule-based heuristics and statistical learning procedures are adopted to attain better performance; however, these systems require predefined feature mappings. Additionally, these techniques lack the flexibility to incorporate real-time data, making them unsuitable for networks such as vehicular networks.

Recent learning-based models, specifically deep learning, graph learning, and attention mechanisms, exhibit better solutions for resource allocation problems. Graph Convolutional Networks and their variants, like Graph Attention Networks, are utilized to model network topologies for extracting spatial features [10]. Recurrent networks such as LSTM are used to model the temporal variations in traffic load and mobility. A few hybrid models combine attention modules with transformers to demonstrate the model's ability to process long-term dependencies. However, these models focus either on spatial or temporal features, which results in limited performance in highly dynamic environments. Moreover, the computational complexity and scalability limit their deployment in real-time environments.

From the brief summary, the research work identifies several questions as follows:

- How can a model effectively process the spatial and temporal dynamics in a mobile wireless network?
- Is there a better architecture that can be developed to perform adaptive fusion so that multimodal inputs like data mobility, signal quality, and network topology can be considered and processed?
- What is the possibility of developing a single module to enable a better balance between different objective functions? If developed, how would the model perform well in a dynamic environment with limited resources?

Considering the above research questions, the proposed research framed an objective to develop a lightweight context-aware architecture for allocating optimal resources in a dynamic wireless network. The core objective is to design a learning model that integrates spatio-temporal features through a novel Spatio-Attentive Graph Mixture Network (SA-GMNet). The proposed approach incorporates a multi-head attention mechanism with a graph mixture layer to combine multiple graph views dynamically for capturing the local and global

connectivity among users. Additionally, the proposed SA-GMNet incorporates spatio-temporal awareness, which makes it highly responsive to user mobility and channel variability. The novelty of the research work lies in its novel fusion of heterogeneous modalities such as spatial, temporal, and contextual features. The primary contributions of this work are summarized as follows:

- Proposed a novel deep learning model, SA-GMNet, for adaptive resource allocation in dynamic wireless environments. The proposed SA-GMNet incorporates a graph mixture layer to provide spatial modeling while capturing the multi-view topologies. Additionally, an attention-based sequence modeling is presented to effectively encode the temporal dependencies and user mobility patterns.
- A detailed experimental analysis is presented using a benchmark vehicular wireless scheduling dataset to validate the proposed model's performance under realistic conditions.
- Experimental results exhibit the better performance of the proposed SA-GMNet with 93.6% spectrum efficiency, 29.1 Mbps throughput, 0.087 interference index, and 3.2 ms inference time, which is better than conventional models.
- The remaining discussions are arranged in the following order: Section 2 provides the literature review of existing research works. Section 3 presents the proposed work, and Section 4 presents the results with relevant discussion. The conclusion is presented in the last section.

2. Related Works

A brief literature review of existing resource allocation approaches is considered for analysis, and the identified limitations are presented as a research summary in this section. A combination of multilayer perceptron with decision tree is presented in [11] to enable energy-efficient access points in wireless networks. Additionally, the analysis incorporates optimization algorithms like PSO and GA to fine-tune the ML model hyperparameters. However, the lack of low latency and high packet loss are the limitations of the presented model when applied in dynamic network traffic conditions.

The distributed resource allocation model presented in [12] incorporates a modified Kuramoto strategy formulated based on node specific weights. The presented model performs allocation considering the dynamic QoS demands. It also adjusts the phase differences between nodes while mapping the time slots in a TDMA system and attains efficient bandwidth utilization. An adaptive resource allocation algorithm is presented in [13] for 5G vehicular cloud communication networks. The presented model integrates in-band and out-of-band communication modes with single-hop and multi-hop configurations to evaluate transmission performance under varying network conditions. The adaptive mechanism utilizes an objective function to optimize allocation dynamically based on current network states. However, the presented model assumes idealized conditions such as fixed communication radii and fully connected networks, which may not fully reflect real-world vehicular scenarios with high mobility and diverse interference patterns.

The resource allocation procedure reported in [14] for next-generation wireless networks integrates Convolutional Neural Networks (CNN) with Game Theory. The presented model utilizes a hierarchical radio resource management architecture in which local radio resource managers handle subchannel distribution and reduce the overhead on central controllers. The CNN efficiently extracts network slicing patterns, and Game Theory optimizes resource distribution among competing slices. An optimal resource allocation strategy is presented in [15] for 5G networks to enable efficient coexistence of cellular and device-to-device (D2D) communication. The approach adopts a two-layer game-theoretic matching algorithm in which the first layer assigns channels to cellular users using a many-to-one matching model. The second layer allocates D2D resources based on many-to-many matching. Additionally, a utility function based on Quality of Experience is used to guide the optimization to attain better user satisfaction and throughput.

The joint resource allocation and power control algorithm presented in [16] for device-to-device D2D communications in 5G networks provides enhanced energy efficiency under quality-of-service constraints. The model includes a modified particle swarm optimization PSO framework to handle both discrete and continuous decision variables. The hybrid update mechanism used in the presented model for subchannel allocation and transmit power matrices preserves network constraints. However, the approach assumes perfect channel state information and static user distribution, limiting its real-time applicability in dynamic and mobile environments. The distributed multi-agent deep reinforcement learning algorithm

presented in [17] solves the joint problem of mode selection and channel allocation in device-to-device D2D communications. The presented Deep Q-Network maximizes system sum-rate while satisfying user Quality of Service QoS constraints, particularly in environments utilizing both millimeter wave and traditional cellular bands. However, the model's assumptions regarding fixed mobility and simplified channel conditions limit its real-world scalability.

The federated learning framework presented in [18] for edge-assisted Device-to-Device D2D communication networks addresses computational and communication heterogeneity among edge nodes. The presented model introduces a D2D offloading mechanism that reallocates data samples between edge devices based on their resource capacities before federated training begins. The methodology transforms a non-convex optimization problem into a convex one and attains an optimal solution using the CVX optimization tool, which is later rounded to find the best discrete task distribution. Simulation using the benchmark dataset shows that the presented model enhances training efficiency and reduces total system time, particularly for larger aggregation rounds. However, the study assumes static channel conditions, which limit its applicability in practical applications.

The resource allocation model presented in [19] utilizes federated learning to utilize the computational features and security of individual terminals. The model allocates resources considering both macro and micro base stations and assigns subcarriers in an optimal manner. The experimental results validate the model’s better performance over conventional approaches. An allocation model presented in [20] incorporates deep reinforcement learning, in which a deep Q-network is used for guaranteed exploration and model convergence in resource allocation. The presented model also considers the various priority levels of users to enhance service quality. The simulation analysis exhibits the model’s better convergence rate, stability, and reduce complexity compared to existing methods. Another resource allocation model presented in [21] for cognitive radio networks optimizes the task scheduling and allocation process by formulating the long-term average system cost. Through the proposed Lyapunov optimization, the presented approach identifies the limits, and by utilizing deep reinforcement learning, the model effectively allocates resources in a better manner compared to conventional methods.

Table 1. Summary of Literature Review

Ref	Methodology	Advantages	Limitations
[11]	MLP and Decision Tree with optimization	Improved energy-focused resource prediction and evolutionary tuning	Lacks in performance due to high packet loss and delayed responses in dynamic network loads
[12]	Modified Kuramoto-based TDMA slot mapping	Adapts bandwidth based on real-time QoS with node-weight awareness	Restricted to single-hop setups and ignores multi-hop routing and energy constraints
[14]	CNN with Game Theory	Learns network for better resource processing across users with decentralized control	Tested under static traffic and exhibit limited responsiveness
[15]	Dual-layer game matching for cellular and D2D coexistence	Enhanced QoE with layered allocation structure	Assumes fixed transmission power and no mobility variance
[16]	PSO-enhanced joint power and subchannel allocation	Balanced energy efficiency with spectrum reuse	Requires perfect channel estimates and unsuitable for dynamic traffic
[17]	Multi-agent DRL	Improved capacity and user satisfaction	Simplified channel model which lacks adaptability for realistic user motion
[18]	Federated learning	Enhanced distributed training and reduced total system time	Limited responsiveness to runtime variability

Research Gap

The brief literature analysis highlights the critical research gaps in the domain of adaptive resource allocation for wireless and IoT-enabled networks. Many existing approaches dependon static or idealized assumptions such as fixed mobility patterns, perfect channel conditions, or centralized control, which limit scalability and responsiveness in real-world dynamic environments. In a few research works, reinforcement learning and game-theoretic frameworks are used; however, they often neglect the joint modeling of spatial, temporal, and contextual factors crucial for real-time adaptation. The existing models mainly focus on energy efficiency and throughput without ensuring fairness or latency guarantees, which is more essential for heterogeneous and delay-sensitive applications. Additionally, the existing methodologies often lack multi-modal data fusion as they depend on single-dimensional optimization strategies. This leads to issues in allocation performance as it fails to capture the complexity of modern network topologies. These limitations highlight the need for a unified, lightweight, and context-aware model that can intelligently learn from spatial and temporal variations while adapting decisions in real-time precisely the motivation behind the proposed SA-GMNet framework.

The proposed SA-GMNet addresses these limitations through a combined learning module that fuses multi modal data. The incorporation of spatial graphs, temporal sequences and contextual signals can enable the learning of mobility patterns, dynamic interferences, and dynamic traffic behaviors. Specifically, the graph mixture module considers multiple spatial features and temporal attention predicts the future states based on previous data. The complete architecture will provide better fairness and responsiveness with low computational overhead which is an improvement over conventional method.

3. Proposed Work

The proposed work, SA-GMNet, is a combined deep learning framework designed to perform adaptive resource allocation in wireless networks by integrating spatio-temporal attention with graph-based topology modeling. This architecture dynamically responds to mobility, interference, and traffic variations, enabling efficient, fair, and low-latency scheduling decisions in real-time environments. Before introducing the mathematical model for the proposed approach, the input modalities considered in the proposed work are mathematically formulated. The SA-GMNet model requires a wide range of input features that

describe the dynamic behavior of nodes and links in a wireless network. These features are encoded as node-level and edge-level data which can be captured over time and organized into structured tensors to feed the learning components. First the device mobility trajectories are considered for the nodes. Each node v_i in the wireless network possesses a mobility state defined by its spatial coordinates and velocity. This can be mathematically represented as a time-series vector as follows

$$m_i^t = [x_i^t, y_i^t, s_i^t] \tag{1}$$

where $x_i^t, y_i^t \in R$ indicates the Cartesian coordinates of the node i at time t, $s_i^t \in R$ indicates the instantaneous speed of the node i at time t. These values capture mobility-related dynamics, which are crucial for modeling topology changes and handover behavior in mobile environments. Second channel state information is considered. The channel characteristics are encoded as a vector that reflects link quality over time and frequency. For each node i and its neighbor, j the CSI at time t is mathematically expressed as

$$c_{ij}^t = h_{ij}^t(f_1), h_{ij}^t(f_2), \dots, h_{ij}^t(f_F) \tag{2}$$

where $h_{ij}^t(f_k) \in C$ indicates the complex channel gain between nodes i and j at frequency sub-band f_k, F indicates the total number of frequency sub-bands considered. CSI provides a high-resolution view of link reliability and frequency selectivity, essential for adaptive spectrum allocation. However, in practical systems, the CSI vector is affected by variations in link quality, which occur due to user mobility, interference, and fading issues. These changes introduce temporal instability and noise that should be carefully addressed in the resource allocation process. Since CSI changes over frequency bands, it requires adaptive instability considering the spatial and temporal features. Thus, the proposed model incorporates graph network which effectively handles the degraded link quality. The temporal attention encoder captures the long- and short-term CSI variations, which enable better allocation even under dynamic environments. Next, the Signal-to-Noise Ratio (SNR) is considered, in which each node observes its SNR from communication with a base station or peer node. Mathematically, it is expressed as

$$SNR_i^t = \frac{P_{rx,i}^t}{N_0 + I_i^t} \tag{3}$$

where $P_{rx,i}^t$ indicates the received signal power at node i, N_0 indicates the noise power spectral density, I_i^t indicates the aggregate interference experienced by node i at time t. SNR is used as a primary indicator for link quality and serves as a basis for modulation and coding decisions. To model traffic behavior over time each node maintains a buffer or load metric that tracks pending data. Mathematically, it is expressed as

$$L_i^t = \sum_{\tau=0}^{W} \delta_i^{t-\tau} \tag{4}$$

where $L_i^t \in R_+$ indicates the cumulative traffic load at the node i at time, t, $\delta_i^{t-\tau}$ indicates the data arrival at time $t - \tau$, W indicates the time window for accumulation. This metric informs the model of demand intensity, guiding fair scheduling and congestion-aware allocation. Furthermore, the interference profile is defined using the spatial and spectral interference experienced by a node. Mathematically, it is quantified as

$$I_i^t = \sum_{j \neq i} \frac{P_j^t G_{ji}^t}{d_{ji}^{\gamma}} \tag{5}$$

where P_j^t indicates the transmit power of interfering node j, G_{ji}^t indicates the channel gain from node j to node i, d_{ji} indicates the distance between nodes j and i, γ indicates the path loss exponent. This feature allows the model to reason about spectral congestion and selects less interfered links. While constructing the topology graph, the connectivity structure is encoded in a dynamic adjacency matrix A_t. Mathematically, it is expressed as

$$A_t = \begin{cases} 1 & if\ a\ direct\ link\ exists\ between\ i\ and\ j\ at\ time\ t \\ 0 & Otherwise \end{cases} \tag{6}$$

Additionally, edge weights are introduced to represent link quality, which is mathematically formulated as

$$W_t(i,j) = f\left(SNR_{ij}^t, d_{ij}, c_{ij}^t\right) \tag{7}$$

where $f(\cdot)$ indicates the function that combines the signal strength, distance, and CSI to determine the reliability of the link. Considering all the above features, the final input tensor is constructed to obtain a final feature vector for each node. Mathematically, it is formulated as

$$x_i^t = [m_i^t || SNR_i^t || L_i^t || I_i^t] \tag{8}$$

The full graph at time t is described by (X_t, A_t, W_t), where $X_t \in R^{N \times d}$ is the node feature matrix and N is the number of nodes. This structured input allows the proposed SA-GMNet to process heterogeneous data types through graph-based and attention-based operations. The complete overview of the proposed model process flow is presented in Figure 1.

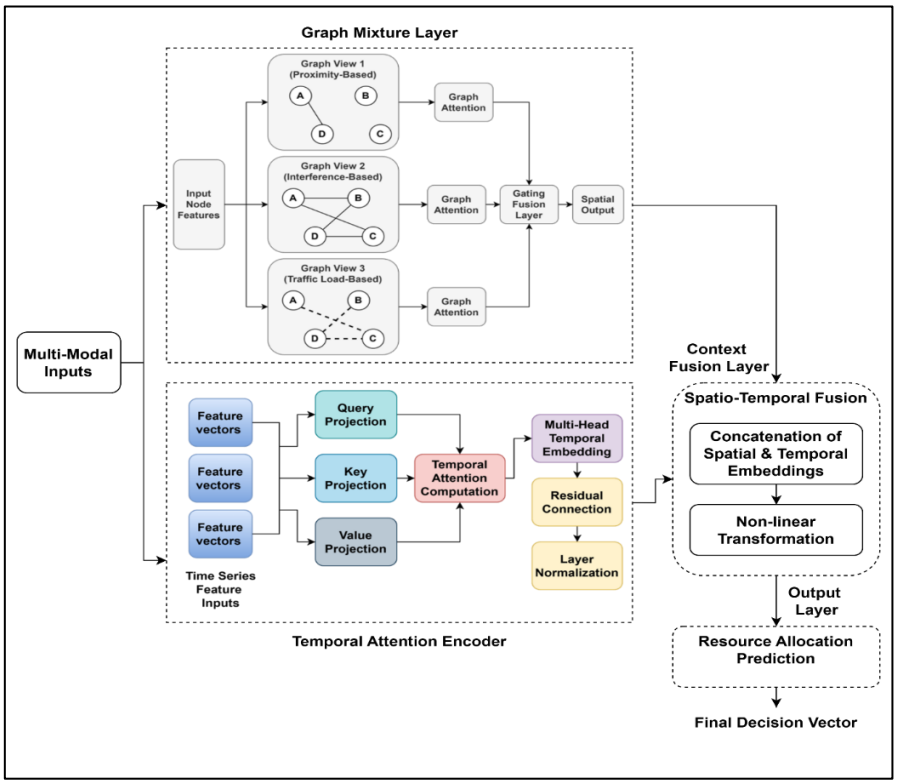

Figure 1. Process Flow of Proposed Model

3.1 Graph Mixture Layer

The Graph Mixture Layer in SA-GMNet is designed to extract spatial representations from multiple topological perspectives within the wireless network. Unlike traditional graph neural networks that depend on a single static graph structure, this layer introduces a mixture of graphs to capture diverse connectivity patterns such as proximity, interference, mobility similarity, or traffic load correlation. Each of these perspectives is encoded as a separate graph view $G_t^j = (V_t, E_t^j)$, where $j \in 1,2, \ldots, k$ represents the graph index, and k is the total number of graph views.

For each graph G_t^j, the neighborhood information around node i is processed using a graph attention mechanism. The intermediate embedding for node i in graph j is mathematically expressed as

$$h_i^{(j)} = \sigma\left(\sum_{v_k \in N_i^j} \alpha_{ik}^{(j)} W^{(j)} x_k\right) \tag{9}$$

where $h_i^{(j)} \in R^d$ indicates the output embedding of node i for graph view j, N_i^j indicates the set of neighbors of node i in graph G_t^j, $W^{(j)} \in R^{d \times d}$ indicates the learnable weight matrix for graph j, $\sigma(\cdot)$ indicates the non-linear activation function such as ReLU, $\alpha_{ik}^{(j)}$ indicates the normalized attention score between node i and neighbor k in graph j. The attention coefficients $\alpha_{ik}^{(j)}$ are then computed using a shared attention mechanism that considers the similarity between nodes i and k. Mathematically, it is formulated as

$$\alpha_{ik}^{(j)} = \frac{exp\left(LeakyReLU\left(a^\top [W^{(j)} x_i | W^{(j)} x_k]\right)\right)}{\sum_{v_l \in N_i^j} exp\left(LeakyReLU\left(a^\top [W^{(j)} x_i | W^{(j)} x_l]\right)\right)} \tag{10}$$

where $a \in R^{2d}$ indicates the shared attention vector, $|$ indicates the vector concatenation operator, $LeakyReLU(\cdot)$ indicates the activation function to stabilize gradients during learning. Once the embeddings for each graph view are computed, they are combined into a fused spatial representation through a learnable gating mechanism, which is mathematically formulated as

$$h_i^{spatial} = \sum_{j=1}^{k} g_j \cdot h_i^{(j)} \tag{11}$$

where $h_i^{spatial} \in R^d$ indicates the final spatial embedding of node i, $g_j \in [0,1]$ indicates the soft gating coefficient for graph view j, $\sum_{j=1}^{k} g_j = 1$ ensures that the mixture remains a convex combination. The gating vector $g = [g_1, \ldots, g_k]$ is derived through a softmax transformation of a trainable parameter vector $z \in R^k$. Mathematically, it is expressed as

$$g_j = \frac{exp(z_j)}{\sum_{l=1}^{k} exp(z_l)} \tag{12}$$

This adaptive mixture allows the proposed model to selectively highlight the most informative graph views based on the training objective, network state, and node context. Thus,

the Graph Mixture Layer in SA-GMNet allows for generalization across diverse spatial conditions and dynamically integrates heterogeneous topological information. This flexibility enhances its capacity to make accurate and fair scheduling decisions in a wide range of wireless environments.

3.2 Temporal Attention Encoder

The Temporal Attention Encoder in the proposed SA-GMNet captures dynamic temporal patterns across past observations of each network node. In wireless environments, device behavior, channel quality, and traffic demand often fluctuate over time. Therefore, modeling this temporal variation is essential for making adaptive and context-aware scheduling decisions. To achieve this, the encoder processes sequential input features for each node using a multi-head self-attention mechanism, allowing the model to learn temporal relevance across a sliding window of observations. Let each node v_i be associated with a time-series input $x_i^{t-T+1}, x_i^{t-T+2}, \dots, x_i^t$, where T denotes the temporal window length, and $x_i^{t-\tau} \in R^d$ indicates the feature vector of the node i at time step $t-\tau$. These vectors are stacked to form a temporal sequence matrix $X_i^t \in R^{T\times d}$. The encoder first projects each input into three separate vector spaces—queries, keys, and values using learned linear transformations, which is mathematically, formulated as follows.

$$Q_i^t = X_i^t W_Q, \quad K_i^t = X_i^t W_K, \quad V_i^t = X_i^t W_V \tag{13}$$

where $W_Q, W_K, W_V \in R^{d\times d_a}$ indicates the learnable projection matrices, $Q_i^t, K_i^t, V_i^t \in R^{T\times d_a}$ indicates the query, key, and value matrices, d_a indicates the attention dimension. Further, the temporal attention scores are computed using scaled dot-product attention, which quantifies the importance of each past time step relative to others in the sequence. Mathematically, it is formulated as:

$$A_i^t = softmax\left(\frac{Q_i^t {K_i^t}^{\top}}{\sqrt{d_a}}\right) \tag{14}$$

where $A_i^t \in R^{T\times T}$ indicates the attention weight matrix, $softmax(\cdot)$ ensuring each row in A_i^t sums to 1, assigning relative importance to each timestep. The attention-weighted temporal representation is mathematically expressed as:

$$H_i^t = A_i^t V_i^t \tag{15}$$

where $H_i^t \in R^{T \times d_a}$ contains the context-aware output vectors that reflect weighted contributions from all time steps. To enhance the model's capacity to capture multiple types of temporal dependencies, multi-head attention is applied. For M parallel attention heads, each with its own parameter set, the individual outputs $H_i^{t,1}, H_i^{t,2}, \ldots, H_i^{t,M}$ are concatenated and linearly transformed. Mathematically, it is expressed as:

$$\hat{h}_i^{temporal} = Concat\left(H_i^{t,1}, \ldots, H_i^{t,M}\right) W_O \tag{16}$$

where $W_O \in R^{M d_a \times d}$ indicates the output projection matrix, $\hat{h}_i^{temporal} \in R^d$ indicates the final temporal embedding for node i. This representation effectively captures both periodic and irregular variations in traffic, channel quality, and mobility trends over time. To further stabilize learning and enhance generalization, residual connections and layer normalization are optionally applied, which is mathematically formulated as:

$$h_i^{temporal} = LayerNorm\left(X_i^t + \hat{h}_i^{temporal}\right) \tag{17}$$

Through this mechanism, the Temporal Attention Encoder allows SA-GMNet to focus on the most relevant historical states while down-weighting the less informative ones. The temporal representation of the network is then passed into the context fusion layer to obtain the final feature vector.

The temporal attention encoder in the proposed work analyzes the traffic demands, resource availability, and topological behaviors over time. The encoder module dynamically assigns importance to each time step and focuses on the most important features. The temporal attention unit analyzes the time dependent feature maps. Specifically, it analyzes the time dependent feature maps which contributes contextual information such as mobility, quality of the channel and interference patterns in dense traffic. The attention mechanism calculates the weight for each time point to determine the current resource allocation decisions. The weights are obtained by a function which compares the past feature representation with learnable query which represents the current temporal features. Finally, all the weighted features are combined through the fusion process to obtain a single output that provides the most useful information from the sequence. This process helps the model focus on significant changes in the network instead of considering all time steps as equal. Thus, it improves the accuracy and timing of resource allocation decisions. The temporal attention module enhances the efficiency and precision with reduced impact of noises. This module enhances the model ability to allocate

bandwidth and ensures the stable responsiveness for short and long variations in the wireless environment.

3.3 Context Fusion Layer

The Context Fusion Layer in the SA-GMNet is basically an integration module that combines the spatial and temporal representations of each network node into a combined feature vector. This operation is essential for capturing the combined influence of topological relationships and dynamic behavior on wireless resource demands. While the Graph Mixture Layer captures spatial dependencies and the Temporal Attention Encoder models sequential trends, the fusion layer combines this information to form a context-aware embedding suitable for decision-making. Consider the output of the graph mixture layer for node v_i be denoted as $h_i^{spatial} \in R^{d_s}$, and the temporal attention output as $h_i^{temporal} \in R^{d_t}$. These two vectors are concatenated to form an extended representation which is mathematically expressed as

$$z_i = \left[h_i^{spatial} || h_i^{temporal}\right] \in R^{d_s+d_t} \tag{18}$$

where $||$ indicates the concatenation operator. This intermediate vector z_i contains comprehensive context for node i, encompassing both localized structural information and time-evolving patterns. To process this joint embedding, the model applies a linear transformation followed by a non-linear activation which is mathematically formulated as

$$\tilde{h}_i = ReLU\left(W_f z_i + b_f\right) \tag{19}$$

where $W_f \in R^{d_f \times (d_s+d_t)}$ indicates the learnable weight matrix for feature compression, $b_f \in R^{d_f}$ indicates the bias term for the transformation, $\tilde{h}_i \in R^{d_f}$ indicates the fused output vector for node i, $ReLU(\cdot)$ indicates Rectified Linear Unit activation function that introduces non-linearity and prevents vanishing gradients. The output $\widetilde{h_i}$ represents a condensed, high-level context vector that encapsulates the most informative features from both spatial and temporal domains. This vector forms the basis for downstream predictions such as resource allocation, power control, or scheduling decisions. In scenarios where different applications or tasks are to be addressed the fusion layer are extended to support task-specific projections. In such a case, separate transformation is applied which is mathematically expressed as

$$\tilde{h}_i^{(t)} = \phi\left(W_f^{(t)} z_i + b_f^{(t)}\right), \quad t \in 1,2,\dots,T \tag{20}$$

where $W_f^{(t)} \in R^{d_f \times (d_s + d_t)}$ indicates the task-specific weight matrix, $b_f^{(t)}$ indicates the task-specific bias term, $\phi(\cdot)$ indicates the optional task-specific non-linearity, T indicates the total number of output prediction tasks. To ensure stable learning and faster convergence, the model include dropout regularization and layer normalization after fusion which is mathematically expressed as

$$h_i^{fused} = LayerNorm\left(Dropout\left(\tilde{h}_i\right)\right) \tag{21}$$

This final fused embedding h_i^{fused} is passed to the output layer for generating allocation decisions that consider both the structural configuration of the network and the historical behavior of each user. In the proposed model, different graphs are used to understand the network nodes and their relationships. The graph structures include mobility, interference levels and traffic patterns through unique graphs that have connected nodes. The graphs first consider each process individually and perform their neural operations to extract the features that indicate its specific connectivity. Further the attention-based fusion learns the importance of each graph view and provides its current decision by assigning weights to each graph view. The weighted features are then combined to obtain a combined graph so that the model adaptively select the connections based on the mobility and load conditions. Thus, using this process, the proposed model obtains complete details about the network which further improves its ability to allocate resources in dynamic environments.

3.4 Output Layer

The Output Layer in SA-GMNet process the fused context representation of each node into actionable predictions for wireless resource allocation. These predictions include spectrum block assignment, transmission power level selection, and scheduling flags personalized to the primary application objective. The output layer processes the final fused embedding $h_i^{fused} \in R^{d_f}$ produced by the Context Fusion Layer through a set of trainable linear projections to produce decision scores or allocation probabilities. The output for each node v_i is computed as

$$y_i = W_o h_i^{fused} + b_o \tag{22}$$

where $W_o \in R^{m \times d_f}$ indicates the weight matrix that maps the fused feature vector to the decision space, $b_o \in R^m$ indicates the bias term, $y_i \in R^m$ indicates the raw output vector containing scores for m decision classes. For classification tasks such as assigning a specific

channel from a predefined set, the raw outputs are passed through a softmax activation to obtain normalized probability distributions which is mathematically expressed as

$$\hat{y}_i^{(c)} = \frac{exp\left(y_i^{(c)}\right)}{\sum_{j=1}^{m} exp\left(y_i^{(j)}\right)} \quad \forall c \in 1,2,\dots,m \tag{23}$$

where $\hat{y}_i^{(c)}$ indicates the probability of node i being assigned to class c, m indicates the total number of discrete allocation options. For multi-task learning the model predicts multiple allocation parameters simultaneously in that case, multiple output heads are used. In this scenario, each task $t \in 1,\dots,T$ has its own output layer which is mathematically expressed as

$$y_i^{(t)} = W_o^{(t)} h_i^{fused} + b_o^{(t)} \tag{24}$$

where $W_o^{(t)} \in R^{m_t \times d_f}$ indicates the task-specific weight matrix, $b_o^{(t)} \in R^{m_t}$ indicates the task-specific bias, $y_i^{(t)} \in R^{m_t}$ indicates the prediction vector for task t. Each task output is evaluated against its ground truth label using an appropriate loss function, and the total training loss is defined as a weighted combination of all task-specific objectives which is mathematically expressed as

$$L_{total} = \sum_{t=1}^{T} \lambda t \; Lt\left(y_i^{(t)}, \hat{y}_i^{(t)}\right) \tag{25}$$

where λ_t indicates the weight assigned to task t, L_t indicates the loss function used for task t, $\hat{y}_i^{(t)}$ indicates the ground truth for task t. The output layer is designed to be modular and lightweight, making it suitable for real-time inference in resource-constrained environments.

3.5 Loss Function

The loss function in the SA-GMNet architecture is constructed to optimize multiple performance criteria simultaneously which ensures that the model does not only learn to allocate resources accurately but also considers fairness, efficiency, and real-time responsiveness. Since the model handles both classification and regression tasks such as spectrum assignment, power level prediction, and scheduling decisions the overall loss function integrates task-specific objectives along with domain-specific regularization terms. The main objective is formulated as

$$L_{total} = L_{alloc} + \lambda_1 L_{fair} + \lambda_2 L_{energy} \quad (26)$$

where L_{alloc} indicates the primary task loss, L_{fair} indicates the auxiliary loss to promote fairness in scheduling, L_{energy} indicates the penalty term to discourage energy-inefficient decisions, $\lambda_1, \lambda_2 \in R_+$ indicates the weighting coefficients to balance the influence of each term. The allocation L_{alloc} varies depending on whether the target task is for scheduling classification tasks. the categorical cross-entropy loss is used which is mathematically expressed as

$$L_{alloc} = -\sum_{i=1}^{N}\sum_{c=1}^{m} y_i^{(c)}\, log\left(\hat{y}_i^{(c)}\right) \quad (27)$$

where N indicates the total number of nodes devices, m indicates the number of output classes, $y_i^{(c)}$ indicates the ground truth one-hot encoded for node i, $\hat{y}_i^{(c)}$ indicates the predicted probability for class c. For continuous variables like transmission power, a regression loss such as mean squared error is used which is mathematically expressed as

$$L_{alloc} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (28)$$

The Fairness Loss L_{fair} is used to encourage equitable resource distribution across users. The loss incorporates a differentiable approximation of Jain's fairness index. Let r_i be the resource allocated to node i, then the fairness loss is formulated as

$$L_{fair} = 1 - \frac{\left(\sum_{i=1}^{N} r_i\right)^2}{N \cdot \sum_{i=1}^{N} r_i^2} \quad (29)$$

The above loss function penalizes skewed distributions and ensures that all devices receive a balanced share of bandwidth or scheduling opportunities. A lower value of this term indicates greater fairness. To reduce unnecessary power consumption, the Energy Efficiency Loss L_{energy} penalizes solutions that allocate excessive energy. The energy loss is mathematically expressed as

$$L_{energy} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{P_i}{\eta_i}\right) \quad (30)$$

where P_i indicates the predicted power for node i, η_i indicates the spectral efficiency for node i. By integrating these components, the total loss ensures that SA-GMNet does not just perform well in isolated metrics, but aligns with real-world system objectives like

accuracy, fairness, and efficiency. The coefficients λ_1 and λ_2 are selected through validation to best reflect deployment priorities. This multi-objective optimization approach allows the proposed SA-GMNet to perform reliably in dynamic, heterogeneous wireless environments. The summarized pseudocode for the proposed SA-GMNet is presented in Table 2 (Appendix A).

The proposed SA-GMNet model is developed to handle high user mobility in a better manner compared to traditional models, which handle static conditions. In dynamic network environments, users frequently move from one location to another, affecting signal quality, interference levels, and resource demand. The proposed SA-GMNet uses spatial-temporal graph views that are dynamically updated based on current mobility patterns. Using unique graphs, the model describes user movements, signal interference, traffic flows, and adjusts them based on users' position changes. As a result, the proposed model is aware of real-time network changes and the use of attention mechanisms helps focus on the most recent and relevant changes in user behavior. Overall, SA-GMNet performs significantly better in high mobility cases due to its continuous learning and adjusting features to motion, which effectively handling the dynamic changes in the environment.

4. Results and Discussion

The experimentation for the proposed SA-GMNet model is carried out using the python tool and benchmark vehicle wireless network scheduling dataset [22] from the Kaggle repository is used to evaluate the model's performance. The dataset is initially preprocessed to normalize the continuous features like speed, signal strength and traffic load. Further graph generation is done based on node proximity, reflecting the dynamic network topology over time. The dataset is divided into two parts for training and testing. Each sample is processed as a multi-modal input that includes spatial coordinates, traffic demands and graph connectivity. The multi-head attention encodes the spatial and temporal dependencies whereas the graph mixture model obtains the structural dynamics of the network which varies dynamically. The final output layer predicts the channel and power for each node. The simulation hyperparameters of proposed and existing models are presented in Table 3 (Appendix B).

The average throughput comparative performance analysis presented in Figure 2 for the proposed SA-GMNet model and existing DL methods across 100 epochs. The proposed model consistently achieves the highest throughput stabilizing at approximately 29.1 Mbps,

outperforming the Transformer-GAT, which peaks around 27.6 Mbps. ST-GCN follows with 24.8 Mbps, while GCN and LSTM exhibits less throughput as 24.5 Mbps and 23.2 Mbps, respectively. DNN records the lowest throughput among all as 21.7 Mbps which indicates its limitations in capturing temporal and structural dependencies. The superior performance of SA-GMNet is attributed to its multi-modal attention mechanism and graph mixture fusion, which effectively utilize spatio-temporal patterns and dynamic topologies for more precise resource allocation.

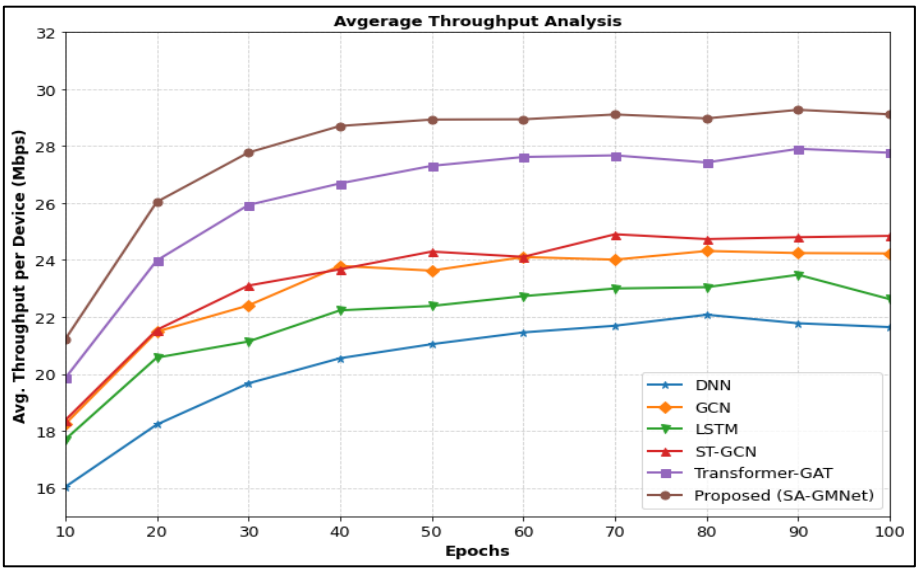

Figure 2. Average Throughput Analysis

The scheduling fairness comparative analysis presented in Figure 3, highlights the performance of various models using Jain's Index. The higher value of scheduling fairness indicates more equitable resource allocation across users. The proposed SA-GMNet model exhibits the highest fairness, reaching a stable value of 0.961 due to its ability to contextually fuse spatial and temporal information with dynamic graph attention. Thus, it allows the model to adapt allocations fairly under varying mobility and load. Whereas existing transformer-GAT exhibits less performance as 0.931 due to its lacking in temporal coordination. The existing ST-GCN and GCN achieve 0.899 and 0.872 respectively which is lesser than the proposed due to limited cross-modal integration. LSTM which lacks spatial context converges at 0.849 and the DNN exhibits the lowest at 0.823 due to its inability to model temporal trends.

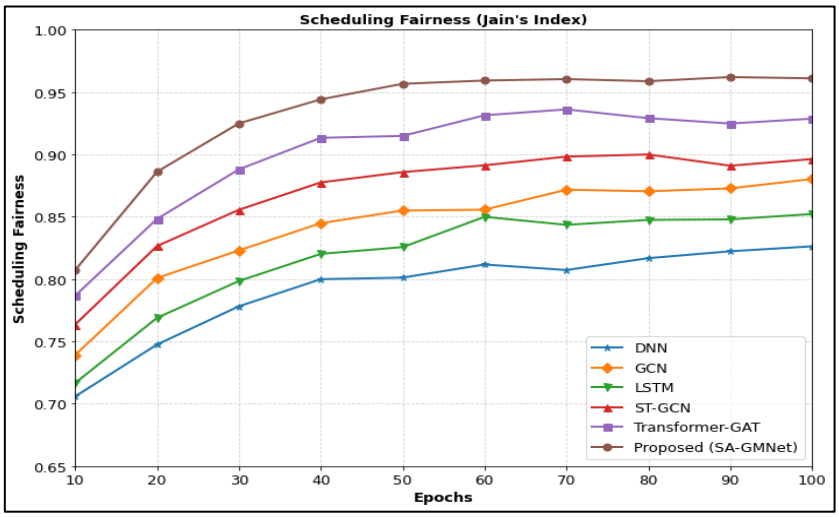

Figure 3. Scheduling Fairness Analysis

Figure 4 presents the energy efficiency performance of all models in terms of output per unit power consumption, measured in Mbps/Watt. The proposed SA-GMNet achieves the highest energy efficiency stabilizing around 3.26 Mbps/Watt due to its optimized resource control and context-aware decision-making that minimizes power usage without compromising throughput. Whereas existing transformer-GAT exhibits 2.87 Mbps/Watt which is lesser than the proposed due to inefficiency in mobility fusion. The existing ST-GCN and GCN attain 2.53 and 2.41 Mbps/Watt respectively. Their limited temporal coordination reduces adaptability under variable loads. LSTM reaches 2.19 Mbps/Watt only due to the absence of structural learning which affects allocation under network congestion. DNN the least performer which exhibits 2.04 Mbps/Watt due to its flat architecture and inability to utilize graph topology and temporal trends which results the model to exhibit poor performance in energy allocation.

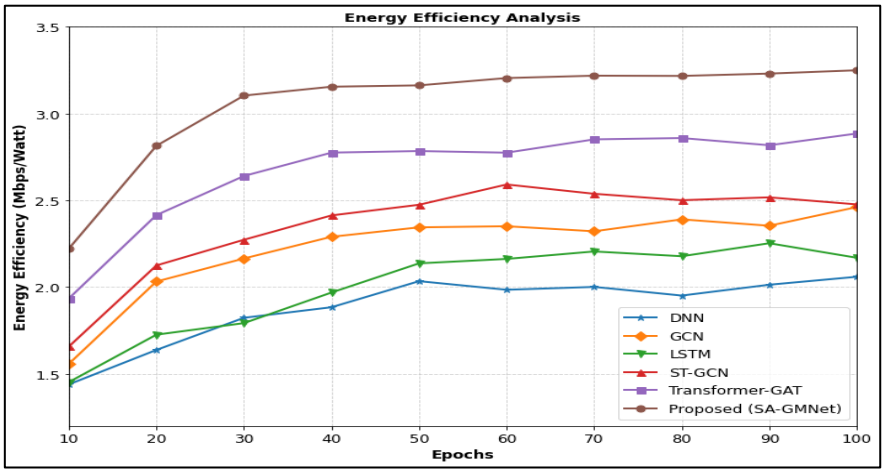

Figure 4. Energy Efficiency Analysis

The energy efficiency exhibits the model efficiency in terms of data transmitted per unit of power consumed. The proposed lightweight architecture allows to deploy the model on different types of edge devices without requiring any additional computation resources. The proposed model is executed in a general-purpose processor and exhibits best performance with high data rates and reduced parameter count. The spectrum utilization efficiency analysis presented in Figure 5 compares the effectiveness of different models in maximizing spectrum use across 100 epochs. The proposed SA-GMNet model outperforms all existing methods achieving a peak efficiency of 93.6%, attributed to its dynamic fusion of spatial-temporal features and graph mixture attention. The existing transformer-GAT exhibits 89.4% but lacking in performance due to its integrated mobility adaptation. ST-GCN and GCN reach 86.2% and 83.7% respectively which is lesser than the proposed. LSTM achieves 81.2% which is lesser than the proposed as it cannot handle spatial network shifts effectively. The existing DNN shows the lowest performance at 79.3% which is lesser than the proposed due to inefficient resource assignments.

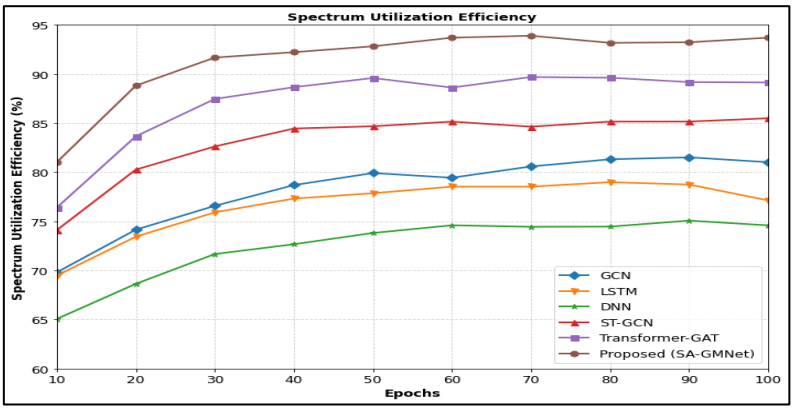

Figure 5. Spectrum Utilization Efficiency Analysis

The allocation latency analysis presented in Figure 6 describes the time each model takes to complete a resource assignment operation. The lower latency values indicate faster decision-making. The proposed SA-GMNet consistently achieves the lowest latency, stabilizing around 5.9 ms, owing to its efficient attention-driven feature integration and lightweight fusion of spatial-temporal information, which streamlines inference. Transformer-GAT follows with 6.4 ms, benefiting from parallelizable attention but lacking the mobility adaptation seen in SA-GMNet. ST-GCN and LSTM record 7.8 ms and 7.3 ms, respectively, with their temporal modeling offering moderate speed but constrained by sequential processing overhead. GCN reaches 8.6 ms, impacted by static graph assumptions and slower contextual

updates. DNN performs worst at 9.7 ms, due to its lack of structural optimization and absence of time-aware modeling, leading to inefficient computation paths.

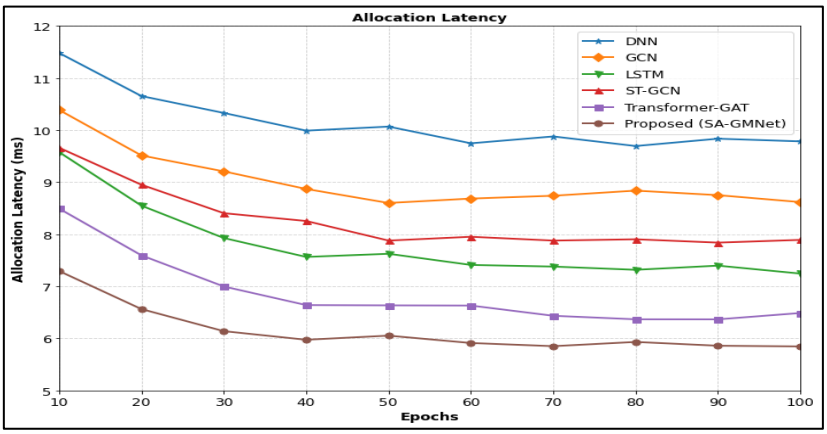

Figure 6. Allocation Latency Analysis

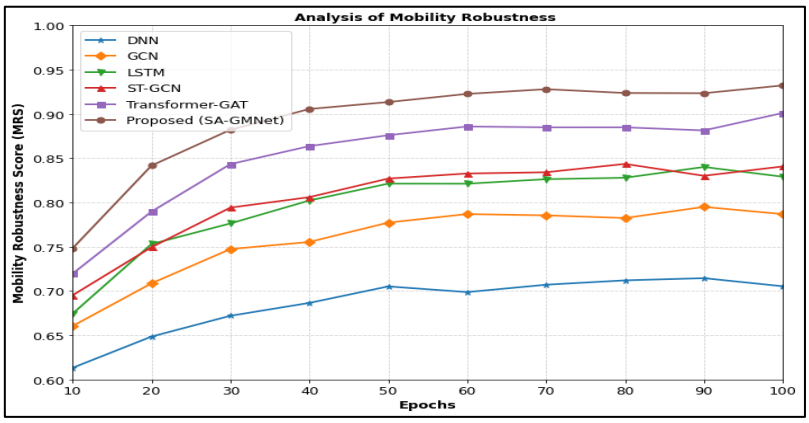

Figure 7. Mobility Analysis

The Mobility Robustness Score (MRS) presented in Figure 7 compares how effectively each model sustains performance under dynamic user mobility. The proposed SA-GMNet outperforms all others, reaching a peak MRS of 0.928, due to its design that integrates mobility context with spatial-temporal graph attention, allowing it to adapt resource allocation in fluctuating topologies. Transformer-GAT follows with 0.891, performing well by utilizing attention but lacking integrated temporal mobility awareness. ST-GCN and LSTM exhibit competitive performance, with final scores of 0.837 and 0.834, respectively, as both can capture time-based variations, though ST-GCN benefits from added graph representation. GCN achieves an MRS of 0.791, limited by its static graph assumption and inability to process temporal shifts. DNN yields the lowest score at 0.716, primarily because it lacks both sequential and topological modeling, making it ineffective in mobile network scenarios.

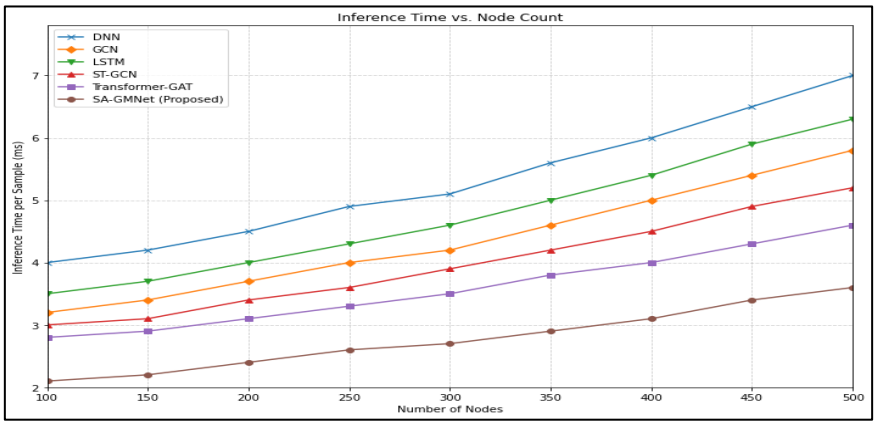

Figure 8. Inference time Analysis vs Nodes

Further the inference time is analyzed over different node counts. Figure 8 presents the inference time analysis over 100 to 500 nodes. The graph describes the inference time scalability of the proposed SA-GMNet model and its objective is to evaluate whether the model remains efficient when deployed in dense environments like vehicular networks. In this each vehicle is considered as nodes and increasing the node count indicates the dense network environment. The inference time indicates how long the model takes time to make resource allocation decisions. Generally, the inference should be low and the proposed model also exhibits the lower inference time which means it takes quick decisions in resource allocation process. The proposed SA-GMNet exhibits lowest inference which starts from 2.1ms for 100 nodes and increased gradually at 3.6ms for 500 nodes. The proposed model exhibits better performance due to its efficient attention-based fusion and feature processing abilities. The conventional models DNN exhibits less performance as 7.0ms and LSTM attains 6.3ms which is lesser than the proposed. The transformer model exhibits better latency by reaching 4.6ms, however it is lesser than the proposed model.

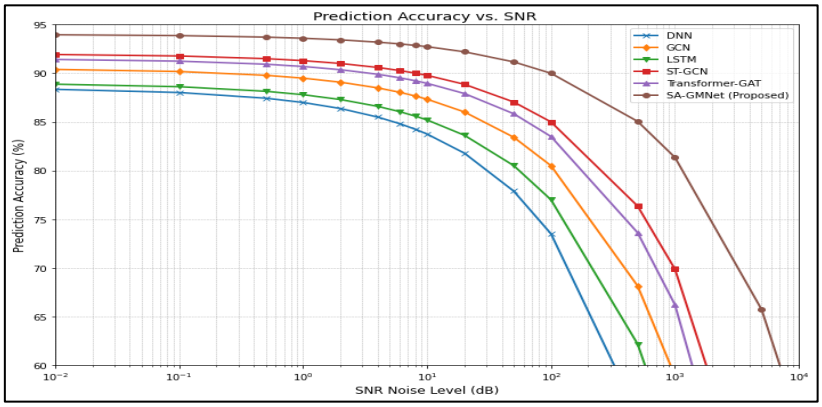

Figure 9. Prediction Accuracy Analysis

The comparative analysis graph presented in Figure 9 depicts the prediction accuracy performance of proposed and existing models over different SNR range. The proposed model exhibits 93.8% even at low distortion levels and attains 85.6% at extreme noisy condition. Whereas the existing DNN and LSTM models exhibits low performance from maximum to 64.3% and 70.4% at high SNR condition. This limited performance is due to the existing model limited spatial temporal feature processing abilities. The results clearly exhibit the better performance of the proposed model under different fluctuating conditions.

The overall performance comparison presented in Table 4 (Appendix C) clearly demonstrates the superior capability of the proposed SA-GMNet model across all evaluated metrics. In terms of spectrum utilization efficiency, SA-GMNet achieves 93.6%, outperforming Transformer-GAT 89.4% and ST-GCN 86.2%, due to its adaptive context-aware spectrum allocation. It also leads in average throughput, reaching 29.1 Mbps, significantly higher than GCN 24.5 Mbps and LSTM 23.2 Mbps, reflecting its effective handling of dynamic data demands. In energy efficiency, it reaches 3.26 Mbps/Watt, while the next best, Transformer-GAT, reaches 2.87 Mbps/Watt. SA-GMNet also scores the highest in scheduling fairness 0.961 and mobility robustness 0.928, both critical in mobile wireless systems. In contrast, DNN shows the weakest performance overall, with the lowest throughput 21.7 Mbps, highest latency 9.7 ms, and poorest robustness 0.716, primarily due to its lack of temporal and spatial modeling. These results confirm that SA-GMNet offers a comprehensive and efficient solution for intelligent wireless resource management.

The effectiveness of the proposed model though validated through simulation, incorporating it in real world environment will collect different types of data like user movement, traffic loads and signal interferences. All the collected data can be processed through graph views and then combined through the attention-based fusion to understand which information is useful to improve the network performance. Based on the information, decisions about the users can be made to provide access to resources. Specifically, the temporal attention adapts the changes in network traffic and user mobility and the spatial component analyzes the user interferences. Thus, the proposed model might avoid interferences and allocates resources to the users considering the dynamic demands. However, in real time environment few resource collisions might occurs due to real time network conditions which is considered as minor limitation of the proposed work.

5. Conclusion

This research presents a novel Spatio-Attentive Graph Mixture Network (SA-GMNet) for intelligent wireless resource allocation in dynamic environments. The proposed model integrates graph-based topological awareness with temporal mobility patterns using a multi-modal attention mechanism, enabling context-driven scheduling decisions. The experimentation was conducted using a benchmark vehicular wireless scheduling dataset from Kaggle, featuring time-variant topology, mobility, and traffic load characteristics. SA-GMNet was compared against existing models such as DNN, GCN, LSTM, ST-GCN, and Transformer-GAT. Across eight key metrics, the proposed SA-GMNet achieved the best results, including 93.6% spectrum efficiency, 29.1 Mbps average throughput, 0.087 interference index, 3.26 Mbps/Watt energy efficiency, and 0.961 scheduling fairness. Additionally, it recorded the lowest latency 5.9 ms and inference time 3.2 ms, demonstrating real-time capability. While SA-GMNet outperforms in multi-dimensional optimization, its limitation lies in computational overhead from graph attention fusion, which may scale unfavorably in ultra-dense networks. Future work can explore model pruning, hardware-aware deployment, and transfer learning to improve scalability. Furthermore, the framework could be extended to support federated or edge-based architectures, making it adaptable for 6G applications involving distributed intelligence and real-time responsiveness.

References

[1] Naren, Abhishek Kumar Gaurav, Nishad Sahu, Abhinash Prasad Dash, G S S Chalapathi, and Vinay Chamola, “A Survey on Computation Resource Allocation in IoT enabled Vehicular Edge Computing,” Complex & Intelligent Systems, vol.8, no.7, 2021, 1-18.

[2] Huanhuan Li, Hongchang Wei, Zheliang Chen, Yue Xu, “Adaptive Resource Allocation Algorithm for 5G Vehicular Cloud Communication,” Computers, Materials & Continua, vol. 80, no. 2, 2199-2219.

[3] Augustian Isaac.R, Sundaravadivel.P, Nici Marx.V.S, Priyanga.G, “Enhanced novelty approaches for resource allocation model for multi-cloud environment in vehicular Ad-Hoc networks,” scientific reports, 15, 2025, 1-18.

[4] Seyed Salar Sefati, Asim Ul Haq, Nidhi, Razvan Craciunescu, Simona Halunga, Albena Mihovska, Octavian Fratu, “A Comprehensive Survey on Resource Management in 6G Network Based on Internet of Things,” IEEE Access, vol. 12, 2024, 113741-13784.

[5] Maraj Uddin Ahmed Siddiqui, Faizan Qamar, Faisal Ahmed, Quang Ngoc Nguyen, Rosilah Hassan, “Interference Management in 5G and Beyond Network: Requirements, Challenges and Future Directions,” IEEE Access, vol. 9, 2021, 68932- 68965.

[6] Fereshteh Atri Niasar, Amir Reza Momen, Seyed Abolfazl Hosseini, “A novel approach to fairness-aware energy efficiency in multi-RAT heterogeneous networks,” AEU - International Journal of Electronics and Communications, vol.152, 2022, 1-28.

[7] J. Simon, M.A. Elaveini, N. Kapileswar and P.P. Kumar, "ARO-RTP: Performance analysis of an energy efficient opportunistic routing for underwater IoT networks", Peer-to-Peer Networking and Applications, 2023, 1-17.

[8] Franciskus Antonius, “Efficient resource allocation through CNN-game theory based network slicing recognition for next-generation networks,” Journal of Engineering Research, vol. 12, no. 4, 2024, 793-805.

[9] Aristidis G. Vrahatis, Konstantinos Lazaros, Sotiris Kotsiantis, “Graph Attention Networks: A Comprehensive Review of Methods and Applications,” Future Internet, vol.16, no. 9, 2024, 1-34.

[10] Lucas R. Frank, Antonino Galletta, Lorenzo Carnevale, Alex B. Vieira, Edelberto Franco Silva, “Intelligent resource allocation in wireless networks: Predictive models for efficient access point management,” Computer Networks, vol. 254, 2024, 1-12.

[11] Kimchheang Chhea, Dara Ron and Jung-Ryun Lee, “Weighted De-Synchronization Based Resource Allocation in Wireless Networks,” Computers, Materials & Continua, vol.75, no.1, 2023, 1815-1826.

[12] Huanhuan Li, Hongchang Wei, Zheliang Chen and Yue Xu, “Adaptive Resource Allocation Algorithm for 5G Vehicular Cloud Communication,” Computers, Materials & Continua, vol.80, no.2, 2024, 2199-2219.

[13] Franciskus Antonius, "Efficient resource allocation through CNN-game theory based network slicing recognition for next-generation networks," Journal of Engineering Research, vol.12, 2024, 793-805.

[14] J. Simon, N. Kapileswar, B. Padmavathi, K.D. Devi, and P.P. Kumar, "Optimization of node deployment in underwater internet of things using novel adaptive long short-term memory-based egret swarm optimization algorithm" International Journal of Communication Systems, 37(17), p.e5926, August 2024.

[15] Fahd N. Al-Wesabi1, Imran Khan, Saleem Latteef Mohammed, Huda Farooq Jameel, Mohammad Alamgeer, AliM. Al-Sharafi and Byung Seo Kim, "Optimal Resource Allocation Method for Device-to-Device Communication in 5G Networks," Computers, Materials & Continua, vol.71, no.1, 2022, 1-15.

[16] Fahad Ahmed Al-Zahrani, Imran Khan, Mahdi Zareei, AsimZeb and AbdulWaheed, "Resource Allocation and Optimization in Device-to-Device Communication 5G Networks," Computers, Materials & Continua, vol.69, no.1, 2021, 1201-1214.

[17] Yuan Zhi, Jie Tian, Xiaofang Deng, Jingping Qiao, Dianjie Lu, "Deep reinforcement learning-based resource allocation for D2D communications in heterogeneous cellular networks," Digital Communications and Networks, vol.8, 2022, 834–842.

[18] Shahad Alyousif, Mohammed Dauwed, Rafal Nader, Mohammed Hasan Ali, Mustafa Musa Jabar and Ahmed Alkhayyat, "An Optimal Algorithm for Resource Allocation in D2D Communication," Computers, Materials & Continua, vol.75, no.1, 2023, 531-546.

[19] Bin Jiang; Lixin Cai; Guanghui Yue; Fei Luo; Shibao Li; Jian Wang, "Energy-Efficient Wireless Resource Allocation for Heterogeneous Federated Multitask Networks Based on Evolutionary Learning," IEEE Transactions on Industrial Informatics, vol. 21, no. 5, 2025, 4094-4104.

[20] Xiaochuan Sun; Jinpeng Han; Yingqi Li; Kaiyu Zhu; Haijun Zhang, "Make Power Allocation More Adaptive in Ultra Dense Networks: Priority-Driven Deep Reinforcement Learning via Noise-Perturbations," IEEE Transactions on Wireless Communications, vol. 24, no. 5, 2025, 4010-4023.

[21] Chi Xu; Peifeng Zhang; Haibin Yu, “Lyapunov-Guided Resource Allocation and Task Scheduling for Edge Computing Cognitive Radio Networks via Deep Reinforcement Learning,” IEEE Sensors Journal, vol. 25, no. 7, 2025, 12253-12264.

[22] https://www.kaggle.com/datasets/ziya07/vehicle-wireless-network-scheduling

Appendix A

Table 2. Pseudocode for the Proposed SA-GMNet – Spatio-Attentive Graph Mixture Network

Pseudocode for the Proposed SA-GMNet – Spatio-Attentive Graph Mixture Network

Input: $G_t^j = (V_t, E_t^j)$ Set of graph views at time t, $j = 1, 2, \ldots, k$, $X_t \in R^{N \times d}$: Node feature matrix for all N nodes, $\{X_i^{t-\tau}\}_{\tau=0}^{T}$ Historical feature sequence for each node v_i
Output: $\hat{Y}$ Predicted resource allocation decisions channel, power, scheduling for all nodes
Initialization: $W^{(j)}$, $a^{(j)}$ for all graph attention heads, Temporal projections W_Q, W_K, W_V for attention, Fusion parameters W_f, b_f Output parameters W_o, b_o, attention heads M, window size T, number of graph views k
Begin
For each node $v_i \in V_t$
For each graph view $j = 1$ to k
Compute transformed features $x'_k = W^{(j)} x_k$
For each neighbor $v_k \in N_i^j$
Calculate attention score $e_{ik}^{(j)} = LeakyReLU\left(a^{(j)\top}[x'_i | x'_k]\right)$
Normalize using SoftMax $\alpha_{ik}^{(j)} = \frac{exp\left(e_{ik}^{(j)}\right)}{\sum_{l \in N_i^j} exp\left(e_{il}^{(j)}\right)}$
Aggregate spatial features $h_i^{(j)} = \sum_{k \in N_i^j} \alpha_{ik}^{(j)} x'_k$
Compute gating weights $g_j = \frac{exp(z_j)}{\sum_{l=1}^{k} exp(z_l)}$
Compute mixed spatial embedding $h_i^{spatial} = \sum_{j=1}^{k} g_j \cdot h_i^{(j)}$
For temporal attention encoding for each node v_i
Stack sequence: $X_i^t = [x_i^{t-T+1}, \ldots, x_i^t] \in R^{T \times d}$
Project to Q, K, V $Q = X_i^t W_Q, K = X_i^t W_K, V = X_i^t W_V$
Compute scaled attention $A = softmax\left(\frac{QK^\top}{\sqrt{d_a}}\right)$
Generate temporal embedding $H = A\,V$
Apply multi-head attention and concatenate outputs
Project and normalize $h_i^{temporal} = LayerNorm(X_i^t + Concat(H_1, \ldots, H_M)W_O)$
Concatenate features $z_i = \left[h_i^{spatial} | h_i^{temporal}\right]$
Apply fusion transformation $\tilde{h}_i = ReLU\left(W_f z_i + b_f\right)$
Apply dropout and normalization if enabled $h_i^{fused} = LayerNorm\left(Dropout\left(\tilde{h}_i\right)\right)$
Compute final output
If single-task
$\hat{y}_i = Softmax\left(W_o h_i^{fused} + b_o\right)$
If multi-task
For each task t, $\hat{y}_i^{(t)} = Activation\left(W_o^{(t)} h_i^{fused} + b_o^{(t)}\right)$
Compute total loss $L_{total} = L_{alloc} + \lambda_1 L_{fair} + \lambda_2 L_{energy}$
Update all trainable parameters
Return
End
End
End
End

Appendix B

Table 3. Simulation Hyperparameters for Proposed and Existing Models

S.No	Method/Algorithm	Parameter	Type / Range
1	Proposed SA-GMNet	Learning Rate	0.0005
2		Number of Attention Heads	8
3		Graph Mixture Size	3
4		Embedding Dimension	128
5		Batch Size	64
6		Optimizer	Adam
7		Dropout Rate	0.2
8		Number of Epochs	100
9	DNN	Learning Rate	0.001
10		Hidden Layers	3
11		Units per Layer	256
12		Activation Function	ReLU
13		Batch Size	64
14		Optimizer	Adam
15		Number of Epochs	100
16	GCN	Learning Rate	0.0008
17		Number of GCN Layers	2
18		Hidden Dimension	128
19		Dropout Rate	0.3
20		Batch Size	64
21		Optimizer	Adam
22		Number of Epochs	100
23	LSTM	Learning Rate	0.001
24		Hidden Size	128
25		Number of Layers	2
26		Sequence Length	10
27		Batch Size	64
28		Optimizer	RMSProp
29		Number of Epochs	100
30	ST-GCN	Learning Rate	0.0007
31		Spatial Kernel Size	3
32		Temporal Window	5
33		Hidden Channels	64
34		Dropout Rate	0.25
35		Batch Size	64
36		Number of Epochs	100
37	Transformer-GAT	Learning Rate	0.0006
38		Number of Attention Heads	4
39		Layers	6
40		Hidden Dimension	128
41		Dropout Rate	0.3
42		Batch Size	64
43		Number of Epochs	100

Appendix C

Table 4. Overall Performance Comparative Analysis

Metric	DNN	GCN	LSTM	ST-GCN	Transformer-GAT	Proposed SA-GMNet
Spectrum Utilization Efficiency %	79.3	83.7	81.2	86.2	89.4	93.6
Avg. Throughput per Device (Mbps)	21.7	24.5	23.2	24.8	27.6	29.1
Energy Efficiency (Mbps/Watt)	2.04	2.41	2.19	2.53	2.87	3.26
Scheduling Fairness (Jain's Index)	0.823	0.872	0.849	0.899	0.931	0.961
Allocation Latency (ms)	9.7	8.6	7.3	7.8	6.4	5.9
Mobility Robustness Score (MRS)	0.716	0.791	0.834	0.837	0.891	0.928