International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

A Systematic Review of Internet of Things Data Aggregation Methods

Mr. Ramakrishna Prasad A L
VTU Centre for PG Studies, Mysore, Karnataka, India

Dr. Shiva Murthy G
VTU Centre for PG Studies, Muddenahalli, Chikkaballapur, Karnataka, India

Abstract
The Internet of Things (IoT) connects physical objects via sensors, software, and other technologies so that they can communicate and exchange data with other devices via the internet. As devices, sensors, and other objects become more readily available, human life are going to be made simpler and easier than it has ever been. Many IoT-specific routing, power management and data aggregation protocols are developed. This paper summarizes current IoT data aggregation routing and outlines the design problems for data aggregation routing in the internet of things, followed by an examination of various aggregation routing strategies. The three categories of aggregation routing approaches are Network Architecture, Network Flow and Quality of Service (QoS). This paper investigates the planning tradeoffs between data aggregation and security overhead that exist in every routing paradigm.

Keywords
internet of things (IoT), Classification of data aggregation in IoT, Network Lifetime, Data Latency, Data Accuracy and Energy Consumption.

1. INTRODUCTION

The Internet of Things (IoT) is a network of interconnected items such as computers, mechanical and digital machinery, objects, animals, and other people with unique identification numbers (UIDs) that can communicate with one other without human or computer contact [11][12][13].We are much smarter than we used ten years ago to work out every technology object we use now. On the one hand, individuals were obliged to urge out of the sofa to change the channel, or switch the air conditioning on, and on the opposite hand to speak to their TV or air conditioner to instruct them what to do. While it's going to sound like something very interesting and beneficial, it's not without its disadvantages. IoT-enabled devices have the advantage of having the ability to monitor things remotely and send commands to them without being physically present. In leveraging the web of things concept, the device would be ready to be used from anywhere because this concept is also intended to minimize the limitations of using the devices at a distance. In contrast, it must suits certain standards as outlined below. This is often because all of those standards were developed especially for IoT to ensure a common working method.

2. IOT PROTOCOLS AND STANDARDS

The section that follows provides an overview of the standards, technologies, and protocols that enable things and environments to become IoT-enabled. IEEE.15 and IEEE.11 [1] are the foundations for Internet of Things protocols.

2.1 Standardizations/ Standards

Most IoT devices adhere to a variety of standard specifications, but IEEE and the most frequently used standards are those of the Internet Engineering Task Force (IETF)[2]. IEEE established the LR-WPAN (Low Speed Wireless Personal Area Networks) standard for low-power devices (IEEE 802.15.4)[3]. This standard is mainly used in IoT environments. There was an announcement in 2011 that IEEE 802.15.4 [5] would be released, and later In 2012, IEEE 802.15.4e was released. Later, time slots channel capabilities were added to MAC standards. Standardization was developed by the IETF in 2007 for resource-constrained devices, 6LOWPAN. The Internet Engineering Task Force (IETF) approved "Routing over Low Power Lossy Networks (ROLL)" as a routing protocol in 2008. "Constrained RESTful Environments (CORE)" was published in 2010, "DTLS in Constrained Environment (DICE)" was published in 2013, and "Authentication and Authorization in Constrained Environment (ACE)" was published in 2014 [6] [7].

2.1.1 IEEE 802.15.4

Many wireless and wired communications working groups are sponsored by IEEE. On a broad scale, companies are implementing the The 802.15.4e standard is used to monitor smart grids via smart utility networks, whereas 802.15.4f is used for active Radio Frequency Identification (RFID). The protocols and radio technology used in both versions, specifically 802.15.4a/b, are the same. For communications between nodes and conveyed data, 802.15.4 specifies star and peer-to-peer topologies, both of which must pass via a coordinator node or centre. This standard, which specifies the media access control (MAC) layer and the physical layer (PHY), the lowest layers of the OSI network model, is maintained by the IEEE 802.15 [4] group. A standard stack based on the 802.15 protocol is shown in Figure 1. In this case, IEEE 802.2 is the logical link control that communicates with the convergence sub-layers. LLC is the upper portion of DLL in the OSI model. A well-known LR-WPAN technological

standard is IEEE 802.15.4.

2.1.2 6LoWPAN

IPv6 over Low Power Personal Area Networks was also developed by the IETF (6LoWPANs). Because the number of IoT smart devices is constantly increasing, these devices require a large number of robust, scalable, and secure IP addresses. IPv6 is a very efficient and enabling technology when a large number of IP addresses are required. 6LoWPAN is a protocol that provides IP connectivity in resource-constrained network systems by transporting IPv6 packets via IEEE 802.15.4 links [4] [6]. The 6LoWPAN working group focuses on improving IPv6 protocols across the network utilizing 802.15.4. 6LoWPAN also aids in the implementation of IPv6 on the 802.15.4 MAC and Network levels.

This technique can also be used to replace expensive Wi-Fi. Figure 2 depicts the 6LoWPAN protocol stack for the TCP/IP paradigm. The Adaptation Layer divides and reassembles IPv6 packets between the Network and Data Link Layers. Because 6LoWPAN can also be used to make routing decisions, it is also known as the 6LoWPAN Border Router (6LBR)[16].

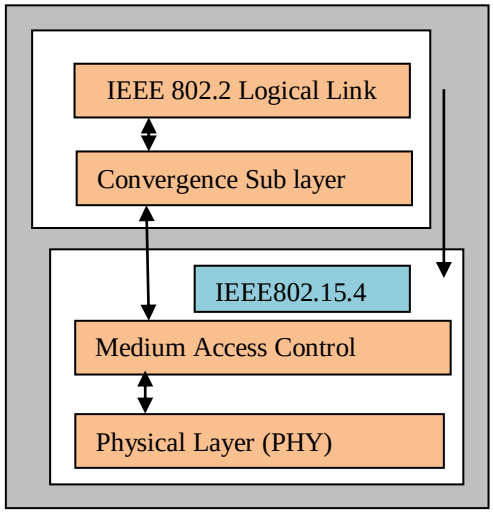

Figure 1: IEEE 802.15 Standard Stack

2.1.3 LoRAWAN

In the Internet of Things, the Low Power Wide Area Network (LPWAN) [3][17] technology was developed to connect a large number of devices. This technology is the polar opposite of short-range cellular networks, which include devices with vast communication ranges, low costs, and extended battery life. Long Range Wide Area Network (LoRAWAN) was developed by optimizing the LPWAN for low cost, low energy consumption, and a wide range and capacity. The LoRAWAN Alliance, a non-profit open association, created and maintains the network. In LoRAWAN networks, end devices transmit messages to gateways, which subsequently relay them to the server using the star topology.

2.2 Wireless Network Protocols

For starters, wireless Sensor Network connections improve the productivity of many smart domains such as smart home, smart city, and so on, but they also raise some concerns. Attackers can use wireless physical networks to target communicating data and the Internet of Things (IoT) to obtain important information. Stefan et al. [18] addressed some of the security concerns associated with various protocols used in the Smart home sector. According to the authors, in order to create a secure IoT environment, security analysis in all existing technologies must be performed in order to identify the root cause and validate the security system. In terms of security, the authors of [18] discussed certain IoT protocols used for smart home applications. The OSI model defines a standard architecture that includes all network communication layers. The TCP/IP model, which is a simplified version of the OSI network model, has four levels that are used to communicate across the Internet.

TCP /IP Model	6LoWPAN Stack	
Application layer	HTTP/CoAP/XMPP	
Transport layer	UDP	ICMPv6
	Adopted IPv6	
	6LoWPAN	
Data Link Layer	IEEE802.15.4 MAC	
Physical Layer	IEEE802.15.4 PHY	

Figure 2: 6LoWPAN protocol stack corresponding to TCP/IP Model

2.3 IoT Messaging Protocols

Instant Messaging Protocols (IM) is IoT messaging protocols that are mostly used for Internet chat communication. Protocols used in IoT applications include HTTP, MQTT, CoAP, XMPP, and AMQP. These protocols cover message management, lightweight message overhead, and compact messaging.Hyper Text Transfer Protocol (HTTP) has been a well-known communication protocol for many years. Its APIs are widely utilized in a variety of programming languages. This is one of the first Internet of Things protocols. In comparison to many other current protocols utilized in the IoT world, the authors in [25] there are countless footprints on it. Because it uses TCP and a three-way handshake process, it demands additional resources. It is unsuitable for running low-power embedded processes. It is unsuitable for running low-power embedded processes. This can only be accomplished by optimizing TCP. The client-server approach is used in this protocol. The request/response message format is used for communication. REST and HTTP are connected. It is based on a standard developed by the Internet Engineering Task Force. Update, create, read, and remove actions are performed via the GET, POST, PUT, and DELETE methods [26] [27].

2.3.1 Message Queuing Telemetry Transport (MQTT)

This is a popular lightweight protocol and it is uses a publish-subscribe mechanism to communicate. This protocol is created for devices of limited resources and network connections that have undesired features such as poor bandwidth and excessive latency [26]. This is a straightforward structured protocol with a better level of dependability. It uses less power and has a smaller preamble than other reliable messaging protocols. It is commonly recommended for IoT connectivity over other messaging protocols due to its simplicity and minimal message header. The MQTT For transition and implementation flexibility, the protocol has a publish/subscribe structure. It's a lightweight messaging system. MQTT is a message-centric protocol that was developed for M2M (mobile-to-mobile) communication and remote telemetry applications [8] [29].

2.3.2 Advanced Message Queuing Protocol (AMQP)

AMQP is the messaging protocol used by the session layer. It was intended to give nonproprietary ways for sharing massive amounts of data to industrial and corporate management. The two most common message delivery methods in AMQP is a point-to-point and store-and-forward protocol. OASIS [30] established the Advanced Message Queuing Protocol as an open standard protocol. MQTT and AMQP share many features. The message delivery techniques used by MQTT and AMQP are the same [31]. It has publisher/subscriber architecture and runs on the TCP platform.

The interoperability feature of AMQP is crucial because it enables message interchange between platforms written in various languages. It may thus be useful in diverse systems [26]. AMQP is a producer/consumer and broker entities must interoperate with the protocol paradigm, comparable to an email or instant messaging system. The data contained in AMQP messages is opaque, and the message processing is self-contained. AMQP may handle messages of any size [45]. AMQP is a message-exchange middleware protocol used in distributed systems.

2.3.3 Constrained Application Protocol (CoAP)

The CoAP is a synchronous request/response protocol. The Constrained RESTful Environment (CORE) and the Internet Engineering Task Force (IETF) developed it to provide a lightweight RESTful interface [19]. CoAP is used in a wide range of applications, from smart energy systems to environmental monitoring. CoAP is used in tiny devices with limited power, processing, and communication capabilities to enable RESTful interaction. CoAP is a web transfer protocol similar to HTTP that can be used to extend the architecture from REST to LoWPANs [8]. REST is a client-server protocol designed to allow low-power sensors to communicate. CoAP is a binary protocol built on the UDP architecture that replaces TCP, which is used in HTTP. The main rationale for using UDP for development is to avoid TCP overhead and hence reduce bandwidth requirements [34].

2.4 Importance of Data Aggregation in IoT

Data aggregation, as defined by current definitions, is the process of gathering and combining data from many sources. Aggregated data is often found in a data warehouse. It can deliver analytical responses and cut the time it takes to query massive data sets in half. Data evolves, expands, and becomes more complicated with each auctioned input and output in our technologically evolved society. Data is one of our time's most valuable currencies, but it's worthless without organization, classification, and comprehension. The extraction of insights that point to noteworthy trends, results, and provide a deeper knowledge of the information at hand is what makes data important. Data aggregation is the process of seeking, aggregating, and presenting data in a summarized, report-based style, helps businesses achieve specific business goals or do process/human analysis on a large scale.

Data aggregation is primarily used to save energy and minimize network bandwidth requirements. Using various IoT data aggregation methods, unnecessary data is removed. The quantity of data packages transferred is considerably reduced, which minimizes network traffic. IoT sensor nodes can also reduce redundancy in data received from neighboring nodes before delivering the final data packages.

2.5 Organization of the Paper

The remainder of the paper is laid out as follows. Different data aggregation routing methods in IoT and data aggregation classification in IoT will be examined in the next section. In Section 3, gives different data aggregation protocols on network architecture. Section 4, 5 and 6 gives the brief discussion on Data aggregation on network flow and quality of services. And also section 7 summarizes the open issues and challenges of data aggregation in IoT. Finally concludes the paper.

3. EFFICIENT DATA AGGREGATION ROUTING PROTOCOLS IN IOT

As shown in Figure 4, the performance of data aggregation techniques is heavily influenced by network design. We'll look at the most recent developments in IoT Data Aggregation in this part. Data aggregation in IoT is broken into three components in general: The Network claims this.

Service-Based Architecture Network Flow and Quality (QOS). Flat networks and hierarchical networks are the two types of networks based on Network Architecture. In a hierarchical network, however, nodes will have distinct tasks to play. The four parts of hierarchical data aggregation are tree, cluster, grid, and chain. The three types of Flat Networks are Push Diffusion, Two Phase Full Diffusion, and One Phase Full Diffusion. The sink sends a query message to the sensors, which receive

response messages from sensors with data that meets the query [17] [18] [19]. The sink node's battery power is depleted more quickly due to excessive communication and computation. Network functionality is disrupted when the sink node fails. The various data-aggregation protocols and their features are described in this paper. Network Flow contains Network Correlated and Network Lifetime Maximization. Finally, end-to-end reliability and congestion control, optimal information extraction, and consensus-based quality of service methodologies are all included in Quality of Service.

4. PROTOCOLS FOR DATA AGGREGATION BASED ON NETWORK ARCHITECTURE

The sensor network's architecture is crucial for the efficiency of various data aggregation protocols. In this part, we'll look at a number of data aggregation protocols that are tailored to distinct network designs. Flat networks and hierarchical networks are the two types of data aggregation based on network design. Flat wireless sensor networks have no hierarchical structure; therefore all sensors play an equal function. Every sensor node has the same purpose and is a peer in the Internet of Things. Data aggregation occurs exclusively at the sink node level in flat wireless sensor networks, which is a disadvantage. As a result, network latency can be extremely high. Furthermore, the entire network suffers if the sink node fails. Cluster, chain, tree, grid, and hierarchical data aggregation are the four parts of hierarchical data aggregation. Flat networks are divided into three types: push diffusion, two-phase full diffusion, and one-phase full diffusion.

4.1 Flat Networks

In a flat network, each sensor node performs the same function and has nearly the same battery capacity. Data aggregation is accomplished in such networks through data centric routing, in which the sink sends a query message to the sensors, for example, through flooding, and sensors with data matching the query send back answer messages to the sink. The communication protocol that is used depends on the application.

4.1.1 Push Diffusion

In a push diffusion scheme, the sources initiate the diffusion process, while the sinks react to the sources. When an Event is detected, the sources flood the data, while the sinks subscribe to the sources via enforcements. A push-based diffusion technique, sensor protocol for information via negotiation [38], is used. Negotiation and resource adaptation are two of SPIN's primary qualities. Sensor nodes require a descriptor to briefly explain their observed data in order to negotiate data. SPIN classifies these descriptions as metadata. Application-specific metadata formats exist. Sensors that cover many regions, for example, can use their unique ID as metadata in area coverage challenges. The data is promoted to the network's neighboring nodes via metadata by the initial node with fresh data. An neighboring node that is interested in this sort of data makes a data request to the starting node. The initiator node responds by sending data to the sinks. Each node has a resource management system that monitors its energy consumption. Before transmitting data, each node polls its resources, such as battery power. When the sensor's energy levels are low, it can perform fewer activities. In terms of data collection over time, simulation results show that SPIN works in a similar way to floods.

4.1.2 Two Phase Pull Diffusion

Directed diffusion is an energy-efficient data aggregation approach developed by Intanagonwiwat et al. [39]. Two phase pull diffusion is exemplified by directed diffusion. It is a sensor-based data-centric routing approach. In network messages, the data attributes are used. The propagation of interest in directed diffusion is depicted in Figure 3. If the properties of the data supplied by the source correspond to the qualities of interest, a gradient is put up to identify the data generated by the sensor nodes. At first, the sink sends an interest message to the network. The data rate and data transmission direction are both specified by the gradient.

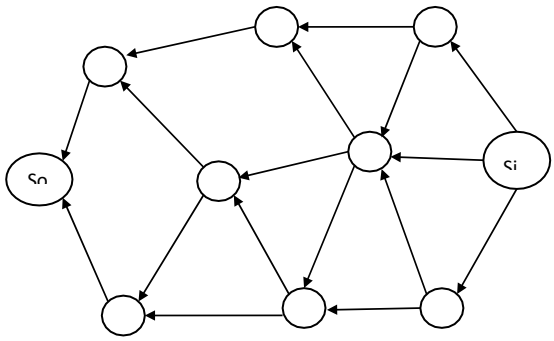

Figure 3: Interest propagation in directed diffusion.

Data can be cached and transformed by intermediate nodes. Each node has a data cache that holds data items that have been accessed lately. After receiving low data rate events, the sink strengthens a certain neighbor in order to attract higher quality data. Data-driven local rules are used to accomplish directed dissemination. The performance measures were average dissipated entries and average delays, which are the ratio of total energy dissipated per node to the number of different events observed by sinks. According to simulation tests, directed diffusion consumes significantly less energy than an omniscient multicast scheme in which each node sends data to all sinks via the shortest path multicast tree. Compared to omniscient multicast, directed diffusion consumes 60% less energy. Directed diffusion's average delay is comparable to that of omniscient multicast. For applications with a lot of sources but few sinks, directed diffusion is a good option. Directed diffusion, unlike SPIN, does not necessitate the maintenance of global network

topology. Directed diffusion, on the other hand, is ineffective for applications that demand constant data delivery to the sink.

4.1.3 One Phase Pull Diffusion

When there are a lot of sources and sinks, two-phase pull diffusion has a high overhead. Krishna Machari et al. [40] suggested a one-phase pull diffusion approach that avoids flooding caused by directed diffusion. Sinks send interest messages that flow through the network, establishing gradients, in one phase pull diffusion. On the other hand, the sources do not provide exploratory data. Only the sinks with the shortest latency gradient receive data from the sources. As a result, the fastest route (from source to sink) is the reverse route. Eliminating exploratory data transfer minimizes control overhead, allowing an efficient data aggregation tree to conserve energy.

4.2 Hierarchical Networks

A flat network might overburden the sink node with communication and processing, causing its battery to drain more quickly. When a sink node dies, the network's operation is disturbed. As a result, a slew of scalable and energy-efficient hierarchical data aggregation methods have emerged. In hierarchical data aggregation, data fusion at specific nodes minimizes the quantity of messages sent to the sink. This increases the network's energy efficiency. The remainder of this section discusses the various hierarchical data aggregation algorithms, as well as their principal benefits and drawbacks.

4.2.1 Tree-based Aggregation

In this scenario, data is aggregated by creating a data aggregation tree. Data aggregation occurs at intermediate nodes along the "tree" because sensor nodes are grouped in this fashion. The "root node" is given only a structured representation of the data that already exists. This aggregation method is appropriate for network-based applications that require data aggregation. The creation of an energy-efficient data aggregation tree that maximizes network longevity while minimizing transmissions is one of the key problems of tree-based aggregation. Tree-based approaches feature higher overhead, higher energy uniformity, and more strength, flexibility, and scalability when compared to cluster-based methods.

Sensor nodes are placed in a tree in a tree-based network, with data aggregation occurring at intermediate nodes along the tree and a concise representation of the data being transmitted to the root node. Tree-based data aggregation is useful for in-network data aggregation applications. In the case of radiation level monitoring in a nuclear power plant, for example, the maximum value provides the most significant information for the plant's safety. The building of an energy-efficient data aggregation tree is one of the most important aspects of tree-based networks.

Figure 4: Classification of Data Aggregation in IoT

EADAT-Energy Aware Distributed Aggregation Tree

In [41], the author proposed a distributed heuristic for creating and maintaining a data aggregation tree in sensor networks that is energy-aware (EADAT). The sink initiates the algorithm by transmitting a control message. The destination is the aggregate tree's root node. The sensor id, parent, residual power, status (leaf, non-leaf node, or undefined state), and hopcnt fields in the control message, respectively, specify the sensor id, parent, residual power, status (leaf, non-leaf node, or undefined state), and the number of hops from the sink. A sensor s sets its timer to Ts after receiving the control message for the first time. When the channel is not in use, Ts counts down. During this phase, the sensor selects the node with the largest residual power and the shortest path to the sink as its parent. The control message conveys this information to node S. When the timeout expires, Node s adds one hop to its hop count and sends the control message. If node t receives a message stating that node s is its parent node, it marks itself as a non leaf node. If it isn't otherwise, the node is identified as a leaf node.

PEDAP- Power Efficient Data gathering and Aggregation Protocol

This protocol assumes that the base station is already aware of all node locations. They are both centralized techniques in which the base station computes the routing information. This is because in systems where some parts are resource constrained but one or more elements are powerful, it is preferable to distribute the computation load to the system's more capable elements [42].

Prim's minimal spanning tree technique is used to calculate the routing information, with the base station serving as the root. The algorithm operates as follows: In our example, we began by adding a node to the tree that would serve as the base station. Following that, we pick the least weighted edge from a tree vertex to a non-tree vertex in each iteration and add it to the tree. This signifies that data from the newly added vertex will be routed through that edge in our example. This technique is continued until the tree's nodes have all been added. The resulting routing paths for a sample network are shown in Figure 4. The algorithm's running time complexity is O (n^2) assuming n nodes in the network. The expense of implementing the scheme on a regular basis is quite minimal when compared to LEACH and PEGASIS.

TAG -Tiny Aggregation

The sink causes the nodes to arrange into a routing tree by sending a message to every one of its neighbors. Every node receives a message and relays it to its progeny nodes. The sending node's level is included in the message, allowing the receiver to set its own level as the sender's plus one, its parent, and gain an ID before forwarding the message to its neighbors with the modified level [43].

TAG has two phases: distribution (where aggregate queries are

pushed down into the network) and collection (where individual requests are collected) (where aggregate values are constantly routed up from children to parents). Remember that our query semantics divide time into epochs of duration, and that if we don't group, we'll end up with a single aggregate value that incorporates the readings of all network devices during that epoch. Given our goal of using fewer messages, the collecting phase must ensure that parents in the routing tree wait until they hear from their children before propagating an aggregate value for the current epoch. We'll do this by having parents partition the epoch, requiring children to deliver their half state records over a time window chosen by the parent. This time interval was set to allow the parent to integrate partial state records and propagate its own record to its parent. Destination-Oriented Directed Acyclic Graph (DODAG) is a directed acyclic graph with one root and no outbound edges. Various multipath routing algorithms function by first selecting a set of paths for each node-destination pair and then spreading the flow along these paths. Some methods, such as equal cost multi-path (ECMP), make it simple to segregate these duties. ECMP divides demand from the source to the destination equitably across all equal-cost shortest paths. Other techniques, such as iterative gradient minimization algorithms [39], combine these two phases, and they can't be done independently. In circumstances where network topology changes are infrequent and energy is limited, such as wireless sensor networks for environmental monitoring, separating these two activities could result in improved overall performance, given the energy used for calculations and transmission.

PERLA: Power Efficient Routing with limited Latency

The current work builds on the concepts described in [41], but instead of using the IEEE 802.11 MAC layer, it uses IEEE 802.15.4 and handles some specific challenges relating to the adoption of standard synchronization among nodes. The method employs a spanning tree for ordinary routing operations and only uses alternative paths when a fault is discovered. Errors in a WSN are frequently caused by failures in links or nodes. Channel failures and collisions cause the former, whereas poor synchronization of the nodes sleep/listen schedules causes the latter. They are usually transient in nature, and the network layer does not explicitly handle them. Node failures, on the other hand, are permanent and might be caused by malfunctioning, battery depletion, or other external causes; they introduce dead routes, which the routing layer must identify in order to adjust the topology. Routing tables are not consistent with the real topology during the time it takes to discover node failures, and data is likely to be lost in part or all. Although latency may not be a major consideration for all sensing applications, it is preferable for the network to respond quickly to persistent failures that result in topology changes. Increased routing protocol responsiveness might solve this problem, but it could also lead to excessive fluctuations if frequent connection failures, which are typical in densely populated WSNs and interfering situations, are misread as node failures. PERLA focuses on connection failures using a unique technique that avoids overreacting by implementing permanent route adjustments.

4.2.2 Cluster-based Networks Data Aggregation

In large energy-constrained sensor networks, sensors transmitting data straight to the sink is wasteful. Sensors in such circumstances can provide data to a local aggregator, also known as a cluster head, which gathers data from all sensors in the cluster and sends a brief digest to the sink. For the energy-constrained sensors, this saves a large amount of energy.

Cluster heads can communicate with the sink directly or over long distances through other cluster heads. Several network organization and data aggregation methods based on clusters have recently been suggested.

Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH is a hierarchical protocol in which data is sent from most nodes to cluster chiefs, who aggregate and compress it before sending to the base station. Every node employs a stochastic algorithm to determine whether it will become the cluster leader for that round at the end of each round. LEACH implies every node has a radio that allows it to communicate directly with the base station or the cluster head closest to it, although it is inefficient to use it at full power all of the time [42].

Nodes that have been cluster heads before are not allowed to become cluster heads again for R rounds, where R is the required percentage of cluster heads. After that, each node has a 1/R probability of becoming a cluster head again. Any node that is not a cluster head chooses the closest cluster head and joins that cluster at the conclusion of each cycle. After that, the cluster head prepares a data transmission schedule for each of the cluster's nodes.

All nodes except the cluster head communicate with the cluster head only via TDMA, according to the cluster head's schedule. To decrease inter-cluster interference, LEACH uses CDMA, with each cluster has its own CDMA code set.

Hybrid Energy Efficient Distributed Clustering Approach (HEED)

HEED is an outstanding cluster-based protocol for power balancing that selects CHs based on residual energy and node degree or density of nodes as a cluster selection parameter, which is a rational improvement over LEACH. It uses a combination of two clustering parameters to choose CHs on a regular basis. The principal parameter is each sensor node's residual energy, whereas the secondary parameter is the cost of intra-cluster communication as a function of cluster density. The main parameter is specified by the node's residual energy and is used to probabilistically select an initial set of CHs, whilst the secondary parameter is used to bind the bond and account for communication costs within the cluster. It was created with four

main objectives in mind: I extending network lifetime by distributing energy consumption; ii stopping the clustering process after a certain number of iterations; iii reducing control overhead; and iv producing well-distributed CHs and compact clusters The Hybrid Energy Efficient Distributed Clustering (HEED) algorithm is a multi-hop wireless sensor network clustering technique that delivers energy-efficient clustering routing by taking explicit energy into account. If a node receives either a tentative or final CH, it is considered covered. If a node completes HEED execution without selecting a state final CH cluster head, it is considered uncovered and declares itself to be the state final CH cluster head [43].

Clustered Diffusion with Dynamic Data Aggregation (CLUDDA)

CLUDDA [22] is a hybrid method that incorporates both clustering and diffusion principles. Within interest messages sent by the base station, CLUDDA provides query definitions. Each interest message contains a query specification, which explains the operations that must be done in order to generate a proper response based on the data components. By exploiting current query knowledge, interest transformation reduces processing overhead. During the initial stages of interest propagation, CLUDDA mixes directed diffusion [10] and clustering. Only cluster heads engaged in inter-cluster communication are responsible for transmitting interest signals due to the clustering mechanism. Regular sensor nodes are only required to broadcast data by CLUDDA if they can respond to a request, which saves energy. Any cluster head that is familiar with the query description in CLUDDA can aggregate data, therefore the aggregate points are dynamic. Additionally, each cluster head keeps a query cache that lists all the different data elements that were combined to create the final data. Cluster chiefs also keep track of the addresses of the surrounding nodes from which data transmissions come. Instead than broadcasting, these addresses are used to send interest messages to specific nodes.

Cross-Layer Commit Protocol (CLCP)

The CLCP is divided into two phases. During the decentralized commit phase of the first step, participants cast votes while also attempting to come to a decentralized commit conclusion. In contrast to [10], which requires one or more centralized leaders, CLCP can abandon a transaction during, if the database does not vote for commit at all, a decentralized commit phase will be used. A termination phase will occur if the protocol is unable to advance owing to network partitioning. Similar to [10], the commit decision is organized by one member who is designated as the leader, who also makes sure that it is approved by a majority of participants, or more than half of the transaction participants. A new participant becomes the new leader with a new version number to distinguish it if the current leader fails or if a commit decision cannot be made after a timeout. During the decentralized commit step, the transaction choice is usually determined. This phase of CLCP allows for more transactions to finish than [10] because to a decentralized timeout mechanism, which increases CLCP performance and lowers energy costs.

Clustered Aggregation Technique (CAG)

This protocol is primarily intended for reactive networks. All sensor nodes that detect the same physical data constitute a cluster, which performs data redundancy checks by filtering out undesired elements and thereby reduces reaction time. Improved storage efficiency and decreased communication costs are also addressed by CAG. The improved CAG method is an improvement over the original CAG algorithm in that clusters are still formed by nodes that detect comparable values under a predetermined threshold, but the clusters endure for as long as the sensor readings continue to fall inside that threshold over time (temporal correlation). The size of sensor readings or network topology no longer have an impact on CAG performance. The protocol alternates the question and response phases when used in interactive mode to conserve energy. A WSN homogeneous clustering approach called Energy Efficient Homogeneous Clustering Method for Wireless Sensor Networks [22] reduces power consumption and increases network lifespan.

4.2.3 Grid-based Aggregation

This approach divides the area of a sensor network into numerous grids. In the sensor network's predetermined zones, a group of sensors performs the role of data aggregators. As a result, there is a data aggregator in each grid (also known as an integrator). In this area of the sensor network, the sensor array serves as an aggregator and integrator. The data aggregator, which gathers data from all of the grid's IoT sensors, receives data directly from the sensors in that grid. Grid-based aggregation does not allow individual IoT sensors to connect with one another. Grid-based data aggregation is noted for its ability to adapt to changing network conditions.

Grid-clustering Routing Protocol (GROUP)

This WSN routing system is based on clusters and uses little energy. One of the sinks creates a cluster grid dynamically, proactively, and randomly to convey query messages and data packets in this protocol [42].

Real-time applications like the detection of forest fires can use GROUP. To identify a forest fire, a number of sensor nodes are placed throughout the forest. These nodes can measure the temperature, smoke content, and relative humidity of the air. With the use of this protocol, these nodes are grouped into clusters so that each node always has a matching cluster head. As a result, sensor nodes receive query messages from sinks and send data packets to sinks using GROUP [42].

Aggregation Tree Construction Based on Grid (ATCBG)

ATCBG is considered as a better alternative to GROUP. In an event-driven WSN, this protocol seeks to aggregate the periodic data acquired by all network nodes. ATCBG is built on a set of

assumptions that are outlined below [30].

In the network, there is just one static sink. Each node in the network is fixed and aware of its own location, which may be ascertained via positioning methods. Based on their actual distance, nodes can modify their transceiver power.

ATCBG's main purpose is to create an aggregation tree with the sink as the grid's center and a cell size of R. Grids are used to organize the network, while clusters are created by combining many grids. CH is in charge of data fusion and is chosen based on parameters such as distance to the grid center, residual energy, and so on. JianShu et al. examine the aggregation tree structure [30], which is formed by all CH.

4.2.4 Chain based Data Aggregation

This strategy decreases the energy consumed in a single round by having each node just interact with its neighbor node and wait for its turn to transfer data to the Base Station (BS). Nodes must alternately take on the role of leaders in order to send data to the BS. In a network of sensor nodes, this method evenly distributes the energy burden. set of nodes on a play field at random, so the node is in a random location. The nodes are arranged in a chain that can be completed by sensor nodes or by a greedy algorithm starting at any node. BS then recalculates the chain and broadcasts it to all sensor nodes.

5. DATA AGGREGATION BASED ON NETWORK FLOW

Some protocols employ a graph to describe the sensor network, while the majority of data aggregation techniques can be categorized depending on network design. These protocols are referred to as network flow-based protocols since data aggregation is defined as a network flow problem. Network flow-based protocols' primary goal is to increase network longevity while taking energy constraints on sensor nodes and information flow limitations into consideration. Network flow-based protocols and techniques for optimizing them are covered in this section.

5.1 Maximum Lifetime Data Aggregation

Using effective data aggregation strategies, [17] looked into the maximum lifetime data collection with aggregation (MLDA) problem. The MLDA issue's goal is to develop a data collection schedule that enables sensors to gather incoming data packets for as long as is practically possible. Theoretically, the sensor network is represented as a directed graph G=. (V, E). The capacity fi,j of the edges of G represents the quantity of packets that were transferred from node I to node j.

Integer programming with linear constraints is used to create the best acceptable flow network. The integer program determines the maximum system longevity T given the constraints on edge capacity and sensor energy. As a scheduling method, the creation of a chain of aggregation trees that can aggregate and transport T data packets from each sensor to the sink is proposed.

5.2 Network Correlated Data Gathering

Data acquired by spatially close sensors is frequently linked in sensor networks. Cristescu et al. [20] looked on how to get network linked data. We are faced with the twin optimization challenge of rate allocation and transmission structure when sensors use source coding techniques. Data collection has been studied using Slepian-Wolf coding and mixed entropy coding with explicit communication. In Slepian-Wolf coding, greater rates are assigned to nodes that are nearer the sink, and lower rates are assigned to nodes that are farther from the sink. Nodes farther away from the sink are given greater rates in the explicit communication model, whereas nodes closer to the sink are given lower rates. A weighted graph G= serves as a representation of the sensor network (V, E). Each node I sends data via the network to the sink at a rate of Ri. Finding a spanning tree (ST) of G with rate allocations Ri that minimizes the weight of the path in the spanning tree from node I to node s is the objective of the lowest cost data gathering tree problem. The shortest route tree (SPT) is the optimal rate allocation in Slepian-Wolf coding when there is a single sink. In [20] provides an optimal Slepain-Wolf rate allocation approach. In this architecture, data is coded at a rate determined by the unconditioned entropy of the nearest node to the sink. The main drawback of this approach is that each sensor needs to be aware of the global network in terms of node distances. This issue has been addressed using a completely distributed approximation method with almost ideal solutions. This method only applies conditioning to nodes that are physically closer to the sink than the relevant node, and data is locally coded at each node.

6. DATA AGGREGATION BASED ON QUALITY OF SERVICE (QOS)

Since most of the data aggregation methods discussed thus far are designed with energy efficiency in mind, they provide networks that last for a very long time. However, for some applications, the most important need is a desired quality of service in terms of bandwidth, end-to-end delay, and information throughput. This section describes the data aggregation algorithms that are primarily concerned with ensuring such QOS metrics. The primary distinction is the performance metric used. Data aggregation protocols that concentrate on congestion control and end-to-end reliability and data aggregation protocols that maximize the amount of data collected at sinks while taking energy, latency, and dataflow constraints into consideration are two different research areas in QOS aware data aggregation.

6.1 End-to-end reliability and congestion control using the Data Aggregation Protocol

An aggregation method that carries out adaptive and time-sensitive application independent data aggregation was proposed in [23]. (AIDA). In their work, aggregation options are divided

into a module that sits between the network and data link layers. The main objective is to make the most of the communication channel. AIDA employs lossless aggregation, with the higher layer determining when information compression is necessary or not. A functional unit aggregates and de-aggregates network packets in the AIDA architecture. A control device that manages timing settings and adaptively changes the degree of aggregation is also present. By integrating several network units into a single AIDA aggregate, transmission and control overhead are decreased. AIDA aggregates a fixed number of network units into one AIDA packet when using the fixed aggregation approach. AIDA layer data aggregation only takes place in the on-demand aggregation strategy when the MAC layer is available for transmission. By dynamically altering the degree of aggregation threshold, the dynamic feedback technique combines on-demand and fixed aggregation. This method enhances aggregation performance by modifying the degree of aggregation threshold and the transmission rate.

6.2 Protocols for data aggregation for the best information extraction.

In [21] looked into how to get the most data out of energy-constrained heterogeneous sensor networks. The difficulty of maximizing data extraction from energy-constrained sensors is referred to as a multi-commodity flow issue with flow conservation limits. Efficacious heuristics like distance, hop count, and residual energy are used in a new approximation method that incorporates selfish and greedy behavior while minimizing the number of repeats. The link measure used for distance vector routing varies amongst the techniques.

In The link metric of a sensor, the exponential metric, changes exponentially with the remaining energy of the sensor at any iteration. All greedy techniques work the same when all nodes have the same data and energy levels. The exponential metric works better than the other heuristics when there are nodes with high energy but little data. Data flows produced by the exponential heuristic are within 15% of ideal. Contrarily, the performance of the exponential heuristic is affected by the node of the sensor and the heterogeneity of the data. Other greedy tactics, such distance and hop count, function effectively when all of the sensors are homogeneous. Additionally, the difficulty outlined in [21] does not solve the issue of data fairness. Priority must be considered when solving the data extraction problem since data from different sensors may have differing priorities.

7. OPEN ISSUES AND CHALLENGES

7.1 Gathering, Storing, and Processing Data

Given the large number of sensors that will be installed, the main focus of data collection will be on how to carry it out effectively while taking into account relative data compression. Sensitive data must be protected by ciphering and using the distributed system in order to preserve data collecting security. To use less energy, the best energy transmission and path selection will be investigated. With better data routing and well-known block chain technology, data storage can be secured while yet facing the same difficulties as energy reservation and data compression. Accuracy is broken down into precision and trueness in the ISO standard 5725:1994, which prioritizes sensor integrity. Contrarily, RF sensing uses channel state information for sensing and relies on machine learning to categorize sensed data due to its inherent EM nature. It also faces additional difficulties such as linearity, repeatability, resolution, hysteresis, temperature coefficients, stability, and calibration.

7.2 Scalability

Scalability in IoT systems has become a problem due to the increasing number of devices that need connectivity at once. There are two different types of scalability concerns in the Internet of Things: horizontal scalability, which refers to adding or removing IoT nodes, and vertical scalability, which refers to increasing or removing computational capabilities from an IoT node. IoT scalability has received a lot of attention in the literature because to its significance, and cloud computing or cloud-based architectures have been suggested. The need for IoT nodes to provide a greater variety of services, including functional scalability, access control, data storage, fault tolerance, privacy, and security, to name a few, still exists despite these efforts.

7.3 Security and Privacy

Lack of end-to-end security solutions and privacy standards has long been a barrier to the adoption of traditional IoT, and these barriers are resurfacing with wireless IoT. Several solutions are being developed to address privacy and security issues from both a hardware and software perspective. RFID, later generations of 5G, and other local network protocols are essential for addressing security issues at the hardware level. Software solutions like the Key Management System (KMS) and block chain, which have a zero-trust network characteristic, are quickly addressing security issues including privacy and trust. The main challenge for IoT devices is the interdependence of security, privacy, and trust for IoT ecosystems, which can be overcome by using contemporary communication protocols, KMS, and block chain. To attain maximum integrity and performance, the problem must always be considered as a totality.

7.4 Energy Efficiency

Building energy-efficient IoT networks has been attempted using a variety of strategies, including: Examples of data reduction techniques include developing energy-efficient routing protocols to decrease the number of hops, improving communication connection status, implementing wake-sleep algorithms based on network traffic, and decreasing data via network topology control. the network's usage of renewable energy sources and load-balancing algorithms. In order to solve the fundamental

problem of power management, which is crucial in large-scale heterogeneous IoT networks, wireless charging technologies are being deployed. However, from a hardware perspective, there is a significant need to create net zero-energy sensor nodes because the current trend is to cram an energy-constrained node with ever-increasing functionality, which may compromise fidelity and power efficiency.

Protocol	**Brevity**	**Advantages**	**Disadvantages**	**Net type**	**Ref**
Energy-aware data aggregation tree	EADAT	Sink node initiates the broadcast method.	Auxiliary broadcast messages do not have the ability to give a formula for calculating the power limit.	Tree	[44]
Power Efficient Data gathering and Aggregation Protocol	PEDAP	PEDAP extends the lifetime of the network..	This scheme only considers the shortest path. Bandwidth Utilization is not met. It is unable to reduce the Resource Utilization load, in particular.	Tree	[45]
Tiny aggregation	TAG	The use of multiple casts and query-based methods is supported.	Building an overhead track is underway.	Tree	[34]
Power efficient Routing with limited latency	PERLA	Don't take any unnecessary routes.	More effort is required to fix any issues.	Tree	[38]
LEACH	LEACH	cutting back on energy use	A hole close to the Sink node caused the first node to die prematurely.	Clu ster	[28]
Clustered diffusion withdynamic data Aggregation	CLUDD A	Communication inside the same cluster	There is still a pressing need for memory.	Clu ster	[32]
A grid- clustering routing protocol for wireless sensor networks	GROUP	the network's sensors are divided up according to the load	Aggregation tree is done on a regular basis, and clusters are chosen depending on grid distance.	Grid	[39]
Aggregation tree construction based on grid	ATCBG	Select a cluster based on energy and distance. less than half the needed energy in the cluster head.	Only energy is used to build trees construction.	Grid	[40]
Power efficient gathering in sensor information systems	PEGASI S	A different course of action is taken. The dissipation of network energy is balanced.	The chain leader is chosen by taking turns delaying data.	Chain	[31]
Chain oriented sensor network	COSEN	Reduce energy consumption and transmission time.	There are a lot of transmission paths that aren't needed.	Chain	[45]
An energy-efficient chain-based hierarchical routing protocol in wireless sensor networks	CHIRON	Reduce superfluous transmission paths to save energy and reduce data propagation time.	a large number of short chains	Chain	[40]

Table 1: list of the various methods used to describe the trade-offs in data aggregation

Category	Protocol	Accuracy	Energy consumption	Fault tolerant	Latency	Heterogeneity	Network lifetime	Scalability	Security	Traffic load
	Energy-aware dataaggregation tree	×	×	×	×	×	✓	✓	×	×
	Power Efficient Datagathering and Aggregation Protocol	×	✓	×	×	×	✓	×	×	×
	Tiny aggregation	✓	×	×	✓	×	✓	×	×	×
	Power Efficient Routing With Limited Latency	×	✓	✓	×	×	✓	×	×	×
	Tree Based Energy Efficient Protocol for Sensor Information	×	✓	×	×	✓	×	×	✓	×
	Low Energy Adaptive Clustering Hierarchy	×	×	×	×	×	✓	✓	×	✓
	Clustered Diffusion With Dynamic Data Aggregation	×	✓	×	×	✓	×	×	×	✓
	A Grid Clustering Routing Protocol For Wireless Sensor Networks	✓	✓	×	✓	×	✓	×	×	×
	Aggregation treeconstruction based on grid	✓	×	✓	×	×	×	×	✓	×
	Power Efficient Gathering In Sensor Information Systems	✓	×	✓	×	×	×	✓	×	×
	Chain oriented sensor network	×	×	×	✓	×	✓	×	×	✓
	An Energy Efficient Chain-Based Hierarchical Routing Protocol In Wireless Sensor Networks	×	×	✓	×	✓	×	×	✓	×

Table-2: Evaluation criteria for data aggregation systems and those mechanisms' key characteristics

Conclusion

This study looked closely at data aggregation methods in the internet of things. All of them are working to increase important performance metrics like energy usage, network longevity, and data latency and accuracy. Effective organization, routing, and data aggregation tree construction are the three key areas of interest for data aggregation algorithms. This study summarized the key traits, advantages, and disadvantages of each data aggregation technique. Trade-offs has been emphasized for energy efficiency, data accuracy, and latency. The majority of earlier research has been on creating an effective data aggregation routing system. The network infrastructure, however, plays a crucial role in how well the data aggregation protocol performs. The impact of heterogeneity and communication mode (single hop versus multi-hop) on the effectiveness of data aggregation algorithms has not received much attention.

References

[1] S. S. and I. P. Arda Surya Editya, "Performance IEEE 802.14.5 and ZigBee protocol on realtime monitoring augmented reality based wireless sensor network system," *Int. J. Adv. Intell. Informatics*, vol. 3, pp. 90–97, 2017.

[2] and K. K. L. Zhengguo Sheng, Shusen Yang, Yifan Yu, Athanasios

V. Vasilakos, Julie A. McCann, "A survey on the IETF Protocol suite for the Internet of Things: Standards, Challenges, and Opportunities," *IEEE Wirel. Commun. Mag.*, vol. 20, no. 6, pp. 91–98, 2013, doi: https://doi.org/10.1109/MWC.2013.6704479.

[3] S. S. and R. A. R. Hong Min Bae, Chan Min Park, "Performance Improvement in Beacon-enabled LR-WPAN-based Wireless Sensor Networks," 2016, [Online]. Available: https://doi.org/10.5220/0005632400890094.

[4] A. M. I. A. and D. S. B. Thora, "Internet of Things (IoT) Standards, Protocols and Security Issues," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 11, pp. 491–495, 2015, [Online]. Available: https://doi.org/10.17148/IJARCCE.2015.411109.

[5] *802.15.4-2011:IEEE Standard for Local and Metropolitan Area Networks- Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*,. 2011.

[6] *802.15.4e-2012: IEEE Standard for Local and Metropolitan Area Net-works - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC Sublayer*. .

[7] C. L. Devasena, "IPv6 low power wireless personal area network (6LoWPAN) for networking Internet of Things (IoT) - Analyzing its suitability for IoT," *Indian J. Sci. Technol.*, vol. 9, no. 2016, pp. 1–6, [Online]. Available: https://doi.org/10.17485/ijst/2016/v9i30/98730.

[8] and A. moeni Alireza radan, Hoseine samimi, "A new lightweight authentication protocol in IoT environment for RFID tags," *Int. Journl Eng. Technol.*, vol. 7, pp. 44–51, 2018, [Online]. Available: https://doi.org/10. 14419/ijet.v7i4.7.23028.

[9] A. M. and D. C. Jianping Song, Song Han, "Wirelesshart: Applying Wireless Technology in Real-Time Industrial Process Control," in *Conference: Real-Time and Embedded Technology and Applications Symposium*, 2008, pp. 77–86, [Online]. Available: https://doi.org/10.1109/RTAS.2008.15.

[10] M. S. and K. P. S. Suman Chhajed, "Wireless Sensor Network implementation using MiWi wireless protocol stack," *2014 IEEE Int. Conf. Adv. Comput. Conf.*, pp. 239–244, 2014, [Online]. Available: https://doi.org/10.1109/IAdCC.2014.6779327.

[11] J. A.-E. and P. M. T.-R. Juan Aponte-Luis, Juan Antonio Gómez-Galán, Fernando Gómez-Bravo, Manuel Sánchez-Raya, "An Efficient Wireless Sensor Network for Industrial Monitoring and Control," *Sensors 2018*, vol. 18, pp. 1–15, 2018, [Online]. Available: https://doi.org/10.3390/s18010182, MDPI Journals.

[12] H. V. and H. Z. Stefan Marksteiner, Víctor Juan Exposito Jimenez, "An overview of wireless IoT protocol security in the smart home domain," *Internet Things Bus. Model. Users, Networks*, 2018, [Online]. Available: https://doi.org/ 10.1109/CTTE.2017.8260940.

[13] J. M. and L. Han, "Performance analysis of the ZigBee networks in 5G environment and the nearest access routing for improvement," *Ad Hoc Networks*, vol. 56, pp. 1–12, 2017.

[14] M. H. M. and D. T. P. Kalaignan, "A Literature Survey On Zigbee," *Int. J. Curr. Eng. Sci. Res.*, vol. 5, pp. 57–60, 2018, [Online]. Available: https://doi.org/10.21276/Ijcesr.

[15] and C. Z. S. Wang, J. Wan, D. Li, "Implementing smart factory of industrie 4.0: an outlook," *Int. J. Distrib. Sens. Networks*, vol. 12, 2016, [Online]. Available: https://doi.org/10.1155/2016/3159805.

[16] S. D.-B. and D. S. S. Sonavane, "Implementation of 6LoWPAN Border Router (6BR) in Internet of Things," *Int. J. Innov. Adv. Comput. Sci.*, vol. 7, no. 3, pp. 269–273, 2018, [Online]. Available: https://doi.org/10.1109/ICNETS2.2017.8067900.

[17] "A technical overview of LoRa and LoRaWAN Alliance Technical Marketing Workgroup," 2015.

[18] H. V. Stefan Marksteiner, V. J. Exposito Jimenez and and H. Zeiner, "An Overview of Wireless IoT Protocol Security in the Smart Home Domain," in *Joint 13th CTTE and 10th CMI Conference on Internet of Things Business Models, Users, and Networks, Copenhagen*, 2008, pp. 1–8, [Online]. Available: https://doi.org/10.1109/CTTE.2017.8260940.

[19] T. S. and R. Jain, "Networking Protocols and Standards for Internet of Things," in *chapter 13*, John Wiley & Sons, Inc, 2017.

[20] J. León, "A proposal for a Bluetooth Low Energy (BLE) autoconfigurable mesh network routing protocol based on proactive source routing," 2016.

[21] S. C. and Dan Dragomir, Laura Gheorghe and A. Radovici, "A survey on secure communication protocols for iot systems." pp. 47–62, 2016, [Online]. Available: https://doi.org/10.1109/SIoT.2016.8.

[22] M. A. K. and K. Salah, "IoT security: Review, blockchain solutions, and open challenges", Future Generation of computer system," *Elsevier*, vol. 395–411, 2018, [Online]. Available: https://doi.org/10.1016/j.future.2017.11.022.

[23] E. Abdulrahman BIN Rabiah, K. K. Ramakrishna and L. and K. Kar, *A Lightweight Authentication and Key Exchange Protocol for IoT*. San Diego, CA, USA, 2018.

[24] and T. Anna Triantafillou, Panayiotis Sarigiannidis and D. Lagkas, "Network Protocols, Schemes, and Mechanisms for Internet of Things (IoT): Features, Open Challenges, and Trends," *Wirel. Commun. Mob. Comput. 2018*, pp. 24–30, 2018, [Online]. Available: https://doi.org/10.1155/2018/5349894.

[25] M. Asim, "A Survey on Application Layer Protocols for Internet of Things (IoT)," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, pp. 996–1000, 2017, [Online]. Available: https://doi.org/10.26483/ijarcs.v8i3.3143.

[26] A. J. and Jasenka Dizdarevic, Francisco Carpio and X. Masip-Bruin, "A survey of communication protocols for Internet-of-Things and related challenges of fog and cloud computing integration," *ACM Comput. Surv.*, vol. 1, pp. 1–27, 2018.

[27] E. by A. B. and R. Gupta, "MQTT Version 3.1.1.," 2014.

[28] B. T. and H. P. Nishant M. Sonawala, ""IoT Protocol based environmental data monitoring," in *IEEE Proceedings International Conference on Computing Methodologies and Communication*, 2017, pp. 1041–1045.

[29] T. M. T. and R. M. Banakar, "Data Transfer Protocols in IoT-An overview," *Int. J. Pure Appl. Math.*, vol. 118, pp. 121–138, 2018.

[30] OASIS, "Advanced Message Queuing Protocol 2012. Version 1.0," 2012.

[31] P. B. Jorge E. Luzuriaga, Miguel Perez and C. C. and P. M. Juan Carlos Cano, "Testing AMQP Protocol on Unstable and Mobile Networks," in *International Conference on Internet and Distributed Computing Systems Springer link*, 2014, pp. 250–260, [Online]. Available: https://doi.org/10.1007/978-3-319-%0A11692-1_22.

[32] N. Naik, "Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP," *IEEE Int. Syst. Eng. Symp.*, pp. 1–7, 2017.

[33] K. J. and K. P. Sneha Shailesh, ""Performanc analysis of RabbitMQ as a message bus," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 6, no. 1, pp. 241–246, 2018.

[34] H. T. Sye Loong Keoh, Sandeep S. Kumar, "Securing the Internet of Things: A Standardization Perspective," *Internet Things J. IEEE*, vol. 1, no. 3, pp. 265–275, 2014, [Online]. Available: https://doi.org/10.1109/JIOT.2014.2323395.

[35] K. H. C.Bormann, S.Lemay, H.Tschofenig and and B. R. B.Silverajan, "CoAP(Constrained Application Protocol over TCP, TLS and Websockets.RFC8323 , 2018.," in *RFC Editor*, 2018.

[36] P. S. Andre, "Extensible Messaging and Presence Protocol (XMPP): Core RFC 3920," in *RFC Editor*, 2004.

[37] J. R. and C.Pedraza, "Performance analysis of communication protocols for Internet of things platforms," in *IEEE Colombian Conference on Communications and Computing (COLCOM)*, 2017, pp. 1–7.

[38] W. R. H. and H. B. J. Kulik, "Negotiation-based protocols for disseminating information in wireless sensor networks , vol. 8, March 2002, pp. 169-185.," *Wirel. Networks*, vol. 8, pp. 169–185, 2002.

[39] R. G. and D. E. C. Intanagonwiwat, "Directed Diffusion: A Scalable and robust communication paradigm for sensor networks," *Proc. Sixth Annu. Int. Conf. Mob. Comput. Netw.*, 2000.

[40] B. K. and J. Heidemann, "Application specific modeling of information routing in wireless sensor networks," in *Proc. IEEE international performance, computing and communications conference*, 2004, pp. 717–722.

[41] X. C. and G. X. M. Ding, "Aggregation tree construction in sensor networks," in *IEEE 58th Vehicular Technology Conference*, 2003, pp. 2168–2172.

[42] H. O. T. and I. Korpeoglu, "Power efficient data gathering and aggregation in wireless sensor networks," *SIGMOD Rec.*, vol. 32, pp. 66–71, 2003.

[43] P. S. K. Vaidhyanathan, S. Sur, S. Narravula, "Data aggregation techniques sensor networks," 2004.

[44] P. T. et al. T. Winter, "RPL: IPv6 Routing Protocol forLow power and Lossy Networks," 2011.

[45] V. K. P. B. Hong, "Optimizing system lifetime for data gathering in networked sensor systems," *Workshop on Algorithms for Wireless and Ad-hoc Networks (A-SWAN)*. 2004.

[46] S. M. B. Abrar Alkhamisi, Mohamed Saleem, "A Cross-Layer Framework for Sensor Data Aggregation for IoT Applications in Smart Cities," *IEEE*, 2016.

[47] G. G. Sagi Sai Sruthi, "Efficient Secure Data Aggregation Technique for Internet of Things Network," 2016.

[48] and I. H. H. Rahmani, N. Ahmed I, "Comparison of Data Aggregation Techniques In Internet of Things (IoT)," 2016.

[49] J. J. Firas Al-Doghman, Zenon Chaczko, "A Review of Aggregation Algorithms for the Internet of Things," 2017.

[50] and A. S. Tianqi Yu, , Xianbin Wang, "Recursive Principal Component Analysis based Data Outlier Detection and Sensor Data Aggregation in IoT Systems," *IEEE Internet Things Journal.*, 2017, doi: 10.1109/JIOT.2017.2756025.

[51] and D. E. W. Ye, J. Heidemann, "An energy efficient mac protocol for wireless sensor networks," in *In Proc. of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies*, 2002, pp. 1567–1576.

[52] and X. C. Ruinian Li, Carl Sturtivant, Jiguo Yu, "A Novel Secure and Efficient Data Aggregation Scheme for IoT algorithm," *IEEE Internet Things Journal.*, 2018, doi: 10.1109/JIOT.2018.2848962.

[53] S. K. and V. K. Chaurasiya, "A Strategy for Elimination of Data Redundancy in Internet of Things (IoT) Based Wireless Sensor Network (WSN)," *IEEE Syst. J. 2018*, 2018, doi: 10.1109/JSYST.2018.2873591.