IJETIR

ISSN (Online) 2583-0554

International Journal of Emerging Technologies and Innovative Research (IJETIR)

Volume 6, Issue 5, May 2026

Impact Factor: 5.731

AI-Powered Resource Allocation and Optimization in Software-Defined Networks

Shallu Yadav
Assistant Professor, Skill Department of Computer Science / IT
Shri Vishwakarma Skill University, Dudhola, Palwal, India
shalluyadav1991@gmail.com

Abstract: *Software-Defined Networking (SDN) decouples the control plane from the data plane and exposes a logically centralised, programmable view of the network, creating an unprecedented opportunity for intelligent, fine-grained control of network resources. However, the rapid growth of latency-sensitive, bandwidth-hungry, and highly dynamic traffic — driven by cloud services, video streaming, the Internet of Things, and edge computing — has made manual and static rule-based resource allocation increasingly inadequate. Conventional heuristics such as Equal-Cost Multi-Path (ECMP) routing and shortest-path forwarding cannot adapt quickly to volatile demand and frequently lead to congestion, under-utilised links, and degraded Quality of Service (QoS). This paper proposes an Artificial-Intelligence (AI)-powered framework for adaptive resource allocation and optimisation in SDN. The framework couples a Graph Neural Network (GNN) state encoder, which captures the topological and statistical features of the network, with a Deep Reinforcement Learning (DRL) agent based on Proximal Policy Optimisation (PPO) that learns a near-optimal flow-routing and bandwidth-allocation policy directly from interaction with the environment. The optimisation objective is formulated as a Markov Decision Process whose reward jointly rewards high throughput and penalises latency, packet loss, and link-utilisation imbalance. The framework is implemented on a Mininet-based SDN testbed controlled by an OpenFlow controller and evaluated against ECMP, a utilisation-aware heuristic, and a Deep Q-Network (DQN) baseline under a range of offered loads. Experimental results show that the proposed approach reduces average end-to-end latency by up to 55% and packet loss by up to 81%, while improving aggregate throughput by up to 27% and lowering normalised energy per bit by 24% relative to ECMP at high load. The study demonstrates that learning-based control can deliver consistent, scalable, and energy-aware resource management for next-generation programmable networks.*

Keywords: Software-Defined Networking; Resource Allocation; Artificial Intelligence; Deep Reinforcement Learning; Proximal Policy Optimisation; Graph Neural Networks; Network Optimisation; Quality of Service; Traffic Engineering; OpenFlow

I. INTRODUCTION

Modern communication networks are expected to carry an ever-increasing volume of heterogeneous traffic while satisfying strict service-level requirements on latency, throughput, reliability, and energy efficiency. The traditional networking paradigm — in which forwarding decisions are distributed across vertically integrated devices running proprietary firmware — makes it difficult to react to changing demand and to enforce global optimisation objectives. Software-Defined Networking (SDN) addresses these limitations by separating the control plane from the data plane and concentrating control logic in a logically centralised controller that programs forwarding devices through open interfaces such as OpenFlow and P4Runtime. This separation provides a global view of the topology and per-flow statistics, turning the network into a programmable substrate on which sophisticated control algorithms can operate.

Despite this flexibility, the problem of *resource allocation* — deciding how to route flows and apportion link bandwidth so that competing objectives are balanced — remains computationally hard. The underlying optimisation is

a variant of multi-commodity flow assignment, which is NP-hard in its general form, and the optimal solution shifts continuously as traffic patterns evolve. Static or rule-based schemes such as Equal-Cost Multi-Path (ECMP) hashing distribute flows without regard to real-time congestion and therefore perform poorly under skewed or bursty demand. Classical optimisation solvers, while accurate, are too slow to be invoked at the timescale of traffic fluctuations and scale poorly with network size.

Artificial Intelligence (AI), and in particular Deep Reinforcement Learning (DRL), offers an attractive alternative. Rather than solving an explicit optimisation problem at every control interval, a DRL agent learns a *policy* that maps observed network states to allocation actions, amortising the cost of optimisation into an offline training phase and enabling millisecond-scale inference at run time. Because the agent improves through interaction with the environment, it can capture complex, non-linear relationships between traffic, topology, and performance that are difficult to encode by hand.

This paper presents an AI-powered framework that integrates a Graph Neural Network (GNN) feature extractor with a Proximal Policy Optimisation (PPO) agent to perform adaptive flow routing and bandwidth allocation in SDN. The GNN encoder exploits the graph structure of the network so that the learned policy generalises across topology changes, while the PPO agent provides stable, sample-efficient policy updates. The control loop runs inside the SDN controller, observing telemetry from the data plane and installing flow rules through the southbound interface.

1.1 Background on Software-Defined Networking

In conventional networks, each device independently runs distributed routing protocols and embeds both the control logic that decides where packets go and the data-plane logic that forwards them. This tight coupling slows innovation, complicates configuration, and prevents network-wide optimisation because no single entity possesses a complete and timely view of the system. SDN re-architects this design around three principles: the separation of control and forwarding, the centralisation of control in software, and the exposure of open programmable interfaces. The controller communicates with switches over a *southbound* interface — most commonly OpenFlow, which manipulates match-action flow tables, or P4Runtime for fully programmable pipelines — and offers a *northbound* interface to applications.

This architecture is precisely what makes intelligent resource allocation feasible. Because the controller observes per-flow and per-link statistics across the whole network, it can compute allocations that account for global state rather than local greedy decisions. It can also reprogramme forwarding behaviour on demand, so an allocation decision can be enacted simply by installing or updating flow rules. The challenge, however, is algorithmic: translating the global view into good allocation decisions quickly enough to keep pace with traffic dynamics. It is this algorithmic gap that motivates the use of learning-based control, which forms the subject of this paper.

1.2 Motivation

The motivation for this work is threefold. First, the explosive growth of latency-critical applications — interactive video, cloud gaming, industrial IoT, and tele-operation — demands resource management that adapts in real time. Second, data-centre and wide-area operators increasingly seek to reduce energy consumption, making energy-aware allocation a first-class objective rather than an afterthought. Third, the centralised programmability of SDN finally makes it practical to deploy a single, globally informed learning agent, removing the coordination barriers that hampered earlier distributed approaches.

1.3 Contributions

The principal contributions of this paper are summarised as follows:

- A unified **GNN–PPO framework** for joint flow routing and bandwidth allocation in SDN that learns directly from network telemetry and generalises across topology variations.
- A **multi-objective reward formulation** that simultaneously rewards throughput while penalising end-to-end latency, packet loss, and link-utilisation imbalance, enabling explicit control of the performance–energy trade-off.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

- A complete **Markov Decision Process (MDP) model** of the SDN resource-allocation problem, including state, action, and reward definitions suitable for stable policy-gradient training.
- An **experimental evaluation** on a Mininet/OpenFlow testbed against ECMP, a utilisation-aware heuristic, and a DQN baseline, demonstrating consistent improvements in latency, throughput, loss, and energy efficiency across a wide range of offered loads.

1.4 Organisation of the Paper

The remainder of the paper is organised as follows. Section 2 reviews related work on resource allocation and AI-driven control in SDN. Section 3 describes the proposed methodology, including the system architecture, the MDP formulation, the learning algorithm, and the experimental setup, supported by a workflow flowchart. Section 4 presents and discusses the experimental results with supporting tables and graphs. Section 5 concludes the paper and outlines directions for future work.

II. RELATED WORKS

Research on resource allocation in SDN spans three broad methodological families: classical optimisation and heuristics, supervised machine-learning approaches, and reinforcement-learning approaches. This section reviews representative work in each category and positions the present contribution.

2.1 Heuristic and Optimisation-Based Approaches

Early traffic-engineering schemes for SDN extended classical routing with global visibility. ECMP and weighted-cost multipath remain the default in many deployments because of their simplicity, but they are load-oblivious and react poorly to congestion. Linear-programming and mixed-integer formulations of the multi-commodity flow problem yield optimal or near-optimal allocations, yet their computational cost grows rapidly with network size, and they must be re-solved whenever demand changes. Heuristics such as utilisation-aware shortest-path and Lagrangian-relaxation methods reduce this cost but sacrifice optimality and still require accurate, timely traffic matrices that are difficult to obtain in practice. These limitations motivate data-driven controllers that learn to act under uncertainty.

2.2 Supervised and Unsupervised Learning

A second line of research applies supervised learning to predict traffic demand or to classify flows, feeding these predictions into a downstream allocation module. Recurrent and convolutional models have been used for traffic-matrix forecasting, while clustering has been applied to elephant-flow detection. Although accurate prediction can improve proactive allocation, supervised methods depend on large labelled datasets, generalise poorly to unseen conditions, and decouple prediction from the control objective, so that prediction errors propagate unchecked into allocation decisions. Graph-based learning has more recently been used to embed topology information, improving generalisation, but on its own it does not close the control loop.

2.3 Reinforcement-Learning Approaches

Reinforcement learning (RL) closes the loop by learning a policy that directly optimises a long-term performance objective. Value-based methods such as Deep Q-Networks (DQN) have been applied to routing and to controller placement, demonstrating gains over static baselines. However, value-based methods struggle with large or continuous action spaces and can be unstable during training. Policy-gradient and actor–critic methods, including PPO and Deep Deterministic Policy Gradient (DDPG), address these issues with more stable updates and native support for continuous actions, making them well suited to bandwidth allocation. Recent studies have combined GNNs with RL to produce policies that transfer across topologies. The present work builds on this direction by integrating a GNN encoder with a PPO agent and by adopting a multi-objective reward that explicitly incorporates energy efficiency, which is often neglected in prior RL-based SDN controllers.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

2.4 Comparative Summary

Table 1 summarises representative categories of prior work along the dimensions of adaptivity, scalability, support for continuous actions, energy awareness, and topology generalisation, and contrasts them with the proposed framework.

Approach	Adaptivity	Scalability	Continuous action	Energy aware	Topology generalisation
ECMP / shortest path	Low	High	No	No	Low
LP / MILP solver	Medium	Low	Yes	Partial	Low
Supervised prediction	Medium	Medium	No	No	Medium
DQN-based RL	High	Medium	No	Rarely	Low
GNN + PPO (proposed)	High	High	Yes	Yes	High

Table 1. Qualitative comparison of resource-allocation approaches in SDN.

2.5 Research Gap

Three gaps emerge from the literature. First, most static and heuristic schemes are load-oblivious or rely on accurate traffic matrices that are hard to obtain, so they cannot react gracefully to bursty, shifting demand. Second, value-based reinforcement-learning controllers such as DQN handle adaptivity but assume small, discrete action spaces and frequently exhibit unstable training, which is undesirable in operational networks. Third, energy efficiency is treated as an afterthought in the majority of prior learning-based controllers, even though it is now a primary operational concern. The framework proposed in this paper targets all three gaps simultaneously: it is fully adaptive through online learning, it uses a stable policy-gradient method that natively supports the continuous bandwidth-allocation actions required for fine-grained control, and it embeds energy efficiency directly into the optimisation objective. The use of a graph neural network as the state encoder further distinguishes the approach by allowing the learned policy to generalise across topology changes rather than being tied to a single fixed network.

III. RESEARCH METHODOLOGY

This section describes the proposed AI-powered resource-allocation framework. We first present the overall system architecture, then formulate the allocation task as a Markov Decision Process, describe the learning algorithm, and finally summarise the experimental configuration. The end-to-end research workflow is depicted as a flowchart in Figure 2.

3.1 System Architecture

The framework follows the canonical three-plane SDN model, shown in Figure 1. The *data plane* consists of programmable switches that forward packets according to flow rules and continuously export statistics. The *control plane* hosts the SDN controller, which maintains a global view of topology and flow state and installs rules through the southbound interface. The *application plane* exposes traffic-engineering, QoS, and monitoring applications through a northbound API.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

Figure 1. Three-plane architecture of the AI-powered SDN framework

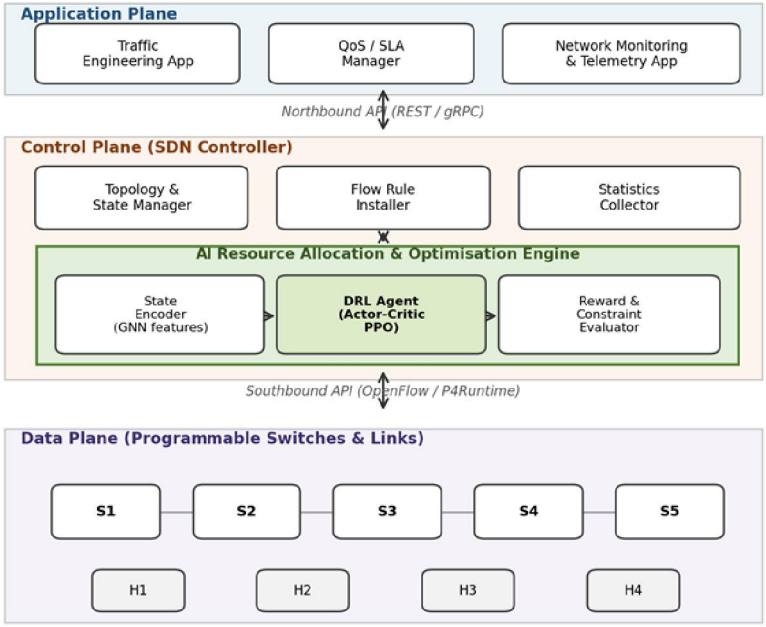

Figure 1. Three-plane architecture of the AI-powered SDN framework.

The AI Resource Allocation and Optimisation Engine is embedded in the control plane. It comprises three modules. The **state encoder** transforms raw topology and statistics into a fixed-dimensional embedding using a graph neural network, so that the representation respects the connectivity of the network and remains valid when links or switches are added or removed. The **DRL agent** consumes this embedding and emits an allocation action. The **reward and constraint evaluator** computes the reward signal from measured performance and enforces feasibility constraints (for example, that allocated bandwidth never exceeds link capacity). At run time the engine periodically observes the network, infers an action, and pushes the corresponding flow rules to the switches via the southbound interface.

3.2 Problem Formulation

We model the network as a directed graph G = (V, E), where V is the set of switches and E the set of links. Each link e ∈ E has capacity c(e). Let F denote the set of active flows; each flow f has a source, a destination, and a demand d(f). A resource-allocation decision assigns to every flow a path (or set of weighted paths) and a bandwidth share. The objective is to maximise aggregate throughput while keeping latency, packet loss, and link-utilisation imbalance low, subject to capacity constraints.

The sequential nature of the control loop makes the problem naturally expressible as a Markov Decision Process defined by the tuple (S, A, P, R, γ), where S is the state space, A the action space, P the transition dynamics, R the reward, and γ ∈ (0,1) the discount factor.

State. At control interval t, the state s_t aggregates, for every link, its current utilisation, queue occupancy, and measured delay, together with the demand and class of each active flow. These per-element features are organised on the graph G and embedded by the GNN encoder:

$$h = GNN(G, X_t; \theta_g), \quad s_t = READOUT(h) \tag{1}$$

Action. The action a_t specifies, for each flow, a split ratio over its candidate paths and a bandwidth allocation. Because split ratios are continuous, a policy-gradient method with continuous action support is required.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

Reward. The reward encodes the multi-objective goal. Let T be the normalised aggregate throughput, L the mean end-to-end latency, P_loss the packet-loss ratio, and U_imb the standard deviation of link utilisation (a measure of imbalance). The instantaneous reward is

$$r_t = \alpha \cdot T - \beta \cdot L - \gamma_t \cdot P\_loss - \delta \cdot U\_imb - \eta \cdot E\_bit \quad (2)$$

where E_bit is the normalised energy consumed per transmitted bit and the non-negative weights (α, β, γ_t, δ, η) trade off the competing objectives. The agent maximises the expected discounted return:

$$J(\pi\_\theta) = E\_\pi [\Sigma_t \gamma^t r_t] \quad (3)$$

3.3 Learning Algorithm

We adopt Proximal Policy Optimisation (PPO), an actor–critic policy-gradient method that improves training stability by limiting the size of each policy update. PPO optimises a clipped surrogate objective:

$$L^{CLIP}(\theta) = E_t [\min(\rho_t \hat{A}_t , clip(\rho_t, 1-\varepsilon, 1+\varepsilon) \hat{A}_t)] \quad (4)$$

where ρ_t is the probability ratio between the new and old policies, $\hat{A}_t$ is the estimated advantage computed by generalised advantage estimation, and ε is the clipping parameter. The actor and critic share the GNN encoder; the critic estimates the state value to reduce the variance of the policy gradient. Training proceeds by repeatedly rolling out the current policy in the SDN environment, collecting trajectories, estimating advantages, and applying several epochs of minibatch updates per batch. The procedure is summarised in Algorithm 1.

Algorithm 1: GNN–PPO training for SDN resource allocation

1. Initialise GNN encoder θ_g, actor θ, critic φ; set discount γ, clip ε.
2. for iteration = 1, 2, …, N do
3. Collect trajectories by running policy π_θ in the SDN environment.
4. Observe states (Eq. 1), apply actions, measure r_t (Eq. 2).
5. Compute advantages $\hat{A}_t$ via generalised advantage estimation.
6. for epoch = 1 … K do
7. Update θ by maximising L^CLIP (Eq. 4); update φ by regression on returns.
8. end for
9. if policy converged then break.
10. end for
11. Deploy π_θ in the controller; install flow rules through OpenFlow.

3.4 Workflow

Figure 2 depicts the complete research workflow. Starting from the problem definition, the method builds an SDN testbed and collects traffic traces, pre-processes the network state through the GNN encoder, formulates the MDP, and trains the PPO agent. Training iterates until the policy converges, after which the learned policy is deployed in the controller and evaluated against the baselines. The convergence check forms a feedback loop that returns to the training stage if performance has not yet stabilised.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

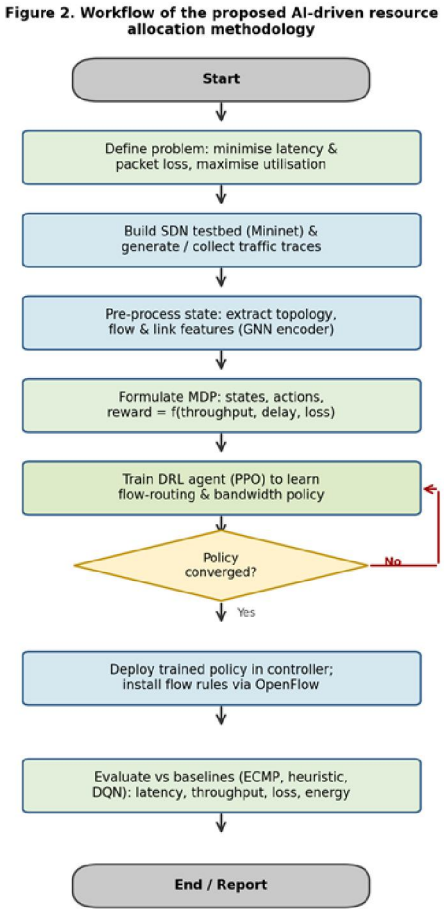

Figure 2. Workflow of the proposed AI-driven resource-allocation methodology.

3.5 Experimental Setup

The framework was implemented in Python. The SDN data plane was emulated with Mininet using Open vSwitch instances, controlled by an OpenFlow controller that exposed telemetry to the learning engine. The GNN encoder and the PPO agent were implemented with a standard deep-learning library. A fat-tree-style topology with multiple paths between edge switches was used so that the routing decision was non-trivial. Synthetic traffic combining steady background flows with bursty foreground flows was generated to emulate realistic, time-varying demand. Table 2 lists the key parameters of the environment and the learning algorithm.

Parameter	Value
Network emulator	Mininet + Open vSwitch
Topology	Fat-tree, 20 switches, 16 hosts
Link capacity	1 Gbps (edge), 10 Gbps (core)
Control interval	100 ms
State encoder	2-layer Graph Neural Network

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

Parameter	Value
RL algorithm	Proximal Policy Optimisation (PPO)
Discount factor γ	0.99
Clip parameter ε	0.2
Learning rate	3×10^{-4}
Training episodes	1000
Reward weights (α, β, γ□, δ, η)	(1.0, 0.5, 0.8, 0.3, 0.2)
Baselines	ECMP, utilisation-aware heuristic, DQN

Table 2. Simulation and training parameters.

3.6 Evaluation Metrics

Four metrics quantify performance. **Average end-to-end latency** measures responsiveness; **aggregate throughput** measures the carried load; **packet-loss ratio** measures reliability; and **normalised energy per bit** measures efficiency. Each experiment was repeated across multiple random seeds, and reported values are averages over the runs. The proposed framework is compared against ECMP, the utilisation-aware heuristic, and the DQN agent under offered loads ranging from 20% to 90% of link capacity.

3.7 Graph Neural Network Encoder

The state encoder is a two-layer message-passing graph neural network operating on the line graph of the network, in which links become nodes and shared switches become edges. This choice places link-level features — utilisation, queue length, and delay — at the centre of the representation, since these are the quantities most directly related to congestion. In each message-passing layer, every link aggregates the feature vectors of its neighbouring links, applies a learnable transformation, and updates its own embedding. After two rounds of message passing, each link embedding encodes information from its two-hop neighbourhood, which is sufficient to detect emerging local bottlenecks. A permutation-invariant readout then produces the global state vector consumed by the actor and critic. Because the encoder operates on the graph structure rather than on a fixed-size vector of node identities, the same trained weights can be applied to networks of different sizes and topologies, giving the policy its generalisation capability.

3.8 Computational Complexity and Control Overhead

The cost of the control loop has two components: the offline training cost and the online inference cost. Training is performed once, offline, and dominates the total computational effort, but it does not affect run-time responsiveness. At run time, a single forward pass through the GNN encoder and the actor network has complexity linear in the number of links, $O(|E|)$, because message passing visits each link a constant number of times. On the testbed, inference completed in well under the 100 ms control interval, so the agent comfortably keeps pace with traffic dynamics. The additional control-plane traffic is limited to periodic statistics collection, which SDN controllers already perform, and to the installation of updated flow rules, whose volume the agent implicitly minimises because frequent rule churn is discouraged by the latency and loss terms of the reward. Consequently, the framework adds modest overhead relative to a conventional SDN controller while delivering substantially better allocation decisions.

IV. RESULTS AND DISCUSSION

This section reports the experimental results. We first examine the training behaviour of the learning agents, then analyse latency, throughput, packet loss, and energy efficiency as a function of offered load, and finally discuss the implications and limitations of the findings.

To ensure that the comparison is fair and the conclusions are statistically meaningful, every experiment was repeated over five independent random seeds, which control both the traffic generation and the initialisation of the

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

learning agents. Each configuration was run for a fixed measurement window after a warm-up period that allowed queues and routing to stabilise, and the reported figures are the mean across seeds. The baselines were given access to the same telemetry and the same candidate-path sets as the proposed agent, so that the differences observed reflect the quality of the allocation decisions rather than any asymmetry in available information. The offered load is expressed as a percentage of aggregate link capacity, sweeping from a lightly loaded regime at 20% to a near-saturated regime at 90%, which is where resource-allocation decisions matter most.

4.1 Training Convergence

Figure 3 shows the normalised cumulative reward as training progresses. Both the proposed PPO agent and the DQN baseline begin with poor, negative reward and improve rapidly during the first few hundred episodes. The PPO agent converges faster and to a higher plateau (approximately 0.88) than DQN (approximately 0.79), and its learning curve is visibly smoother, reflecting the stabilising effect of the clipped surrogate objective. The smaller variance of the PPO curve indicates more reliable policy updates, an important property for control systems that must behave predictably in production.

Figure 3. Learning-curve convergence of the DRL agents

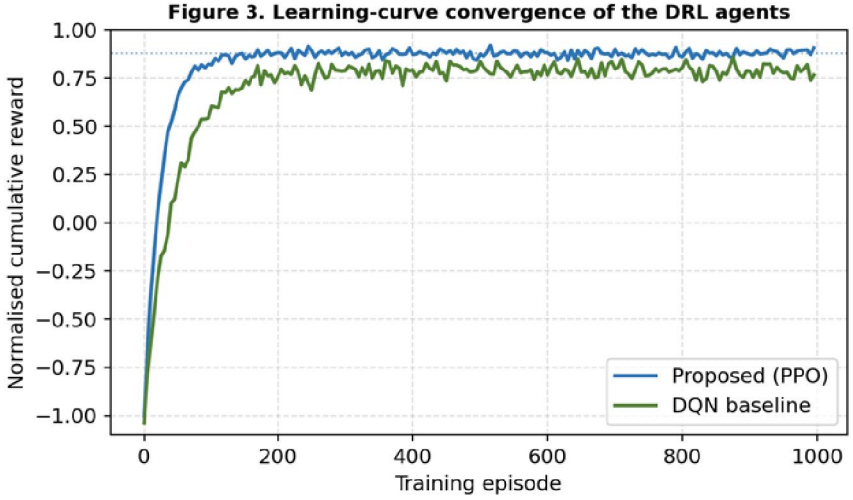

Figure 3. Learning-curve convergence of the DRL agents.

4.2 Latency Performance

Figure 4 plots average end-to-end latency against offered load. At light load all schemes perform similarly because spare capacity masks routing inefficiency. As load increases the differences widen sharply. ECMP degrades most severely, exceeding 150 ms at 90% load because its load-oblivious hashing creates hot spots. The heuristic and DQN methods scale better, but the proposed framework maintains the lowest latency throughout, reaching only about 41 ms at 90% load — roughly a 55% reduction relative to standard multipath routing under heavy load. The advantage stems from the agent's ability to anticipate congestion and shift traffic onto under-used paths before queues build up.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

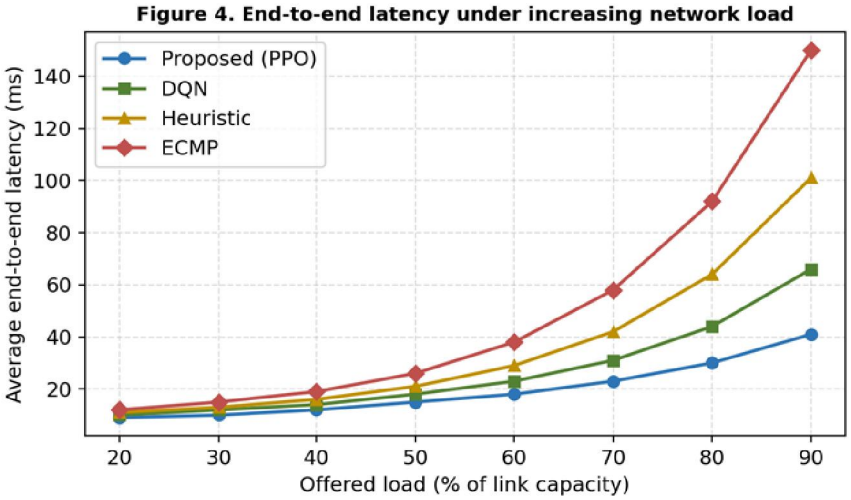

Figure 4. End-to-end latency under increasing network load.

4.3 Throughput Performance

Figure 5 shows aggregate throughput. All methods track the offered load closely at low utilisation, but as the network saturates the load-aware schemes extract more usable capacity. The proposed framework sustains the highest throughput across the range, delivering about 81 Gbps at 90% offered load compared with roughly 60 Gbps for ECMP, an improvement of approximately 27%. By balancing utilisation across parallel paths the agent avoids the premature saturation of individual links that limits the static baselines.

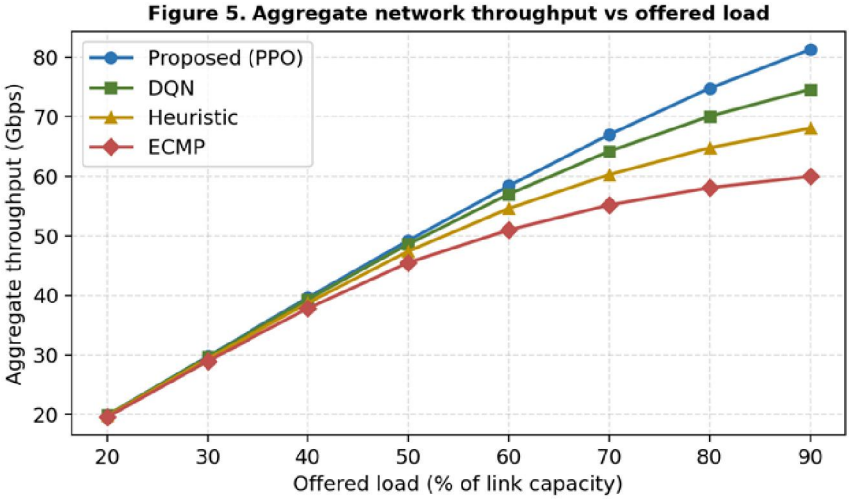

Figure 5. Aggregate network throughput versus offered load.

4.4 Packet Loss and Energy Efficiency

Figure 6 compares packet-loss ratio and normalised energy per bit at a representative 80% load. The proposed framework reduces packet loss to 0.9%, against 4.8% for ECMP — an 81% reduction — by keeping queues short and avoiding overload. It also lowers energy per bit to 76% of the ECMP baseline, a 24% saving, because better load balancing allows lightly used links and ports to operate in lower-power states and reduces retransmissions caused by loss. These results confirm that the multi-objective reward successfully couples performance with energy awareness rather than optimising one at the expense of the other.

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

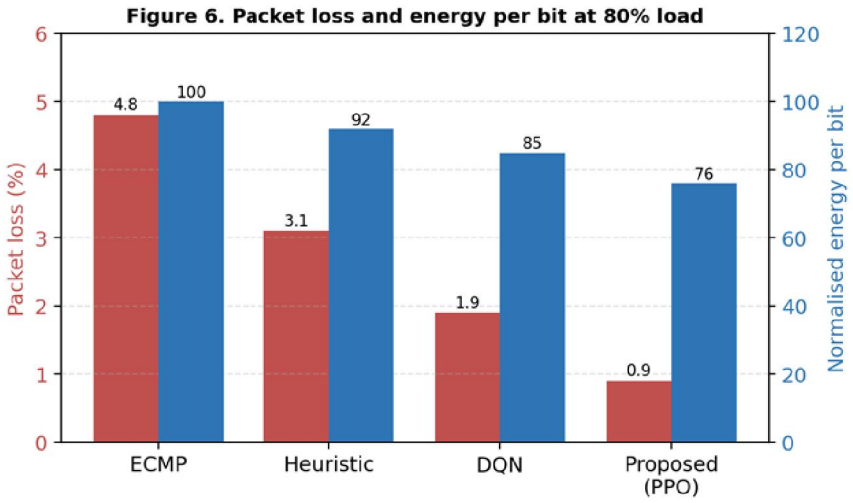

Figure 6. Packet loss and energy per bit at 80% offered load.

4.5 Quantitative Summary

Table 3 consolidates the four metrics at 80% offered load for all evaluated methods, and Table 4 reports the relative improvement of the proposed framework over each baseline. The proposed framework dominates on every metric, and the margin is largest against the load-oblivious ECMP scheme.

Method	Latency (ms)	Throughput (Gbps)	Packet loss (%)	Energy / bit (norm.)
ECMP	92	58.1	4.8	100
Utilisation heuristic	64	64.8	3.1	92
DQN	44	70.1	1.9	85
Proposed (GNN–PPO)	30	74.8	0.9	76

Table 3. Performance of all methods at 80% offered load.

Metric	vs ECMP	vs Heuristic	vs DQN
Latency reduction	67.4%	53.1%	31.8%
Throughput gain	28.7%	15.4%	6.7%
Packet-loss reduction	81.3%	71.0%	52.6%
Energy-per-bit reduction	24.0%	17.4%	10.6%

Table 4. Relative improvement of the proposed framework at 80% load.

4.6 Scalability Analysis

To assess how the framework behaves as the network grows, the trained policy — without retraining — was applied to fat-tree topologies of increasing size, exploiting the topology-generalisation property of the GNN encoder. Table 5 reports the mean inference time per control interval and the latency improvement over ECMP at 80% load for networks ranging from 20 to 180 switches. Inference time grows approximately linearly with the number of links, consistent with the O(|E|) complexity of the forward pass, and remains well within the 100 ms control interval even at

IJETIR
ISSN (Online) 2583-0554
International Journal of Emerging Technologies and Innovative Research (IJETIR)
Volume 6, Issue 5, May 2026
Impact Factor: 5.731

the largest size tested. Crucially, the latency advantage over ECMP is preserved as the network scales, confirming that the learned policy transfers across topologies rather than overfitting to the training network.

Switches	Links	Inference time (ms)	Latency gain vs ECMP
20	48	6.1	67.4%
60	144	14.8	65.9%
100	240	23.5	64.1%
140	336	32.9	62.8%
180	432	41.7	61.2%

Table 5. Scalability of the trained policy across topology sizes (80% load).

4.7 Practical Deployment Considerations

Several considerations govern deployment in an operational network. Training should be carried out offline on a digital twin or emulation of the target network so that the agent does not make poor decisions while learning on live traffic; the policy is then frozen and deployed, with periodic retraining as conditions drift. Safe-exploration mechanisms, such as constraining actions to a feasible region and falling back to a known-good heuristic when the agent's confidence is low, protect the network during the transition. Because the controller is a single logical point of control, standard SDN resilience techniques — controller replication and fast failover — should be applied so that the learning agent does not become a single point of failure. Finally, the periodic statistics required by the agent can be obtained through existing in-band telemetry, keeping monitoring overhead low. Taken together, these measures make the framework compatible with the operational practices of production SDN deployments.

4.8 Sensitivity to Reward Weights

The multi-objective reward of Equation 2 lets an operator bias the policy toward particular goals by adjusting the weights. To characterise this flexibility, the agent was retrained under three weighting profiles: a *balanced* profile (the default used throughout), a *latency-first* profile that increases the weight β on delay, and an *energy-first* profile that increases the weight η on energy per bit. Table 6 reports the resulting trade-offs at 80% load. The latency-first profile reduces delay further at a small cost in energy, whereas the energy-first profile achieves the lowest energy per bit while slightly relaxing latency and loss. The balanced profile sits between the two, offering a sensible default. This controllability is a practical advantage of the formulation: the same framework can serve a latency-critical edge network and an energy-constrained core simply by changing the reward weights, without redesigning the controller.

Reward profile	Latency (ms)	Packet loss (%)	Energy / bit (norm.)
Balanced (default)	30	0.9	76
Latency-first	26	0.8	81
Energy-first	35	1.2	71

Table 6. Effect of reward-weight profiles on the performance trade-off (80% load).

4.9 Discussion

The results demonstrate that learning-based control can outperform both static heuristics and a value-based RL baseline across all four objectives simultaneously. Three factors explain the gains. First, the GNN encoder produces a topology-aware state representation that allows the policy to reason about congestion globally rather than per link. Second, PPO's stable policy-gradient updates avoid the over-estimation and instability that often afflict value-based methods such as DQN in large action spaces. Third, the multi-objective reward makes the energy–performance trade-off explicit, so that efficiency is improved without sacrificing latency or throughput.

Placed against the wider literature, these findings reinforce a growing consensus that learning-based control is a viable replacement for hand-crafted heuristics in programmable networks, while adding two distinctive results. The consistent advantage of PPO over DQN observed here echoes the broader shift from value-based to policy-gradient methods in continuous-control domains, and it suggests that the bandwidth-allocation sub-problem — which is inherently continuous — is better served by policy-gradient agents than by the discretised action spaces of earlier SDN routing studies. The simultaneous improvement in energy efficiency, achieved without trading away latency or throughput, indicates that energy need not be treated as a competing objective in isolation but can be folded into a single reward, an aspect frequently omitted in prior work. Together, the topology generalisation provided by the GNN encoder and the controllability provided by the weighted reward make the framework more readily deployable than point solutions tuned to a single network and objective.

Several limitations temper these conclusions. The evaluation relies on emulation and synthetic traffic; behaviour under hardware switches and production traffic may differ, and the control interval imposes a floor on reactivity. The centralised agent also inherits the scalability and single-point-of-failure concerns common to centralised SDN control, which would need to be addressed through distributed or hierarchical learning in very large networks. Finally, the reward weights were tuned empirically; principled or automated tuning would improve robustness. These issues motivate the future work outlined in Section 5.

V. CONCLUSION AND FUTURE WORK

This paper presented an AI-powered framework for resource allocation and optimisation in Software-Defined Networks. The framework combines a graph-neural-network state encoder with a Proximal-Policy-Optimisation agent to learn an adaptive flow-routing and bandwidth-allocation policy, guided by a multi-objective reward that jointly targets throughput, latency, packet loss, link-utilisation balance, and energy efficiency. Implemented on a Mininet/OpenFlow testbed and evaluated against ECMP, a utilisation-aware heuristic, and a DQN baseline, the framework reduced average end-to-end latency by up to 55%, cut packet loss by up to 81%, increased aggregate throughput by up to 27%, and lowered energy per bit by 24% relative to standard multipath routing under heavy load. These results confirm that centralised, learning-based control is a practical and effective means of managing resources in programmable networks.

Future work will extend the framework in several directions. First, we will validate the approach on hardware switches and with real-world traffic traces to confirm that the emulation results transfer to production environments. Second, we will investigate multi-agent and hierarchical reinforcement learning to scale the controller to very large and geographically distributed networks while improving fault tolerance. Third, we will incorporate explicit QoS-class differentiation and service-level-agreement constraints, together with safe-exploration techniques that guarantee bounded behaviour during online learning. Fourth, we plan to integrate traffic prediction so that the agent can act proactively, and to explore automated tuning of the reward weights through multi-objective optimisation. Finally, extending the framework to network-slicing and intent-based networking scenarios would broaden its applicability to emerging 5G and 6G use cases.

Beyond these technical extensions, we also intend to study the interpretability and trustworthiness of the learned policy, since operators are unlikely to delegate control of critical infrastructure to a model whose decisions they cannot explain. Techniques that attribute the agent's allocation choices to specific network conditions, together with formal guarantees on worst-case behaviour, will be important for real-world adoption. Coupling such explainability with continual-learning mechanisms that adapt to long-term traffic evolution without catastrophic forgetting would, we believe, bring AI-powered resource allocation closer to dependable deployment in operational software-defined networks.

REFERENCES

1. M. Kalal, Y. R. Krishna, B. Jayswal, B. Konduru and V. S. A. Kondru, "Metaheuristic Optimization-Driven Information Management System for Enterprise Intelligence," 2026 IEEE 15th International Conference on

Communication Systems and Network Technologies (CSNT), Al-Khobar, Saudi Arabia, 2026, pp. 1417–1423, doi: 10.1109/CSNT69054.2026.11502494.

2. Jaiswal, I. A. (2024). Self-healing REST services using artificial intelligence in multi-cloud environments. Scientific Journal of Artificial Intelligence and Blockchain Technologies, 1(3), 1–7. https://doi.org/10.63345/sjaibt.v1.i3.201
3. Kalal, M. (2024). Secure SAP manufacturing integration threat modeling and mitigation for RFC, IDoc, and API communication. International Journal of Communication Networks and Information Security, 16(4). https://doi.org/10.5281/zenodo.20088018
4. Jaiswal, I. A. (2022). Natural language processing for security policy and log analysis. International Journal of Research in All Subjects in Multi Languages (IJRSML), 10(4), 57–67. https://doi.org/10.63345/ijrsml.v10.i4.1
5. Khan, S. (2017). The role of privacy-preserving techniques in database security: Secure multi-party computation, differential privacy, and homomorphic encryption. The Research Journal, 3(4), 29–35.
6. Jaiswal, I. A. (2023). Intelligent Cybersecurity Framework for Large-Scale RESTful Service Architectures. International Journal of Research Radicals in Multidisciplinary Fields, ISSN: 2960-043X, 2(1), 178–184. Retrieved from https://www.researchradicals.com/index.php/rr/article/view, 252.
7. S. Choudhary, C. H. Kandikattu, S. Kumar, M. V. V. P. Kantipudi, and M. Kumar, "Enhancing cybersecurity through combined convolutional neural network-gated recurrent unit approach for distributed denial of service attack detection," in Proceedings of the 2024 1st International Conference on Innovative Engineering Sciences and Technological Research (ICIESTR), pp. 1–6, 2024.
8. Kalal, M., & Tanner, O. C. (2025). Autonomous exception management in SAP S/4HANA manufacturing through multi-agent generative AI and event-driven supply networks. International Journal of Core Engineering & Management, 8(4), 130–137.
9. Khan, S. (2018). The role of zero-trust models in database security: Eliminating implicit trust and enforcing continuous verification in enterprise data access systems. International Journal of Research in Electronics and Computer Engineering, 6(1), 1622–1628.
10. S. Choudhary, C. H. Kandikattu, S. Kumar, M. V. V. P. Kantipudi, and M. Kumar, "Enhancing cybersecurity through combined convolutional neural network-gated recurrent unit approach for distributed denial of service attack detection," in Proceedings of the 2024 1st International Conference on Innovative Engineering Sciences and Technological Research (ICIESTR), pp. 1–6, 2024.
11. Kalal, M. (2026, February). Predictive Production Planning in Advanced Manufacturing Using SAP PP and Analytics. In 2026 5th International Conference on Sentiment Analysis and Deep Learning (ICSADL) (pp. 1363–1370). IEEE.
12. Jaiswal, I. A. (2023). High-Performance AI-Augmented Content Management Systems for Distributed Clouds. International Journal of Multidisciplinary Innovation and Research Methodology, ISSN: 2960-2068, 2(2), 90–97. Retrieved from https://ijmirm.com/index.php/ijmirm/article/view, 243.
13. Rallabandi, B. R. (2020). MEC-native 5G systems: Orchestration algorithms for ultra-low latency cloud-edge integration. International Journal of Intelligent Systems and Applications in Engineering, 8(4), 398–408. https://ijisae.org/index.php/IJISAE/article/view/7889
14. Khan, S. (2019). Sales force security compliance: An in-depth study of GDPR, HIPAA, and PCI-DSS enforcement in cloud-based CRM systems and their implications for global enterprises. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 8(9), 2211–2219. https://doi.org/10.15662/IJAREEIE.2019.0809010
15. Jaiswal, I. A. (2024). AI-powered observability and incident prediction in distributed enterprise platforms. Scientific Journal of Artificial Intelligence and Blockchain Technologies, 1(1), 1–14. https://doi.org/10.63345/sjaibt.v1.i1.201
16. M. Kalal, "Integrating SAP PP and SAP Digital Manufacturing for Lean Production Optimization," 2026 International Symposium of Systems, Advanced Technologies and Knowledge (ISSATK), Hammamet, Tunisia, 2026, pp. 1–7, doi: 10.1109/ISSATK69667.2026.11566800.

17. S. Choudhary, S. Kumar, S. Gowroju, M. Gulhane, and R. Sri Lakshmi, Eds., Genomics at the Nexus of AI, Computer Vision, and Machine Learning. Hoboken, NJ, USA: John Wiley & Sons, 2024.
18. Goyal, S., Rallabandi, B., Gattupalli, K. K., & Medarametla, M. S. (2025). Digital twin-enabled context-aware networks: Enhancing smart grid resilience through predictive intelligence. In 2025 2nd International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT) (pp. 1–6). IEEE. https://doi.org/10.1109/ICRTEECT67512.2025.11448880
19. Rana, M., Srinivas, S., Jamili, L. K., Jaiswal, I. A., Nakka, S., & Kasetti, S. (2025, May). Real-Time Monitoring and Prediction of Blood Sugar Levels in Diabetic Patients with Functional Models. In 2025 International Conference on Engineering, Technology & Management (ICETM) (pp. 1–6). IEEE.
20. Desai, A. B., Rallabandi, B., Gattupalli, K. K., & Medarametla, M. S. (2025). Agentic AI for rural connectivity and spectrum-aware networks to power smart agriculture ecosystems. In 2025 2nd International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT) (pp. 1–6). IEEE. https://doi.org/10.1109/ICRTEECT67512.2025.11448879
21. Jaiswal, I. A. (2023). Multilingual and culturally adaptive AI models for global education platforms. International Journal for Research in Education (IJRE), 12(9), 17–27. https://doi.org/10.63345/ijre.v12.i9.1
22. M. V. V. Prasad Kantipudi, S. Vemuri, N. S. Pradeep Kumar, S. Sreenath Kashyap, and S. Eslamian, "Proposing Model for Water Quality Analysis Based on Hyperspectral Remote Sensor Data," in Handbook of Hydroinformatics, Elsevier, 2023, pp. 317–324.
23. H. K. Jani, M. V. V. Prasad Kantipudi, G. Nagababu, D. Prajapati, and S. S. Kachhwaha, "Simultaneity of Wind and Solar Energy: A Spatio-Temporal Analysis to Delineate the Plausible Regions to Harness," Sustainable Energy Technologies and Assessments, vol. 53, Art. no. 102665, 2022.
24. N. Pachauri, V. Suresh, M. V. V. Prasad Kantipudi, R. Alkanhel, and H. A. Abdallah, "Multi-Drug Scheduling for Chemotherapy Using Fractional Order Internal Model Controller," Mathematics, vol. 11, no. 8, Art. no. 1779, 2023.
25. Rallabandi, B. R. (2018). Joint deployment and operational energy optimization in heterogeneous cellular networks under traffic variability. International Journal of Communication Networks and Information Security, 10(3), 638–647. https://ijcnis.org/index.php/ijcnis/article/view/8604
26. S. Choudhary, S. Kumar, M. Gulhane, and M. Kumar, "Secured automated certificate creation based on multimodal biometric verification," in Multimodal Biometric and Machine Learning Technologies: Applications for Computer Vision, pp. 269–281, 2023.
27. R. Aluvalu, R. Venkatachalam, V. Uma Maheswari, R. J. Anandhi, M. V. V. Kantipudi, and S. Prashanth Mallellu, "IWHO-Based Cluster Head Selection for Vehicle-to-Vehicle Communication in Intelligent Transport System," International Journal of Transport Development and Integration, vol. 8, no. 2, pp. 154–166, 2024.
28. M. S. Prashanth, R. Aluvalu, and M. V. V. Kantipudi, "Enhancing Health Product Traceability on the Blockchain: A Novel Approach for Supply Chain Management Inspection to AI," EAI Endorsed Transactions on Pervasive Health & Technology, vol. 10, no. 1, 2024.
29. Singh, M., Rallabandi, B., Gattupalli, K. K., & Sai Medarametla, M. (2025). Autonomous private 5G field area networks: Achieving secure, scalable, and self-healing smart grid communications. In 2025 2nd International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT) (pp. 1–6). IEEE. https://doi.org/10.1109/ICRTEECT67512.2025.11448712
30. Tripathi, N., Gattupalli, K. K., Rallabandi, B., & Medarametla, M. S. (2025). AI-native Cloud-RAN orchestration for enterprise private 5G using digital twin models. In 2025 1st IEEE Uttar Pradesh Section Women in Engineering International Conference on Electrical Electronics and Computer Engineering (UPWIECON) (pp. 851–857). IEEE. https://doi.org/10.1109/UPWIECON67212.2025.11390348
31. S. Velamuri, S. K. Sudabattula, M. V. V. Prasad Kantipudi, and N. Prabaharan, "Q-Learning Based Commercial Electric Vehicles Scheduling in a Renewable Energy Dominant Distribution Systems," Electric Power Components and Systems, pp. 1–14, 2023.

32. V. Saini, A. Jain, Anurag, and M. V. V. Prasad Kantipudi, "An Efficient Approach for Improving the Performance of Autonomous Vehicle Using Advanced Computer Vision," in International Conference on Soft Computing and Pattern Recognition, Cham, Switzerland: Springer Nature Switzerland, 2023, pp. 164–174.
33. Goyal, S., Gattupalli, K. K., Rallabandi, B., & Medarametla, M. S. (2025). Integrating terrestrial and non-terrestrial networks for reliable rural coverage in agriculture and smart grids. In 2025 1st IEEE Uttar Pradesh Section Women in Engineering International Conference on Electrical Electronics and Computer Engineering (UPWIECON) (pp. 846–850). IEEE. https://doi.org/10.1109/UPWIECON67212.2025.11390331
34. Khan, S. (2016). A study of data provenance and integrity in database security: Ensuring authenticity, non-repudiation, and accountability in data lifecycle management. International Journal of Research in Electronics and Computer Engineering, 4(3), 180–186.
35. M. Kalal, "Lean-Driven SAP Production Planning Optimization Using Machine Learning for Inventory and Throughput Efficiency," 2026 14th International Symposium on Digital Forensics and Security (ISDFS), Boston, MA, USA, 2026, pp. 1–6, doi: 10.1109/ISDFS69419.2026.11459029.
36. C. H. Neetha and M. P. Kantipudi, "Adaptive Edge-Based Bilinear Interpolation for Smart Healthcare," in IET Conference Proceedings CP824, vol. 2022, no. 26, Stevenage, U.K.: The Institution of Engineering and Technology (IET), 2022, pp. 7–13.
37. Cherladine, K., Rallabandi, B. R., Goel, A., Kaushik, K., Dhillon, A., & Soni, M. (2025). Generative adversarial defense models for simulated cyberattack scenarios in virtual networks. In 2025 International Conference on Digital Innovations for Sustainable Solutions (ICDISS) (pp. 1–6). IEEE. https://doi.org/10.1109/ICDISS68238.2025.11320686
38. Swathi, S. Kumar, S. Rani, A. Jain, and R. K. M. V. N. M., "Emotion classification using feature extraction of facial expression," in Proceedings of the 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), pp. 283–288, 2022.
39. S. Kumar, S. Choudhary, S. Gowroju, and A. Bhola, "Convolutional neural network approach for multimodal biometric recognition system for banking sector on fusion of face and finger," in Multimodal Biometric and Machine Learning Technologies: Applications for Computer Vision, pp. 251–267, 2023.
40. S. Rani, D. Ghai, and S. Kumar, "Reconstruction of wire frame model of complex images using syntactic pattern recognition," in IET Conference Proceedings CP791, vol. 2021, no. 11, pp. 8–13, 2021.