Hindawi
Journal of Sensors
Volume 2023, Article ID 9824038, 1 page
https://doi.org/10.1155/2023/9824038

*Retraction*

# Retracted: Smart Grid Security Based on Blockchain with Industrial Fault Detection Using Wireless Sensor Network and Deep Learning Techniques

**Journal of Sensors**

Received 12 December 2023; Accepted 12 December 2023; Published 13 December 2023

This article has been retracted by Hindawi, as publisher, following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of systematic manipulation of the publication and peer-review process. We cannot, therefore, vouch for the reliability or integrity of this article.

Please note that this notice is intended solely to alert readers that the peer-review process of this article has been compromised.

Wiley and Hindawi regret that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

## References

[1] M. Kandasamy, S. Anto, K. Baranitharan, R. Rastogi, G. Satwik, and A. Sampathkumar, “Smart Grid Security Based on Blockchain with Industrial Fault Detection Using Wireless Sensor Network and Deep Learning Techniques,” *Journal of Sensors*, vol. 2023, Article ID 3806121, 13 pages, 2023.

Hindawi
Journal of Sensors
Volume 2023, Article ID 3806121, 13 pages
https://doi.org/10.1155/2023/3806121

*Research Article*

# Smart Grid Security Based on Blockchain with Industrial Fault Detection Using Wireless Sensor Network and Deep Learning Techniques

**Manivel Kandasamy,[1] S. Anto,[2] K. Baranitharan,[3] Ravi Rastogi,[4] Gunda Satwik,[5] and A. Sampathkumar[6]**

[1] *UnitedWorld School of Computational Intelligence, Karnavati University, Gandhinagar, Gujarat 382422, India*
[2] *School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India*
[3] *Department of ECE, Alva's institute of Engineering & Technology, Mangalore, Karnataka, Affiliated to VTU, India*
[4] *Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India*
[5] *Department of Computer Science and Engineering, Graphic Era Deemed to Be University, Dehradun, India*
[6] *Department of Computer Science and Engineering, Dambi Dollo University, Dambi Dollo, Ethiopia*

Correspondence should be addressed to A. Sampathkumar; dr.sampathkumar@dadu.edu.et

Received 19 September 2022; Revised 12 December 2022; Accepted 20 April 2023; Published 9 May 2023

Academic Editor: C. Venkatesan

Low-cost monitoring and automation solutions for smart grids have been made viable by recent advancements in embedded systems and wireless sensor networks (W.S.N.s). A well-designed smart network of subsystems and metasystems known as a "smart grid" is aimed at enhancing the conventional power grid's efficiency and guaranteeing dependable energy delivery. A smart grid (S.G.) requires two-way communication between utility providers and end users in order to accomplish its aims. This research proposes a novel technique in enhancing the smart grid security and industry fault detection using a wireless sensor network with deep learning architectures. The smart grid network security has been enhanced using a blockchain-based smart grid node routing protocol with IoT module. The industrial analysis has been carried out based on monitoring for fault detection in a network using Q-learning-based transfer convolutional network. The experimental analysis has been carried out in terms of bit error rate, end-end delay, throughput rate, spectral efficiency, accuracy, M.A.P., and RMSE. The proposed technique attained bit error rate of 65%, end-end delay of 57%, throughput rate of 97%, spectral efficiency of 93%, accuracy of 95%, M.A.P. of 55%, and RMSE of 75%. This proposed paradigm is advantageous for the operation of smart grids for increased security and industrial fault detection across the network because security is the biggest barrier in smart grid implementation.

## 1. Introduction

Due to its portability, affordability, and ease of deployment, WSN is one of the best approaches for many real-time applications. Monitoring the area of interest, gathering data, and sending it to BS for postprocessing and analysis are the duties of the WSN [1]. Some WSN implementations make use of a lot of sensor nodes. Additionally, the battery life and memory of these wireless nodes are constrained. Therefore, in order to maximise the benefits of these WSNs, these WSN nodes must have a management system capable of controlling both their interactions with one another and with the access point. For instance, the Internet Engineering Team (IETF) established the ZigBee and 6LoWPAN protocols for common transmission over IEEE 802.15.4 [2] to allow administration of WSNs. These protocols allow for the usage of IEEE 802.15.4 in 2.4 GHz band and the support of brief transmissions by contemporary management systems. For instance, based on IP addresses on various tiers, 6LoWPAN IPv6 offers a connection between WSNs. The network architecture is also mapped using the 6LoWPAN Low Power and Loss Network (RPL) standard, and WSN connection is secured using the AES encryption technique [3]. These networks' dynamic topologies, however, will affect network routing tactics, delay, multilayer design, coverage, QoS, and fault detection. As part of the smart grid revolution, the electrical grid is being changed. An automated and widely dispersed energy generating, transmission, and distribution network is known as a "smart grid." It is distinguished by a full duplex network with a two-way flow of information and electricity. It is a closed-loop monitoring and reaction system [4]. Many organisations around world, including NIST (National Institute of Standards and Technology), IEEE (Institute of Electrical and Electronics Engineers), ETP (European Technology Platform), IEC (International Electro technical Commission), and EPRI (Electric Power Research Institute), are developing and conceptualising the smart grid. These organisations are also diligently researching the harmonisation of numerous standards and a wide range of standards. It is defined in a variety of ways depending on how useful, technological, or functional it is. As per definition represented by U.S. Department of Energy, "A smart grid uses digital technology to improve reliability, security, and efficiency (both economic and energy) of the electric system from large generation, through the delivery systems to electricity consumers and a growing number of distributed-generation and storage resources" [5]. The power grid (PG) can be made more dependable, adaptable, efficient, and durable through the use of smart grid technology, which integrates electrical, informational, and communication technologies. It is an intelligent PG that incorporates a variety of renewable and alternative energy sources. Key components of a SG implementation include automated monitoring, data collecting, control, and developing communication methods. Utilizing a wide range of communication standards necessitates analysis and optimization based on requirements and limits. These specifications are chosen based on factors including bandwidth needs, application kind, and coverage area. According to applications of communication methods at different levels of SG deployment, the hierarchical communication network for SG may be divided into 3 methods: HAN (home area network), NAN (neighbourhood area network), and WAN (wide area network). Global effect of ML and DL methods is growing and looking positive. The original use of ML and DL was in the condition monitoring of electric machinery. Emerging models offer reliable and precise measurements for fault prediction in rolling bearings and electric machinery. Applications can also be found in supply chains and logistics. A supply chain that is connected will change and accommodate new information as it is supplied. A linked method can proactively respond to that reality and shift manufacturing priorities if a shipment is associated to a weather delay. Another industry where ML and DL methods are used is transportation. Secure IoT methods are also being developed to store and handle massive data from scalable sensors for health care applications. Another platform for applying ML and DL models is smart grids [6].

Contribution of this research is as follows:

(1) To propose novel method in enhancing the smart grid security and industry fault detection using wireless sensor network with deep learning architectures

(2) The smart grid network security has been enhanced using blockchain-based smart grid node routing protocol with IoT module

(3) The industrial analysis has been carried out based on monitoring for fault detection in network using Q-learning-based transfer convolutional network

The organisation of this article is as follows: Section 2 gives the related works, the proposed technique is described in Section 3, Section 4 explains the performance analysis, and the conclusion is given in Section 5.

## 2. Related Works

The following are the main issues in a smart city: smart grids in smart buildings, smart classrooms, traffic monitoring, education and classrooms, waste management, governance, environment monitoring, health care in hospitals, agriculture, industrial IoT, etc. We will now map each smart city issue with solution offered by WSN-IoT ML methods. In field of machine learning, WSN node localization issue is regarded as a classification or multivariate regression problem. To address node localization issues in WSN-IoT, SVM classification [7] or SVM regression method [8] methods are used. Correlation techniques and the Bayesian learning methodology are used to address security challenges, as shown in [9]. In the ML domain, clustering tasks in the WSN-IoT are referred to as cluster head selection tasks. For clustering, k-NN, PCA, and ANN have all been employed. In the ML field, WSN node energy management is seen as a prediction issue. Energy difficulties have been predicted using Q-learning [10]. Similar to this, energy harvesting-based WSN (EH-WSN) uses reinforcement learning methods like Q-learning, SARSA, and deep Q-learning to forecast future energy availability [11]. Problems with fault detection and event monitoring are regarded as classification models. SVM [12] and rule-based learning [13] techniques are used to resolve this. The approach proposed by work [14] employs RSSI to forecast the link quality. Author [15] uses RSSI calibration to enhance measurement quality; however, because this method may increase computational complexity, it is not appropriate for low-cost WSNs. LQI can, however, be utilized to find high-quality links when it is very high [16]. Otherwise, LQI has trouble determining if a link is of good quality or not. A Kalman filter-based LQP approach is proposed by the author in [17]. To gather smooth value of SNR, they filter RSSI and eliminate noise floor. ANNs are used in several manufacturing processes, such as process control and the production of semiconductors. Additionally, ANNs were used in [18–20] to predict as well as evaluate machine specification data, such as machine geometry and design, motor performance, range, and cost. Exhaustiveness, comparable incentive structure with an untraceability characteristic, exhaustiveness, and the compact outcomes of a different neural network technique are measured empirically to determine the success of the suggested model [21, 22]. The processing and data transfer of physical processes is known as the cyberphysical system (CPS) [23]. Advancement in artificial neural networks (ANNs) was also utilized to predict and estimate jet engine component manufacturing costs during the early design phase [24]. Last but not least, ANNs were employed to monitor machine tools in real time [25].

More expensive nodes want greater rewards for accomplishing transactions in a business which work with the code of demand and supply [26]. Smaller ledger: this could affect the security and the immutability of the blockchain and all the data stored in it. Slower transactions: transactions could be slower than usual process even with the absence of third parties. Transaction expenses and speed of network: the transaction charge of the blockchain technology is rather high after being advertised as "nearly free" during the first few years. Analysis of variance (ANOVA) and back propagation neural networks (BPNN) with feed-forward architecture are two techniques for locating approximations and the optimum fit for optimization and search issues [27]. To evenly distribute traffic across these sensor nodes, several routing protocols must be developed [28]. The purpose of this review is to give readers a greater understanding of the function and application of security-based architecture in various approach. It will therefore help us assess the size of our problem.

## 3. System Model

This section discusses novel technique in enhancing the smart grid security and industry fault detection using wireless sensor network with deep learning architectures. The smart grid network security has been enhanced using blockchain-based smart grid node routing protocol with IoT module. The industrial analysis has been carried out based on monitoring for fault detection in network using Q-learning-based transfer convolutional network. The proposed blockchain-based smart grid sensor network architecture is shown in Figure 1.

*3.1. Blockchain-Based Smart Grid Node Routing Protocol with IoT Module.* Figure 2 displays the network model taken into consideration in this study. In this paradigm, a smart metre ($\mathrm{SM_i}$) is connected to a number of consumers, and a service provider ($\mathrm{SP_j}$) is connected to a number of smart metres. Peer-to-peer (P2P) service provider networks, often known as P2P SP networks, are created by a collection of service providers. All installed smart metres $\mathrm{SM_i}$ and service providers $\mathrm{SP_j}$ must be registered with a trustworthy registration authority (RA) in offline mode. The RA conducts the registration procedure in a secure manner. Smart metres $\mathrm{SM_i}$ and service providers $\mathrm{SP_j}$ interact securely using a session key they establish among themselves with the use of an access control mechanism, whereas users and smart metres $\mathrm{SM_i}$ communicate via secure communication. The SP network's service providers additionally create private pairwise keys among themselves for their secure connections. In accordance with this network paradigm, $\mathrm{SM_i}$ surreptitiously collects data from its affiliated users before bringing it to the service provider $\mathrm{SP_j}$, with whom the smart metres $\mathrm{SM_i}$ are registered. Using the information gathered, $\mathrm{SP_j}$ then builds a block of transactions. Once the service providers in the SP network have reached consensus, the newly produced block can be added to the blockchain that already exists.

When estimating IoT device energy usage, we need take into account both receiving and delivering energy. Let $E_{\mathrm{Trams}}(n, d)$ represent the price of sending $n$ bits of data over $d$ metres, and let $E_{R\,\mathrm{Rev}}(n)$ represent the price of receiving $n$ bits of data over $d$ metres. For sending $n$ bits using

$$E_{\mathrm{Trams}}(n, d) = \begin{cases} E_{Emb\omega} * n + E_{Amp} * n * d^2, & d \le d_0, \\ E_{Emb\omega} * n + E_{\mathrm{Amp}} * n * d^4, & d > d_0. \end{cases} \tag{1}$$

For receiving $n$ bits by

$$E_{R\,\mathrm{Rev}}(n) = E_{Embb} * d. \tag{2}$$

IoT device energy consumption is calculated using

$$E_{\mathrm{slepp}}(t) = E_{\mathrm{low}} * t, \tag{3}$$

where flow represents the power used by any device during a single second of sleep. $T$ seconds are spent in sleep mode in total. Each IoT device in the network uses up equivalent to

$$E_{\mathrm{Total}} = E_{\mathrm{Trans}}(n, d) + E_{\mathrm{Rece}}(n) + E_{\mathrm{sleepp}}(t). \tag{4}$$

The distance formula uses the space taken up by data as it travels from the CH to the sink and distance covered by data packets as they go from sink to the cluster node. Distance should fall between 0 and 1. The normalisation is finished as a result. The distance metric is normalised using the denominator $\sum_{k=1}^{m} \sum_{i=1}^{m} |N_k^n - N_l^H|$. When the distance between the CH and normal node is great, as illustrated in equation (5), the distance parameter receives a substantial value. Route discovery of packets in the networks is represented in Algorithm 1.

$$F_i^d = \frac{\sum_{k=1}^{m} \sum_{l=1}^{n} \left|N_k^n - N_l^H\right| + \left|N_i^h - N^s\right|}{\sum_{k=1}^{m} \sum_{i=1}^{m} \left|N_k^n - N_i^H\right|}, \tag{5}$$

where $m$ represents all of the network's nodes and $h$ represents total number of CHs. The symbols for sink node, normal node, and CH node are $N^{\mathrm{s}}$, $N^{\mathrm{n}}$, and $N^{\mathrm{h}}$. Maximise problem becomes a minimising problem by eliminating the cumulative energies from one, as shown in (9). Energy is the most important measure, and it may be estimated by figuring out how much energy each node still has. By calculating cumulative cluster energy as well as total

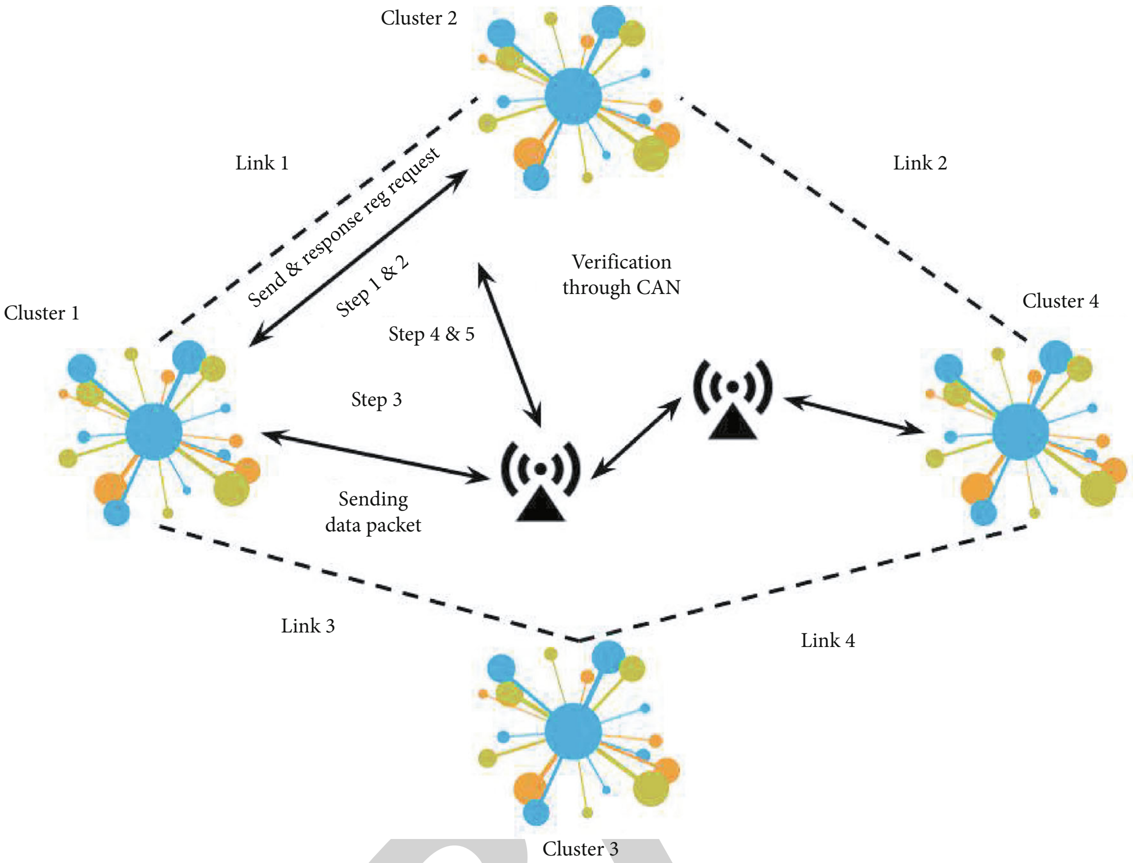

Figure 1: Blockchain-based smart grid sensor network architecture.

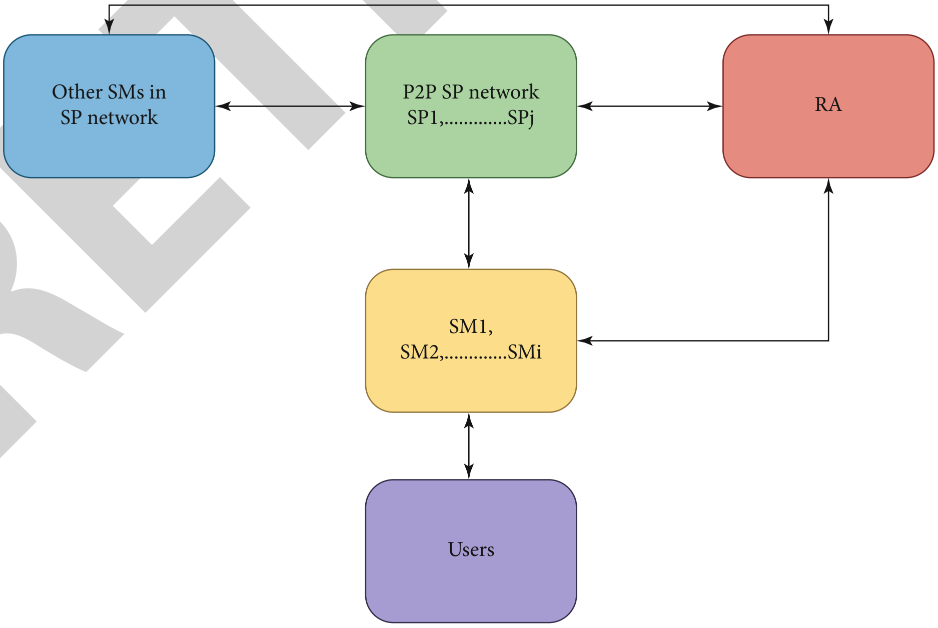

Figure 2: Blockchain-based IoT-enabled smart grid flow chart.

energy from all clusters, remaining energy is determined. The modelled energy metric is displayed in

$$F_i^* = \frac{\sum_{i=1}^{h} N_c^E(t)}{h \times \mathrm{Max}_{l=1}^{h}[\varepsilon(N_i^n)] \times \mathrm{Max}_{l=1}^{h}\left[\varepsilon\left(N_l^H\right)\right]},$$
$$N_c^e(l) = \sum_{k=1}^{m} \left[1 - \varepsilon(N_k^n) * \varepsilon\left(N_l^H\right)\right], \ (1 \le l \le h). \tag{6}$$

The node with the highest energy will be regarded as the ideal CH. The symbol for the total energy associated with CH is $\sum_{l=1}^{h} N_c^E(t)$.Maximum energy represented by CH and other nodes plus sum of all CHs is expressed as $h$ $\mathrm{Max}_{!=1}^{h}[\varepsilon(N_l^n)] \times \mathrm{Max}_{l=1}^{h}[\varepsilon(N_l^H)]$. The denominator can only show a maximum value of 1.When choosing the best CH, the network delay must be minimised, and all cluster members are immediately affected. The network delay increases by equation (7) if the number of cluster members rises.

$$F_i^\delta = \frac{\mathrm{Max}_{l=1}^{h}\left(C_{m,l}^H\right)}{m}. \tag{7}$$

The network's $i^{\text{th}}$ CH is represented by the letters $C$, $H$, $m$, and $l$. The delay value might range between 0 and 1. A minimum level of traffic density must be maintained to ensure an efficient network. The key factors affecting traffic density are packet loss, channel load, and buffer usage. The traffic density by equation (8) is determined by the average of these three metrics.

$$F_i^l = \frac{1}{3}[B_{ut} + P_{dr} + C_l]. \tag{8}$$

The best CH is believed to be the node with the most energy, shortest distance to the sink node, lowest traffic density, and shortest delay. Following the manta rays that came before it, each one swims in the direction of the best plankton. Each person updates their position based on the best answer found. In equation (9), the charging foraging model is illustrated.

$$x_i^d(t+1) = \begin{cases} x_i^d(t) + r \cdot \left(x_{\text{best}}^d(t) - x_i^d(t)\right) + \alpha \cdot \left(x_{\text{bett}}^d(t) - x_i^d(t)\right), & i = 1, \\ x_i^d(t) + r \cdot \left(x_{i-1}^d(t) - x_i^d(t)\right) + \alpha \cdot \left(x_{\text{best}}^d(t) - x_i^d(t)\right), & i = 2, \cdots N, \end{cases} \tag{9}$$

where $d$ and $t$ stand for dimension and iteration number, and $\alpha = 2 \cdot r \cdot p\,|\log\ (r)|$. Random vector whose value ranges from [0, 1] is $r$, while position of $i^{\text{th}}$ individual is $x_i^d(t)$. denotes weight coefficient. The cluster formations are represented in Algorithm 2.

Area with a higher concentration of plankton is shown as $x_{\text{best}}^d(t)$. $x_i^d(t)$ is used to denote the updated position of individual $i$. Then, the participants are engaged in a spiral path, which is modelled in

$$\begin{cases} X_i(t+1) = X_{\text{best}} + r \cdot (X_{i-1}(t) - X_i(t)) + e^{h_\omega \omega} \cdot \cos\ (2\pi\omega) \cdot (X_{\text{best}} - X_i(t)), \\ Y_i(t+1) = Y_{\text{best}} + r \cdot (Y_{i-1}(t) - Y_i(t)) + e^{\phi\omega} \cdot \sin\ (2\pi\omega) \cdot (Y_{\text{best}} - Y_i(t)), \end{cases} \tag{10}$$

where the random number in equation (10) is denoted by the symbol, whose value can fall anywhere between [0, 1]. The definition of mathematical expression for cyclone foraging in the $n - D$ dimension is as follows:

$$\beta = 2e^r \frac{Z_{\text{lng}\gamma} - t_{+1}}{r} \cdot \sin\ (2\pi r_1), \tag{11}$$

where $r_1$ is a random number with a value that can be between 0 and 1. Each person conducts a random search using reference position. Cyclone foraging improves the exploratory capability while also achieving good exploitation. Each person must adjust their position rather than remain in the same one in order to arrive at the best answer. A new reference position is assigned to each person in order to accomplish this position change in

$$x_{\text{namd}}^d = Lb^d + r \cdot \left(Ub^d - Lb^d\right). \tag{12}$$

Equation (13) represents the RBM 1 mathematical model

$$N^2 = \left\{N_1^2, N_2^2, \cdots, N_g^2, \cdots, N_r^2\right\},$$
$$G^2 = \left\{G_1^2, G_2^2, \cdots, G_z^2, \cdots, g_h^2\right\}, \tag{13}$$

where hidden neuron $g$ of RBM 1 is $G_n^1$ and $N_m^1$ denotes $j^{\text{th}}$ input neuron. Both visible and hidden levels receive bias. The total number of neurons in hidden and input layers is denoted in RBM 1 by letters $r$ and $v$ in

$$G_n^1 = \kappa\left[\sigma_n^1 + \sum_m N_m^1 \times w_{mn}^1\right], \tag{14}$$

where weight corresponding to hidden neuron $n$ and input neuron $m$ is $w_{mn}^1$ and bias supplied to $n^{\text{th}}$ hidden layer of RBM 1 is $N_r^2$. RBM 1 output is based on the DBN classifier's

```
Produce a Random Connected Graph
Start$Ec_{i.}$
Start maximum energy capacity value max $E_{\mathrm{caj}}$
Start energy harvesting value $E_H$
Start Activated Services Sact so
Start Objective Function to reducenum Periods = 0
node donumPeriods= numperiods +1
Solve Paths = MathematicalModel($E_{ci}$, Sact$^{*0}$, $F$)
for everyi-node in Paths do
Update Ecc$_i$
end for
for everyi-node in network do $Ec_i = Ec_i - E_H$
end for
for every origin node do
Determine a path $P$ in Paths to transmit
if $P = \theta$ then
                        Sect$^{\infty}$ = 0
end if
end for
end while
return numPeriods
```

Algorithm 1: Route discovery algorithm.

```
INPUT: CH sends CH$_{\mathrm{info}}$ packet packet to the CMs.
OUTPUT: CH sends Cluster member (CH) to the hub and TDMA slots to each CM.
Device a receives the CH$_{\mathrm{info}}$ packet from the Device $\beta$, where $\beta \in$ CH
                              CH$_{\mathrm{info}}f_0$ : <CH$_{\mathrm{in}}f_0$, $ID_\beta$ >
Device a selects the CH with the maximum received signal intensity as its CH after receiving
all CH$_{\mathrm{info}}$ packets.
Device $a$ sends the CH$_{\mathrm{join}}$ packet to the selected CH.
Device a receives the $CH_{\mathrm{join}}$ packet from the Device $\beta$, where $\alpha \in CH$
                              CH$_{\mathrm{join}}$ : <CH$_{\mathrm{join}}$, $ID_\beta$, $ID_{\mathrm{CH}}$ >
if ($ID_\alpha = 1D_{CH}$) then
Device $a$ sends the Cluster momier ($\alpha$) to the hub after receiving all the CH$_{\mathrm{join}}$ packets.
Device a sends the TDMA slots to each CM.
Else
Discard the packet.
end if
```

Algorithm 2: Cluster formation algorithm.

input features. Then, RBM 2, which is specified in, receives the produced output as an input in

$$N^2 = \left\{N_1^2, N_2^2, \cdots, N_g^2, \cdots, N_r^2\right\},$$
$$G^2 = \left\{G_1^2, G_2^2, \cdots, G_z^2, \cdots, g_h^2\right\}, \tag{15}$$

where RBM 1 and RBM 2 layers' input and hidden neurons, respectively, are represented by $A$ and $G$. The weight value derived from subsequent layers is denoted as equation (16) in RBM 2.

$$w^2 = \left\{w_{8R}^2\right\}. \tag{16}$$

In RBM 2, hidden neuron $n$ and visible neuron $n$ 0 are combined as $w_{mN'}^2$. The output of RBM 2 is given by

$$G_n^2 = \omega\left[\varpi_n^2 + \sum_m N_m^2 \times w_{mN'}^2\right] \forall N_m^2 \approx G_n^1. \tag{17}$$

*3.2. Q-Learning-Based Transfer Convolutional Network Based on Monitoring for Fault Detection.* Each batch of data, comprising action, reward, and state, is utilized to update Q table in Q-learning method. Entry $Q_k$ ($S_k$, $a_k$) in Q table is desirability of actions in finite sequence Ajj∈J+ in relation to states in the finite sequence (Si)i∈I+. The central component of reinforcement learning consists of a system and an agent, as shown in Figure 1. The agent examines the current state sk at time step $k$ before choosing action $a_k$ from a list of possible actions ($A$). Based on an acceptable reward, the results of the chosen action $a_k$ are scored ($r_{k+1}, R$). The agent determines whether the previous action was "good" or "poor" based on the reward's worth. Utilizing the Q-learning method, the agent finds the best possible course of action to maximise expected value $E$ [] of discounted reward, which is determined by

$$J(r_k) = \mathbb{E}\left[\sum_{k=1}^{\infty} \theta^{k-1} r_k\right]. \tag{18}$$

When $\theta = 0$, the agent just examines the current reward; however, when approaches 1, the agent considers both the current and future rewards. This is represented by $\theta \in [0, 1]$ in equation (18). In this regard, the Q table will be updated based on the Q-learning method, which is given by equation (19), when the agent calculates action $a_k$ and reward $r_{k+1}$ with respect to state transition $s_{k+1}$

$$\begin{aligned} Q_k(s_k, a_k) = {} & Q_{k-1}(s_{k,} a_k) + \eta_k(s_{k,} a_k) \\ & \times \left[r_{k+1} + \max\nolimits_{\Delta_{k+1}} Q_{k-1}(s_{k+1}, a_{k+1}) - Q_{k-1}(s_{kr} a_k)\right]. \end{aligned} \tag{19}$$

Notably, Q-learning method starts with a Q1 initialization ($s_1, a_1$). The Q table will then be modified in light of the observations. It is usual to employ a tolerance parameter with the condition $Q_k \mid Q_k - Q_k - 1 \mid \leq \delta$ to determine the minimal threshold for convergence. Actually, the agent's decision-making is supported by this knowledge. The controller will select the action $a_k$ as equation (20) at each time step.

$$a_k = (A_j)_{j \in J+j} j = \operatorname{argmax}(Q_k(s_{k+c})). \tag{20}$$

Equation (21) is the function that is used to determine the agent's reward for moving from state $s_k$ to state $s_{k+1}$.

$$r_{k+1} = \begin{cases} \left.\dfrac{e(kT) - c(k+1)T}{e(kT)}\right|, & |e((k+1)T)| < |e(kT)|, \\ -\xi, & |e((k+1)T)| < |c(kT)|. \end{cases} \tag{21}$$

The algorithm is able to reach the ideal Q table when $k \longrightarrow \infty$. Additionally, systems often converge to their optimal solution with an acceptable tolerance $\delta$ for a limited value of $k$. For each agent $I$, dynamic of local neighbourhood tracking error is defined as

$$\varepsilon_i(k+1) = \sum_{j \in \mathcal{N}_i} e_{ij}\left(x_j(k+1) - x_i(k+1)\right) + b_i(x_0(k+1) - x_i(k+1)). \tag{22}$$

It can be further rewritten as

$$\varepsilon_i(k+1) = A\varepsilon_i(k) - (d_i + b_i)B_i u_i(k) + \sum_{j \in \mathcal{N}_i} e_{ij} B_j u_j(k). \tag{23}$$

The definition of local performance index for each agent $I$ is

$$J_i\left(\varepsilon_i(k), u_i(k), u_j(k)\right) = \sum_{k=0}^{\infty} \gamma^k U_i\left(\varepsilon_i(k), u_i(k), u_j(k)\right). \tag{24}$$

With the utilitarian purpose, $U_i$ is expressed as equation (25) for each agent $I$.

$$\begin{aligned} U_i\left(\varepsilon_i(k), u_i(k), u_j(k)\right) = {} & \varepsilon_i^T(k) Q_{ii} \varepsilon_i(k) + u_i^T(k) R_{ii} u_i(k) \\ & + \sum_{j \in \mathcal{N}_i} u_j^T(k) R_{ij} u_j(k), \end{aligned} \tag{25}$$

where $Q_{ii} \geq 0 \leq \mathbb{R}^{n \times n}, R_{ii} > 0 \in \mathbb{R}^{m_i \times m_i}$ and $R_{ij} > 0 \in \mathbb{R}^m \times m_j$ are all positive symmetric weighting matrices and $0 < \gamma \leq 1$ is a discount factor. Value function of every agent $I$ is therefore described as equation (26) given fixed control $(u_i(l), u_j(l))$ of agent $I$ and its neighbours.

$$\begin{aligned} V_i(\bar{\varepsilon}_i(k)) &= \sum_{l=k}^{\infty} \gamma^{l-k} U_i\left(\varepsilon_i(l), u_i(l), u_j(l)\right), \\ \overline{\varepsilon_i}(k) &= \begin{bmatrix} \varepsilon_i(k)^T & \varepsilon_{j1}(k)^T & \varepsilon_{j2}(k)^T & \cdots & \varepsilon_{jp}(k)^T \end{bmatrix}^T \\ & \in \mathbb{R}^{n \times (p+1)}\,; j1, j2, \cdots, jp \in \mathcal{N}_i, \end{aligned} \tag{26}$$

where $p$ is number of neighbours of agent $I$. Each agent's performance is rated by local performance index (9). Local information is captured by value function for each agent $I$ (11). As a result, value function's solution structure is expressed in terms of local vector$i\,(k)$. We can derive by equation using equations (25) and (26) and

$$\begin{aligned} V_i(\bar{\varepsilon}_i(k)) &= \sum_{l=k}^{\infty} \gamma^{l-k} U_i\left(\varepsilon_i(l), u_i(l), u_j(l)\right) \\ &= \sum_{l=k}^{\infty} \gamma^{l-k} \left(\varepsilon_i^T(l) Q_{ii} \varepsilon_i(l) + u_i^T(l) R_{ii} u_i(l) + \sum_{j \in \mathcal{N}_i} u_j^T(l) R_{ij} u_j(l)\right) \\ &= \sum_{l=k}^{\infty} \gamma^{l-k} \left(\varepsilon_i^T(l) Q_{ii} \varepsilon_i(l) + \bar{u}_i^T(l) R_i \bar{u}_i(l)\right), \end{aligned} \tag{27}$$

where control law of agent $I$ ($u_i\,(l)$) and neighbouring agents' control laws are included in the vector $u_j\,(l)$, i.e., $u_j(l)$ and $\bar{u}_i(l)[u_i(l)^T\, u_{j1}(l)^T\, u_{j2}(l)^T \cdots u_{jp}(l)^T]^T\,; j1, j2, \cdots, jp \in \mathcal{N}_i, R_i$.

Each agent's diagonal matrix, $R_i$, contains the diagonal entries $R_{ii}$ and $R_{ij}$. We may find equation (14) and control law $\bar{u}_i(k) = -K_i\varepsilon_i(k)$ by using the following two equations:

$$\begin{aligned} V_i(\bar{\varepsilon}_i(k)) &= \sum_{l=k}^{\infty} \gamma^{l-k}\left(\varepsilon_i^T(l)Q_{ii}\varepsilon_i(l) + \bar{u}_i^T(l)R_i\bar{u}_i(l)\right) \\ &= \sum_{l=0}^{\infty} \gamma^{l}\left(\varepsilon_i^T(l+k)Q_{ii}\varepsilon_i(l+k)+\bar{u}_i^T(l+k)R_i\bar{u}_i(l+k)\right) \\ &= \sum_{l=0}^{\infty} \gamma^{l}\varepsilon_i^T(l+k)\left(Q_{ii} + K^TR_iK\right)\varepsilon_i(l+k). \end{aligned} \tag{28}$$

Dynamic of neighbourhood tracking error in a local setting can be rewritten as

$$\begin{aligned} \varepsilon_i(k+1) &= A\varepsilon_i(k) - (d_i + b_i)B_iu_i(k) + \sum_{j\in\mathcal{N}_i} e_{ij}B_ju_j(k) \\ &= A\varepsilon_i(k) + \left[-(d_i + b_i)B_i \quad e_{ij1}B_{j1} \quad e_{ij2}B_{j2} \quad \cdots \quad e_{ijp}B_{jp}\right] \\ &\quad \times \left[u_i(k) \quad u_{j1}(k) \quad u_{j2}(k) \quad \cdots \quad u_{jp}(k)\right]^T \\ &= A\varepsilon_i(k) + B\bar{u}_i(k), \end{aligned} \tag{29}$$

where $j_1, j_2 \cdots, j_\mathrm{p} \in \mathcal{N}_i$. Substituting $\bar{u}_i(k) = -K_i\varepsilon_i(k)$ into equation (16), next equation is deduced by

$$\varepsilon_i(k+1) = (A - BK_i)\varepsilon_i(k) = K_{1i}\varepsilon_i(k), \tag{30}$$

where $K_{1i} = A - BK_i$.

The suggested approach should be conditional on features having similar distributions across domains to transfer knowledge from source domain to target domain. Using back propagation computation of the pretrained CNNs, an error minimization optimization method is used to overcome feature distribution mismatch. Maximum mean discrepancy, or MMD, was a widely used distance metric for comparing probability distributions between two domains in earlier literature. That is, $DS = \{X_T, X_S\}$ and $DT = \{X_T, P(X_T)\}$, respectively, represent datasets in source domain and target domain. In the meantime, $X_S = \prod\{X_T,\}\ nsi = 1$ and $XT = \prod\{xTi\}\ nti = 1$ with $n_t$ samples. Equation (31) determines their MMDs:

$$\begin{aligned} \mathrm{Mean}_H(X_S) &= \frac{1}{n_s}\sum_{i=1}^{n_s} H\left(x_s^i\right), \\ \mathrm{Mean}_H(X_T) &= \frac{1}{n_t}\sum_{j=1}^{n_t} H\left(x_T^j\right), \end{aligned} \tag{31}$$

where $H()$ is an RKHS and sup () is supremum of aggregate (reproducing kernel Hilbert space). For evaluating feature distribution difference of domain invariant features in this study, MMD is used. MMD $(X_S, X_T)$ is taken into consideration as optimization objective to regularise weights of CNNs in order to attain similar distributions from two domains. A linear-time approximation of MMD is utilized by equation (32) in place of MMD due to computational expense of doing MMD calculation on feature embeddings. The transfer of cluster process by utilizing CL is represented in Algorithm 3.

$$\mathrm{MMD}_1^2(X_S, X_T) = \frac{2}{M}\sum_{\mathrm{i}=1h_l(\mathbf{z}_i)}^{M/2} h_l(\mathbf{z}_i), \tag{32}$$

where $\mathbf{z}_i = (\mathbf{x}_{2i-1}^s, \mathbf{x}_{2i}^s, \mathbf{x}_{2j-1}^t, \mathbf{x}_{2j}^t)$ and $h_l(\mathbf{z}_i)$ is a kernel operator described on quad-tuple as follows by

$$h_l(\mathbf{z}_i) = k(x_{2i-1}^s, \boldsymbol{x}_{2i}^s) + k\left(x_{2j-1}^t, \boldsymbol{x}_{2j}^t\right) - k\left(x_{2i-1}^s, x_{2j}^t\right) - k\left(x_{2i}^s, x_{2j-1}^t\right). \tag{33}$$

While CNNs are being reweighted, the prediction error should also be kept to a minimum. Therefore, another optimization goal is the prediction error. $\mathrm{MMDH}(X_S, X_T)$ and MSE can therefore be used to compute the overall loss. Normalisation is necessary since the value ranges of MSE and $\mathrm{MMDH}(X_S, X_T)$ vary. Nadir and utopia points are used in this study to normalise the aforementioned goals. Lower bound of no. $I$ goal, as determined by minimising objective as given by equation (34), is provided by the utopia point $z_i{}^u$:

$$z_i{}^u = \min f(i). \tag{34}$$

By maximising the objectives according to equation (35), nadir point $z_i{}^N$ gives upper bound of objective number $I$:

$$z_i{}^N = \max_{1\ll j<l} f(j), \tag{35}$$

where $I$ represents how many objective functions there are in total. Equation (36) can be used to calculate the normalised MMD and MSE in accordance with equations (34) and (35):

$$\begin{aligned} \mathrm{NMMD}_H &= \frac{(\mathrm{MMD}_{H1}(X_S, X_T) - z_1{}^u)}{(z_1{}^N - z_1{}^u)}, \\ \mathrm{NMSE} &= \frac{(\mathrm{MSE} - z_2{}^u)}{\left(z_2^N - z_2{}^u\right)}, \end{aligned} \tag{36}$$

where NMMDH and NMSE are, respectively, normalised $\mathrm{MMDH}(X_S, X_T)$ and MSE. Total loss function is the last.

```
Initialize $X_i^s, X_i^\tau ; Y_i^s \longleftarrow 0$,
Evaluate initial kernel parameter list $\sigma \sim [2^u], -1 \le n \le 12$
iteration = 0;
while training do
iteration = iteration +1;
Evaluatei forward mini-batch predictions utilizing CNNs layers on target data
    $\varphi_i^t = \mathbf{W}_{CNN}(X_i^t) + \mathbf{B}_{CNN}$
Evaluatei forward feature embeddings for source and target domain batch:
    $\varphi_{s,l}(X_i^s) \longleftarrow f(X_i^s, l)$
    $\varphi_{t,l}(X_i^t) \longleftarrow f(X_i^t, l)$
Project feature embeddings $\varphi(\mathrm{X}_s)$ and $\varphi(\mathrm{X}_t)$ into RKHS with chosen Gaussian kernels $\boldsymbol{N} \sim (0, \boldsymbol{\sigma})$
    $h_l(\mathbf{z}_i) = k(\mathbf{x}_{2i-1}^s, \mathbf{x}_{2i}^s) + k(\mathbf{x}_{2j-1}^t, \mathbf{x}_{2j}^t) - k(\mathbf{x}_{2i-1}^s, \mathbf{x}_{2j}^t) - k(\mathbf{x}_{2i}^s, \mathbf{x}_{2j-1}^t)$
Select optimal kemel parameter $\sigma \in \sigma$ to enhance distribution difference between embeddings
Evaluate layer-wise MMD as
    $\mathrm{MMD}_i^2(s, t) = 2/M\sum_{i=1}^{M/2} h_l(\mathbf{z}_i)$
Evaluate mini-batch loss on $i$ examples:
    $\mathscr{L}_{\text{total}}(\mathrm{X}_s, \mathrm{X}_t, Y^\wedge, Y) = w_1\mathrm{MSE}(Y, Y^\wedge) + (w_2/R)\sum_{r=1}^{R} \mathrm{MMD}_l^2(\mathrm{X}_s, \mathrm{X}_t)_r$
End while
```

ALGORITHM 3: Algorithm of transfer CL.

TABLE 1: Comparative analysis of bit error rate.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 56 | 52 | 45 |
| 100 | 59 | 55 | 48 |
| 150 | 63 | 59 | 51 |
| 200 | 66 | 62 | 53 |
| 250 | 68 | 63 | 55 |
| 300 | 71 | 65 | 56 |
| 350 | 75 | 69 | 59 |
| 400 | 79 | 71 | 61 |
| 450 | 81 | 73 | 63 |
| 500 | 83 | 75 | 65 |

The weighted sum of the two normalised targets by equation (37) can be used to determine loss.

$$\mathscr{L}_{\text{total}}\left(X_s, X_t, \widehat{Y}, Y\right) = w_1 \cdot \mathrm{NMMD}_H + w_2 \cdot \mathrm{NMSE}, \quad (37)$$

where $w1$ and $w2$ are weights of two objectives and $\sum wi = 1\ 2i = 1$. Weighting is used to compromise between task loss objective and MMD minimization. In light of this, these are set to $w1, w2 = [0.9, 0.1]$.

## 4. Experimental Analysis

A sample distribution grid made up of a 15 kV 485 MV grid and 400 V LV grids is simulated in order to test the planned services. Used grid is made up of buses on MV side, one of which is main HV/MV substation, 9 nodes connected to MV/LV 488 substations feeding residential loads. Radial operation of grid is constrained in experiments that follow. A reference case for tests is one of the branches that is regarded as normally open. However, potential to open or close any of the MV lines is taken into consideration while rating the Network Topology Reconfiguration service.

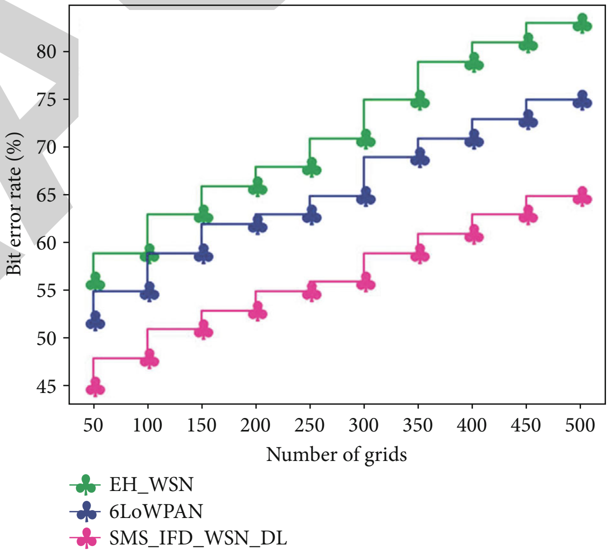

FIGURE 3: Comparative analysis of bit error rate.

Table 1 and Figure 3 show comparative analysis between proposed and existing techniques in terms of BER. BER, which is typically stated as ten to a negative power, is the proportion of bits that are incorrect to the total amount of bits received during a transmission. The bit error ratio is evaluated by dividing total number of bits transferred over time period under consideration by number of bit mistakes. BER is a performance metric that has no units and is frequently stated as a percentage. Expected value of BER is known as the bit error probability. The proposed technique obtained BER of 65%, while existing technique EH_WSN attained 83% and 6LoWPAN attained 75%.

Table 2: Comparison of end-to-end delay.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 45 | 42 | 38 |
| 100 | 48 | 45 | 39 |
| 150 | 51 | 48 | 42 |
| 200 | 53 | 51 | 43 |
| 250 | 55 | 55 | 45 |
| 300 | 59 | 58 | 49 |
| 350 | 63 | 62 | 51 |
| 400 | 66 | 64 | 53 |
| 450 | 69 | 65 | 55 |
| 500 | 75 | 72 | 57 |

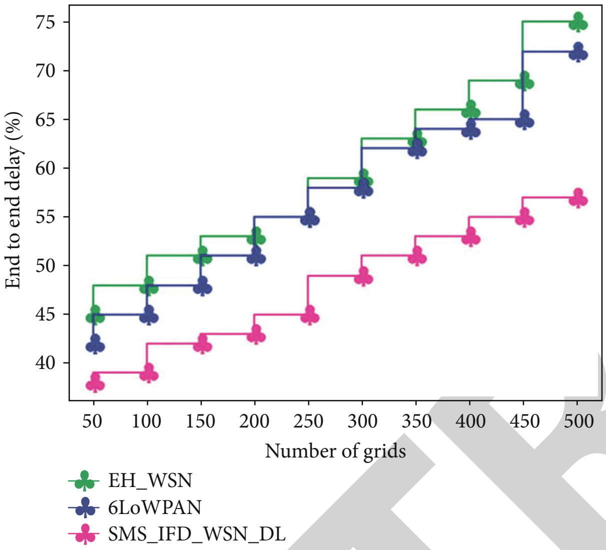

Figure 4: Comparison of end-to-end delay.

Table 3: Comparison of throughput rate.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 70 | 75 | 79 |
| 100 | 72 | 77 | 81 |
| 150 | 75 | 79 | 83 |
| 200 | 77 | 81 | 85 |
| 250 | 79 | 83 | 88 |
| 300 | 81 | 85 | 90 |
| 350 | 83 | 88 | 92 |
| 400 | 85 | 89 | 94 |
| 450 | 88 | 91 | 96 |
| 500 | 89 | 93 | 97 |

From Table 2 and Figure 4, the comparison of end-end delay has been analysed between proposed and existing techniques. One-way delay (OWD), often known as end-to-end delay, is the amount of time it takes a packet to travel from source to destination across a network. This term, which is frequently used in IP network monitoring, varies from RTT in that it only measures the journey from source to

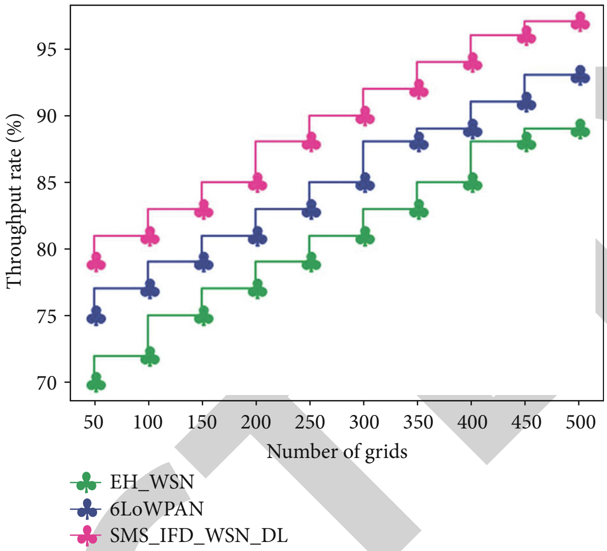

Figure 5: Comparison of throughput rate.

Table 4: Comparative analysis of spectral efficiency.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 61 | 66 | 72 |
| 100 | 63 | 71 | 75 |
| 150 | 65 | 73 | 77 |
| 200 | 68 | 75 | 79 |
| 250 | 71 | 79 | 82 |
| 300 | 73 | 81 | 85 |
| 350 | 75 | 82 | 88 |
| 400 | 79 | 84 | 89 |
| 450 | 81 | 85 | 92 |
| 500 | 83 | 88 | 93 |

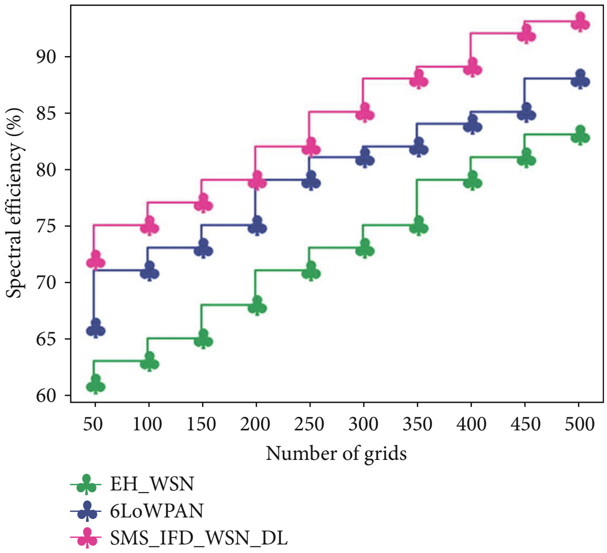

Figure 6: Comparative analysis of spectral efficiency.

Table 5: Comparison of accuracy.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 60 | 65 | 72 |
| 100 | 63 | 66 | 75 |
| 150 | 65 | 69 | 79 |
| 200 | 66 | 72 | 81 |
| 250 | 71 | 74 | 83 |
| 300 | 73 | 78 | 85 |
| 350 | 75 | 79 | 88 |
| 400 | 79 | 81 | 91 |
| 450 | 81 | 83 | 92 |
| 500 | 83 | 85 | 95 |

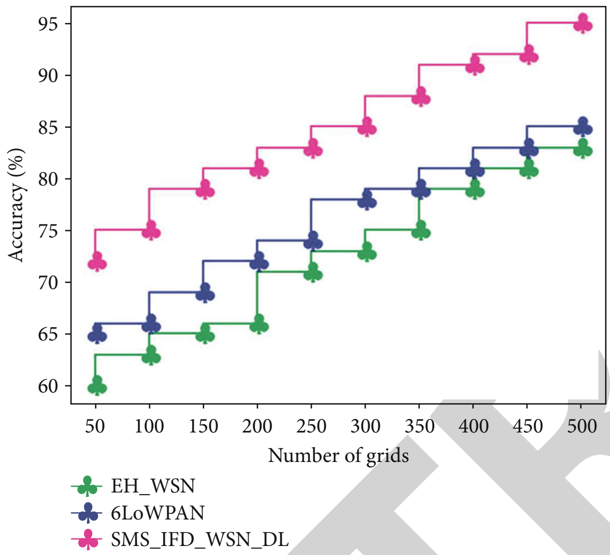

Figure 7: Comparison of accuracy.

Table 6: Comparison of RMSE.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 65 | 62 | 55 |
| 100 | 68 | 65 | 56 |
| 150 | 71 | 66 | 58 |
| 200 | 73 | 72 | 59 |
| 250 | 77 | 75 | 62 |
| 300 | 80 | 79 | 63 |
| 350 | 82 | 81 | 65 |
| 400 | 84 | 83 | 71 |
| 450 | 86 | 85 | 73 |
| 500 | 89 | 88 | 75 |

destination in a single direction. The proposed technique obtained end-to-end delay of 57%, while existing technique EH_WSN attained 75% and 6LoWPAN attained 72%.

Table 3 and Figure 5 show comparative analysis between proposed and existing techniques in terms of throughput rate. There are several ways to calculate the throughput efficiency formula, but the fundamental formula is $I = R^* T$. In other terms, when "rate" refers to the throughput, inventory is equal to rate times time. Throughput rate attained by proposed technique is 97%; existing EH_WSN obtained 89%, and 6LoWPAN obtained 93%.

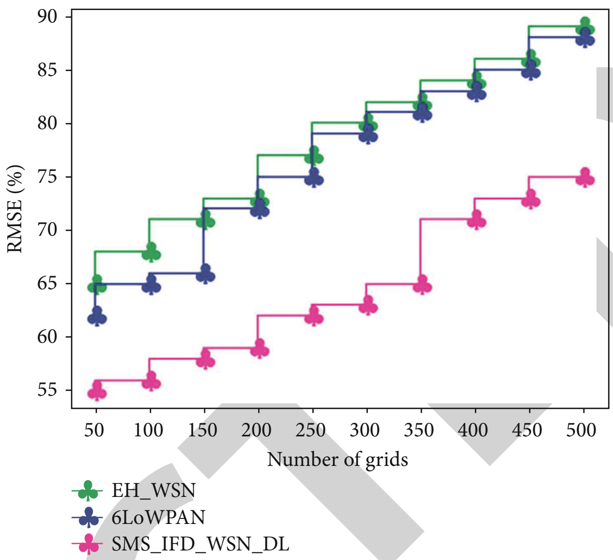

Figure 8: Comparison of RMSE.

Table 7: Comparative analysis of MAP.

| Number of grids | EH_WSN | 6LoWPAN | SMS_IFD_WSN_DL |
|---|---|---|---|
| 50 | 45 | 42 | 36 |
| 100 | 48 | 44 | 38 |
| 150 | 49 | 46 | 39 |
| 200 | 52 | 49 | 41 |
| 250 | 54 | 51 | 43 |
| 300 | 58 | 53 | 45 |
| 350 | 61 | 55 | 49 |
| 400 | 63 | 57 | 51 |
| 450 | 66 | 61 | 53 |
| 500 | 69 | 63 | 55 |

Table 4 and Figure 6 show comparative analysis of spectral efficiency between proposed and existing techniques. The maximum amount of data that may be sent over a cellular network to a given number of users per second while preserving a reasonable level of service is referred to as spectral efficiency. When we talk about spectral efficiency, we often refer to the total spectral efficiency of all transmissions within a cellular network cell. It is expressed as bit/s/Hz. Bits/s/Hz (b/s/Hz) is the unit of measurement for spectral efficiency, which is a measure of how quickly data can be delivered within a designated bandwidth. There is a maximum theoretical spectral efficiency value for each type of modulation. Another significant element that affects spectral efficiency is SNR. Spectral efficiency attained by proposed technique is 93%; existing EH_WSN obtained 83%, and 6LoWPAN obtained 88%.

From Table 5 and Figure 7, the comparative analysis has been carried out in terms of accuracy between proposed

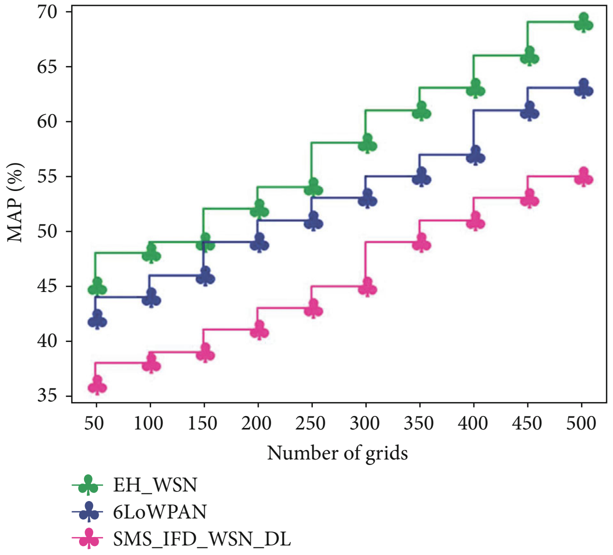

Figure 9: Comparative analysis of MAP.

and existing techniques. One parameter for assessing classification models is accuracy. Percentage of predictions that our method correctly predicted is called accuracy. It is one way to evaluate a model's performance, but by no means the only one. The proposed technique attained accuracy of 95%, existing EH_WSN obtained 83%, and 6LoWPAN obtained 85%.

Table 6 and Figure 8 show comparative analysis of RMSE between proposed and existing techniques. One of the methods most frequently utilized assess accuracy of forecasts is RMSE (root-mean-square deviation). It illustrates the Euclidean distance between measured true values and forecasts. The model can be deemed to be reasonably accurate in predicting the data if the RMSE values are between 0.2 and 0.5. Proposed method attained RMSE of 75%, existing EH_WSN obtained 89%, and 6LoWPAN obtained 88%.

Table 7 and Figure 9 show comparative analysis of MAP between proposed and existing techniques. Using a model and a prior probability or belief about the model, MAP entails computing a conditional probability of observing the data. For machine learning, MAP offers an alternative probability framework to maximum likelihood estimation. It uses the mean average precision (mAP). mAP evaluates a score by comparing detected box to ground-truth bounding box. Method detections are more precise in higher score. MAP attained by proposed technique is 55%; existing EH_WSN obtained 69%, and 6LoWPAN obtained 63%.

## 5. Conclusion

In this research, the proposed model is designed for improving the security of smart grid based on blockchain and routing. Here, the aim is to enhance the smart security using blockchain-based smart grid node routing protocol with IoT module. Then, the industrial analysis based on monitoring for fault detection using Q-learning-based transfer convolutional network is carried out. The seamless operation of energy management is ensured by smart grids, which respond to home and industrial requests from the cloud server and send the precise amount of energy. Each demand is filtered out by a cloud server, which reports on any unusual energy requests made by customers. Additionally, it stores energy projection data that can be used for more thorough research. This paper outlines an infrastructure for deploying resource-limited controlled devices at various consumer locations. These devices will be connected to a cloud monitoring server using an IoT network to upload their current demands and alert them of future needs. The experimental analysis has been carried out in terms of bit error rate of 65%, end-end delay of 57%, throughput rate of 97%, spectral efficiency of 93%, accuracy of 95%, MAP of 55%, and RMSE of 75%. For future work, we will consider an edge computing enabled blockchain network in the smart grid, where energy nodes can access and utilize computing services from an edge computing service provider. This integration may help the energy nodes achieve optimal energy management policy.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflict of interest

## References

[1] M. Marei, S. El Zaatari, and W. Li, "Transfer learning enabled convolutional neural networks for estimating health state of cutting tools," *Robotics and Computer-Integrated Manufacturing*, vol. 71, article 102145, 2021.

[2] G. Arya, A. Bagwari, and D. S. Chauhan, "Performance analysis of deep learning-based routing protocol for an efficient data transmission in 5G WSN communication," *IEEE Access*, vol. 10, pp. 9340–9356, 2022.

[3] R. K. Lenka, M. Kolhar, H. Mohapatra, F. Al-Turjman, and C. Altrjman, "Cluster-based routing protocol with static hub (CRPSH) for WSN-assisted IoT networks," *Sustainability*, vol. 14, no. 12, p. 7304, 2022.

[4] C. Mu, Q. Zhao, Z. Gao, and C. Sun, "Q-learning solution for optimal consensus control of discrete-time multiagent systems using reinforcement learning," *Journal of the Franklin Institute*, vol. 356, no. 13, pp. 6946–6967, 2019.

[5] S. Sengan, V. Subramaniyaswamy, V. Indragandhi, P. Velayutham, and L. Ravi, "Detection of false data cyber-attacks for the assessment of security in smart grid using deep learning," *Computers and Electrical Engineering*, vol. 93, article 107211, 2021.

[6] T. Kotsiopoulos, P. Sarigiannidis, D. Ioannidis, and D. Tzovaras, "Machine learning and deep learning in smart manufacturing: the smart grid paradigm," *Computer Science Review*, vol. 40, article 100341, 2021.

[7] S. Sivarajan and S. S. Jebaseelan, "Efficient adaptive deep neural network model for securing demand side management in IoT enabled smart grid," *Renewable Energy Focus*, vol. 42, pp. 277–284, 2022.

[8] P. Gope, P. K. Sharma, and B. Sikdar, “An ultra-lightweight data-aggregation scheme with deep learning security for smart grid,” *IEEE Wireless Communications*, vol. 29, no. 2, pp. 30–36, 2022.

[9] P. Singh, M. Masud, M. S. Hossain, and A. Kaur, “Blockchain and homomorphic encryption-based privacy-preserving data aggregation model in smart grid,” *Computers and Electrical Engineering*, vol. 93, article 107209, 2021.

[10] F. Shehzad, N. Javaid, A. Almogren, A. Ahmed, S. M. Gulfam, and A. Radwan, “A robust hybrid deep learning model for detection of non-technical losses to secure smart grids,” *IEEE Access*, vol. 9, pp. 128663–128678, 2021.

[11] Y. Tian, Q. Wang, Z. Guo et al., “A hybrid deep learning and ensemble learning mechanism for damaged power line detection in smart grids,” *Soft Computing*, vol. 26, no. 20, pp. 10553–10561, 2022.

[12] T. Teng and L. Ma, “Deep learning-based risk management of financial market in smart grid,” *Computers and Electrical Engineering*, vol. 99, article 107844, 2022.

[13] S. H. Majidi, S. Hadayeghparast, and H. Karimipour, “FDI attack detection using extra trees algorithm and deep learning algorithm- autoencoder in smart grid,” *International Journal of Critical Infrastructure Protection*, vol. 37, article 100508, 2022.

[14] C. Song, Y. Sun, G. Han, and J. J. Rodrigues, “Intrusion detection based on hybrid classifiers for smart grid,” *Computers and Electrical Engineering*, vol. 93, article 107212, 2021.

[15] Y. Ding, K. Ma, T. Pu, X. Wang, R. Li, and D. Zhang, “A deep learning-based classification scheme for false data injection attack detection in power system,” *Electronics*, vol. 10, no. 12, p. 1459, 2021.

[16] H. Wang, Z. Huang, X. Zhang, X. Huang, X. wei Zhang, and B. Liu, “Intelligent power grid monitoring and management strategy using 3D model visual computation with deep learning,” *Energy Reports*, vol. 8, pp. 3636–3648, 2022.

[17] T. Cheng, X. Zhu, X. Gu, F. Yang, and M. Mohammadi, “Stochastic energy management and scheduling of microgrids in correlated environment: a deep learning-oriented approach,” *Sustainable Cities and Society*, vol. 69, article 102856, 2021.

[18] A. Chehri, I. Fofana, and X. Yang, “Security risk modeling in smart grid critical infrastructures in the era of big data and artificial intelligence,” *Sustainability*, vol. 13, no. 6, p. 3196, 2021.

[19] C. Ma, “Smart city and cyber-security; technologies used, leading challenges and future recommendations,” *Energy Reports*, vol. 7, pp. 7999–8012, 2021.

[20] A. Belhadi, Y. Djenouri, G. Srivastava, A. Jolfaei, and J. C. W. Lin, “Privacy reinforcement learning for faults detection in the smart grid,” *Ad Hoc Networks*, vol. 119, article 102541, 2021.

[21] S. Rahmadika, P. V. Astillo, G. Choudhary, D. G. Duguma, V. Sharma, and I. You, “Blockchain-based privacy preservation scheme for misbehavior detection in lightweight IoMT devices,” *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 710–721, 2023.

[22] M. M. Gawde, G. Choudhary, S. K. Shandilya, R. U. Rahman, H. Park, and I. You, “Improvised model for blockchain in distributed cloud environment,” in *Cyberspace Safety and Security. CSS 2022. Lecture Notes in Computer Science, vol 13547*, X. Chen, J. Shen, and W. Susilo, Eds., Springer, Cham, 2022.

[23] M. Ramanan, L. Singh, A. S. Kumar et al., “Secure blockchain enabled cyber- physical health systems using ensemble convolution neural network classification,” vol. 101, Article ID 108058, 2022.

[24] S. A. Hoda and A. C. Mondal, “A study of data security on E-governance using steganographic optimization algorithms,” *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 10, no. 5, pp. 13–21, 2022.

[25] R. Paliwal and I. Khan, “Design and analysis of soft computing based improved routing protocol in WSN for energy efficiency and lifetime enhancement,” *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 10, no. 3, pp. 12–24, 2022.

[26] S. Murugan, A. Sampathkumar, S. Kanaga Suba Raja, S. Ramesh, R. Manikandan, and D. Gupta, “Autonomous vehicle assisted by heads up display (HUD) with augmented reality based on machine learning techniques,” in *Virtual and Augmented Reality for Automobile Industry: Innovation Vision and Applications. Studies in Systems, Decision and Control, vol. 412*, A. E. Hassanien, D. Gupta, A. Khanna, and A. Slowik, Eds., Springer, Cham, 2022.

[27] V. N. R. Jampana, P. S. V. Ramana Rao, and A. Sampathkumar, “Experimental and thermal investigation on powder mixed EDM using FEM and artificial neural networks,” *Advances in Materials Science and Engineering*, vol. 2021, Article ID 8138294, 12 pages, 2021.

[28] B. Banuselvasaraswathy, A. Sampathkumar, P. Jayarajan, M. Ashwin, and V. Sivasankaran, “A review on thermal and QoS aware routing protocols for health care applications in WBASN,” in *2020 International Conference on Communication and Signal Processing (ICCSP)*, pp. 1472–1477, Chennai, India, 2020.