Applying Artificial Intelligence Techniques For Resource Management in the Internet of Things (IoT)

Original Scientific Paper

Salwa Othmen*
Computers and Information Technology Department, College of Science and Arts Turaif, Northern Border University, Arar, Saudi Arabia
e-mail: Salwa.Othmen@nbu.edu.sa

Wahida Mansouri
Computers and Information Technology Department, College of Science and Arts Turaif, Northern Border University, Arar, Saudi Arabia

Radhia Khdhir
Department of Computer Science, College of Science and Arts in Qurayyat, Jouf University, Saudi Arabia

*Corresponding author

Abstract *– Internet of Things (IoT) applications in smart cities (SCs) rely on free-flow services streamlined by artificial intelligence (AI) paradigms. However, the nature of resource constraint prevails due to external infrastructure costs and energy-based allocations. Existing approaches to smart city resource distribution rely on static thresholds or reactive responses, which are not always sufficient. These approaches may limit system performance and scalability in dynamic IoT environments owing to increased energy consumption, postponed resource allocation, and frequent device failures. This article introduces a Concerted Resource Management (CRM) using the Leveled Reinforcement Training (LRT) method. The proposed method accurately identifies cost-complex and high energy-consuming sharing intervals based on service response time and device failure. The reinforcement learning and training concerts both energy and device incorporations for SC applications based on its demand. This process requires leveled training in resource management, from energy depletion to device activeness. The interrupted sessions are identified using resource allocation failures, and the active resources with optimal energy expenses are selected to pursue resource management. The training method thus identifies the demands based on independent or concerted resource allocations to mitigate the management constraints in an SC environment. This proposed method reduces the resource constraint-based waiting for allocations and allocation failures in any SC application services. Under the varying devices, the following is observed: Improvements: 9.1% (Allocation Rate), 10% (Device Detection), 11.88% (Constraint Mitigation—Energy), 9.06% (Constraint Mitigation—Resource Allocation); Reduced: 8.01% (Allocation Failure), 9.64% (Waiting Time).*

Keywords: *Power flow control, three port converter, high frequency transformer, phase shift control technique*

Received: August 23, 2024; Received in revised form: November 24, 2024; Accepted: November 25, 2024

1. INTRODUCTION

Resource management is a process that ensures the network has all the required resources to complete or perform a task. RM is widely used in organizations to reduce unwanted difficulties in performing user tasks [1, 2]. IoT-based smart city environments are commonly implemented to enhance the overall performance level of the network [3, 4]. The analyzed data produces optimal features for resource allocation in smart cities. The management framework improves the services' lifespan, maximizing the performance range of smart cities [5, 6].

IoT devices are advanced technology that increases the capability level of smart cities. IoT-enabled devices reduce smart cities' latency and energy consumption [7, 8]. IoT devices provide various heterogeneous services to users, which enhances the feasibility and robustness of smart cities. IoT devices use wireless sensors, which reduces the energy consumption ratio when providing services to users. Wireless sensors collect the data from the base station, which minimizes the complexity of analyzing the necessity of the request [9, 10].

Many cities are embracing smart technology to build data-centric settings that can instantly analyze massive amounts of data. City planners, traffic engineers, and resource allocators may all benefit from the insights provided by machine learning algorithms that study this data [11, 12]. The study's effectiveness in an ever-changing Internet of Things (IoT) setting depended on Leveled Reinforcement Training's (LRT) hierarchical framework for smart city resource management. Learning rate tuning (LRT) allows for the progressive learning and adaptation of resource allocation choices in response to changing demand and energy usage [13, 14]. The system may progressively enhance its decision-making capabilities using LRT's multi-tiered learning technique. It starts with basic scenarios with low energy consumption and moves on to more complicated ones with high device activity and resource constraints. By adopting this tier-wise method, the system can pinpoint crucial stress points, including spikes in energy usage or device failures, and then prioritize managing available resources accordingly [15].

The RM process uses machine learning (ML) methods and techniques in smart cities. ML methods are mainly used to reduce the computational cost level in the RM process [16, 17]. The examined data produces optimal information for the RM process. The dataset analyzed minimizes the management process's latency and energy consumption ratio [18, 19]. This article introduces CRM using Leveled Reinforcement Training (LRT) to address resource constraint issues due to flexible device features. LRT helps smart city resource management adapt to demand and energy consumption variations. This method proactively plans resources in response to energy availability and real-time device manipulation, reduces waiting times, and increases system efficiency, making it appropriate for the recommended system.

The paper’s novelty is that this article discusses smart city resource optimization utilizing CRM based on leveled Reinforcement Training. Our method dynamically distributes resources based on service demand and device performance to identify energy-intensive periods and prices. CRM optimizes IoT-based smart city services by monitoring energy usage and device failures. CRM is faster and eliminates resource allocation errors than previous methods.

Therefore, the research contributions are as follows:

- Design of a CRM method aiding less complex constraint mitigation for ease of sharing and allocation.
- The modification and implication of interrupt considered application services for SC environments.
- A comparative study will be conducted to validate the method’s performance using allocation, constraint, and time-based metrics.

This diagram shows one possible smart city resource management system based on reinforcement learning. A Smart City (SC) 's fundamental considerations are resource management and allocation.

2. RELATED WORKS

Researchers in [20] created AACF, an application-centric framework for IoT powered smart city applications. AACF prioritizes user-friendly and adaptable IoT architectures, emphasizing robust quality of experience in real-time smart city applications. The method focuses on the flexible distribution of application services, considering both application state and longevity. The proposed method enhanced the overall user experience in smart city applications. Alam et al. [21] devised an access control system for IoT devices in smart cities using blockchain and big data. Su et al. [22] suggested that smart cities use cloud computing and the IoT for information processing. The method provides helpful insights for future smart city development in the same context. To ensure uninterrupted smart services by balancing energy trade-offs between utility cost and consumption, Xiaoyi et al. [23] proposed a multi-objective distributed dispatching algorithm (MODDA) for efficient green energy management in IoT-driven smart cities. Liu et al. [24] proposed a method for efficiently allocating edge computing resources in IoT-based smart cities (MJOM). Zhang et al. [25] created an IoT green energy system for smart cities. The suggested method plans and assesses the development of smart power systems, considering various on-site and off-site resources. Zhong et al. [26] proposed a method for environmentally friendly smart cities using Green IoT. CRM with LRT is a better and faster alternative to conventional resource management methods in smart cities. This method improves the system's overall speed and scalability, making it better manage the intricate IoT settings in smart cities today. Fawzy et al. [27] introduced TPRUDF, a data-fusion framework for efficient resource utilization in smart environments leveraging the IoT. This method demonstrates a reduction in energy consumption and an improvement in throughput.

3. CONCERTED RESOURCE MANAGEMENT (CRM) USING THE LEVELED REINFORCEMENT TRAINING (LRT)

Real-world applications of the IoT include smart city traffic management systems that employ LRT to distribute resources optimally. Interconnected sensors and cameras monitor traffic. LRT examines energy consumption, traffic patterns, and traffic signal performance when detecting congestion. The preliminary step is identifying the cost complex derived above expressed in Equation (1).

$$\alpha = \frac{1}{d_n} + \sum_{s_c}^{s_v}(i_t * d_0) + \left(\frac{\left(\frac{i_t * s_c}{s_v}\right)}{\sum \frac{1}{d_n}}\right) * \left(\frac{i_t + d_0/s_v}{d_n * s_c}\right) + \left[\left(\prod_{s_c}(d_0 * s_v)\right) * (s_c + i_t)\right] * \left\{\left((d_0 + \cdots + d_n) * \left(i_t + \frac{s_c}{s_v}\right)\right)\right\}. \quad (1)$$

Calculating the complicated cost of resource management in a smart city setting. The number of devices $\{d_0,\ldots d_n\}$. and services s_v, s_c that are engaged in distributing resources is one of the factors included in the calculation. The cost complex in IoT provides reliable smart city device sharing, and it is expressed as $[(\prod_{s_c}(d_0{*}s_v))*(s_c+i_t)]$. IoT is initialized for the service-oriented computation in the environment study. The scope of this is to forward the resources to the requested devices in IoT, for this cost complexity is identified.

$$\theta = \prod_{s_c}(d_0 + e_y) * c_r + \left\{ \left(\frac{\frac{i_t + s_c}{\sum_{d_0}^{d_n}(s_v + w')}}{e_y / g'} \right) \right\} * [(e_y - g') + (i_t + w')] + \int_{d_0}^{d_n} (c_r(e_y) + v') - \left(\frac{v'}{w' + d_0} \right) \tag{2}$$

The high energy-consuming resources are observed, and the intervals are detected above (2). They depict an Equation (3) that deals with device failure inside an IoT framework and distributes resources to devices according to their requirements. Before allocating resources, the system examines energy use and service demands to determine the most intricate cost possibilities. Identifying resources with high energy consumption and closely tracking them at certain points can improve the efficiency and reliability of smart city applications. Equation (3) detects the IoT services handling where the device management forwards the resources, which is θ. The symbol represents this management function θ, and the highlighted Equation (3) is concerned with identifying IoT services that handle resource allocation via device management. It is essential for the effective administration of IoT services and optimizing resource sharing by deciding when devices send resources. The equation considers important variables such as reaction time and device failures to determine the period and cash needed for resource sharing. Improved service delivery in smart cities, managing resource limits, and navigating complicated IoT settings rely on the following equation.

$$\mu = \begin{cases} \left(s_v * s_c / w' + d_0 \right) + (o_c - q_e) - \left(v' * \frac{1}{d_n} \right), \in r_m \\ \prod_{i_t}(e_y - g') + q_e(s_v) - w' + \sum_{v'}(e_y - c_r), \in d_f \end{cases} \tag{3}$$

The complexity occurring intervals are decided as presented in Fig. 1. This complexity is considered based on device cost (replacement and functional) and energy drain (for g') such that w' is nevertheless defaced (Fig. 1). Here, the processing step relies on the response time to analyse a particular method where concurrent detection is required. The second derivation states device failure, where resource management defines the constraints of energy devices. Energy-based device management provides artificial intelligence in a better manner. By stating the same device request for the identical resource, then, the device failure occurs, and it is $\prod_{i_t}(e_y\text{-}g') + q_e(s_v)$. Based on this approach, the device failure is addressed in this derivation.

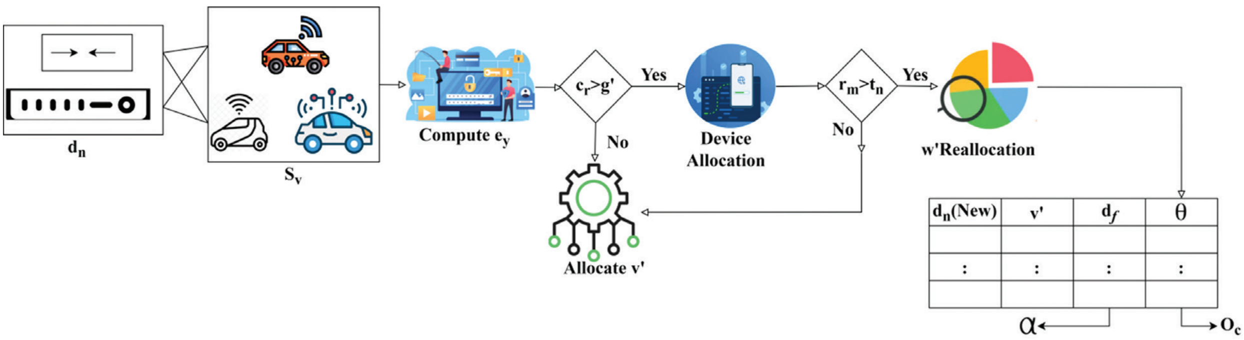


Fig. 1. Complexity Occurring Sharing Intervals

$$p_a = (r_a + d_0) * \left(\frac{\prod_{s_v}^{w'}(v' - r_m) + c_r}{\frac{e_y / m_d}{a_p + i_t}} \right) + (a_p * d_f) + \sum_\theta [(v' - i_t) * (q_e + m_d)] + \sum \left[(r_m - d_f) + \frac{w'}{s_v} \right] \tag{4}$$

Equation (4) expresses that incorporating energy and devices for the smart city application is based on demand. Equation (4) combines smart city energy and device management for distributing resources based on demand. The real-time device failure handling and resource distribution are effective. This dynamic method boosts smart city reliability.

3.1. REINFORCEMENT LEARNING FOR CONSTRAINT ANALYSIS

It is a decision-making approach suitable for the specific situation and improves the reward function.

Three phases of computation are pragmatic in this learning; the first is the input, which is the initial stage of acquiring the request from the device.

In Equation (5), energy depletion is derived. Equation (5) calculates smart city energy depletion using resource allocation and forwarding time intervals. Demand and energy use are measured. This equation optimizes resource management in smart cities by reducing energy waste.

$$l_p = \left(\frac{\prod_{d_0}(i_t + m_d) / r_a}{w' + m_d / \sum_{r_a}(e_y - d_f)} \right) * [(o_c + a_p) + (v' + \mu)] * \left(\frac{r_a + d_0 / q_e}{\prod_{g'}(r_m + d_f)} \right) + \{[(c_r + e_y) + (w' + d_0)] * (\textstyle\sum_{m_d} o_c + v') + \alpha\}. \tag{5}$$

The level-based constraint representation is given below. The reinforcement learning states the active response where the device-based service is defined for the demand. The allocation failure is equated below.

$$a_f = (q_e + s_v) + d_0 * \left(\frac{\theta + c_r}{\Sigma_{w'}(l_p + s_c)}\right) - v' \tag{6}$$

In Equation (6), the allocation failure is examined. Allocation failure (a_f) in Equation (6) happens when demand exceeds supply, and a system cannot allocate resources to a device. Resource depletion (v') is added to service and queuing times. The resource management system targets this failure to deliver continuous service at desired intervals. In Algorithm 1, the pseudo-code for a_f is presented.

Algorithm 1 a_f Detection

Step 1: for all d_n do {
Step 2: Allocate s_v and μ
Step 3: compute p_a using equation (2b)
Step 4: if $\{p_a>\mu/s_v\}$ then
Step 5: Estimate l_p using equation (3)
Step 6: while $\{d_f != 1\}$ do
Step 7: Assign s_v to d_n $\forall(d_n*\mu)=r_m/v_e$
Step 8: compute c_r and g'
Step 9: if $\{c_r<g'\}$ then
Step 10: Assign a new d_n in μ and Repeat from Step 5 until $r_m/q_e=1$
Step 11: else if $\{c_r>g'\}$ then
Step 12: Repeat from Step 4
Step 13: Update q_e
Step 14: $a_f=(q_e+s_v)$
Step 15: End if
Step 16: End while
Step 17: $l_p=l_v(e_y)$
Step 18: End if
Step 19: $a_f=a_f-v$^'
Step 20: End for

Hence, the allocation failure is observed in this part, and the session-based interruption is derived in the section below (Fig. 2):

$$\mu(i_p) = |w' + q_e| * \{a_f - l_p\} + \left(\frac{a_p * g'}{\alpha * o_c}\right) - i_t. \tag{7}$$

As found in Equation (7), the allocation failure is observed, and session-based interruption has been examined. In Equation (7), session-based interruption $\mu(i_p)$ refers to service disruption caused by allocation failure a_f. When session resources are misallocated, this disturbance occurs. The Equation considers interruption time (i_p) and resource depletion parameters to evaluate service effects in IoT applications, especially smart cities. Thus, incorporating the service along with the energy and device faces the interruption of services to avoid this identification of resource allocation failure and active resource is equated below.

$$\alpha(r_a) = \left[\left(a_p + (i_p * p_a)\right)\right] * \left(\frac{d_0 + w'}{\Sigma(v' + m_d)}\right) - o_c. \tag{8}$$

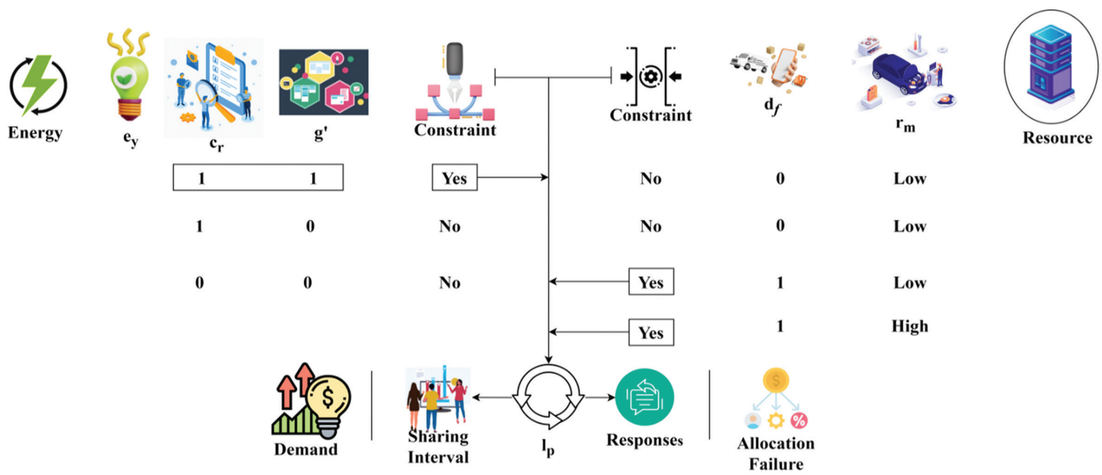


Fig. 2. Level-based Constraint Representation

As deliberated in Equation (8), resource allocation failure and active resources have been described. Consider active resource equations and allocation failure to maximize resource distribution in dynamic situations like smart cities. They assist proactive management in sustaining service operations by predicting resource shortages and breakdowns. These equations effectively detect active devices and energy use to scale systems and reduce waste. The training is introduced in this learning method, where demands are identified based on the independent or concerted resource allocation. The derivation is expressed as follows.

$$t_n = \prod_{d_n}(s_v * m_d) + \left(\frac{a_f + l_p/r_a}{w' + i_t}\right) * (i_p + c_r) - p_a \tag{9}$$

As examined in Equation (9), concerted resource allocation has been explored. The effective and coordinated distribution of resources across various devices is crucial in dynamic contexts, such as smart cities, and this can only be achieved by concerted resource allocation. The system optimizes energy utilization, minimizes delays, and eliminates competition for scarce resources by aligning allocation with real-time demand. Overall, this method prevents allocation errors and enhances system performance and scalability. This interconnection of levels is discussed and equated to the below equation.

$$l_v(e_y) = [(w' + d_0) + (r_a - v')] * \left(\frac{(d_f + a_p)}{\sum_{m_d} t_n}\right) + (o_c - i_p) \quad (10)$$

As discussed in Equation (10), interconnection levels are explained. A high degree of connectivity is crucial for the smooth operation of the many IoT devices in smart cities. With their help, extending resource management becomes simpler, allowing the system to react quicker to changes in real-time demand.

The level of interconnection is based on this approach where the cost complex is involved for the service forwarding for the allocation failure, and it is described as l_v. In this case, an interruption of service is associated with device failure, where training is involved for the processing levels. The learning process for $l_v(e_y)$ is illustrated in Fig. 3.

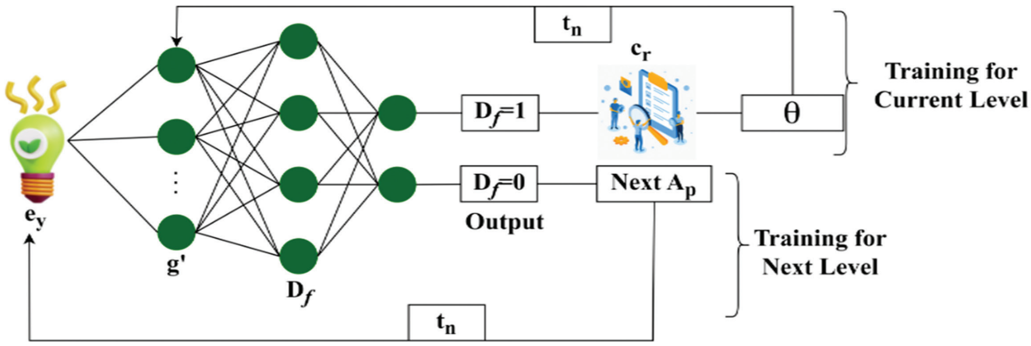


Fig. 3. $l_v(e_y)$ Learning Process

The equation below states the device failure:

$$l_v(d_f) = \prod_{d_0}(a_p * w') + \left(\frac{l_p * i_p}{m_d}\right) - i_t(d_0) \quad (11)$$

As explored in Equation (10), device failure was calculated. After this, resource allocation is performed, and training levels are observed in the equations below.

$$l_v(r_a) = \left(\frac{\alpha + l_p}{a_f}\right) * (s_v - \theta) + \left(\frac{{}^{o_c}/_{g'}}{t_n + a_f}\right) * (d_f + t_n) \quad (12)$$

$$t_n(l_v) = (e_y + d_f + r_a) + \left(\theta * \frac{\mu}{v'}\right) * r_m. \quad (13)$$

Training level observation was derived as deliberated in Equation (12) and Equation (13) performed for resource allocation. Before allocating resources, it is essential to conduct training-level observations to identify and respond to evolving needs and performance measures.

This procedure makes the system more adaptable to new circumstances, reduces delays, and guarantees efficient use of resources in smart city settings. The learning process for $l_v(r_a)$ is illustrated in Fig. 4.

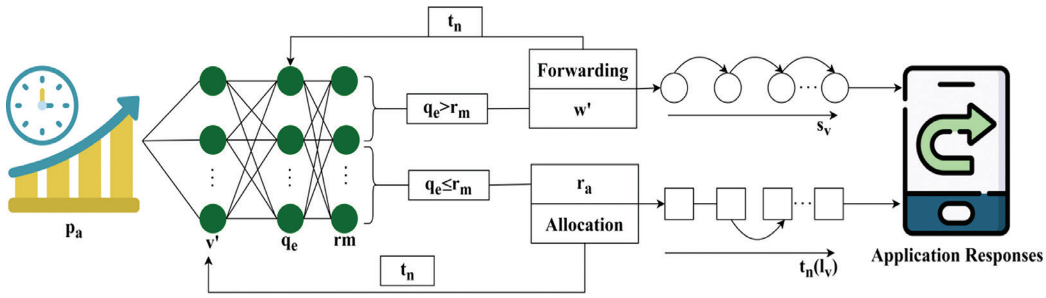


Fig. 4. Learning Process for $l_v(r_a)$

The learning process connected to levels is presented as a pseudo-code in Algorithm 2.

Algorithm 2 Learning Process for Connected Levels

Input: p_a, m_d

Step 1: for all p_a m_d ∀ d_n do

Step 2: compute $\mu(ip)$ using equation (4)
and perform α (r_a)

Step 3: if $\{\alpha(r_a)!=(a_p*r_a/(t_n(l_v)))\}$ then

Step 4: Estimate o using equation (1b)
and a_f using equation (3b)

Step 5: if $\{t_n=e_y\}$ then

Step 6: Allocate m_d with d_n such that $(d_f+a_p)=\sum t_n$

Step 7: Update $(e_y, d_f)|(c_r, o)$ ∀ d_f=1 in Step 6

Step 8: Perform t_n for o and ap∈(c_r,o)
until $\alpha(r_a)=a_p*r_a/(t_n(l_v))$

Step 9: Estimate r_m ∀ p_a under q_e

Step 10: if $\{r_m \geq [q_e - t_n(l_v)]\}$ then

Step 11: Include $l_v(d_f)$ ∀ i_t (d_o), d_o ∈ m_d=Allocation

Step 12: Update $(p_a=p_a$-v'/$(t_n$ $(l_v)))$

Step 13: else

Step 14: l_v $(r_a)=(d_f+t_n)$ until $t_n=(e_y+1)$ ∈ d_n

Step 15: Goto Step 3 for all p_a=0 in $t_n(l_v)$
Step 16: End if
Step 17: Perform l_p for l_v (e_y) and l_v (r_a) until r_m=0
Step 18: End if
Step 19: End if
Step 20: End for

4. PERFORMANCE ASSESSMENT

This article discusses layered reinforcement learning and its potential applications in coordinated resource management to ease SC service constraints. With limited energy and resource allocation, this concept took the big picture into account while managing SC resources. The approach first uncovered the intricate processing limitations based on energy to guarantee high device availability. Device availability and resource allocation concerns are associated with excessive energy usage and depletion. This section describes the performance of the proposed method through a comparative study. This section discusses comparative analysis based on resource allocation rate, active device detection, constraint mitigation, allocation failure, and allocation wait time. The number of devices (20 to 240) and the sharing interval time (30s to 300s) are the X-variants considered. The proposed method is compared with MODDA [23], MJOM [24], and TPRUDF [27] methods discussed in the related works section.

Table 1 shows the experimental setup.

Table 1. Experimental Setup

Parameters	Description
Processor	Intel core i7;3.5
Memory	16GB RAM
Storage	1 TB SSD
IoT Devices	Raspberry Pi 4
Sensors	Temperature, humidity, energy sensors
Network	5G
Operating System	Ubuntu 20.04
Programming Language	Python 3.8
Machine Learning Library	TensorFlow
Reinforcement Learning	OpenAI, stable baselines 3
Simulation environment	MATLAB
Monitoring Tool	Grafana
Blockchain Integration	Hyperledger Fabric

Dataset Description: Turning the city into a "smart city" is the government's goal. The plan is to turn it into an intelligent and digital city to make services more efficient for the people. The administration is dealing with traffic as one of its challenges. Data scientists are contributing to improved municipal traffic management and future infrastructure development. So, that we can plan accordingly for the next four months' worth of traffic at each of these intersections, the traffic data from many periods since the sensors at each junction were taking readings at various times. Some intersections have supplied little or incomplete data, which further complicates matters and necessitates care in developing future predictions. According to data collected over the last twenty months, the government depends on reliable traffic forecasts for the next four months. A bigger change is coming to the city, and machine learning algorithms will be the cornerstone of it. It will become smart and intelligent.

4.1. RESOURCE ALLOCATION RATE COMPARISON

The resource allocation rate for the proposed work increases for the smart city where the demands are satisfied by the number of devices and constraints used to provide energy based on the levels of observation. The identification of the cost-complex is developed for the consuming sharing intervals, and it is represented as $((((i_t*s_c)/s_v))/(\sum 1/d_n))*(((i_t+d_0)/s_v)/(d_n*s_c))$. This processing step allocates resource management to the appropriate device based on reinforcement learning. Here, the high energy-consuming sharing intervals detect device failure in smart city applications. In this case, the incorporation of energy and device is examined for the interrupt sessions, and it is equated in Equation (2a). Device failure provides energy depletion and allocation failure, whereas the cost complex provides the response time analysis. The incorporation rate in this work increases where the energy drop is detected, and based on this, resource allocation is performed. The device failure is identified for the levels of observation where the output is trained in this session using reinforcement learning as shown in Fig. 5 (a) and (b).

Fig. 5 (a) and (b) show that the proposed CRM method with LRT regulates the resource allocation rate by adapting the distribution of resources in real-time to patterns in energy consumption and device demand. It discovers energy-consuming periods and optimizes the sharing time by assessing service response times and likely device faults. This approach minimizes power consumption during sharing times, adjusts to the requirements of various devices, continuously educates the system to deal with allocation mistakes, and allows for the efficient and timely allocation of resources to running devices.

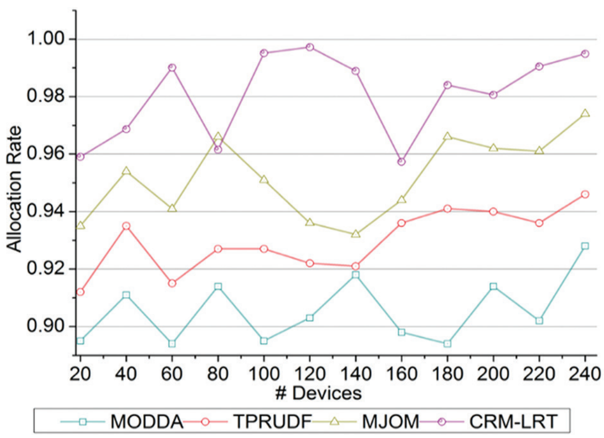


Fig. 5 (a). Resource Allocation Rate for #Devices

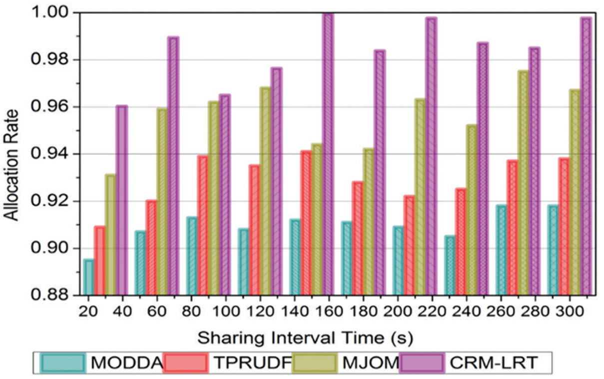


Fig. 5 (b). Resource Allocation Rate for Sharing Interval Time(s)

4.2. ACTIVE DEVICE DETECTION COMPARISON

In Fig. 6 (a) and (b), active device detection improved for the varying energy drop and resource allocation. Based on the request, the response time is observed and forwards the resource to the appropriate device on time. It states the alternative allocation where it provides the consuming and sharing intervals. Energy and devices are incorporated to state smart applications and provide resource management. The interruption is detected based on the forwarding interval and allocates the resource to the device. In this case, an active device is detected to forward the resource in this process, and the failure is reduced.

Fig. 6 (a) and (b) express how tracking power utilization and service response with the LRT during sharing times enhances active device recognition. It accurately detects active devices using real-time performance data, including energy utilization and demand fulfillment. The technique ensures that only active session participants acquire resources by monitoring and reacting to each device's operating state. With dynamic detection, smart city resources may be dispersed more effectively between devices.

4.3. CONSTRAINT MITIGATION COMPARISON

The constraint mitigation is enhanced in this work; if device failure is detected, then the computation process is developed to detect whether the device is active or not. Resource management is developed in this IoT based on device activation, which provides service-based demands. The device failure occurs and observes whether any resource allocation is occurring. In terms of this method, the levels of observation take place for the constraint mitigation, and it is equated as $d_0*((\theta+c_r)/(\sum(w')(l_p+s_c)))$ and shown in Fig. 7 (a) and (b).

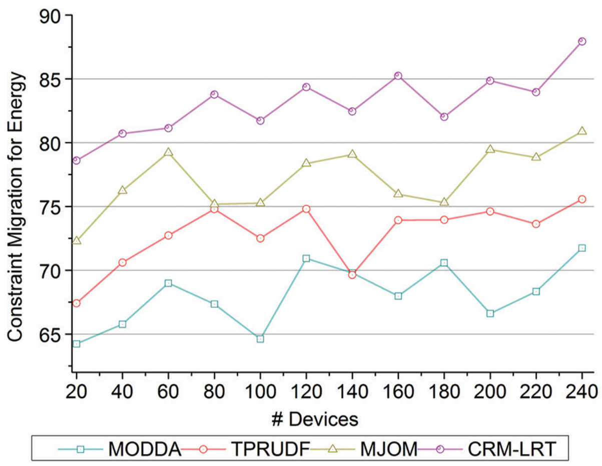


Fig. 6 (a). Active Device Detection for #Devices

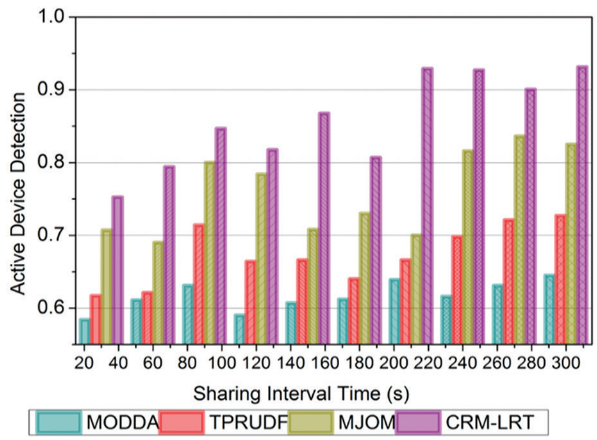
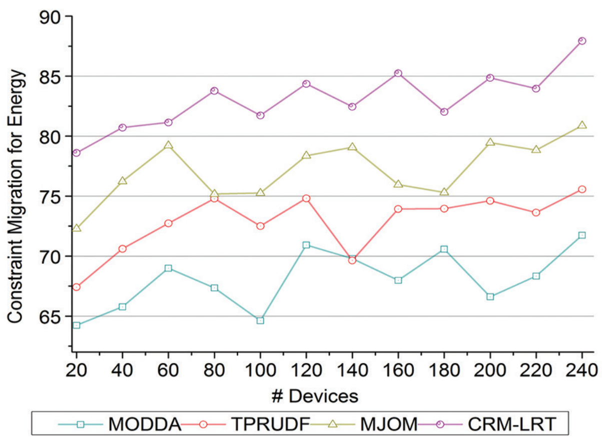


Fig. 6 (b). Active Device Detection for Sharing Interval Time(s)



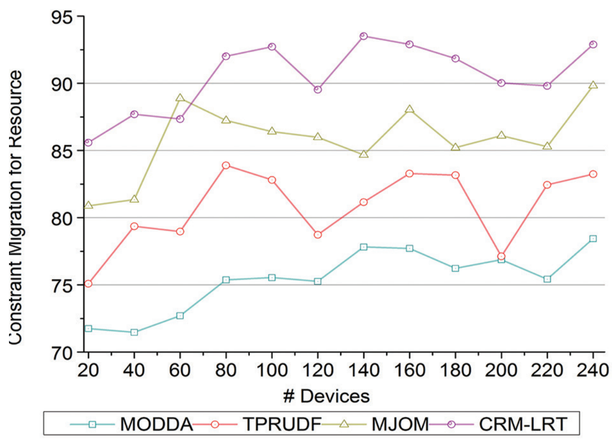


Fig. 7 (a). Constraint Mitigation for Energy and Resources against #Devices

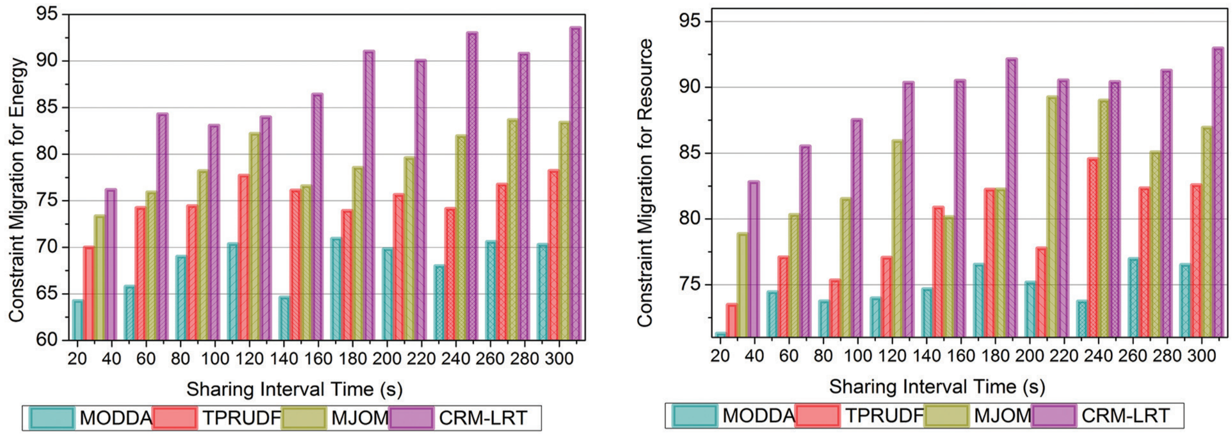


Fig 7 (b). Constraint Mitigation for Energy and Resources against Sharing Interval Time(s)

Fig. 7 (a) and (b) explain how to reduce limitations caused by various devices and sharing intervals. The suggested solution employs LRT to adapt resource management on the flight.

4.4. ALLOCATION FAILURE COMPARISON

The allocation failure is due to the device failure, and a response is given to the identification method. This processing case was developed for resource allocation and is based on the levels of observation. From these levels, the training is processed and meets the demands of examination in resource management. From this management system, the n-number of devices is associated with energy consumption, and depletion is detected in this reinforcement learning. The learning estimates and delivers the smart city application for the high energy in this processing step. The active resource is forwarded to the end device that requests the resource, and it is termed concerted resource allocation or independent resource management. The level of interconnection is formulated as $\{a_f\text{-}l_p\}+((a_p*g')/(\alpha*o_c))\text{-}i_t$, from the processing intervals and examines the device failure Fig. 8 (a) and (b).

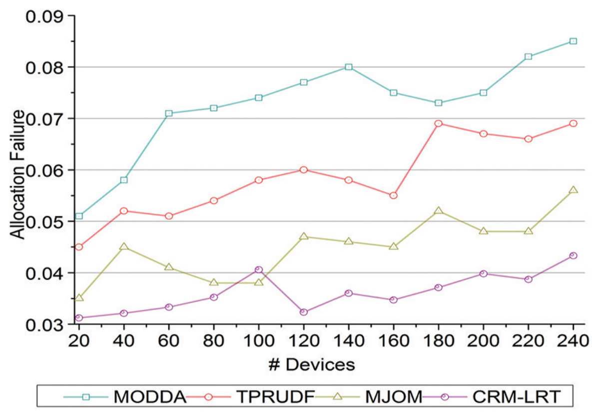


Fig. 8 (a). Allocation Failure for #Devices

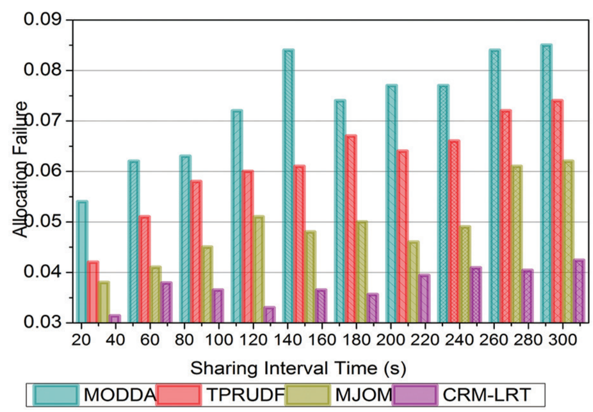


Fig. 8 (b). Allocation Failure for Sharing Interval Time

Fig. 8 (a) and (b) explored the proposed LRT to solve allocation failure by anticipating and controlling resource requirements across devices and sharing durations. In the event of a failure caused by inadequate resources or high energy use, it identifies this immediately and modifies allocations appropriately.

4.5. ALLOCATION WAIT TIME COMPARISON

In Fig. 9 (a) and (b), allocation wait time is reduced in this process, where energy depletion and allocation failure occur. Here, demand estimation is performed to detect the interruption of the session. The levels of interconnection are processed for resource allocation where the satisfaction of the demands. The service to the smart city is deployed in the IoT device and detects depletion and allocation failure. In this processing step, the incorporative state is used to identify the consuming sharing intervals, and it is represented as [(a_p+(i_p*p_a))]. The training in the levels of interconnection is used to state the independent or concerted resource allocation in which the demands are satisfied. This allocation wait time is decreased if the active resource is forwarded to the requested device. Incorporating energy consumption provides an alternative allocation where demands are identified based on this computation process. Here, it states the device failure and reduces the waiting time. The wait time is observed, and the smart city application is provided with better resource forwarding.

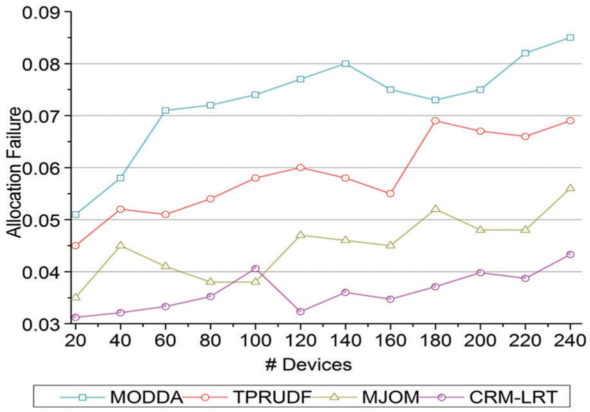


Fig. 9 (a). Allocation Failure for # Devices

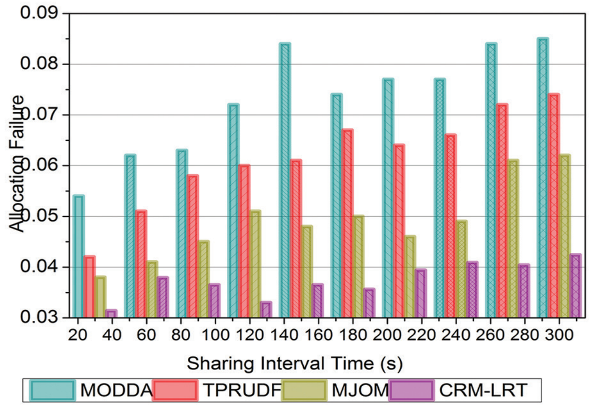


Fig. 9 (b). Allocation Failure Sharing Interval Time

Fig. 9 (a) and (b) deliberated that the suggested technique uses patterns of device activity and sharing intervals to forecast resource demands and allocate them using LRT, reducing allocation wait time.

Table 2. Execution Time for Different Devices

	MODDA	TPRUDF	MJOM	CRM-LRT
20	0.418	0.324	0.554	0.326
40	0.581	0.538	0.464	0.526
60	0.524	0.479	0.656	0.313
80	0.289	0.335	0.614	0.346
100	0.656	0.417	0.452	0.323
120	0.549	0.584	0.413	0.419
140	0.442	0.611	0.804	0.333
160	0.534	0.643	0.721	0.526
180	0.554	0.698	0.821	0.418
200	0.587	0.624	0.522	0.581
220	0.602	0.587	0.513	0.524
240	0.624	0.673	0.818	0.326

The Leveled Reinforcement Training (LRT) approach, which the suggested CRM method employs to optimize resource allocation in response to changing service needs and device statuses in real-time, improves execution time. In contrast to more conventional approaches, CRM anticipates and resolves device failures and high-cost, energy-intensive sharing periods. This proactive method expedites service execution by minimizing delays caused by resource restrictions.

Table 3. Memory Usage for Different Devices

Number of Devices	MODDA	TPRUDF	MJOM	CRM-LRT
20	0.503	0.718	0.599	0.478
40	0.51	0.704	0.587	0.464
60	0.529	0.69	0.566	0.45
80	0.532	0.68	0.559	0.44
100	0.548	0.678	0.543	0.438
120	0.555	0.658	0.532	0.428
140	0.562	0.647	0.522	0.417
160	0.577	0.639	0.50	0.409
180	0.587	0.625	0.491	0.395
200	0.593	0.605	0.481	0.386
220	0.553	0.625	0.491	0.395
240	0.544	0.605	0.481	0.386

The suggested CRM technique decreases memory utilization by enhancing resource management via reinforcement learning and selective allocation. Instead of continuously storing and processing data for all smart city devices and services, the CRM technique prioritizes data points with significant energy and cost consumption. By recognizing and handling these crucial periods, the system may reduce the processing and storage of unnecessary or redundant data. Leveled Reinforcement Training (LRT) also allows the approach to adjust resource allocation in real-time in reaction to demand, which might reduce the need to retain vast volumes of past data. Smart city applications benefit greatly from the system's enhanced speed and efficiency through memory management and selective data processing.

Table 4. Performance of the Study

Number of Devices	Energy Consumption	Resource Management	Waste Management	Decision Making
20	80.2	80.9	70.8	87.6
40	81.3	81.6	71.7	88.7
60	84.6	82.5	72.4	89.1
80	86.9	83.2	73.5	90.3
100	88.5	84.4	74.7	87.7
120	90.4	85.1	75.2	88.9
140	91.6	86.2	76.3	89.5
160	93.2	87	77.6	90.9
180	94.4	88.1	86.8	91.2
200	95.7	88.9	86.8	93.1
220	96.4	88.1	88.5	94.4
240	96.7	90.3	89.8	95.7

Table 4 shows the performance of the proposed study. Maintaining unrestricted service flow in a smart city (SC) setting is achieved by carefully managing resources, including energy consumption and device performance. These services are often interrupted when resources are scarce, leading to increased energy consumption, slowed resource allocation, and broken devices. Attempts to control resources in SC systems using static thresholds or reactive strategies have failed miserably. An intermediate-sized smart city employs CRM-LRT to address these issues. Intelligent transportation, smart waste management, and public safety applications are just a few of the urban systems enabled by IoT. These systems have difficulties allocating resources because of fluctuations in device performance and energy consumption patterns, especially during high service demand. Issues with managing resources in the city's smart services were successfully managed by the CRM solution that was based on LRT.

Implementation of CRM-LRT:

I. Following these phases, the CRM approach is implemented throughout several city sectors:

II. The city's IoT sensors track energy use, response times to service requests, and device malfunctions. This data is analyzed using AI algorithms to comprehend cost-complex intervals and energy consumption patterns.

III. LRT uses the data to determine when resources are most needed during critical service periods. The training is centered on allocating resources from energy depletion stages to device activity degrees. The predictions of the LRT model are used to pick devices with the best combination of energy efficiency and availability.

IV. The LRT method employs a coordinated resource allocation, considering energy needs and device preparedness. By working together, we can decrease allocation failures and energy waste by letting the system dynamically assign resources based on real-time requests.

V. Management of Interruptions and Failures: The CRM system can identify instances when services are interrupted due to problems with devices or the distribution of resources. They were redistributing resources to devices actively using less energy, which guarantees that services will continue uninterrupted.

The approach enhanced system performance, reduced energy consumption, and expanded scalability of SC applications by dynamically distributing resources according to real-time data. Collaborative decision-making for smart city resource management is shown in this scenario.

The performance assessment in Section 4 shows how effectively the suggested CRM system works, but a complete investigation of device counts and sharing intervals may reveal its true scalability. Given the growing number of connected devices, comparing results across different IoT device sizes is one approach to evaluate the strategy. This investigation may reveal the CRM technique's resilience under more rigorous resource allocation and energy consumption testing. Testing the technique with different sharing intervals will reveal its flexibility to shift demand and performance in real-time, dynamic environments. Comparing the two shows the strategy's practical scalability; this may help determine its feasibility for larger smart city networks.

5. CONCLUSION

This article introduced and discussed the performance of concerted resource management using leveled reinforcement learning for SC service constraint mitigation. This proposal considered the energy and resource allocation constraints that are chained together in retarding SC resource management. First, the method identified the complex processing constraints based on energy to ensure high device availability. Computed with energy depletion and high energy utilization features, device availability and resource allocation failures are connected. This connectivity provides shared or re-allocated/failure-less resource management for different SC application services. The shared sessions are validated using complex constraints that remain unaddressed post the allocation. The concerted reinforcement learning was used to identify the allocation failures using the connectivity between different levels in the resource management process. Both constraints rely on different training inputs per energy and resource allocation conditions to improve the allocation rate. Under the varying devices, the following is observed: Improvements: 9.1% (Allocation Rate), 10% (Device Detection), 11.88% (Constraint Mitigation—Energy), 9.06% (Constraint Mitigation—Resource Allocation); Reduced: 8.01% (Allocation Failure), 9.64% (Waiting Time). Potentially impassable difficulties with the proposed method include very dense IoT network scalability and unexpected, rapid shifts in resource requirements. The scalability of the CRM technique might be studied further by testing it in various real-world IoT scenarios and incorporating deep reinforcement learning for increased adaptation. Improved performance optimization and resilience in ever-expanding smart city applications can only be achieved using edge computing and beefing up security measures.

6. ACKNOWLEDGMENT

The author gratefully acknowledges the approval and the support of this research study by the grant no CSAT-2023-12-2159 from the Deanship of Scientific Research at Northern Border University, Arar, K.S.A

7. REFERENCES

[1] T. Shafique, R. Gantassi, A. H. Soliman, A. Amjad, Z. Q. Hui, Y. Choi, "A review of Energy Hole mitigating techniques in multi-hop many to one communication and its significance in IoT oriented Smart City infrastructure", IEEE Access, Vol. 11, 2023, pp. 12345-12367.

[2] V. M. Kuthadi, R. Selvaraj, S. Baskar, P. M. Shakeel, A. Ranjan, "Optimized energy management model on data distributing framework of wireless sensor network in IoT system", Wireless Personal Communications, Vol. 127, No. 2, 2022, pp. 1377-1403.

[3] I. Hussain, A. Elomri, L. Kerbache, A. El Omri, "Smart City Solutions: Comparative Analysis of Waste Management Models in IoT-Enabled Environments Using Multiagent Simulation", Sustainable Cities and Society, Vol. 103, 2024, pp. 1-14.

[4] A. Nauman, N. Alruwais, E. Alabdulkreem, N. Nemri, N. O. Aljehane, A. K. Dutta, W. U. Khan, "Empowering smart cities: High-altitude platforms based Mobile Edge Computing and Wireless Power Transfer for efficient IoT data processing", Internet of Things, Vol. 22, 2023, pp. 1-26.

[5] R. Selvaraj, V. M. Kuthadi, S. Baskar, P. M. Shakeel, A. Ranjan, "Creating security modelling framework analysing in internet of things using EC-GSM-IoT", Arabian Journal for Science and Engineering, Vol. 48, 2023, pp. 2607-2620.

[6] F. Beştepe, S. Ö. Yildirim, "Acceptance of IoT-based and sustainability-oriented smart city services: A mixed methods study", Sustainable Cities and Society, Vol. 80, 2022, p. 103794.

[7] J. Duque, "The IoT to Smart Cities - A design science research approach", Procedia Computer Science, Vol. 219, 2023, pp. 279-285.

[8] C. Li, "Development of IoT Smart Cities and Optimization of English Education Systems Based on 5G Networks", Soft Computing, Vol. 27, No. 12, 2023, pp. 1-19.

[9] M. Hosseinzadeh, A. Hemmati, A. M. Rahmani, "Clustering for smart cities in the internet of things: a review", Cluster Computing, Vol. 25, No. 6, 2022, pp. 4097-4127.

[10] A. Ullah, S. M. Anwar, J. Li, L. Nadeem, T. Mahmood, A. Rehman, T. Saba, "Smart cities: The role of Internet of Things and machine learning in realizing a data-centric smart environment", Complex and Intelligent Systems, Vol. 10, No. 1, 2023, pp. 1607-1637.

[11] K. Peng, H. Huang, P. Liu, X. Xu, V. C. Leung, "Joint optimization of energy conservation and privacy preservation for intelligent task offloading in MEC-enabled smart cities", IEEE Transactions on Green Communications and Networking, Vol. 6, No. 3, 2022, pp. 1671-1682.

[12] L. A. Ajao, S. T. Apeh, "Secure edge computing vulnerabilities in smart cities sustainability using Petri net and genetic algorithm-based reinforcement learning", Intelligent Systems with Applications, Vol. 18, 2023, p. 200216.

[13] A. Ullah, S. M. Anwar, J. Li, L. Nadeem, T. Mahmood, A. Rehman, T. Saba, "Smart cities: The role of Internet of Things and machine learning in realizing a data-centric smart environment", Complex & Intelligent Systems, Vol. 10, No. 1, 2024, pp. 1607-1637.

[14] X. Li, D. Zhang, Y. Zheng, W. Hong, W. Wang, J. Xia, Z. Lv, "Evolutionary computation-based machine learning for smart city high-dimensional big data analytics", Applied Soft Computing, Vol. 133, 2023, p. 109955.

[15] A. Qadeer, M. J. Lee, "HRL-edge-cloud: Multi-resource allocation in edge-cloud based smart-streetscape system using heuristic reinforcement learning", Information Systems Frontiers, Vol. 26, 2023, pp. 1399-1415.

[16] Y. Kim, B. C. Jung, Y. Song, "Online learning for joint energy harvesting and information decoding optimization in IoT-enabled smart city", IEEE Internet of Things Journal, Vol. 10, No. 12, 2023, pp. 10675-10686.

[17] Z. Huang, G. Jin, "Navigating urban day-ahead energy management considering climate change toward using IoT-enabled machine learning technique: Toward future sustainable urban", Sustainable Cities and Society, Vol. 101, 2024, p. 105162.

[18] O. M. Prabowo, E. Mulyana, I. G. B. B. Nugraha, S. H. Supangkat, "Cognitive city platform as digital public infrastructure for developing a smart, sustainable and resilient city in Indonesia", IEEE Access, Vol. 11, 2023, pp. 120157-120178.

[19] Y. Y. Liu, Y. Zhang, Y. Wu, M. Feng, "Healthcare and fitness services: A comprehensive assessment of blockchain, IoT, and edge computing in smart cities", Journal of Grid Computing, Vol. 21, No. 4, 2023, p. 82.

[20] Y. Cui, X. Song, J. Liu, K. Chen, G. Shi, J. Zhou, G. S. Tamizharasi, "AACF—Accessible application-centric framework for the Internet of Things backhauled smart city applications", IEEE Transactions on Network Science and Engineering, Vol. 9, No. 3, 2021, pp. 980-989.

[21] T. Alam, "Blockchain and Big Data-based access control for communication among IoT devices in smart cities", Wireless Personal Communications, Vol. 132, No. 1, 2023, pp. 433-456.

[22] P. Su, Y. Chen, M. Lu, "Smart city information processing under Internet of Things and cloud computing", The Journal of Supercomputing, Vol. 78, No. 3, 2022, pp. 3676-3695.

[23] Z. Xiaoyi, W. Dongling, Z. Yuming, K. B. Manokaran, A. B. Antony, "IoT driven framework based efficient green energy management in smart cities using multi-objective distributed dispatching algorithm", Environmental Impact Assessment Review, Vol. 88, 2021, p. 106567.

[24] Z. Liu, "A multi-joint optimization method for distributed edge computing resources in IoT-based smart cities", Journal of Grid Computing, Vol. 21, No. 4, 2023, pp. 1-11.

[25] X. Zhang, G. Manogaran, B. Muthu, "IoT enabled integrated system for green energy into smart cities", Sustainable Energy Technologies and Assessments, Vol. 46, 2021, p. 101208.

[26] Y. Zhong, Z. Qin, A. Alqhatani, A. S. M. Metwally, A. K. Dutta, J. J. Rodrigues, "Sustainable environmental design using green IoT with hybrid deep learning and building algorithm for smart city", Journal of Grid Computing, Vol. 21, No. 4, 2023, p. 72.

[27] D. Fawzy, S. M. Moussa, N. L. Badr, "An IoT-based resource utilization framework using data fusion for smart environments", Internet of Things, Vol. 21, 2023, pp. 1-24.