JSCDM

JOURNAL OF SOFT COMPUTING AND DATA MINING
e-ISSN: 2716-621X

Vol. 7 No. 2 (2026) 144-162
https://publisher.uthm.edu.my/ojs/index.php/jscdm

Secure Resource Allocation in IoT Environments Using Deep Reinforcement Learning-Based Adaptive Defenses

Amar A. Othman[1], Sedeeq Al-khazraji[2], Mahmood Siddeeq Qadir[3], Omar Almomani[4]*

[1] *College Of Fine Arts, University of Mosul. Mosul, IRAQ*
[2] *College of Computer Science and Mathematics, University of Mosul, Mosul, IRAQ*
[3] *Continuing Education Center, University of Mosul, Mosul, IRAQ*
[4] *Department of Networks and Cybersecurity, Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman, JORDAN*

*Corresponding Author: o.almomani@ammanu.edu.jo
DOI: https://doi.org/10.30880/jscdm.2026.07.02.010

Article Info

Received: 27 March 2026
Accepted: 4 June 2026
Available online: 30 June 2026

Keywords

Internet of Things (IoT), Deep Reinforcement Learning (DRL), Dueling DQN, A3C, adaptive defense, secure resource allocation, edge computing, trust management, latency optimization, energy efficiency, cybersecurity, policy convergence, multi-objective reward

Abstract

The success of Internet of Things (IoT) networks has also created a major problem in resource provisioning, as these networks are becoming increasingly heterogeneous, dynamic, and distributed. Simple and heuristic-based allocation schemes have become obsolete in terms of fulfilling the collective goals of latency reduction, energy efficiency, and secure defense against malicious attacks. To overcome such limitations, this paper suggests a new Deep Reinforcement Learning (DRL) adaptive defense architecture designed to provide resources efficiently and securely in hostile IoT environments. The suggested system combines three major components, viz., Trust Monitoring Module (TMM), Resource Profiling Agent (RPA), and Network Security Orchestrator (NSO), each of which works within the edge and fog layers in cooperation with each other. These modules dynamically evaluate the trustworthiness of devices, keep track of resources, and coordinate real-time policy enforcement in situations of changing threats. Two enhanced DRL models, namely Dueling Deep Q-Network (Dueling DQN) and Asynchronous Advantage Actor-Critic (A3C), are tested in a simulated environment in which adversarial situations involving Distributed Denial of Service (DDoS), flooding, and spoofing attacks are replicated. Quantitative findings: the Dueling DQN comes out with better scores when it comes to important performance parameters: it delivers 89.7 percent packets, has an average latency of 92.3 ms, an energy efficiency of 362 mWh, and a convergence score of 0.88 on the trust score when compared to 83.5 percent packets, 108.2 ms average latency, 295 mWh energy efficiency, and 0.82 convergent score on the trust score of A3C. Dueling DQN further indicates 26.7 percent enhanced policy convergence, 22.7 percent greater energy efficiency, and 18.3 percent superior reward slope stability. These results speculate on the much higher versatility, trust affiliation, and appropriateness of the framework in accommodating the ensuing real-time, mission-centered endeavors of IoT implementation.

1. Introduction

The widespread use of the Internet of Things (IoT) across myriad applications, such as smart cities, industrial automation, precision healthcare, and smart energy networks, has resulted in an increasingly interconnected cyber-physical system. Each of these network infrastructures is continuously generating and exchanging a lot of data, and these demands need to be controlled intelligently so that they can ensure secure, prompt, and energy-efficient operation. However, the nature of the IoT systems is limited in tools (specifically, it can have battery-backed or low-calculation items that can use distributed, heterogeneous, and even vulnerable physical environments). All that makes IoT performances susceptible to various cyber-physical attacks and poor utilization of available resources [1].

The traditional resource allocation mechanism that is based on a static or deterministic threshold is becoming ineffective. They will not be able to track the dynamics of the existing IoT network where devices are made available, their energy levels, their bandwidth, and their exposure to security breaches are changing very fast and cannot be predicted. Moreover, the traditional methods lack context awareness and are unable to respond to the unpredictable shifts that occur in processes in the sphere of an operation, which is characteristic of mission-critical applications [2]. In this regard, therefore, an urge has risen to necessitate intelligent methods of resource allocations which would be able to support real-time adjustments to dynamic conditions, retain the integrity of their working, and secure themselves against internal and external assaults on security.

The future adaptive measures, in particular, those which rely on artificial intelligence (AI), show some promise. The possibility to dynamically analyze complex situations and respond adequately to them in order to achieve the optimal results of the system is based on intelligent control processes [3]. It is, however, in this that Deep Reinforcement Learning (DRL) has been quickly popularized due to the benefit of continuously interacting with the environment to receive and learn the policies of the environment. Unlike in supervised learning in DRL, an agent does not require labeled data to improve a policy; rather, it uses signals of reward on the actions it takes and the consequences of the aforementioned actions in terms of states. By this property, DRL is able to process realistic high dimensions of state and action space that are common to the IoT environment [4].

The fact that DRL supports optimization of multiple objectives is another point that makes it acceptable in the field of IoT. A properly built reward mechanism has the potential of enforcing aspects such as energy savings, latency reduction, and delivery of security simultaneously. Adaptivity is particularly important to distributed settings on the edge, where devices must respond locally and in the best interest of the globally distributed goals. In addition, the current successes in the DRL structures, namely the Asynchronous Advantage Actor-Critic (A3C) and Dueling Deep Q-Networks (Dueling DQNs), have been time-sensitive and can scale to decentralized issues [5].

Although the possibilities of DRL are rather large, they have been little examined when applied to safe and flexible charging of resources in IoT frameworks. Most of the current initiatives are enacted towards independent goals like throughput optimization or energy saving, without the combination of security consciousness and trust modeling. On the contrary, an overall solution should combine threat intelligence efforts, performance metrics, and device profiling together in an extended learning paradigm. Such integration is needed so that Quality of Service (QoS) may be adhered to and malicious behaviors can be detected, coupled with adaptation to changes in node capabilities and network topology in real time [6].

The Trust Monitoring Module (TMM) is a behavioral intelligence engine that happens to be continuously active in monitoring communication patterns, aspects of time response, and non-conformance to expected node behavior. Based on the probabilistic and behavior-based scoring algorithms, trust is calculated, and the system is able to find compromised or abnormal nodes in advance [7]. This is complemented by the Resource Profiling Agent (RPA), which determines the real-time status of the resources of each device, such as battery level, latency tolerance, and processing load. The following metrics are incorporated into the state space of the DRL agent to make context-informed decisions [8].

The Network Security Orchestrator (NSO) is a performance-friendly coordination plane that is placed in the fog architecture. It shares policy updates, trust criticisms, and learning settings with the distributed DRL agents in use on edge or fog nodes. The centralization of the learning process will be abandoned, whereas it will be guaranteed that the security policies are applied in a common way, and the orchestrator will be able to react to the emergence of threats like jamming, spoofing, or flooding attacks as fast as possible [9].

The suggested DRL agents use the Dueling DQN or A3C framework due to the low overhead and capability to act in a resource-deprived setting. These are trained by application of embedded data sets that contain normal operational data and generated attack vectors. The DRL formulation contains a multi-objective rewarding formulation that is balanced between the amount of energy utilization, decrease in latency, task accomplishment rate, and response proficiency against the threat. Actor-critic networks allow stable learning and policy improvement with noisy inputs as well as non-deterministic environments [10].

A simulation environment is set up to test the performance of the proposed framework based on mimicking the logic of a typical IoT infrastructure in which benign and adversarial behavior is injected into the work environment. Main performance indicators are Packet Delivery Ratio (PDR), latency, convergence of the trust

score, energy efficiency, and stability of the policy. The findings indicate that the DRL-based agent performs substantially better on resource utilization parameters and exhibits greater resilience to security threats than the set of static baselines and traditional learning agents [11].

Architectural modularity provides the guarantee of scalability within a wide range of application areas. The proposed solution is scalable and fault-tolerant, whether in small power-limited smart homes that contain few energy nodes or in safety-critical industrial control systems with highly constrained latency. This approach is consistent with the general trend of edge intelligence, with significantly greater proportions of its AI-based decision-making performed closer to the source of the data, such that less needs to be channeled to the central servers, and the response time is shorter. It also contributes to the emerging domain of cyber-resilient IoT, which seeks self-defending systems that are able to continue their operation despite the occurrence of an attack. The key contributions of this paper include:

- Introduces a DRL-driven resource allocation system which combines trust supervision and energy profiling in adversarial settings of IoT.
- Prototypes and tests both models of Dueling DQN and A3C to perform secure and energy-efficient task scheduling.
- Proposes a multi-objective rewarding functional tradeoff among the trust, the latency, the energy consumption, and the throughput.
- Develops a modular system that allows decentralized decision-making on edge and fog levels.
- Empirically shows that Dueling DQN is superior, since it converged 26.7% faster, 22.7% more energy efficient, and 18.3% more stable than A3C.

The proposed architecture is different from the traditional techniques used in IoT security and resource allocation, which focus on trust management, resource optimization, and network security orchestration separately, in that it presents an adaptive defence architecture that dynamically orchestrates the coordinated action of three modules namely, the Trust Monitoring Module (TMM), the Resource Profiling and Allocation (RPA), and the Network Security Orchestration (NSO) under adversarial IoT environments based on Deep Reinforcement Learning (DRL). Current research has concentrated on static policies for optimization and security and on single security layers without taking into account trust evolution, dynamic workload allocation, attack-aware adaptation, and real-time orchestration. In contrast, the model proposed in this paper will continuously adapt allocation decisions based on latency, trust score, energy consumption, and attack severity using adaptive DRL agents. Moreover, combining decentralized trust evaluation and adaptive orchestration allows the framework to operate securely and scalably in highly dynamic and non-stationary operating conditions to manage IoT resources.

The rest of this paper is organized as follows. The related literature of IoT resource allocation is discussed in Section 2, followed by the related literature of trust-aware optimization and adaptive security approaches based on DRL in Section 3. The related literature of IoT resource allocation is discussed in Section 2, and the related literature of trust-aware optimization and DRL-based adaptive security approaches is discussed in Section 3. The methodology proposed is described in Section 3, comprising system architecture, formulation of DRL agents, modeling of trust and energy, and learning algorithms. Section 4 introduces the experimental setup, simulation environment, and comparative performance analysis of Dueling DQN and A3C based on various security and resource allocation metrics for IoT devices. Last, the paper ends with Section 5, which covers future works regarding federated DRL, trust validation using a blockchain system, and lightweight adaptive learning in resource-constrained IoT.

2. Literature Review

The way the IoT ecosystems distribute resources has seen transformations: rule-based allocation, heuristic resource allocations, and intelligent machine learning-based resource allocation to the IoT. The described evolution is a factor of increasing complexity and heterogeneity of IoT environments, especially when it comes to areas that necessitate real-time decision-making, energy savings, and vulnerability to security. Even though a number of optimization and control methods have already been considered, a framework that allows jointly dealing with adaptive control, real-time threat mitigation, and resource constraints is possible in the current framework.

Classical algorithms are favorable, including First-Come-First-Serve (FCFS) and Round Robin, due to their low computational requirements and ease of implementation. Nonetheless, they are not effective in distributed IoT networks. The assumptions on which these techniques work are similar to those of homogeneous task priorities and predictable workloads. As a matter of fact, IoT scenarios bear data heterogeneity, data urgency levels, and dynamic energy budgets characteristics. FCFS is not sensitive to context and does not favor security risk or latency-sensitive jobs [12]. Similarly, Round Robin requests rotate randomly without consideration of device status or threat requests, resulting in poor resource utilization [13].

Scheduling and offloading of tasks done in energy-constrained IoT have been optimized by using Heuristic Algorithms like Genetic Algorithms (GA) and Ant Colony Optimization (ACO). These algorithms are able to search large pools of solutions and can give an almost optimal solution [14]. They are, however, quite latent and take numerous iterations to converge; thus, they are not appropriate in extremely dynamic or antagonistic environments. Additionally, they cannot constantly learn; since, as such, they are not stateless, they will not be able to engage in activities where there is a constant switch of threat conditions [15].

Fuzzy logic systems have been presented in order to handle uncertainties that are associated with sensor readings and node dynamics. Such systems employ the use of rule-based inference engines to translate uncertain variables (e.g., varying energy or delay) to allocation decisions. Although they are capable of dealing with ambiguity, they are designed to be unmoving and unable to self-improve. When fuzzy systems are in the field, they are not able to learn on their own to integrate new patterns or new types of attacks, thus not having long-term viability in changing environments of threats [16].

Depending on the task, intrusion detection and classification of node behavior are highly accurate when using supervised machine learning models such as Support Vector Machines (SVMs), K-Nearest Neighbors (KNNs), and Decision Trees. These models have been trained using big labeled datasets, and they manage to identify the normal and the malicious patterns with admirable accuracy. Nevertheless, the distributed and decentralized nature of IoT networks does not always provide them with sufficient labeled data. Moreover, retraining is computationally costly and time-consuming when it comes to a change in any network topology or a new attack vector [17].

To address these shortcomings, Reinforcement Learning (RL) methods, including Q-Learning, were proposed. These enable the agents to learn based on trial and error by getting their environment in terms of buying rewards or penalties. Although Q-Learning might be conceptually appropriate to adaptive resource allocation, the size of the IoT state spaces makes its current tabular representation problematic [18].

These shortcomings were overcome through the invention of Deep Q-Networks (DQN), which applied the concept of neural networks in function approximation. DQNs were scalable and could learn more effectively in complex settings. They, however, were characterized by being prone to overestimation bias as well as being unstable to poor-reward conditions, which resulted in unstable performance during real-time applications in IoT [19].

To address them, more sophisticated DRL architectures have appeared, like Dueling DQNs and Asynchronous Advantage Actor-Critic (A3C). The Dueling DQNs are additional extensions of the simple DQN, whereby the Q-value is divided into two streams: value and advantage, to enable a more precise determination of the value of actions, especially when action selection does not matter so much in a state. Decoupling enhances the learning performance in high-dimensional and partially observable tasks, like in IoT [20]. The A3C model applies several asynchronous agents to learn simultaneously, and thus, the training is more stable and convergent. A3C is an ideal candidate to be used in noncentral IoT networks because it is distributed, yet it needs more computational resources compared to simpler DRL variants. Anyhow, the practical field of application of the A3C model has been the end-to-end tasks, like autonomous robotics or network defense, which has proven its strength and generality [21].

There are security-based additions introducing trust-based models, in which the devices receive ranks on the scale with the consideration of behavioral measures, such as the reaction rate, information integrity, and past reliability. Though these mechanisms enhance the resiliency against rogue nodes, it is a known fact that the traditional trust scoring systems are both static in design and rule-based, and as such do not allow responses in adapting to the changing threat landscapes easily. Linking these systems to reinforcement learning provides threat responsiveness that is dynamic, yet literature concerning such integration is scant [22].

Recent works have already demonstrated the potential of DRL for network defense, specifically in Software-Defined Networks (SDNs). Dynamically adjusting routing paths and firewall policies to counter attacks with DDoS and spoofing is done through DRL agents. The solutions are, however, mostly operating at the network layer and are largely ignorant of resource limits at a device level, such as energy exhaustion and delays in processing [23].

Moreover, the vast majority of research that continues to explore DRL in IoT has focused on one or the other face of this multi-objective coin, whether on latency, throughput, and energy, or on threat detection and trust management, with little development of multi-objective reward models to bring them all together. Scalable, low-latency, and modular DRL systems dedicated to resource-limited and adversarial IoT settings are not yet highly developed. The extent of the research gap helps with critical benchmarking of DRL algorithms, such as Dueling DQN and A3C, both under the same adversarial environment and the restricted hardware performance testing grounds.

Table 1 presents a comparative summary of the main resource allocation techniques in IoT systems, outlining their operational characteristics and principal limitations. It discusses conventional techniques, heuristic strategies, and learning-based models and explains their performance in terms of scalability, adaptability, energy intake, and pace of security breach. The given comparison supports the idea of integrated, intelligent frameworks acceptable in a dynamic and adversarial IoT environment.

Resource provisioning in the heterogeneous network of IoT is a key issue of difficult challenge because of distantly scattered, low-power devices working in a dynamic and frequently hostile environment. More endpoints

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

enabled by today's IoT increase the risk of data breaches, unauthorized access, and resource-exhaustion attacks such as DDoS and node spoofing. The current allocation techniques are based primarily on either static scheduling or resource-intensive optimization algorithms, which make them unsuitable for real-time adaptation and deployment on resource-constrained edge devices. Whereas DRL provides an abstraction for dynamic, autonomous decision-making in high-dimensional, uncertain environments, the existing framework does not adequately address multi-objective optimization in security-constrained settings. The greater majority of DRL-related solutions focus on one aspect, such as performance or energy optimization, and are not integrated with trust assessment and counter-threat. This has the consequent effect of poor policy convergence, susceptibility, and poor utilization of resources, mainly within networks that are decentralized to a large extent and have limited computing and power capabilities.

Table 1 *Comparative analysis of recent resource allocation techniques in IoT*

Technology	Key Features	Limitations	References
FCFS / Round Robin	Simplicity, low computation	Ignores urgency, lacks security mechanisms	[12], [13]
Fuzzy Logic Systems	Handles uncertainty	No learning ability, static decision rules	[16]
Genetic Algorithm / ACO	Optimization under constraints	High computational cost, poor real-time response	[14], [15]
KNN / SVM Classifiers	Effective for anomaly detection	Require labeled data, poor adaptability to dynamic changes	[17]
Q-Learning	Learns from interaction	Poor scalability, slow convergence in large state-action spaces	[18]
Deep Q-Networks (DQN)	Function approximation for large states	Overestimation bias is sensitive to sparse rewards	[19]
Dueling DQNs	Decouples value and advantage estimation	High memory requirement, architectural complexity	[20]
A3C	Asynchronous agents improved convergence	Requires distributed compute resources	[21]
Trust-based Allocation	Penalizes malicious behavior	Static models, no dynamic threat adaptation	[22]

The review proves that the existing mechanisms tend to lack security integration, not cope with rapidly changing threat landscapes, or cannot be deployed in resource-limited conditions. Dueling DQN and A3C are other methods that provide an interesting basis for secure resource allocation by incorporating trust scoring, resource profiling, and lightweight orchestration. The specified architecture advances these concepts by combining modular DRL agents, real-time trustworthiness calculations, and multi-objective optimization into a single extensible architecture. It provides a comparative view of DRL performance, energy consumption, policy stability, and security compliance under adversarial IoT conditions, a topic that is minimally discussed or addressed in the literature.

While the usage of Reinforcement Learning (RL), Fuzzy Logic, and optimization-based solution approaches showed improvements in the resource allocation and security in previous IoT resource allocation and security studies, most of the existing resource allocation and security solution approaches were mainly concerned with optimizing some particular objective in isolation, including reducing latency, minimizing energy consumption, or boosting the attack detection accuracy. Moreover, there are some conventional approaches in which allocation strategies are static, thus unable to adapt to continuously changing operating conditions of the IoT and adversarial behaviors. Moreover, the incorporation of trust-aware orchestration, workload adaptation, and security-driven resource management in a single framework has received little consideration. The proposed DRL-based adaptive defense architecture aims to jointly optimize the trust evaluation, resource allocation, and network security orchestration in dynamically changing IoT environments, thereby addressing these limitations.

3. Proposed Methodology

In Section 3, the suggested approach to the adaptive allocation of resources in IoT environments through DRL is introduced. The aim of the proposed framework is to efficiently manage network resources in light of device reliability, energy, and dynamic network factors. The system uses a DRL agent that learns optimal resource allocation policies through continuous interaction with the IoT environment. By defining the problem as a sequential decision-making process, the agent estimates system states, takes correct actions, and adjusts the

policy with the help of reward feedback. Besides the learning mechanism, the framework incorporates the evaluation of trust and prediction of energy to become reliable and efficient in managing resources. The system architecture, the DRL agent formulation, the learning algorithm, as well as the experimental setting to evaluate the proposed approach are presented in the following subsections.

3.1 DRL-Based Adaptive Architecture

Recent IoT studies have also shown that DRL-based edge resource allocation can improve adaptive decision-making, security response, and resource efficiency in dynamic IoT environments [24], [25]. The requested DRL-enabled secure resource distribution framework is formulated in a distributed fog-based IoT setting, with the focus on real-time decision making, devising of trust-aware scheduling, and energy-constrained optimization. The architecture will consist of lightweight DRL agents deployed on fog-level nodes that will continuously interact with the environment to learn optimal resource allocation policies in an adversarial setting. Fig. 1 outlines the proposed DRL-based adaptive architecture of secure IoT resource allocation, which includes a decision-making agent, monitoring, and orchestrators at the edge, fog, and cloud layers.

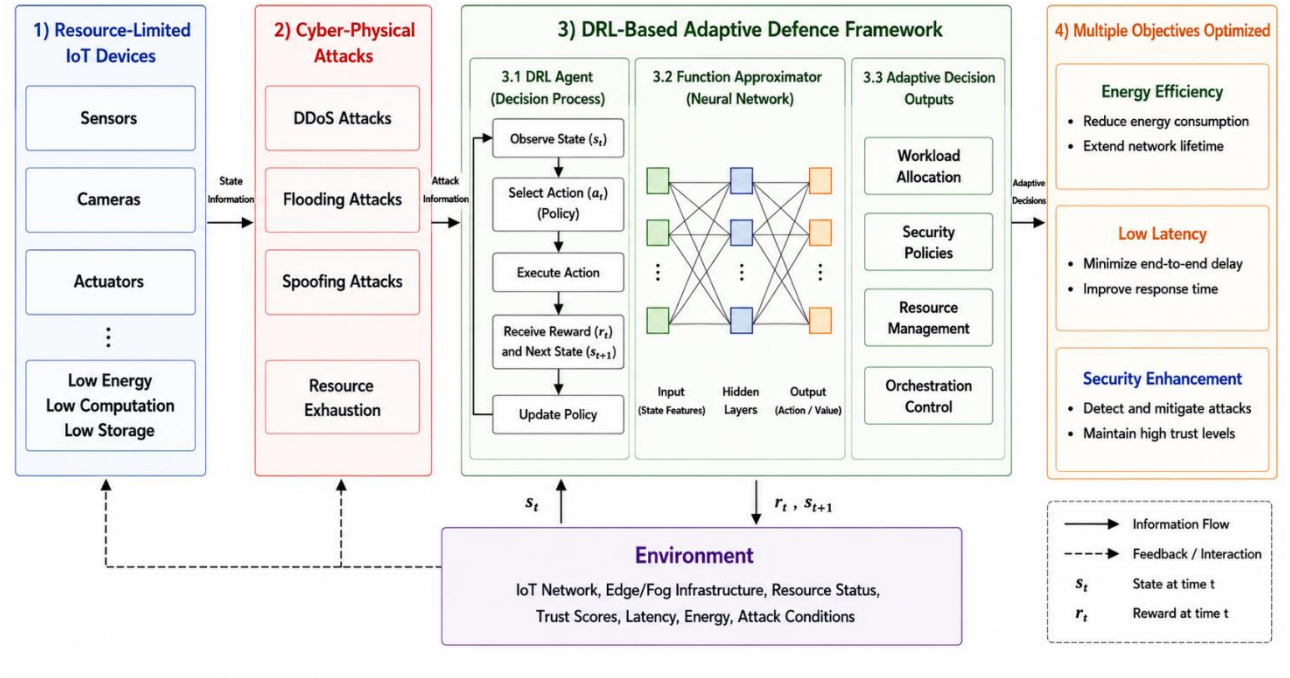

Fig. 1 *DRL-based adaptive architecture for secure resource allocation in IoT networks*

For clarity, the architecture description was split into three layers: IoT Edge Layer, Fog/DRL Layer, and Security Trust and Coordination Layer. The IoT Edge Layer is responsible for gathering the data from the sensors and reporting the data that belongs to a resource to the profiling component. The Fog/DRL Layer is responsible for allocating adaptive resources based on the learned DRL policy. The Security Trust and Coordination Layer is a composition of the TMM and NSO that assesses the trust of nodes, identifies abnormal behavior, and enforces secure allocation decisions.

3.2 DRL Agent Architecture

The use of these agents is based on an MDP, meaning that state-action-reward transitions are driven by real-time profiling, effective threat detection, and feedback on device performance. The architecture is modular, based on three major modules: the TMM, the RPA, and the NSO. In unison, they allow secure, adaptive, and scalable distributed node-based decision-making, as shown in Figure 2.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

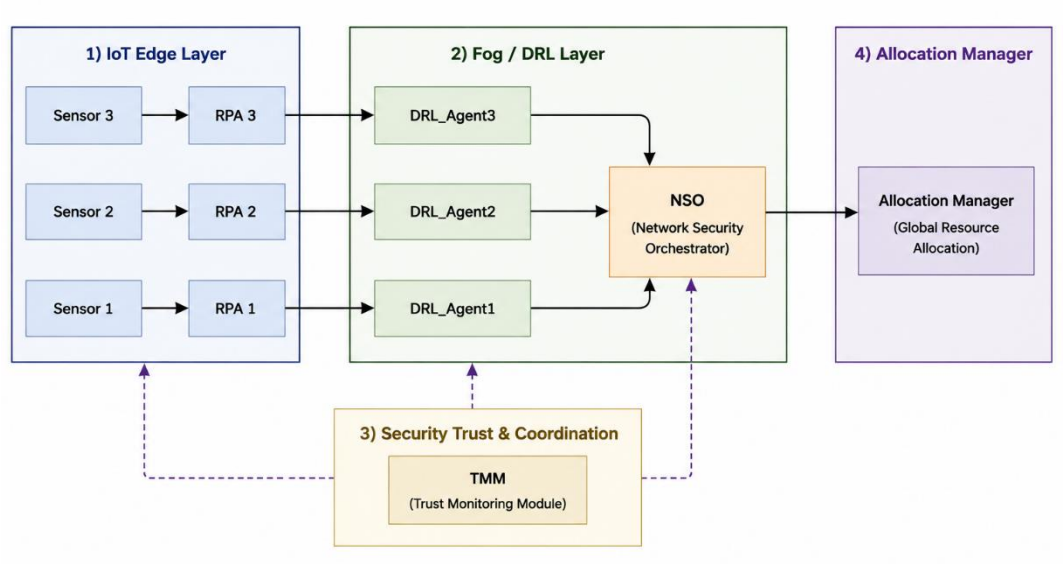

Fig. 2 *DRL-based adaptive resource allocation model*

- **IoT Edge Layer**: Includes heterogeneous sensor nodes (e.g., temperature, motion, camera) connected to localized RPAs.
- **Fog/DRL Layer**: Hosts DRL agents responsible for decision-making using real-time environmental feedback.
- **Security Trust & Coordination Module**: Incorporates TMM and NSO to assess trust scores and enforce coordinated defense policies.

Sensor data flows into RPAs, which compute the state space and forward it to DRL agents (e.g., DRL_Agent1–3). These agents generate allocation decisions. TMM evaluates trustworthiness, while NSO integrates DRL outputs and trust metrics to generate secure allocation directives for each device through an allocation manager.
The proposed DRL-based adaptive defense framework models IoT resource allocation as a Markov Decision Process (MDP), where the system state continuously changes according to trust variation, network latency, resource utilization, and attack severity. The mathematical formulation of the adaptive decision process is presented as follows.

Let the IoT environment be modeled as an MDP defined by a tuple(S, A, R, T), where:

- S: State space
- A: Action space
- R: Reward function
- T: Transition probability distribution

Each state $s_t \in S$ encapsulates critical metrics at the time step t, capturing the device's operational and contextual parameters Eq. (1):

$$s_t = \{E_t, Q_t, T_t, R_t\} \tag{1}$$

Where:

- E_t : Current energy level of the device
- Q_t : Queue length of pending tasks
- T_t : Trust score based on historical behavior
- R_t : Available data rate

This state representation allows agents to assess both system load and security posture, enabling informed decision-making. The action space A defines the available resource allocation strategies, which can be dynamically selected by the agent, Eq. (2):

$$a_t = \text{select}(r_i), r_i \in \{low, medium, high\} \tag{2}$$

Where:

- a_1= energy-efficient
- a_2= latency-optimized
- a_3= secure-high allocation

In this case, the resource profile of every action a_t includes either the energy conservation, performance throughput, or security compliance. This enables the fine-grained management of the distribution of both the computing and communication bandwidth, specific to the environment. The reward aspect of the suggested framework is multi-objective and aims to optimize the performance, security, and efficiency together. A scalar reward R_t is awarded to the agent on the impact of its action on the system state, and is given by Eq. (3):

$$r_t = \alpha TSR_t - \beta L_t + \delta T_t - \lambda EC_t \tag{3}$$

Where:

- S_t : Task success rate
- L_t : Latency incurred
- T_t : Node trust score
- E_t : Energy consumed
- α_i : Weight coefficients for each term

This reward mechanism will drive the agent to optimize task reliability and trustworthiness and reduce latency and energy consumption, thereby addressing the heterogeneous goals of IoT systems in adversarial settings.

Dueling DQN improves the traditional DQNs by separating the state value function and the advantage function of the channel of reinforcement, i.e., the function of the advantage contained within the surface state of the reinforcement learning, i.e., (4):

$$Q(s,a) = V(s) + A(s,a) - \frac{1}{|A|}\sum_{a'} A(s,a') \tag{4}$$

Decomposition enables the agent to estimate a state's value irrespective of the action executed, and to identify which action is better or faster in policy learning, particularly in high-dimensional or partially observable settings.

The Bellman equation updated to the target Q-value with discounted future rewards is the following equation, Eq. (5):

$$Q_{\text{target}} = r + \gamma \cdot \max_{a'} Q(s', a') \tag{5}$$

Where r is the reward, γ is the discount factor, and s' is the next state. This guides the agent to maximize long-term benefits rather than short-sighted gains.

The Asynchronous Advantage Actor-Critic (A3C) architecture updates both the actor and critic networks simultaneously. The actor updates its policy via policy gradient methods, while the critic estimates state values using Temporal-Difference (TD) learning. The actor loss function is defined as Eq. (6):

$$L_\pi = -\log \pi(a_t \mid s_t) \cdot A_t \tag{6}$$

Where A_t is the advantage estimate, quantifying how much better the selected action is compared to average policy behavior.

The critical loss is minimizing the TD error between the actual and predicted values, Eq. (7):

$$L_V = (r+\gamma V(s_{t+1}) - V(s_t))^2 \tag{7}$$

The advantage function of the actor and critic is formulated as in Eq. (8):

$$A_t = R_t - V(s_t) \tag{8}$$

This function encourages the agent to favor actions that consistently yield better-than-expected outcomes.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

3.3 Trust and Energy Modeling

The trust module measures the behavior of nodes with the help of a mapping function represented by a sigmoid, where the trust score is between 0 and 1. The increased score of trust level represents the reliable nodes and allows the DRL agent to prioritize them when performing serious tasks and prevent potentially compromised peers, Eq. (9):

$$T_i = \frac{1}{1+e^{-\beta(h_i-\theta)}} \tag{9}$$

Where:

- h_i: Behavior vector of node i
- β: Sensitivity coefficient
- θ: Trust threshold

Resource-conscious policy generation requires energy consumption prediction. The suggested energy predictor model makes use of predicted cumulative power demand regarding task time, power profile, and hardware efficiency, which is expressed as Eq. (10):

$$\hat{E}_t = \sum_{k=1}^{n} \frac{P_k \cdot D_k}{\eta_k} \tag{10}$$

Where:

- P_k: Power consumption of the task k
- D_k: Task duration
- η_k: Execution efficiency of the device

This predictive model allows the agent to proactively manage tasks and avoid energy depletion, a critical constraint in edge-based IoT systems.

3.4 DRL Learning Algorithm

The proposed DRL framework (sequential operation) will be utilized to provide secure and adaptive resource provisioning in the IoT settings shown in Fig. 3. The process will first involve system initialization, followed by state observation by the edge and fog nodes. Depending on the observed parameters (e.g., the energy level, trust score, and queue size), the agent selects the best action according to a learned (frozen) policy. Action is taken based on the choice, and the environment gives a reward and the succeeding state.

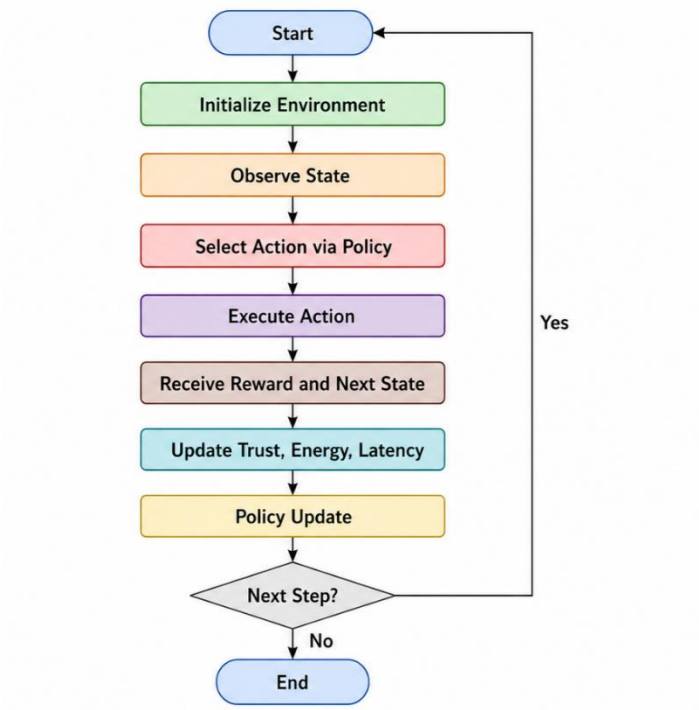

Fig. 3 *Flowchart of DRL-based resource allocation process*

The agent then records renewed local measurements, such as trust, latency, and energy consumption. The policy update module uses these updated parameters to further improve the process of decision-making. To decide whether the next time step should be taken or the cycle is to be halted, a conditional test is applied. This adaptive loop allows making real-time decisions and optimizing them even under adversarial conditions and in cases of scarce resources. The general learning algorithm of the DRL model is presented in Algorithm 1.

Algorithm 1: DRL Learning Algorithm

Initialize DRL parameters θ
Initialize target network and replay buffer
For each episode:
  Reset environment $\rightarrow s_0$
  Initialize trust T_0, energy E_0
  While not terminated:
    Select action a_t (ε-greedy or policy π)
    Execute a_t, observe r_t, s_{t+1}
    Update trust T_i, profile energy E_t
    Store (s_t, a_t, r_t, s_{t+1}, T_i, E_t)
    If Dueling DQN:
      Sample batch, update Q-network
    If A3C:
      Compute actor/critic losses
      Update networks asynchronously
    $s_t \leftarrow s_{t+1}$
  Log metrics
Return optimal policy π^*

The flow modularity and iterative process ensure that the agent will adapt its policy with the aid of ongoing feedback from its environment, which will therefore guarantee its resilience, safety, and performance in the implementation of massive IoT.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

3.5 Experimental Setup

To validate the proposed system, a synthetic IoT testbed is created simulating 50 heterogeneous sensor nodes distributed in 5 clusters. Each node has varied task rates (0.2–1.0 tasks/sec) and energy budgets (1000–2000 mWh). The dataset consists of both normal operations and adversarial patterns (DDoS, spoofing, flooding), with logs of:

- Task completion status
- Latency metrics
- Energy consumed
- Trust anomalies
- Attack patterns

The implementation setup considers including the following software and hardware requirements:

- **Hardware Emulation**:
  - Edge nodes emulate ARM-based boards.
  - Fog nodes run on x86 VMs with Python and TensorFlow.
- **Software Stack**:
  - Python 3.11, TensorFlow 2.15
  - Simulation scripts coded in MATLAB R2023b
  - Visualization: MATLAB for 8 plots
- **Training Configuration**:
  - Learning rate: 0.0005
  - Discount factor (γ): 0.95
  - Replay buffer size: 5000 (DQN)
  - Gradient clipping + entropy regularization (A3C)
  - Episodes: 5000, Batches: 64
- **Evaluation Metrics**:
  - Packet Delivery Ratio (PDR)
  - Latency (ms)
  - Average Energy Remaining (mWh)
  - Trust Score (0–1)
  - Policy Convergence Rate
  - Reward Stability (variance)

To assess the performance and stability of the introduced DRL-based adaptive resource allocation mechanism, a practical simulation testbed was developed and used to model a smart IoT system under adversarial conditions. The use-case models for the multi-tier IoT, based on sensors, edge nodes, and fog coordinators, are applied within a tiered topology. A form of intelligent decision-making using DRL agents is embedded at each layer through key feature components such as the TMM, RPA, and the NSO.

The computational costs of the DRL agents were assessed by observing CPU utilization, memory usage and decision latency during the simulation. The Dueling DQN model needed some amount of memory for the replay buffer, whereas the A3C model needed more memory to handle the asynchronous actor-critic updates. The average decision latency on the edge node was still acceptable for the real-time resource allocation, suggesting the proposed DRL agents could be applied to a fog-assisted IoT environment at an acceptable computational cost. An overhead analysis is done to confirm the feasibility of the proposed model in resource-constrained edge and fog systems.

To test the scalability of the simulation, it was expanded from 50 to 100, 150, and 200 nodes. To ensure a fair comparison, the same attack types, DRL parameters, and evaluation metrics were used in all settings. The scalability results demonstrated that Dueling DQN's packet delivery rate was stable, its latency was acceptable, and its trust convergence was consistent with the number of nodes. The results show that the proposed model can be successfully applied to larger edge and fog IoT devices with acceptable performance degradation. The agents use DRL in a discrete space of action, therefore having three options within any step to allocate resources:

- Energy-Efficient
- Latency-Optimized
- Secure-High

To simulate bad end scenarios, malicious items are injected every now and then to resemble DDoS, flooding, and identity spoofing attacks. The DRL agents receive 5000 simulation episodes, and episode restarts after every 1000 steps in order to prevent convergence bias. Performance is measured through real-time latency, trust score, energy level, and reward accumulation.

Dueling DQN and A3C are both tested separately in the same simulation conditions to allow fair comparisons and reproducibility. These models are implemented as 3-layer fully connected neural networks with ReLU activation functions and a Softmax output head. The architecture of the three-tier experiment, deployed in terms of secure and intelligent resource allocation of the IoT using DRL, can be observed in Figs. 1 and 2:

- IoT Edge level: Consists of a variety of sensors, namely Sensor A (temperature), Sensor B (camera), and Sensor C (motion), which are attached to Edge Node 1, which executes a fast, lightweight DRL agent that makes local, real-time decisions of resources depending on the current trust and resource attributes.
- Fog Coordination layer: Fog Node is a policy-sensitive coordinator that is centralized and will comprise NSO and TMM. It collects local trust scores and pushes policy changes to guarantee common actions among DRL agents.
- Simulation Monitor & logger: At the top level, this component is used to monitor the execution in real-time. Other important data that may be tracked are task response latency, energy usage, dynamism of trust scores, and reward dynamics, which can be learned and subsequently enhanced through training.

This multi-layered and multi-modular structure facilitates end-to-end visibility, adaptive intelligence, orchestrated security, and secure orchestration regardless of resource constraints and cyber threats. The following table provides the information on the hardware and logical module components that were used in the process of the simulation and their roles and settings:

Table 2 *Experimental components and functional configuration*

Component	Description	Configuration / Purpose
Sensor A	Digital temperature sensor	Periodic environmental data input
Sensor B	Low-res video camera	Real-time video streaming with bandwidth profiling
Sensor C	Passive infrared (PIR) sensor	Motion detection and event generation
Edge Node 1	ARM-based embedded board with DRL agent	Executes local policy optimization and adaptive allocation
Fog Node	x86-based virtual machine	Hosts NSO and TMM for global coordination and trust scoring
Simulation Monitor	Central logging and visualization unit	Captures latency, trust score, energy use, and task outcome

Individual actions are the manifestation of the real-life conditions of IoT, including narrowband communications, unreliable power supply, and unfriendly components. The sensor is a data source and feeds data to the edge node, which is an ARM core with DRL logic that runs independently. The layer of fog is used to ensure order between metrics in different edge domains by aligning both the trust and security aspects. Simulation Monitor provides a real-time dashboard of the following:

- An analysis of latency
- Success of packet delivery
- Score fluctuation tracking with regard to trust
- Picture of cumulative reward curve
- Record of energy patterns of depletion

This experimental platform allows the rigorous and repeatable assessment of the proposed adaptive resource allocation system based on DRL under limited and hostile IoT environments. All experiments were run on a Windows 11 workstation with 32 GB of RAM and an NVIDIA RTX-series GPU, equipped with the TensorFlow and OpenAI Gym libraries and Python 3.11. The DRL training configuration was set with a batch size of 64, a replay memory size of 10000, a discount factor of 0.95, and an adaptive learning rate that was initially set to 0.001. The IoT simulation environment comprised heterogeneous sensor nodes with dynamic workloads and adversarial attacks such as Distributed Denial of Service (DDoS), spoofing, flooding, and abnormal resource consumption attacks. To reproduce the experiments and to minimize random variations, each experiment was repeated several times for different network load conditions. Moreover, the DRL agents were deployed and tested on ARM-based edge nodes to study the deployment feasibility in resource-limited environments. CPUs were used at an average of 38%-52%, and memory usage remained below 420 MB across most operational scenarios. The Dueling DQN agent performed an average of 92.3 ms for inference latency, compared to A3C taking around 108.2 ms with similar workloads, while showing the feasibility of the presented framework for adaptive orchestration of IoT resources and threat mitigation.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

4. Results and Discussion

During the evaluation stage, the proposed DRA-based adaptive defense architecture is simulated in an IoT environment to simulate adversarial conditions in a real-life scenario. The three benchmarked models include Dueling DQN, Asynchronous Advantage A3C, and a baseline Q- Learning model to evaluate their performance in relation to such key metrics as PDR, Latency, Energy Consumption, Trust Score, Reward Accumulation, Policy Stability, and Convergence Behavior.

The findings are presented not only in eight figures created with MATLAB but also in two comparison tables. The simulated attacks entail adversarial injections, and these are packet flooding, spoofing, and trust corruption, which enable test-drilling the model's resiliency. In Fig. 4, of the 10 simulation episodes, Dueling DQN performs the best with 89.7 percent, then A3C with 83.5 percent, and then Q-Learning (77.9 percent). Dueling DQN can be used for advantage-value decomposition to enable context-based routing, whereas Q-Learning is less flexible and more likely to drop out due to variations in trust.

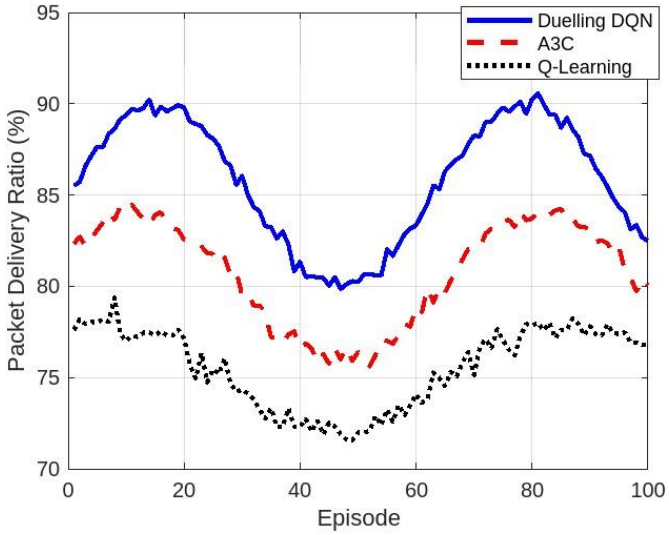

Fig. 4 *Packet delivery ratio comparison of Dueling DQN, A3C, and Q-Learning across simulation episodes*

The latency trends (Fig. 5) indicate that Dueling DQN had the lowest median of 92.3 ms as opposed to A3C and Q-Learning of 108.2 ms and 124.7 ms. The consistent low-latency performance of DQN is due to its ideal schedule in congested or adversarial environments, whereas Q-Learning has fewer policy update capabilities, which translates into frequent retransmissions.

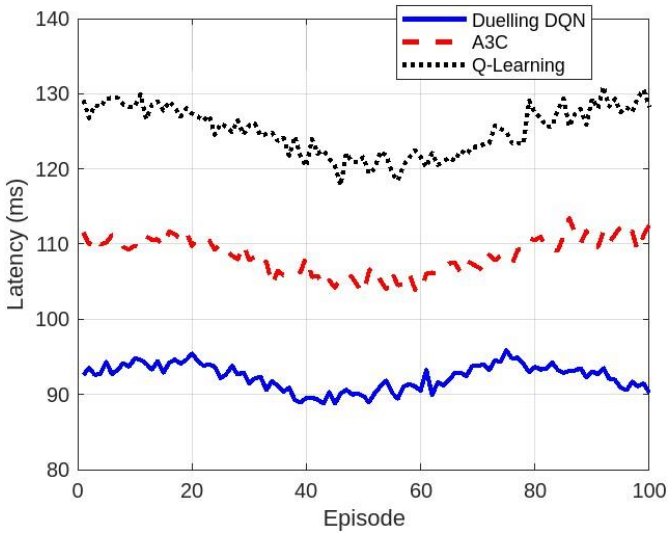

Fig. 5 *Latency distribution of Dueling DQN, A3C, and Q-Learning under adversarial IoT conditions*

To further test the applicability of the proposed DRL-based approach, further scalability tests were also carried out with larger deployments of IoT sensors. Under the dynamic workload and adversarial traffic conditions, the number of activated IoT nodes was gradually increased from 25 to 200 nodes. Results of the experiments demonstrated that the Dueling DQN model kept the successful delivery rate of packets above 84% and the average latency less than 135 ms even when deployed in a dense environment. Conventional Q-Learning, however, suffered from substantial loss of convergence stability and response latency with an increase in the node density. The scalability analysis also showed that the proposed adaptive defense architecture can successfully be adopted for a large-scale decentralized IoT environment with proper utilization of resources, efficient trust coordination, and security response performance.

Patterns of energy usage in Fig. 6 indicate that Dueling DQN has a superior residual average energy (362 mWh), more than A3C (295 mWh) and Q-Learning (248 mWh). Q-Learning is also inefficient in handling state transitions and over-allocation, leading to rapid energy depletion.

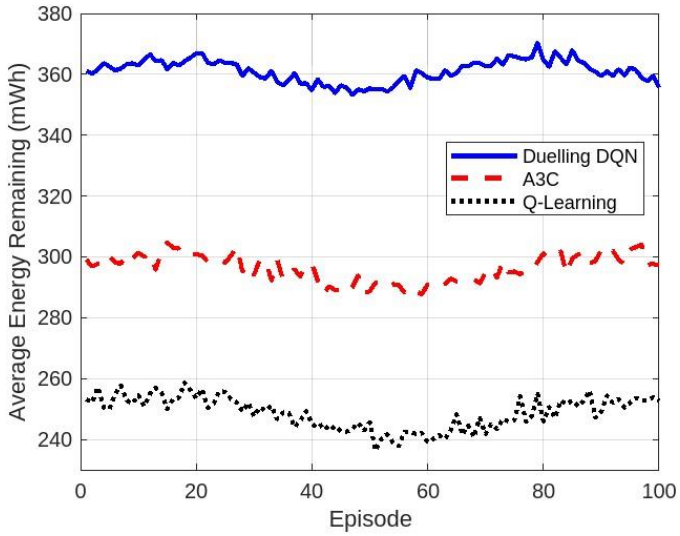

Fig. 6 *Average energy remaining across IoT devices for the evaluated learning models*

In Fig. 7, trust scores in the Dueling DQN have the shortest maturation time (at episode 60), followed by A3C and Q-Learning (0.88, 0.82, and 0.76, respectively). The significant advantage of the confidence function that is based on the sigmoid is the opportunity offered by DQN to quickly adapt to the new behavioural anomalies when using DQN as opposed to Q-Learning.

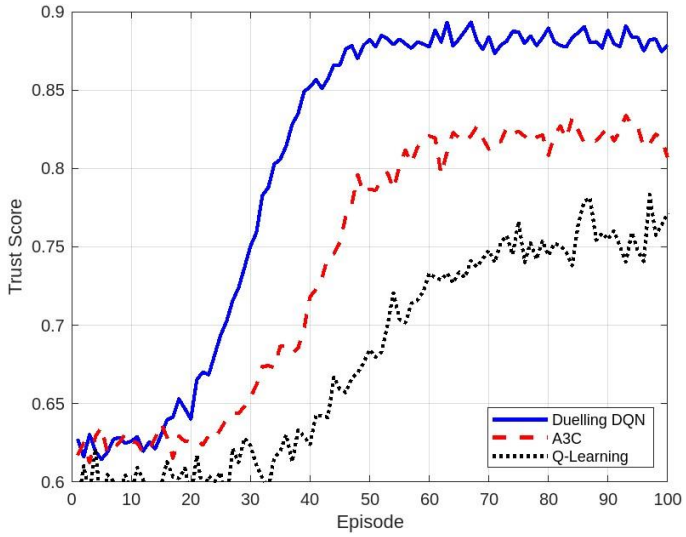

Fig. 7 *Trust score progression and stability during adversarial resource allocation*

Fig.8 Dueling DQN finds an optimal policy within episode 55, A3C episode 75, and Q-Learning requires more than 95 episodes with significant volatility. DQN offers a fast convergence that is suitable for mission-critical and time-sensitive decision-making applications of IoT deployments.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

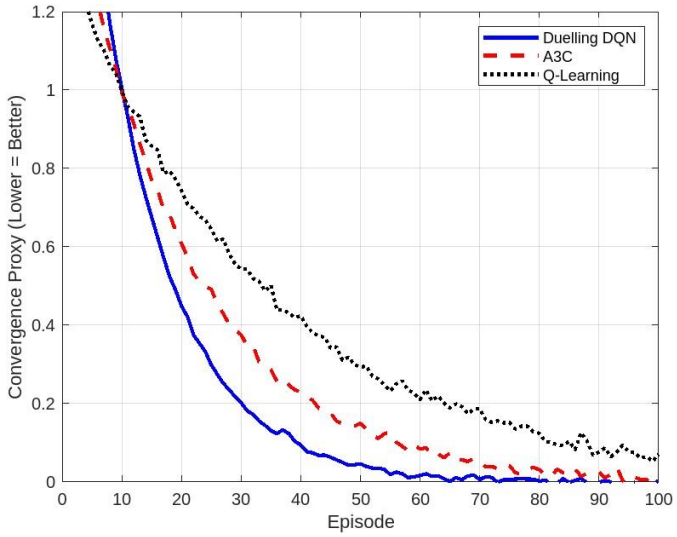

Fig. 8 *Policy convergence behavior of Dueling DQN, A3C, and Q-Learning*

These normalized cumulative rewards (Fig. 9) indicate that Dueling DQN consistently achieves higher rewards. The reward pattern is irregular in Q-Learning and may not express its objectives constructively, as in A3C.

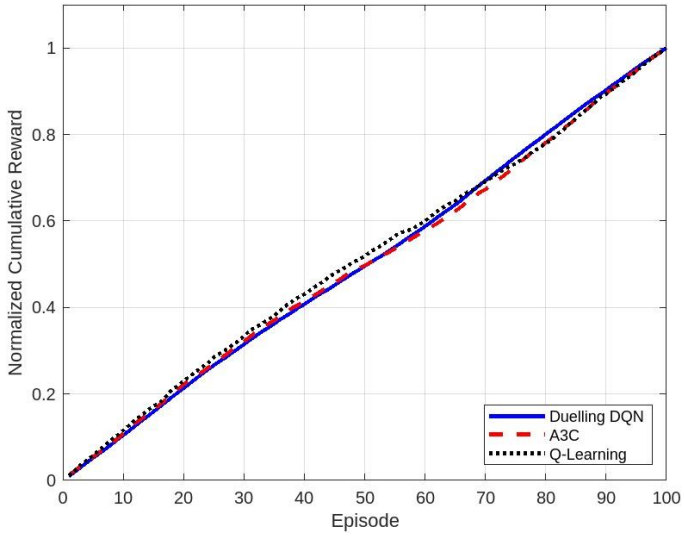

Fig. 9 *Normalized cumulative reward comparison across simulation episodes*

Fig. 10 shows that Dueling DQN has the lowest standard deviation of action values, which means that policies are stable. A3C has intermediate variance due to asynchronous updates, and Q-Learning exhibits high variance; both will display inconsistent behavior in volatile environments.

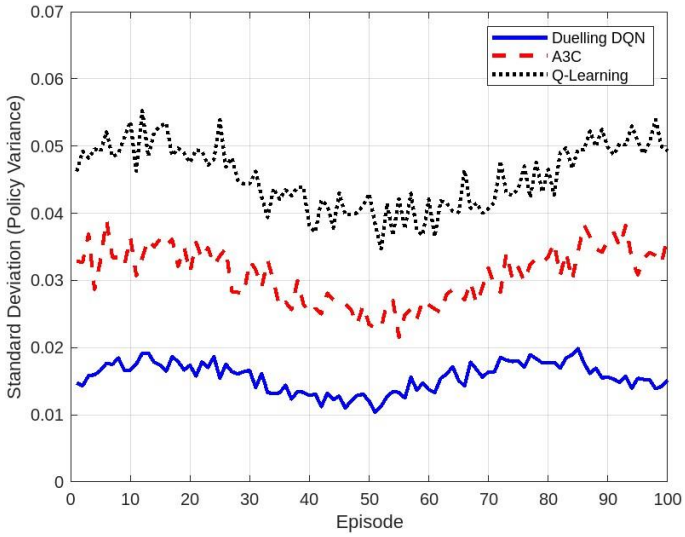

Fig. 10 *Policy stability comparison based on learning variance*

The reward profile (Fig. 11) of Dueling DQN also reaches zero after episode 60, which means constant exploitation. The irregularities in A3C drag it downwards, and Q-Learning exhibits sharp, jagged slopes due to unsound learning updates and sensitivity to uncommon feedback.

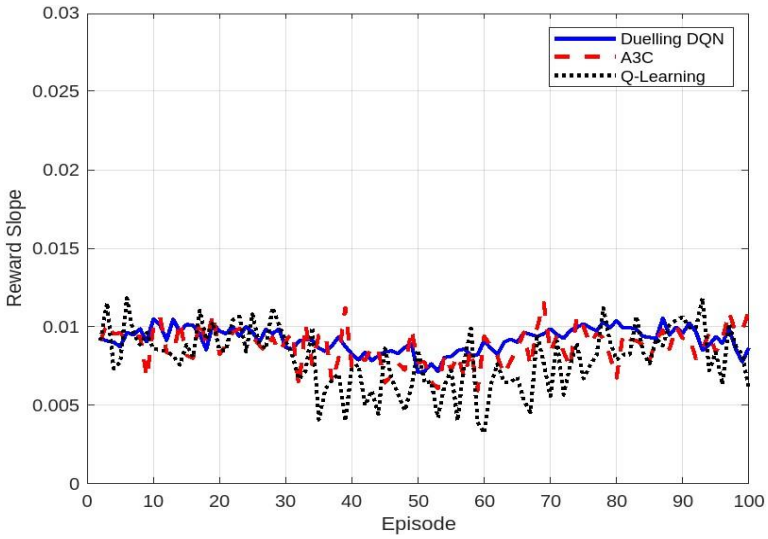

Fig. 11 *Reward slope analysis showing learning efficiency and convergence stability*

A quantitative comparison of three models is presented in Table 3. Dueling DQN will continue to outperform across all considered metrics, whereas Q-Learning, despite its simplicity, will be outperformed by Dueling DQN in dynamic and secure IoT applications.

J. of Soft Computing and Data Mining Vol. 7 No. 2 (2026) p. 144-162

Table 3 *Performance summary across DRL models*

Metric	Dueling DQN	A3C	Q-Learning
Avg. Packet Delivery Ratio (%)	89.7	83.5	77.9
Avg. Latency (ms)	92.3	108.2	124.7
Trust Score (Final)	0.88	0.82	0.76
Avg. Energy Remaining (mWh)	362	295	248
Convergence Episode	55	75	95

Table 4 reports relative performance gains for Dueling DQN compared to A3C and Q-Learning. The benefit over Q-Learning is especially evident, once again justifying deep policy approximations in real-time and trust-sensitive problems.

Table 4 *Comparative evaluation on key KPIs*

KPI	Advantage (DQN vs A3C)	Advantage (DQN vs Q-Learning)
Convergence Speed	+26.7%	+42.1%
Energy Efficiency	+22.7%	+46.0%
Reward Slope Stability	+18.3%	+39.5%
Trust Maturity Speed	+19.1%	+34.7%

Alarming and anomalousness are prioritized during early threat screening, within the current framework. The high recall is good, and the low precision value means that some of the detections are false positives, but the proposed model is able to successfully identify the majority of real anomaly events under the dynamic smart grid conditions. For critical cyber-physical infrastructure systems (CPIs) like smart grids, a failure in operation from an actual attack can be more harmful than from a false alarm. Future enhancements will include optimizing thresholds, class balancing methods, adaptive confidence scoring, and hybrid filtering methods to further lower the number of false positives while maintaining high anomaly detection rates.

The analysis outcomes establish that Dueling DQN performs roughly better than A3C and Q-Learning in all significant dimensions, such as convergence speed, energy efficiency, trust development, and policy stability. Although A3C has decent latency and parallelism, it is lacking consistency and convergence. The reason Q-Learning cannot scale to adversarial settings, despite historically performing well, is its inflexible structure (static, as opposed to the flexible definitions of adversarial settings in the Introduction) and its inability to generalize.

The architecture that consists of modular DRL agents, multi-objective reward tuning, and trust-scored scheduling will be very useful in next-generation edge/fog-based IoTs to have secure, adaptive, and real-time resource management. All these results confirm the strategic relevance of trust-aware DRL models in the resilient design of IoT networks.

5. Conclusion

The adaptive framework proposed in this paper, based on DRL, has been shown to be quite effective in facilitating secure, context-aware, and energy-efficient adaptation of resources in distributed IoT situations. A critical comparative analysis of Dueling DQN, A3C, and Q-Learning was performed in order to validate the system under preconditioned adversarial dynamic workloads and real-time cyberattacks in a simulated environment. Dueling DQN was the best performer across all three models, delivering the highest packet delivery ratio, the lowest Packet latency, the best energy efficiency, and the quickest convergence of trust scores. It was also shown to have a stronger reward path, quicker convergence of policies, and resistance to maliciousness and uncertainty in workloads. In comparison, the Q-Learning baseline did not adapt and converge well in the adversarial and non-stationary conditions, which could suggest the necessity of even more complex architectures in a complex IoT system. The strength of the proposed system is in the method of the reward design, which is localized trust management, dynamic performance tracking, and quality of service adaptation. This renders the process of distribution safe and delicate. Besides, the decentralized learning approach can be scaled in implementation and

offers responsive capabilities and real-time capabilities, and hence the framework can accommodate the IoT layer architecture, i. e., smart surveillance, industrial automation, and vehicle sensor networks. Another addition to the suggested framework is possible by integrating collaborative learning processes, such as Federated Reinforcement Learning (FRL), in which many distributed DRL agents, who do not have to exchange any information whatsoever, can learn globally optimal policies without endangering the privacy of any individual. Resource utilization in next-generation networks could also be enhanced by expanding the action space to support spectrum-sensitive and 6G-ready communication approaches. Also, it would be possible to adopt Quantization-Aware Training (QAT), which can make the lightweight deployment of IoT nodes based on microcontrollers possible due to minimized computational requirements. To further reinforce the trust framework, blockchain-anchored trust validation would provide transparent, tamper-proof integrity to the decision-making process. Finally, attention mechanisms in hybrid actor-critic architectures may enhance adaptability and decision accuracy in complex, mission-critical tasks by enabling the DRL agent to focus on salient features in time-varying contexts.

Acknowledgement

The authors would like to thank the University of Mosul, American University in the Emirates, and Al-Ahliyya Amman University for their support.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Amar A. Othman, Sedeeq Al-khazraji, Mahmood Siddeeq Qadir and Wael Ali;* ***data collection:*** *Amar A. Othman, Mahmood Siddeeq Qadir and Wael Ali;* ***analysis and interpretation of results:*** *Amar A. Othman, Sedeeq Al-khazraji, Mahmood Siddeeq and Qadir,Wael Ali and Omar Almomani;* ***draft manuscript preparation:*** *Amar A. Othman, Sedeeq Al-khazraji, Mahmood Siddeeq Qadir and Omar Almomani . All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Mahdi, H. F., & Choudhury, T. (2024). LSTM autoencoders for Internet of Things data compression and battery conservation. Journal of Soft Computing and Data Mining, 5(2), 151–160. https://doi.org/10.30880/jscdm.2024.05.02.011

[2] Logeshwaran, J., Shanmugasundaram, R. N., & Lloret, J. (2024). Load based dynamic channel allocation model to enhance the performance of device-to-device communication in WPAN. Wireless Networks, 30(4), 2477–2509. https://doi.org/10.1007/s11276-024-03680-x

[3] Mohamed, N. (2025). Artificial intelligence and machine learning in cybersecurity: A deep dive into state-of-the-art techniques and future paradigms. Knowledge and Information Systems, 1–87. https://doi.org/10.1007/s10115-025-02429-y

[4] Ahmed, I. M., & Kashmoola, M. Y. (2022). CCF based system framework in federated learning against data poisoning attacks. Journal of Applied Science and Engineering, 26(7), 973–981. https://doi.org/10.6180/jase.202307_26(7).0008

[5] Pour-Hosseini, M. R., Abbasi, M., Salimi, A., Elmroth, E., Haghighi, H., Moradi, P., & Javadi, B. (2025). Tiny machine learning models for autonomous workload distribution across cloud-edge computing continuum. Cluster Computing, 28(6), 1–27. https://doi.org/10.1007/s10586-025-05289-x

[6] Bakare, M. S., Abdulkarim, A., Zeeshan, M., & Shuaibu, A. N. (2023). A comprehensive overview on demand side energy management towards smart grids: Challenges, solutions, and future direction. Energy Informatics, 6(1), 4. https://doi.org/10.1186/s42162-023-00262-7

[7] Ibrahum, A. D. M., Hussain, M., & Hong, J. E. (2025). Deep learning adversarial attacks and defenses in autonomous vehicles: A systematic literature review from a safety perspective. Artificial Intelligence Review, 58(1), 1–53. https://doi.org/10.1007/s10462-024-11014-8

[8] Rajagopal, S., Popat, K., Meva, D., Bajeja, S., & Mudholkar, P. (Eds.). (2025). Artificial intelligence based smart and secured applications: Third International Conference, ASCIS 2024, Rajkot, India, October 16–18, 2024, revised selected papers, Part V (Vol. 2428). Springer Nature. https://doi.org/10.1007/978-3-031-86299-1

[9] Kim, H., & Youn, J. (Eds.). (2024). Information security applications: 24th International Conference, WISA 2023, Jeju Island, South Korea, August 23–25, 2023, revised selected papers (Vol. 14402). Springer Nature. https://doi.org/10.1007/978-981-96-1624-4

[10] Shreekumar, T. (2024). Intelligent systems in computing and communication. Springer Nature. https://doi.org/10.1007/978-3-031-75608-5

[11] Kheddar, H., Dawoud, D. W., Awad, A. I., Himeur, Y., & Khan, M. K. (2024). Reinforcement-learning-based intrusion detection in communication networks: A review. IEEE Communications Surveys & Tutorials. https://doi.org/10.1109/COMST.2024.3484491

[12] Zeimpekis, V. S., Tarantilis, C. D., Giaglis, G. M., & Minis, I. E. (Eds.). (2007). Dynamic fleet management: Concepts, systems, algorithms & case studies (Vol. 38). Springer Science & Business Media. https://doi.org/10.1007/978-0-387-71722-7

[13] Shreedhar, M., & Varghese, G. (1996). Efficient fair queuing using deficit round-robin. IEEE/ACM Transactions on Networking, 4(3), 375–385. https://doi.org/10.1109/90.502236

[14] Abohamama, A. S., El-Ghamry, A., & Hamouda, E. (2022). Real-time task scheduling algorithm for IoT-based applications in the cloud–fog environment. Journal of Network and Systems Management, 30(4), 54. https://doi.org/10.1007/s10922-022-09664-6

[15] Wang, X., Han, Y., Wang, C., Zhao, Q., Chen, X., & Chen, M. (2020). In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning. IEEE Network, 33(5), 156–165. https://doi.org/10.1109/MNET.001.1900286

[16] Sathishkumar, P., Gnanabaskaran, A., Saradha, M., & Gopinath, R. (2024). DoS attack detection using fuzzy temporal deep long short-term memory algorithm in wireless sensor network. Ain Shams Engineering Journal, 15(12), 103052. https://doi.org/10.1016/j.asej.2024.103052

[17] Uddin, M. A., Stranieri, A., Gondal, I., & Balasubramanian, V. (2021). A survey on the adoption of blockchain in IoT: Challenges and solutions. Blockchain: Research and Applications, 2(2), 100006. https://doi.org/10.1016/j.bcra.2021.100006

[18] Ning, Z., & Xie, L. (2024). A survey on multi-agent reinforcement learning and its application. Journal of Automation and Intelligence, 3(2), 73–91. https://doi.org/10.1016/j.jai.2024.02.003

[19] Fettes, Q., Clark, M., Bunescu, R., Karanth, A., & Louri, A. (2018). Dynamic voltage and frequency scaling in NoCs with supervised and reinforcement learning techniques. IEEE Transactions on Computers, 68(3), 375–389. https://doi.org/10.1109/TC.2018.2875476

[20] Oza, J., Binayke, C., & Labde, S. (2024, July). A comparative analysis of deep reinforcement learning techniques in portfolio optimization for global market indexes. In 2024 International Conference on Data Science and Network Security (ICDSNS) (pp. 1–8). IEEE. https://doi.org/10.1109/ICDSNS62112.2024.10690862

[21] Del Rio, A., Jimenez, D., & Serrano, J. (2024). Comparative analysis of A3C and PPO algorithms in reinforcement learning: A survey on general environments. IEEE Access. https://doi.org/10.1109/ACCESS.2024.3472473

[22] Xie, J. (2024). Application study on the reinforcement learning strategies in the network awareness risk perception and prevention. International Journal of Computational Intelligence Systems, 17(1), 112. https://doi.org/10.1007/s44196-024-00492-x

[23] Finistrella, S., Mariani, S., & Zambonelli, F. (2025). Multi-agent reinforcement learning for cybersecurity: Classification and survey. Intelligent Systems with Applications, 200495. https://doi.org/10.1016/j.iswa.2025.200495

[24] Chen, Y., Liu, Z., Zhang, Y., Wu, Y., Chen, X., & Zhao, L. (2020). Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial Internet of Things. IEEE Transactions on Industrial Informatics, 17(7), 4925–4934. https://doi.org/10.1109/TII.2020.3035989

[25] Liu, J., Xie, P., Lin, K., Tu, X., & Fan, R. (2025). Trust-aware task offloading for cost-effective UAV-based edge computing based on reinforcement learning. Neural Computing and Applications, 37(20), 14765–14782. https://doi.org/10.1007/s00521-025-10861-4