International Journal of Artificial Intelligence and Machine Learning
Publisher's Home Page: https://www.svedbergopen.com/
Research Paper Open Access

G Network Slicing: Dynamic Resource Management For Ultra-Reliable Communication

Sanjay Nana Gunjal[1], Dr. Aarti Suryakant Pawar[2], Dr. Dhanashree K. Barbole[3], Leena Deshpande[4], Dr. Shubhangi Joshi[5], Balkrishna K. Patil[6]

[1]Department of Computer Engineering, Sanjivani College of Engineering (An Autonomous Institute), Savitribai Phule Pune University, Kopargaon - 423 603, Maharashtra, India. gunjalsanjay@sanjivani.org.in gunjalsanjay@sanjivani.org.in
[2]Department of Electronics and Telecommunication Pimpri Chinchwad College of Engineering,Pune, Maharashtra, India Email id: pawaraarti1610@gmail.com orcid id:https://orcid.org/0000-0001-6708-1042
[3]Symbiosis Institute of Technology, Pune Campus Symbiosis International (Deemed University), Pune, India Dhanashree.barbole@sitpune.edu.in
[4]Associate Professor Department of Computer Engineering - Software Engineering Vishwakarma Institute of Technology, Pune, Maharashtra, 411037 leena.deshpande@vit.edu
[5]Associate Professor, Jspm's Rscoe Tathawade Pune sght276@gmail.com
[6]Department of Computer Science and Engineering, Symbiosis Institute of Technology, Pune Campus, Symbiosis International (Deemed University), Pune, India author: balkrishna.patil@sitpune.edu.in

Abstract— Next-generation G networks will enable mission-critical services that will be supported by abstract-reliable communication and at the same time be able to support heterogeneous traffic that will have varied quality-of-service needs. Network slicing has been one of the enablers that enable logical separation of shared physical infrastructure into service specific virtual networks that are isolated. The paper introduces a dynamic resource management system of G network slicing in an attempt to provide ultra-reliable, low-latency, and adaptive communication in time-varying traffic conditions. The framework proposed combines the slice level resource abstraction with smart utilization of bandwidth, transmission power and edge cloud computing resources. The recurrent learning models, such as LSTM-TDP and GRU-NetSlice, are used to predict traffic and demand dynamics, which allow a proactive slice provisioning. To resolve real-time uncertainty, deep reinforcement learning agents, i.e., DQN-Slice and PPO-SliceManager are used to optimize the policies of slice that maximize the reliability, spectral efficiency and resource utilization simultaneously. Reliability-conscious optimization models are added to provide probabilistic latency and packet delivery constraints that are needed in ultra-reliable low-latency communication services. In addition, redundancy-based and fast slice reconfiguration-based fault tolerance mechanisms are presented to improve service availability in the event of failures. The evaluation by simulation shows that the proposed intelligent slicing framework is much more reliable, less end-to-end latency, and more adaptable to network dynamism than the existing, and the heuristic allocation schemes.

Keywords— G network slicing; ultra-reliable communication; dynamic resource management; deep reinforcement learning; traffic prediction.

I. INTRODUCTION

The fast development of next-generation G wireless networks is predetermined by the increasing demand on the heterogeneous services that require strict and incompatible quality-of-service (QoS) criteria. New applications like autonomous transportation, remotely operated surgery, industrial automation, smart grids and immersive extended reality require very reliable communication with very low latency, high availability and resiliency to failures. Simultaneously, the improved mobile broadband and massive machine-like communication services still demand high data rates, colossal connectivity, and effective use of the spectrum. The need to support these different classes of services on a common physical infrastructure is very challenging to the traditional and fixed network design and resource allocation techniques. Network slicing has now become a structural architectural paradigm to manage this challenge, offering the opportunity to build a number of logical networks, or slices, upon a shared physical infrastructure. Every slice may be configured to satisfy the particular performance, reliability, and latency needs of a certain service category and remain logically isolated with other slices. Nonetheless, at times of high dynamic wireless environment, the static slice configuration fails to work, as traffic demand, channel conditions and service priorities vary with time [1]. Also as a result, it becomes necessary to have dynamic resource management which will ensure that the ultra-reliable communication requirements can always be met without unduly over-provisioning the already limited network resources. The dynamic resource management of G network slicing is based on the adaptive distribution of radio, computing and networking resources in one slice to another based on real-time conditions. This

covers the bandwidth and power provisioning in radio access network, computational resource provisioning at edge and core, and smart scheduling, which collectively optimize the latency, reliability and efficiency [2]. The most important issue is how to make sure there is a balance between the reliability guarantees of mission critical slices and the overall system goal of maximizing resource utilization and scalability. Traditional rule-based and optimization-oriented schemes tend to be unable to handle the next-generation wireless network due to its high dimensionality, uncertainty, and nonlinearity [3]. In order to address these drawbacks, machine learning based and deep reinforcement learning based intelligent resource management methods have become more and more popular. Learning-based models have the ability to learn intricate temporal traffic distribution, forecast upcoming demand, and optimize the learning process of the best slicing of policies by interacting with the network environment. Through traffic prediction, adaptive decision-making and reliability-aware optimization, intelligent network slicing models are able to preemptively assign resources and react promptly to network dynamics to enhance service reliability as well as operational efficiency [4].

II. FUNDAMENTALS OF G NETWORK SLICING

A. Concept and architecture of network slicing

The principle of network slicing in G networks is a major architectural polity that allows creating two or more logical networks on a shared physical infrastructure. The network slices are tailored to a particular service category and have a specific performance requirement (in server latency, reliability, throughput, and security). Network slicing enables operators to partition and customize resources flexibly, unlike using separate physical networks to support different applications, enhancing scalability and economic performance by a significant margin [5]. Network slicing architecture is commonly implemented based on software-defined networking (SDN) and network function virtualization (NFV) technologies that are based on the principle of virtualization and softwarization. Radio access, transport, computing, and storage resources make up the physical infrastructure layer [6]. To add to this, a virtualization layer is a layer which abstracts the physical resources into a pool which can be dynamically assigned to slices. The slice layer is comprised of several end-to-end slices which span across the radio access network, core network, and the edge/cloud computing layers. The slice generation, configuration, monitoring, and adjustment are governed by a single centralized management and coordination layer [7]. This layer liaises with analytics and control modules to make sure slice specific service level agreement is met. On the whole, the network slicing architecture allows flexible, programmable, and service-aware operation of the network, which is necessary to support the wide range of applications and highly demanding applications in the future G networks [8].

B. Service Types: eMBB, URLLC, and mMTC

The necessity to serve heterogeneous types of services with fundamentally different communication demands is the major driving force of G network slicing. One of the areas of Enhanced Mobile Broadband (eMBB) is the emphasis on high data rates, coverage, and spectral efficiency which is mainly used in high-definition video streaming, virtual reality and immersive multimedia services [9]. The Ultra-Reliable Low-Latency Communication (URLLC) is designed to address mission-critical applications with extremely low latency, high reliability and deterministic performance [10]. Examples of common applications are in the industrial automation, autonomous vehicles, remote surgery, and smart grid protection. To maintain the delivery of packets even in unfavourable network conditions, URLLC slices are planned with rigid latency constraints, redundancy schemes and reliability conscious scheduling. In large numbers of deployments in Internet of Things, mMTC services a vast array of low-power low-data-rate devices like sensors and actuators with slices that are focused on scalability, low energy consumption, and minimal signaling overhead [11]. Isolating and separating such types of services into special slices, G networks can effectively satisfy the needs of a wide range of requirements without affecting the overall system.

C. Slice Isolation, Orchestration, and Lifecycle Management

There are critical elements of effective network slicing in G networks, i.e., slice isolation, orchestration and lifecycle management. Slice isolation makes sure that the performance, security and reliability of a single slice are not negatively impacted by the behavior of other slices sharing the same physical infrastructure. Isolation may occur at several levels among them radio resources, computing capacity, control signaling and data forwarding paths. High isolation is especially necessary when the communication service is ultra-reliable and the performance can not be affected because of interference with other slices [12]. Orchestration is the coordinated control and management of network slices in various domains which includes radio access, transport, core network and edge computing. A smart orchestrator is a dynamically-allocating resource, dynamically-deploying virtual network functions, and dynamically-configuring the slice, depending on real-time network conditions and service demands [13]. Service-level agreements and operational policies are used to make orchestration decisions. Table 1 discusses a comparison of current slicing methods, reliability assistance, and restrictions. Lifecycle management encompasses all the life cycle of a slice, which is slice creation, activation, scaling, modification and termination.

TABLE 1. SUMMARY OF RELATED WORK ON NETWORK SLICING AND ULTRA-RELIABLE COMMUNICATION

Network Type	Slicing Objective	Resource Types	Learning Technique	Key Limitation

5G	eMBB-centric slicing	Bandwidth	Convex optimization	Ignores URLLC constraints
5G	URLLC support	Bandwidth, Power	Heuristic scheduling	Static allocation
5G [14]	Multi-slice QoS	Bandwidth	Lyapunov optimization	High control overhead
5G	Dynamic slicing	Bandwidth, Compute	Reinforcement learning	Slow convergence
5G	End-to-end slicing	Bandwidth, Power	SDN/NFV orchestration	Lacks reliability modeling
Beyond 5G	URLLC enhancement	Radio resources	Stochastic optimization	Single-domain focus
Beyond 5G	Adaptive slicing	Bandwidth, Compute	Deep Q-learning	Discrete action space
6G	AI-driven slicing	Multi-resource	Federated learning	Communication overhead
6G	Reliability-aware slicing	Bandwidth, Power	Chance-constrained optimization	High computational cost
6G	URLLC optimization	Radio, Edge compute	PPO-based DRL	Training complexity
6G	Cross-slice coordination	Multi-resource	Hybrid ML + optimization	Limited fault tolerance

III. Dynamic Resource Management Framework

A. Problem formulation and system assumptions

G network slicing is formulated as a dynamic resource management optimization problem that will meet heterogeneous slice-specific quality-of-service requirements without wasting the shared physical resources. The system takes into consideration a multi-slice G network comprising of radio access, transport and edge-core computing infrastructures. Every slice is defined by specific service-level contract, such as limits of latency, reliability goals, minimum data rates and priority in the resources. The demand of each slice is supposedly time-varying and stochastic which is affected by user mobility, application behaviour and channel conditions. The resource management problem is primarily aimed at the joint optimization of bandwidth allocation, transmission power and computational resources provisioning across slices. This optimization is under the constraints of a total available resources, interference and probabilistic reliability, especially in ultra-reliable communication slices. The issue is also limited by the latency deadlines, energy efficiency and fairness among non-critical services.

B. Slice-Level Resource Abstraction and Virtualization

Resource abstraction at slice level level is a basic mechanism that makes it possible to flexibly and scalably manage resources in G network slicing. Using abstraction, heterogeneous physical resources like radio spectrum, power of transmission, computing cycles and storage are modeled as logical resources units that can be independently allocated to each slice. This abstraction moves the service requirements off of the underlying hardware, which enables the network operators to operate resources in a coherent and service-conscious fashion. As depicted in figure 1, virtualized resources allow a flexible and isolated network slicing. Resource abstraction is majorly conducted through the use of virtualization technologies.

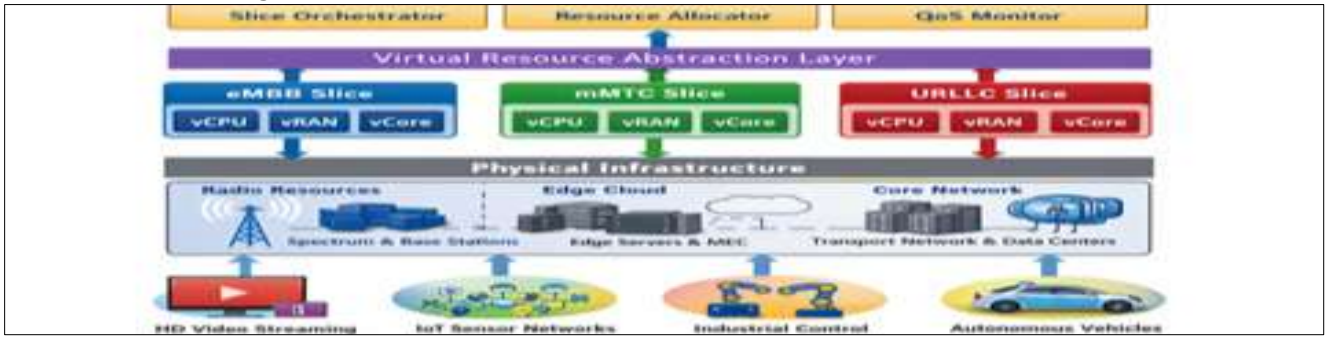

Fig.1. Slice-Level Resource Abstraction and Virtualization Architecture for G Network Slicing

Radio resources are virtualized by dividing spectrum, time slots, and spatial dimensions among the slices whereas computing resources are virtualized by deploying virtual machines or containers at edge nodes and core nodes. Network functions are realized as virtual network functions which can be moved easily, scaled, and migrated between infrastructure components. The abstracted resources at the slice level are then packaged into slice specific resource profiles which specify the capacity, latency and resources reliability characteristics. These profiles allow fast slice creation and elasticity on-demand basis.

C. Dynamic Allocation of Bandwidth, Power, and Compute Resources

Both time-varying bandwidth, power, and compute resources allocation is a key requirement of ensuring ultra-reliable communication in G network slicing. In radio access space, the allocation of bandwidth defines the possible data rate and latency of each slice whereas the transmission power directly affects the signal quality, coverage and reliability. The allocation of compute resources in edge and core nodes influences processing delay, time to complete a task and service responsiveness. The dynamic allocation system adjusts the assignment of resources constantly depending on real time measurements and projected demand. Elastic services like improved mobile broadband, on the other hand, can withstand adaptive resource scaling to enhance efficiency in general. Optimization of bandwidth, power and computing spheres allows them to make joint decisions which prevent local inefficiencies. As an illustration, moving computations to edge nodes can decrease latency, but it needs to have enough radio and processing capabilities. Dynamic allocation models use opinion-feedback provided by the monitoring and analytics modules in order to proactively increase or decrease resource shares.

IV. Intelligent Resource Allocation Techniques

A. Machine learning–based traffic and demand prediction

1) LSTM-TDP

LSTM-TDP is a long short-term memory-based traffic and demand prediction model that aims at capturing the temporally related correlations of the G network slicing environments. Because network traffic is sequential, LSTM-TDP is a good fit because it learns long-range dependencies between past network traffic measurements, slice utilization patterns and service requests arrivals. The model takes time-series as its input, including the rate of receiving packets, the density of users, and consumption of resources, which allows predicting the demand in the slices correctly in the future. The vanishing gradient problem found in the traditional recurrent models is avoided through LSTM-TDP by retaining the relevant information of the past by using gated memory cells. Proper demand forecasting enables the network orchestrator to allocate bandwidth, power and computing resources in advance before congestion sets in.

Step 1: Input Sequence Construction

Historical slice traffic data are organized into a time-series sequence as

$X = \{x(t-T+1), x(t-T+2), \dots, x(t)\}$

where x(t) represents the traffic demand at time t and T denotes the observation window length.

Step 2: LSTM State Update

The LSTM updates its internal memory using gated mechanisms:

Forget gate:

$$f(t) = \sigma(Wf \cdot [h(t-1), x(t)] + bf)$$

Input gate:

$$i(t) = \sigma(Wi \cdot [h(t-1), x(t)] + bi)$$

Cell state update:

$$c(t) = f(t) \odot c(t-1) + i(t) \odot tanh(Wc \cdot [h(t-1), x(t)] + bc)$$

2) GRU-NetSlice

GRU-NetSlice is a prediction model based on a gated recurrent unit and optimized to effectively predict traffic and demand in G network slicing. GRU-NetSlice has a simple gating mechanism in comparison to LSTM architecture and it is a simpler architecture that lowers the computational complexity and still has great temporal modeling power. The model trains both short-term and medium-term changes in traffic by examining historical slice demands, mobility and service utilization trends. GRU-NetSlice has a very lightweight design and therefore it is highly applicable in real time applications at edge controllers that may have limited processing capabilities. The model can scale slices in time and make decisions on resource allocation through timely and accurate demand predictions. GRU-NetSlice has found notable application in the case of traffic that is changing dramatically within a short period like in mobile broadband and mixed-service slices.

Step 1: Traffic Feature Input

Slice-level traffic demand features are represented as a temporal input sequence:

$$X = \{x(t-T+1), \dots, x(t)\}$$

Step 2: Gate Computation

The update and reset gates are calculated as:

Update gate:

$$z(t) = \sigma(Wz \cdot [h(t-1), x(t)] + bz)$$

Reset gate:

$$r(t) = \sigma(Wr \cdot [h(t-1), x(t)] + br)$$

Step 3: Hidden State Update

The candidate hidden state and final hidden state are obtained as:

$$\tilde{h}(t) = tanh(Wh \cdot [r(t) \odot h(t-1), x(t)] + bh)$$

$$h(t) = \big(1 - z(t)\big) \odot h(t-1) + z(t) \odot \tilde{h}(t)$$

B. Deep reinforcement learning for adaptive slicing decisions

1) DQN-Slice

DQN-Slice is a deep Q-learning based algorithm that allows running intelligent and adaptive slicing choices in G networks. The slicing problem is represented as a Markov decision process where the network controller monitors the system state such as the slice load, channel quality, latency violations and the availability of resources. According to this observation, DQN-Slice picks actions like bandwidth reallocation, power adjustments, or downsizing compute resources. A deep neural network estimates the action value function and the agent can use the high-dimensional state space. By communicating with the network environment, DQN-Slice acquires policies that trade off a reliable operation of critical slices with effective overall resource consumption. Target networks together with experience replay enhance stability and convergence of training. DQN-Slice is especially useful in moderate dynamic settings in which the discrete resource adjustment actions can make a significant positive contribution to the performance of latency and reliability.

Step 1: State, Action, and Reward Modeling

The adaptive slicing problem is formulated as a Markov Decision Process (MDP).

The system state at time t is defined as:

$$s(t) = \{L(t), Q(t), \gamma(t), R(t)\}$$

where L(t) denotes slice traffic load, Q(t) is queue length, γ(t) represents channel quality, and R(t) indicates available resources.

The action a(t) corresponds to discrete slicing decisions such as bandwidth, power, or compute reallocation.

The reward function is defined as:

$$r(t) = \alpha \cdot Reliability(t) - \beta \cdot Latency(t)$$

Step 2: Q-Value Update

The deep Q-network updates the action–value function as:

$$Q\big(s(t), a(t)\big) \leftarrow Q\big(s(t), a(t)\big) + \eta\left[r(t) + \gamma \cdot max\, Q(s(t+1), a') - Q\big(s(t), a(t)\big)\right]$$

2) PPO-SliceManager

PPO-SliceManager uses proximal policy optimization to ensure that the decisions to slice network stabilize and continuous control. PPO-SliceManager does not use value-based approaches but rather directly learns a parameterized policy that takes the observed states of the network to actions that allocate resources. The cut-off objective functional feature of a PPO makes sure there is a controlled update of policy that avoids sudden updates that may destabilize the ultra reliable communication services. This renders PPO-SliceManager highly appropriate in setup where there is constant action space, including fine bandwidth, power, and compute distributions. The agent can dynamically adjust to traffic variations, channel uncertainty and service priority changes.

Step 1: Policy and State Definition

The system state is represented as:

$$s(t) = \{D(t), U(t), R(t)\}$$

where D(t) is traffic demand, U(t) denotes resource utilization, and R(t) indicates slice reliability status.

A stochastic policy πθ(a(t)|s(t)) maps system states to continuous resource allocation actions.

Step 2: PPO Objective Function

The clipped PPO objective is defined as:

$$L_{PPO(\theta)} = E\left[min\big(r(t)(\theta) \cdot A(t), clip(r(t)(\theta), 1 - \varepsilon, 1 + \varepsilon) \cdot A(t)\big)\right]$$

where

$$r(t)(\theta) = \frac{\pi\theta\big(a(t)\big|s(t)\big)}{\pi\theta_{old}\big(a(t)\big|s(t)\big)}$$

V. ULTRA-RELIABLE COMMUNICATION SUPPORT MECHANISMS

A. Reliability modeling and probabilistic guarantees

In G network slicing, reliability modeling is an essential part of ultra-reliable communication support because it facilitates the quantitative evaluation and implementation of strict service guarantees. Reliability is often described via the probability of the successful delivery of the packets within a given latency constraint. Probabilistic models are used in order to capture the stochastic nature of the wireless channels, arrival of the traffic as well as processing delays.

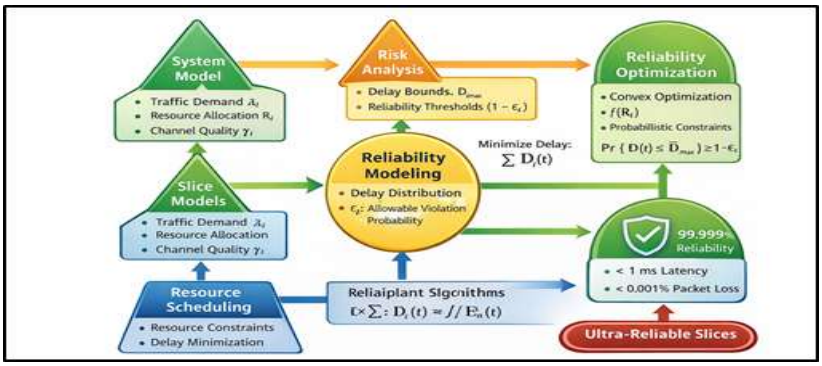

Fig.2. Reliability Modeling and Probabilistic Guarantee Framework for Ultra-Reliable G Network Slicing

Such models model the packet error rates, queueing delays and transmission failures in terms of statistical distributions or stochastic network calculus. Figure 2 depicts reliability modeling which guarantees the probabilistic ultra-reliable communication. The probabilistic guarantees are implemented by adding the reliability constraints to the resource allocation and scheduling decisions. As an example, ultra-reliable slices can be defined as needed to have a probability of packet delivery greater than a set limit, like 99.999%. This is done through provision of enough bandwidth, transmission power as well as computing resources considering the channel uncertainty and interference.

VI. Result and Discussion

The performance analysis shows that the suggested dynamic resource management model can considerably increase the ultra-reliable communication in the G network slicing context. Based on learning, traffic prediction models are able to provide proactive slice provisioning to minimize congestion and latency violations in the event of traffic surges. Slicing decisions based on deep reinforcement learning are more responsive to time-varying conditions in the network, which is more reliable and consumes less resources than both the static and heuristic schemes. Ultra-reliable slices are regularly capable of achieving a high latency and packet delivery threshold, whereas the elastic services are able to efficiently serve up the overall leftover resources without severe service degradation. Scheduling with reliability reduces additional the packet loss and jitter caused by uncertain channel conditions further.

Table 2. Performance Comparison of Resource Management Schemes

Resource Management Scheme	Avg. End-to-End Latency (ms)	Packet Delivery Reliability (%)	Spectral Efficiency (bps/Hz)	Resource Utilization (%)
Static Allocation	7.8	99.12	4.2	68.5
Heuristic-Based Slicing	5.6	99.45	4.9	74.2
DQN-Slice	3.9	99.87	5.6	82.4
PPO-SliceManager	3.2	99.94	5.9	86.7

Table 2 is the comparative performance analysis of various resource management schemes of G network slicing. The end-to-end latency of static allocation is the highest (7.8 ms) and the resource usage is the lowest, which means that it is not able to effectively manage the demands of dynamic traffic. Figure 3 compares the latency and reliability of schemes of slicing.

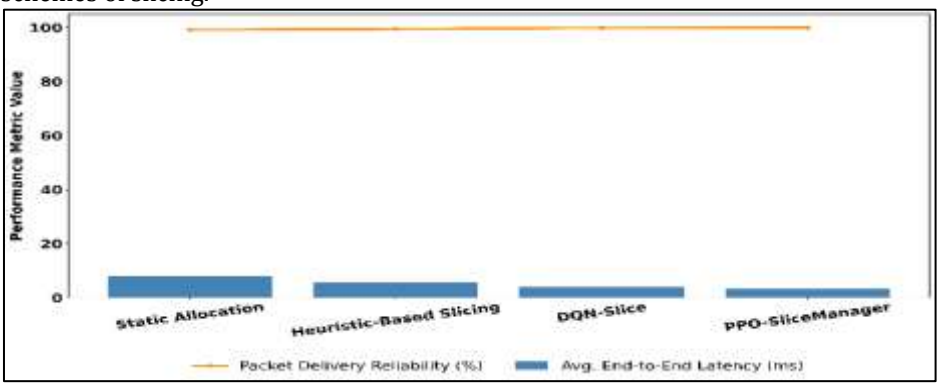

Fig.3. Latency and Reliability Comparison across Resource Management Schemes

The Slicing Heuristic Slicing is better at reducing latency and reliability by changing the rules, but it does not respond well to the changing network environment. Spectral efficiency improvement and resource utilization improvement are

illustrated in figure 4. Approaches based on learning show high performance improvements. DQN-Slice has a latency of 3.9 ms in addition to spectral efficiency and resource usage, and its optimal policy of network state learning across the allocation policy space.

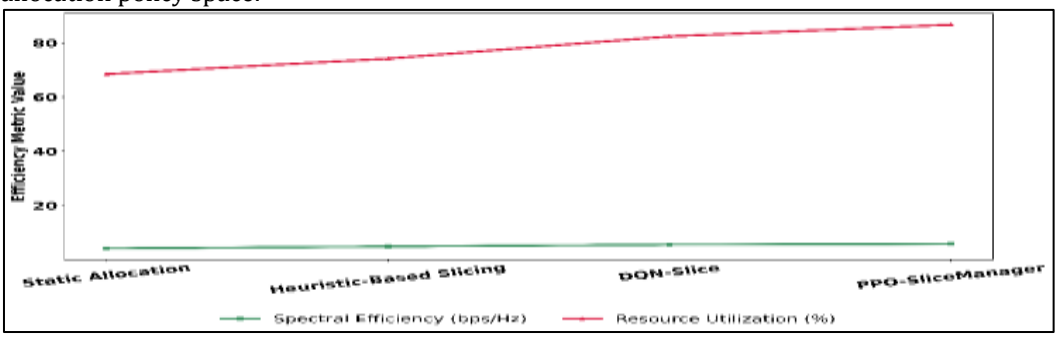

Fig.4. Spectral Efficiency and Resource Utilization Trend

PPO-SliceManager attains yet higher values of the lowest latency of 3.2 ms and the highest reliability of 99.94, which indicates the capacity to do sustained and stable policy optimization.

TABLE 3. RELIABILITY AND LATENCY PERFORMANCE ACROSS NETWORK SLICES

Slice Type	Avg. Latency (ms)	Latency Jitter (ms)	Packet Loss Rate (%)	Reliability Guarantee (%)
eMBB	9.4	2.1	0.42	99.5
mMTC	12.8	3.6	0.58	99.2
URLLC	1.9	0.4	0.01	99.999
URLLC (w/o Optimization)	3.7	1.3	0.08	99.85

Table 3 shows the comparison of reliability and latency performance of various types of network slices in the suggested G network slicing framework. The eMBB slice has a moderate latency of 9.4 ms with a reasonable degree of reliability, as it is concerned with high data throughput, but not tight delay limits. Figure 5 is a comparison of latency and jitter between network slices.

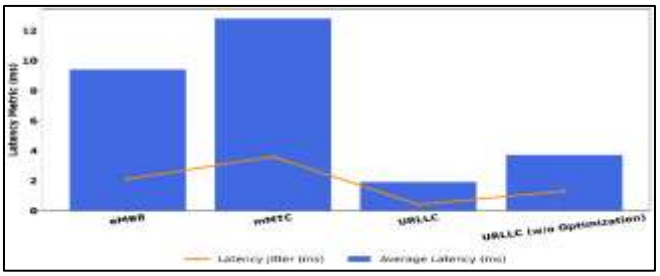

Fig.5. Latency and Jitter Comparison across Network Slice Types

The mMTC slice has a greater latency and jitter as it exhibits massive device connectivity and lower priority scheduling, which has slightly reduced reliability. Figure 6 indicates reliability and lost packets among the types of slices. The URLLC slice, in turn, exhibits a better performance with the lowest latency of 1.9 ms, the lowest jitter, and extremely low packet loss rate of 0.01% with the reliability guarantee of 99.999. The URLLC slice has to undergo a certain enhancement in its latency and packet loss, which is why the intelligent management of the resource matters.

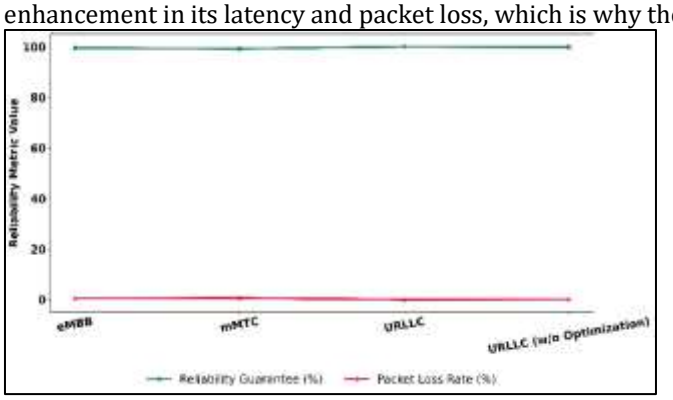

Fig.6. Reliability and Packet Loss Performance Across Slice Types

These findings attest the fact that reliability-conscious scheduling and adaptive resource allocation work wonders to the extreme performance of ultra-reliable communication under dynamic of the network slicing environments.

VII. CONCLUSION

This paper examined dynamic resource management of G network slicing with a particular emphasis on the provision of ultra-reliable communication in a heterogeneous and time-varying network environment. Network slicing was discussed as an underlying architectural tool that allows service-specific logical networks to co-exist on a commonly shared physical infrastructure, and meet varying quality-of-service needs. The offered framework incorporates slice-level resource abstraction, traffic prediction using machine learning, decision-making based on deep reinforcement learning, and reliability-conscious optimization to overcome the drawbacks of a fixed and rule-based resource allocation strategy. The findings reveal that proactive prediction of traffic and demand is very important in predicting the resource needs and avoiding the congestion-induced performance deterioration. Deep reinforcement learning agents successfully learn adaptive slicing policies that meet strong reliability and latency requirements on mission-critical services and make good use of network resources. The framework guarantees the uniformity and reliability of communication performance even in circumstances of uncertainty and dynamic operation by explicitly integrating probabilistic reliability guarantees, minimization of latency and fault tolerance mechanisms.

REFERENCES

[1] Masoudi, M.; Demir, O.T.; Zander, J.; Cavdar, C. Energy-Optimal End-to-End Network Slicing in Cloud-Based Architecture. IEEE Open J. Commun. Soc. 2022, 3, 574–592.

[2] Akin, O.; Gulmez, U.C.; Sazak, O.; Yagmur, O.U.; Angin, P. GreenSlice: An Energy-Efficient Secure Network Slicing Framework. J. Internet Serv. Inf. Secur. (JISIS) 2022, 2, 57–71.

[3] Rasti, M.; Taskou, S.K.; Tabassum, H.; Hossain, E. Evolution Toward 6G Multi-Band Wireless Networks: A Resource Management Perspective. IEEE Wirel. Commun. 2022, 29, 118–125.

[4] Hossain, A.R.; Liu, W.; Ansari, N.; Kiani, A.; Saboorian, T. AI-Native for 6G Core Network Configuration. IEEE Netw. Lett. 2023, 5, 255–259.

[5] Alwakeel, A.M.; Alnaim, A.K. Network Slicing in 6G: A Strategic Framework for IoT in Smart Cities. Sensors 2024, 24, 4254.

[6] Gabilando, A.; Fernandez, Z.; Viola, R.; Martín, Á.; Zorrilla, M.; Angueira, P.; Montalbán, J. Traffic Classification for Network Slicing in Mobile Networks. Electronics 2022, 11, 1097.

[7] S. K. Ramareddy, "Integrating AI-Driven Data Visualization and Data Warehouse Analytics for Predictive Decision-Making in Financial and Stock Market Systems," 2026 International Symposium of Systems, Advanced Technologies and Knowledge (ISSATK), Hammamet, Tunisia, 2026, pp. 1-6, doi: 10.1109/ISSATK69667.2026.11566934.

[8] Dangi, R.; Lalwani, P. Optimizing Network Slicing in 6G Networks through a Hybrid Deep Learning Strategy. J. Supercomput. 2024, 80, 20400–20420.

[9] Jiang, W.; Fricker, C.; Izal, M.; Rahman, M.; Woodward, M. Machine Learning for 5G Network Slicing and Orchestration: Opportunities, Challenges, and Solutions. IEEE Commun. Surv. Tutor. 2022, 24, 772–805.

[10] Sunkara, S. P. (2025). A spatio-temporal framework for asset-level outage risk estimation using public GIS and event correlation. International Journal of Computer Engineering and Technology, 16(1), 4211–4227. https://doi.org/10.34218/IJCET_16_01_286

[11] Nassar, A.; Yilmaz, Y. Deep Reinforcement Learning for Adaptive Network Slicing in 5G for Intelligent Vehicular Systems and Smart Cities. IEEE Internet Things J. 2022, 9, 222–235.

[12] Sharma, N.; Kumar, K. A Novel Latency-Aware Resource Allocation and Offloading Strategy with Improved Prioritization and DDQN for Edge-Enabled UDNs. IEEE Trans. Netw. Serv. Manag. 2024, 21, 6260–6272.

[13] Prasad, R. N. A. (2026). Quantum Computing Applications in Theoretical and Applied Physics: A Systematic Review. IJISEA Publications. https://doi.org/10.5281/zenodo.20082583

[14] D. R. Bhoyar, “The Role of Deep Learning in Network Security using Explainable Artificial Intelligence”, IJACECT, vol. 14, no. 2, pp. 1–8, Sep. 2025.