Hindawi
Journal of Computer Networks and Communications
Volume 2019, Article ID 9651915, 33 pages
https://doi.org/10.1155/2019/9651915

Review Article

QoS Strategies for Wireless Multimedia Sensor Networks in the Context of IoT at the MAC Layer, Application Layer, and Cross-Layer Algorithms

Muwonge Ssajjabbi Bernard[1,2], **Tingrui Pei**[1,3] **and Kimbugwe Nasser**[1,2]

[1]*College of Information Engineering, Xiangtan University, Xiangtan, Hunan 411105, China*
[2]*Department of Networks, College of Computing & I.S, Makerere University, Kampala, Uganda*
[3]*Key Laboratory of Hunan Province for Internet of Things & Information Security, Xiangtan University, Xiangtan, China*

Correspondence should be addressed to Tingrui Pei; peitingrui@xtu.edu.cn

Received 24 May 2019; Revised 11 September 2019; Accepted 9 October 2019; Published 29 December 2019

Academic Editor: Zhiyong Xu

Wireless multimedia sensor networks (WMSNs) have got capacity to collect both scalar sensor data and multidimensional sensor data. It is the basis for the Internet of things (IoT). Quality of service (QoS) pointers like energy efficiency, reliability, bit error rate, and latency can be helpful in data collection estimation over a network. In this paper, we review a number of QoS strategies for WMSNs and wireless sensor networks (WSNs) in the IoT context from the perspective of the MAC and application layers as well as the cross-layer paradigm. Considering the MAC layer, since it is responsible for regulating the admittance to the shared medium and transmission reliability and efficiency through error correction in wireless transmissions, and for performance of framing, addressing, and flow control, the MAC protocol design greatly affects energy efficiency. We thus review a number of protocols here including contention-free and contention-based protocols as well as the hybrid of these. This paper also surveys a number of state-of-the-art machine-to-machine, publish/subscribe, and request/response protocols at the application layer. Cross-layer QoS strategies are very vital when it comes to system optimization. Many cross-layer strategies have been reviewed. For these QoS strategies, the challenges and opportunities are reviewed at each of the layers considered. Lastly, the future research directions for QoS strategies are discussed for research and application before concluding this paper.

1. Introduction

Wireless sensor networks (WSNs) have a significantly large number of interconnected sensor nodes which can sense their environmental attributes like pressure, light, temperature, sound, humidity, and location. They further cooperate among themselves over wireless transmission media across which they transmit their data as they monitor their environment [1]. For this to be done more accurately, there is a need for multimedia system support for better information gathering and environmental monitoring. WSNs have thus given birth to a new paradigm shift toward WMSNs. This is inspired by the recent advances in technology that have given rise to portable, cheap multimedia capture, transmission, and storage devices such as digital video cameras, microphones, low-cost smart phones, imaging sensors, memory cards, and hard disks. These technologies are easy to integrate into a node and have made information gathering and monitoring of their environment easier and cheaper. Being low-cost smart devices, they have motivated many scholars to undertake research on WMSNs. These can promptly transmit, store, compare, and combine data from heterogeneous sources.

WMSNs are an enhanced kind of WSNs that can sense and/or transmit both scalar and multimedia data including image, audio, and video streams in real-time or non-real-time transmission. They are networks for wireless embedded devices that can permit the users to retrieve multimedia information from their environment [2]. They have many applications in surveillance and environmental monitoring

systems, traffic monitoring, target tracking, intrusion detection systems, telemedicine for advanced health care, etc. Iqbal et al., for instance, developed an efficient power allocation and personal wireless hub (PWH) placement strategy for maximizing the data rate under the cognitive radio interference constraint that has an efficacy with low complexity to facilitate paramedic staff in next-generation health care facilities using multimedia in smart hospitals [3].

1.1. Classification of WSNs (Wireless Sensor Networks). Different types of WSNs include terrestrial WSNs, underground WSNs, underwater WSNs, multimedia WSNs, and mobile WSNs.

1.1.1. Terrestrial WSNs. These can communicate with base stations efficiently and comprise 100 s to 1000 s of wireless sensor nodes deployed in the unstructured (ad hoc) or structured (preplanned) style. In the ad hoc mode, nodes are randomly distributed in the target region that is dropped from a fixed plane. The structured mode considers optimal placement, grid placement, and 2D and 3D placement models. Energy is conserved by use of low duty cycling, delay minimization, and optimal routing.

1.1.2. Underground WSNs. These are more expensive than terrestrial WSNs in terms of deployment, maintenance, equipment cost, and careful planning. The WSNs comprise many sensor nodes hidden in the ground to monitor underground conditions. To relay information from the sensor nodes to the base station, additional sink nodes are located above the ground. These are highly affected by attenuation and signal loss and are very difficult to charge.

1.1.3. Underwater WSNs. These have multiple sensor nodes and vehicles deployed under water. Autonomous underwater vehicles are used to gather data from the sensor. Long propagation delay and bandwidth and sensor failures are a major challenge here.

1.1.4. Multimedia WSNs (WMSNs). These have been proposed to enable tracking and monitoring of events in the form of multimedia such as imaging, video, and audio. These networks comprise low-cost sensor nodes equipped with microphones and cameras. The nodes are interconnected with each other over a wireless connection for data compression, retrieval, and correlation. The challenges with WMSNs include high energy consumption, data processing and compressing techniques, and high bandwidth requirements for proper and easy content delivery.

1.1.5. Mobile WSNs. They comprise a collection of sensor nodes which can move on their own and can interact with the physical environment. Mobile nodes have the ability to compute, sense, and communicate. The mobile WSNs are more versatile than static sensor networks. The advantages of WMSNs over the static WSNs include better and improved coverage, better energy efficiency, and superior channel capacity.

1.2. The Architecture of a WMSN. The WMSN architecture has the following 3 subdivisions:

(i) Single-tier flat architecture: this consists of homogeneous multimedia nodes which are able to execute any function to the sink via multihop routes.

(ii) Single-tier clustered architecture: this consists of heterogeneous nodes passing sensed information to the cluster head for processing.

(iii) Multitier architecture: this too consists of heterogeneous nodes and does object sensing and target capturing and tracking.

Figure 1 shows a typical example of a WMSN architecture. Figure 1(a) shows a single-tier flat and homogeneous architecture, in which sensors having the same physical abilities are utilized. In Figure 1(b), we have a single-tier clustered and heterogeneous architecture, having nodes with different physical capabilities, e.g., multimedia nodes and scalar nodes. The multitier clustered and heterogeneous architecture (Figure 1(c)) has several layers of nodes having diverse types and processing tasks per layer [2]. WMSNs are widely deployed to offer infrastructural support and sensor accessibility rendering them suitable for Internet of multimedia things (IoMT) transmission [4]. Most applications (apps) in the IoMT, e.g., wearable devices, utilize the WMSN technology.

1.3. The Internet of Things (IoT). In the IoT today, we visualize a situation whereby smart devices connect to a single network—the Internet. It is becoming more popular than it has been because of the multitude of connected devices available recently. A number of IoT- and IoMT-based applications are coming lately while attracting enormous attention.

These applications include smart cities, smart vehicles, homes, factories (Figure 2), and GPS tracking devices [5, 6] and have attracted many inventions in the Americas, Asia, Europe, etc. A number of commercial and military applications come up because of introduction of multimedia objects in transmission of data such as remote patient monitoring in telehealth and telemedicine and traffic management systems enhanced by smart video cameras, among others [4]. This calls for an upgrade in functionality of IoT systems to the IoMT. Figure 2 shows an illustration of the IoT architecture which could be aided by RFID, optical tags, QR codes, Bluetooth low energy, Wi-Fi direct, and LTE-Advanced, among others. Most of the scholars emphasize improvement of efficiency for handling a lot of real-time information (info) but ignore multimedia transmission aspects [7, 8]. The direction of research is shifting from the ordinary IoT to the multimedia-based IoT because of the need to enable smart devices to efficiently observe, sense, and understand their environment through multimedia data [9, 10], hence resulting in the emergence of the newer field of Internet of multimedia things (IoMT).

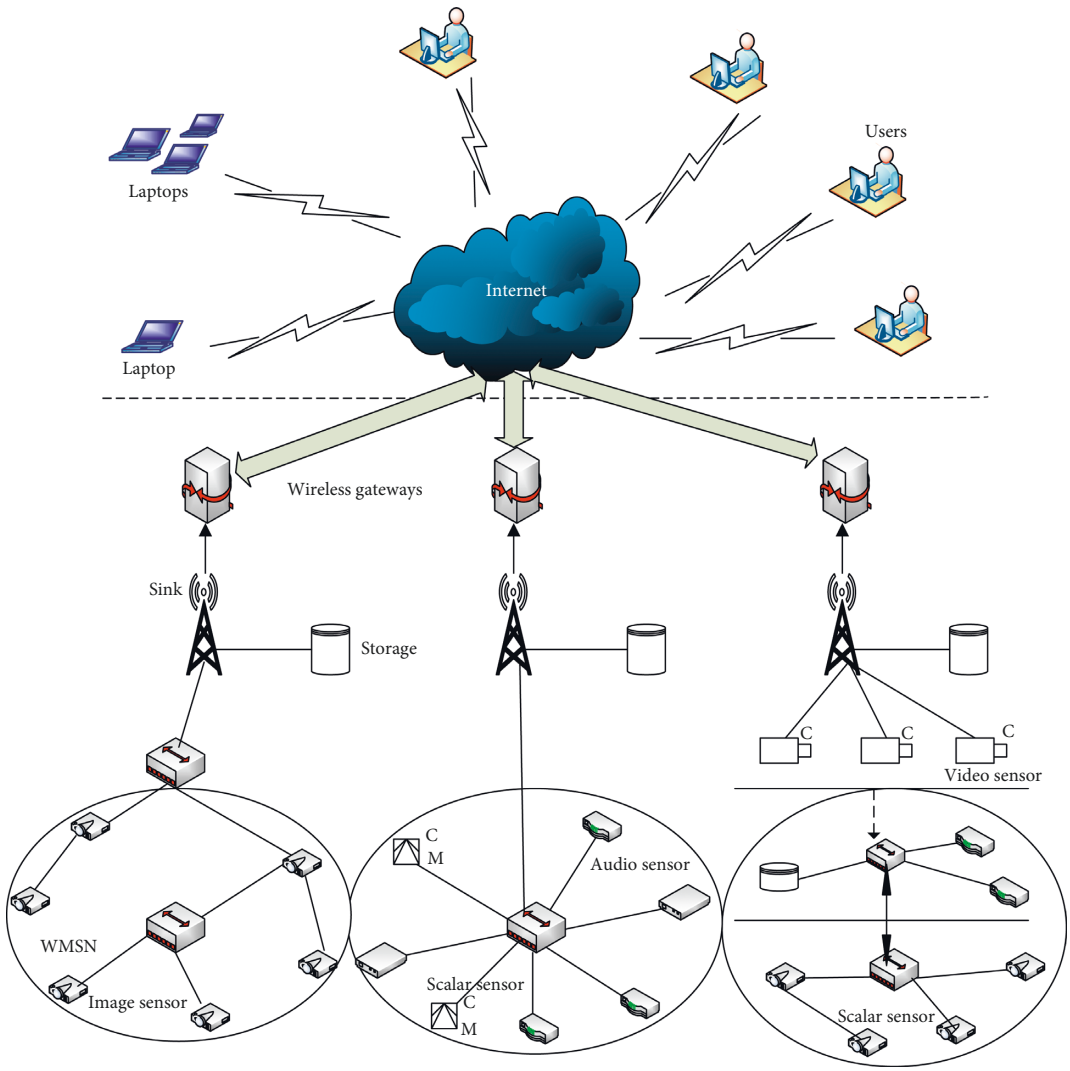

Figure 1: Architecture of the WMSN: (a) single-tier flat and homogeneous; (b) single-tier clustered and heterogeneous; (c) multitier heterogeneous.

IoMT is "*the IoT-based paradigm that enables objects to connect and exchange structured and unstructured data with one another to enable multimedia-based services and applications*" [3]. There is a need for vast processing power, memory, and bandwidth for high QoS when transmitting multimedia data in the IoMT in comparison with scalar data in a traditional IoT. IoT system functionality should therefore be upgraded to the IoMT. We compare the two as discussed in [7]:

(i) The IoT has standardized communication protocols, whereas the IoMT's protocols are nonstandardized.

(ii) In terms of QoS, the IoT requires low bandwidth, whereas the IoMT requires higher bandwidth.

(iii) The IoMT transmits heterogeneous multimedia data, whereas IoT data transmitted have limited heterogeneity.

(iv) IoT sensor nodes consume less energy than IoMT sensor nodes.

(v) IoT devices are deployed in application-dependent RFID tags, but the IoMT is in video and audio sensors.

(vi) In terms of service composition, the IoMT has no available specialized middleware, whereas the IoT has specialized service-oriented, architecture-based, and event-based middleware.

For improved QoS, best effort services, and higher energy efficiency in IoMT networks and applications, enormous multimedia-supported routing is gaining ground in the research arena in the WMSN area in routing protocols, algorithms, and techniques based on network architectures and application requirements [11]. Nevertheless, according to Ahmad et al. [12], the enormous, resource-constrained, heterogeneous environment of the IoT challenges its expansion and deployment. This is because most existing IoT apps comprise overlaid deployments of wireless sensor and actuator networks in which apps cannot interact with each other or share and reuse the few available resources. In addition to that, efficient sensing and propagation of info

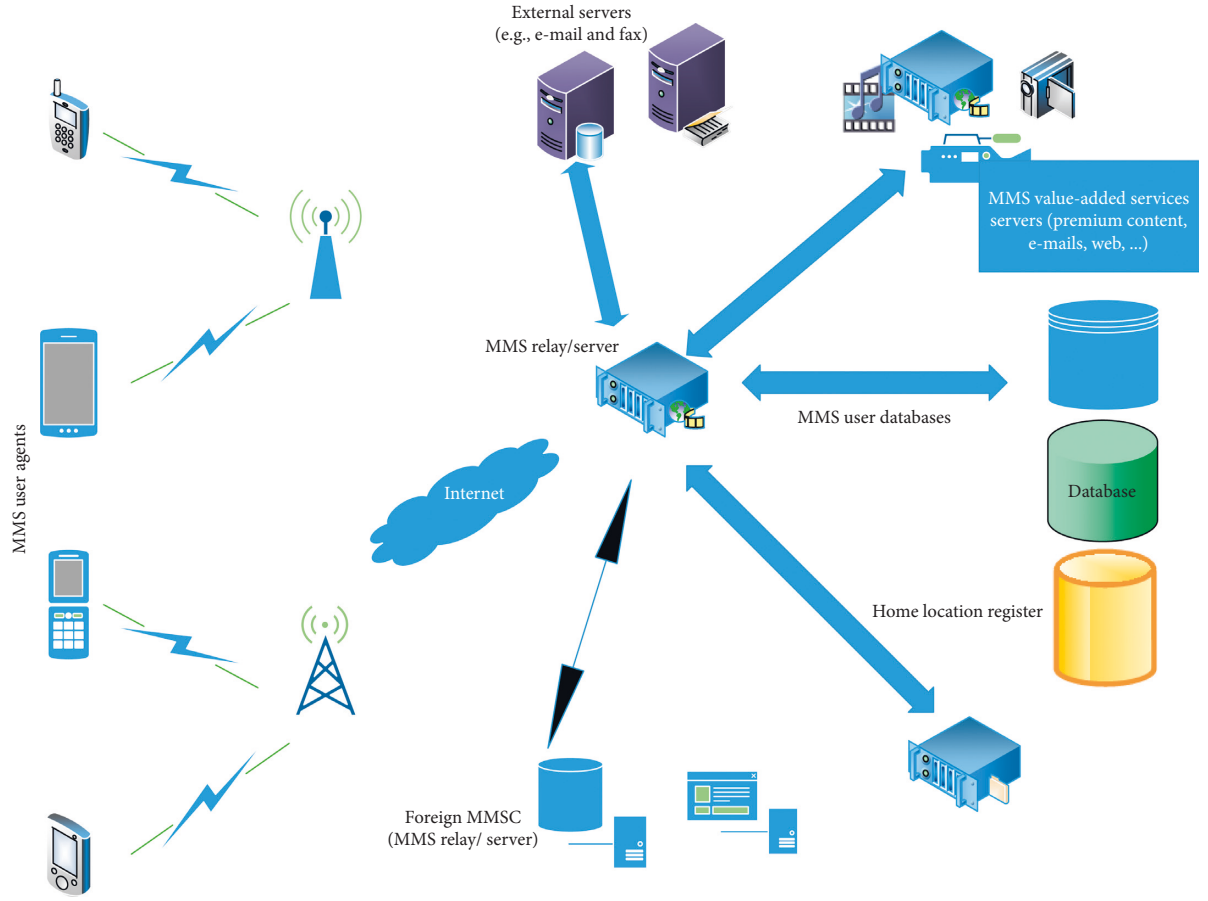

Figure 2: Example of the multimedia service architecture in the context of IoT.

and swift response to changes in the physical world are challenging IoT requirements [12].

1.4. What Has Been Done? In wireless communication systems, routing is one issue that is quite a challenge, yet there are very few surveys available especially regarding WMSN protocols [1] and more so the MAC layer, application layer, and cross-layer protocols although some surveys have been published on WSNs. To the best of our knowledge, no survey combines all these aspects. This makes our survey important to add to the existing literature. The authors in [1] discuss a number of WMSN routing techniques and the properties and shortcomings of these. However, they do not discuss any application layer techniques. Muzakkari et al. [13] survey some recent WSN contention-based, scheduling-based, and hybrid MAC protocols whereby they focus on the underlying principles, advantages, limitations, and their applications. But much as they only focus on MAC protocols, they still leave out a number of them in their survey. Yigitel et al. [14] carry out a survey on QoS-aware MAC protocols for WSNs. They review the QoS challenges and views for WSNs, study the QoS mechanisms, categorize the state-of-the-art QoS-aware MAC protocols, and also talk about the advantages and drawbacks of the same. But being a 2011 release, a number of key protocols developed since then to date are not presented therein. Abbas and Kure [15] review various methods for QoS provisioning at the levels of routing, MAC layer, and cross-layer including the schemes for admission control and scheduling for QoS provisioning as well as the problems and challenges involved. But they leave out the protocols from 2010 to date. Shatnawi [16] and Karagiannis et al. [17] review the application layer protocols for the Internet of things. In their survey, they address a number of application layer protocols which are employed for IoT, for affirming a reliable tie among objects and things. They evaluate the reviewed protocols in terms of architecture, communication model, security, and QoS as well as the weaknesses and strengths per reviewed protocol. However, they only concentrate on application layer protocols and still leave out some critical protocols. In a recent survey, Bernard et al. [18] discuss a number of cross-layer QoS strategies for IoT. But because of space constraint, we leave out some critical protocols, and besides, this paper does not talk about application layer and MAC layer protocols. AlAmri and Abdullah [19] carry out a survey on cross-layer QoS protocols for WMSNs in which they state that the cross-layer architecture is a novel idea that brings together a number of layers for enabling integration and exchange of information in between them with higher efficiency compared to the traditional layered model. They discuss and compare the available cross-layer WMSN protocols that cross the uses of adjacent or nonadjacent layers. But they concentrate on

cross-layer protocols and leave out the MAC layer and application layer protocols, and also some cross-layer protocols are left out. The surveys that have been reviewed are compared in Table 1.

The major contributions that make our survey important are outlined below.

1.5. Importance of the Survey

(i) We discuss and draw a comparison between the IoT and the IoMT by discussing the differences between the two.

(ii) We further discuss the newer paradigm of WMSNs and compare WSNs and WMSNs in our discussion.

(iii) We make a comprehensive survey of the recent state-of-the-art routing protocols focusing on the MAC layer, the application layer, and the cross-layer paradigm.

(iv) We discuss the WMSN protocols as well as the WMSN-enabled protocols that have been developed in the recent studies by several scholars.

(v) In some of the reviewed protocols, we talk about the relative advantages and disadvantages of some of the routing approaches for readers to be in position to comprehend different techniques and so choose the most appropriate technique depending on the user requirements.

(vi) In all these, we cover the challenges and opportunities existing for the discussed categories.

(vii) We consider the possible future research trends before concluding this paper.

1.6. Challenges

(i) Since WSNs and WMSNs are usually distributed and ad hoc networks, their nodes are powered by batteries making energy optimization a challenge that needs to be addressed when designing their routing protocols.

(ii) Considering WSNs, their routing protocols do not put into consideration multimedia applications which need a lot of bandwidth and processing energy and must be transmitted in real time with utmost fidelity. Extra challenges in routing are imposed in WMSNs in deployment of heterogeneous sensors since there could be audio, video, and still pictures in addition to scalar data during transmission. Besides, the heterogeneous data thereby handled all have different QoS requirements, and with many different business needs, different services with varying requirements give rise to a big challenge for the routing design.

(iii) WMSNs have also got nonrechargeable batteries placing a limitation on the energy of the nodes, yet multimedia applications consume a lot of energy. Quite a number of complications and limiting factors exist practically despite the fact that sensors can be self-powering as enabled by energy harvesting. Predictive video encoding using MPEG-4 and other standards is also an energy-consuming process and so causes enormous degradation in routing efficiency complicating the design of WMSN protocols even more.

(iv) The heterogeneous nature of the network is another challenge since there is a difference in the categories of nodes needed for communication so as to enable the facilitation of effective data collection and processing as well as efficient transmission. It is not easy to have a uniform communication protocol platform because of different functionalities unlike the case of the conventional WSNs compared to the WMSNs.

(v) There must be a trade-off between the energy efficiency and the multimedia QoS when deciding on the route taken. Data aggregation and compression routing protocols may be applicable for energy-saving though they can lead to intolerable delays in WMSNs. Furthermore, there might be network congestion due to the many-to-many and mutual interference in wireless routes and the scarce WMSN resources. In case a given node gets so many high-rate streams, it lowers the performance of the network and raises the possibility of node failure due to energy drainage.

This paper mainly discusses QoS strategies for WSNs and WMSNs in the context of IoT from the MAC layer, application layer, and cross-layer perspective. This paper presents a state-of-the-art survey on routing in WSNs and WMSNs. The rest of this paper is organized as follows: In Section 2, we discuss the QoS strategies at the MAC layer. In Section 3, we discuss the QoS strategies at the application layer. Section 4 discusses the cross-layer QoS strategies, and in Section 5, we give some future research directions before concluding this paper in Section 6.

2. QoS Strategies at the MAC Layer

The medium access control (MAC) protocol is meant to regulate admittance to the shared medium as well as transmission reliability and efficiency through error correction in wireless transmissions among others. It is also responsible for performance of framing, addressing, and flow control. The MAC protocol design affects energy efficiency too. This is majorly categorized into contention-free protocols, contention-based protocols, and a hybrid of these.

2.1. MAC Protocols for WSNs. Hybrid MAC protocols: these combine contention-free and contention-based MAC protocols on whose advantages they capitalize. Some of these are discussed below.

2.1.1. S-MAC (Sensor MAC). This is a MAC protocol designed for WSNs. The main components of S-MAC include (i) periodic listen and sleep, (ii) collision and overhearing avoidance, and (iii) message passing. Energy

Table 1: Comparison of this survey with existing surveys.

Author(s)	Year	Characteristics
Bhandary et al. [1]	2016	Routing strategies in WMSNs, properties, and limitations
Muzakkari et al. [13]	2018	WSN contention-based, scheduling-based, and hybrid MAC protocols
Ahmad et al. [12]	2011	QoS-aware MAC protocols for WSNs
Muzakkari et al. [13]	2010	QoS provisioning at the levels of routing, MAC layer, and cross-layer
Shatnawi [14]	2016	IoT application layer protocols
Karagiannis et al. [15]	2015	Application layer protocols for the Internet of things
Bernard et al. [16]	2019	Cross-layer QoS strategies for the IoT in WMSNs
AlAmri and Abdullah [17]	2017	Cross-layer QoS protocols for WMSNs
Guo et al. [20]	2012	Cross-layer and multipath-based video transmission for WMSNs
Ksentini et al. [21]	2006	Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture for WMSNs

consumption and self-configuration are controlled using these three techniques. Nodes periodically go into the sleep mode to control the energy consumed in listening to idle channels, and virtual clusters are formed by neighbor nodes for sleep schedule autosynchronization. The protocol employs in-channel signaling when the radio is in the sleep mode and message passing is applied to decrease contention latency for store-and-forward WSN applications during data transmission [22]. For adjacent nodes to solve their clock drift, S-MAC requires periodic synchronization. To reduce synchronization errors, exchanged timestamps are relative not absolute.

Advantages:

(i) S-MAC has a longer listening period than the clock error.
(ii) It has more capacity to conserve energy than IEEE 802.11.
(iii) It can do trade-off between energy and latency depending on traffic conditions [22].

2.1.2. B-MAC (Berkeley MAC). This is a carrier-sense MAC protocol designed for WSNs featuring a simplistic, scalable implementation that does network variations and is effective in channel valuation [23]. The protocol has an interface with some flexibility leading to very low-power usage and controls collisions through collision avoidance mechanisms leading to effective use of the frequency band. It uses a sampling scheme that is adjustable to minimize duty cycling and idle listening so as to reduce power wastage. Nodes are scheduled with their duty cycle each within which they send the preamble to the channel in case of availability of data for transmission. In different duty-cycle schedules, channel usage is checked by the destination nodes; that is, if the preamble found is long, it will remain powered on until the destination address is obtained, and in case it is the target, it will either await sender's data or go to sleep. The protocol has lower latency than synchronous MAC protocols but has a problem of overhearing where the neighbor node gets lengthy preambles though not the intended receiver, hence wasting a lot of power in doing so [24]. It outperforms a number of protocols via reconfiguration, feedback, and bidirectional interfaces for upper layer services. It can run at low duty cycles, and its apps cannot experience synchronization overhead [23].

Advantages:

(i) The protocol features a simplistic, scalable implementation that does network variations and is effective in channel valuation.
(ii) It has an interface with flexibility leading to very low-power usage.
(iii) It controls collisions through collision avoidance mechanisms leading to effective use of the frequency band.
(iv) It uses an adjustable sampling scheme to minimize duty cycling and idle listening, reducing power wastage.

Disadvantages:

(i) It has a problem of overhearing where the neighbor node gets lengthy preambles though not the intended receiver, hence wasting a lot of power in doing so.

2.1.3. T-MAC. This is a contention-based MAC protocol for WSNs that dynamically with little difficulty via fine-grained timeouts adapts a listen/sleep duty cycle. It uses this duty cycle by ending its active part so as to vary load in time and position, thereby decreasing the energy wastage on idle listening [25]. When starting an active period, it uses a short listening window, thereby improving the use of energy by S-MAC. Through adaptive duty cycling, T-MAC saves power but at a cost of less throughput and enhanced delay. Both S-MAC and T-MAC under similar conditions perform in the same way though in variable workloads S-MAC uses 5 times more power than T-MAC. But T-MAC is complex and not scalable. If its active window is decreased, its snooping capability on nearby traffic and adaptation to variable conditions of the network is reduced [23]. All nodes in this protocol are configured to wake up periodically, listen to adjacent nodes, and go back into the sleep mode till the next frame as more messages come into the queue. Collision avoidance and reliable communication are ensured through Request-to-Send (RTS), Clear-to-Send (CTS), and Acknowledgment (ACK) messages among nodes [25].

T-MAC design: the major forms of energy consumption in T-MAC are through (i) idle listening, (ii) collisions, (iii) protocol overhead, and (iv) overhearing. In contrast, in the active period, the node remains listening and sending data. If no activation event occurs for a certain time t, then the active period ends. The activation event might be data reception on a radio, communication sensing on a radio, and end of transmission for the data packet of the node or ACK, knowing that the neighbor has finished transmitting.

Clustering and synchronization: on waking up, a node begins by listening to the channel. If after a certain time there is nothing heard, it chooses a schedule for transmitting Synchronization (SYNC) packets appended with the time of the next frame startup. If during this time it hears another node's SYNC packet, it follows its schedule subsequently transmitting its own SYNC. Following both schedules enables it to have an activation incident at the beginning of both frames.

RTS and t choice: if a node does not get a CTS message after sending an RTS, then either (i) there is a collision preventing the destination node from hearing the RTS, (ii) RTS or CTS is overheard and destination node barred from responding, or (iii) the destination node is in the sleep mode. So, if after time, t, there is no answer, the source node switches to the sleep mode. It should only do so if none of its neighbors is still communicating. A new interval, t, can be triggered by a neighbor's RTS or CTS reception. A node may be out of range to hear the initiating RTS. The length t of the interval should be long enough to get the CTS packet's beginning [25], and t is thus given by the comparison in the following equation:

$$t > C + R + T, \tag{1}$$

where C = contention interval length, R = RTS packet length, and T = turn-around time.

Advantages:

(i) It saves power through adaptive duty cycling.

Disadvantages:

(i) It is unscalable and more complex.
(ii) It offers less throughput and enhanced delay.

2.1.4. CU-MAC. This protocol supports IoT standards requiring request-response transmissions. It employs a multichannel mechanism for constant bidirectional transmission of packets at low duty cycle to resolve hidden-terminal issues. According to Danmanee et al. [24], CU-MAC is an SI-MAC protocol that is aimed at improving bandwidth of the channel by initiating connections and transmitting data through different channels using the multichannel approach.

Protocol design: the protocol has the following three properties that make it perform more efficiently with WSNs in the IoT:

(i) It enables multichannel transmission using 9 useable channels in IEEE 802.15.4.
(ii) It can enable linked data transmission using one connection which reduces additional overhead and delay.
(iii) It can support continuous transmission of packets. This is enabled depending on how the buffer overflow protection policy is set in the buffer stack. This stack has 2 parts: (a) bidirectional buffer slots for enabling sensor nodes to support linked data transmission and (b) normal buffer slots for normal data queuing. Buffer stack policies are set as follows:
(a) A packet is only dropped in the buffer when it expires.
(b) Unless the packet is from a target node to which a receiver is sending the data, the receiver denies packet reception when buffer slots fill up.
(c) The bidirectional buffer slot is used for bidirectional buffer transfer.

CU-MAC can reduce power wastage by choosing to employ only a single channel for transmitting both control packets and data in case there is only one sender in the same region of coverage. It can further enable multiple transmissions through another usable channel if there are other sending nodes in this area [24]. The protocol has two phases: (i) advertising phase, in which a source node announces to neighbors before transmission, and (ii) data transfer phase which commences when the source node receives a Ready-to-Receive Information packet from the destination node.

Advantages:

(i) It has a higher packet delivery ratio compared to other MAC protocols.
(ii) It performs well in high traffic situations since its duty-cycle is lower.
(iii) It reduces power wastage by employing only a single channel for transmitting both control packets and data.
(iv) It can enable multiple transmissions through another usable channel if there are other sending nodes.

2.1.5. R-MAC. This is a reservation-based MAC protocol designed majorly with an aim to be fair and efficient in terms of energy use. To eliminate collision of data packets, R-MAC [26] schedules control and data packet transmissions at the source and destination nodes rather than using RTS/CTS message exchanges. Channel utilization in R-MAC is enhanced by an ARQ method called burst-based acknowledgment which together with other scheduling algorithms addresses the exposed terminal problem besides enhancing the network throughput.

R-MAC design: all nodes in this protocol intermittently function in listen and sleep states so as to avoid wasting energy in overhearing and idle modes. The time taken to listen/sleep is the same for all nodes each of which

chooses a schedule for itself rendering central scheduling and synchronization irrelevant. So, in the absence of traffic in the locality, the node simply does periodic listening and sleeping, and to eliminate collisions, transmissions are distributively synchronized using a reservation-based model in case there is a sender to transmit data. R-MAC differs from S-MAC in that it transmits data to a number of nodes using the sleep mode [24]. There are 3 stages in this protocol: (i) latency detection, (ii) period announcement, and (iii) periodic operation, of which (i) and (ii) are for node synchronization and (iii) is for listen/sleep procedures.

(i) Latency detection phase: here, all nodes broadcast a Neighbor Discovery control packet (ND) at an arbitrarily chosen time since they are powered on. When the ND is received from the neighbor node, the receiver notes its arrival time and chooses a random time to send an acknowledgment (ACK-ND)—the same as the received ND in size—in which the duration from ND arrival to ACK-ND transmission I_2 is specified and the time from ND packet transmission to ACK-ND arrival I_1 is computed.

(a) Propagation latency between the 2 nodes $L = (I_1 - I_2)/2$.

(b) Thus, propagation latency can be defined as "*the interval from the time the first node sends the first bit of a packet to the time the second node receives the last bit of the packet*" [24].

(ii) Period announcement phase: here, nodes choose their own random listen/sleep time and start time and broadcast these, and when packets are received, the node turns the received schedule into its own. All nodes will record their neighbors' schedules in relation to theirs. These two phases make a number of rounds to ensure all nodes are informed about their neighboring nodes.

(iii) Periodic operation phase: here, there is data transmission [26], and nodes intermittently go into the sleep mode and wake up. One period is a listen/sleep cycle, and nodes possess similar periods. Two parts make a period T_0: listen window T_L and sleep window T_S, as shown in the following equation:

$$T_0 = T_L + T_S. \tag{2}$$

In this protocol, the nodes make use of control packets for communication. These include REV (reservation packet), ACKREV (acknowledgment packet for REV), and DATA/ACK-DATA (acknowledgment packet for data) message exchanges all of which are similar in size.

Advantages:

(i) It is fair and efficient in terms of energy use.
(ii) It eliminates collision of data packets.

2.1.6. AR-MAC (Adaptive-Reliable Media Access Control). This is a TDMA-based MAC protocol for wireless body area networks (WBANs) designed to reduce energy consumption as proposed by Rahim et al. [27]. The protocol allocates a guaranteed time slot (GTS) to all nodes for transmission depending on their needs. It utilizes periodic sleep and wakeup to minimize overhearing and idle listening based on the needs of the node. It has a central node (CN) having big batteries with higher computational power and single or double transceivers. For double transceivers, the sum of the time frame, T_{Frame}, is assigned for node transmission. But in AR-MAC, a single transceiver CN is considered with T_{Frame} having 3 subdivisions: contention-free period (CFP) for sensor connection, contention access period (CAP) for emergencies, and time T_{MS} to communicate with the monitoring station (MS).

Channel selection: the CN first scans for redundant channels from which it chooses one for transmission and broadcasts its address and other information to the nodes. The destination nodes search for channels by scanning the radio frequency (RF) channels. If free, it switches to the next, and if busy, it will wait for time T_{CP} listening to packets and again switch to the subsequent channel if it does not get the channel packets. When the node finally succeeds in getting a channel packet, transmission commences and an ACK packet is sent to the CN.

Time slot assignment: a Time Slot Request (TSR) packet is sent to the CN by the sensor node after it has selected the RF channel. The TSR packet has data rate and time slot info for the node. Static-size time slots with static guard-band time, T_{GB}, are proposed in [28]. This protocol employs the adaptive algorithm of time slot (TS) and GB time, and the CN allocates TS while sending Time Slot Request Reply (TSRR) based on the node traffic pattern. The size of the time slots varies depending on the node requirements. Based on the transmission model, the data packet transmission, ACK packet reception, and some amount of delay can be tolerated by allocated TS. To prevent interference between adjacent time slots resulting from the node and CN clock drift, T_{GB} is inserted between them, and its value is given as follows:

$$\begin{aligned} T_{\text{GB}_1} &= \frac{Fx\text{TS}_1}{100}, \\ T_{\text{GB}_n} &= \frac{Fx\text{TS}_n}{100}, \\ T^{\text{GB}}_{n,n+1} &= \frac{F}{100} X \frac{1}{2} \left[\text{TS}_n + \text{TS}_{n+1}\right], \end{aligned} \tag{3}$$

where F = GB factor, depending on the mean drift value. Nodes go to sleep after slots have been successfully assigned and wake up to transmit in assigned slots, thus saving energy wasted in idle listening, since even allocated slots are collision free.

Synchronization: for periodic synchronization, TDMA needs additional energy [29]. It needs a lot of energy to synchronize nodes after several cycles. To reduce energy wastage and control collisions, AR-MAC employs

a new synchronization scheme. When a data packet arrives, the current and expected packet arrival times with allowable delay (D) are compared by the CN, and a drift value (DV) is computed based on the difference, ΔT. The DV is sent to the node in the ACK packet for changing the slot in subsequent transmissions though the value is based on allowable delay and F. In case $\Delta T > D$, the CN will send the DV in the SYNC-ACK packet for subsequent SYNC to nodes, or else a mere ACK packet is sent by the CN for the received packet. For any forthcoming data transmissions, the node will change its wake-up time plan based on the DV. With such a sync approach, a sleeping node will not lose sync for all cycles, N. Acceptable delay (D) is given as

$$D = \min\left(\mathrm{TS}_1, \ldots, \mathrm{TS}_n\right) X \frac{F}{100}, \tag{4}$$

where F is the guard-band factor.

$$\Delta T = \text{expected arrival time} - \text{current arrival time},$$

$$\mathrm{DV} = \begin{cases} 0, & \text{if } |\Delta T| < D, \\ \Delta T, & \text{if } |\Delta T| > D. \end{cases} \tag{5}$$

Frame format: the AR-MAC protocol employs two forms of packets, namely, (i) data packets in which a node transmits in its assigned time slot and employs CAP for emergency data and (ii) control packets that include channel packet, Time Slot Request (TSR) packet, Time Slot Request Reply (TSRR) packet, Synchronization-Acknowledgment (SYNC-ACK) packet, Data Request (DR) packet, and Acknowledgment (ACK) packet [27].

Energy consumption: if N is the number of cycles, energy consumption for these is measured as follows:

$$\begin{aligned} E_{\mathrm{Total}} &= \sum_{k=1}^{N} E_{\mathrm{Sleep}_k} + \sum_{k=1}^{N} E_{\mathrm{Active}_k}, \\ E_{\mathrm{Sleep}} &= T_{\mathrm{Sleep}}\, x\, I_{\mathrm{Sleep}}\, x\, V, \\ T_{\mathrm{Sleep}} &= T_{\mathrm{frame}} - T_{\mathrm{active}}, \end{aligned} \tag{6}$$

where I_{Sleep} = current from the voltage source V when the node is asleep. If E_{sw} is the switching energy, E_{trans} the transmission energy, E_{rec} the receiving energy, and $E_{T\mathrm{out}}$ the time-out energy, then

$$E_{\mathrm{Active}} = 2E_{\mathrm{sw}} + E_{\mathrm{trans}} + E_{\mathrm{rec}} + E_{T\mathrm{out}}. \tag{7}$$

Advantages:

(i) It reduces energy consumption.
(ii) It minimizes overhearing and idle listening based on the needs of the node via periodic sleep and wakeup.

2.1.7. ER-MAC. It is a combination of TDMA and CSMA methods making it flexible and adaptable to topological and traffic changes and hence applicable in rapid response situations in WSNs. Collision-free slots can be planned using TDMA [30]: ER-MAC is designed in such a way that the following are met:

(a) It has a higher ratio of packet delivery and low delay under lower energy consumption levels outperforming Z-MAC.
(b) It keeps two priority queues in order to distinguish high from least priority packets.
(c) It permits contention in TDMA slots so as to handle enormous traffic volumes.
(d) It has a harmonized slot shape, and the nodes are able to locally change their plans enabling them connect to or exit the network with ease.

Protocol design: it discovers the topology in such a way that it uses the simple flooding mechanism to build the tree as initiated by the base station (BS). The topology discovery aims at neighbor discovery and change tracking in addition to setting up a routing tree. A TOPOLOGY_DISCOVERY message comprising hop_count, new_parent_id, and old_parent_id is generated by the BS. A node broadcasts it to discover if it has any potential children and a response to the parent. In case of the desire to change the parent, the former parent is notified. For all nodes, the hop count to the BS, parent ID, children, and one-hop neighbor are registered with the BS at this stage.

TDMA slot assignment: when a BS sends a SYNC message, it switches to the TDMA mode which when received by the child uses the broadcast slot for its children's synchronization. Neighboring nodes cannot share a slot since slots are assigned and schedules are exchanged accordingly. Slot assignment begins with a leaf node in a bottom-up mode such that the flow of messages to the BS may be smooth. Before getting the child-node schedules, a nonleaf node will not transmit its data via a unicast slot, descendant's data via many unicast slots, and child synchronization via a broadcast slot. Child nodes send SCHEDULE_NOTIFICATION messages to the BS to end this phase.

Local time synchronization is done via parent-child broadcast sync since they both need to operate the same clock so that when one is transmitting, the other is ready to receive. Packet prioritization happens in such a way that a high priority queue is first emptied before a lower priority queue is allowed to transmit. In case a queue is filled up, the shortest slack packet is dropped and then fairness over source is considered as queue modification is done to enable the BS balance info from different nodes. Energy conservation is by sender turning off the radio in absence of any info to transmit, and the receiver will switch to the sleep mode in case no packets are received after a certain timeout [30].

Advantages:

(i) It is flexible and adaptable to topological and traffic changes.

(ii) It is applicable in rapid response situations in WSNs.
(iii) It has a higher ratio of packet delivery and low delay under lower energy consumption levels.

2.1.8. RI-MAC (Receiver-Initiated MAC). This is an asynchronous duty-cycle protocol that tries to reduce time occupied by the communicating nodes on the medium to agree on time for data exchange. The sending node waits for the receiving node to signify the beginning of the communication session through a short beacon frame that it sends. The channel is only occupied by the beacon and data transmission, reducing channel occupancy and thus enabling data exchange for other nodes too. This implies an improvement in channel utilization that subsequently leads to increment in throughput, packet delivery ratio, and power efficiency over a wide-ranging traffic load [31]. Being a receiver-initiated model, it ensures substantial reduction in overhearing and lowers the probability of colliding and the cost of recovery compared to B-MAC [23] or X-MAC [32]. RI-MAC is further capable of using a beacon with ACK and Ready-to-Receive packets though it has a problem if a sending node with Ready packets needs to have the radio on till a ready-to-receive node is awake [13] which then sends it a beacon and transmission begins immediately and later is acknowledged by another beacon. Nodes intermittently wake up to listen to any possible frames meant for them depending on their schedules. A node announces its readiness to receive a frame through a broadcast beacon on waking up when it switches on the radio. When a data frame has been received, the receiver will spend some more time, known as "dwell time," in the active state so as to receive queued packets, which is the time in RI-MAC that depends on the number of contending senders. If there are several contending senders that might not have been predicted, there is a challenge in trying to reduce the receiver's active time to optimize power efficiency and reduce the cost of collision detection and lost data recovery even in the case of hidden senders. For this protocol, this can be addressed by a beacon frame from the receiving node for coordinating data frame transmissions of contending sending nodes. The protocol also beats B-MAC and X-MAC in minimizing the cost of collision detection and data recovery through coordination of the receivers' data frames. This is because the receiving node is aware of the maximum delay prior to the arrival of the frame. The ability to broadcast is supported by RI-MAC when it sends data as unicast to each sender's neighboring node or continuously transmits the frame in a back-to-back mode for as long as the sleep interval [31].

Advantages:

(i) It improves channel utilization leading to increment in throughput, packet delivery ratio, and power efficiency over a wide-ranging traffic load.
(ii) It ensures substantial reduction in overhearing and lowers the probability of colliding and the cost of recovery.

2.1.9. SRI-MAC (Synchronous Receiver-Initiated MAC). This is a synchronous duty-cycle protocol that implements the method of receiver-initiated data transmission with an aim of enhancing network lifetime through energy wastage avoidance via collision, overhearing, and idle listening. To do this and also minimize duty cycle, it uses a sequence of adaptive beacons and RTS/CTS packets. In this protocol, sending nodes do not go to sleep till they have got CTS packets from receiving nodes to begin transmitting except for nonsending nodes that return into the sleep mode right away as if the channel was idle [33]. Since sending nodes have separate schedules, it reduces the contention probability leading to more energy conservation. A time frame is divided into the information period, the allocation period, and the communication period which are discussed as follows:

Information period: at frame initiation, a beacon is sent by the receiving node declaring its return from sleep and readiness to accept data packets in case the channel is not busy. The beacon is a small packet alerting other listening nodes of the transmitter's readiness to get a request for channel. It consists of the receiver's ID and the duration allocation period (DAP) that relies on number of the receiving node's neighbors [13].

Allocation period: here, the receiving node stays awake for some little time after broadcasting its beacon to confirm if there is any node having an RTS packet to transmit. When the beacon has been received, an RTS packet (with 3 fields—sender ID, receiver ID, and data size) is sent by the sender prompting the receiver to send a CTS for SYNC as there is neighbor information exchange.

Communication period: SRI-MAC is capable of organizing and scheduling transmissions by means of sensor node IDs. The receiver assigns order and transmitting time for every sender such that they switch off their radios till their time to transmit is due. However, the receiving node remains in the wake-up mode to get all sending node's data. SRI-MAC can enable sensors have lengthy sleeping times during transmission such that energy can be saved as long as it is neither sending nor receiving data. The protocol's communication period is premised on the TDMA technique which conserves energy since the radio's duty cycle is lowered and collision is eliminated [33].

Advantages:

(i) Since it employs the TDMA principle, it eliminates collisions while maintaining energy conservation too.
(ii) Noncommunicating nodes switch off their radio and get into the sleep mode which eliminates overhearing.
(iii) Since receiving nodes broadcast SYNC signals to all nodes in the form of CTS packets, overemitting is avoided.
(iv) It decreases duty cycling as sleep time increases and lowers idle listening.

2.1.10. H-MAC. This protocol is built on the power saving mechanism (PSM) of IEEE 802.11 and slotted Aloha protocols to enhance performance using multiple slots. Time here comprises large frames each having an active and a sleeping part [34]. If any node has got data to send, it will ask for slots from the receiver node during active time and later transmit the packets in the prenegotiated slots. Nodes go into the sleep mode during sleep-time slots if they do not have data to transmit.

Latency in IEEE 802.11: if N is the number of hops, n the value for the current hop, $t_{cs,n}$ the back-off time, and t_{tx} the transmission delay, the total latency is calculated using the following equation:

$$D(N) = \sum_{n=1}^{N} \left(t_{cs,n} + t_{tx}\right), \tag{8}$$

and the average latency is given by the following equation:

$$E[D(N)] = N\left(t_{cs,n} + t_{tx}\right). \tag{9}$$

Throughput in H-MAC is given by the following equation:

$$\text{Th} = \frac{n_p n_m}{t_{act} + Ct_s}, \tag{10}$$

where n_p is the number of transmitted packets, n_m is the maximum number of nodes connected in a time frame, C is the number of slots with the same length, t_{act} is the listening time, and t_s is the sleep delay.

Advantages:

(i) It enhances performance using multiple slots.

2.1.11. Z-MAC (Zebra MAC). This rides on the strengths and offsets weaknesses of TDMA and CSMA. It yields very good channel usage and low delay under low contention and lowers collision among two-hop neighbors at a reduced cost. Its performance is strong via error synchronization, failure in assigning slots, and channel conditions that change with time. In the set-up phase, the protocol operates through certain steps including neighbor discovery, slot assignment, local frame exchange, and global time synchronization, which unless the topology of the network changes significantly do not run again. When the data are being transmitted, there is improvement in network throughput and energy efficiency which covers up for the initial upfront, operational costs [35].

Disadvantages:

(i) Initial slot assignment is quite a cumbersome task here since it is done at the beginning, yet this may not be at the same time in reality. It further assumes static links until setup happens again which too is impractical.

(ii) It also suffers from the hidden-terminal problem since slot owners have a small contention window and priority for slots allocated to them due to the ECN messages.

(iii) Synchronizing time also poses a challenge in this protocol due to a large clock drift between nodes.

(iv) Source and destination nodes too are not well coordinated in Z-MAC [36].

2.1.12. Q-MAC (QoS-Aware Media Access Control). This protocol offers high QoS with improved energy efficiency and accesses the wireless channel using MACAW as the underlying protocol [37]. Traffic from different nodes is prioritized for QoS satisfaction depending on how critical the data are. It mainly has two scheduling algorithms in WSNs: (i) intranode and (ii) internode scheduling. The *intranode* scheduling scheme adopts a first-in first-out- (FIFO-) based queuing algorithm based on application and MAC layer abstraction for data packet classification. *Internode* scheduling reduces idle listening and collision on the channel for improved energy consumption.

Advantages:

(i) It gives a high QoS with improved energy efficiency.

2.1.13. WiseMAC (Wireless Sensor MAC). WiseMAC is based on the Aloha protocol and employs preamble sampling in achieving low-power transmissions in infrastructure sensor networks by occasionally sampling the channel to check for activity [32,38]. The protocol employs the same method as B-MAC except that the sender shortens the extended preamble's length by scheduling its transmissions accordingly after learning the receiver's awake schedules. This is done by the receiver node putting its subsequent awake time on the ACK packet such that when the sender needs to send to it again, it starts the preamble shortly before it wakes up so as to avoid energy wastage in transmitting the preamble [32]. The data frame is repeated instead of the extended preamble for less traffic transmissions that have a lengthier preamble than the data frame in WiseMAC. After processing the frame, a node that is not the target recipient will go back into the sleep mode, while the recipient stays awake till the session ends thereby sending an ACK packet. For an idle channel, the power consumption is rather low with this method. Nevertheless, there is a problem of long wake-up preamble limiting throughput and causing an enormous power consumption overhead in reception [38]. All nodes overhearing a transmission bear the overhead together with the recipient node. The protocol can address issues concerning low-power transmission but has no mechanism of nodes adapting to varying traffic patterns [32]. To compensate for drift between clock at the access point and on the sensor node, there is a need to compute the wake-up preamble duration. The drift varies directly with time since previous resynchronization and is given by the following equation:

$$T_P = \min\left(4\theta l, T_W\right), \tag{11}$$

where θ is the time-base quartz frequency tolerance and l is the interval between two transmissions. Increase in traffic volume naturally reduces overhearing.

Radio model: for protocols with lower power, transition delays between transceiver states and their power consumption are modeled. Some of the states include *DOZE* where the transceiver neither transmits nor receives but is prepared to power on to the receive/transmit state very fast, *RX* where the transceiver listens to the channel, receives data, or demodulates data from the noisy/idle channel, and *TX* which is the transmission state by the transceiver. Let T_S be the set-up time needed to turn the transceiver from the DOZE to the RX or TX state, T_T the turn-around time needed to switch between RX and TX states, and B the transceiver's bit rate. P_Z, P_R, and P_T are the values of power consumed in the DOZE, RX, and TX states, respectively:

$$\begin{aligned} \widehat{P}_R &= P_R - P_Z, \\ \widehat{P}_T &= P_T - P_Z, \end{aligned} \tag{12}$$

where $\widehat{P}_R$ and $\widehat{P}_T$ are the changes in power as a result of being in RX and TX states, respectively.

Traffic model: let N be the number of sensor nodes connected to an access point (AP) and λ be the global rate at which the downlink Poisson traffic arrives at the AP from the fixed network. Traffic toward each sensor node is the same and is given by λ/N. Average packet interarrival time L with which data packets arrive at a node is given by $L = N/\lambda$. If T_C and T_D are the control packets and data packets, respectively, assuming a small traffic volume, the global interarrival $1/\lambda$ far exceeds the summation of durations for the data packet and control packet and the turn-around time. Mathematically, it is given as

$$\frac{1}{\lambda} \gg T_D + T_C + T_T. \tag{13}$$

The average power consumed by WiseMAC and transmission delay incurred are given by equations (14) and (15) [38]:

$$\begin{aligned} P_W = P_Z &+ \frac{\widehat{P}_R\left(T_S + (1/B)\right)}{T_W} + \frac{\widehat{P}_R\left(\overline{X} + T_D + T_T\right)}{L} \\ &+ \widehat{P}_R\,(N-1)\frac{\overline{Y}}{L}, \end{aligned} \tag{14}$$

where

$$\begin{aligned} \overline{X} &= 2\theta L\left(1 - e^{-\left(T_D/4\theta L\right)}\right), \\ \overline{Y} &= \frac{T_D^2 + 12T_D\theta L}{2T_W}\left(1 - e^{-\left(T_W/4\theta L\right)}\right), \\ D_W &= T_D + \frac{T_W}{2}\left(1 - e^{-\left(T_W/4\theta L\right)}\right) \\ &\quad + 2\theta L\left(2 - e^{-\left(T_D/4\theta L\right)} - e^{-\left(T_W/4\theta L\right)}\right), \end{aligned} \tag{15}$$

where P_W and D_W are the power consumed by WiseMAC and the transmission delay for WiseMAC, respectively.

Advantages:

(i) It achieves low-power transmissions in infrastructure sensor networks through preamble sampling while checking for activity.
(ii) It reduces overhearing due to increased traffic volume.

Disadvantages:

(i) It has a problem of long wake-up preamble limiting throughput and causing enormous power consumption overhead.

2.1.14. X-MAC. It is a simple, minimal transmission power MAC protocol that uses a short preamble with decoupling for the sending and receiving node sleep plans in WSNs. This leads to a significant reduction in energy usage at sending and receiving nodes and decrease in per-hop latency and gives flexibility in adapting to bursty and periodic sensor data sources [32].

Protocol design: X-MAC's design is aimed at increasing energy efficiency, reducing data latency, and increasing throughput. This enables application in different kinds of packetized and bit stream digitized radios as well as simplified, low-overhead, distributed application. The design is as such in order to address the low-power listening problems including overhearing, excessive preamble, and incompatibility with packetizing radios as in [32].

Asynchronous duty cycling: a node with information to transmit will send the packets after sending a preamble, whereas the remaining nodes will preserve their sleep schedules that are not synchronized. The receiver will sample the link on waking up and stays awake if it detects a preamble and if it is the target; it will return to sleep if it is not the target after getting the whole preamble.

Embedding the target ID in the preamble to avoid overhearing: the lengthy preamble is divided into many shortened packets embedded with the receiver node's ID so as to eliminate overhearing. One lengthy preamble is constituted by the shortened preamble packet stream. Any node on receiving the short preamble after waking up will examine the target ID on the packet. It goes back into the sleep mode right away resuming duty cycling if it is not the targeted recipient or stay awake for successive packets if it is the one. This approach checks energy wastage due to network density. It can also be used on several types of radios.

Reducing excessive preamble using strobing: it is possible to enable low-power transmission and also save energy by reducing time for preamble transmission through the use of enhanced preambles in addition to preamble sampling. X-MAC does not send an endless preamble packet stream but puts slight

pauses in between packets to allow the sending node to pause and listen to the channel and allow the receiving node to resend an early ACK packet during that pause. This makes the sending node begin sending data packets and the receiving node cut excessive preambles and thus decrease per-hop latency and energy wastage on unnecessary waiting and transmission.

Packetizing radios: low-power listening is limited in capability for supporting packetizing radios but X-MAC's short strobed preambles can support all types of digital radios.

Adaptation to traffic load: while a number of WSN applications are reliable in traffic production, they still must adapt to changing traffic. Nodes are going to have separate sleep schedules depending on traffic load. Considering systems that have time-fluctuating traffic, all predetermined static schedules are suboptimal.

Optimality: Buettner et al. [32] model the expectation of energy as follows:

$$\begin{aligned} E_s &= (\text{preamble energy} + \text{energy per ACK listen}) \\ &\quad * (\text{expected preamble} - \text{listen iterations}) \\ &\quad + (\text{energy to send parket}), \\ E_s &= \frac{\left(P_{Tx}S_p + P_{Rx}S_{al}\right)\left(R_l + R_s\right)}{R_l - S_p} + S_d P_{Tx}, \end{aligned} \tag{16}$$

where E_s = expected energy to send a packet; P_{Tx} and P_{Rx} are the transmission and receiver power, respectively; S_p, S_{al}, and S_d are the duration of the sender's preamble, ACK listen, and data transmission periods, respectively; and R_l and R_s are the receiver's listen and sleep periods, respectively. Expected energy to receive a packet is given as

$$\begin{aligned} E_r &= (\text{listen cycle energy} + \text{sleep cycle energy}) \\ &\quad * (\text{expected iterations for preamble arrival}) \\ &\quad + (\text{energy to send ACK}) + (\text{energy to receive packet}), \\ E_r &= \frac{\left(P_s R_s + P_{Rx}R_l\right)}{1 - \left(1 - P_d(t)\right)\left(R_l + R_s\right)} + R_a P_{Tx} + R_d P_{Rx}. \end{aligned} \tag{17}$$

And the one-hop expected latency is given as

$$\begin{aligned} L_{at} &= (\text{preamble duration} + \text{ACK listen}) \\ &\quad * (\text{expected number of iterations}) \\ &\quad + \text{duration to send packet}, \\ L_{at} &= \frac{\left(S_p + S_{al}\right)\left(R_l + R_s\right)}{R_l - S_p}. \end{aligned} \tag{18}$$

Advantages:

(i) It increases energy efficiency.
(ii) It reduces data latency.
(iii) It increases throughput.

2.1.15. A-MAC (Advanced MAC). A-MAC is a TDMA-based protocol whose design is aimed at enhancing the lifetime of the network for low rate and reliability of data transmission [39]. To assign time slots, it does not depend on a central controller like most protocols, but it gathers information from its neighbors so as to allocate the slots through a distributed algorithm. Energy is controlled by scheduling power when there is not a single event. Its design comprises many frames with each having many time slots which at the beginning have got a beacon transmitted to synchronize and exchange information with neighbors. Nodes to participate in the subsequent session are selected by the controlled node, so they do not go into the sleep mode with others [13]. This protocol operates under four time slot divisions, namely, initial, wait, discover, and active states. A node in the initial state begins by listening to the medium for the beacon packet from other nodes for network synchronization, which when received makes it change the timer through subtraction of beacon transmission time from reception time. To avoid synchronization problems due to drift and allow for continuity, the node has to get the strongest beacon signal. It then enters the wait state after choosing a number of waiting frames so as to reduce the probability that nodes get into the discover state simultaneously. In case the synchronization beacon is lost while still in the wait state, it will return to the initial state for another one. On expiry of the waiting counter, it enters the discover state and collects information from neighbors by listening to their beacon messages. On successful selection of a time slot, it enters the active state where it endlessly transmits beacons at the initiation of the slot. A node goes to sleep when a beacon has been transmitted and there are no more data for transmission or in case the neighbor's beacon received indicates no data packets are coming in [39].

Advantages:

(i) It can enable multicasting.
(ii) It enhances the lifetime of the network.
(iii) It offers reliable data transmission.

2.1.16. L-MAC (Lightweight MAC). Based on TDMA, this protocol is aimed at minimizing the number of transceiver switches, adapting the nodal sleep interval to data traffic amount, and making implementation less complex [40]. A packet is transmitted comprising a control and a data part. Control packets have a fixed size and the time slot controller's ID. They show the distance from the transmitting node for simplified routing to the network gateway as well as addressing target node and reporting data unit length and internode synchronization. Adjacent nodes are more interested in getting control packets from nearby nodes. The node's transceivers are switched off until the subsequent time slot unless the node is also addressed or the message is omnicast. If so, it listens to the data unit which may not fill the remaining time slot in totality. The transceivers on either side are turned off when the transfer of packets is successful. Energy wastage through idle listening in uncontrolled time slots is avoided via a short time-out interval. Network nodes

are able to communicate without colliding. This saves energy and increases the lifetime of the network. A node in L-MAC can only transmit one message per frame [13]. The network is set up in such a way that all nodes are not synchronized on being switched on. The gateway takes control of a timeslot so as to have them synchronized, and its immediate neighbor gets the control messages thereby synchronizing their timers with the gateway's. Neighboring nodes learn of multiple gateways' time slots in their vicinity just after one frame. The just synchronized nodes randomly choose time slots for controlling as long as they are not occupied. The nodes thus preserve a table for the neighboring nodes which makes a timeslot reusable after a minimum of 3 hops and prevents message collision. All nodes keep their slots before the battery is down or learn of an impending collision. All nodes can track their hop distances to a selected gateway node, and the information is communicated in control messages [40].

Advantages:

(i) It minimizes the number of transceiver switches.
(ii) It adapts the nodal sleep interval to data traffic amount.
(ii) It makes implementation less complex.
(iii) It offers improved network lifetime.
(iv) Energy wastage through idle listening in uncontrolled time slots is avoided via a short timeout interval.

Disadvantages:

(i) L-MAC is not very good for mobile SNs since slots are calculated only once.
(ii) It suffers idle listening because nodes keep listening to slots' control parts so as to get data and enable other nodes to join the network all the time [13].

2.1.17. C-MAC. This was designed for WSNs with 3 segments, namely, (i) aggressive RTS, having double channel check to assess the channel, (ii) anycast to initialize the flow, and (iii) convergent packet forwarding to stabilize the flow. It promises an improvement in energy efficiency, lower latency, and enhanced throughput while avoiding synchronization overhead though it operates only in low duty-cycling apps especially where there is less traffic. But when there is more traffic, the protocol employs anycast-based packet forwarding for waking the nodes up or finding a forwarder quickly and converges from route-suboptimal to route-optimal unicast. After this, energy could be preserved by the nodes using a synchronized wake-up plan [13].

Advantages:

(i) It improves energy efficiency.
(ii) It lowers latency.
(iii) It enhances throughput avoiding synchronization overhead.

Disadvantages:

(i) It operates only in low duty-cycling applications especially where there is less traffic.

2.2. MAC Protocols for WMSNs

2.2.1. Diff-MAC. This is a CSMA/CA-based, QoS-aware, hybrid-priority-based MAC protocol for WMSNs. It targets enhancing the channel utilization while making use of efficient service differentiation to do coordination for medium access of each traffic class while giving fair, quick data transfer. It offers QoS using the following features [14]:

(i) It balances energy consumption and delay through sensor node duty-cycling adaptation depending on the dominant traffic class.
(ii) It has a feature for fragmentation and message passing that breaks lengthy video frames into smaller packets that are transmitted as a burst, thus reducing retransmission costs if there are any MAC failures.
(iii) There is fair data delivery amongst the nodes and traffic classes, respectively, resulting from intranode and intraqueue prioritization, reducing unbearable performance.
(iv) The number of collisions can be reduced and packet latencies are kept within limits by adjusting the size of the congestion window in this protocol as per the traffic needs.

Advantages:

(i) Diff-MAC is very suitable for WMSNs since it adapts quickly to changes in network conditions.
(ii) It is a fair protocol.
(iii) It is scalable.

Disadvantages:

(i) It is quite difficult to monitor the statistics of the network.
(ii) Adapting dynamically to the network conditions is not an easy task.
(iii) It exhibits some degree of latency in packet delivery.

The implementation of Diff-MAC on Imote2 indicates that resource requirements can be met on the sensor hardware. Results of extensive simulation runs showed that Diff-MAC outperforms its competitors.

2.2.2. Cluster-Based On-Demand Multichannel MAC Protocol for WMSN (COM-MAC). This is an energy-efficient protocol in WMSNs with high throughput and reliable data transmission. It achieves energy efficiency by scheduling multichannel media access to reduce channel contention and remove collision, idle listening, and overhearing [41].

Design: it has a three-phased operation with the (i) request phase, (ii) scheduling phase, and (iii) data transmission phase.

Request phase: here, all nodes have two protocols each for sending requests to the cluster head (CH), i.e., the contention-based and contention-free TDMA/FDMA

protocols. The contention-based protocol allocates a channel per node for sending the request message, and free channels are useable as control channels. For contention-free protocols, a control slot is allocated when a network is deployed. Here, a slot per channel is allocated to a different node enabling requests to be sent without interfering with transmissions from other nodes. The contention-based protocol operates under two steps: (i) Control channel assignment phase: a channel is assigned per node for transmitting a request (REQ) message. A channel is assigned to multiple nodes, and nodes are evenly distributed to available channels for congestion control. (ii) Request transmission phase: nodes with data send a REQ message via the assigned control channel to notify the CH which responds with an ACK message. The contention-free TDMA/FDMA protocol too has two phases including the control slot assignment and request transmission phases. For unreliable channel conditions and low traffic load, the contention-based protocol is fit for use because of less collisions and congestion, whereas the contention-free protocol is preferable in heavy traffic load and reliable channel applications.

Scheduling phase: from the above phase, a schedule is generated by the CH for coordination, and this enables nodes to transmit and broadcast their data via control channels. The schedule for broadcasting only comprises data for nodes assigned to a channel. The schedule is rebroadcast for transmission reliability enhancement, but for energy efficiency, the node's transceiver is switched off when all the schedule is completely delivered till it is the time for transmitting again. The time slot and data communication radio channel are all included in the schedule generated for each node.

Data transmission phase: on getting the schedule, the sensor node sends its data as per the time slot and the channel assigned. The time slot is divided into data transmission and ACK sections. ACK supports link layer error control. Using implicit selective repeat ARQ, the CH sends an ACK message for all received packets. If there is no ACK, the packet is assumed lost, and hence, a retransmission during the subsequent interval is done. This enhances reliability of transmissions though excess delay is possible here which is also intolerable for real-time critical apps like video or audio. This gives rise to a hybrid MAC protocol to exploit the unused spectrum during data transmission sessions [41].

Advantages:

(i) It achieves energy efficiency by scheduling multichannel media access to reduce channel contention and remove collision, idle listening, and overhearing.
(ii) It maximizes network throughput by dynamically allocating time slots and channels for nodes by execution of a traffic-adaptive, QoS-aware scheduling algorithm depending on needs for QoS and traffic data information.
(iii) It increases transmission reliability by incorporating spectrum-aware ARQ to avoid spectrum wastage.

2.2.3. EQoSA. This protocol is developed for video and image transfer across WMSNs in a hybrid mode with provision of QoS. Concerning the active sensor nodes and their traffic loads, it employs dynamic session sizes and changes bitmap-assisted (BMA) MAC's fixed session size. Nodes declare any data available for transmission during contention time. This forces the cluster head to make a schedule of how slots have been assigned. It communicates this schedule to the nodes giving each node a given number of slots per session and so enabling the busty traffic to be accommodated [14]. The protocol does not perform well with DiffServ mechanisms but does well with schemes supporting busty traffic loads, and strong cluster heads are necessary in WMSNs that can assign and broadcast slots.

2.2.4. QoS-Supported Energy-Efficient MAC (QE-MAC). Based on the IEEE 802.11e standard, this protocol ensures fairness and has low latency and jitter and better efficiency on energy usage than other QoS-aware MAC protocols for WMSNs. It operates under a two-phased routine.

The first phase introduces the innovating priority mechanism in IEEE 802.11e QoS, and several data types are assigned priorities. Its coordination function is a combination of distributed, contention-based channel allocation and centralized, polling-based channel access mechanisms.

The second phase preserves the nodes' energy using dynamic duty cycling. There is an exchange of RTS/CTS packets to supplement CSMA/CA by muting nodes near the sender/receiver during packet duration [42].

Advantages:

(i) It ensures fairness.
(ii) It has low latency and jitter and better efficiency on energy usage than other QoS-aware MAC protocols for WMSNs.

2.2.5. Black-Burst (BB) Contention. A contention-based MAC scheme for enhancing instantaneous QoS access to carrier-sense WSNs is proposed in [42]. In this scheme, nodes contend for the channel until it is free. Real-time packets are given higher priority over non-real-time packets, and if a node is transmitting real-time packets, it is prioritized over other nodes [15]. Voice and video are some of the real-time apps in consideration here which periodically access the common radio channel when transmitting. These need constrained end-to-end delay of packets at the MAC layer. This protocol majorly does the following [42]:

(i) Prioritization of real-time traffic
(ii) Implementation of the round-robin mechanism in real-time nodes
(iii) Bounded access delays among real-time packets

(iv) Implementation of real-time communication with varying needs of bandwidth, good for multimedia traffic

A mechanism for dynamically assigning packet priorities based on their deadlines and the traversed hops is proposed in [43]. The ReAllocative Priority (ReAP) mechanism is meant to provide QoS at the MAC layer to video traffic. This scheme introduces adaptive TXOP by modifying the TXOP dynamically based on the queue length [15]. Other protocols in this category include dynamic duty cycle and adaptive contention window-based MAC protocol, frame sharing (FRASH) MAC protocol, and real-time independent channels (RICH) MAC protocol.

Advantages:

(i) It can implement real-time communication with varying needs of bandwidth.

2.3. Challenges and Opportunities

(a) One of the challenges in satellite transmission is cross-talk leading to interference in different frequency bands. It is necessary to increase the rise and fall times as well as protect sensitive nodes as we circumvent floating ones if we are to regulate cross-talk as in satellite communication therefore. We further need to investigate the possibility of using differential signaling. This constitutes an opportunity for further research.

(b) Another challenge is that the hidden and exposed terminal problems are still noticeable despite the use of RTS/CTS solutions in MAC since collisions are still possible from transmissions of separate nodes. At the MAC layer, ACKs might block efficient channel usage by prohibiting exposed terminals from reusing the channel and may be eliminated leaving no proof of sender data packet reception. Using a directional antenna is one of the proposed techniques of high-speed WSNs like IEEE 802.11. This constitutes an opportunity for further study into these issues for better solutions, though.

(c) Another challenge is that, since nodes in contention-based MAC protocols are mobile, there are many collisions by the packets and schedule inconsistencies in schedule-based protocols in the two-hop neighborhood information when nodes are entering or leaving [44]. Deng et al. [45] propose a mobility-based clustering (MBC) protocol for mobile nodes to enhance the packet delivery rate. This protocol in comparison with CBR has 25% chances of decreasing collision-induced packet losses and 50% chances in comparison with low-energy adaptive clustering hierarchy (LEACH).

(d) The process of analyzing and evaluating routing performance for WMSNs is quite complex because of the bursty multimedia traffic. Because of the way encoding scheme frames are structured as well as the changes in between the scenes that occur naturally, the compressed video frames display some substantial burstiness on several time scales. There will also be frequent route recalculation that is as well energy intensive for nodes frequently capturing and transmitting energy. The solution here could as well possibly be the use of the MBC protocol. Route recalculation too should be dynamic to cater for the mobile nodes which might be in and out of the range at different instants of time. Recently, the use of smart dumpsters for effective waste management in smart cities has also been proposed as a possible solution [46].

(e) For WSNs, their routing protocols do not look at multimedia applications which require enormous bandwidth and processing energy and should be transmitted in real time with utmost fidelity. Extra challenges in routing are imposed in WMSNs in deployment of heterogeneous sensors since there could be audio, video, and still pictures in addition to scalar data during transmission. Heterogeneous data handled also have different QoS requirements, and with many different business needs, different services with varying requirements give rise to a big challenge for the routing design. There is a need to design protocols that can support WMSNs such as EQoSA [14] for this purpose.

Table 2 summarizes the reviewed MAC protocols for WSNs and the ones designed for suitability in WMSNs.

3. QoS Strategies at the Application Layer

We discuss many protocols used to solve different needs of communication between machines.

3.1. Constrained Application Protocol (CoAP). It is a request/response protocol designed by the Internet Engineering Task Force (IETF) for constrained devices such as those having insufficient RAM or CPU and WPANs with low-power constraints, resulting in poor transfer of packets and enormous overhead. The protocol was designed for machine-to-machine (M2M) apps and system automation to reduce bandwidth requirements, increase packet transfer, decrease the overhead, and make work simpler, leading to lightweight implementation. HTTP commands like GET, POST, PUT, and DELETE are used in a client-server architecture. Using the publish/subscribe architecture, CoAP can support multiple users and hence yield better performance via multicasting, enabling asynchronous message exchange. This protocol can support both multicasting and unicasting via UDP and hence is a reliable and simple protocol. This is done via the messaging sublayer (for reliability based on UDP) and the request/response sublayer (for communication and interaction) [16].

It uses the following messages:

(i) Confirmable message: this guarantees reliability using ACK messages sent by the receiver node. These could be synchronous or asynchronous if there is a need for more computational time.

Table 2: Comparison of MAC layer protocols used in QoS provisioning.

Protocol	Features	QoS parameter	Comments	Access	Priority assignment	Energy eff.	Designed for
S-MAC [22]	Collision avoidance	Energy, latency	Requires periodic synchronization	CSMA	Hybrid	Medium	WSNs
B-MAC [23]	Collision avoidance	Energy, delay, throughput	Flexible interface	CSMA	Dynamic	High	WSNs
T-MAC [25]	Collision avoidance	Energy	Complex, nonscalable	CSMA	Dynamic	High	WSNs
C-MAC [13]	Dynamic channel allocation	Energy	Multichannel MAC	CSMA	Dynamic	Medium	WSNs
CU-MAC [24]	Multichannel mechanism	Throughput, latency	High packet delivery ratio	TDMA	Dynamic	—	WSNs
A-MAC [39]	Per flow service guarantees	Delay	m-ary tree model, dynamic priorities	TDMA/ CSMA	Dynamic	Medium	WSNs
ER-MAC [30]	Emergency response	Energy, delay, delivery ratio	Adapts to traffic and topology changes	TDMA/ CSMA	Hybrid	Higher than Z-MAC	WSNs
RI-MAC [31]	Collision detection	Delay, packet delivery ratio	High packet delivery ratio and throughput	CSMA	Dynamic	Medium	WSNs
SRI-MAC [33]	Collision avoidance	Energy, delay	Good data transfer and energy consumption	TDMA	Dynamic	High	WSNs
H-MAC [34]	Slotted sleep time	Throughput, latency	Data transmission only in sleep time	CSMA/ Aloha	Dynamic	Medium	WSNs
Z-MAC [36]	Collision avoidance	Energy, delay, throughput	Very good channel usage and error sync	CSMA/ TDMA	Dynamic	Medium	WSNs
Q-MAC [37]	Intra/internode scheduling	Energy	High QoS, improved energy efficiency	CSMA	Hybrid	High	WSNs
X-MAC [32]	Overhearing avoidance	Energy, delay, throughput	Reduces excessive preamble by strobing	CSMA	Adaptive/ static	Medium	WSNs
L-MAC [40]	Collision avoidance	Network lifetime, energy	Reduces the number of transceiver switches	TDMA	Static	Medium	WSNs
WiseMAC [38]	No downlink channel collision	Power, delay	Significantly low-power consumption	Aloha/ CSMA	Static	Higher than Zigbee	Infrastructure WSNs
AR-MAC [27]	Collision avoidance	Energy, delay	Uses data packets and control packets	TDMA	Static	Higher than IEEE 802.15.4	WBANs
BB-MAC [42]	Real-time traffic prioritization	Delay	Employs round robin in real-time nodes	CSMA/ TDMA	Dynamic	High	WMSNs
COM-MAC [41]	Collision avoidance	Packet delay, throughput	Balances reliability and retransmission	TDMA/ FDMA	Dynamic	High	WMSNs
EQoSA [14]	Busty traffic load accommodation	Energy	Provides QoS support for video and images	TDMA/ CDMA	—	High	WMSNs
QE-MAC [42]	Collision avoidance	Energy, jitter, latency	Runs central polling-like channel access	CSMA	Dynamic	High	WMSNs
R-MAC [26]	Collision avoidance	Energy, delay	Reservation based	—	Static	High	Underwater SNs

(ii) Nonconfirmable message: here, no ACK messages are sent back.

(iii) Acknowledgment (ACK) message: this confirms receipt of a confirmable message.

(iv) Reset message: this confirms receipt of an unprocessed message, in case any critical parts necessary for interpreting the message are missing.

(v) Piggybacked response: the receiver directly responds while getting the ACK message.

(vi) Separate response: the reply by the receiver is discrete from the ACK message.

CoAP has no in-built security features. It is secured by the datagram transport layer security (DTLS) that runs on top of UDP though it was not designed for the IoT. This makes its suitability debatable since DTLS offers no support for multicasting. It also needs more packets for handshaking, thereby increasing the traffic volume and computational resources and so shortening the lifespan of mobile devices since they use batteries [17].

Advantages:

(i) It can support multiple users and hence yield better performance via multicasting.

(ii) It enables asynchronous message exchange.
(iii) It is a reliable and simple protocol.

3.2. Message Queue Telemetry Transport (MQTT). Developed by IBM and later improved by OASIS with an intention of decreasing bandwidth, MQTT is a simple, open-source publish/subscribe protocol analogous to the client-server version. It is appropriate for lightweight IoT applications and M2M transmissions in constrained situations like limited power, computation capability, and memory, or bandwidth. The network bandwidth and the computational resource usage are decreasing since clients are not updated making publish/subscribe protocols outperform request/response protocols via IoT requirements [16]. This protocol can run over TCP/IP and so has the following *advantages*:

(i) Delivering of packets with reliable guarantee
(ii) Enabling multicasting
(iii) Remote device session establishment
(iv) Reduction of transport overhead and protocol exchange to minimize network traffic
(v) Provision of the notification mechanism when anomalies occur [47]

MQTT's design targets economical use of bandwidth and battery, so it has a major application in Facebook Messenger. MQTT ensures QoS and reliability of message delivery by providing the following three QoS levels:

(1) Fire and forget: a message is sent once depending on the network's best effort, and there is no need for ACK (lowest QoS level).
(2) Delivered at least once: a message is sent at least once, and an ACK message is needed to avoid duplicates.
(3) Delivered exactly once: by a four-way handshake, it ensures a message is sent exactly one time (highest QoS level)

The architecture: it has three key components: (i) publishers—lightweight nodes sending data to the broker before returning to sleep; (ii) broker, which organizes sensory data in topics and sends them to interested subscribers; and (iii) subscribers—IoT apps interested in data sent by sensors through the broker, as shown in Figure 3.

MQTT is a simple, lightweight protocol that is better than CoAP via low and high packet loss probability in terms of throughput and exhibits lower latency in low sampling rates [48]. It supports a number of implementations for embedded devices and mobile applications and offers web-socket support too. Nevertheless, it has some shortcomings:

(i) It is only employed in simple data types.
(ii) It exhibits high latency in high sampling rates.
(iii) It is unsuitable for real-time apps in such cases.
(iv) It does not provide automatic discovery and/or sufficient security at the protocol level [49].

3.3. Secure Message Queue Telemetry Transport (SMQTT). This is an extension of MQTT that is based on lightweight attribute-based encryption using elliptic curve cryptography. The protocol has three entities, as shown in Figure 4: (i) publisher that publishes information under a certain topic, (ii) subscriber that obtains the same topic information via a broker, and (iii) public key generator (PKG) or broker that acts as the trusted third party. The protocol has four phases, namely, (a) set-up phase, (b) encrypt phase, (c) publish phase, and (d) decrypt phase.

(a) Set-up phase: publisher and subscriber provide a unique ID with attributes for registration with the PKG. This generates a secret master key set and public parameters that it publishes with set $U = \{A_1, A_2, \ldots, A_n\}$. For ciphertext policy-attribute-based encryption (CP-ABE), the PKG gives a key set to devices with attributes [50].
(b) Encrypt phase: based on the access tree, the publisher creates an access policy. An access tree is sent by the publisher to the PKG for generating the key-generation policy in the key policy- (KP-) based ABE algorithm. The publishers know beforehand the topics and subscribers to access as well as the rules for access. Subscribers are issued with private keys for all communication sessions. Publishers in the CP-ABE algorithm come up with access trees and rules, where the payload is encrypted by the publisher who also issues more info needed for decryption and the rules. With the help of public key cryptography, data are encrypted giving rise to ciphertext on the basis of KP/CP-ABE and the receiver gets SUBACK from the broker [51].
(c) Publish phase: ciphertext is embedded in the SPublish command as payload, and topic name is defined in the header by the publisher. The broker receives the SPublish packet and replies with PUBACK prompting the publisher to respond with the PUBREL packet. The broker then broadcasts to topic subscribers and deletes the data, forwarding the PUBCOMP packet to the sender node.
(d) Decrypt phase: the encrypted information is decrypted by the subscriber with the help of a secret key. Considering KP-ABE on verification that it satisfies the access policy, the subscriber invokes the PKG for a corresponding key which it sends after verifying the request. With CP-ABE, subscribers use the private key for decryption after authentication, thereby enabling offline interaction [51].

Advantages:

(i) The information being encrypted makes it more secure.

3.4. MQTT for Sensor Networks (MQTT-SN). This is a version of MQTT designed to operate under environments of low bandwidth, high link failures, and short message length. It is a publish/subscribe (pub/sub) protocol designed for

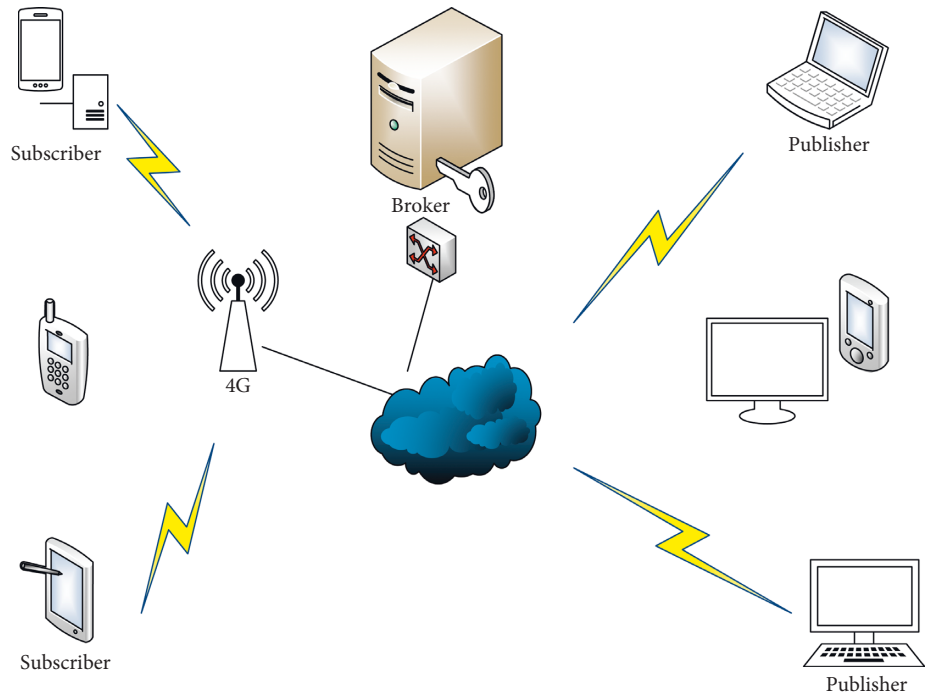

Figure 3: Architecture of the MQTT protocol.

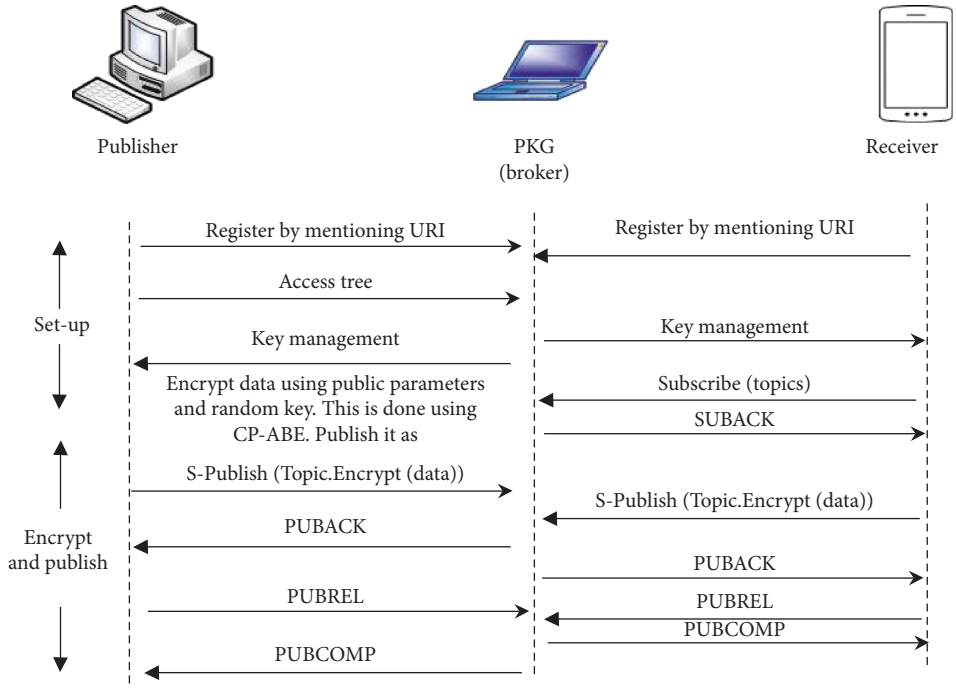

Figure 4: SMQTT protocol.

UDP-based WSNs to visualize communication for low-power devices [51,52]. It comprises a publisher device acting as the MQTT-SN client to forward communications to a gateway that also changes the MQTT-SN message to the MQTT message before delivering it to the broker that sends it to MQTT-SN subscribers [51].

It has a gateway for changing protocols from MQTT to MQTT-SN (Figure 5) [52]. This protocol supports nodes in the sleeping mode using an offline keep-alive procedure that can be used by battery-operated devices to get into the sleeping mode. To support sleeping clients, MQTT-SN has a new offline keep-alive procedure with which battery-operated devices can go to the sleeping state in which their messages are buffered at the server until they are awake. In MQTT-SN, clients with no preconfigured gateway address can figure out the real address of their operational gateway

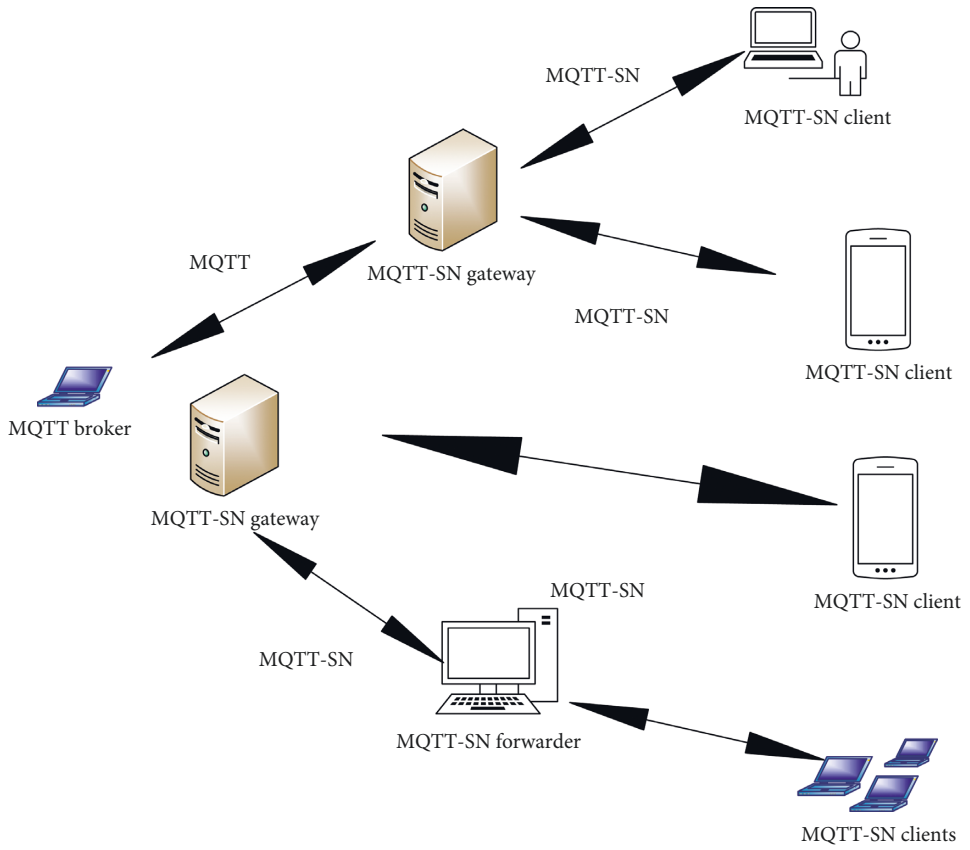

Figure 5: MQTT-SN architecture.

within the network through a discovery procedure. Simultaneously, we may have several gateways in a single wireless network that can be cooperative via load sharing.

MQTT-SN architecture: it is composed of MQTT-SN clients, gateways, and forwarders (Figure 5). Through the MQTT-SN gateway (GW), the MQTT-SN protocol is used by the MQTT-SN clients to connect to the MQTT server. Integration between the MQTT-SN GW and the MQTT-SN server might be possible or not. Translating between the MQTT-SN and the MQTT for a stand-alone GW is done using the MQTT-SN protocol. If a GW is detached from the network, it can be accessed by the MQTT-SN clients through a forwarder. This just encapsulates the received MQTT-SN frames that are on the wireless side before forwarding them to the GW unaltered. It then decapsulates the received frames too, resending them to the client as sent by the GW [52]. There are 2 types of gateways: (a) transparent gateway and (b) aggregating gateway, as shown in Figure 6.

(i) Transparent gateway: this launches and sustains a direct MQTT connection between the MQTT-SN client and the MQTT server which is entirely reserved for end-to-end, client-server messaging. The number of servers and that of MQTT-SN clients connected to the GW are the same. This transparent GW translates syntax between these protocols and is easier to implement compared to the aggregating GW. But the MQTT server must support different sessions for all connected clients. Some MQTT servers are configured to enable few parallel sessions.

(ii) Aggregating gateway: this maintains a single MQTT connection to the server on behalf of all clients. All exchanged messages from the MQTT-SN client to the aggregating GW are directed to the GW which sorts the information to forward to the server. The aggregating GW decreases the number of MQTT sessions simultaneously supported by the server in WSNs with many battery-operated sensors and actuators (SAs) having limited storage and processing capabilities. It is however more complex to implement compared to a transparent GW. The general format of the MQTT-SN protocol message has two or four message header octets and n octets of variable header that is optional depending on the message in question. MQTT-SN has 256 message types (MsgType) with those from 0X1E–0XFD, 0X19, and 0XFF reserved for future usage [51].

Advantages:

(i) The protocol can operate under environments of low bandwidth, high link failures, and short message length.

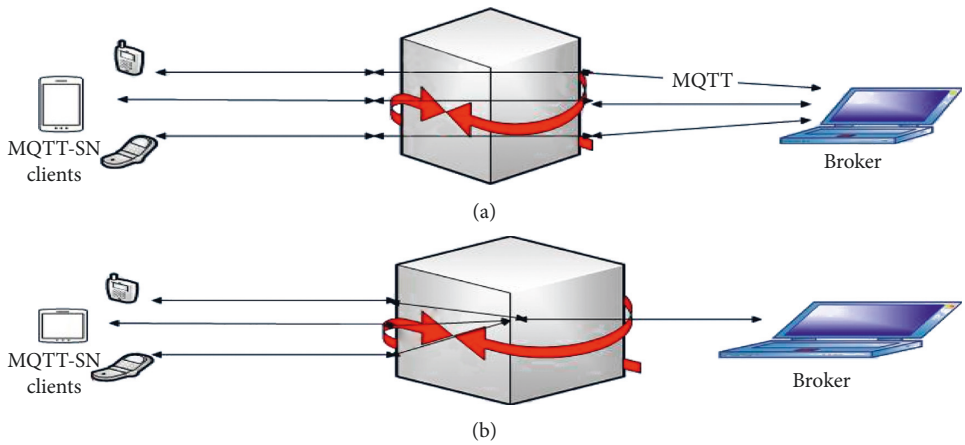

Figure 6: (a) Transparent gateway and (b) aggregating gateway.

(ii) It is a pub/sub protocol designed for UDP-based WSNs to visualize communication for low-power devices.
(iii) It can be optimal for low-cost, battery-operated nodes having low processing power and less storage.
(iv) It supports nodes in the sleeping mode using the offline keep-alive procedure used by battery-operated devices.

3.5. The Extensible Messaging and Presence Protocol (XMPP). The standardization of this protocol by the IETF made it popular for usage over the Internet especially in instant messaging and chatting platforms. It is however susceptible to spamming attacks, and its failure to have worldwide support has made Google too withdraw its support of the standard [17]. The protocol is mainly useable in the client-server architecture but can also be applicable in client-client XMPP communication [53]. Among the application layer protocols, XMPP is the only one that supports both asynchronous (publish/subscribe) and synchronous (request/response) services while running over TCP but can still run over other protocols.

Given the simplicity of its implementation and ease of usability in XMPP, instant messaging (IM) is suitable for XMPP and is implemented using the message stanza <message/>. These stanzas are usable in one-to-one or multiuser sessions or chatting or in error reporting and sending headlines. Attribute type of the stanza defines the message type. The attribute type could be a chat, group chat, headline, normal, or error. An instant messaging network in XMPP can support many XMPP clients and servers with a gateway configured to enable other IM network users to connect to other XMPP users. The XMPP server mainly controls sessions and XML streams to clients and servers in addition to routing XML stanzas in the streams [53]. The TCP ports 5269 and 5222 have been, respectively, registered for the XMPP server-to-server connections and XMPP client connections by the Internet Assigned Numbers Authority (IANA). XMPP enables specification of XMPP Extension Protocols (XEP) to enhance its usefulness [17].

XMPP security: clients connect to the server using transport layer security (TLS) and simple authentication and security layer (SASL) protocols which are at the same time used by the servers for interdomain communications. For data confidentiality and integrity, TLS will encrypt the streams of XML for all sessions. To confirm the identity of any client that needs to access the server, the SASL protocol does the authentication of the XML stream. For any communicating session with a server to start, the client needs to resolve the server's DNS hostname. For protection of the server of the client from third-party attacks, the server maintains the authentication details secret, i.e., IP address and access method, and only original connections of the server are required.

In server-to-server communications, the SASL protocol is employed for authentication and confidentiality. Server-to-server communications may as well be deactivated from a server by an administrator depending on the organizational policy. To guard against domain-spoofing attacks, server dialback can be a good solution since servers support it securing more the XML stanzas. Nevertheless, it cannot be used to authenticate, secure, or encrypt in-server streams, and the server identification resulting therein is not strong enough to be relied on. It further cannot secure against DNS poisoning attacks and IP session hijacking for remote domains [53]. High-security domains can use TLS and SASL.

Advantages:

(i) It is popular for usage over the Internet especially in instant messaging and chatting platforms.

Disadvantages:

(i) It is susceptible to spamming attacks.
(ii) It has failed to have worldwide support making Google withdraw support of the standard.
(iii) It is not secure against DNS poisoning attacks and IP session hijacking for remote domains.

3.6. Advanced Message Queuing Protocol (AMQP). This protocol can be utilized by a number of platforms for message exchange since it is open source and written in many languages. It is a publish/subscribe model depending

on a reliable and efficient messaging queue with capacity for instant messaging or e-mail for different applications especially in IoT environments. It differs from others in the sense that message types can be specified and have their source traceable and how to do trade-off between performance, security, and reliability. For reliability of message transmission, AMQP makes use of some delivery guarantees like at most once, at least once, and exactly once. It can also however utilize the TCP transport layer for guaranteeing reliability. The AMQP's publish/subscribe method has two components, namely, (i) exchange queue meant for routing of messages to an appropriate order in the queue and (ii) message queue that acts as storage pending their delivery to receivers [16]. In this protocol model, the broker can make routing decisions contrary to other messaging systems where apps using the queue have in them embedded the decision-making logic for recipients or senders. The developer may need to check each of the affected apps so as to change the logic for routing and delivery of the messages [54].

Advantages:

(i) It is convenient.
(ii) It has capacity for interoperability between different vendors.
(iii) It can use open standards to enable connection between business partners.
(iv) It can enable innovations based on AMQP foundations [55].
(v) It can enable supporting of mission-critical applications in e-commerce.
(vi) It can guarantee reliable delivery of messages.

Disadvantages:

(i) It is not very appropriate for real-time applications.
(ii) It may not offer automatic discovery.
(iii) Interoperability is not guaranteed.
(iv) It lacks open-source libraries for constrained gadgets [49].

3.7. Data Distribution Service (DDS). A product of the Object Management Group (OMG), DDS, is a pub/sub protocol meant for M2M communications with two sublayers. These are the data-centric pub/sub sublayer responsible for connecting anonymous data publishers to subscribers and the data-local reconstruction sublayer that is optional and integrates DDS into the application layer [56]. The DDS architecture connects system components, as shown in Figure 7. Publishers and subscribers are disconnected with respect to (i) time, (ii) space, (iii) flow and behavior, (iv) platforms, and (v) programming languages [57]. A DDS data publisher generates topics, and publisher-subscriber coupling is represented as topic name, data type schema, and publishers and subscribers' QoS attributes. Delivering of data is done by the publisher layer. The data writer and publishers together make a decision about any necessary alterations for subsequent forwarding to the subscribers who need to send data to IoT applications. Data readers are responsible for delivering topics to the subscribers after reading through them [56].

Some of the major components of the DDS architecture include the following [57]:

(i) Topic: this is a logical path specifying publication and subscription's data type between DataWriters and DataReaders. For a successful session to begin, topic names, types, and DataWriters and DataReaders' QoS should correspond.
(ii) Domain: this is a logical setting for transmission that can be utilized for separating and optimizing network transmission in a setup of similar applications. If DDS apps share a domain ID, then they can exchange data among themselves.
(iii) DataWriter and DataReader: these are terminal objects for writing and reading messages, respectively, to and from a universal data space.
(iv) Participant: this is an object that stands for a publisher or subscriber of the DDS app in a domain acting like a container for other objects.

Advantages:

(i) It is highly reliable and offers QoS support.
(ii) It is highly scalable.
(iii) It is appropriate for real-time apps.
(iv) It is fault tolerant.
(v) It has high level of interoperability.
(vi) It does not need brokers that may act as bottlenecks.
(vii) It offers automatic discovery.
(viii) It has a user-defined data structure for topic.
(ix) It supports publisher and subscriber decoupling.

Disadvantages:

(i) It is originally developed for only stand-alone LANs.
(ii) A lot of memory is needed.
(iii) It has no open-source libraries to support constrained devices.
(iv) It is hard to design and configure compared to other protocols [49].

3.8. RESTful Services: REpresentational State Transfer (REST). This is an architectural style based on a set of principles that define and address networked resources. It offers web services that enable transmission from the HTTP device to the device within the IoT architecture and enables clients to reach the server [16]. This protocol utilizes HTTP commands to backup the request/response messaging model. HTTP is a popular and secure protocol in the WWW using TLS/SSL [16]. A RESTful application has the following characteristics:

(i) State and functionality are separated into distributed resources.
(ii) All resources are distinctively named by the use of unique HTTP commands (GET, POST, PUT, or DELETE over the Internet).

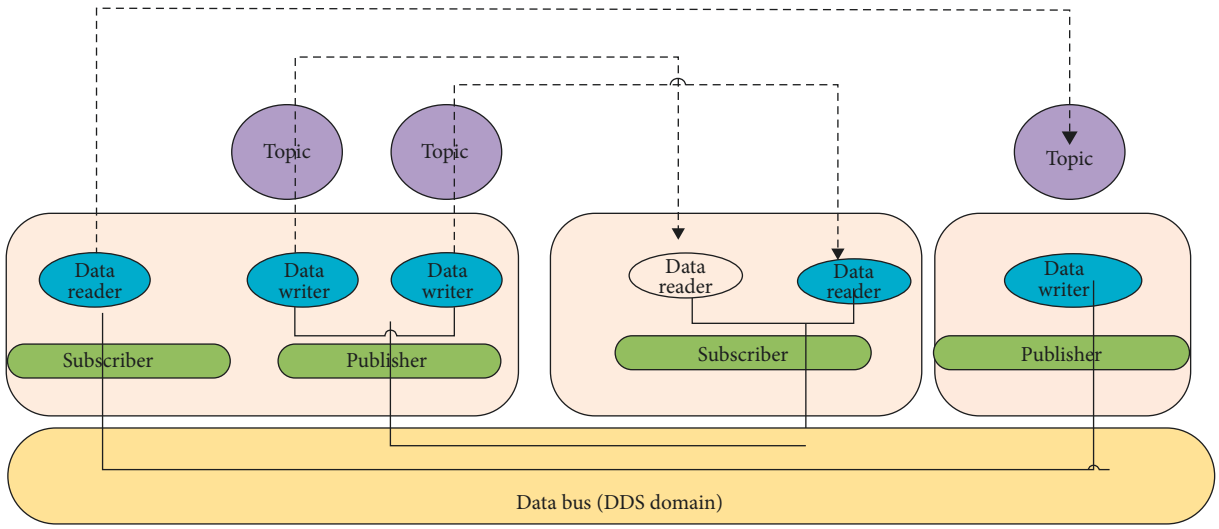

Figure 7: DDS architecture.

(iii) It is a client/server, stateless, layered protocol that supports caching. REST uses the same methods as HTTP for all requests or responses, with POST and GET for creating and retrieving resources, PUT for updating and changing the state of resources, and DELETE for removing resources.

Advantages of REST over arbitrary web services (SOAP) in the IoT context:

(i) It reduces overhead.
(ii) It reduces parsing complexity.
(iii) It is stateless.
(iii) It integrates more tightly with HTTP [58].
(iv) It is easy to implement and learn. This together with its natural fit over HTTP makes it a preferred method for Web 2.0 applications to showcase their data and secure M2M sessions in the IoT.
(v) Applications that support RESTful web services exhibit better performance than the resource-constrained WSN nodes [59].
(vi) REST is important for the IoT since it supports commercial M2M cloud platforms.
(vii) Its implementation in smart phones and tablet apps is also easy since it needs only an HTTP library that is accessible to all operating system platforms [56].

Disadvantages:

(i) Although it enjoys wide usage in commercial M2M platforms, it is not likely to dominate because it is hard to implement.
(ii) Its HTTP usage makes it incompatible with constrained M2M devices leaving its usage to end applications.

3.9. WebSockets. It is a web-based protocol operating on a single TCP channel and providing full-duplex transmissions. It permits secure duplex communication between a client executing unreliable code in a restricted environment and a distant node communicating from that code while using the origin-based security model employed by web browsers over TCP [60]. The protocol is not a request/response model and is not a publish/subscribe model. So, unlike other protocols, a web-socket session is established by the client initiating a handshake to the server to launch the session. This, once established, a full-duplex client-to-server connection begins asynchronously and runs until terminated by either party [56]. It gives a model for browser-based apps utilizing bi-directional communication with no need of many HTTP connections to servers.

WebSocket comprises two parts, i.e., (i) the *handshake* having a client's message and the server's handshake response and (ii) the *data transfer*. HTTP headers should be repeated by applications in all client requests and server responses arising from the endless polling which might increase the communication overhead subject to the application [61]. WebSocket can be used to come up with scalable, real-time web applications since it can enable full-duplex transmission via one socket across the Web. Its security can be ensured with the help of TLS/SSL. This protocol is suitable for resource-constrained devices, and IoT applications will not have a good user experience with it since it operates a client/server architecture. Nevertheless, it can outcompete all TCP-based protocols because of the following advantages:

(i) It is fit for real-time transmission.
(ii) It has improved security.
(iii) It minimizes transmission overhead.
(iv) It can provide efficient messaging systems using WebSocket Application Messaging Protocol (WAMP) [17].

3.10. Streaming (Simple) Text-Oriented Messaging Protocol (STOMP). This is a TCP-based protocol designed for text message-oriented middleware (MOM) with the use of

HTTP-like commands for interoperability among platforms, languages, and brokers. Data communication takes place between the client and the broker in line with commands (like CONNECT, DISCONNECT, ACK, NACK, SUBSCRIBE, UNSUBSCRIBE, SEND, BEGIN, COMMIT, or ABORT), then headers in the form <key> : <value> (one per line), then a blank line, and then body content, terminating in a null character. Server-client communication takes place over a MESSAGE, RECEIPT, or ERROR frame, with header and body content format being analogous. MOM products supporting STOMP include Apache ActiveMQ or Fuse Message Broker, HornetQ, Open Message Queue (OpenMQ), RabbitMQ, and syslog-ng through its STOMP destination plugin [62]. In addition to these, other application layer protocols include simple media control protocol (SMCP), lightweight local automation protocol (LLAP), simple sensor interface (SSI), lightweight M2M (LWM2M), XMPP-IOT, and simple object access protocol (SOAP).

Advantages:

(i) It is lightweight for constrained networks.
(ii) It has flexibility to choose quality of services with the given functionality.
(iii) It is standardized by OASIS Technical Committee.
(iv) It is easy and quick to implement.

Challenges and opportunities:

Here, security threats are imminent because of the open-endedness of the application layer [63]. These could include the following:

(i) There is poor or lack of a security design for an application's function. Coming up with a good design to address this could be a good idea/opportunity.
(ii) Some of the programs might have backdoors evading otherwise secure controls to allow unauthorized access to network resources. We require mechanisms to patch up such loopholes.
(iii) Applications that lack or have weak authentication mechanisms can be easily targeted by unlicensed users to abuse the network. Such mechanisms need to be strengthened further.
(iv) Excessively complicated access control rules may because of their misunderstanding or poor writing give unauthorized access allowing dubious activity rather than protecting the network they purport to protect. These should be simple and clear to understand by all users.
(v) In booting workstations without disks and managing network devices, we usually use the TFTP protocol although access does not require authenticating the user by username/password. This makes an intruder able to gain easy access to config/access info as long as he can guess the filenames. This needs to be fixed.

A few steps can be used to secure the application layer. To protect network data transmitted across the network, applications need to strengthen their encryption and authentication mechanisms. Apps too need to enable strict controls of privileges for data access using good mechanisms to balance between usability and effectiveness [63]. Applications handling sensitive data need detailed logging and audit capability as well as testing and reviewing.

Table 3 summarizes the application layer protocols discussed with their weaknesses and differences among them.

4. Cross-Layer QoS Strategies

4.1. Service-Differentiated Real-Time Communication Scheme (SDRCS). This is an event-driven routing protocol that routes real-time traffic by featuring a cross-layer packet-forwarding design in which it integrates the real-time routing functionality with a new priority-based MAC scheme. The protocol approximates distributed packet traversal speed for traffic classification and admission control on the basis of this design. It further localizes decision-making by prioritizing packet forwarding to maximize the speed of packets [64].

Advantages:

(i) It improves bandwidth utilization: bandwidth degradation due to unschedulable packets can be avoided.
(ii) It offers a lightweight packet schedulability estimation mechanism. Here, it uses received signal strength and admission control, and then early missed-deadline packet-dropping policies are made.
(iii) It delivers high end-to-end throughput via higher source data rates alongside strict end-to-end latency needs.
(iv) Nodes which meet real-time requirements of less traffic load and better channel quality are given highest priority when it comes to packet forwarding.

The SDRCS operates via 5 mechanisms, namely, (1) RSS-(Rich Site Summary/Really Simple Syndication-) based grouping, (2) admission control, (3) prioritized queuing, (4) real-time MAC, and (5) dynamic forwarding. This protocol has a queuing policy that employs per-hop deadline-based queues, and the nodes schedule their packets using FIFO priority-based queues [1]. The principal features of this protocol are mentioned as follows:

(i) It is event driven and is easily adaptable to any network changes.
(ii) It does not need additional hardware for localization or multichannel transmission.
(iii) The cross-layer packet-forwarding strategy can be used when transmitting multimedia traffic.
(iv) It is able to efficiently circumvent any voids therein.

Early deadline miss (EDM) policy drops all unscheduled packets. The SDRCS circumvents wrong packet drops due to EDM since it adjusts to network changes easily.

4.2. Sensor Fuzzy-Based Image Transport (SUIT). This protocol regulates congestion using a scheme based on fuzzy

Table 3: Differences and problems with application layer protocols.

Protocol	Architecture	Transport	QoS options	Security	Weaknesses
CoAP	Request/response	UDP		DTLS	No in-built security features. DTLS does not support multicast
MQTT	Publish/subscribe	TCP		TLS/SSL	Insufficient security at the protocol level since payload values are not encrypted
SMQTT	Publish/subscribe	TCP		CP/KP-ABE	Key revocation and group pub/sub for distributed SMQTT are still a challenge
MQTT-SN	Publish/subscribe	UDP		TLS/SSL	SSL/TLS suffers from attacks like BEAST, CRIME, RC4, and Heartbleed
XMPP	Request/response Publish/subscribe	TCP	X	TLS/SSL	Susceptible to spamming attacks and lacks worldwide support
AMQP	Publish/subscribe	TCP		TLS/SSL	Inappropriate for real-time applications
DDS	Publish/subscribe	TCP/UDP		TLS/SSL	Memory intensive and no open-source libraries for constrained devices
REST	Request/response	HTTP	X	HTTPS	Hard to implement. Uses HTTP, so incompatible with constrained apps
WebSockets	Publish/subscribe Client/server	TCP	X	TLS/SSL	Bad user experience for IoT apps since it runs the client/server architecture
STOMP	Client/server	TCP		HTTP	The broker can act as a bottleneck

logic. It transmits packets with reduced quality to allowable levels if there is congestion, enabling it to transmit lower-quality packets/frames without being dropped. The number of packets delivered per second thus rises, and while this enhances video streaming apps, it is more appropriately made for those that quickly transmit non-real-time video such as in video surveillance. SUIT can stream JPEG pictures other than the traditional predictive video coding methods that are not suitable for sensor devices because of their constraints of energy, memory, and computational speed. Progressive JPEG-PJPEG performs better via packet transmission rate and delay though it degrades the image quality because of congestion [65]. This protocol employs cross-layer methods for interlayer data exchange, though it still employs some layering, making it still possible to preserve all advantages of the layered design since it has application and transport layers but has no MAC or routing layer.

For congestion detection and control, SUIT bases on the percentage number of in-bound and out-bound frames per window, number of contenders, and buffer occupancy of the next-hop node. Since buffer management has an effect on effectiveness and QoS in WMSNs, it is useful in the transport layer, so the protocol prioritizes packets for better QoS. SUIT reorders packets by tracking the sequence numbers of the delivered packets since they all have a source ID, frame number, and sequence number that it uses to detect any missing packets. The protocol however does not provide packet reliability [65].

Advantages:

(i) The protocol prioritizes packets for better QoS.
(ii) It can transmit lower-quality packets/frames without being dropped.

Disadvantages:

(i) It does not provide packet reliability.

4.3. Network Layer QoS Support Enforced by a Cross-Layer Controller (NLQS). This scheme permits packet-level service differentiation as a function of throughput, end-to-end packet error rate, and delay [66] (see Figure 8). This enhances network layer QoS. It comprises a cross-layer control unit (XLCU) for configuring and controlling networking functionalities at physical, MAC, and network layers [18]. This depends upon the unified logic which affects choices for application layer needs and status of functional blocks which do implementation of networking functions [66].

Advantages:

(i) Cross-layer interactions can be controlled without weakening the upgradability, modularity, and simplicity of designing the system.

4.4. Cross-Layer Signaling Shortcuts (CLASS). CLASS has a high level of effectiveness, flexibility, and comprehensiveness. According to [67], it has the following features:

(i) Direct signaling among remote layers leading to faster transmission.
(ii) Lightweighted internal message format: standardized protocols are not suitable for use in internal signaling since they are not usually lightweight, for instance, transmitting against faults in the network. CLASS requires only three fields, namely, *destination address*, *event type*, and *event contents*.
(iii) The setup of its exterior messages is standard: considering external signaling, ICMP and TCP/IP are suitable for general messages and short notifications, respectively.
(iv) Message control protocol should ensure optimized and organized exchange of dense simultaneous messages across layers for high efficiency and avoidance of possible conflicts. The protocol applies to many cross-layer signaling situations.

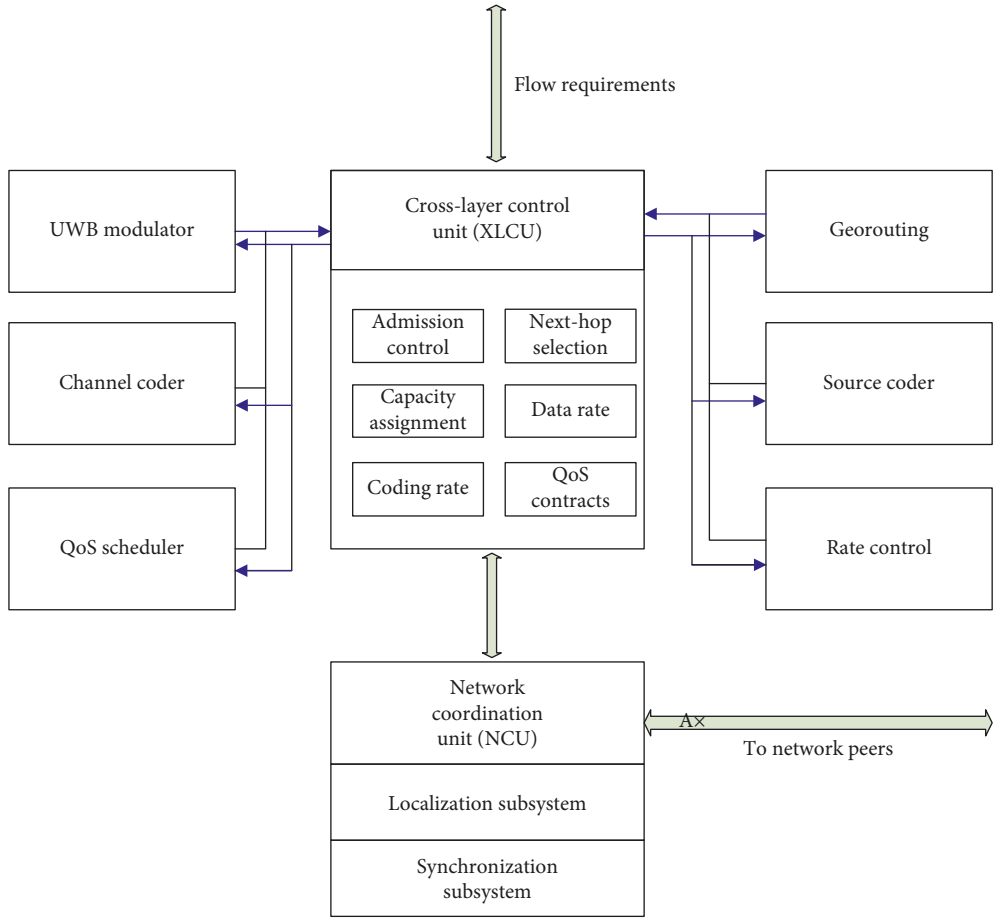

Figure 8: Cross-layer controller architecture.

Advantages:

(i) The protocol is scalable and has a very low propagation delay.
(ii) It is very efficient and flexible since it uses direct signaling between any two layers.
(iii) It is very comprehensive and effective.

4.5. Cross-Layer and Multipath-Based Video Transmission (CMVT). CMVT is a hybrid of application and network layers. At the application layer, it carries out the encoding of video streams to video data frames (I-, P-, and B-frames) by means of the MPEG-4 encoding format. At the network layer, route discovery and data transfer take place. Two algorithms, greedy forwarding and rollback, are used to discover a number of paths from the source to the sink node via the route discovery method [20]. A specified node i calculates the evaluation of its adjacent node j using the following equation:

$$f_{ij} = (1-\alpha)\frac{d^2(j,D) - d^2_{\min}(i)}{d^2_{\max}(i) - d^2_{\min}(i)} + \alpha\frac{e_{\text{init}}(j) - e_{\text{res}}(j)}{e_{\text{init}}(j)}, \tag{19}$$

where f_{ij} = evaluation value for the node i to j; $d^2(j,D)$ is the distance from the node j to the destination node D; $d^2_{\min}(i)$ and $d^2_{\max}(i)$ are the minimum and maximum distances of neighbors of the node i to D, respectively; $e_{\text{init}}(j)$ is the initial energy of the node j; $e_{\text{res}}(j)$ is the current residual energy of the node j; and α is the energy coefficient given as follows:

$$\alpha = \frac{e_{\max}(i) - e_{\min}(i)}{e_{\max}(i)}, \tag{20}$$

where $e_{\max}(i)$ and $e_{\min}(i)$ are, respectively, the maximum and minimum energy left of all node i's neighbors.

The network layer further transmits video streams with CMVT doing status evaluation choosing an appropriate communication path for any packets, and the QoS guarantee level for the path i is calculated as

$$f_i = (1-\omega)\frac{h_i}{\sum h_i} + \omega\frac{n_i}{\sum n_i}, \tag{21}$$

where f_i is the evaluation value for the path i, h_i are the hops for the path i, n_i is the summation of packets sent via i, $\sum n_i$ is the summation of packets sent by sources, and ω is the energy consumption factor [18].

Advantages:

(i) The CMVT protocol is very superior in media transmission, particularly large-scale WMSNs.

4.6. Cross-Layer Cooperative MAC (CoopMAC). CoopMAC [68] is based on the IEEE 802.11 DCF approach. There is a reduction in interference of neighboring cells leading to uniform coverage in densely deployed networks [68]. A connection is established by two-way authentication between the sender and the receiver. When a Clear-to-Send (CTS) packet and short interframe space (SIFS) are obtained by the sender, packets are transmitted right to the receiver with no valuable cooperation. If available, the two nodes find out whether they can exchange data. This is done by the use of a helper identification signal to establish how feasible cooperative communication is. It is established in the presence of a signal, and if it is not available, direct communication is activated. For distributed systems, helper-initiated cooperation is preferable because of the presence of the RTS/CTS packets [69].

Advantages:

(i) CoopMAC has an advantage over others due to the spatial diversity amongst the three nodes [70].
(ii) It has the potential to achieve significant throughput and delay with no complexity in the design of the system.

4.7. Cooperative MAC Protocol for Multihop Networks (M-CMAC). Just as CoopMAC, M-CMAC is made in such a way that high data rate stations help low data rate stations to forward traffic for broadcasting [71]. There are also helpers chosen in such a way that 2 fast-hop transmissions replace a slow hop transmission. The helper via whom there exists least delay from the sender to the receiver is considered best because of the minimum two-hop transmission rate and so is the chosen neighbor to the duo. Here, the working assumption is that all nodes have their position coordinates known, and thus, the Euclidean distances between all pairs can be calculated and converted to the data rate for that link. All nodes have cooperative tables (CT) of potential helpers with destination and helper MAC address, Euclidean distance, and total distance through the helper. Any sender with data to transmit checks out for the existence of any helper in the CT for the destination and if available forwards RTS for channel reservation for single-hop duration. M-CMAC has got an RTS format having a shape with five fields similar to the one given as follows:

Frame control	Duration	Source address	Destination address	Helper address

The destination node's address is kept by the source in the part for the helper address that is in turn saved as the destination address. Nodes inspect helper and destination address fields on receipt of RTS, whereby the node acts as the other node's helper in case the helper address field is not the same and the helper transmits CTS back to the source if at all it wants to forward data. On delivery of the CTS packet, the source transmits to the helper which in turn sends the packets to the destination that also sends ACK to the helper on receipt of the packet. In M-CMAC, there exists a higher level of channel reuse (parallel transmissions) due to the increased number of nodes therein that subsequently raises availability of helpers for data forwarding. This results in increased throughput in comparison with CoopMAC and IEEE 802.11 DCF [71].

Advantages:

(i) It promises a higher throughput compared to CoopMAC and IEEE 802.11 DCF.

4.8. Cluster-Based Cooperative Routing (CBCR) Protocol. The CBCR protocol has a multihop data-forwarding function realized at the link layer with cooperative links that use M-CMAC. This protocol encompasses two stages, namely, the routing relay selection phase and data forwarding phase.

(i) Routing relay selection phase: all nodes announce their presence to their neighbors by broadcasting periodical beacon messages that carry the MAC address of the node. Each of these constructs a relay table that contains all neighboring nodes with which it is able to communicate. The node further broadcasts its neighbor list in case of any change to its entries since the previous broadcast. The MAC addresses of nodes next to the node X are contained in its relay table's column one, and the neighbor node's row has MAC addresses for neighbors of the adjacent node. Based on its relay table, each node independently chooses routing relays. Choosing a node as a relay node depends on the number of nodes it connects—it should be the highest number [71].
(ii) Data forwarding phase: a node with packets to transmit needs to first verify if the receiver is in the same cluster and if that receiver has a helper to which the packet is thence forwarded but in whose absence the packet is delivered right to the receiver. In case the intended receiver is in a different cluster, the relay table will be checked to see whether that receiver is reachable via other routing relays. If it is reachable, packets will be sent to the routing relay directly or via the helper if at all the relay has one. In case of destination unreachable via relays, packets are multicast to all relays by the node [71].

Advantages:

(i) It enables multicasting.

4.9. MAC-PHY Cross-Layer Protocol. A cooperative cross-layer standard for cooperation at the physical layer in next-generation WMSNs is developed in [70]. It provides a complete MAC layer algorithm that gives a supportive shell for the PHY-MAC layer. Like CoopMAC, its scheme depends on the middle node for communication between nodes. The MAC layer protocol has been changed to regulate information exchanges at the physical layer. The receiver node receives duplicate packets from source and helper nodes to decode the data [70].

4.10. MAC-Centric Approach. This cross-layer protocol targets multimedia applications by using the MPEG-4

scheme [21]. It is characterized by 4 access categories (AC3–AC0) according to their priority in transmission. It is meant for supporting different QoS needs in upcoming video applications and enabling MAC layer differentiation for H.264 partitioning [18]. Because of low bandwidth, delay, and other QoS challenges that cause inefficiency in transmitting multimedia data in WMSNs, a number of algorithms depending on IEEE 802.11e have been proposed to support transmission of quality videos [72]. Selecting an AC depends on the measures for QoS such as the loss rate and delay. Thus, AC3 having the highest priority is mapped to the parameter set concept since the stream is sensitive to bit loss especially in video transmission [21].

4.11. Adaptive Cross-Layer Forward Error Correction (ACFEC). In the ACFEC [73] model, data packets are exchanged among nodes through the access point (AP) which operates the infrastructure mode in which it adds FEC to video data [72]. These data are dealt by encapsulation using a streaming server to the receiver through the wireless AP as RTP packets. An adaptive FEC controller senses the category of packets out of the RTP header and recovers the header of packets from UDP. The encoder generates some error-correcting packets whose number is determined by the source packet number of the block. Multimedia transmissions are monitored by the controller using MAC failure data, and its counter is increased by one if there is a failed transmission. The controller uses the failure counter to change the packet number produced after transmitting a block [18]. In case there are lost packets, it changes the redundancy rates and produces extra packets to replace the lost ones and satisfy the requirements of the receiving node [72]. The FEC packet number is enhanced or reduced to satisfy the receiver's requirements and stop packet losses. This is done through accurate detection of packet losses and adjustment of redundancy rates. If all video data packets are well received, there will not be generation of FEC packets [73].

Advantages:

(i) It promises a good QoS through packet loss reduction and redundancy rate adjustment.

4.12. Balanced Cross-Layer Fuzzy Logic (BCFL) Design Routing Algorithm in WSNs. A new fuzzy logic-based routing algorithm (BCFL) [74] was designed using dispersion of the cross-layer parameter as the fuzzy logic inference system input. Each cross-layer parameter has a dynamic weight depending on the value of dispersion. The design comes with some innovations as follows: (i) for fuzzy logic inference system input, the absolute parameter value is substituted by parameter dispersion, thereby significantly reducing algorithmic complexity; (ii) dispersion does not change with the order of magnitude according to the dispersion formula; and (iii) the weight of the parameter depends on the size of its dispersion, and the two are inversely proportional to each other. This enables BCFL to have some nobility unlike other algorithms as per the following distinguishing properties making it *advantageous*:

(i) It has got simple if-then rules which remain constant even when the constraints increase.

(ii) It is capable of dealing with many constraints with no increment in complexity.

(iii) It has capacity to yield a more balanced solution than other algorithms.

(iv) It is easily adaptable to changes in network conditions and topology even when the changes are frequent like in underwater WSNs. The algorithm is useable in choosing the CH in cluster-based routing protocols [74].

4.13. Minimum Hop Disjoint Multipath Routing Algorithm with Time Slice Load-Balancing Congestion Control Scheme. MHDMwTS is a two-phased routing protocol comprising path build-up and path acknowledgment phases. It consists of multiple sources each of which has three build-up disjoint paths, namely, primary, alternate, and back-up paths. On activation, the source node requests to build up a route to the nearest hop neighbor, which is the path build-up phase [19]. In this phase, step one has source activation in which the requested node for path building adds its number and timestamp and sends to the least hop-count neighboring node. This goes on until the least time latency sink having the information needed to construct the primary route is reached [75]. In step two, there is path extraction when the new package from another path arrives whereby the extracted path is compared with the primary path. If the node is shared, the package will be rejected or an alternative path will be searched for to get a backup through comparison of the preceding two paths. Phase two is path acknowledgment where we have the third step in which the sink returns the ACK packet to the sender with path information having nodes and their time information after it has computed it using the timestamp [18,19].

Advantages:

(i) The protocol minimizes end-to-end latency.

(ii) It enhances congestion control.

4.14. Cross-Layer Optimal Design (CLOD). Authors in [76] come up with a CLOD for scheduling at the data link layer, routing at the network layer, and controlling congestion at the transport layer with an assumption of fixed link capacity. Through congestion control, energy efficiency is improved. Transport layer congestion at the nodes is minimized by compressed sensing (CS) in which transmitted bits are reduced, whereas optimally allocating resources decreases congestion on the links at the data link layer. CLOD promises minimized computational complexity and better performance in light traffic scenarios [77]. Generating and storing CLOD is more efficient compared to Gaussian random matrices. It prolongs network lifetime and saves energy.

Advantages:

(i) It does congestion control subsequently increasing energy efficiency.

Table 4: A summary of some of the reviewed cross-layer models.

Protocol	Layers	Aims	Comments	QoS parameters
SDRCS	MAC/PHY	Routing real-time traffic	Transmits multimedia traffic via cross-layer packet forwarding	Throughput, latency
SUIT	Application/ transport	Transmission of non-real-time video	Uses fuzzy logic to regulate congestion	Transmission rate, delay
CLASS	Any two	Design serves as a framework for different implementations of different application scenarios	Scalable, very efficient, flexible, and has very low propagation delay	Propagation delay, jitter
NLQS	PHY/MAC/ network	Permits packet-level service differentiation as a function of throughput, packet error rate, and delay	Network layer QoS is enhanced	Throughput, delay, packet error rate
CMVT	Application/ network	Encodes video streams to video data frames via MPEG-4 encoding and does route discovery and data transfer	Greedy forwarding and rollback are used to find source-to-sink paths	Energy
CoopMAC	PHY/MAC	Offers spatial diversity among the three nodes	Helper ID signal finds out how feasible cooperative communication is	Throughput, rate, delay
M-CMAC	PHY/MAC	Increases end-to-end throughput and packet delivery ratio	Euclidean distances between nodes are calculated and converted to the data rate for a link	Throughput, packet delivery ratio
CBCR	PHY/MAC	Minimizes control overhead and time consumed in establishing the cooperative paths than M-CMAC	Energy consumption is more uniformly distributed in a network enhancing network lifetime	Throughput, packet delivery ratio, energy
MAC-PHY	MAC/PHY	Meant to enable PHY layer cooperation and maximize gains of cooperation at the MAC layer	Leverages both spatial diversity and coding gain	Throughput, delay
MAC-centric	MAC/APPL	Meant to support QoS needs in new video apps and enable MAC layer differentiation for H.264 partitioning	Targets multimedia applications by using the MPEG-4 scheme	Delay, packet loss rate
ACFEC	MAC/network (UDP)	Meant to enhance the quality of video streaming over 802.11 WLANs and overcome packet losses	Adjusts redundancy rates to overcome channel fluctuations and detect and reduce packet loss	Packet loss rate
MHDMwTS	—	Meant to provide reliable data transfer with the multipath routing and load-balancing congestion control method in WMSNs	More reliable than basic routing schemes for transport multimedia data	Latency, package transmit rate
BCFL	—	Introduces dispersion into fuzzy logic-based routing and sets every cross-layer parameter with a dynamic weight	Can be used to select a CH in cluster-based routing protocols and proposes a dispersion formula	Node utility, dispersion
CLOD	Datalink/ network/ transport	Prolongs network lifetime and achieves congestion control and designed for lightly loaded WSNs	Assumes fixed link capacity and integrates compressed sensing technology	Throughput, average energy, CS error ratio

(ii) It reduces computational complexity.
(iii) It improves performance in light traffic scenarios.

4.15. Challenges and Opportunities of the Cross-Layer Design (CLD)

(i) Different layers in a real network assume different functions and/or services. A layer only communicates with its neighboring layers. The layered model degrades system performance because of many features in wireless transmission. QoS parameters in the layers of WMSNs are shared by routing protocols to optimize the performance. Nevertheless, there is a need to design cross-layer models for increased efficiency in routing [1].

(ii) The physical layer plays a highly vital role in CLD. Rate adaptation and channel allocation take place at the physical layer via signal processing to enhance QoS. End-to-end performance in wireless media is affected by the changes therein given the effect on

the purposefulness of the protocols at the network layer. CLD offers solutions to conserving power, minimizing energy, and controlling both the flow and congestion in the network, as well as fault tolerance, so it is an opportunity for designers considering other layers. We further desire to develop CLDs that shall make an impact on network operation and get close attention [18].

(iii) CLD interface standardization: the architecture must deliver module functionality although there exist some queries amongst modules about possible interfaces predicted by the necessity to share information among remote protocol layers. Technical challenges include developing, designing, and standardizing of cross-layer interfaces and algorithms that satisfy the cross-layer optimization requirements among protocol layers.

Table 4 summarizes these cross-layer models with a few comments.

5. Future Research Direction

5.1. At the MAC Layer. In WMSNs, achieving duty cycling is tricky since video traffic is volatile. Listening to adjacent nodes wastes energy. We need to switch states from awake to sleep. This requires further research.

Self-organizing networks and green communication: although most CRWSN energy-saving algorithms are designed to reduce node transmission power, energy consumption is inevitable during the operational mode of BSs because of internal processing. Scholars are trying with policy-makers to encourage a shift to green WSNs to reduce operational costs and carbon footprints. With self-organizing CRWSNs for automatic switch-off when idle, energy is saved, which calls for cognitive switching algorithms. So integrating cognitive switching with radio resource allocation in CRWSNs is a promising research area [78].

5.2. At the Application Layer. Considering a WMSN architecture, end users access the network via the BS to and from which they send their sensed data. We desire to get guaranteed QoS and energy efficiency by designing a new architecture in the form of hardware and software which needs to be investigated further for a fix of the same [1].

QoS and energy efficiency are very crucial in WMSNs and mostly real-time apps that require guaranteed bandwidth and throughput in their network lifetime. Most protocols ignore base station (BS) mobility and WMSN nodes. Traffic management, telemedicine, and battlefield surveillance apps require mobile nodes or BSs, so designing dynamic routing protocols to be adaptable in these circumstances is necessary.

5.3. Cross-Layer. Security of data transmitted on WMSNs, e.g., in military surveillance and e-commerce, is key with QoS and energy efficiency. WMSNs are vulnerable to attacks like worms and sinkhole. Moreover, their computation power is not that high complicating implementation of strong, secure protocols. This needs further attention.

For fairness and priority issues in CRWSNs, nodes are assigned diverse priorities depending on importance and urgency of their data. For better network performance, there is a need to develop fair resource allocation mechanisms with priority-based fairness in sensors. There is still potential in this research area [78].

We must develop multiobjective, adaptive protocols to optimize QoS metrics involved in routing for best transmission results to get a trade-off between different optimization metrics. We need to design standardized cross-layer algorithms which are able to meet cross-layer optimization standards.

6. Conclusion

We have reviewed the different QoS strategies for WSNs in the context of IoT from the MAC layer and application layer as well as the cross-layer paradigm. For the MAC layer, we have reviewed protocols for WSNs such as the hybrid of the contention-free and contention-based MAC protocols and those for WMSNs. We further reviewed a number of application layer protocols including many machine-to-machine request/response and publish/subscribe protocols. For system optimization, cross-layer QoS strategies are very important in wireless communication. We have reviewed a number of cross-layer strategies. For all categories reviewed, challenges and opportunities are discussed. Finally, some possible future directions are discussed for research and application showing a promising potential for future research in this relatively new area especially as more WMSN applications emerge recently.

Conflicts of Interest

The authors hereby declare no conflicts of interest regarding the publication of this work.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (Nos. 61602398, 61672447, and 61711540306).

References

[1] V. Bhandary, A. Malik, and S. Kumar, "Routing in wireless multimedia sensor networks : a survey of existing protocols and open research issues," *Journal of Engineering*, vol. 2016, Article ID 9608757, 27 pages, 2016.

[2] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "Wireless multimedia sensor networks: applications and testbeds," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, 2008.

[3] M. Iqbal, M. Naeem, A. Ahmed, M. Awais, A. Anpalagan, and A. Ahmad, "Swarm intelligence based resource management for cooperative cognitive radio network in smart hospitals," *Wireless Personal Communications*, vol. 98, no. 1, pp. 571–592, 2018.

[4] S. A. Alvi, B. Afzal, G. A. Shah, L. Atzori, and W. Mahmood, "Internet of multimedia things: vision and challenges," *Ad Hoc Networks*, vol. 33, pp. 87–111, 2015.

[5] Q. F. Hassan, A. U. R. Khan, and S. A. Madani, *Internet of Things: Challenges, Advances, and Applications*, CRC Press Taylor & Francis Group: A Chapman & Hall Group, Boca Raton, FL, USA, 2018.

[6] Q. Wang, Y. Zhao, W. Wang et al., "Multimedia IoT systems and applications," in *Proceedings of the 2017 Global Internet of Things Summit (GIoTS)*, vol. 2, Geneva, Switzerland, June 2017.

[7] A. Aslam and E. Curry, "Towards a generalized approach for deep neural network based event processing for the internet of multimedia things," *IEEE Access*, vol. 6, pp. 25573–25587, 2018.

[8] M. Jridi, T. Chapel, V. Dorez, L. Bougeant, and A. Le Botlan, "SoC-based edge computing gateway in the context of the internet of multimedia things : experimental platform," *Journal of Low Power Electronics and Applications*, vol. 8, no. 1, p. 1, 2018.

[9] H. Noura, A. Chehab, L. Sleem, and M. N. Rapha, "One round cipher algorithm for multimedia IoT devices," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 18383–18413, 2018.

[10] T. Balan, D. Robu, and F. Sandu, "Multihoming for mobile internet of multimedia things," *Mobile Information Systems*, vol. 2017, Article ID 6965028, 16 pages, 2017.

[11] H. Shen and G. Bai, "Routing in wireless multimedia sensor networks: a survey and challenges ahead," *Journal of Network and Computer Applications*, vol. 71, pp. 30–49, 2016.

[12] A. Ahmad, M. H. Rehmani, H. Tembine, O. A. Mohammed, and A. Jamalipour, "IEEE access special section Editorial : optimization for emerging wireless Networks : IoT, 5G, and smart grid communication networks," *IEEE Access*, vol. 5, pp. 2096–2100, 2017.

[13] B. A. Muzakkari, M. A. Mohamed, M. F. A. Kadir, and Z. Mohamad, "Recent advances in energy efficient-QoS aware MAC protocols for wireless sensor networks," *International Journal of Advanced Computer Research*, vol. 8, no. 38, pp. 212–228, 2018.

[14] M. A. Yigitel, O. D. Incel, and C. Ersoy, "QoS-aware MAC protocols for wireless sensor networks: a survey," *Computer Networks*, vol. 55, no. 8, pp. 1982–2004, 2011.

[15] A. M. Abbas and O. Kure, "Quality of Service in mobile ad hoc networks: a survey," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 6, no. 2, p. 75, 2010.

[16] M. Q. Shatnawi, "Application layer protocols for the internet of things : a survey," in *Proceedings of the International Conference on Engineering & MIS (ICEMIS)*, Agadir, Morocco, September 2016.

[17] V. Karagiannis, P. Chatzimisios, F. Vazquez-gallego, and J. Alonso-zarate, "A survey on application layer protocols for the internet of things research motivation," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 3, pp. 1–10, 2015.

[18] M. S. Bernard, T. Pei, Z. Li, and K. Li, "QoS strategies for wireless multimedia sensor networks in the context of IoT," *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 4, pp. 228–253, 2019.

[19] A. AlAmri and M. Abdullah, "Cross-layer quality of service protocols for wireless multimedia sensor networks," in *Proceedings of the International Conference on Communication, Management and Information Technology*, pp. 649–658, Cosenza, Italy, April 2017.

[20] J. Guo, L. Sun, and R. Wang, "A cross-layer and multipath based video transmission scheme for wireless multimedia sensor networks," *Journal of Networks*, vol. 7, no. 9, pp. 1334–1340, 2012.

[21] A. Ksentini, A. Naimi, and M. Guéroui, "Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 107–114, 2006.

[22] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, New York, NY, USA, June 2002.

[23] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks categories and subject descriptors," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems, SenSys'04*, pp. 95–107, Baltimore, MA, USA, November 2004.

[24] T. Danmanee, K. N. Nakorn, and K. Rojviboonchai, "CU-MAC: a duty-cycle MAC protocol for internet of things in wireless sensor networks," *ECTI Transactions on Electrical Engineering, Electronics, and Communications*, vol. 16, no. 2, pp. 30–43, 2018.

[25] T. Van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, Los Angeles, CA, USA, November 2003.

[26] P. Xie and J. Cui, "R-MAC : an energy-efficient MAC protocol for underwater sensor networks," in *International Conference on Wireless Algorithms, Systems and Applications (WASA 2007)*, Chicago, IL, USA, August 2007.

[27] A. Rahim, N. Javaid, M. Aslam, U. Qasim, and Z. A. Khan, "Adaptive-reliable medium access control protocol for wireless body area networks," in *Proceedings of the Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, Seoul, Korea, June 2012.

[28] H. S. Kwak, S. Ullah, D. H. Kwak, and C. H. Lee, "A power efficient MAC protocol for wireless body area networks," *Journal of the Korean Institute of Intelligent Systems*, vol. 8, no. 6, pp. 131–140, 2009.

[29] C. Omeni, A. J. Burdett, and M. Park, "Energy efficient medium access protocol for wireless medical body area sensor networks," in *Proceedings of the 4th IEEE/EMBS International Summer School and Symposium on Medical Devices and Biosensors*, pp. 29–32, Cambridge, UK, August 2007.

[30] L. Sitanayah, C. J. Sreenan, and K. N. Brown, "ER-MAC: a hybrid MAC protocol for emergency response wireless sensor networks," in *Proceedings of the Fourth International Conference on Sensor Technologies and Applications*, pp. 244–249, Mestre, Italy, July 2010.

[31] Y. Sun and D. B. Johnson, "RI-MAC : a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems—SenSys'08*, pp. 1–14, Raleigh, NC, USA, November 2008.

[32] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC : a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, SenSys 2006*, pp. 307–320, Boulder, Colorado, USA, October 2006.

[33] S. Boulfekhar and M. Benmohammed, "Synchronous receiver initiated MAC protocol for long-lived sensor networks," *Computers & Electrical Engineering*, vol. 40, no. 2, pp. 504–516, 2014.

[34] S. Mehta and K. S. Kwak, "H-MAC: a hybrid Mac protocol for wireless sensor networks," *International Journal of Computer Networks & Communications*, vol. 2, no. 2, pp. 108–117, 2010.

[35] I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu, "Z-MAC: a hybrid MAC for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 511–524, 2008.

[36] S. Arshad, A. Al-sadi, and A. Barnawi, "Z-MAC: performance evaluation and enhancements," *Procedia Computer Science*, vol. 21, pp. 485–490, 2013.

[37] Y. Liu, I. Elhanany, and H. Qi, "An energy-efficient QoS-aware media access control protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Mobile ad hoc and Sensor Systems*, pp. 9–11, Washington, DC, USA, December 2005.

[38] J. Decotignie, "WiseMAC : an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Proceedings of the Ninth International Symposium on Computers And Communications*, pp. 244–251, Alexandria, Egypt, June 2004.

[39] R. A. Rashid, "Development of energy aware TDMA-based MAC protocol for wireless sensor network system," *European Journal of Scientific Research*, vol. 30, no. 4, pp. 571–578, 2009.

[40] H. P. Van Hoesel, "A lightweight medium access protocol (LMAC) for wireless sensor networks," in *Proceedings of the International Workshop on Networked Sensing Systems*, Baltimore, MD, USA, November 2004.

[41] C. Li, P. Wang, H. Chen, and M. Guizani, "A cluster based on-demand multi-channel MAC protocol for wireless multimedia sensor networks," in *Proceedings of the IEEE International Conference on Communications*, pp. 2371–2376, Beijing, China, May 2008.

[42] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-service in ad hoc carrier sense multiple access wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1353–1368, 1999.

[43] J. P. John, "Providing MAC QoS for multimedia traffic in 802.11e based multi-hop ad hoc wireless networks," *Computer Networks*, vol. 51, no. 1, pp. 153–176, 2007.

[44] S. Lofty and N. Padmavati, "A survey on mobility based protocols in WSNs," in *Proceedings of the International Conference Communication and Manufacturing*, pp. 12–15, Hangzhou, China, July 2014.

[45] S. Deng, L. Shen, and J. Li, "Mobility-based clustering protocol for wireless sensor networks with mobile nodes," *IET Wireless Sensor Systems*, vol. 1, no. 1, pp. 39–47, 2011.

[46] M. Khalil, A. Khalid, F. U. Khan, and A. Shabbir, "A review of routing protocol selection for wireless sensor networks in smart cities," in *Proceedings of the 24th Asia-Pacific Conference on Communications (APCC)*, Ningbo, China, November 2014.

[47] O. Message, Q. Telemetry, and T. Mqtt, Committee Specification Draft 02, April 2014.

[48] M. Collina, M. Bartolucci, A. Vanelli-coralli, and G. E. Corazza, "Internet of things application layer protocol analysis over error and delay prone links," in *Proceedings of the 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop*, Livorno, Italy, September 2014.

[49] A. Talaminos-barroso, M. A. Estudillo-valderrama, L. M. Roa, J. Reina-tosina, and F. Ortega-ruiz, "A machine-to-machine protocol benchmark for e health applications—use case: respiratory rehabilitation," *Computer Methods and Programs in Biomedicine*, vol. 129, pp. 1–11, 2016.

[50] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, Alexandria, VA, USA, October 2006.

[51] M. Singh, M. A. Rajan, V. L. Shivraj, and P. Balamuralidhar, "Secure MQTT for internet of things (IoT)," in *Proceedings of the Fifth International Conference on Communication Systems and Network Technologies*, Gwalior, India, April 2015.

[52] A. Stanford-Clark and H. L. Truong, *MQTT for Sensor Networks (MQTT-SN) Protocol Specification*, http://mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf, 2013.

[53] RFC, "Extensible messaging and presence protocol (XMPP)," in *Proceedings of the IETF*, pp. 1–9, San Diego, CA, USA, August 2005.

[54] S. Vinoski and I. Technologies, "Queuing protocol," *IEEE Internet Computing*, vol. 10, no. 6, pp. 87–89, 2006.

[55] J. L. Fernandes, I. C. Lopes, J. J. P. C. Rodrigues, and S. Ullah, "Performance evaluation of restful web services and AMQP protocol," in *Proceedings of the Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 2–7, Da Nang, Vietnam, July 2013.

[56] M. Asim, "A survey on application layer protocols for internet of things (IoT)," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 3, pp. 996–1000, 2017.

[57] K. An, A. Gokhale, D. Schmidt, S. Tambe, P. Pazandak, and G. Pardo-castellote, "Content-based filtering discovery protocol (CFDP): scalable and efficient OMG DDS discovery protocol," in *Proceedings of the 8th ACM International Conference on Distributed Event-Based Systems*, New York, NY, USA, July 2014.

[58] E. Web, What is a web service? pp. 52–57, 2010.

[59] M. Laine, *RESTful Web Services for the Internet of Things*, pp. 2–4, 2012, http://mediaTkk.

[60] IETF, The WebSocket protocol, pp. 1–71, 2011.

[61] V. Pimentel, "Communicating and displaying real-time data with websocket," *IEEE Internet Computing*, vol. 16, no. 4, pp. 45–53, 2012.

[62] Wikipedia, 2019, https://en.wikipedia.org/wiki/Streaming_Text_Oriented_Messaging_Protocol.2019.

[63] K. Raghavendra and S. Nireshwalya, "Application layer security issues and its solutions," *IJCSET*, vol. 2, no. 6, pp. 1266–1269, 2012.

[64] Y. Xue, B. Ramamurthy, and M. C. Vuran, "SDRCS : a service-differentiated real-time communication scheme for event sensing in wireless sensor networks," *Computer Networks*, vol. 55, no. 15, pp. 3287–3302, 2011.

[65] C. Sonmez, S. Isik, M. Y. Donmez, O. D. Incel, and C. Ersoy, "SUIT: a cross layer image transport protocol with fuzzy logic based congestion control for wireless multimedia sensor networks," in *Proceedings of the 5th International Conference on New Technologies, Mobility and Security (NTMS)*, pp. 1–6, IEEE, Istanbul, Turkey, May 2012.

[66] T. Melodia and I. Akyildiz, "Cross-layer QoS-aware communication for ultra wide band wireless multimedia sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 5, pp. 653–663, 2010.

[67] Q. Wang and M. A. Abu-Rgheff, "Cross-layer signalling for next-generation wireless systems," in *Proceedings of the 2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003*, vol. 2, pp. 1084–1089, IEEE, New Orleans, LA, USA, March 2003.

[68] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. Panwar, "CoopMAC: a cooperative MAC for wireless LANs," *IEEE*

Journal on Selected Areas in Communications, vol. 25, no. 2, pp. 340–354, 2007.

[69] H. Shan, H. T. Cheng, and W. Zhuang, "Cross-layer cooperative MAC protocol in distributed wireless networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 8, pp. 2603–2615, 2011.

[70] F. Liu, T. Korakis, Z. Tao, and S. Panwar, "A MAC-PHY cross-layer protocol for wireless Ad-Hoc networks," in *Proceedings of the 2008 IEEE Wireless Communications and Networking Conference*, pp. 1792–1797, IEEE, Las Vegas, NV, USA, March 2008.

[71] L. Jacob and H. R. Shamna, "Efficient cooperative MAC and routing in wireless networks," *Transactions on Networks and Communications*, vol. 3, no. 5, 2015.

[72] S. Rao and K. Shama, "Cross layer protocols for multimedia transmission in wireless networks," *International Journal of Computer Science & Engineering Survey*, vol. 3, no. 3, pp. 15–28, 2012.

[73] L. Han, S. Park, S. Kang, and H. Peter, "An adaptive cross-layer FEC mechanism for video transmission over 802.11 WLANs," in *Proceedings of the 2009 International Conference on Internet*, pp. 209–215, Las Vegas NV, USA, December 2009.

[74] N. Li, J. Martínez, and V. H. Díaz, "The balanced cross-layer design routing algorithm in wireless sensor networks using fuzzy logic," *Sensors*, vol. 15, no. 8, pp. 19541–19559, 2015.

[75] G. Sun, J. Qi, Z. Zang, and Q. Xu, "A reliable multipath routing algorithm with related congestion control scheme in wireless multimedia sensor networks," in *Proceedings of the 2011 3rd International Conference on Computer Research and Development*, vol. 4, pp. 229–233, IEEE, Shanghai, China, March 2011.

[76] M. Li, Y. Jing, and C. Li, "A robust and efficient cross-layer optimal design in wireless sensor networks," *Wireless Personal Communications*, vol. 72, no. 4, pp. 1889–1902, 2013.

[77] J. Yan, M. Zhou, and Z. Ding, "Recent advances in energy-efficient routing protocols for wireless sensor networks: a review," *IEEE Access*, vol. 4, pp. 5673–5686, 2016.

[78] A. Ahmad, S. Ahmad, M. H. Rehmani, and N. U. Hassan, "A survey on radio resource allocation in cognitive radio sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 888–917, 2015.