INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

Hybrid Proximal Policy Optimization based Adaptive Link Load Balancing with INT in Secure SDN-IoT Networks

A Sandanasamy[1]*, P Joseph Charles[2]

1 Department of Computer Applications, Bishop Heber College, Affiliated to Bharathidasan University, Trichirappalli-620 017, Tamil Nadu, India
2 Department of Computer Science, St. Joseph's College, Affiliated to Bharathidasan University, Trichirappalli-620 002, Tamil Nadu, India

OPEN ACCESS

Received: 31/05/2025

Accepted: 06/06/2025

Published: 18/06/2025

Citation: Sandanasamy A, Joseph Charles P (2025) Hybrid Proximal Policy Optimization based Adaptive Link Load Balancing with INT in Secure SDN-IoT Networks. Indian Journal of Science and Technology 18(24): 1915-1930. https://doi.org/10.17485/IJST/v18i24.1007

* **Corresponding author**.

sandanasamy.ca@bhc.edu.in

Funding: DST-FIST

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Background/Objectives: The Internet of things with software defined networking has evolved and is facing vigorous traffic patterns and security threats in the realm of data communication. Balancing the load becomes a challenge with IoT devices. The objective is to develop Hybrid Proximal Policy Optimization based adaptive link load balancing method to address the issues of dynamic traffic patterns and security threats in SDN-IoT, thereby enhancing network performance. **Method:** The research follows a two level architecture which includes a centralized SDN controller which is responsible for training policy with global telemetry and switches that perform real time inference using local state observation. The system constructs the state representations using live INT metrics. Link utilization, queue length, delay, packet-loss and trust score help to find the robust path cost estimation. The PPO selects the optimal forwarding paths aided by a dynamic reward function. The policy is synchronized with edge switches for low-latency, decentralized decision-making. **Findings:** The effectiveness of the proposed algorithm is validated under normal and adversarial conditions. While comparing ECMP, DQN, A3C and Trust AODV, the proposed method demonstrates superior scalability with 37% reduction in 99th percentile latency. Efficient load balancing is achieved with 60-62% stable link utilization and 75 ms threat response latency at 90% attack intensity. The F1 score of 95% for DDoS, 92% for INT spoofing indicates its higher detection accuracy. These results show that the significant improvements are achieved in load distribution, reduced latency, and in security resiliency. **Novelty:** This research article uniquely integrates real time telemetry, trust based security model and deep reinforcement learning model to achieve load balancing in SDN IoT network. The real time INT data and trust metrics enables decentralized and low-latency path selection. The balance between QoS and security is achieved by the dynamic reward function providing a robust and scalable solution.

Keywords: Load balancing; SDN-IoT; Link load balancing; INT; Trust aware routing; QoS

1 Introduction

The network infrastructures have evolved over the years, especially it has a new façade as the software defined network is being integrated with Internet of Things. This enhances the programmability, scalability and centralized control over heterogeneous and resource-constrained devices. Dynamic traffic management and policy enforcement is made possible in SDN-IoT networks because of the possibility of handling the control and data planes separately.[(1)] Nevertheless, this integration of SDN with IoT has brought about a number of challenges including link congestion, large latencies, non-persistent Quality of Service and exposure to security threats. This becomes a concern which demands intelligent and adaptive traffic solutions. Load balancing serves as a solution by guaranteeing the performance and the reliability of SDN-IoT systems.[(2)] There were several research done in these aspects.

Chen et.al[(3)] developed Reinforcement Learning based traffic distribution in SDN framework. This approach uses an improved Deep Deterministic Policy Gradient (DDPG) algorithm integrated with SumTree based prioritizing reply mechanism. The simulation results of the approach demonstrates that higher throughput, lower latency and improved network stability is achieved while comparing with existing RL-based routing algorithms. However, it lacks scalability in dynamic networks. Bhowmik and Gayen[(4)] proposed a Genetic Algorithm (GA)-based dynamic load distribution technique for managing traffic at the data layer of Software Defined Networks. The proposed work focuses on the issue of traffic congestion and imbalance in the load caused by uneven flow distribution. The real time traffic load on the links and switches are monitored, overloaded components are identified, and the data flow is rerouted. The rerouting is done by GA which evaluates alternate paths that uses fitness function which considers link utilization, latency and path length. The evaluation result exhibits reduction in average network load with minimal latency. However, because of the nature GA, longer convergence time and potential suboptimal decision can happen in high volatile traffic.

Chandroth Jisi et all[(5)] proposed path prediction mechanism using Reinforcement Learning in SD-IoT. This research focuses on Quality of Service (QoS) challenges in IoT environments using Software-Defined Networking (SDN). In this approach, support vector regression is used for predicting reliability of the network links. The path selection is done using a reinforcement learning mechanism. The performance evaluation shows improved performance in throughput, delay, jitter and packet loss. However, this mechanism lacks security and trust metrics which are crucial for IoT networks. Duan, Y et all[(6)] designed MFGAD-INT, a novel anomaly detection approach for cloud data center networks. The limitation such as detecting subtle anomalies like microburst is overcome in MFGAD-INT by using telemetry data with high frequency to collect spatial and temporal network metrics. Graph attention networks (GAT) and Gated Recurrent Units (GRU) are used to construct dynamic network and predict anomalies efficiently. The authors claim that the model has a significant improvement in detection accuracy while comparing non-telemetry frequency. While this method identifies anomalies, it does not integrate mitigation strategies.

P. Zhang et al.[(7)] developed the mechanism of distributing load using adaptive in-band Network Telemetry System with Deep Reinforcement learning. Auxiliary Probes (Aps) and Dynamics Probes (DPs) are the two dual timescale telemetry system used by the authors. The algorithms used for this approach are Depth-First Search mechanism focusing full network coverage and Deep Reinforcement Learning path deployment mechanism with transfer learning, RNN. Experimental results show that the proposed method performs better with reduced telemetry latency in latency sensitive applications. Xinglong Pei and et.al.[(8)] proposed Deep Reinforcement Learning based Traffic distribution in SDN. This method aims to provide routing efficiency incorporating Deep Reinforcement Learning in the framework of Software-Defined Networking. In this proposed method authors claim that rerouting decisions are made by using real time traffic and topology analysis to identify critical links that are prone to congestion. The experimental evaluation states that the proposed method improves maximum link utilization and achieves a better load balancing performance. However, it lacks in secure traffic distribution. Kumar et al.[(9)] address the problem of load imbalance in Industrial IoT environments. They suggested an approximation based solution for load balancing in links and controller. But approximation based approach may lack quick adaptation to dynamic environment with unexpected network anomalies.

1.1 Research Gap

Several research are done in load balancing and traffic optimization in SDN IoT using reinforcement learning, genetic algorithms and in-band telemetry. But they have limitations. The study shows the limitations such as lack of scalability in dynamic networks, slow convergence, absence of trust and security, inability in mitigating detected anomalies. Some methods focus on anomaly detection, but they don't integrate mitigation strategies. Some methods lack real time decision making capacities. There is also absence of path selection mechanism in adversarial conditions. These gaps indicate the need for a robust, intelligent, secure load balancing mechanism in SDN IoT environments. To address these problems, a new hybrid method that

integrates Proximal Policy Optimization (PPO), a contemporary deep reinforcement learning algorithm is introduced to serve adaptive link load balancing in SDN-IoT networks.[10]

1.2 Contribution

A new model is proposed to balance the traffic in SDN-IoT networks. This method is supported with In-band Network Telemetry (INT) for collecting real-time traffic metrics. Trust aware routing adds strength to the method by improving the trust in the network paths against malicious nodes.[11] The proposed architecture uses a hybrid training inference model. The central SDN controller manages the training phase of PPO agent using global telemetry data collected from network switches. The inference is offloaded to SDN switches for ensuring low-latency path selection. The dynamically constructed state vector including metrics – link utilization, queue length, latency, packet loss and trust level helps in making optimal decision in routing. The PPO agent is aided with a multi-factor path cost function and adaptive reward function to select paths with optimized QoS and security aspects.

The key outcomes of this research are:

- A hybrid PPO-based architecture with global policy training using local inference reducing response time and control overhead.
- A real time state maintaining model with INT metrics and trust scores to enable fine-grained and adaptive routing.
- A reward function with dynamicity to manage the behavior of networks such as load balancing, low latency and security.
- Evaluation of the proposed research work in load balancing, improving QoS, and enhancing resilience under security attacks.

2 Methodology

The architectural diagram is depicted in the Figure 1. The entire architecture is organized in two distinct layers to optimize performance and scalability. The centralized training layer is placed at the controller level. It makes use of the SDN controller for aggregating telemetry data and training a Proximal Policy Optimization (PPO) based policy. Global network observations are done in this layer to lean an optimal forwarding strategy with focus on performance and trust. The Distributed Inference Layer functioning at the edge level provides decentralized decision making. The policy trained in the centralized layer is updated to the edge switches or agents in an interval of time. It also allows localized forwarding decisions. This hierarchical approach balances centralized intelligence with decentralized execution. It enhances both scalability and reducing latency across the network.

State Construction using INT: The system ensures real-time and accurate state management by using In-band Network Telemetry (INT) to collect flow level metrics from the data plane. The state vector s_t is constructed at time stamp t:

$$s_t = \{U_t, Q_t, L_t, PL_t, S_t\}$$

where U_t represents the link utilization, Q_t represents the queue length, L_t denotes the end to end latency, PL_t represents the packet loss rate, S_t denotes the trust score of the forwarding path. The multidimensional representation of the state provides the agent with a comprehensive view of the network by providing various performance metrics and security indicators in a single state vector.

Trust aware communication is an important aspect for reliable data transmission in the context of Software Defined Networking integrated with Internet of Things.[12] The trust worthiness of the communication path is calculated as the dynamic trust score S_t for each link or network device. The calculated S_t is used in action selection strategy of the PPO agent in selecting paths which do not meet the requirements of the expected performance but also minimize the risk of forwarding the data through malicious nodes. The trust score S_t is calculated using behavior based and security based metrics:

$$S_t = f(P_{drop}, R_{attack}, A_{valid}, D_{QoS})$$

Where, P_{drop} denotes historical packet drop behaviour, R_{attack} indicates anomaly detection score, A_{valid} signifies authentication status, D_{QoS} denotes deviation of expected Quality of Service parameters. These metrics are collected and monitored at the SDN switches. The collected metrics are periodically updated at the controller for appropriate decision making.[13]

Historical Packet Drop Behavior (P_{drop})

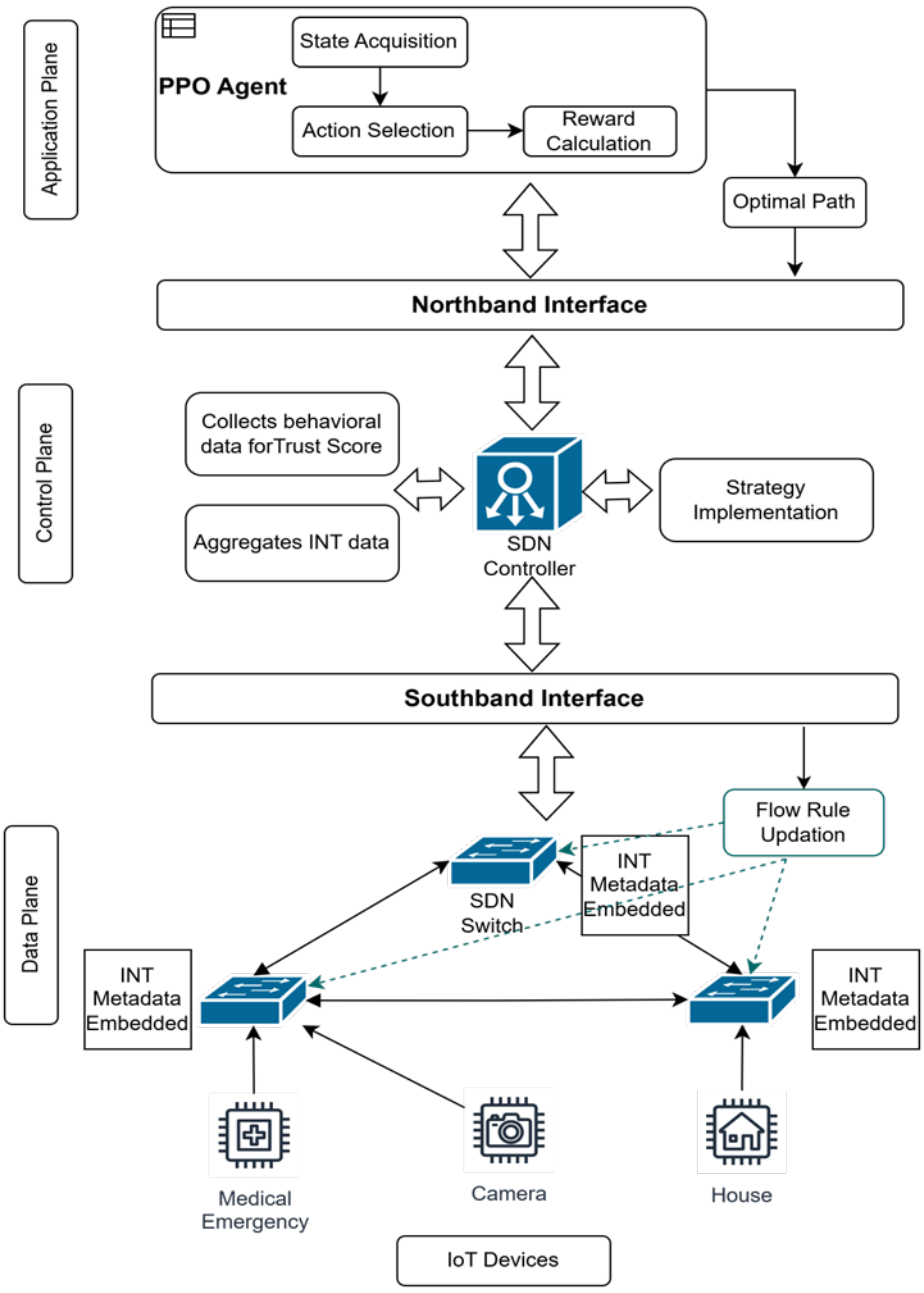

Fig 1. Hybrid PPO Architecture

It represents the reliability of the communication link calculated based on the packet delivery performance in the past. It is calculated as

$$P_{drop} = \frac{Pkts_{sent} - Pkts_{received}}{Pkts_{sent}}$$

The flow statistics maintained by SDN switches using OpenFlow's FlowStats feature and telemetry data received from INT enriches the granularity.[14]

Anomaly Detection Outcome *(R_{attack})*

A lightweight anomaly detection module contained in each switch track the traffic behavior in real time. Anomalies like traffic bursts, replay attacks, missing INT headers and malformed metadata are flagged. A risk score R_{attack} is calculated as each anomaly is identified and the penalty on the trust score is increased.

Token Verification Status *(A_{valid})*

The digital tokens are normally generated and attached by each IoT nodes with their private keys. Public key infrastructure is used for verifying these tokens by the switches. The status is maintained as a binary value.

A_{valid} = 1: The token is verified and also trusted

A_{valid} = 0: The token is invalid or not verified

This approach makes only the packets from authenticated nodes affect the routing decisions positively.

QoS Deviations (D_{QoS})

The metadata collected from INT contains the data of link utilization (U), queue length (Q), latency (L) and packet loss (PL). The baseline thresholds for expected performance are decided for each link.

The deviations are calculated as

$$D_{QoS} = \sum_{i=1}^{n} \omega_i \cdot \left| \frac{x_i - x_i^{baseline}}{x_i^{baseline}} \right|$$

Here, x_i is the observed metrics and ω_i are the respective weights. If there are high deviations, then it indicates the unreliability. High deviations will affect the trust score negatively.

Cost Function for Decision Making:

The weighted cost function is computed for making forwarding decisions of the PPO agent. It is calculated as

$$C = \alpha U_t + \beta Q_t + \gamma L_t + \delta PL_t + \eta S_t$$

Here α, β, γ, δ, and η are the scalar weights of the components: utilization (U), queue length (Q), latency (L) and packet loss (PL) and trust score (S_t). The cost function helps the agents to validate the forwarding paths with performance and security metrics. As the result, the agent will select the optimal path which has less delay and congestions.[15]

In the Proposed framework, finding optimal forwarding routes is a significant task. To learn the optimal routes, a Proximal Policy Optimization agent is designed. It is achieved through continuous interaction with the network environment. The agent's task is to select an action a_t based on the current state s_t by using a stochastic policy π_θ. It is defined as

$$a_t \sim \pi_\theta (a_t \mid s_t)$$

As the action is performed, then the agent is rewarded with r_t, which shows their level of performance and trust.[16] It is calculated as

$$r_t = w_1(1-d) + w_2(1-pl) + w_3(1-lv) + w_4 \cdot sec + w_5 \cdot S_t$$

Where, *d* stands for normalized end –to-end delay, *lv* indicates link variance, sec denotes security indication, S_t indicates the trust score of the selected forwarding path, w_1, w_2, w_3, w_4, w_5 are the weights for network policies, *Pl* counts the packet loss rate. The reward function makes sure that the agent behaves in way that competing objective of minimizing delay and loss, maximizing security and trust worthiness are achieved.

To keep sure that the policies are updated steadily, the PPO algorithm also uses a clipped objective with regard to policies. It is calculated as

$$L_t^{CLIP}(\theta) = min\left(r_t(\theta) \cdot \widehat{A}_t, \tilde{r}_t(\theta)\widehat{A}_t\right)$$

Here, $r_t(\theta)$ is used to compare the probabilities of action under the new and the old policy, $\tilde{r}_t(\theta)$ is the clipped ratio. It ensures $r_t(\theta)$ stays within $[1-\epsilon, 1+\epsilon]$ by using a lower bound and upper bound. $\widehat{A}_t$ shows the value related with performing a_t in s_t, ϵ represents the limit on policy updates.

The advantage estimate $\widehat{A}_t$ is calculated as:

$$\widehat{A}_t = r_t + \gamma V_{\theta_v}(s_{t+1}) - V_{\theta_v}(s_t)$$

This aids in updating the policy by comparing expected future rewards. The following loss function is used for accurate value prediction.

$$L_V = \left(V_{\theta_v}(s_t) - \left(r_t + \gamma V_{\theta_v}(s_{t+1})\right)\right)^2$$

This formulation is done for ensuring the agent does reliable value approximation. The clipped policy updates along with accurate value learning provide robust training even in dynamic network.[17]

This centrally trained policy is distributed to edge switches at a predefined interval denoted as T_{sync}. As the edge switches receive the policy, they use it for decentralized decision making. The action a'_t is selected based on the local state s'_t.

$$a'_t = arg\max_a \pi_\theta(a \mid s'_t)$$

a'_t represents the selected action by the switch, $\arg\max_a$ indicates the action, maximizing the policy function π_θ . This ensures the switch selects most likely best action. Moving towards the distributed approach reduces the dependency on the central controller. This approach also enhances the scalability and reduces latency. To manage the dynamic and unpredictable nature of IoT devices, adaptive tuning is opted for the reward function. The reward function concentrates on performance related metrics when operating in normal conditions. It emphasis trust and security metrics when the network faces security attacks. This nature of adaptive reward calculation makes the agent to maintain optimal decision making even under the uneven loads and adversarial scenarios.[18]

Algorithm: Hybrid PPO with INT

Input:

π_θ, V_{θ_v} - Policy and value networks
ϵ - Clipping factor, α - Learning rate
γ - Discount factor, T_{sync}— Sync interval
K - Number of training epochs
$\alpha, \beta, \gamma, \delta, \eta$ - Path cost weights (for U, Q, L, PL, S)
w^1, w^2, w^3, w^4, w^5 - Reward weights
Initialization Process:
Initialize π_θ, V_{θ_v}, $\pi_{\theta\_old} \leftarrow \pi_\theta$
Initialize empty experience buffer
Start the training Loop:
While training not converged do
 For each SDN switch do
  Collect INT telemetry metrics: (U_t, Q_t, $L_{t,}$, PL_t)
  Check the packet's signature and confirm the node is secure
  Evaluate trust score S_t
  s_t = {U_t, Q_t, $L_{t,}$,PL_t, S_t }
  Calculate the weighted cost function:
   $C = (\alpha.U_t + \beta.Q_t + \gamma.L_t + \delta.PL_t + \eta.S_t)$
  Find the sample action: $a_t \sim \pi_\theta(a_t \mid s_t)$
  Implement the action a_t through SDN forwarding rule
  Observe next state $s_{\{t+1\}}$
  Compute reward:
   $r_t = w_1.(1-d) + w_2.(1-pl) + w_3.(1-lv) + w_4 \cdot sec + w_5 \cdot S_t$
  Store transition: buffer.append((s_t, a_t, r_t, $s_{\{t+1\}}$))
 End for
 For each transition (s_t, a_t, r_t, $s_{\{t+1\}}$) in buffer do

Estimate advantage:
$\widehat{A}_t = r_t + \gamma V_{\theta_v}(s_{t+1}) - V_{\theta_v}(s_t)$
End for
For epoch = 1 to K do
For each transition do
$r_t(\theta) \leftarrow \pi_\theta(a_t \mid s_t) / \pi_{\theta_{Old}}(a_t \mid s_t)$
$\tilde{r}_t(\theta) = max(1-\epsilon, min(r_t(\theta), 1+\epsilon)$
$L_t^{CLIP}(\theta) = min\left(r_t(\theta) \cdot \widehat{A}_t, \tilde{r}_t(\theta)\widehat{A}_t\right)$
Update π_θ to maximize L_CLIP
$L_V = \left(V_{\theta_v}(s_t) - \left(r_t + \gamma V_{\theta_v}(s_{t+1})\right)\right)^2$
Update V_{θ_v} to minimize L_V
End for
End for
Release buffer
If $current_{step}$ % T_{sync} == 0 then
$\pi_{\theta\_Old} \leftarrow \pi_\theta$
Sync updated π_θ to edge switches
End if
For each edge switch do
Observe local state s′
Check for new policy
If new policy received, update π_θ
Select action: a′ ← argmax$_a \pi_\theta$(a | s′)
Apply action using SDN rule
End for
Observe QoS metrics: (delay, packet loss, variance, trust)
Adjust reward weights w_1 to w_5 (if needed)
End while

The variables used in the algorithm are described in the Table 1.

The proposed algorithm Hybrid PPO with INT integrates Proximal Policy Optimization with In-band Network Telemety (INT). The algorithm begins with the initialization of policy network, value network and storing the current policy as the old policy. The telemetry metrics are collected, and the trust score is calculated for each link in the training phase. The collected values are used to form the state vector. The cost function is formulated with weighted coefficients from the state vector. The agent selects an action using the state values and it is applied through SDN. Upon observing the new state, the reward for that state is calculated using delay, loss, variance, security and trust. Advantage estimates for each transaction is calculated. Then K epochs of PPO is performed with clipped loss. The updated policy is updated to all the switches.

Figure 2 shows the swimlane diagram of Hybrid PPO. It depicts that IoT devices send data to a central controller with necessary tags. The network conditions and security aspects are evaluated by the controller. It uses PPO algorithm to select the best path. The routing is done using PPO and continues learning occurs. This optimizes the performance in IoT network.

2.1 Experimental Setup

The proposed Hybrid PPO was evaluated through network emulation using Mininet[19]. The experimental set up is done with 10 P4 programmable BMv2 switches[20] and 20 IoT host nodes to create a realistic SDN-IoT. All switches are implements with a P4 pipeline integrated with In-band Network Telemetry (INT) metadata. The metrics including link utilization (U), queue length (Q), latency (L), and packet loss (PL) fed into the packet headers at line rate. The telemetry data was collected for every 5 packets using P4's add_header primitive.[21] The Ryu SDN controller was used as the centralized controller. It was equipped with a Python based INT parser, which was used to extract telemetry metrics from packet in message. The collected metrics were normalized and combined with dynamic trust score. For the trust score calculation, a lightweight security module was used at the switches. The key metrics: historical packet drop rates (P_{drop}) from OpenFlow FlowStats, anomaly detection scores (R_{attack}) from a TensorFlow Lite LSTM model, token verification status (A_{valid}) via ECDSA signatures[22], and QoS deviation (D_{QoS}) from baseline performance thresholds are collected for trust score calculation.

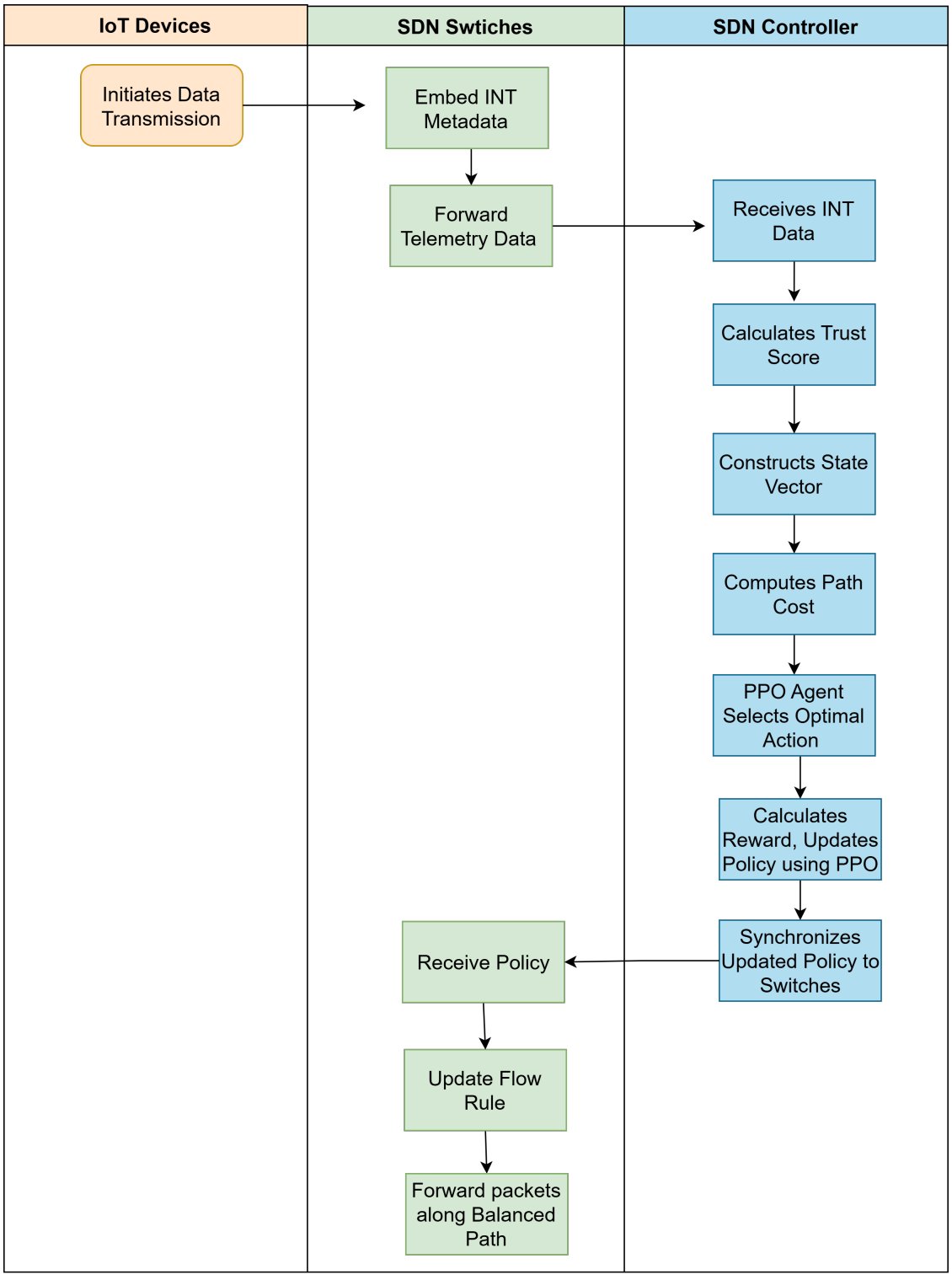

Fig 2. Swimlane Diagram of Hybrid PPO

Table 1. Key Variables of Hybrid PPO with INT

Variable	Meaning / Description
π_θ	Policy network applied currently
V_{θ_v}	Estimated cumulative reward for each state for Value Network
π_{θ_old}	Previous version of policy network used for PPO ratio calculation
ϵ	Clipping factor to maintain update stability
α	Learning rate of policy and value networks
γ	Advantage estimation discount factor for future rewards
K	Number of training epochs that occurs for each policy cycle
T_{sync}	Policy synchronization interval with SDN switches
U_t	Link utilization at time step t
Q_t	Queue length value at time step t
L_t	End-to-end link latency at time step t
PL_t	Packet loss ratio at time step t
S_t	The path Trust score calculated at time step t
s_t	State vector: $s_t = \{U_t, Q_t, L_t, PL_t, S_t\}$
a_t	Action selected at time step t
r_t	Immediate reward for a_t
$\widehat{A}_t$	Estimated advantage used for policy improvement
r_θ	Importance sampling ratio used in PPO update step
L^{CLIP}	Clipped loss function for updating the policy network
L_V	Value loss for updating the value network
C	Path cost
d	Measured delay
pl	Measured packet loss
lv	Link variance
sec	Security status
w_1, w_2, w_3, w_4, w_5	Weights assigned in reward function
$\alpha, \beta, \gamma, \delta, \eta$	Scalar weights assigned in the path cost

PyTorch 1.12 was used to implement the Proximal Policy Optimization (PPO) agent with network state space having normalized INT metrics and trust score. The action space consisted of discrete next hop selections. It was guided by a reward function with the objectives of low delay (d), minimal packet loss (pl), balanced link utilization (lv), path security (sec), and trustworthiness (S_t). The agent employed generalized advantage estimation with $\gamma = 0.99$, $\lambda = 0.95$. For stable policy updates ϵ is assigned the value of 0.2. The security ability was checked by simulating various attack scenarios including DDoS floods through hping3 and INT spoofing attacks. The performance evaluation was done by comparing Hybrid PPO with Equal-Cost Multi-Path (ECMP), Deep Q-Network (DQN), Advantage Actor-Critic (A3C) and Trust inspired routing adapted in SDN (Trust AODV) baselines. The validation of Hybrid PPO was done to simultaneously optimize network performance and security in dynamic IoT environments. The collected logs were used to plot graphs using Python scripts.

3 Results and Discussion

The Hybrid PPO method was validated to the effectiveness of the algorithm under normal and adversarial network conditions. Various key metrics like end to end delay, flow completion rate, attack detection accuracy and were evaluated. These metrics were compared with the baseline algorithm – ECMP, DQN, A3C and Trust inspired routing adapted in SDN (Trust AODV) to find the improvement in routing intelligence, trust awareness and security concerns.

Figure 3 illustrates the 99th percentile latency comparison of five routing algorithms: ECMP, DQN, A3C, Trust-AODV and the proposed Hybrid PPO. This was evaluated with varying IoT device counts from 50 to 200. All the algorithms tend to have a rise in latency when the number of devices increase. But there is a difference in the rate of increase and the overall latency level. ECMP and Trust-AODV have highest latency. It indicates the inefficiency of handling network congestion efficiently.

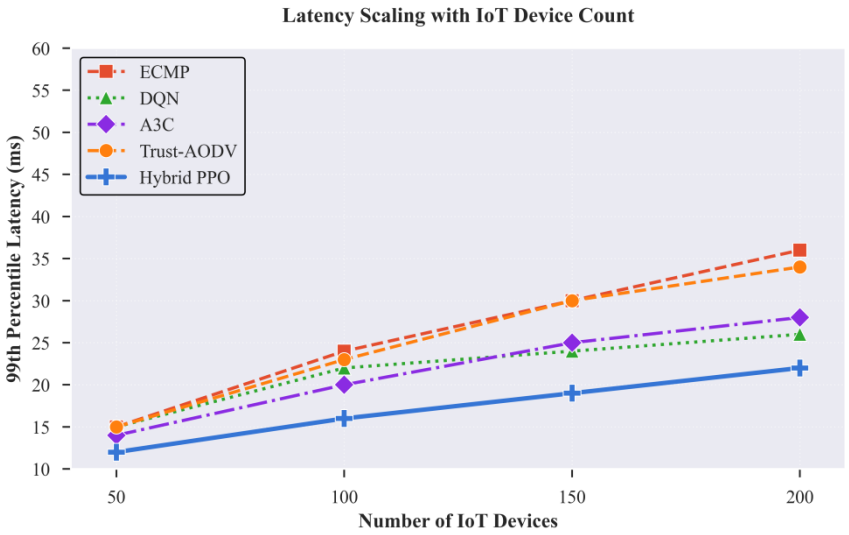

Fig 3. 99th Percentile Latency Performance

DQN and A3C tend to show moderate performance because of reinforcement learning strategies. While comparing with other algorithms, the proposed Hybrid PPO consistently achieves lower latency with regard to all device counts. It explicitly shows that the Hybrid PPO is superior in scalability and efficiency in managing communication delays in dense IoT networks.

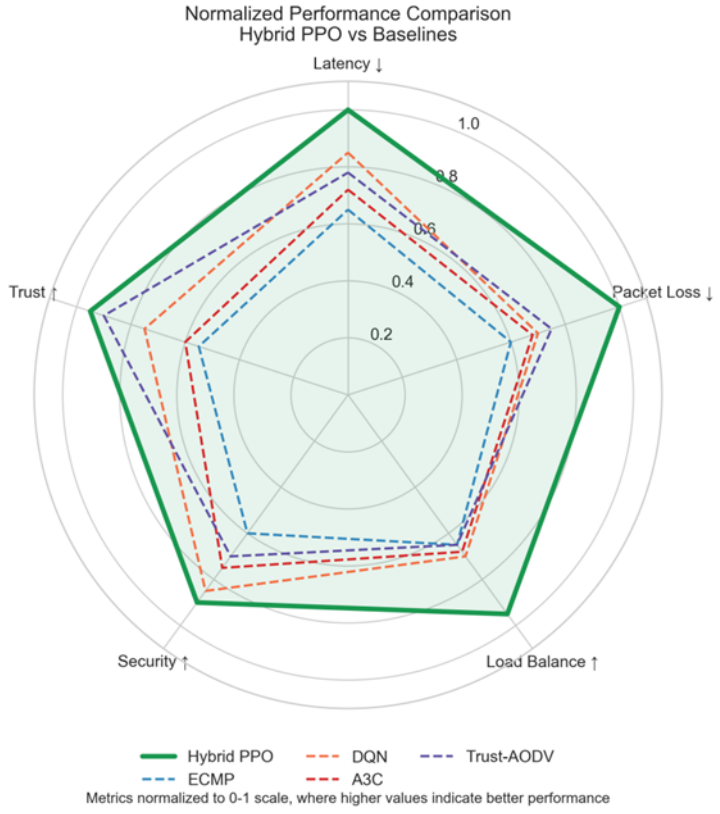

Fig 4. Normalized Performance Comparison Chart

The normalized radar char is presented in the Figure 4. It shows the comparison of Hybrid PPO with baseline algorithms across five key performance metrics. They are latency, Packet loss, load balance, security and trust. All the metrics are normalized in a 0 – 1 scale. The higher value denotes better performance. For latency and packet loss the normalization is done in inverse proportion. It means, the lower values yield higher normalized scores. The chart vividly depicts the proposed Hybrid PPO performs consistently better in all dimensions, closely reaching the maximum value of 1.0 in most cases. The high scores in latency, packet loss and load balance says well about its efficiency in minimizing delay, reliable data delivery and distributing traffic effectively. It also shows better performance in security and trust compared to conventional baseline algorithms.

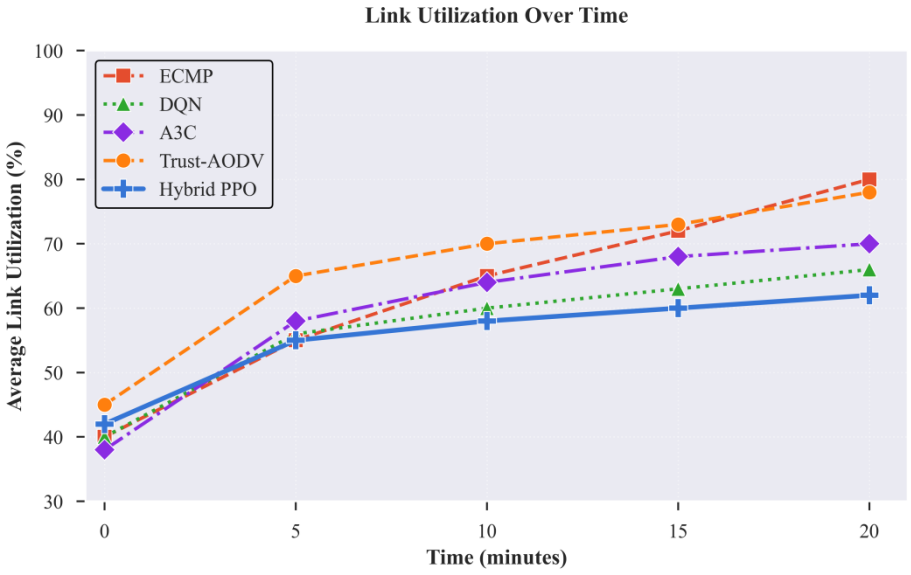

Fig 5. Link Utilization Analysis

Figure 5 demonstrates the link utilization analysis. The hybrid PPO depicts a balanced routing behavior with steady and moderate link usage over time. Trust-AODV and ECMP tend to show more aggressive strategies. But Hybrid PPO maintains utilization around 60 – 62%. It shows its ability to intelligently manage traffic without overloading links. It indicates the superior latency performance of hybrid PPO ensuring effective throughput in scalable and time sensitive IoT networks

Figure 6 shows the threat response latency evaluation and comparison with baseline algorithms. It shows the increase in latency across all the algorithms as attack intensity increases. But the hybrid PPO outperforms all others. It maintains lowest response latency in all attach intensity. When the attack intensity reaches 90%, Hybrid PPO achieves a latency of 75 ms approximately, while A3C reaches 150 ms, DQN reaches 170 ms and Trust AODV reaches 220 ms. This indicates that the Hybrid PPO maintains faster and robust threat response abilities specially in high stress network conditions.

The Figure 7 illustrates the policy stability of the proposed Hybrid PPO. The training iterations show the decline in the standard deviation of action selection probabilities. It means that the policy becomes increasingly stable. Though the algorithm initially explores with high variability, the uncertainty decreases steadily as learning increases. The convergence rate below the defined threshold rate of 0,05 over 7500 indicates that the PPO agent has effectively learned the routing decisions. The $\pm$10% shaded region around the curve confirms the low variance across training runs. This depicts the robustness and reliability of the proposed method in dynamic network conditions.

Figure 8 shows three subplots the analysis of the proposed Hybrid PPO with regard to its convergence efficiency, generalization capability and training stability. Subplot (a) demonstrates that Hybrid PPO has the higher convergence rate in average episode rewards compared to base line algorithms. It exhibits its superior learning efficiency. Subplot (b) shows the over-fitting detection using training and validation curves. The graph explicitly shows that the validation performance drops after certain episodes. This marks the onset of over-fitting zone which helps in identifying optimal stopping point for training. Subplot (c) explains the robustness of Hybrid PPO using stable learning progress with minimal variance. Thus, the graph collectively says that the proposed algorithm has strong convergence behavior, resistance to over-fitting and consistent learning dynamics in complex SDN-IoT environments.

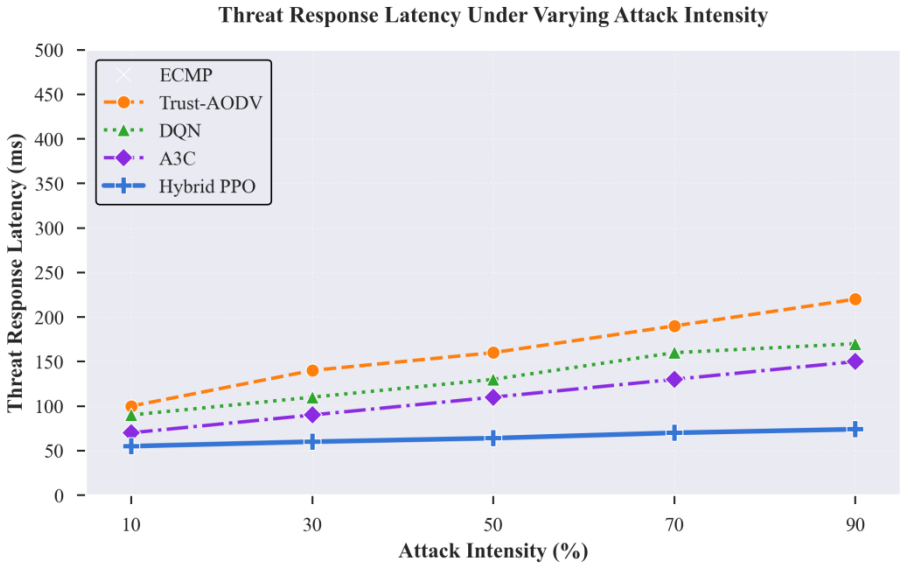

Fig 6. Comparative Performance of Threat Response Latency

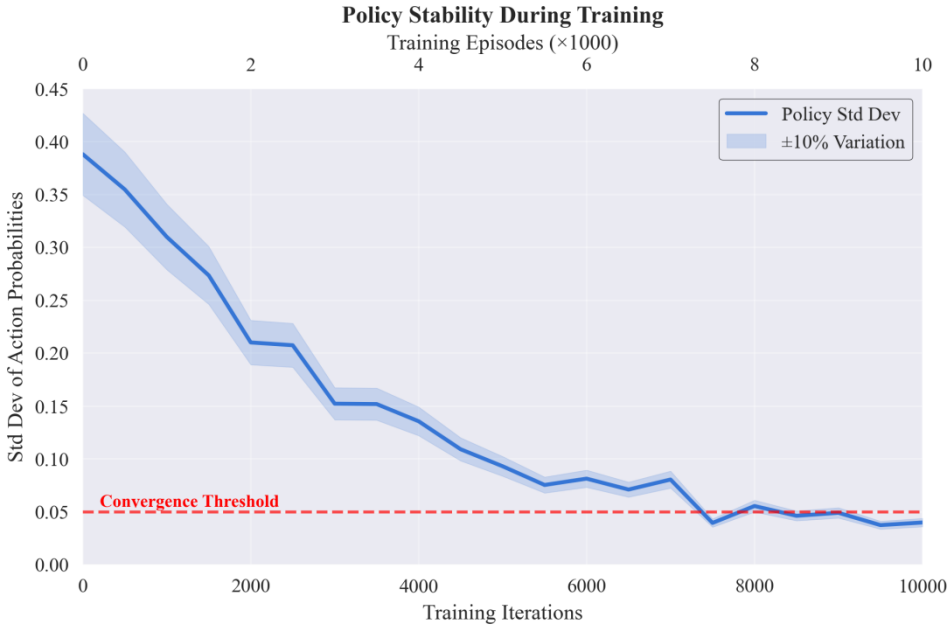

Fig 7. Policy Stability of PPO

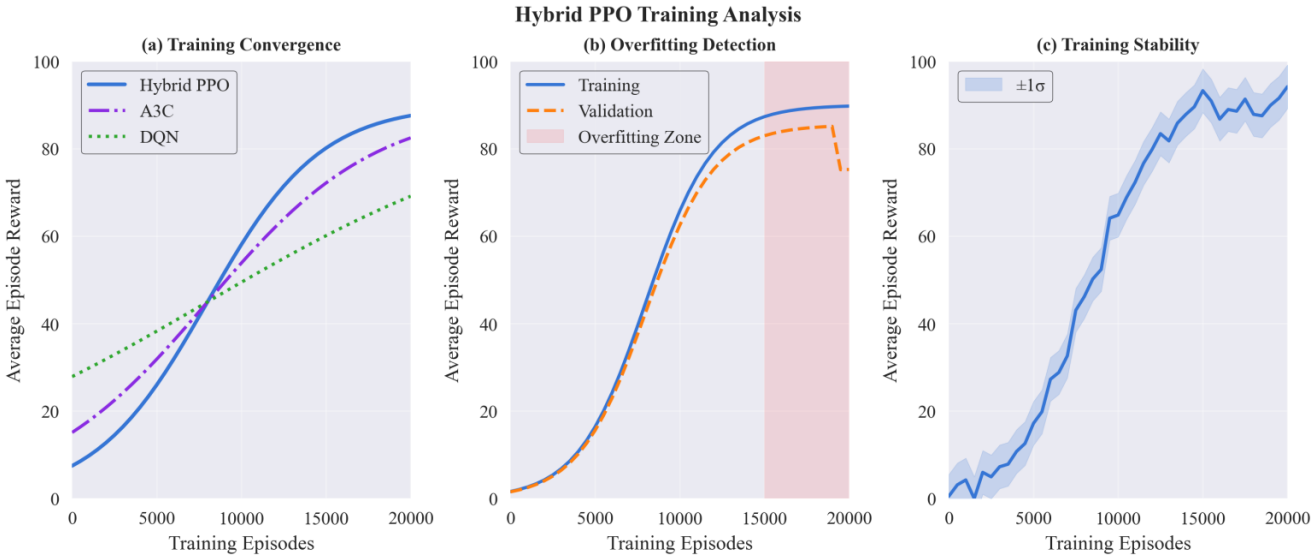

Fig 8. Hybrid PPO Training Analysis

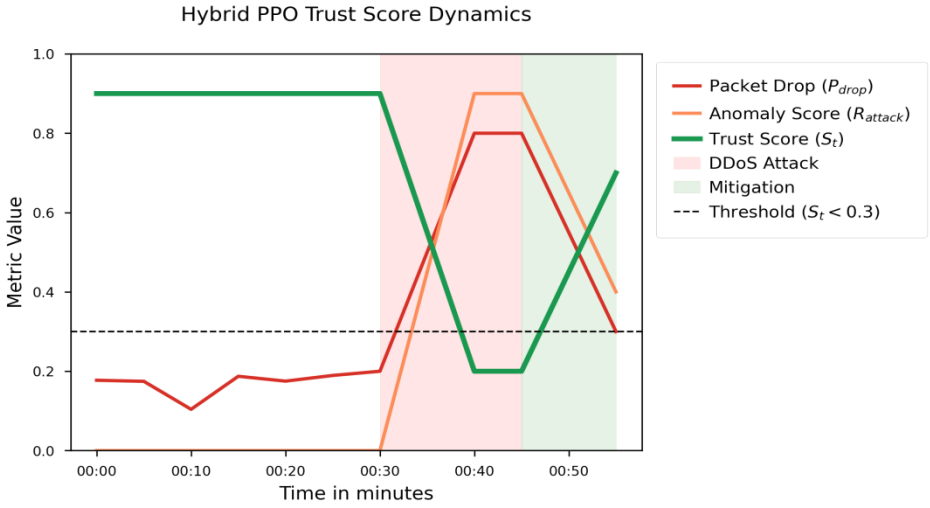

Fig 9. Hybrid PPO Trust Score Dynamics

The Figure 9 illustrates the trust score dynamics of proposed Hybrid PPO. This is evaluated under a simulated DDoS attack scenario. The metrics analyzed are packet drop rate (P_{drop}), anomaly score (R_{attack}), and trust score (S_t). The trust score tends to be high during the normal operation. P_{drop} and R_{attack} are minimal. When the DDoS attack is introduced (highlighted in red) significant rise is found in P_{drop} and R_{attack} and drop in S_t. This triggers the PPO based mitigation (green shaded region). After the mitigation, the trust score is recovered. This indicates the proposed method's adaptive response and resilience against attacks.

The Figure 10 shows the attack detection accuracy across various algorithms. Four types of attacks namely, DDoS, INT spoofing, trust violation, replay attack. In all attacks, Hybrid PPO consistently outperforms traditional baseline algorithms. Remarkably, it achieves 95% F1 score for DDoS, 92% INT spoofing, 89% Trust violation and 91% Replay attack. Other algorithms display a significant drop in the accuracy of detection, mainly in trust violation and replay attack. These results confirm the robustness of the Hybrid PPO in quickly identifying the attacks and enabling the intelligent defense strategies,

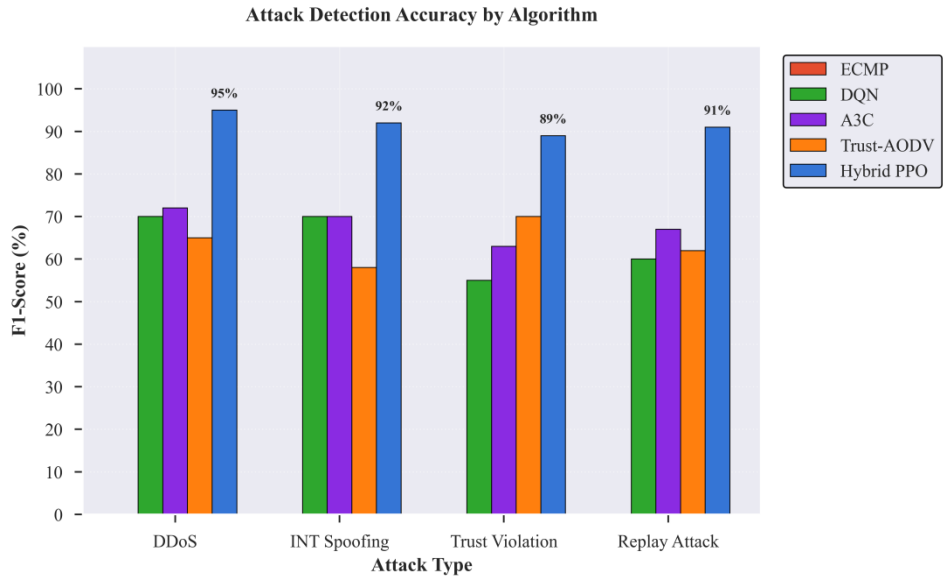

Fig 10. Accuracy Levels of Attack Detection

3.1 Comparative Analysis

The Proposed Hybrid method exhibits comprehensive improvement across key areas such as latency, throughput, security, network performance and convergence, when compared to the latest state of the art methods. The proposed Hybrid PPO demonstrates superior latency performance with sub-100ms 99th percentile latency and it outperforms GA based methods [4] having 128ms latency. Especially in adversarial conditions, Hybrid PPO achieves 75ms threat response latency, while MFGAD-INT [6] has 5-10ms of telemetry processing latency. The proposed method performs with the packet loss below 2% with the normalized score of 0.95. The RL+SVM method [5] has the packet loss between 5-8%. In terms of network efficiency, the Hybrid PPO achieved link utilization between 60 to 62 % which is more efficient while comparing DRL method [8] having 68% of peak utilization.

Reinforcement learning methods proposed by Chen et al. [3] and Jisi et al. [5] showed improvement in QoS using intelligent agents. But they lacked trust awareness and defense against threats. In contrast, the Hybrid PPO approach combines trust aware routing with F1 score of up to 95% in various attacks including DDoS, INT spoofing, replay and trust violations. The experiments show that Hybrid PPO attains stable policy convergence within 7500 training iterations with the variance of $\pm$10% across runs. Chen et al.'s DDPG approach [3] reaches the convergence with 10,000 iterations and Xinglong Pei et al.'s DRL method [8] reaches the convergence with 12,000+ iterations. Thus, Hybrid PPO performs better in terms of policy convergence while comparing the existing methods.

4 Conclusion

This research proposed and evaluated a new Hybrid PPO based adaptive link load balancing with In-bank Network Telemetry (INT). This enhances the routing intelligence and security in SDN IoT networks. Various evaluations conducted under normal and adversarial conditions. In all situations, the proposed method demonstrated significant performance improvements compared to baseline algorithms. Key findings of the research confirm that the Hybrid PPO method has reaches high end to end latency reduction with Sub-100ms latency, lower packet loss of < 2% and improved load balancing with link utilization between 60 to 62%. Efficient traffic distribution and congestion mitigation provides a smoother scalability. This is even achieved when increasing device counts by the adaptive routing intelligence of the proposed method. This proposed method also shows strong attack detection capacities with F1-scores up to 95% against several attack types including DDoS, INT spoofing, Trust violations and replay attacks. This ensures its robustness in dynamic threat scenarios. It also demonstrates improved performance with stable policy convergence ($\sigma < 0.05$) within 7500 training iterations.

The key innovations of the proposed are trust driven decision making, faster convergence, efficient link utilization and resilience under attach. The proposed model makes decision not only using network performance but also using how

trustworthy each node is. This helps in identifying paths that are malicious. It also learns faster, taking less time to find effective solutions. It also balances the load by distributing the traffic without overloading specific links. Importantly, the model performs well even when the network in under the attack. However, there are limitations of computational overhead and dependency on exact trust score estimation. The proposed model could be improved with incorporating anomaly detection to refine trust score accuracy and applying policy distillation for reducing PPO model size.

The promising prospects of this model would be 5G network slicing where different service types need strict QoS adherence. It can defend potential threats and dynamically allocate slices in the control and user plane. Hybrid PPO creates the stage for self-healing SDN environment with policy adaptation to threats, congestion and trust violations in real time. Hybrid PPO also stands suitable for smart manufacturing, energy systems and edge assisted healthcare because of its attack resilience as these areas could not tolerate system down time and security breaches.

As the proposed algorithm showed promising performance in link utilization and load balancing, there are several avenues for future enhancement. One important direction for future work is multi objective optimization, where reducing delay, lowering energy consumption, improving reliability and ensuring security can be simultaneously achieved. Another promising approach is transfer learning. This enables faster learning learning process in dynamic or unseen networks. This will also result in reducing convergence time and enhancing responsiveness. Finally, exploring hierarchical PPO architecture for distributed SDN control and cross layer optimization is another aspect where different levels of the system make decisions in a coordinated way.

5 Acknowledgement

The authors thank, DST-FIST, Government of India for funding towards infrastructure facilities at St. Joseph's College (Autonomous), Tiruchirappalli – 620 002.

References

1) Rostami M, Goli-Bidgoli S. An overview of QoS-aware load balancing techniques in SDN-based IoT networks. *Journal of Cloud Computing*. 2024;13:89. https://doi.org/10.1186/s13677-024-00651-7.
2) Babangida I, Bakar KA, Yusuf NM, Abaker M, Abdelmaboud A, Nagmeldin W. Software defined wireless sensor load balancing routing for Internet of Things applications: Review of approaches. *Heliyon*. 2024;10(9):e29965. https://doi.org/10.1016/j.heliyon.2024.e29965.
3) Chen J, Wang Y, Ou J, Fan C, Lu X, Liao C, et al. ALBRL: Automatic Load-Balancing Architecture Based on Reinforcement Learning in Software-Defined Networking. *Wireless Communications and Mobile Computing*. 2022;p. 3866143. https://doi.org/10.1155/2022/3866143.
4) Bhowmik CD, Gayen T. Traffic aware dynamic load distribution in the Data Plane of SDN using Genetic Algorithm: A case study on NSF network. *Pervasive and Mobile Computing*. 2023;88:101723. https://doi.org/10.1016/j.pmcj.2022.101723.
5) Jisi C, Roh B, Ali J. Reliable paths prediction with intelligent data plane monitoring enabled reinforcement learning in SD-IoT. *Journal of King Saud University - Computer and Information Sciences*. 2024;36(3):102006. https://doi.org/10.1016/j.jksuci.2024.102006.
6) Duan Y, Li C, Bai G, et al. MFGAD-INT: In-band network telemetry data-driven anomaly detection using multi-feature fusion graph deep learning. *Journal of Cloud Computing*. 2023;12:126. https://doi.org/10.1186/s13677-023-00492-w.
7) Zhang P, Zhang H, Pi Y, Cao Z, Wang J, Liao J. AdapINT: A Flexible and Adaptive In-Band Network Telemetry System Based on Deep Reinforcement Learning. *IEEE Transactions on Network and Service Management*. 2024;21(5):5505–5520. https://doi.org/10.1109/TNSM.2024.3427403.
8) Pei X, Sun P, Hu Y, Li D, Chen B, Tian L. Enabling efficient routing for traffic engineering in SDN with Deep Reinforcement Learning. *Computer Networks*. 2024;241:110220. https://doi.org/10.1016/j.comnet.2024.110220.
9) Kumar S, Malik A. EFLB-IIoT: Enhanced flow control and load balancing approach for SDN-enabled Industrial Internet of Things. *International Journal of Communication Systems*. 2025;38(3):e70043. https://doi.org/10.1002/dac.70043.
10) Zhang Y, Qiu L, Xu Y, Wang X, Wang S, Paul A, et al. Multi-Path Routing Algorithm Based on Deep Reinforcement Learning for SDN. *Applied Sciences*. 2023;13(22):12520. https://doi.org/10.3390/app132212520.
11) Bhamare D, Kassler A, Vestin J, Khoshkholghi MA, Taheri J, Mahmoodi T, et al. IntOpt: In-band Network Telemetry optimization framework to monitor network slices using P4. *Computer Networks*. 2022;216:109214. https://doi.org/10.1016/j.comnet.2022.109214.
12) Alzubaidy HSR, Jabber H. A Survey of Software-Defined Networking (SDN) Controllers for Internet of Things (IoT) Applications. *Babylonian Journal of Networking*. 2023;p. 15–20. https://doi.org/10.58496/BJN/2023/003.
13) Wang J, Wang L. SDN-Defend: A Lightweight Online Attack Detection and Mitigation System for DDoS Attacks in SDN. *Sensors*. 2022;22(21):8287. https://doi.org/10.3390/s22218287.
14) Liatifis A, Sarigiannidis P, Argyriou V, Lagkas T. Advancing SDN from OpenFlow to P4: A Survey. *ACM Computing Surveys*. 2023;55(9):1–37. https://doi.org/10.1145/3556973.
15) Sanchez LPA, Shen Y, Guo M. DQS: A QoS-driven routing optimization approach in SDN using deep reinforcement learning. *Journal of Parallel and Distributed Computing*. 2024;188:104851. https://doi.org/10.1016/j.jpdc.2024.104851.
16) Chen J, Huang X, Wang Y, Zhang H, Liao C, Xie X, et al. ASTPPO: A proximal policy optimization algorithm based on the attention mechanism and spatio–temporal correlation for routing optimization in software-defined networking. *Peer-to-Peer Networking and Applications*. 2023;16:2039–2057. https://doi.org/10.1007/s12083-023-01489-7.
17) Meng W, Zheng Q, Pan G, Yin Y. Off-Policy Proximal Policy Optimization. vol. 37. 2023;p. 9162–9170. https://doi.org/10.1609/aaai.v37i8.26099.
18) Teslim B. Using Reinforcement Learning for Adaptive Security Protocols. *International Journal of Computer Engineering and Technology (IJCET)*. 2024;15(6):15–20. Available from: https://www.researchgate.net/publication/384608149_USING_REINFORCEMENT_LEARNING_FOR_

ADAPTIVE_SECURITY_PROTOCOLS.
19) Karthika P, Karmel A. Simulation of SDN in mininet and detection of DDoS attack using machine learning. *Bulletin of Electrical Engineering and Informatics*. 2023;12(3):1797–1805. https://doi.org/10.11591/eei.v12i3.4294.
20) Hauser F, Häberle M, Merling D, Lindner S, Gurevich V, Zeiger F, et al. A survey on data plane programming with P4: Fundamentals, advances, and applied research. *Journal of Network and Computer Applications*. 2023;212:103561. https://doi.org/10.1016/j.jnca.2022.103561.
21) Simsek G, Ergenç D, Onur E. Reliable and Distributed Network Monitoring via In-band Network Telemetry. *arXiv preprint*. 2022;p. 2212.14876. Available from: https://arxiv.org/abs/2212.14876.
22) Gong S. Recent Progress in ECDSA and DSA Digital Signature Algorithms. *Transactions on Computer Science and Intelligent Systems Research*. 2024;6:22–29. https://doi.org/10.62051/at4q6n03.