

ORIGINAL RESEARCH

Open access

Efficient Offloading Techniques for Delay and Energy Minimization in IoT-Enabled Water Management Systems

Luca Ferraro^{1*}, Matteo Ricci¹, Giulia Moretti²

Abstract

In today's rapidly advancing landscape of information and communication technologies, metropolitan areas face a pressing challenge: water shortages combined with the urgent need for smart, sustainable water distribution systems. Deploying an Internet of Things (IoT)-driven monitoring solution can substantially strengthen oversight of water supply networks and help address common distribution problems such as leaks, pipe bursts, unauthorized withdrawals, tank overflows, and pump dry-running. Nevertheless, the proliferation of sensors and communication endpoints across smart urban environments is placing considerable strain on conventional networks, leading to higher latency and greater power consumption. This research outlines several practical approaches to minimize both delay and energy consumption in IoT-supported water distribution infrastructure in intelligent cities. A range of IoT-compatible network topologies is introduced, each shaped by specific hydraulic design features, surface land cover characteristics, and prevailing wireless propagation conditions. After assessing these layout options together with all critical communication metrics, the most suitable network structure for ongoing surveillance of a municipal water supply system in an Indian metropolitan setting is selected. As a practical illustration, the study develops and evaluates an IoT architecture tailored to a secondary-tier metropolitan city in India—namely, Kochi. Using location-specific conditions and the prevailing land-use/land-cover distribution, detailed delay and energy performance models for the IoT-enabled distribution network are constructed and examined. The paper also describes algorithms for classifying nodes and assigning tasks across edge and fog layers, supported by quantitative assessments of the resulting delays and energy consumption. These models yield approximate values for overall network latency and power consumption. Drawing on the derived outcomes and the associated state transition representations, the optimal positioning of fog computing nodes relative to edge devices and the central cloud platform can be determined. Moreover, evaluations across multiple configurations show that increasing the number of operational states in the transition diagram can deliver energy savings of up to 40%. The proposed techniques offer a roadmap for realizing low-latency, energy-conscious IoT communication infrastructures customized to particular geographic and functional contexts.

Keywords Delay minimization, Energy minimization, Internet of Things (IoT), Network architecture, Wireless communication

*Correspondence:

Luca Ferraro
luca.ferraro@unibo.it

¹ Department of Smart Grid Systems, Faculty of Engineering, University of Bologna, Bologna, Italy

² Department of Energy Management and Optimization, Faculty of Engineering, Polytechnic University of Turin, Turin, Italy

Introduction

Reliable access to sufficient water remains fundamental to the functioning of cities and societies. Meeting daily water needs hinges on the accurate, timely transport of appropriate volumes from a primary supply hub to diverse endpoints across the service area [1]. Achieving this level of precision in urban settings requires continuous, high-resolution tracking

of essential indicators—namely flow rate, pressure, reservoir/tank levels, and quality—at strategic points throughout the network [2].

A Water Distribution System (WDS) is a carefully planned, geographically distributed infrastructure that provides safe, potable water to communities. It encompasses raw water sources, bulk transmission mains, purification facilities, local distribution pipework, and конечные потребители. Water is first abstracted from the source, conveyed to treatment works, purified at water treatment plants (WTP), and finally delivered via branched piping systems that incorporate valves, hydrants, booster stations, reservoirs, and ancillary components. Used water collected from households and businesses can be treated and returned for non-potable reuse. Employing IoT-driven intelligent control within such systems promotes balanced allocation, ensuring that every designated consumer receives a fair share and that overall demand is met efficiently [3, 4]. City-wide IoT platforms furthermore establish seamless wireless links among all participating devices [5–7].

Despite these promising features, IoT deployments encounter several serious obstacles, including:

- Processing and transmitting extremely large datasets generated by machine-to-machine (M2M) and direct device interactions [8];
- Coordinating edge, fog, and cloud layers in near real-time while contending with constrained bandwidth, processing power, and battery life [9–11];
- Guaranteeing smooth interaction among APIs and services operating at micro, mini, and macro scales across the edge–fog–cloud hierarchy;
- Safeguarding essential security properties—confidentiality, data integrity, source authentication, non-repudiation, and proper access control—throughout the system [12, 13];
- Maintaining dependable operation for traffic ranging from medium to high priority, which demands support for diverse data formats, hardware platforms, and operational behaviors [14];
- Delivering very short response times coupled with high measurement precision, both indispensable for smart-city water management applications [15, 16].

IoT-enhanced water distribution networks are intended to enable intelligent, coordinated interaction among the various sensing and actuation units installed across urban water infrastructure [17, 18]. In the majority of Indian municipalities, freshwater is withdrawn from natural or artificial impoundments (lakes, ponds, lagoons, reservoirs) and gravity-fed or pumped to household, commercial, and industrial endpoints. Uneven availability—manifesting as localized shortages—frequently arises from suboptimal allocation and control strategies [19]. Smart IoT-based water management offers a powerful way to improve equity and efficiency. Successful implementation, however, necessitates selecting an appropriate radio propagation model that reflects the local terrain and predominant land-cover classes [20].

The present study identifies the chief factors responsible for poor performance in current water management practices. It proposes a targeted remedy: strategically placing purpose-designed sensor packages tailored to the identified limitations. These sensors—whether active or passive—are linked either directly or indirectly to critical assets such as pipelines, pumping stations, and control valves. Given that India's mobile networks currently rely predominantly on 4G and 5G technologies, the proposed architecture leverages these standards for all inter-device communication. The overall IoT framework organizes nodes into three tiers:

- Edge nodes — sensor units equipped with integrated communication hardware;
- Fog nodes — intermediate aggregation and preprocessing units that collect, analyze locally, and forward data;

- Cloud nodes — centralized platforms that perform final storage, advanced analytics, decision-making, and remote actuation.

IoT devices deployed in water distribution networks typically operate under strict power constraints, necessitating the adoption of energy-aware offloading mechanisms to reduce the overall power consumption of these field-deployed units. Transferring computational or data tasks to cloud- or edge-based servers, however, introduces additional network-induced latency, which must be mitigated through targeted methods that reduce transmission delays while preserving dependable connectivity suited to the particular geographic context under investigation. Accordingly, this study proposes a multi-tiered, heterogeneous IoT framework to implement delay-aware, power-efficient offloading within a unified water distribution infrastructure, explicitly accounting for the outlined difficulties [21]. The framework is engineered to make optimal use of available resources while simultaneously curbing latency and power consumption, thereby boosting the end-to-end performance of the entire water supply setup. Incorporating IoT solutions into urban water distribution infrastructure in intelligent cities raises notable issues regarding response times and power optimization, underscoring the need to engineer a robust, multi-layered, heterogeneous IoT architecture.

The research gaps highlighted above clearly indicate that existing water distribution systems lack a well-tuned architecture capable of effectively overseeing diverse IoT endpoints, reducing end-to-end latency, and implementing power-saving offloading policies. Therefore, the primary goal of the present research is to develop an IoT-supported framework for low-latency and low-energy offloading tailored to contemporary water distribution networks, while directly addressing the previously identified research hurdles.

To the best of our understanding, no comprehensive design and evaluation model exists in the published literature for location-tailored IoT communication networks that prioritize optimized delay reduction alongside energy-efficient offloading. The modeling approach introduced here holds broad applicability for crafting communication networks customized to any given site, marking a key innovation of this contribution.

- Pinpointed various operational hurdles in a Water Distribution Network (WDN) within a secondary-tier metropolitan area and formulated corresponding communication network components and layout configurations.
- Constructed a wireless signal propagation model customized to the chosen study zone, based on land-use and land-cover distributions, along with considerations for network dependability, channel characteristics, and communication metrics; external weather influences on the network are assumed negligible.
- Combined delay and energy performance evaluations for the formulated wireless channel model within an IoT-oriented communication setup designed for WDNs in secondary metropolitan environments.
- Presented algorithms for grouping nodes and assigning them between layers, incorporating delay-minimizing and energy-optimized offloading within the IoT communication structure for WDNs in secondary metropolitan settings.
- Included quantitative evaluations of the suggested algorithms, encompassing various delay- and energy-aware offloading cases.

This article seeks to establish IoT-enhanced water distribution frameworks suitable for smart urban environments. The core issues examined focus on methods to alleviate latency and power constraints in the underlying communication infrastructure, thereby extending the operational lifespan of the various IoT nodes involved. The discussion covers data exchange pathways from edge-level devices up to the central cloud platform.

Design Considerations for an IoT-Enabled WDN

Despite possessing sufficient overall water resources to support long-term distribution in India, certain regions continue to experience acute shortages, even failing to meet basic daily requirements. Water is gathered from primary sources into large storage reservoirs, then channeled via the distribution grid to individual user connections. **Figure 1** illustrates the overarching water supply arrangement. A WDN comprises five hierarchical levels: end consumers, local supply nodes, intermediate sub-reservoirs, purification facilities, and the primary storage reservoir.

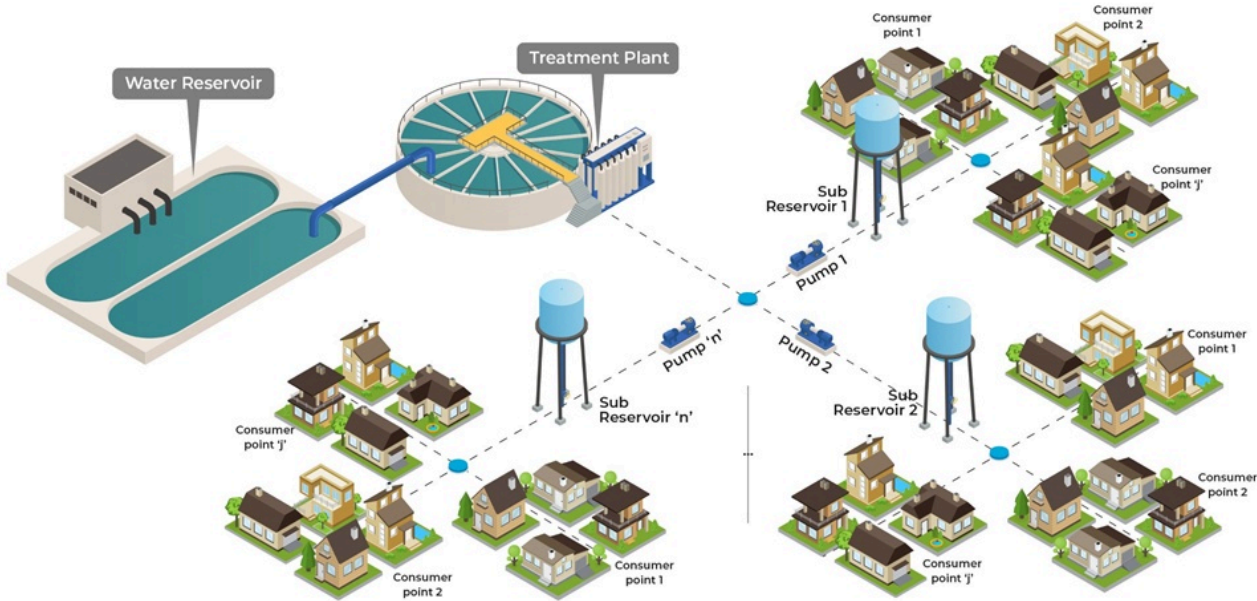


Figure 1. The Generic Water Distribution Network.

The root cause of water supply deficits often lies in deficiencies within the reservoir-to-consumer delivery framework. Contributing factors include deteriorating infrastructure, persistent leaks, and inadequate oversight of network operations. To address these shortcomings, an intelligent IoT-driven water distribution system is advocated to promote sustainable resource management. Such an IoT-integrated network enables real-time observation, regulation, and automation of the WDN, thereby improving water allocation efficiency for both residents and municipal authorities. For this purpose, the distribution setup can be modeled as a multi-tier, heterogeneous IoT architecture comprising various interconnected components. **Table 1** lists the principal entities incorporated into the network design process.

Table 1. Water distribution entities.

Sl. No.	Entity	Parameters	Main function/Purpose	Issues encountered
1	Pipe	Flow, velocity, redness coefficient, and friction	Measures total water consumption	Leakage/breakage
2	Pressure and friction	Pressure and friction	Indicates whether the minimum required pressure is maintained	Leakage/breakage, theft
3	Valve	Control	Controls water flow	Control problems/malfunction
4	Pump	Pressure	Controls pumping pressure	Dry running or overflow

5	Water tower or storage structure	Level	Monitors water storage quantity	Overflow or leakage
---	----------------------------------	-------	---------------------------------	---------------------

Several critical design limitations must be factored into the architecture to ensure effective construction. The key constraints considered when developing the multi-layer heterogeneous IoT framework are detailed below.

Water supply vs. consumption discrepancy

After purification, reservoir water is channeled into secondary storage tanks (sub-reservoirs). From there, it is delivered via network junctions to end-user locations. Although the distribution follows a logically structured, tiered framework designed for optimal performance, specific zones still face water shortages to meet daily requirements. This occurs primarily because of pipeline leaks, unauthorized hookups, and deliberate water pilferage. Detecting mismatches between supplied and used volumes becomes possible by installing flow-measurement devices at strategic points. These flow sensors should be positioned at levels 1 and 5 throughout the distribution system. The layout of flow-tracking sensors and their associated controllers across different tiers is shown in **Figure 2a**.

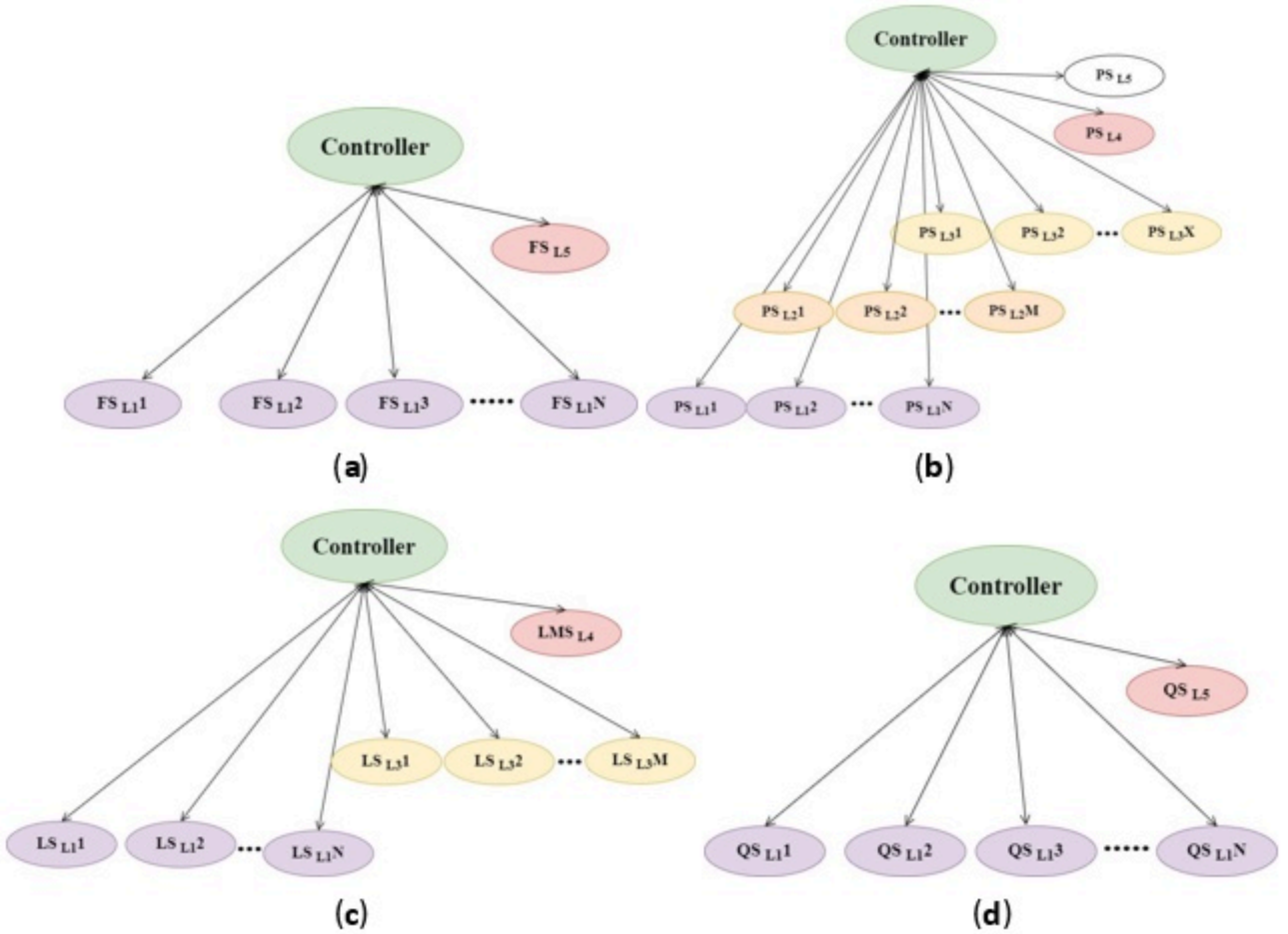


Figure 2. Sensors in Water Distribution Network for monitoring (a) Flow, (b) Level, (c) Quality, and (d) Pressure.

Inefficient operation caused by human factors

At present, manual operators handle tank refilling, as well as the activation and deactivation of pumps and control valves. Inevitably, either through oversight or external constraints, inefficient or erroneous actions occur during these processes. Deploying an automated water-level detection mechanism across the system can significantly reduce, or even eliminate,

such operational mistakes. The sensors would continuously monitor the water level in each tank and automatically trigger the pump to start or stop once the desired fill level is reached. This automation helps conserve both water resources and electrical power, achieving better overall energy efficiency. Level-monitoring sensors are required at levels 1, 3, and 4 of the distribution infrastructure. The configuration of these level sensors and their controllers at each designated level is illustrated in [Figure 2b](#).

Monitoring and ensuring water quality

Once treated at the main reservoirs, water is held in intermediate storage tanks before being sent to downstream distribution zones. Maintaining high water quality is of utmost importance, as contamination can lead to a wide range of health concerns. As drinking water travels from its source to entire neighborhoods, any harmful chemical residues it carries may produce serious, persistent effects on residents' well-being. For this reason, water must undergo quality checks using specialized sensors before being stored in the tanks. These quality-assessment sensors need to be installed at levels 1 and 5 within the network. The positioning of quality sensors along with their control units at the relevant levels is shown in [Figure 2c](#).

Identifying unauthorized water withdrawal from pipelines

Despite ongoing manual oversight of pipeline connections and consumption patterns, illicit tapping and water stealing persist through hidden or illegal connections. Such activities disturb the balance of the entire supply system and leave certain areas without adequate water for essential purposes. Pressure sensors installed at key junction points can reveal these theft incidents—an unexpected pressure drop or abnormal pressure behavior in authorized lines often indicates an illegal draw-off. Pressure sensors should therefore be placed at every level of the water distribution network. The arrangement of pressure sensors and controllers throughout all levels is depicted in [Figure 2d](#).

Automated detection of pipeline defects

Pipeline damage or deterioration is one of the leading contributors to discrepancies between water supplied and water actually consumed. Crack-detection sensors can automatically locate these structural issues. Prompt identification and repair of damaged sections would substantially reduce water losses from leakage.

Fully automated consumer billing

Flow data collected from sensors at individual consumer endpoints enables the system to automatically compute usage volumes and generate accurate, up-to-date billing records for each user.

Enhanced and sustainable water allocation

The comprehensive integration of the sensors described above into the water distribution infrastructure will effectively tackle the persistent problems currently observed and support long-term, environmentally responsible water delivery throughout the network.

IoT-Enabled Water Distribution Network

To overcome the limitations in the existing water distribution setup, appropriate sensors must be installed at critical points where monitoring is essential, with a smart IoT framework linking all these sensor-integrated components. The overall system includes sensors, an IoT gateway, cloud infrastructure, data analytics capabilities, and a user dashboard. Sensors, such as those for pressure, level, and flow, generate analog or digital signals and connect to the IoT gateway, which forwards the gathered information to the cloud via 4G wireless links. Once in the cloud, the information undergoes processing and becomes accessible through the user interface.

The core need is to reliably upload pressure, flow, and level readings from the sensor devices to the cloud platform. Consider 'N' arbitrary locations along buried pipelines equipped with flow and pressure sensors. The cloud server or central monitoring station resides far from these sensor sites. Data from the sensors reaches this distant monitoring hub via the gateway unit. To achieve reliable and efficient data transfer, various transmission methods from sensor devices to the control center have been evaluated. These approaches are explored thoroughly in the following sections.

At the base layer of the architecture, multiple sensor nodes operate, including pressure monitoring sensors (PMS), level monitoring sensors (LMS), flow monitoring sensors (FMS), and various quality monitoring sensors (QMS). Details on these sensor types appear below:

1. PMS: Pressure sensors are installed underground directly under pipelines to track water pressure on an ongoing basis.
2. FMS: Flow sensors are similarly placed underground beneath pipelines to measure water flow rates continuously.
3. LMS: Level sensors are positioned inside storage tanks at primary collection sites to determine the current volume of stored water.
4. QMS: Quality monitoring sensors (QMS) track key water parameters. Specific sensors can be placed at strategic points or delivery outlets to measure indicators like pH, iron levels, fluoride concentration, and calcium content.

Each of these sensors connects to microcontrollers that relay the collected readings to the gateway through 4G networks. A real-time monitoring and control system for the water distribution network (WDN) typically relies on three main node categories: edge nodes, fog nodes, and the cloud layer. Edge nodes handle initial data capture and preliminary processing before passing information to fog nodes, which serve as distributed control hubs in the communication setup. Fog nodes function as signal repeaters and network coordinators. They do not require direct connectivity with every edge node, so they relay aggregated data onward to the cloud. For reliable redundancy, fog nodes should be strategically positioned across several tiers.

The overarching IoT framework features hierarchical layers: edge nodes, fog nodes, and the cloud server. Edge nodes incorporate various sensor modules, microcontrollers, and wireless communication hardware. Aggregated sensor data is forwarded to fog nodes for further processing. The uppermost application layer is the cloud server. In-depth descriptions of each node type follow in subsequent sections.

Edge node

Raw data from various sensors undergoes local computation, simplification, and packaging before being forwarded to the subsequent tier—the fog node.

Fog node

Multiple fog nodes, organized by geographic area, collect preprocessed and condensed data streams from edge nodes. These inputs are further refined through additional calculations and combined before transmission to the cloud server.

Cloud server

The uppermost tier in the IoT communication stack is the cloud server. It gathers refined data streams forwarded from numerous fog nodes after local processing. The cloud platform features advanced visualization tools that can forecast trends, conduct in-depth analysis, and identify faults across the water network in the target area. Machine learning algorithms embedded in the cloud enable behavioral pattern recognition and future demand prediction.

This provides a broad outline of the IoT-supported water distribution network. To establish a functional IoT system for the WDN, a robust communication pathway is essential to ensure seamless and efficient data exchange between the sending and receiving ends. Within the designated study region, suitable wireless signal propagation models must be determined, along with techniques to minimize transmission delays and overall energy usage. The practical considerations for developing an optimal IoT communication setup—one that reduces latency and power demands while accounting for site-specific factors such as propagation characteristics and inherent obstacles—are covered in the following section.

Case Study: Designing an IoT-Enabled Water Distribution Network in the Study Area

Developing an IoT-supported water distribution network (WDN) that achieves low latency and reduced energy consumption must begin with a thorough analysis of the target region's geographical features. This geographical insight enables accurate prediction of the appropriate wireless communication model for the area—an essential step in constructing a reliable IoT framework. Outcomes from channel modeling can then be applied to optimize both transmission delay and power usage across the network. A comprehensive discussion follows in the subsections below.

Study area

According to the 2011 census, the Kochi Municipal Corporation serves a population of 677,381 residents across 94.88 sq. km, divided into 74 wards. In recent decades, rapid urbanization has driven a sharp rise in population growth [22]. Within the broader framework of urban sustainability, a truly creative city is characterized by several key elements, including a healthy environment. This encompasses clean air and water, effective sanitation, abundant green spaces, and accessible recreational zones. Modernizing the water supply infrastructure, encouraging rainwater harvesting, reducing overall consumption, maximizing water reuse, and promptly identifying and fixing leaks are all critical measures. Kochi has already formulated its own water policy, marking it as the pioneering effort by any local self-government authority in India [23]. The study region of Kochi is illustrated in **Figures 3 and 4**.

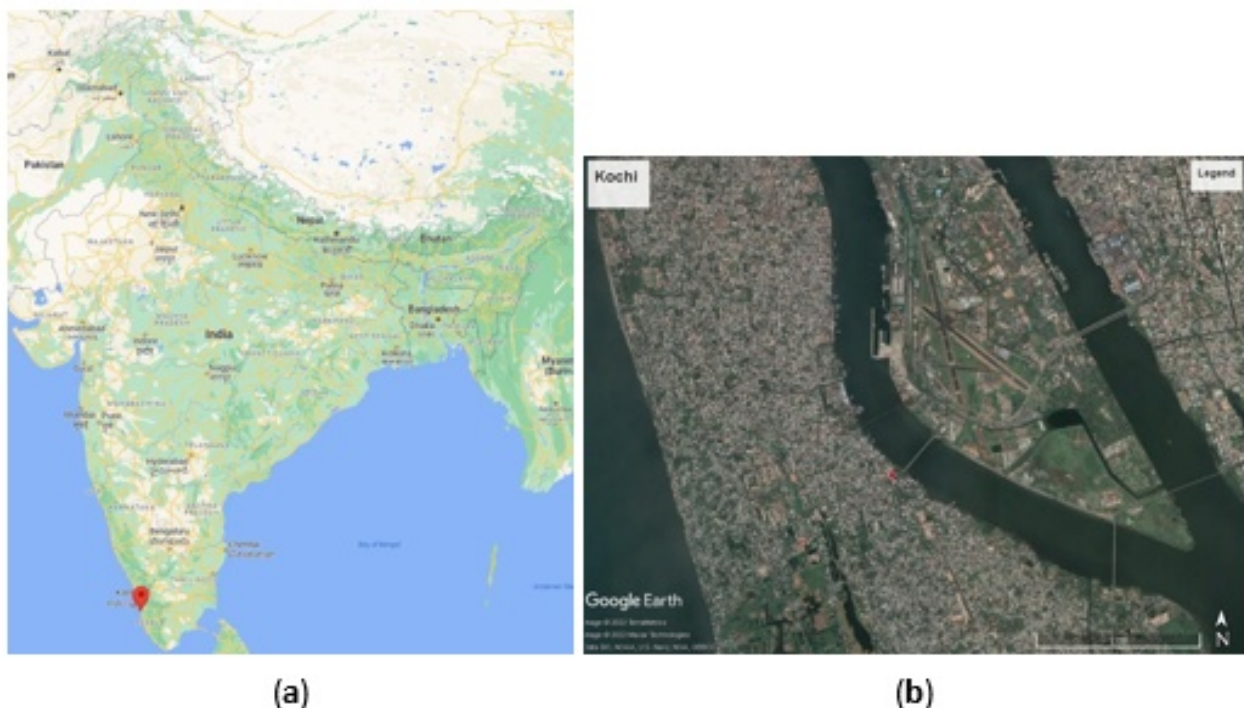


Figure 3. Study area (a) India and (b) Kochi.

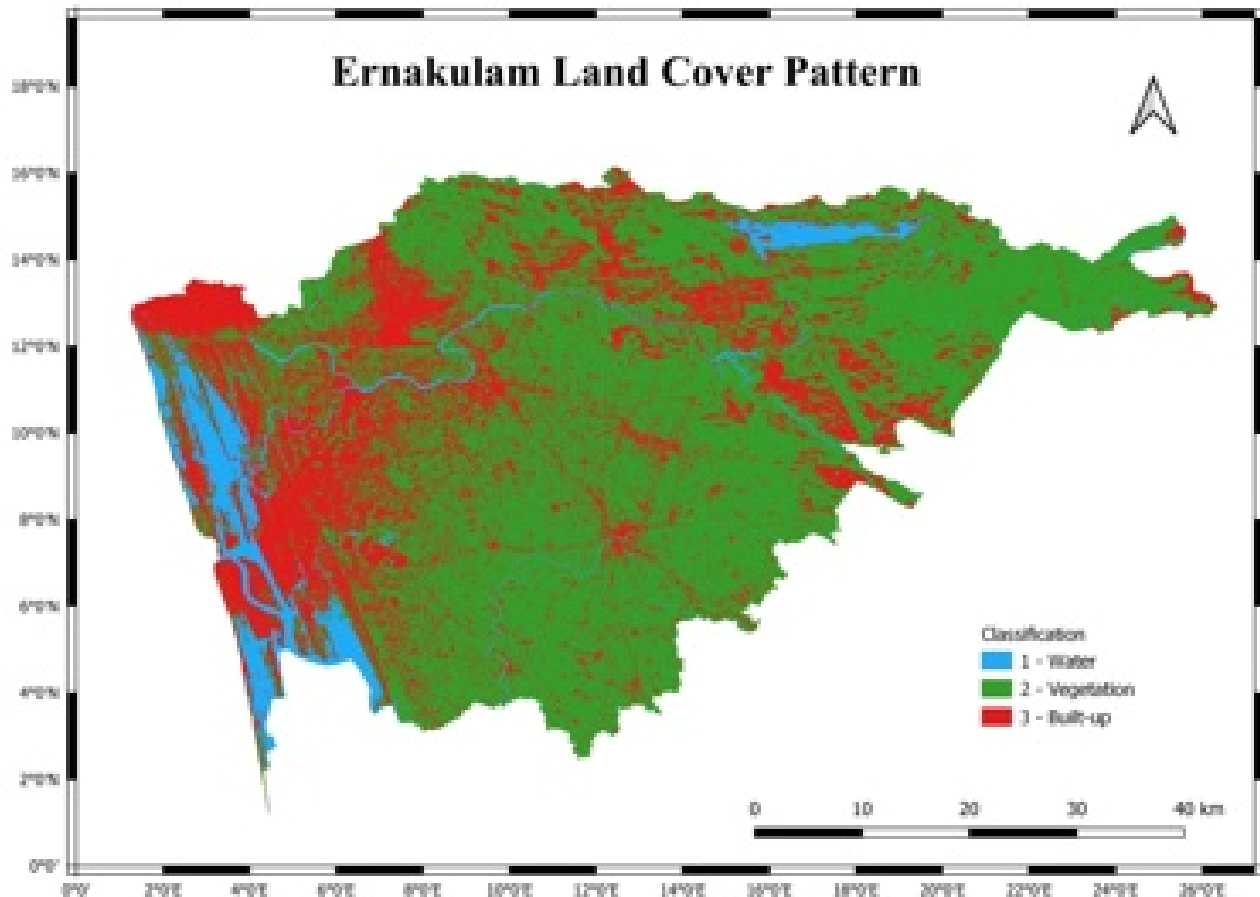


Figure 4. Land cover classification of the study area.

The water distribution system in Kochi encounters numerous difficulties, including the absence of a continuous 24/7 supply, irregular and inconsistent delivery, insufficient sewage treatment facilities, declining groundwater levels and deteriorating quality, ongoing population expansion, widespread water contamination, shifts in land-use patterns, limited resilience to multiple hazards in water infrastructure, and the lack of climate-responsive forecasting tools for managing water quantity and quality [24]. While these represent common obstacles faced by many WDNs, the most pressing issue specific to the Edakochi water distribution network is the advanced age of its infrastructure. The primary supply mains were installed over 40 years ago, and subsequent urban development—including new buildings constructed directly above these lines—has left critical information about larger pipelines largely unavailable. Restricted physical access to the mains, combined with scarce documentation, stems from intense urbanization, high population density, and extensive built-up zones.

An IoT-based communication architecture for the water distribution network can effectively mitigate these various challenges. In Kochi, where the vast majority of residents rely on piped water, maintaining robust infrastructure is a top priority. Consequently, real-time information from any point in the supply system is highly valuable. This makes a delay-tolerant communication network indispensable. The layout of the WDN branches in Kochi evolved organically in response to urban expansion rather than following deliberate, expert-driven planning. As a result, the network contains a high number of interconnections and junctions, which in turn leads to a correspondingly large number of edge nodes. For this reason, an energy-optimized communication system is particularly important in this setting. Implementing a delay- and

energy-efficient IoT-enabled monitoring and control system for the water distribution network could resolve most of the existing challenges.

Site-specific propagation model selection criteria

This section determines the most suitable communication propagation model for the IoT-supported water distribution system in Kochi. As a suburban city in Kerala, Kochi exhibits diverse land-use patterns, including patches of vegetation alongside built-up zones featuring low-rise and high-rise structures, metro rail lines, and roadways. The effectiveness of the IoT communication setup for the WDN largely hinges on selecting an appropriate path-loss model tailored to the local conditions. Path loss arises from multiple factors, including free-space attenuation, refraction, diffraction, reflection, aperture-to-medium coupling losses, and absorption. Additional influences stem from terrain variations, the surrounding environment, propagation medium characteristics, the separation distance between the transmitter and receiver, and antenna heights and positions.

Land cover in the study region was mapped and evaluated using QGIS software equipped with the open-source Semi-Automatic Classification Plugin (SCP) [25]. This plugin supports supervised classification of satellite remote sensing imagery by retrieving free datasets from sources such as Landsat and Sentinel, followed by preprocessing and post-processing steps. Multispectral imagery is analyzed to generate detailed land-cover maps. Classification was conducted on LANDSAT images acquired in January 2021, leveraging tonal and textural differences influenced by elevation-based vegetation patterns and topographic contours. The imagery was classified into ground features, including built-up areas, vegetation, and water bodies. Supervised classification used the maximum likelihood classification (MLC) algorithm, with training samples drawn from the seven-band Landsat data using ground reference points for each land use/land cover (LULC) category. Training areas were chosen based on insights gained from comprehensive field surveys and in-depth on-site investigations of the region. The resulting LULC classification for the study area is presented in [Figure 4](#).

Built-up zones represent a distinctive urban land-use category spanning all of Kochi, subdivided into three urban classes that reflect varying settlement layouts, population densities, and specialized zones like commercial exhibition sites or industrial districts [26]. Analysis of the land cover map reveals a mix of vegetation, water bodies, and extensive built-up development. Given that Ernakulam district includes both urban centers and rural villages, the chosen propagation model must accommodate effects typical of urban, suburban, and rural settings [27]. An overview of propagation model types and their key features is provided in [Table 2](#). The emergence of geographic information systems has enabled efficient storage, indexing, and manipulation of terrain, clutter, and land-use data, improving frequency planning and network management [28]. A propagation model describes the interaction of radio waves with the receiver’s surroundings. The fundamental Friis transmission equation governs signal propagation between a transmitting and receiving antenna [29]:

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{1}$$

Where P_t is the transmitting power, G_t is the gain of the transmitting antenna, G_r is the gain of the receiving antenna, λ is the operating wavelength, d is the distance between transmitting and receiving antennas, and L represents the transmission loss.

Table 2. Propagation models and their parameters.

Model Name	Frequency	Application/Environment
HATA	2500 MHz	Urban areas, shrublands, and open terrain
COMSITE	Up to 2000 MHz	Areas with vegetation cover

OKUMURA	150–1920 MHz	Irregular terrain with environmental clutter
LEE	1–100 MHz	Forested regions
ICEPACK	1.5–30 MHz	Propagation through the ionosphere
ASAPS	1–45 MHz	Propagation through the ionosphere

Free-space transmission loss (FSTL) [30, 31] is expressed as:

$$FSTL = \frac{(4\pi d)^2}{\lambda^2} \quad (2)$$

Rearranged into decibels (dB):

$$FSTL = 32.4 + 20 \log_{10} f_{MHz} + 20 \log_{10} d_{km} \quad (3)$$

FSTL serves as the baseline line-of-sight propagation model, depending solely on distance and representing an ideal scenario. It excludes multipath effects and losses caused by ground clutter, foliage, structures, or other obstacles. Consequently, FSTL is unsuitable for the study area due to its land cover characteristics (Figure 4). The Hata-Okumura model, which incorporates extra attenuation terms for varying urban densities, offers separate formulations for urban, suburban, and rural scenarios, with constraints based on distance and frequency.

For suburban environments [32]:

$$L_U = 69.55 + 26 \log_{10} f_{MHz} - 13.82 \log_{10} h_b - C_H + (44.9 - 6.55 \log_{10} h_b) \log_{10} (d) \quad (4)$$

Where L_U = Path loss in urban areas (dB),

h_b = Height of base station antenna (m),

h_m = Height of mobile antenna (m),

f = Frequency of transmission (MHz),

C_H = Antenna height correction factor (dB),

d = Distance between the base and mobile stations (km).

For small and medium-sized urban cities, the following equation applies [33].

$$C_H = (1.1 \log_{10} f_{MHz} - 0.7) h_m - (1.56 \log_{10} (f) - 0.8) \quad (5)$$

The study area falls into this classification. Therefore, the Hata-Okumura model is well-suited for this context. The chosen propagation path-loss model for the region was simulated using the free web-based tool Radio Mobile by VE2DBE [34] specifically for the Edakochi study area. This simulator generates a propagation profile that incorporates transmission

parameters, channel characteristics, communication settings, associated losses, and local land cover data. **Figure 5** illustrates the connection between the transmitter and receiver.

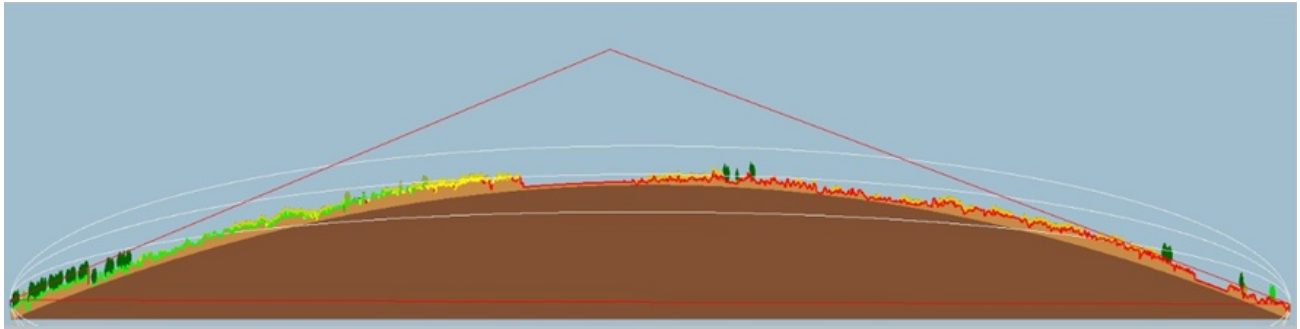


Figure 5. Propagation model.

Simulation parameters and outcomes for the source (transmitter) and destination (receiver) are presented in **Figure 6**. In addition, the coverage prediction tool, Cellular Expert, incorporates several sophisticated algorithms for modeling microwave point-to-point, point-to-multipoint, fixed, and mobile radio networks. These algorithms draw on ITU-R, ETSI, COST 231, and IEEE standards and guidelines. The models can be fine-tuned using drive-test measurements and adapted to match particular terrain profiles and land-use categories. Supported propagation models span distances from a few meters to 150 km and operate across frequencies from 20 kHz to 100 GHz. Cellular Expert supports Line-of-Sight, Hata, COST 231 [35], Walfisch-Ikegami [36], and SUI-type models [37] and provides flexibility to incorporate user-defined prediction methods. A standout feature is its capability to combine multiple prediction models based on prevailing environmental conditions.

Edakochi (1)		(2) Amritapuri	
Latitude	9.912195 °	Latitude	9.085803 °
Longitude	76.281752 °	Longitude	76.491880 °
Ground elevation	9.7 m	Ground elevation	6.0 m
Antenna height	6.6 m	Antenna height	6.6 m
Azimuth	165.90 TN 168.03 MG °	Azimuth	345.94 TN 348.21 MG °
Tilt	-0.43 °	Tilt	-0.42 °
Radio system		Propagation	
TX power	43.98 dBm	Free space loss	131.06 dB
TX line loss	3.00 dB	Obstruction loss	58.53 dB
TX antenna gain	6.00 dBi	Forest loss	1.79 dB
RX antenna gain	2.00 dBi	Urban loss	0.00 dB
RX line loss	0.50 dB	Statistical loss	4.10 dB
RX sensitivity	-113.02 dBm	Total path loss	195.48 dB
Performance			
Distance	94.736 km		
Precision	47.4 m		
Frequency	905.000 MHz		
Equivalent Isotropically Radiated Power	49.882 W		
System gain	161.50 dB		
Required reliability	70.000 %		
Received Signal	-147.01 dBm		
Received Signal	0.01 μV		
Fade Margin	-33.98 dB		

Figure 6. Propagation model results.

Delay-efficient offloading in IoT-enabled WDN

Designing an IoT-supported water distribution network (WDN) requires careful integration of the propagation model specific to the study region. Delay-efficient offloading refers to shifting data processing and computational workloads from local sensor nodes to remote cloud resources, thereby reducing overall processing latency. Such strategies can substantially improve network performance. Key factors to address during network design include task recognition, offloading decisions, task migration, computational offloading strategies, and energy management to ensure high service quality.

Successful data transmission occurs when the signal-to-interference ratio (SIR) measured at the receiving node exceeds a predefined threshold θ . If this condition is not met, retransmission is initiated. Let η represent the spectral efficiency of the wireless link connecting edge device e_i to fog device f_i' . The required transmission power from edge node e_i to fog node f_i' over link j is denoted $P_{ti \rightarrow i'}$, while the received power at fog node i' is P_{rj} . The spectral efficiency of the link η [38] is given by:

$$\eta = \log_2 1 + \frac{P_t^{i \rightarrow i'} \delta}{P_i + B \times P_{n0}} \quad (6)$$

Where P_i is the interference power spectral density, P_{n0} is the noise power spectral density, and δ accounts for combined losses including channel attenuation and shadowing effects.

$$D_{\tau} = D_{prop} + D_t + D_{proc} + D_q \quad (7)$$

Where D_{τ} is the total delay, D_{prop} is the propagation delay, D_t is the transmission delay, D_{proc} is the processing delay, and D_q is the queuing delay.

- D_{prop} : As shown in Eq. 4, propagation delay arises from channel path loss, which grows with increasing distance between antennas [39].
- D_t : This represents the time needed to deliver data bits or packets to the receiver. It depends on the channel capacity $C_{f_i \rightarrow f'_i}$ between e_i and f'_i , the available bandwidth B , and the data size b_i . The delay is influenced by the transmission rate T_R , message length M , and channel access delay d_{θ} [40].
- D_{proc} : Processing delay is determined by the fog node f'_i characteristics, including fog memory $f_m^{f'}$, clock frequency $f_{cf}^{f'}$, collectively referred to as the fog processing capacity $f_{pc}^{f'}$.
- D_q : Queuing delay is considered negligible in this water distribution network due to relatively low data volumes. Further refinement of this aspect is planned for future enhancements.

$$D_{\tau} = L_U \times \alpha + \frac{C_{f_i \rightarrow f'_i}^{i \rightarrow i'}}{B} b_i T_R M + f_{pc}^{f'_i} \quad (8)$$

Where α serves as the constant parameter governing propagation delay.

From Eq. 8, it is clear that the communication delay in the WDN is strongly influenced by the selected propagation model, path-loss characteristics, and a range of network and communication parameters outlined above. A propagation environment with high noise or excessive path loss can degrade spectral efficiency, resulting in increased communication delays. Therefore, the IoT-enabled WDN must be engineered with full consideration of local design constraints and the prevailing propagation model. Reducing network delays is essential for achieving reliable, efficient communication. Techniques for minimizing delay are explored in the subsequent subsection.

Delay modeling

Consider the multilevel network architecture illustrated in Figure 1. Following local computation, data originating from various edge nodes must be forwarded to the central cloud server. Accordingly, readings collected by sensor types such as FMS, LMS, QMS, and PMS undergo necessary processing at the edge before transmission to the server. Transmissions occur either continuously or are triggered by specific events at regular discrete intervals to minimize overall system latency. Each sensor subsystem contributes its own delays across three main stages: sensing, local processing, and data transmission. Sensing delay arises as sensors measure physical parameters—including pressure, water level, flow rates, and similar variables—using specialized detection devices. Processing delay occurs after initial computations when converting raw sensor outputs into equivalent electrical signals or digital values. Transmission delay is the time required to send prepared data across the network. The aggregate delay experienced within the network can be expressed as

$$d_{comp} = d_{sens} + d_{proc} + d_{trans} \quad (9)$$

Where d_{comp} represents the total communication network delay for a single node, d_{sens} is the sensing delay, d_{proc} is the processing delay, and d_{trans} is the transmission delay.

Even after data leaves the node, additional latency is introduced by the communication channel, influenced by available bandwidth and propagation characteristics. Channel delay, therefore, varies as a function, denoted α , of bandwidth limitations and propagation phenomena, and is given by

$$d_{chan} = \alpha \left(d_{band}, d_{prop} \right) \tag{10}$$

Where d_{chan} is the total channel delay, d_{band} reflects delays caused by bandwidth constraints, and d_{prop} captures delays due to propagation effects.

A major contributor to network latency is the sheer number of edge devices. After completing local data processing, each edge device forwards its results straight to the cloud server. Let N_e denote the total number of edge devices in the system. Because every edge node performs independent processing and initiates its own transmission, the cloud server must handle N_e separate instances of data arrival, each involving both processing and transmission overhead. The effective capacity of the cloud server under these conditions can be described using the relation provided below:

$$C_{cloud} = N_e \left(d_{comp} + d_{chan} \right) \tag{11}$$

$$C_{cloud} = N_e \left(d_{sens} + d_{proc} + d_{trans} + \alpha \left(d_{band}, d_{prop} \right) \right) \tag{12}$$

Consider the transmission parameters associated with each edge node and the corresponding channel characteristics as listed in [Table 3](#).

Table 3. Parameters for data transmission.

Parameter	Edge node	Channel values	Fog node	Cloud server
Operating frequencies	2–8 GHz	NA	2–8 GHz	2–8 GHz
Transmitting power	2 mW	NA	2 mW	2 mW
Antenna gain	2 dB	NA	2 dB	10 dB
Antenna height	10 m	NA	10 m	10 m
Number of available channels	11	13	13	13
Data rate	20 Mbps	20 Mbps	20 Mbps	50 Mbps
Signal strength	–27 dB	NA	–20 dB	–15 dB
Bandwidth	5–20 MHz	5–20 MHz	5–20 MHz	100 MHz

The cloud server imposes a limit on the maximum number of simultaneous connections to maintain reliable communication with downstream devices. Relevant cloud server properties are also detailed in [Table 3](#). As the number of connected edge devices grows, increased channel congestion and buffer overflow become likely, leading to packet loss and overall network degradation. When all edge nodes transmit directly to the cloud after local computation—as described in [Eq. 12](#)—the system risks communication link failures or cloud server overload due to a sudden influx of large data volumes.

By introducing an intermediate tier, namely fog nodes, between the edge layer and the cloud layer, an additional level of localized data processing and transmission is created. Fog nodes help mitigate link failures and substantially reduce end-to-end network delay. Incorporating intermediate computations at fog nodes effectively addresses the congestion and overload problems outlined above. The optimal number of fog nodes required depends on the specific demands of both the edge devices and the cloud server to ensure a congestion-free, lossless communication environment. In the scenario under consideration, the essential parameters for the fog nodes are presented in [Table 3](#).

Let N_f represent the total count of fog nodes deployed in the system. Under this setup, the maximum throughput capacity of the cloud server can be recalculated based on the fog layer's operational characteristics, as outlined in [Eq. 13 and 14](#). As a result, placing fog nodes at an intermediate level substantially lowers the computational burden, relieves pressure on processing pipelines, and shortens transmission latencies, leading to a more balanced and efficient overall communication infrastructure.

$$C_{cloud} = N_f (d_{comp} + d_{chan}) \quad (13)$$

$$C_{cloud} = N_f (d_{proc} + d_{trans} + \alpha (d_{band}, d_{prop})) \quad (14)$$

For optimal network performance, a consistent set of edge nodes should be assigned to connect exclusively to a single fog node. Once local processing is complete, the fog node aggregates and forwards the consolidated information to the cloud server. Let N_{ef} indicate the quantity of edge nodes routed through a single fog node. The effective handling capacity of one such fog node is then defined by [Eq. 15 and 16](#).

$$C_{fog} = N_{ef} (d_{comp} + d_{chan}) \quad (15)$$

$$C_{fog} = N_{ef} (d_{sens} + d_{proc} + d_{trans} + \alpha (d_{band}, d_{prop})) \quad (16)$$

Assuming ideal channel conditions when edge nodes communicate straight to the cloud server (which possesses a fixed capacity C_c determined by the aggregate number of edge nodes N_e and the per-node data rate D_e), the communication link will fail under the following overload condition:

This defines the breaking point for a direct edge-to-cloud architecture. The risk of link failure can be eliminated by inserting fog nodes as intermediaries between the edge tier and the cloud tier. Consequently, the necessary number of fog nodes is dictated by both the total population of edge nodes and their individual data transmission rates, given a constant cloud server capacity. When multiple fog nodes are introduced, the corresponding critical threshold becomes:

$$N_{f1}D_{f1} + N_{f2}D_{f2} + N_{f3}D_{f3} + \dots + N_{fn}D_{fn} > C_c \quad (17)$$

Where $N_{f1}, N_{f2}, \dots, N_{fn}$ identify the separate intermediate fog nodes, $D_{f1}, D_{f2}, \dots, D_{fn}$ denotes its respective data rate, and C_c denotes the cloud server's capacity. With network latency in focus, the proposed IoT-driven wireless communication framework and its packet routing behavior are examined in the subsection that follows.

Packet flow and delay-efficient offloaded WDN architecture

The configuration of the IoT-supported communication framework can be visualized through the topology illustrations in **Figure 7**. The design organizes communication into three distinct layers: edge nodes, fog nodes, and the central cloud server. An edge node integrates both a sensing module and a wireless communication module. These nodes are positioned at key data-acquisition sites close to the pipeline infrastructure, with numerous edge nodes distributed across the field. After performing initial edge-level computations, the nodes transmit their results to designated fog nodes. The fog nodes perform additional processing before relaying the refined data to the distant cloud server, which serves as the primary monitoring interface or control dashboard and is located remotely from the on-site equipment.

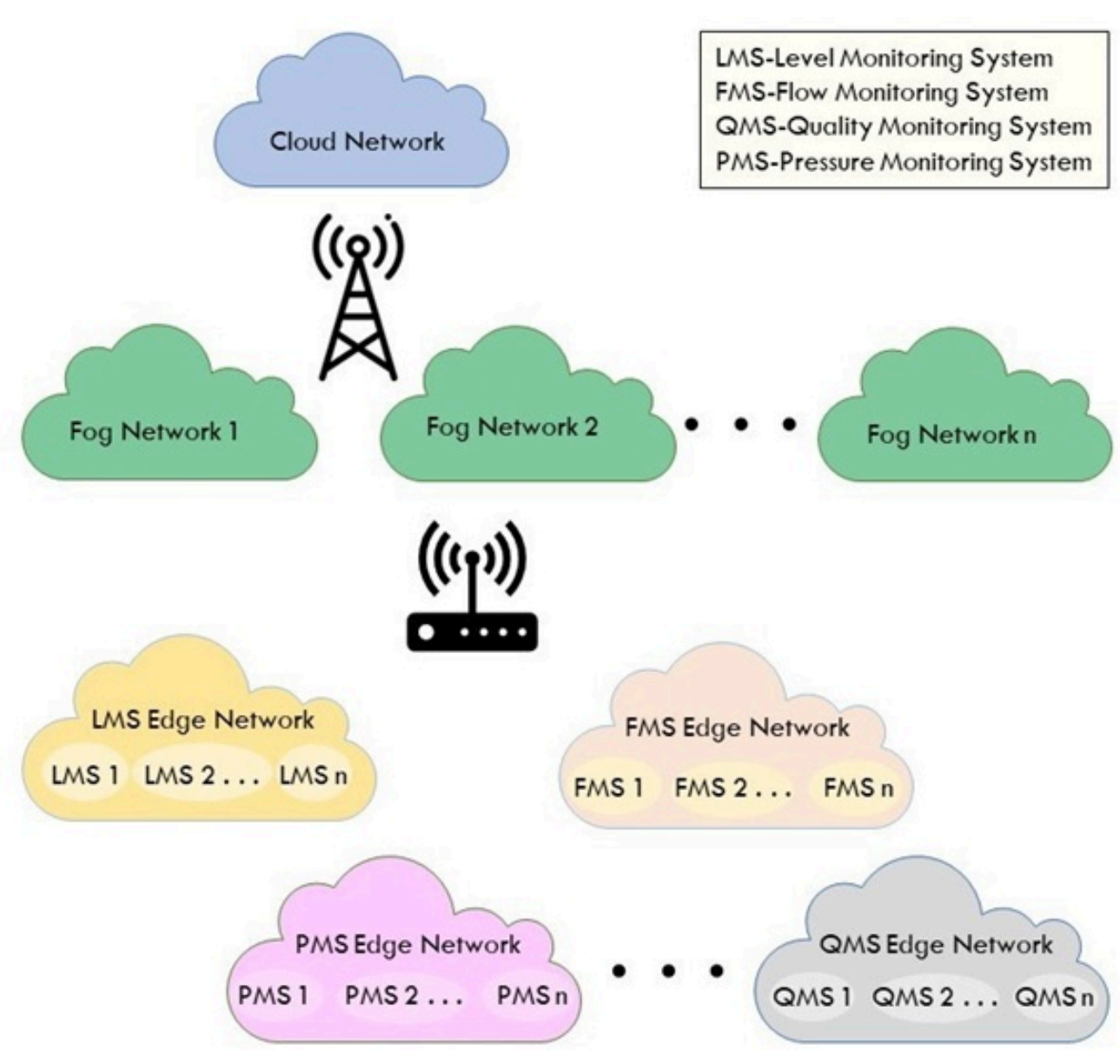


Figure 7. IoT-enabled Communication Network Topology.

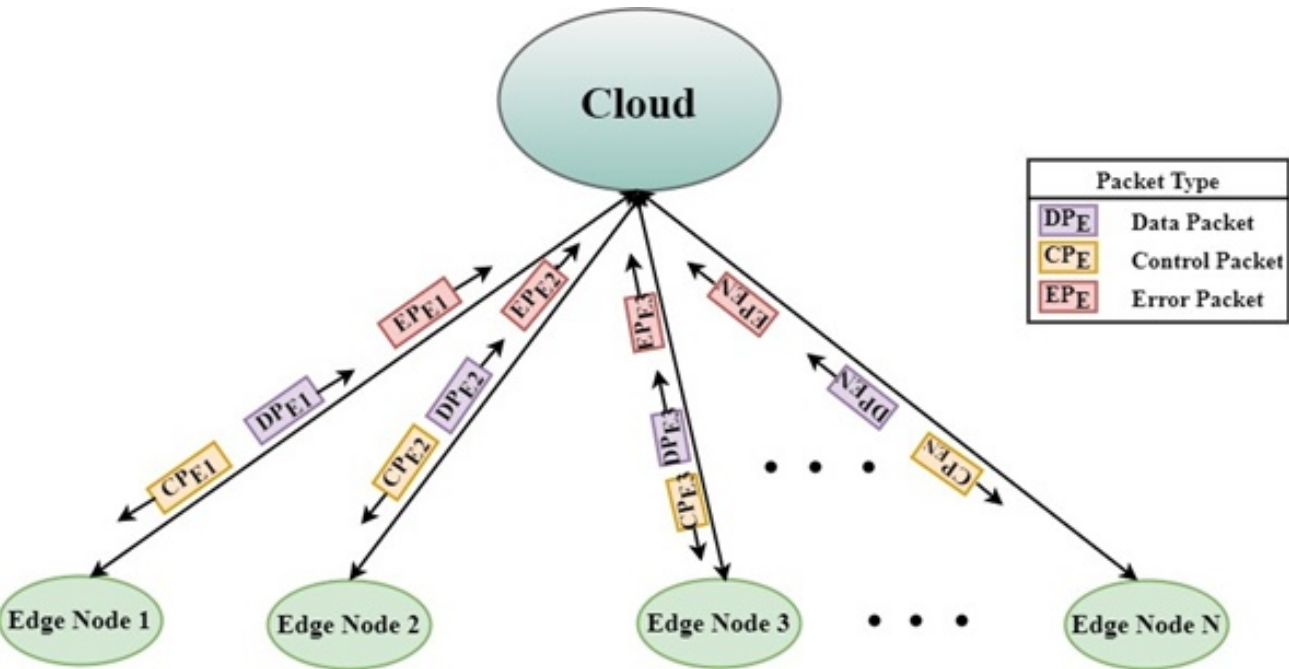


Figure 8. IoT Communication Network Packet Flow Diagram between edge nodes and cloud server.

This layered architecture has been deployed across the edge devices, fog nodes, and cloud infrastructure in the Edakochi study zone. The main packet categories in the IoT communication system are data, control, and error-reporting packets. The full sequence of information exchange from edge nodes to the cloud server is mapped in [Figure 8](#). Data packets encapsulate measurements collected from edge units equipped with PMS, FMS, LMS, and QMS sensors. These devices transform analog measurements into digital form before sending them onward to the fog layer. Control packets are employed to track transmission reliability and overall packet delivery performance. Urgent notifications and alerts are dispatched from edge nodes to fog nodes or directly to the cloud as control packets. Error packets originate from fog or cloud components and are directed back to edge nodes to signal operational faults or malfunctions; they are generated reactively in response to detected events. In a parallel configuration with intermediate fog nodes, the packet routing path from edge nodes to the cloud server is depicted in [Figure 9](#).

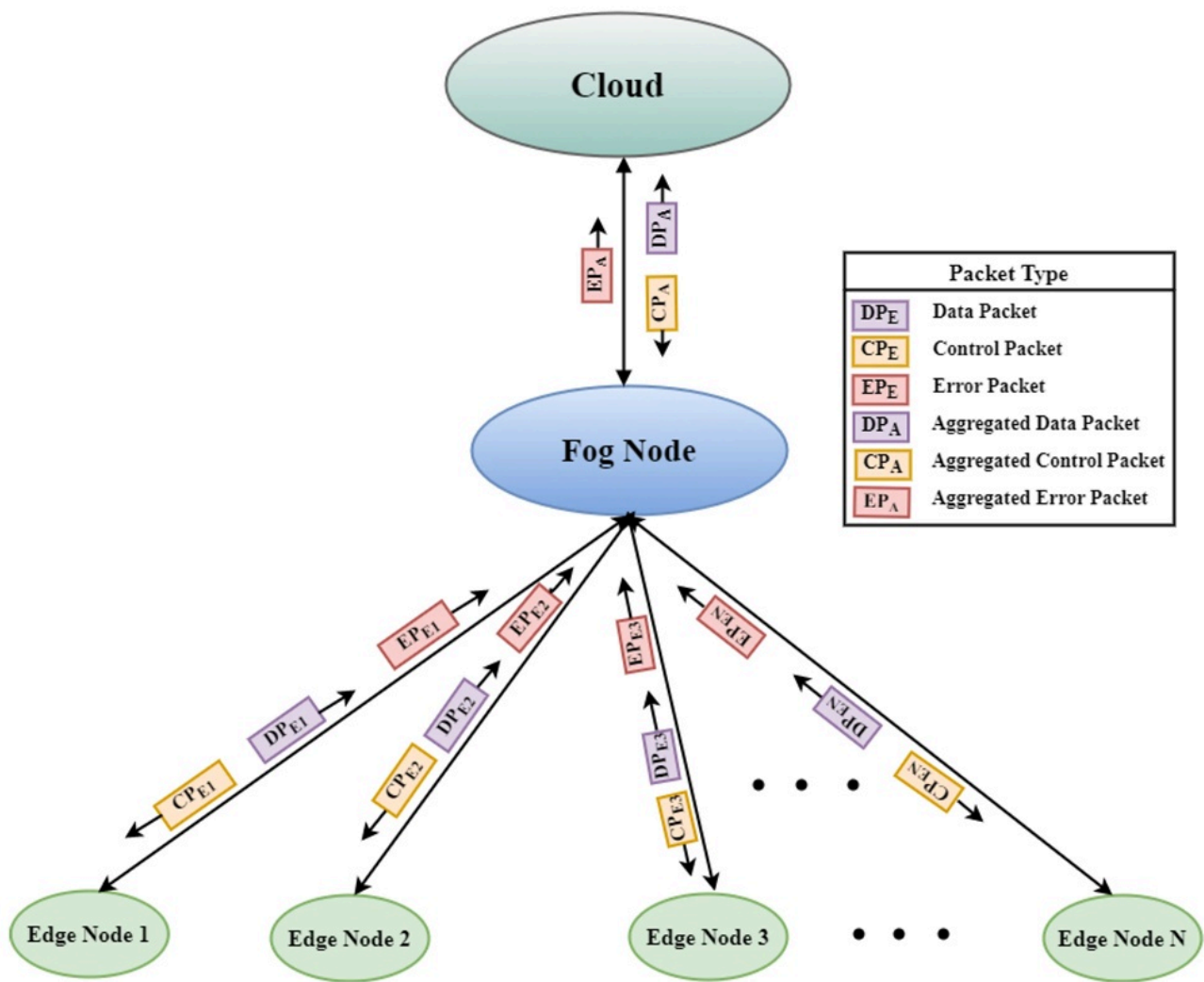


Figure 9. IoT Communication Network Packet Flow Diagram between the edge node and the cloud server using the fog node.

The packet routing illustration for the IoT-supported communication topology with multiple fog nodes is shown in [Figure 10](#). Within this three-tiered hierarchy—comprising the edge, fog, and cloud layers—the primary packet categories are data, control, and error packets. A structured summary of the characteristics of these various packets, as depicted in [Figure 10](#), is provided in [Table 4](#).

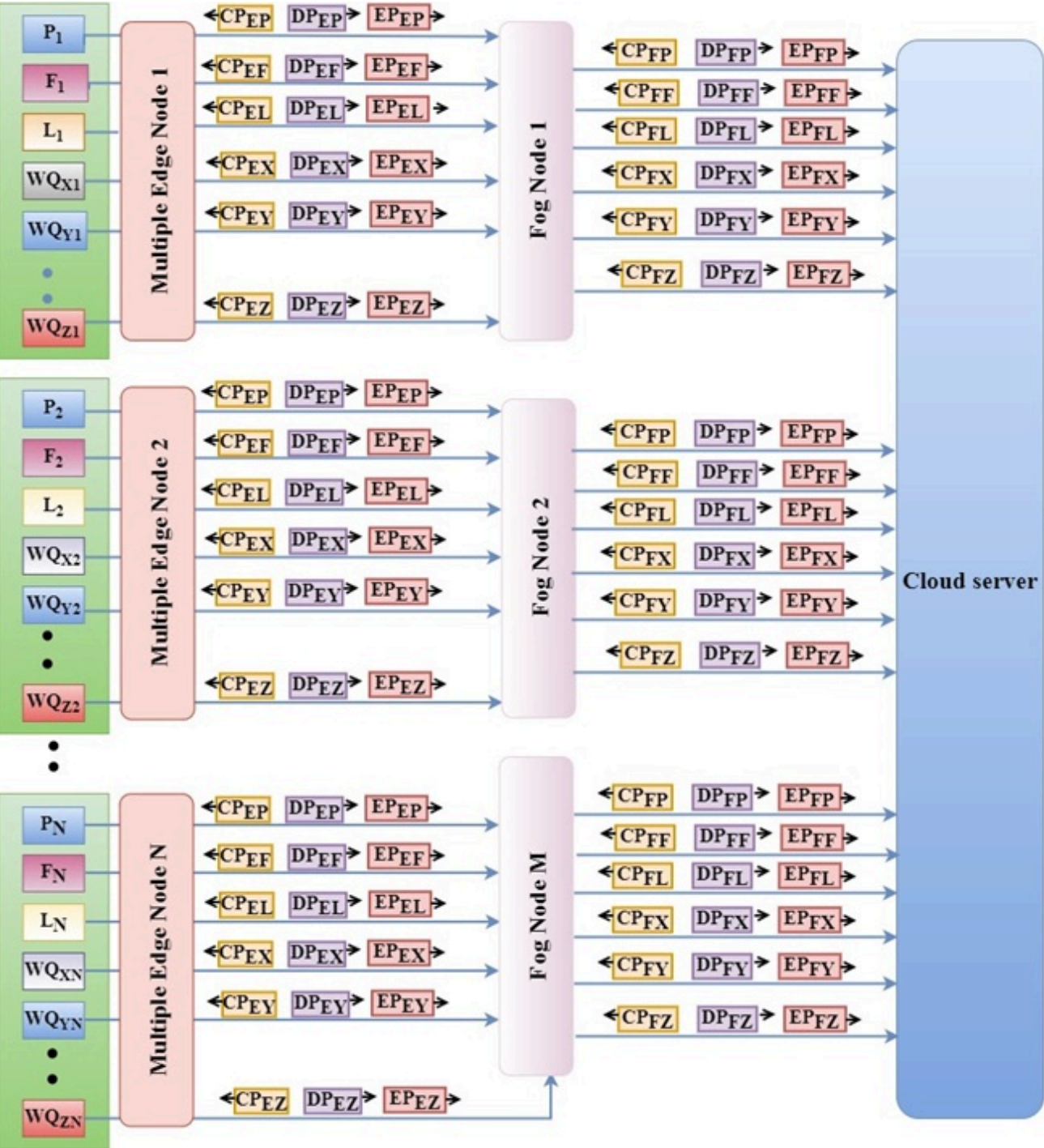


Figure 10. IoT communication network packet flow diagram.

Table 4. Details of the different packets.

Sl. No.	Packet name	Description	Fields in the packet	Path (From → To)	Transmission frequency	Purpose/Functionality achieved
1	DPEL	Data packet from edge node	Water level value, time	LMS (Edge node) → Fog	Every 1 min (after threshold)	Monitor real-time tank water level to prevent overflow

		LMS				
2	CPEL	Control packet from fog to edge node LMS	Control messages (updates, on/off/sleep)	Fog → LMS (Level Monitoring System)	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
3	EPEL	Error packet from edge node LMS	Functional disabilities of sensors	LMS (Edge node) → Fog	Event-triggered	Detect and report sensor sensing failures
4	DPEP	Data packet from edge node PMS	Pressure value, time	PMS (Edge node) → Fog	Every 1 min (after threshold)	Monitor real-time water pressure in the pipeline
5	CPEP	Control packet from fog to edge node PMS	Control messages (updates, on/off/sleep)	Fog → PMS	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
6	EPEP	Error packet from edge node PMS	Functional disabilities of sensors	PMS (Edge node) → Fog	Event-triggered	Detect and report sensor sensing failures
7	DPEF	Data packet from edge node FMS	Flow rate value, time	FMS (Edge node) → Fog	Every 1 min (after threshold)	Monitor real-time water flow rate in the pipeline
8	CPEF	Control packet from fog to edge node FMS	Control messages (updates, on/off/sleep)	Fog → FMS	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
9	EPEF	Error packet from edge node FMS	Functional disabilities of sensors	FMS (Edge node) → Fog	Event-triggered	Detect and report sensor sensing failures
10	DPEQ	Data packet from edge node QMS	Water quality values, time	QMS (Edge node) → Fog	Every 1 min (after threshold)	Monitor real-time water quality parameters in the pipeline
11	CPEQ	Control packet from fog to edge node QMS	Control messages (updates, on/off/sleep)	Fog → QMS	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
12	EPEQ	Error packet from edge node QMS	Functional disabilities of sensors	QMS (Edge node) → Fog	Event-triggered	Detect and report sensor sensing failures
13	DPFF	Data packet from Fog node (flow-related)	Water flow values, time	FMS (Fog node) → Cloud	Every 1 min (after threshold)	Report real-time water flow values to the cloud
14	DPFP	Data packet from Fog node	Water pressure values, time	PMS (Fog node) →	Every 1 min (after threshold)	Report real-time water pressure values to the cloud

		(pressure-related)		Cloud		
15	DPFL	Data packet from Fog node (level-related)	Water level values, time	LMS (Fog node) → Cloud	Every 1 min (after threshold)	Report real-time water level values to the cloud
16	DPFQ	Data packet from Fog node (quality-related)	Water quality values, time	QMS (Fog node) → Cloud	Every 1 min (after threshold)	Report real-time water quality parameter values to the cloud
17	EPFP	Error packet from fog (pressure-related)	Functional disabilities of sensors	Fog → Cloud	Event-triggered	Detect and report sensor sensing failures to the cloud
18	EPFF	Error packet from fog (flow-related)	Functional disabilities of sensors	Fog → Cloud	Event-triggered	Detect and report sensor sensing failures to the cloud
19	EPFL	Error packet from fog (level-related)	Functional disabilities of sensors	Fog → Cloud	Event-triggered	Detect and report sensor sensing failures to the cloud
20	EPFQ	Error packet from fog (quality-related)	Functional disabilities of sensors	Fog → Cloud	Event-triggered	Detect and report sensor sensing failures to the cloud
21	CPFP	Control packet from cloud (pressure-related)	Control messages (updates, on/off/sleep)	Cloud → Fog	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
22	CPFF	Control packet from cloud (flow-related)	Control messages (updates, on/off/sleep)	Cloud → Fog	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
23	CPFL	Control packet from cloud (level-related)	Control messages (updates, on/off/sleep)	Cloud → Fog	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization
24	CPFQ	Control packet from cloud (quality-related)	Control messages (updates, on/off/sleep)	Cloud → Fog	Event-triggered	Ensure the correct operation of sensors, enable remote management, and time synchronization

Energy offloading for IoT-enabled WDN

Higher path loss directly contributes to increased network delay. In turn, prolonged delays extend the active operating periods of nodes and raise the frequency of retransmissions. Channel impairments within the network increase packet loss rates, necessitating repeated retransmissions. In suburban environments, where channel conditions fluctuate and obstacle density is higher, transmission power must often be increased to compensate for increased delay, resulting in higher overall energy consumption across the network. Nevertheless, the primary goal remains to offload and minimize

energy usage. These repeated retransmissions further amplify both delay and energy expenditure. Using Eqs. 7 and 8, network delay can be quantified. Since energy consumption depends on delay and operational duration, the total energy demand of the network can be derived by combining delay calculations with time measurements.

Energy consumption in the communication system is defined as the product of power draw and time. When edge, fog, and cloud nodes remain continuously powered, data exchanges occur at fixed intervals determined by available power reserves. Continuous node activity accelerates battery depletion, underscoring the need for energy optimization. Energy savings are achievable only by shortening active periods while preserving necessary data transmissions. This optimization involves strategically managing power consumption in edge and fog devices by intelligently assigning operational modes. Devices transition between various states—ranging from fully ON to OFF—including SLEEP/IDLE, ACTIVE, MONITOR, COMPUTE, TRANSMIT, and others—to minimize power draw. The complete state transition model linking edge, fog, and cloud nodes is illustrated in Figure 11. The function of each operational state is outlined in Figure 11.

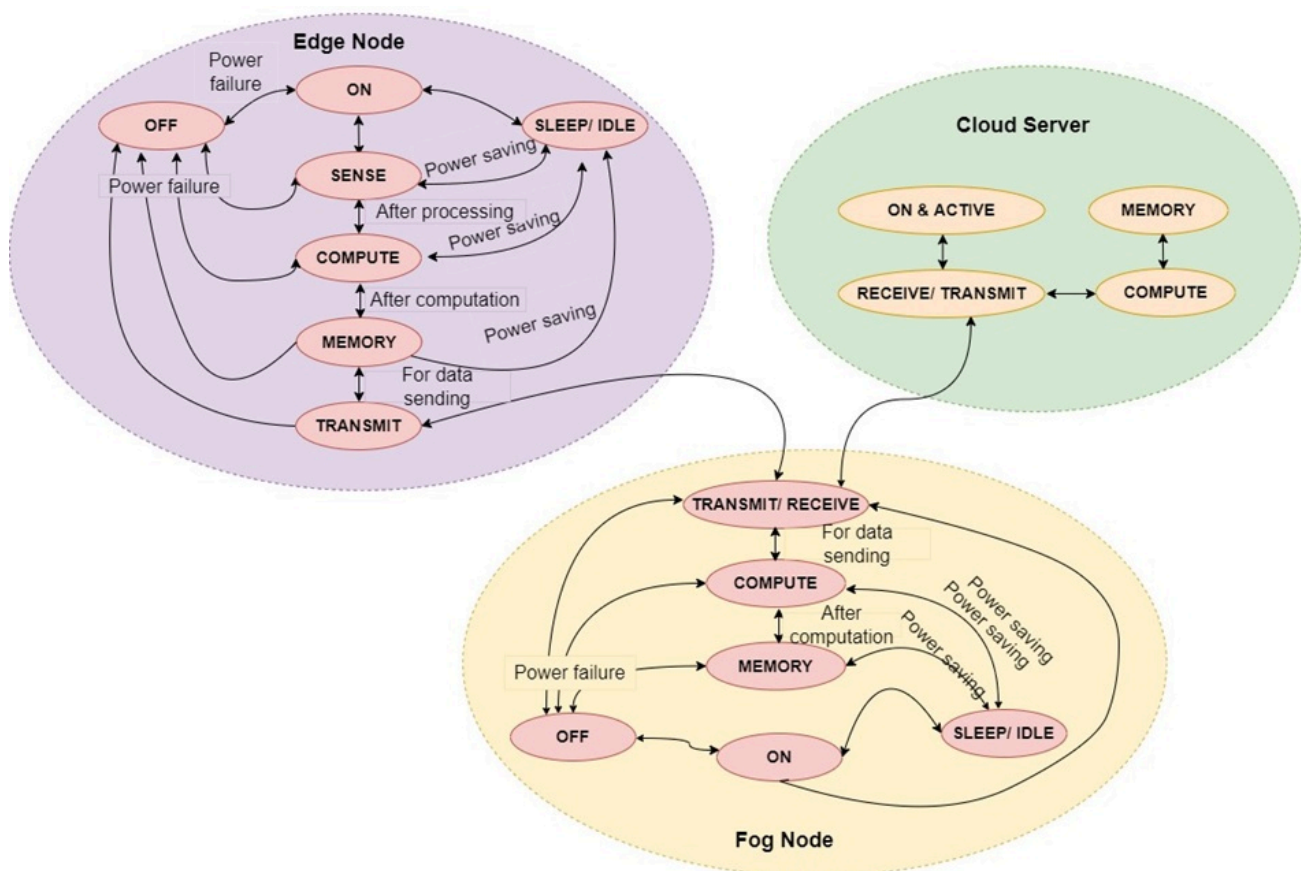


Figure 11. State transition diagram.

ON: Fully powered operational mode for the IoT node, representing the baseline (minimum) power consumption state; OFF: Complete power-down state in which the IoT node is unpowered; SLEEP/IDLE: Low-power temporary shutdown state, even while the node remains nominally powered; this mode consumes the least energy; SENSE: Active sensing mode during which IoT sensors collect measurements from various attached detectors; COMPUTE: Processing mode in which computations and data handling occur following sensor data acquisition; MEMORY: Storage mode where collected sensor information is organized and saved into memory; TRANSMIT: Radio transmission mode with the transceiver and antennas activated for outbound data sending; RECEIVE: Radio reception mode with the transceiver and antennas active for incoming data capture.

Strategies for offloading energy consumption across devices to extend operational lifespan are explored through practical use-case scenarios in the following sections.

Operational strategy 1

Suppose all edge and fog devices remain continuously ON, with transmission and reception occurring every second. In this mode, let PON denote the power draw during the ON state, and EON represent the corresponding energy usage.

$$EON = PON \times tSENSE \quad (18)$$

$$ESENSE = PSENSE \times tSENSE \quad (19)$$

$$ECOMPUTE = PCOMPUTE \times tCOMPUTE \quad (20)$$

Under these conditions—where both edge and fog nodes stay fully active with their transmitting and receiving antennas constantly powered—the overall energy consumption can be summarized as follows:

$$ETotalEdge = EON + ESENSE + ECOMPUTE + ETRANSMIT/RECEIVE \quad (21)$$

$$ETotalFog = EON + ECOMPUTE + ETRANSMIT/RECEIVE \quad (22)$$

Operational strategy 2

This approach incorporates two additional operational states: SLEEP/IDLE and MEMORY. In this configuration, data acquired by the sensors is first processed locally and then stored within the memory unit for later handling.

$$ESLEEP = PSLEEP \times tSLEEP \quad (23)$$

$$EMEMORY = PMEMORY \times tMEMORY \quad (24)$$

$$ETotalEdge = EON + ESLEEP + ESENSE + ECOMPUTE + EMEMORY + ETRANSMIT/RECEIVE \quad (25)$$

$$ETotalFog = EON + ESLEEP + ECOMPUTE + EMEMORY + ETRANSMIT/RECEIVE \quad (26)$$

Algorithms Performed in the Communication Architecture

This section describes the algorithms developed for classifying and managing randomly introduced nodes within the communication framework. Two primary scenarios are addressed:

- Classifying a random node into one of three categories: edge, fog, or miscellaneous
- Assigning an edge node to an appropriate fog node to enable structured hierarchical communication

Node categorizing algorithm

Assigning a newly introduced random node to the edge, fog, or miscellaneous category relies on multiple criteria, including current workload demands, available computational resources, processing capabilities, network connectivity quality, and observed latency. The offloading-based node categorization procedure is outlined in [Algorithm 1](#).

Every candidate node is assigned a unique identifier and can report its capabilities and current workload. During the initialization phase, the node conducts a self-assessment to determine key attributes, such as CPU processing power (N_{cpu}), available memory (N_{mem}), and storage capacity (N_{sc}). It then evaluates its offloading needs by analyzing task requirements and estimating proximity to potential offload targets. Finally, the node compares its own resources, performance characteristics, and offloading demands against predefined baseline thresholds to determine its most suitable category.

Node allocation algorithm

Assigning an edge node to a specific fog node depends on several considerations, including the edge node's workload and service requirements, the computational capacity and performance characteristics of nearby fog nodes, and the quality of network connectivity (including latency) between the edge node and candidate fog nodes. The offloading-based node allocation procedure is detailed in [Algorithm 2](#).

Input: *CandidateNode CN- Random Node*

Output: *Categorize the random node into EdgeNode, FogNode, or MiscellaneousNode E_n , F_n , or M_n*

```

1:  Initialization: The discovery process of capabilities and the tasks it intends to offload
2:  Node capabilities Ncap: Processing power  $N_{cpu}$ , Memory  $N_{mem}$ , storage capacity  $N_{sc}$ 
3:  Network Communications Ncom: Bandwidth  $N_{bw}$ , Data rate  $N_{dr}$ , Latency  $N_{lat}$ 
4:  Energy efficiency Nee: Power source  $N_{ps}$ , Battery life  $N_{bat}$ , Energy consumption  $N_{sc}$ 
5:  Location Nloc: Location of the node  $N_{loc}$ 
6:  Task Assessment Nta: Task complexity  $N_{tcomp}$ ,  $N_{com}$ 
7:  Proximity Nproxy: Signal Strength  $N_{ss}$ , Network topology  $N_{nt}$ 
8:  for CN in All_Node: do
9:      WHILE:  $E_{lat} \leq F_{lat}$ 
10:         Compute:  $N_{cap}$ ,  $N_{com}$ ,  $N_{ee}$ ,  $N_{loc}$ ,  $N_{ta}$ , and  $N_{proxy}$ 
11:         IF  $N_{cap} \geq E_{cap} \ \&\& \ N_{com} \geq E_{com} \ \&\& \ N_{ee} \geq E_{ee} \ \&\& \ N_{loc} \geq E_{loc} \ \&\& \ N_{ta} \geq E_{ta} \ \&\& \ N_{proxy} \geq E_{proxy}$ 
12:             Then  $CN \subset E_n$ 
13:         ELSE
14:             IF  $N_{cap} \geq F_{cap} \ \&\& \ N_{com} \geq F_{com} \ \&\& \ N_{ee} \geq F_{ee} \ \&\& \ N_{loc} \geq F_{loc} \ \&\& \ N_{ta} \geq F_{ta} \ \&\& \ N_{proxy} \geq F_{proxy}$ 
15:                 ELSE IF  $CN \subset M_n$ 
16:             end for
17:         return Result  $E_n$  or  $F_n$  or  $M_n$ 

```

Algorithm 1: Node categorizing algorithm based on offloading

	Input: EdgeNode E_i
	Output: FogNode F_j
1:	<i>Initialize the network with 'N' edge nodes $E_1, E_2, \dots, E_N$, and 'M' fog nodes $F_1, F_2, \dots, F_M$, number of levels PL.</i>
2:	Input: CandidateEdgeNode CEN with Task (Toff)
3:	Task Evaluation CEN- Resource requirement and latency sensitivity for the task.
4:	<i>Proximity Calculation:</i> Calculate CENss, CENnt
5:	<i>Task-Fog Node Compatibility Assessment:</i>
6:	for CEN in FogNodes: do
7:	Toff && CENnt && CENnt
8:	<i>Then:</i> BestFogNode = F_j
9:	return F_j
10:	end for

Algorithm 2: Edge Node to Fog allocation algorithm

Numerical Analysis and Discussions

The quantitative evaluation of the communication network incorporating delay and energy offloading techniques is examined in the subsections below.

Delay-efficient offloading

The numerical verification of the delay offloading approach proceeds as follows: Assume the cloud server has a capacity of 1 Gbps. Suppose ten edge nodes are directly linked to this server, with each edge node transmitting at a rate of 100 Mbps/s. Applying Eq. 17, the aggregate load becomes

$$NeDe = 10 \times 100MBps/s = 1000MBps = 1GBps \quad (27)$$

At this point, the cloud server has reached its maximum capacity and cannot support any additional edge nodes. To accommodate further expansion, fog nodes are introduced to perform intermediate processing, reducing the volume of data forwarded to the cloud and thereby preserving server capacity. The same principle holds when fog nodes serve as intermediaries, as described by Eq. 27. Consider a setup with 15 edge nodes and three fog nodes, where each fog node receives data from five edge nodes. After local computation at the fog layer, the processed information is sent to the cloud server, as illustrated in Figure 12.

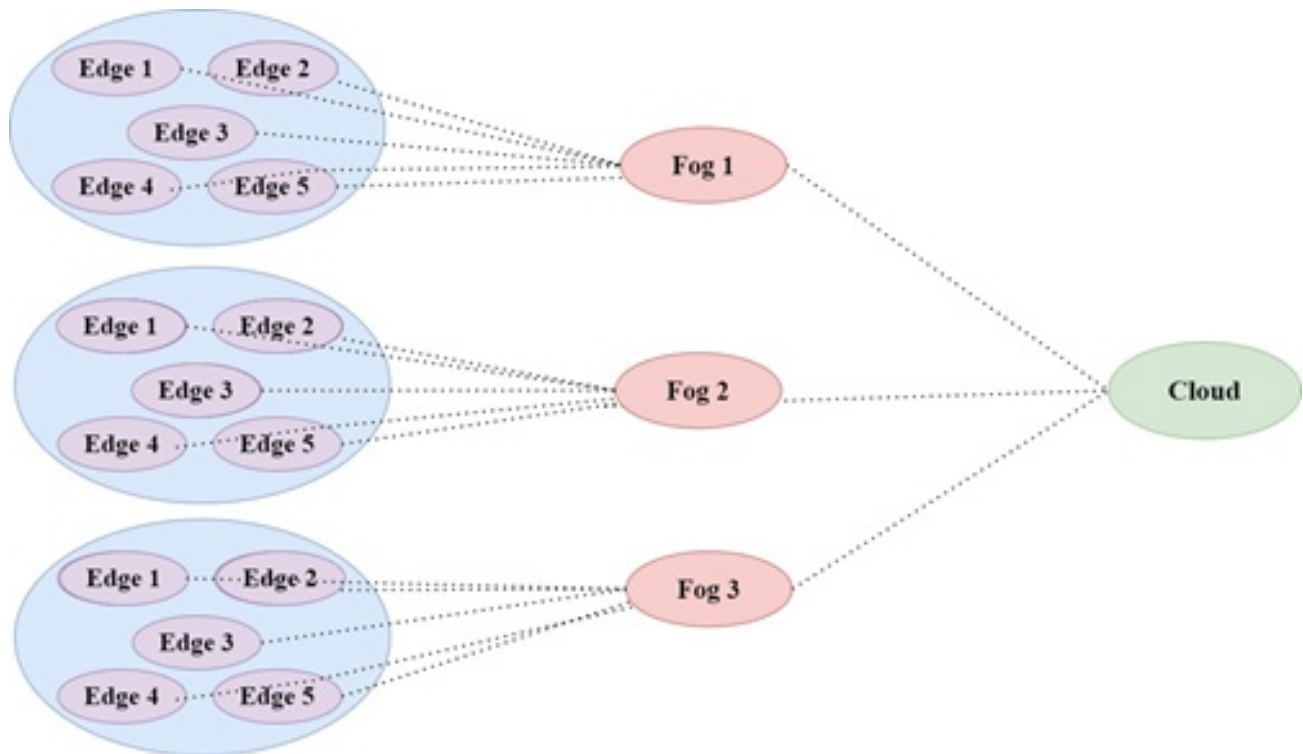


Figure 12. Diagram depicting Delay-efficient Offloading.

Suppose the cloud server capacity is now 1.5 GBps, with three edge nodes (each operating at 500 MBps) connected through the fog layer. The resulting load per fog node is

$$Nf1Df1 = 3 \times 500MBps/s = 1500MBps = 1.5GBps \quad (28)$$

Exceeding this threshold—either in data rate or the number of connected devices—will cause link failure. Therefore, increasing server capacity or optimizing the configuration of edge and fog nodes becomes necessary to avoid communication breakdowns.

Referring to the example configuration in **Figure 12**, assume that network delay doubles as the number of nodes increases. Let the measured delays be 260 μs when edge nodes connect directly to the cloud server, 100 μs when a single fog node is used as an intermediary, and 30 μs when two fog nodes are placed in the path. These delay reductions within the network are graphically represented in **Figure 13**.

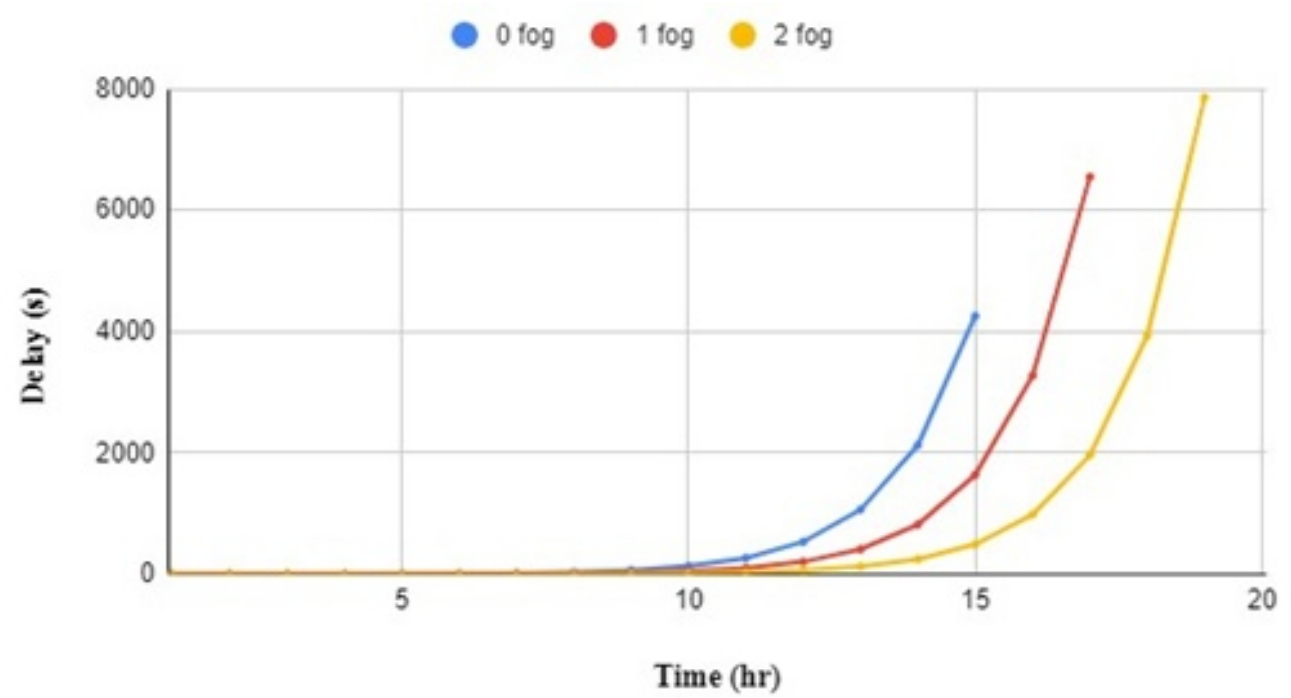


Figure 13. Delay Offloading within the Network is demonstrated using an example scenario.

Energy-efficient offloading

Assume the total available energy budget is 10,000 mWh. Using the power consumption values and daily operating durations for various modes listed in [Table 5](#), the energy usage and expected battery lifetime for different sensor nodes can be computed.

Table 5. Power consumption and operating time for a day in different operational modes.

Operational state	Scenario 1 power consumption (mW)	Edge operating time scenario 1 (h)	Fog operating time scenario 1 (h)	Scenario 2 power consumption (mW)	Edge operating time scenario 2 (h)	Fog operating time scenario 2 (h)
ON	0.5	24	24	0.5	24	24
SENSE	4	18	—	4	12	—
COMPUTE	2	18	12	2	10	8
TRANSMIT	12	18	12	12	4	3
SLEEP	—	—	—	0.5	12	12
MEMORY	—	—	—	1	8	8
RECEIVE	12	18	12	12	4	3

Strategy 1

According to [Eq. 22](#), the total energy consumption of an edge node under Scenario 1 is calculated as

ETotal of Edge Node for Scenario 1 = $0.5 \times 24 + 4 \times 18 + 2 \times 18 + 12 \times 18 = 336$ mWh.

The estimated battery-powered operational lifetime for the edge node in Scenario 1 is therefore $10,000 / 336 = 29.7$ days.

For the fog node in the same scenario:

ETotal of Fog Node for Scenario 1 = $0.5 \times 24 + 2 \times 12 + 12 \times 12 = 180$ mWh.

The corresponding lifetime becomes $10,000 / 180 = 55.5$ days.

Strategy 2

Using [Eq. 26](#), the total energy consumption of an edge node under Scenario 2 is

ETotal of Edge Node for Scenario 2 = $0.5 \times 24 + 4 \times 12 + 2 \times 10 + 12 \times 4 + 0.5 \times 12 + 1 \times 8 + 12 \times 4 = 190$ mWh.

The projected battery lifetime for the edge node in Scenario 2 is $10,000 / 190 = 52.6$ days.

For the fog node:

ETotal of Fog Node for Scenario 2 = $0.5 \times 24 + 2 \times 10 + 12 \times 4 + 0.5 \times 12 + 1 \times 8 + 12 \times 3 = 130$ mWh.

The estimated lifetime is $10,000/130 = 76.9$ days.

Energy reserves for both edge and fog nodes decrease linearly over time as the number of operational days increases. This progressive depletion is visualized for a single edge node in [Figure 14](#) and for a single fog node in [Figure 15](#). These graphs clearly demonstrate that incorporating additional low-power operational states significantly reduces overall power draw, enabling greater energy conservation and an extended device lifespan.

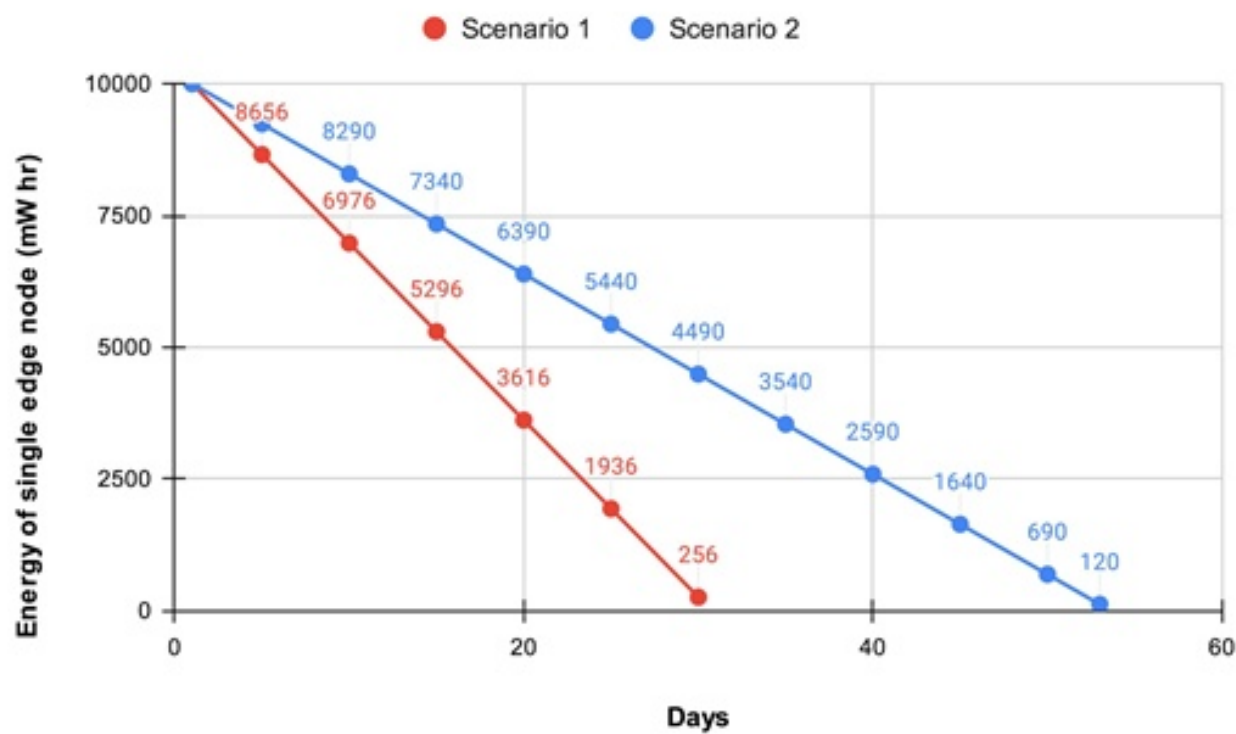


Figure 14. Depleting the energy of a single-edge node.

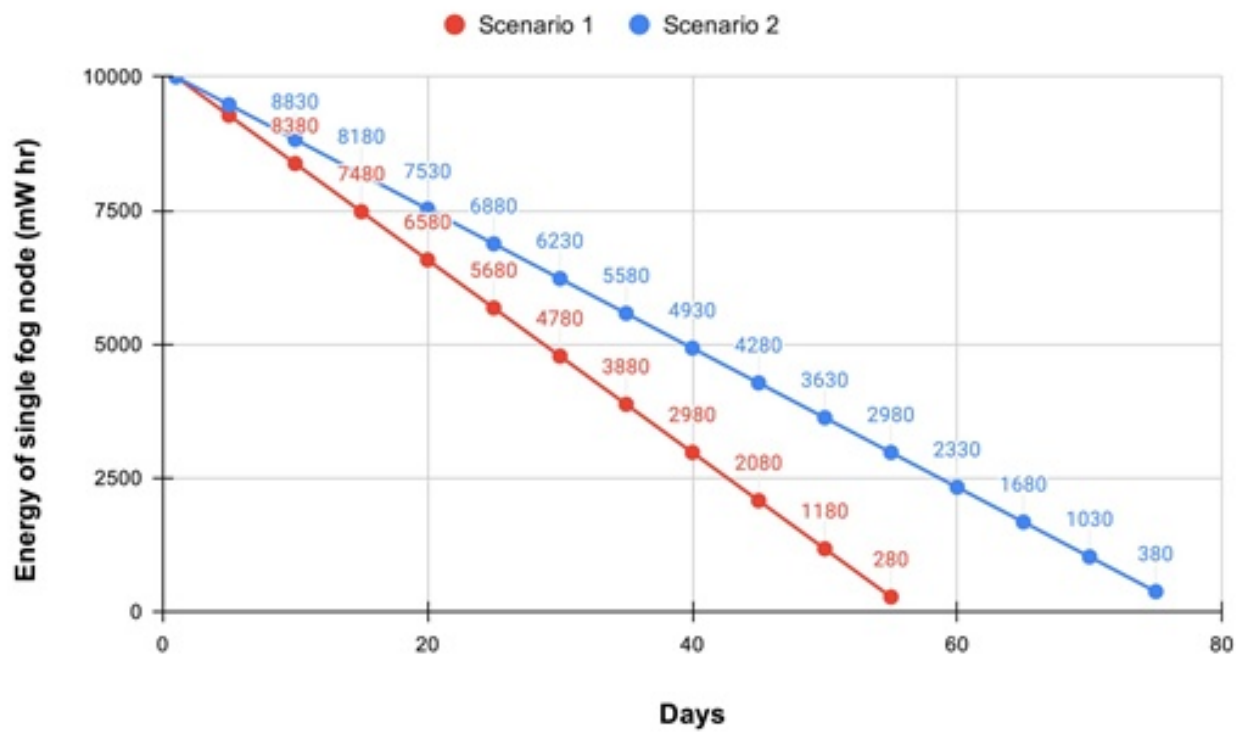


Figure 15. Depleting the energy of a single fog node.

Consider a network configuration in which 15 edge devices connect to the cloud server through three intermediate fog nodes. The aggregate energy consumption across the entire network is expressed by Eq. 29.

$$ET_{Total} = E_{edge} \times \text{Number of edge devices} + E_{fog} \times \text{Number of fog devices} \quad (29)$$

When applying Scenario 1, the aggregate energy consumption across the entire network is 5580 mWh, whereas Scenario 2 yields 3240 mWh. Assuming the network comprises 18 nodes, each equipped with a 10,000 mWh battery, the progressive energy depletion for both scenarios over time is illustrated in Figure 16. The graph clearly reveals the rate of energy drain in each case. Under Scenario 1, the available battery capacity supports only 33 days of operation. In contrast, Scenario 2—with its expanded set of operational states—extends battery life to 55 days. Consequently, implementing the network according to Scenario 2 yields approximately 40% greater energy efficiency compared to Scenario 1.

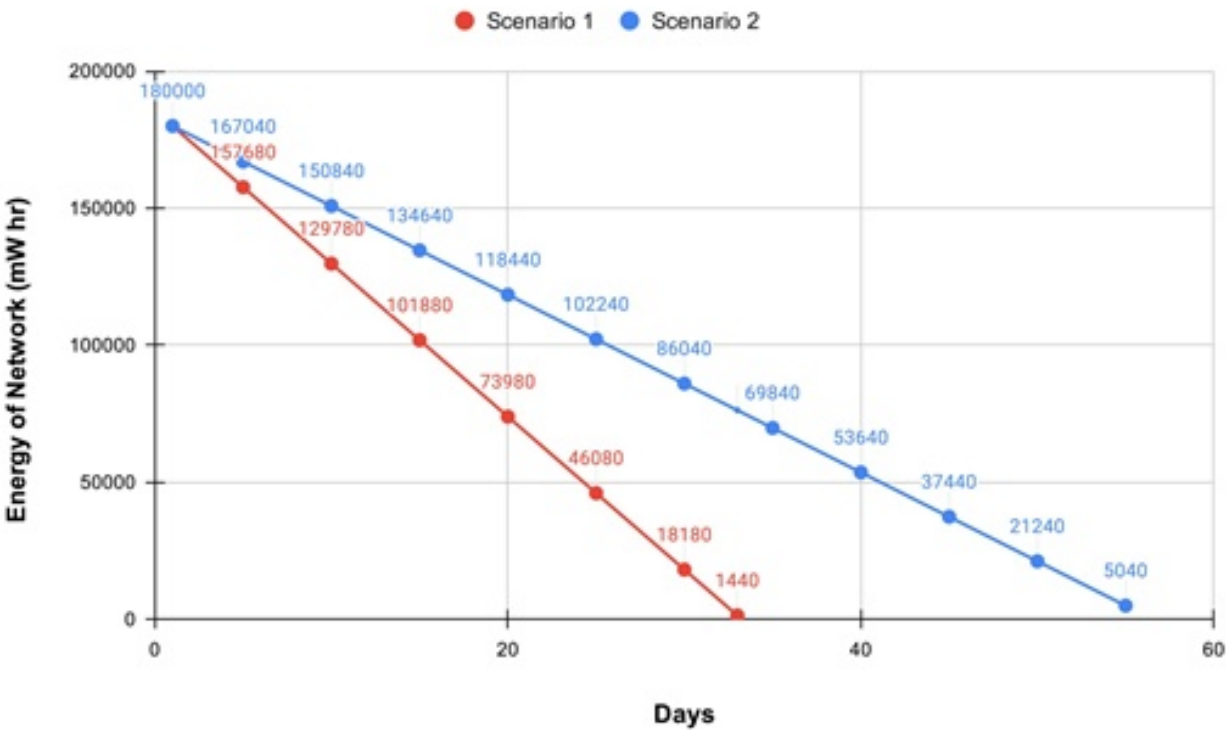


Figure 16. Depleting the energy of the network.

Results and Discussion

This study presents an IoT-enhanced water distribution network (WDN) that optimizes both delay and energy consumption by incorporating a site-specific propagation model. The research begins by pinpointing key challenges in conventional WDNs and proposes the strategic deployment of various sensor types across multiple tiers of the distribution infrastructure. IoT communication components—edge nodes, fog nodes, and cloud servers—are integrated at appropriate levels to enable continuous monitoring of the WDN. A series of practical procedures is outlined to construct a realistic, location-tailored IoT communication framework suitable for any geographically specific IoT-based system. The selected study area is Kochi, India. Land cover in the region was mapped using supervised maximum likelihood classification (MLC), yielding three primary categories: water bodies, vegetation, and built-up zones. Given that Kochi functions as a secondary metropolitan area, analysis of the land cover map alongside applicable propagation models led

to selecting the Hata-Okumura model as the most appropriate fit. The Hata-Okumura path-loss equation tailored for small- and medium-sized urban environments, as given in [Eq. 4](#), was employed. This model accounts for transmitting and receiving antenna heights, antenna gains, transmission frequency, antenna height correction factor, and separation distance between devices. The identical configuration was simulated using the Radio Mobile Freeware tool developed by VE2DBE. Simulation results indicate reliable communication with a 70% coverage probability and a system gain of 161.50 dB, sufficient to deliver the required network reach and connectivity for the intended application.

Delay-efficient offloading for the IoT-supported WDN was formulated with explicit consideration of the wireless propagation characteristics of the study region. The propagation model, path loss, and additional communication parameters govern network delay. Equations were derived to determine the optimal numbers of edge, fog, and cloud nodes, accounting for operating frequency, transmit power, antenna gain, antenna height, channel availability, data rate, signal strength, and bandwidth. Furthermore, the proposed IoT communication topology—illustrated in [Figure 7](#)—incorporates delay-efficient offloading tailored to the Kochi environment. This communication framework addresses multiple WDN challenges through three distinct packet types: data, control, and error packets. Packet exchange sequences among the various IoT components are defined in [Figures 8-10](#). The characteristics and roles of data, control, and error packets are specified in accordance with the network's communication entities and operational requirements.

To ensure reliable and efficient data transmission, both delay and energy must be optimized within the constraints of the propagation model. Energy-efficient offloading is achieved by defining multiple operational states and evaluating different scenarios to identify the most effective configuration for the target area. Energy management begins with low-power signal transmission, incrementally increasing power only as needed until successful reception at the cloud server. However, excessive retransmissions increase end-to-end delay. Therefore, a balanced trade-off must be maintained between delay and path loss on one hand, and delay and energy consumption on the other, to enable optimal performance of the complete IoT-based system.

Taking into account all location-specific factors—including path-loss models, delay-efficient offloading, energy-efficient offloading, node categorization algorithms, and edge-to-fog assignment algorithms—the proposed methods have been developed, analyzed, and numerically validated. Comparative results from the numerical models highlight the superior performance of the recommended operating scenario. These findings enable the optimal deployment of fog nodes in coordination with edge nodes and the cloud server. The resulting high-level, delay- and energy-optimized IoT communication architecture for the WDN is depicted in [Figure 17](#). The methodology and findings presented in this work can be readily adapted to design IoT-enabled communication networks for any other site-specific application.

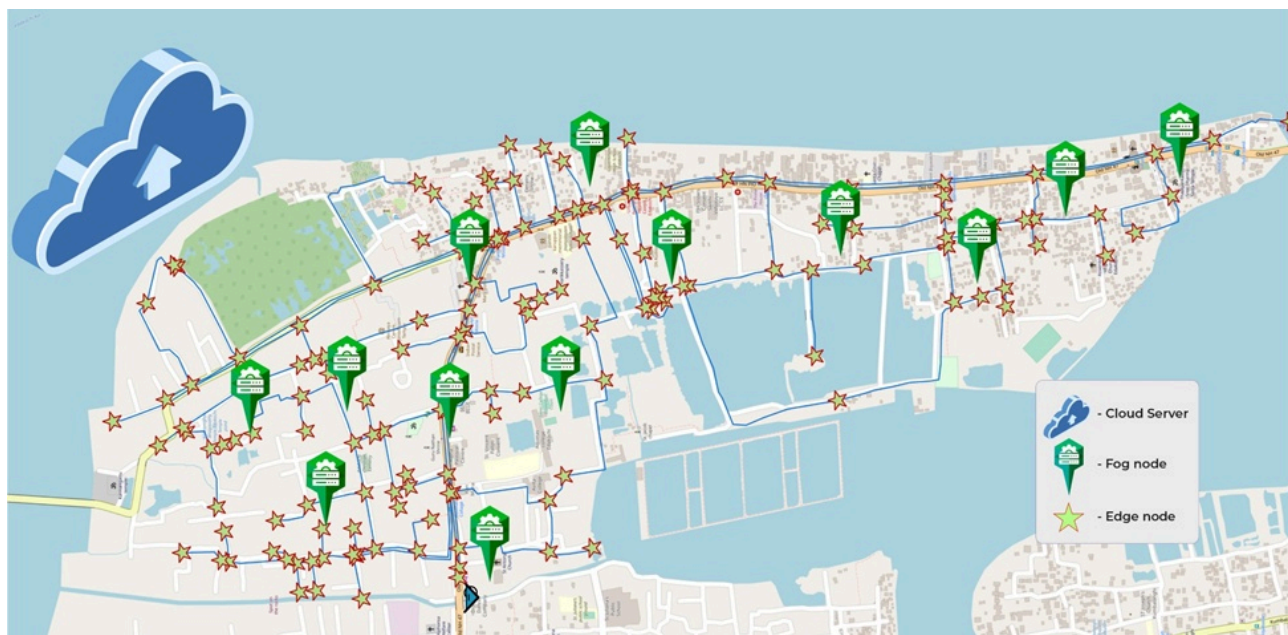


Figure 17. IoT-enabled communication network architecture.

Conclusion

This paper examines the need for an IoT-supported water distribution network and introduces delay- and energy-efficient offloading techniques tailored for smart city environments. Given the various design limitations inherent in developing an IoT-enabled water distribution network (WDN), a comprehensive IoT-based WDN architecture is proposed and developed. As a practical case study, the water distribution network in Kochi, Kerala, India, is selected for detailed investigation. Within this designated study region, an in-depth assessment of land-use patterns, land-cover classification, and prevailing wireless propagation characteristics is conducted. Drawing on these analytical findings, effective offloading strategies for both delay and energy consumption in the WDN are implemented across several operational configurations. The communication architecture incorporates key algorithms—specifically, the node categorization and node allocation algorithms—which are thoroughly described. Detailed numerical evaluations of delay-efficient offloading and energy-efficient offloading under different operational scenarios are also provided. Ultimately, by integrating all relevant parameters, a complete IoT-enabled communication network architecture optimized for the water distribution system is formulated.

Abbreviations

IoT: Internet of Things

ICT: Information and Communications Technology

M2M: Machine-to-Machine

WDS: Water Distribution System

WTP: Water Treatment Plant

PMS: Pressure monitoring sensor

PMS: Pressure Monitoring Sensor

LMS: Flow monitoring sensor

FMS: Flow Monitoring Sensor

QMS: Quality Monitoring Sensors

QGIS: Quantum Geographic Information System

SCP: Semi-Automatic Classification

MLC: Maximum Likelihood Supervised Classification

LULC: Land Use And Land Cover

FSTL: Free-Space Transmission Loss

P_r : Received power

P_t : Transmitted power

G_t : Gain of the transmitting antenna

G_r : Gain of the receiving antenna

λ : wavelength

d : Distance between transmitting and receiving antennas

L : Transmission loss

dkm : Distance on Kilometer

LU : Path loss in Urban areas

h_b : Height of base station antenna (m)

h_m : Height of mobile antenna (m)

f : Frequency of transmission (MHz)

CH : Antenna height correction factor (dB)

η : Spectral efficiency of the wireless link of the edge device

δ : Product of losses in the wireless channel

e_i : Edge node

i' : Fog node

j : Received power at the fog node

P_r

Pn_0 : Noise power spectral density

$P_{ti7 \rightarrow i'}$: Transmission power required from the edge node e_i to fog node f_i' through the link j

j : Received power at the fog node i'

P_r

θ : SIR threshold

D_t : Total delay

D_{prop} : Propagation delay

D_t : Transmission delay

D_{proc} : Processing delay

D_q : Queuing delay

B : Channel bandwidth

$C_{ji7 \rightarrow i'}$: Transmission channel $C_{ji7 \rightarrow i'}$ to the e_i to the f_i'

b_i : Size of the data

TR : Transmission rate

M : Message length

d_θ : The channel access delay

$f_{m'i'} f_{cfi'}$: Fog memory

Clock frequency

$f_{pci'}$: Fog processing capacity

α : The constant delay parameter for the propagation delay

d_{comp} : The total delay in the communication network of a single node

d_{sens} : The sensing delay of a single node

d_{proc} : The processing delay of a single node

d_{trans} : The transmission delay of a single node

d_{chan} : The channel delay

$dband$: The delay due to bandwidth issues

$dprop$: the delay due to propagation effects

N_e : The total number of edge devices

C_{cloud} : The total capacity of the cloud server

N_f : The total number of fog devices

C_c : cloud server capacity

D_e : Data rate of each edge node

$N_{f1}, N_{f2}, \dots, N_{fn}$: The intermediate fog nodes

$D_{f1}, D_{f2}, \dots, D_{fn}$: The data rates of the intermediate nodes

ON : The edge and fog devices are in the ON state, and the transmission and reception are taking place every second

PON : The power consumption state

EON : The energy consumption state

t_{SENSE} : Sensing time

E_{SENSE} : Sensing energy

P_{SENSE} : Sensing power

$E_{COMPUTE}$: Computation energy

$P_{COMPUTE}$: computation power

$t_{COMPUTE}$: Computation time

$E_{TRANSMIT/RECEIVE}$: Transmit/Receive energy

$P_{TRANSMIT/RECEIVE}$: Transmit/Receive power

$t_{TRANSMIT/RECEIVE}$: Transmit/Receive time

$E_{TotalEdge}$: The total energy consumption of the edge

$E_{TotalFog}$: The total energy consumption of the fog

E_{SLEEP} : Sleep energy

P_{SLEEP} : Sleep power

t_{SLEEP} : Sleep time

EMEMORY: Storage energy

PMEMORY: Storage power

tMEMORY: Storage time

Acknowledgements

We are grateful to Sri Mata Amritanandamayi Devi (Amma) for her guidance and for inspiring and motivating us to carry out this work. This research is done under the Project “Fast forward to SDG6: Acceptable and affordable water in secondary Indian cities (4WARD)”, supported by the Department of Science & Technology (DST), Water Technology Initiative-National Consortia on Urban Water System (UWS) with project number: DST/TM/EWO/WTI/2K19/UWS-03(G2). We thank every member of the 4WARD Project Team for their support. We also thank the Kerala Water Authority (KWA) team for their support and guidance.

Conflict of interest

None

Financial support

This research was funded by the Department of Science & Technology (DST), Water Technology Initiative-National Consortia on Urban Water System (UWS) with project number: DST/TM/EWO/WTI/2K19/UWS-03(G2).

Ethics statement

None

Received: 16 Oct 2023

Revised: 22 Nov 2023

Accepted: 31 Dec 2023

Published online: 20 March 2024

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- Loucks DP, van Beek E. Water resources planning and management: An overview. In: Water Resource Systems Planning and Management. Cham, Switzerland: Springer; 2017:1-49.
- Brentan B, Meirelles G, Luvizotto E Jr, Izquierdo J. Hybrid SOM+ k-Means clustering to improve planning, operation and management in water distribution systems. *Environ Model Softw*. 2018;106:77-88.

Rathore MM, Ahmad A, Paul A, Rho S. Urban planning and building smart cities based on the internet of things using big data analytics. *Comput Netw*. 2016;101:63-80.

Cosgrove WJ, Loucks DP. Water management: Current and future challenges and research directions. *Water Resour Res*. 2015;51(7):4823-39.

Chettri L, Bera R. A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. *IEEE Internet Things J*. 2019;7(1):16-32.

Mehmood Y, Ahmad F, Yaqoob I, Adnane A, Imran M, Guizani S. Internet-of-things-based smart cities: Recent advances and challenges. *IEEE Commun Mag*. 2017;55(9):16-24.

Velayudhan NK, Pradeep P, Rao SN, Devidas AR, Ramesh MV. IoT-enabled water distribution systems—A comparative technological review. *IEEE Access*. 2022;10:101042-101070.

Laya A, Wang K, Widaa AA, Alonso-Zarate J, Markendahl J, Alonso L. Device-to-device communications and small cells: Enabling spectrum reuse for dense networks. *IEEE Wirel Commun*. 2014;21(4):98-105.

Sharma SK, Wang X. Live data analytics with collaborative edge and cloud processing in wireless IoT networks. *IEEE Access*. 2017;5:4621-35.

Firouzi F, Farahani B, Marinšek A. The convergence and interplay of edge, fog, and cloud in the AI-driven Internet of Things (IoT). *Inf Syst*. 2022;107:101840.

Venkatesan VK, Izonin I, Periyasamy J, Indirajithu A, Batyuk A, Ramakrishna MT. Incorporation of Energy Efficient Computational Strategies for Clustering and Routing in Heterogeneous Networks of Smart City. *Energies*. 2022;15(20):7524.

Mavromoustakis CX, Mukherjee M, Mastorakis G, Song H, Gorlatova M, Aazam M. Guest Editorial Special Issue on Emerging Trends and Challenges in Fog Computing for IoT. *IEEE Internet Things J*. 2020;7(5):4155-9.

Minoli D, Sohraby K, Occhiogrosso B. IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems. *IEEE Internet Things J*. 2017;4(1):269-83.

Soundararajan R, Stanislaus PM, Ramasamy SG, Dhabliya D, Deshpande V, Sehar S, et al. Multi-channel assessment policies for energy-efficient data transmission in wireless underground sensor networks. *Energies*. 2023;16(5):2285.

Bagchi S, Abdelzaher TF, Govindan R, Shenoy P, Atrey A, Ghosh P, et al. New frontiers in IoT: Networking, systems, reliability, and security challenges. *IEEE Internet Things J*. 2020;7(11):11330-46.

Naseer S, Liu W, Sarkar NI. Energy-efficient massive data dissemination through vehicle mobility in smart cities. *Sensors*. 2019;19(21):4735.

Vinayakumar R, Alazab M, Srinivasan S, Pham QV, Padannayil SK, Simran K. A visualized botnet detection system based deep learning for the internet of things networks of smart cities. *IEEE Trans Ind Appl*. 2020;56(4):4436-56.

Adedeji KB, Ponnle AA, Abu-Mahfouz AM, Kurien AM. Towards digitalization of water supply systems for sustainable smart city development—Water 4.0. *Appl Sci*. 2022;12(18):9174.

Rojek I, Studzinski J. Detection and localization of water leaks in water nets supported by an ICT system with artificial intelligence methods as a way forward for smart cities. *Sustainability*. 2019;11(2):518.

Kumari MS, Kumar N. Channel model for simultaneous backhaul and access for mmWave 5G outdoor street canyon channel. *Wirel Netw*. 2020;26(8):5997-6013.

Anagnostopoulos C, Aladwani T, Alghamdi I, Kolomvatsos K. Data-driven analytics task management reasoning mechanism in edge computing. *Smart Cities*. 2022;5(2):562-82.

Ann JD, Sujith K. Addressing walkability of Kochi corporation area with a focus on inland water transportation. In: IOP Conference Series: Materials Science and Engineering. Kerala, India: IOP Publishing; 2021;1114:012036.

Kuriakose PN, Philip S. City profile: Kochi, city region-Planning measures to make Kochi smart and creative. *Cities*. 2021;118:103307.

Adithya K, Ajith V, Nibi K, Ramesh MV. Design of IoT and ML enabled Framework for Water Quality Monitoring. In: 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC). Coimbatore, India: IEEE; 2022:452-60.

Nibi K, Kumar MN, Devidas AR, Ramesh MV, Thadathil SP. GIS based Urban Water Distribution Network Analysis: A Case Study in Kochi, India. In: 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). Greater Noida, India: IEEE; 2022:956-63.

Richter R, Weingart U, Wever T, Kahny U. Urban land use data for the telecommunications industry. *Photogramm Fernerkund Geoinf*. 2006;2006(4):297.

Ali S, George A. Redressing urban isolation: A multi-city case study in India. *J Urban Manag*. 2022;11(3):338-52.

Armoogum V, Soyjaudah K, Mohamudally N, Fogarty T. Propagation Models and Their Applications in Digital Television Broadcast Network Design and Implementation. London, UK: IntechOpen; 2010.

Friis HT. A note on a simple transmission formula. *Proc IRE*. 1946;34(5):254-6.

Supanakoon P, Aroonpraparat S, Promwong S, Ichi J. Free Space Path Loss of UWB Communications. Tokyo, Japan: Tokyo Institute of Technology; 2011:1-6.

Singh D, Jain A, Yadav RP. Development of circular loop frequency selective surface using 3-D printing technique. *Prog Electromagn Res M*. 2020;90:195-203.

Mourad F. Investigation on LTE for Advanced Real Time Two Way Communication for Railway Operational Excellence: Case of Addis Ababa–Djibouti Line [dissertation]. Addis Ababa, Ethiopia: Addis Ababa University; 2016.

Hata M. Empirical formula for propagation loss in land mobile radio services. *IEEE Trans Veh Technol*. 1980;29(3):317-25.

VE2DBE. VE2DBE-Radio Mobile Web Site. Published 2018. Accessed July 1, 2023. https://www.ve2dbe.com/rmonline_s.asp

Singh Y. Comparison of Okumura, Hata and cost-231 models on the basis of path loss and signal strength. *Int J Comput Appl*. 2012;59(11):37-41.

Cheng X, Guan J, Chao K. A WCDMA propagation model based on COST-231-Walfish-Ikegami pattern for typical residential area. In: 2012 8th International Conference on Wireless Communications, Networking and Mobile Computing. Shanghai, China: IEEE; 2012:1-3.

Milanovic J, Rimac-Drlje S, Bejuk K. Comparison of propagation models accuracy for WiMAX on 3.5 GHz. In: 2007 14th IEEE International Conference on Electronics, Circuits and Systems. Marrakech, Morocco: IEEE; 2007:111-4.

Abbasi M, Mohammadi-Pasand E, Khosravi MR. Intelligent workload allocation in IoT–Fog–cloud architecture towards mobile edge computing. *Comput Commun*. 2021;169:71-80.

Berger U. Microwave Communications. In: Meyers RA, ed. *Encyclopedia of Physical Science and Technology*. 3rd ed. New York, NY: Academic Press; 2003:777-98.

Cabrera GJA, Fitzek FH, Hanisch S, Itting SA, Zhang J, Zimmermann S, et al. Intelligent networks. In: Fitzek FH, Li SC, Speidel S, Strufe T, Simsek M, Reisslein M, eds. *Tactile Internet*. Cambridge, MA: Academic Press; 2021:131-49.