Methods, Performance Bounds, and Routing Approaches for In Network Aggregation in Wireless Sensor Networks

Fabio Caccioli Capra
Department of Computer Science and Engineering, London Global University, London.
fabiocapra121@hotmail.com

Vincenzo Anselmi
Department of Computer Science, University of Salerno, Fisciano SA, Italy.
anselmivin@unisa.it

Article Info
Journal of Computer and Communication Networks
https://www.ansispublications.com/journals/jccn/jccn.html
Received 10 May 2025
Revised from 18 July 2025
Accepted 30 August 2025
Available online 13 October 2025
Published by Ansis Publications

https://doi.org/10.64026/JCCN/2025020

Corresponding author(s):
Vincenzo Anselmi, Department of Computer Science, University of Salerno, Fisciano SA, Italy.
Email: anselmivin@unisa.it

Abstract – In-Network Aggregation (INA) is a process of data collection and analysis within a network architecture that is multi-hop network. The idea behind this process is to decrease the number of resources required to disseminate information and advance the optimality of the model by accumulating information at transitional nodes before forwarding it to the recipient. The process consists of two main methods: (a) the process of reducing the size of information to be sent in a way that minimizes the amount of overhead, and (b) the aggregation and conveyance of packet sizes. INA integrates routing procedures, accretion functions, and data representations as the constituent parts. This research aims to identify INA techniques in multihop networks with the aim of advancing the application of system resources as well as expanding the network size. The study encompasses issues relating to aggregation with or without size reduction, routing protocols, different types of aggregation functions, and different data representation formats. Lastly, this paper provides a detailed discussion that makes it possible to appreciate the strengths and weaknesses of cluster-based, tree-based, multi-path, and combined approaches. It examines the challenges of the reliability-overhead trade-off, thereby providing input on how to enhance the passing of data and use of resources in the WSNs.

Keywords – In-Network Aggregation, Multihop Network, Data Consolidation, Resource Optimization, Routing Protocols, Aggregation Functions, Wireless Sensor Networks.

I. INTRODUCTION

Recent technical advancements have enabled the placement of several nodes in an ad-hoc format [1]. These nodes are capable of coordinating and executing a range of monitoring duties, including detecting climatic parameters such as temperature, pressure, humidity, as well as monitoring noise levels, chemicals, and even performing intricate military vehicle tracking and surveillance operations. Irrespective of the intended use of the detectors, all of the above uses have some basic properties. First, dispensation is often governed by specific nodes that monitor the operation of either the whole network or certain portions of it. Secondly, monitoring is often conducted by transmitting queries over the framework to collect data from the nodes. The monitoring nodes record the results of the queries for further analysis. In monitoring applications, queries are executed iteratively rather than being performed just once, as is the case with traditional database queries. Instead, they are persistent and performed indefinitely over a stated duration, or until they are clearly terminated.

One common property of sensor node applications is the existence of notable constraints on energy and bandwidth in these networks. Sensor nodes in many applications depend on batteries for power. However, replacing these batteries is not only expensive but also sometimes impractical, especially in situations like deploying sensors in disaster-affected areas. So as to assure the long-term viability of the network in these scenarios, it is necessary to establish energy-efficient operations. The research conducted in [2] primarily centers on query optimization based on energy considerations. The limited range of radio transmitters, the wireless nature of node-to-node communication, and the high thickness of model nodes in certain places all contribute to bandwidth limitations. Recent studies [3, 4, 5] have focused on decreasing the amount of information transferred by using INA. The key idea is to create a comprehensive tree structure that will determine the future results. The internal nodes in the tree aggregate the values of their offsprings and then convey the accumulated result to their ancestors. In each epoch, it is preferable for a parent node to merge all partial aggregates from its child nodes and transmit a single partial aggregate upwards for the whole subtree.

In-network data aggregation enables the incremental processing of partial accumulation [6], while also allowing for distinct routing of data along the tree structure towards the root. As a result, each node is responsible for accumulating the whole aggregated results. This phenomenon reduces the amount of data packets being exchanged among neighboring nodes, resulting in minimal consumption of energy in the model and spreading the lifespan of the nodes in the sensor system [7]. Data aggregation is frequently applied in order to manage electricity and improve energy use. It is the act of placing the various input values in one subject. WSNs may contain redundant data because of several sensors present in the network. This redundancy happens where these sensors are placed close to each other and they are perceiving the same thing. Hence, data aggregation significantly reduces redundant data at the senders' and receivers' ends in the entire process. On the other hand, aggregate inquiries might be applied in a given circumstance with a view of reducing the number of messages. These queries are resoluted in accordance to the data passed to the model by each node in the model. For instance, if the query is to get the temperature in a specific area, then the sensor values are averaged through the use of an aggregate function and then conveyed to the base position. Thus, the base station does not require reception of all the detected values. Otherwise, the mean value can be considered by the network during the transferring of packets by the base position. In-network data aggregation is defined as the method in which the intermediate nodes gather and integrate data before forwarding it to the sink.

The article is focused on the application of resource utilization in multihop networks, and especially concerning wireless sensor networks (WSNs). Thus, studying various approaches to data aggregation within a network, our purpose is to improve data communication, to reduce consumption of energy, and to increase a network's ability to process more and more data. Understanding what routing protocols are and how combining functions work, as well as how to represent data, is crucial to creating valuable solutions that will not cost too much in terms of resources while still preserving data's integrity. This paper overviews significant findings in enhancing the network performance, which plays a significant role in many applications including the smart environment, smart manufacturing, and smart structures. The main goal is to provide significant improvements to the field of wireless communication networks and their sustainability and reliability.

The consequent sections of the paper are arranged as shown: Section II provides a discussion of In-Network Aggregation (INA), which integrate its principles, hop-by-hop encoding, and concealed data aggregation. INA techniques, which include routing protocols, aggregation functions, and data representation have been discussed in Section III. Section IV provides a review of theoretical limitations of INA techniques, which Section V reviews networking hierarchies and protocols for INA. In this section, various approaches have been discussed, which integrates tree-based approaches, multi-path approaches, cluster-based approaches, and hybrid data aggregation approaches. Lastly, Section VI provides a summary of the discussions in this article.

II. IN-NETWORK AGGREGATION

One of the key applications for WSNs [8] is to monitor ecosystem data and send it to a central location, also known as a sink. A straightforward approach would be to convey all data packages from the detectors to the sink. Due to the limited radio range, intermediary nodes within the network are responsible for forwarding messages from sensors that are located outside of this range. This simple technique may be readily implemented, as seen in **Fig. 1**.

Principle of In-Network Aggregation

The principle of INA refers to the procedure of accumulating and summarizing data inside a network rather than sending all individual data points to a central location. The transmission and forwarding of data packets from sensors to aggregator nodes need a significant amount of energy. Minimizing network traffic is a substantial challenge in the design of WSN utilizations since delivering one bit of statistics requires the same quantity of energy as performing 50 to 150 directives on motes [9]. The in-network aggregation (INA) is a well-recognized method that minimizes the energy consumption during packet spread in the sensor-to-sink state. Often, the sink does not need precise readings from all detectors, but rather a derivative like the total, average, or aberration. The concept behind the INA is to consolidate the necessary data for calculating derivatives as near to the origin as feasible, rather than broadcasting all detected values over the whole network.

Sensors transmit their readings to aggregator nodes, which consolidate the data into a single data packet. **Fig. 1** depicts the process in which the data collecting sensor nodes S1, S2, and S3 convey their statistics to the transitional collector node A2, 2. This collector node combines the three values into a single packet, rather than just promoting three separate packages. The subsequent aggregator node (A1,2) merges additional data that are ultimately sent to the sink. An important concern related to the INA is the safeguarding of the statistics. If the data is encoded using conventional techniques on the nodes, only the sink has the ability to decrypt it, which would make the INA feature non-functional. By sacrificing end-to-end confidentiality, the statistics could be encoded on a hop-by-hop basis. The hidden data aggregation technique offers the potential to combine ETE-security with INA. CDA enables the aggregation of encrypted data on intermediary nodes deprived of the need for decoding. The following subsections outline relevant methodologies for ensuring safe Information Network Architecture (INA) in Wireless Sensor Networks (WSNs).

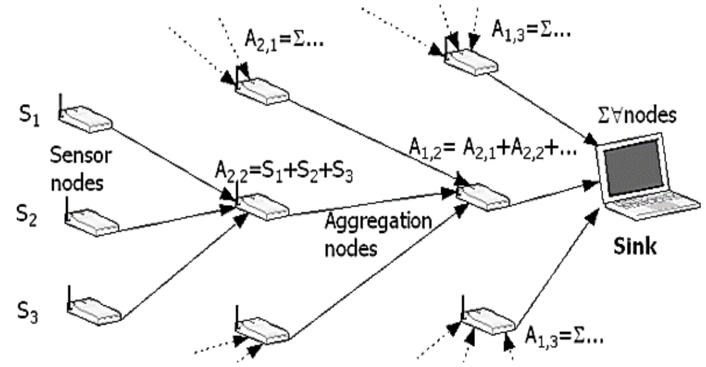


Fig 1. Fundamental Concept of INA.

Hop-by-Hop Encoding
Encrypted accretion on a hop-by-hop basis makes sure that the data being transmitted via the wireless channel is encrypted and the real data cannot be intercepted by eavesdroppers. Each transitional node decodes the data in order to further process it by aggregating it. With that, the node encrypts the outcome and passes it on to the next receiver. The HBH technique is disadvantageous in terms of energy utilization as it involves decryption and encryption of data on each aggregate node. Another major disadvantage is the lack of end-point security. One of the dangers of WSN is the possibility that the nodes in between may be reached or possibly compromised, and this is feasible for an attack. By simply integrating the widely-used cryptographic collections [10] like tinySec, the HBH-INA method successfully ensures the confidentiality of the information transmitted over the wireless channel without significantly increasing the number of sent packets.

Concealed Data Aggregation (CDA)
Concealed Data Aggregation (CDA) is a more advanced type of INA that provides ETE (End-To-End) privacy assurance. While in HBH case, the encrypted variables could not be aggregated without decryption, CDA provides such a provision. Instead, the process of combining data is carried out on encrypted values, and only the recipient has the ability to decode the final outcome. The core concept of CDA revolves on privacy homomorphism purposes, which possess the property of $enc(a + b) = enc(a) \oplus enc(b)$. This means that performing an action on two encoded ideals yields the same outcome as encrypting the sum of the two unencrypted ideals. While this work does not aim to provide a detailed explanation and analysis of CDA algorithms, the following subsections will briefly describe three schemes for WSNs. This introduction will facilitate understanding the property evaluations discussed later in [11].

CaMyTs
Peter, Westhoff, and Castelluccia [12] introduced a CDA technique that is based on a key stream. The concept involves executing a segmental accumulation operation between a traditional stream cipher and the data that has been detected. Every sensor employs a distinct pseudo random stream cipher. During the encoding process, the plaintext is directly combined with the stream`s present key, using the modulo operation with the key space`s length. In order to get the plaintext, the sink must subtract the matching key stream. CaMyTs is demonstrably safe and highly suitable for deployment on WSNs due to its need of just one segmental addition for aggregation and encryption. An issue arises with the decryption process, since it necessitates the use of identical key streams for each individual mote. The task of dispensing all the streams is not just an estimation challenge, but also requires the sink responsible for decrypting the combined ethics to remain coordinated with each key stream. Additionally, the sink must be able to identify which sensor data contributes to the combined.

Domingo-Ferrer
In [13], authors presented a symmetrical Domingo-Ferrer (DF) scheme, which has also been suggested as an effective CDA structure for WSNs in [14]. The method is symmetrical, meaning it uses the same secret key for both decryption and encryption. The accretion is conducted using a publicly available key. Uniform application of a single secret key on all nodes in the model is mandatory. While acknowledging the algorithm's security vulnerabilities, it might be deemed suitable for the majority of sensor network applications, provided long-term confidentiality is not a crucial factor. Due to the segmentation of the plaintext, the ciphertext is somewhat bigger in size than the original plaintext.

Hybrid
A hybrid technique called hCDA has been proposed in [15], which merges two previous CDA systems into a single combined scheme. It runs both embedded techniques successively while achieving the feature of additive privacy homomorphism. The

concept was inspired by the recognition that every scheme had inherent vulnerabilities, which may be effectively addressed by the integration of two distinct systems. The combination of CaMyTs with DF seems to be particularly promising. The enhanced level of security also leads to a corresponding increase in the computational effort required for encryption and decryption. This is due to the fact that the two embedded methods need the execution of two separate encryption processes on the nodes.

III. IN-NETWORK AGGREGATION TECHNIQUES

In-network aggregation, as defined by Fasolo et al. [16], refers to the process of routing and gathering data across a multi-hop framework. In the process of reducing resource consumption and expanding the scope of the framework, the data is processed at the intermediate nodes. The process of INA may be categorized into two methods: accretion with size decrease in a network basically involves joining several pieces of data from dissimilar sources in a network and then decreasing the amount of data which has to be conveyed. For example, there are two sources independent from each other but both of them are sending data to a certain node and it contains the very similar local data, the node may transmit a single packet containing the average value of this data. This method is by far very effective in minimizing the amount of data transferred within the model. INA without size reduction implies that packets that are collected from dissimilar sources are combined into a single package without the reformatting of information. For instance, it is assumed that a node obtains two packages with two dissimilar measurements, for example, humidity and temperature. Two ideals cannot be packed into a single datum, but they may be conveyed in a single package, which may reduce overhead. INA approaches need three fundamental components: appropriate networking protocols, efficient aggregation algorithms, and effective data representation methods. In the next part, we provide a concise introduction to each of these features.

Routing Protocols

A well-designed routing protocol is the crucial element in network aggregation [17, 18, 19]. Data aggregation necessitates a novel approach to routing, beyond conventional network routing techniques. Our objective is to minimize energy usage in data transfer by consolidating data. Data-centric routing is a strategy where nodes choose their next hop in accordance to packet content to facilitate the execution of in-network aggregation process. This approach employs metrics to identify dependable nodes that take into account the optimal sites for aggregation, as well as the nature and importance of the information.

Aggregation Functions

INA solutions should possess the crucial capability to merge data originating from various nodes. There are numerous kinds of accretion purposes [20], most of which are tailored to particular sensor applications. Depending on the following parameters, we can subdivide the mentioned strategies into the following categories:

Lossless and Lossy

There are two methods for performing accretion purposes to combine or condense information: The two major approaches used in compression are the lossless and the lossy methods. In case of a lossy technique, the inventive ideals cannot be produced once the data is combined using an accretion purpose. In addition, it becomes possible to decrease the accuracy when it comes to sending uncompressed information. The second strategy, which is lossless data compression, actually compresses the data at the same time as the original data is retained. This helps us in enabling the restoration of the readings at the receiver end regardless of whether an accretion purpose has been used.

Duplicate Insensitive and Sensitive

A transitional node may occasionally receive redundant duplicates of identical data from multiple sources. However, in this context, the duplication of data should be avoided during the aggregation process to reduce the consumption of resources. The frequency of identical values is also considered in the last result because of using a duplicate-sensitive accretion method. On the other hand, if the aggregate function disregards duplicate values, then it is called a duplication insensitive aggregate function.

Data Representation

A node has a fixed-size buffer and this might sometime cause data overflow because the amount of storage space a node has is limited. Thus, the node could not be able to keep all the data which it acquires from the origins in its immediate vicinity within the internal buffer. Thus, it is required to employ numerous measures to guarantee that the buffer is available for either rejecting or dispatching data. Therefore, it is necessary to display the information in an appropriate manner [21]. Meanwhile, data structure is considered a hurdle because to its need for flexibility in relation to application and geographical factors. Distributed source coding approaches are specifically designed to handle data representation.

IV. THEORETICAL LIMITS OF IN-NETWORK AGGREGATION TECHNIQUES

This aggregation approach involves the consolidation of observed input data at intermediate nodes in order to reduce energy depletion. Additionally, it enhances the longevity of the network by reducing the power consumption at each individual node [22]. The many types of proposals for in-network aggregations include:

- *Lossy aggregation:* Data is gathered from various nodes in the model and subjected to a limited number of aggregate operations, such as SUM (), MIN (), MAX (), and AVG (). The data packet's coverage is diminished. The base station receives the individual element value instead of receiving the whole packet from each node.
- *Lossless aggregation:* Lossy aggregation is required so as to provide a meaningful response to the base station. Here, each packet is merged with an uncompressed packet. For example, in a jungle fire alert, it is necessary to rapidly get the lowest or maximum thermal measurements.

Various theoretical studies establish constraints and limitations on the efficacy of in-network information accretion approaches, hence aiding in the development of appropriate systems. The efficacy of these processes is contingent upon the connection between the data produced by disparate data sources. Connection can occur in three ways: spatially, when values from nearby sensors are connected; temporally, when sensor analyses change gradually over time; or semantically, when dissimilar data packages can be grouped into the same semantic cluster (for example, data created by detectors in the same room). The benefits of in-network information accretion are most evident in situations when data from many origins can be merged into a single package, such as when the origins provide similar information. If there are K origins in close proximity to each other and distant from the sink, combining all data into a single package result in an average decrease of K-fold in broadcasts compared to sending all data separately. In general, the most efficient structure for joint compression and steering is a Steiner tree, which is recognized as being NP hard [23].

Nevertheless, there are polynomial solutions available for certain scenarios in which the data sources are in close proximity to each other [24]. Leibovici [25] provide a model that describes the spatial correlation using joint entropy as a measure. The researchers examine a symmetrical line network that exhibits varying levels of correlation between adjacent nodes. In the scenario when there is no correlation, the authors demonstrate that the optimal steering approach is to transmit packets via the shortest pathways. Conversely, when information is entirely linked, the optimal approach is to promptly combine the data. Subsequently, a solitary packet, comprised of the consolidated data, is sent to the sink through the most direct route. Clustering-based methods are perhaps the most suitable option for all intermediate scenarios, while the research does not provide a formal demonstration.

Luo and Oyedele [26] examine the influence of data connection on the vigour consumption of data circulation algorithms. Their focus is directed towards proposing one or more energy-aware data aggregation trees in various networking environments, like data aggregation factor, source density, node density, and source dispersion factor. These observations are consistent with the outcomes in [27] and present more accurate quantitative values. Bilò, Gualà, and Proietti [28] particularly focus on tree networks and present the analysis of the Shortest Path Tree (SPT) and Minimum Steiner Tree (MST). Unfortunately, it should be noted that aggregating trees have been proven to be in the optimum form through the use of the MST. Although the SPT has minimized latency and may be built online, its aggregation capability is severely less efficient as compared to the MST. In addition, the study in [29] investigates the cost ratio between the opportunistic aggregation and the systematic aggregation. This ratio shows the comparative cost of the connection ignorant SPT tree and the connection aware MST tree, with the same sinks and sources. Based on the analysis, the authors show that the anticipated increase in cost of MST compared to SPT in the sensor models rises at a rate of $O(\sqrt{\log N})$ where N is the sum of nodes in the models. This outcome shows that SPT is a suitable solution in several real settings, especially in case of minuscule networks.

According to Boone and Bullock [30], there is a proposed tree structure known as Spatial/Semantic Correlation Tree (SCT). The SCT method relies on the existence of an accretion backbone that is used to construct the accretion trees that are optimal and do not depend on the dispersion and density of the sources. The goal is to properly build and uphold the model edifice for the coalescence of data. To this end, Liu and Brookfield [31] propose to divide the network into ring sectors. Some of the nodes are chosen to act as aggregation nodes and all the aggregation nodes are connected in a straddling tree to form the data aggregation tree. Messages from a particular sub-area are gathered and consolidated at each node in the backbone network. Zarebavani et al. [32] also propose another kind of aggregation that uses the trees exploiting data dependency. The concept is in accordance to shallow light trees (SLT), which combine the characteristics of MST and SPT. Whereas the lengths (delays) among the sink and each node are only a continuous factor longer than the shortest pathways, the overall cost of the tree in an SLT is only a continuous factor bigger than that of the MST. Narvaez, Siu, and Tzeng [33] examine the accretion characteristics of a tree edifice that relies on a Shortest Path Tree (SPT) of nodes near the base station. Nodes that are extra distant are linked to the SPT`s leaves using pathways determined by an estimate approach for the traveling salesman challenge. Replications demonstrate that these trees surpass SLTs in several instances.

V. NETWORKING HIERARCHIES AND PROTOCOLS FOR INA

In the subsequent, we will examine each category of routing procedures individually, including hybrid, multi-path, cluster-based, and tree-based protocols. We will discuss the key ideas and provide a short analysis of the benefits and limits of each scheme. From the quantity of schemes mentioned in each part, it is evident that several solutions are put up in the cluster-based and tree-based categories. However, only a small number of research use the hybrid and multi-path techniques.

Tree-Based Approaches

Traditional routing algorithms [34], [35] often rely on a ranked structure of the nodes inside the model. Indeed, the most straightforward method to collect data moving from the quotes to the sink is to designate certain nodes as accretion ideas.

These nodes will serve as central locations for data aggregation and will establish a preferred path for data forwarding. Furthermore, a node may be designated as superior based on many variables, including its placement within the information collecting tree [36], its available properties, the kind of data kept in its line, or the dispensation cost associated with accretion methods. The tree-based technique [37] involves constructing a spanning tree with the sink as the root. Consequently, this structure is used in responding to inquiries created by the sink. This is achieved by the process of in-model accretion, where data is aggregated inside the network itself. The aggregation occurs along an aggregation tree, with each level being processed sequentially from the leaves to the root.

Therefore, when many messages reach a certain node, their combined value may be accurately calculated. Nevertheless, this operational approach has several limitations, since real wireless sensor models are not exempt from malfunctions. Specifically, if a package is lost at a certain level of the tree, such as due to conduit damages, the information originating from the corresponding subtree is also lost. Indeed, a solitary message at a fixed stage of the tree has the capacity to consolidate the data originating from the whole associated subtree. Despite the theoretically expensive nature of sustaining a ranked structure in energetic models and the limited resilience of the model in the event of link or gadget letdowns, these techniques are well-suited for conniving optimum accretion purposes and achieving effective energy management. Several studies, [38], [39], [40] and [41], have shown that the sink effectively arranges routing circuits to uniformly and optimally divide energy consumption, while also promoting data aggregation at intermediate nodes. Xue, Cui, and Nahrstedt [42] use linear programming to calculate aggregation topologies, considering each node`s enduring energy.

Akkaya, Demirbas, and Aygun [43] examine the model nodes that may be used as aggregation sites to achieve best performance. In [44] and [45], the main concern is determining which nodes should be responsible for transmitting the detected data. On the other hand, in [46], the primary focus is on arranging the sleeping and active times correctly. Frequently, the most efficient routes are determined in a centralized way at the destination by leveraging various norms about the data association and choosing the most suitable accretion sites using cost purposes. In recent times, there have been proposals for tree-based methods designed specifically for time-constrained or actual-time utilizations [47]. Another strategy, known as aggregation trees, utilizes the creation of linked dominant sets [48]. These sets include a limited number of nodes that provide a linked backbone and are strategically positioned to gather information from any location within the model. Nodes that are not part of these sets are permitted to enter a sleep state when they do not have any data to transmit. It is advisable to rotate the nodes in the dominant set in order to achieve energy balance. In the subsequent paragraphs, we examine the primary routing methods that rely on aggregate trees.

TAG [49] – Tiny AGgregation (TAG), a general-purpose collection service for ad hoc frameworks of TinyOS nodes, was created by the authors. There are two significant characteristics of this service. First, it presents a modest and effective approach to collect and aggregate data by mimicking the assortment and combination properties of database query languages. In addition, it effectively broadcasts and handles the collective query in the sensor network, considering the constraints of resources and lossy nature of communication in WSNs but at the same time focusing on time and power. TAG operates on aggregates in the network in that it performs computations on the data that is being transmitted in the sensors. It eliminates unnecessary data and combines essential readings into more condensed records wherever feasible.

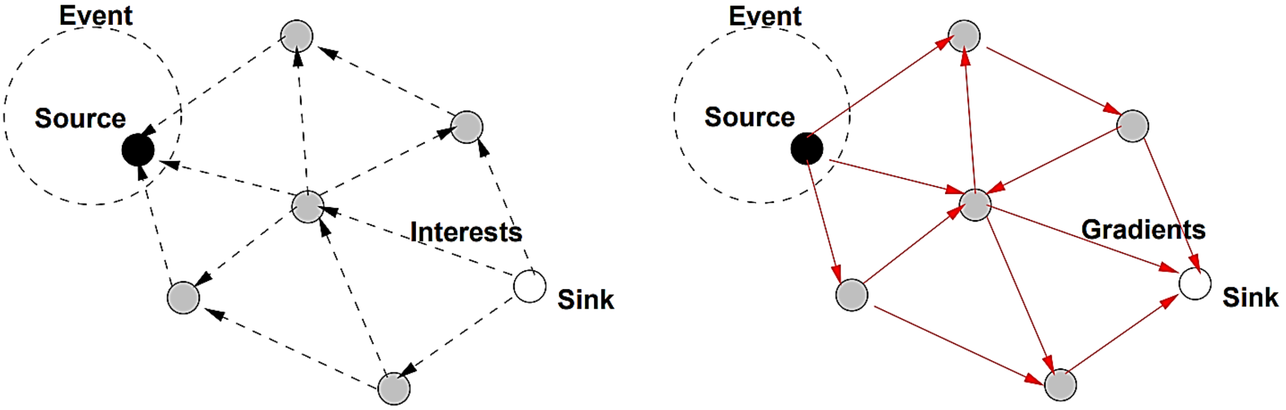
The operators who execute the query are scattered around the model by using the current ad hoc networking procedure. Data collected by detectors is sent back to the user via a steering tree that originates at the base station. As data ascends this hierarchical structure, it is combined together depending on a given accretion purpose and value-created segmentation mentioned in the query. For instance, let's assume a query that calculates the total amount of nodes in a model of unknown size. Initially, the demand to tally is inserted into the model. Subsequently, every terminal node within the tree conveys a numerical value of 1 to its immediate ancestor; intermediate nodes aggregate the numerical values of their offspring, increase the total by 1, and transmit the resulting value to their immediate ancestor. Counts are sent upwards through the tree in this way, and are emitted from the root. The sink regularly broadcasts routing messages to maintain the updated tree topology. Once the tree is built, the inquiries are sent via the network to all nodes in the structure. SQL and other database query languages' selection and aggregation features are integrated into TAG. TAG inquiries are therefore formatted as follows:

```
SELECT{agg(expr), attrs} from SENSOR
WHERE{selPreds}
GROUP BY{attrs}
HAVING{havingPreds}
EPOCH DURATION i
```

Practically, the sink transmits an inquiry that includes the desired amounts to be collected (specified in the attrs field), the method of aggregation (agg(expr)), and the detectors that should be used to get the data. The final demand is defined by the HAVING, GROUP, and WHERE clauses [50]. An EPOCH period sector provides the specific amount of time (in seconds) that each gadget should wait before providing fresh detector data. This indicates that the measurements used to calculate a collective record are all from the same time period, known as an epoch. Throughout the data gathering phase, the parent nodes must wait for data from all of their child nodes since they can transmit their combine data up the tree structure. Epochs are subdivided into smaller time periods known as conversation slots. The quantity of these slots is equivalent to the utmost complexity of the steering tree. The slot system provides a favorable advantage.

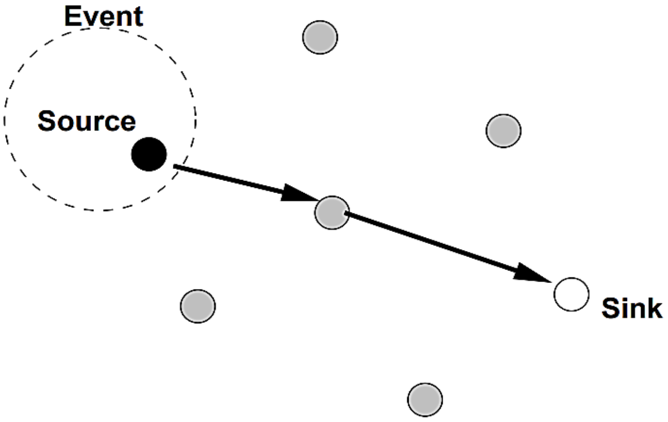
Since the time is divided into slots, motes may be deactivated until the next planned broadcast period. Typically, a node promptly returns to a dormant state after it has completed transmitting its data to its parent. All intermediary nodes execute data aggregation. However, to avoid restricting TAG to only the limited and basic accretion purposes specified by the SQL language (like AVERAGE, SUM, MAX, MIN, and COUNT), a broader categorization is considered based on the Exemplary and Summary, Duplicate Sensitivity, and Monotonic features. Like other tree-based systems, TAG may be inefficient when dealing with changing topologies or link/gadget failures. since mentioned earlier, trees are especially vulnerable to failures at transitional nodes, since this might cause the corresponding subtree to become disconnected. Furthermore, when the topology undergoes changes, TAG must rearrange the tree structure, resulting in significant energy usage and overhead expenses.

Directed Diffusion [51] - Directed diffusion has many components, including reinforcements, gradients, and interests. An attention communication is a request or inquiry that stipulates the user's desired information or action. Each interest message includes a detailed explanation of the data that a user is interested in. Data in sensor models generally refers to the data that is gathered or analyzed from a certain phenomena, in accordance with the preferences or requirements of a user. Data of this kind might refer to an occurrence, which is a concise depiction of the seen reality. Directed diffusion use attribute-value pairs to name data. The interest is distributed during the sensor models to attract named data towards the user. Interest proliferation creates gradients throughout the model to facilitate the passage of data. More precisely, a gradient refers to a directional condition that is established inside each node to obtain an interest. The incline way is determined by the surrounding node from which the attention is established. Events are sent to the individuals who initiated the interest via several gradient pathways. The sensor model strengthens one or a few of these pathways (see **Fig. 2**).



(a) Interest Circulation.

(b) Original Inclines Set Up.



(c) Data Conveyance Along Emphasized Path.

Fig 2. A Streamlined Diagram for Directed Dispersal.

Furthermore, the node establishes interest gradients, which are vectors that specify the next hop neede to convey the query result back to the sink node. For instance, if the base position provides concern that is established by nodes a and b and both nodes communicate the concern to node c, then node c generates two vectors signaling that the statistics corresponding the concern should be transmitted back to either an or b. The intensity of this incline may be adjusted, potentially leading to varying levels of information being routed to each neighboring entity. For this purpose, other parameters like as the node's vigour level, its statement capabilities, and its location inside the system may be used. Each incline is associated with the specific characteristic it has been configured for. Following completion of the incline configuration process for a particular interest, packets are routed towards the sink via a single, reinforced path for each basis (path strengthening and advancing).

Data aggregation occurs when statistics are sent to the base position using appropriate techniques, which may be chosen based on the specific needs of the application. The sink must regularly update the data collection tree, which consists of reinforced routes. This process may be costly, especially when dealing with dynamic topologies. There is a tradeoff among the threshold of the gradient setting and the attained presentation, which depends on the network dynamics. This tradeoff involves the amount of energy used. In the process of creating gradients and fortifying channels, local interactions among nodes play a crucial role in directed diffusion. Because there is no longer a need to provide the entire network architecture to every node in the network, efficiency is increased. It's critical to concentrate on the MAC Layer's design. Consider the IEEE802.11 cellular technology, for instance. As previously mentioned, inquiries are sent by transmissions, which is the fundamental method of access in IEEE802.11.

However, unicast broadcasts are used to return statistics to the sink. Due to MAC impacts and ensuing backoffs, latency can become quite noticeable when node thickness rises or when the replica conquest rule is not applied. Therefore, it is essential to maintain the local traffic at a sufficiently low level to prevent any potential accidents. Various methods [52], [53], [54] have been suggested to minimize the switch traffic produced by the local relations between nodes using Directed Dispersion. The authors use well-defined accretion trees in their systems, primarily aiming to minimize both network traffic and latency. An adapted variant of Directed Dispersion, known as Enhanced Directed Diffusion (EDD), is presented in [55]. This procedure reduces local traffic and linked impacts by using a cluster-based design and Directed Dispersion to gather information and improve the effectiveness of local interactions.

PEGASIS [56] – The PEGASIS protocol is an improved technique in which one node is chosen to be the head node. In each round, this head node is supposed to send the integrated data to the base position. This results in a factor of two improvement over the LEACH procedure as described in [57]. The PEGASIS procedure demands the creation of a chain and it is done in the following two stages.

Chain creation: To form the chain, we start with the node that is situated in the maximum length from the base station (BS). Next, we apply a greed algorithm to build the chain. In **Fig. 3**, node c0 is located utmost from the base station. From point C0 to point C1, point C1 to point C2, point C2 to point C3, and point C3 to point C4 are the connections in the chain. This is how the chain is constructed. Finally, node c4 is connected to c5.

Gathering data: The leader for each round is chosen at random. Choosing the head node randomly also offers an advantage, since it increases the likelihood of nodes dying at various places, so creating a more resilient model. When a node becomes inoperative, the chain is reestablished to circumvent the non-functioning node [58]. After selecting a leader, it passes a token to start gathering data. Passing a token also needs consumption of energy, but because of the token's small size, the cost of passing a token is quite low. **Fig. 4** designates node c3 as the head node for the specific round. Node c5 transmits the data to c3 sequentially throughout the chain. c0 transfers the information to c3 sequentially in the chain. The c3 device obtains the information, consolidates all the received data, and transmits it to the base position.

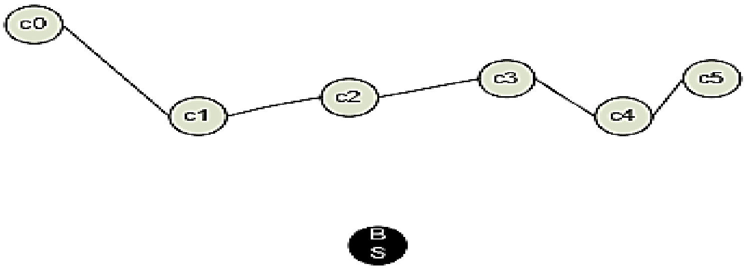


Fig 3. Construction of Chain in PEGASIS Using Greedy Approach.

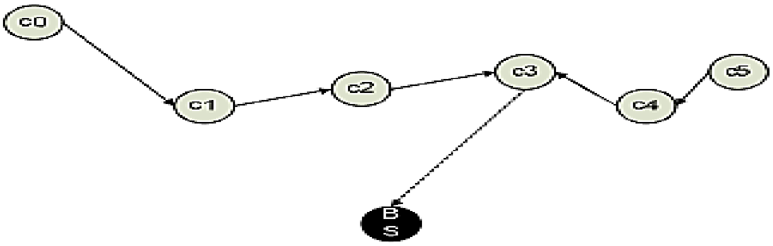


Fig 4. Data Fusion at the Head Node and Transmitting it to BS.

Within the PEGASIS protocol, every node obtains information from a neighboring node and combines it with its own analysis by creating a unified packet of equal size. Afterwards, this collection of data is sent to the next node in the sequence until the package attains the recent head of the sequence. At this juncture, the head incorporates its own information into the package and transmits it to the sink. One potential limitation of the approach arises from the proximity of neighbors. Indeed, when the neighboring entities in the sequence are too far apart, the amount of energy required might be exceedingly enormous. Furthermore, the distribution of transmission energies is not uniform, but rather contingent upon the specific distances between nodes and their neighboring nodes.

Specifically, nodes that have distant neighbors use a greater amount of energy. One way to improve PEGASIS is by implementing a threshold-based leader election mechanism to prevent certain nodes from becoming leaders. Two primary drawbacks of PEGASIS are the need for each node to possess a comprehensive consideration of the framework architecture so as to form a suitable chain, and the necessity for all nodes to have the capability to send information straight to the sink. This renders the method inappropriate for frameworks that have a topology that changes over time. Furthermore, the performance of this protocol may be impacted by connection failures and packet losses. Indeed, if any intermediate node fails, it jeopardizes the delivery of all data that has been collected and sent by the preceding nodes in the sequence. Therefore, certain enhancements to the system may be necessary to enhance its resilience.

DB-MAC [59] - The MAC protocol DB-MAC shown in **Fig. 5** utilizes a contention-based approach and is specifically tailored for delay constrained applications that rely on hierarchical data collecting trees. In fact, a transmission closer to the source is prioritized above a transmission closer to the sink. In addition, nodes will intercept contention time periods from others in order to facilitate early data aggregate integration. Thus, a node will have a higher likelihood of obtaining medium access if it is in close proximity to the source. Meanwhile, it optimizes route aggregation to be as near to the sources as feasible. When a source begins transmitting, the priority Pr is initially set to the highest possible value, PrMAX. Pr is thereafter reduced by one at each hop. The receiving node reduces the priority by one, changing PrMAX to PrMAX - 1, and then conveys the sachet to the next node. The following node will then compete for access to the medium with a priority of PrMAX - 1. The BI value is assigned a numerical value ranging from 0 to 1023, which is determined by the priority level. If a node is in close proximity to the source, it is very likely to get medium access. Priority access allows for reduced latency and energy savings as compared to the IEEE 802.11 system.

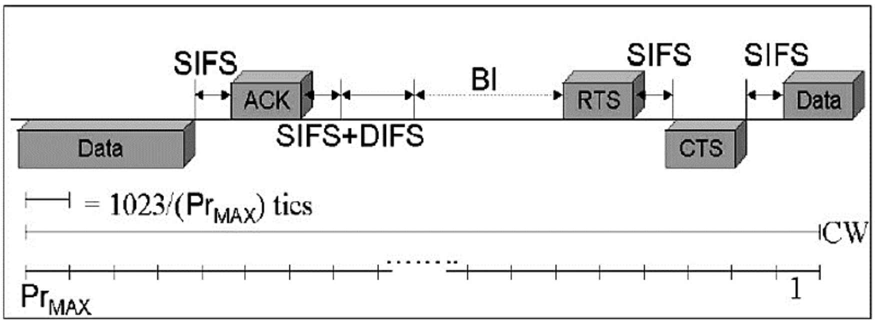


Fig 5. The Contention Mechanism in DB-MAC.

Cluster-Based Methods

Like tree-based networks, cluster-based methods [60], [61], [62], [63] also possess a hierarchical structure in the network. However, in this scenario, the nodes are categorized into smaller groups called bunches. Furthermore, designated nodes known as cluster-heads are chosen to assemble data in a certain area and send the combined statistics to the essential point. The benefits and drawbacks of cluster-based systems closely resemble those of tree-established techniques.

LEACH [64] - The Low-Energy Adaptive Clustering Hierarchy (LEACH) is a procedure that organizes and adapts clusters in a self-regulating manner. It achieves an equitable circulation of consumption of energy among the detectors via the use of randomization. Data aggregation is carried out by using clustered structures, in which cluster-heads serve as aggregate points. The procedure operates in a sequential manner and consists of two primary stages: 1) an initialization stage to arrange the clusters, and 2) a stable phase that handles the actual transmission of data to the central node. The nodes automatically organize into clusters during the initial stage. A node is selected as the bunch head for each bunch. Every sensor independently establishes itself as the local group head for the recent circle during the setup phase. This judgment is determined using a decentralized probabilistic methodology. The objective is to achieve an appropriate proportion P of nodes functioning as cluster-heads, which should be selected based on the node thickness. Practically, detectors compute the frequency provided in Equation (1).

$$T(n) = \begin{cases} \dfrac{P}{1-P\left(R \bmod \left(\frac{1}{P}\right)\right)} & if\ n \in G \\ 0 & otherwise, \end{cases} \tag{1}$$

where G represents all the nodes that have not been central nodes in the past 1/P circles; R is the recent circular number; and P is the objective proportion of cluster-heads. When a random integer is chosen from the interval [0, 1], node n represents the collection head if it is less than $T(n)$. Using the CSMA MAC protocol, a cluster-head communicates with the nodes that surround it. The signal strength of these signals determines the cluster affiliation of neighboring nodes. To efficiently manage local transmissions, each cluster head forms a TDMA schedule while accounting for the number of nodes wishing to join the cluster.

COUGAR [65] - The Cougar system is particularly well-suited for monitoring applications in which nodes frequently provide pertinent information. Since the cougar is essentially a clustering algorithm, the technique can be grouped as a periodic per hop accretion methodology. Once a cluster-head gains statistics from every bud in its group, it sends a partial aggregate to a gateway node. Undoubtedly, Cougar, like LEACH, is also susceptible to the same challenges in very dynamic circumstances. Significantly, Cougar distinguishes itself from prior clustering-based algorithms in its method of choosing cluster-heads. In contrast to LEACH, where cluster-head assortment is based only on signal strength measures, Cougar allows for the possibility of using other parameters to drive the cluster-head selection process. Indeed, a node has the potential to be located at a stretch greater than one hop from its central node. Based on AODV (Ad hoc On demand Distance Vector) technology, the steering system exchanges packets between clusters.

Due to AODV's lack of duplicate data packet generation, Cougar is well-suited for performing in-network aggregation with aggregators that are sensitive to duplicates. The fundamental Cougar method comprises the node synchronization engine, which guarantees accurate aggregation of data. Every cluster-head maintains a queue that includes all the nodes from whom it anticipates receiving communication. Constantly a node obtains a report from another node in its collection, the list is updated. When the cluster-head reaches time send, it waits to submit its interpretation to the entry until it has confirmation from every node in its waiting list. Each cluster head is equipped with a prediction mechanism to estimate the time of send. Furthermore, a subordinate node has the ability to ascertain whether its cluster-head is in a state of anticipation for a package from it and may use a notice package to enhance the accuracy of the forecast at the cluster-head. Backoffs and timeouts are used to address incorrect forecasts.

Multi-Path Approaches

A novel technique has been introduced in recent publications [66], [67], [68] to address the resilience issues associated with aggregation trees. Despite using an accretion tree structure where each node is required to transmit its incomplete aggregate result to a single parent node, these methods use the transmission of data across several channels. The primary concept is that every node has the capability to transfer information to its potentially many coresidents by using the transmission properties of the wireless standard. Therefore, information may be sent from the bases to the sinks by many routes, and each node has the capability to conduct aggregation. It is significant to note that, unlike the tree-based methods mentioned before, multi-path systems enable the transmission of several copies of the same information. Undoubtedly, both methods prioritize increased resilience (by allowing for the transmission of numerous copies of the same data across various routes) at the expense of more overhead (caused by sending duplicates). A suitable aggregation structure for this approach is known as a rings topology, which involves dividing sensor nodes into different levels based on the sum of hops that separate them from the data sink. As packets go towards the sink, data aggregation occurs over various pathways. Next, we will examine the synopsis dissemination model, which falls within this category of procedures.

Synopsis Diffusion [69] -A recent study [70] has introduced a method known as synopsis diffusion, which blends energy-effective multiple-path steering strategies with intelligent algorithms to prevent the duplication of calculations. Synopsis diffusion provides for the separation of aggregate from message routing, enabling the use of arbitrary multiple-path steering and the adjustment of redundancy in message routing (in exchange for energy consumption) based on sensor network circumstances. This research aims to offer a formal framework for this novel method. Synopsis diffusion facilitates the separation of routing and aggregation by using order- and duplicate-insensitive (ODI) synopses [71]. As far as we know, this research is the first to provide a rigorous definition and examination of this significant category of summaries. ODI synopses are concise summaries of the incomplete findings obtained at a node, ensuring that each sensor reading is included only once. Put simply, the summary at a node remains unchanged nonetheless of (1) the sequence in which readings are established, and (2) the frequency at which a specific reading from a particular sensor reaches the node (either indirectly or directly via incomplete results). Creating summaries for aggregates like Max and Min is easy, but creating summaries for aggregates that are sensitive to duplicates (like Uniform sample, Median, Avg, Count, Sum) is difficult to design.

In the distribution phase, a node u in ring i now monitors the frequency node from every node n_{i-1} in ring $i - 1$, including its own information, throughout the recent epochs. Node u verifies in case its data is combined with the data sent by any node in ring $i - 1$. If the value of node is modest, you should attempt to locate a more suitable ring so as to increase the amount of its own data that may be encompassed in future broadcasts. Additionally, rings i, $i + 1$, $i - 2$, and $i + 2$ may be included in the data aggregation process. It is worth perceiving that ring $i - 2$ and $i + 2$ may be captured even when there is mobility. In order to facilitate these verifications, the header of each packet contains a comprehensive list of all node IDs involved in the creation of the summary, which is the aggregated data result. This functionality is used at every node as a kind of implicit recognition. Ultimately, the choice of which ring to join is determined based on informative that rely on the values of n_{i-1}, n_i, n_{i+2}, ni+1, and n_{i-2} [72]. The question accretion period is partitioned into epochs, with a single combined being generated at the conclusion of each epoch. Each epoch is separated into certain time slots, which are then used to plan the node spreads in a TDMA (Time Division Multiple Access) way. Sensors have the capability to enter a state of dormancy and be subsequently activated within their designated time windows for data transmission.

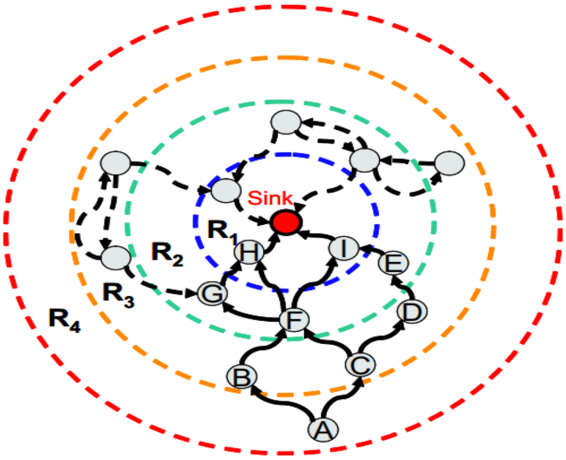


Fig 6. Various Aggregation Patterns Illustrated By A Ring Structure.

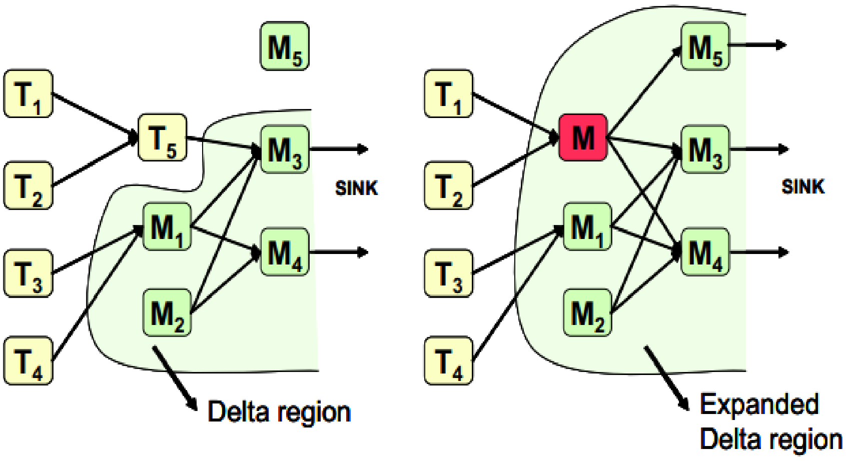


Fig 7. Tributary and Delta Data Collection Areas Illustrated.

The aggregation process begins at the outermost ring, denoted as Ri, and then progresses to the next ring, denoted as Ri−1, continuing in this manner until it reaches the sink, advancing stage by stage. In the instance shown in **Fig. 6**, the data produced at node A may be sent to the destination over seven distinct paths: $\{A, C, F, I, S\}$, $\{A, B, F, I, S\}$, $\{A, C, D, E, I, S\}$, $\{A, B, F, H, S\}$, $\{A, B, F, G, H, S\}$, $\{A, C, F, H, S\}$, and $\{A, C, G, H, S\}$. It is important to understand that with Synopsis Diffusion, data may travel along several pathways, which means that a node may get numerous copies of the same information. This might potentially impact the outcome of the aggregate, particularly when the aggregation algorithms are sensitive to duplicates. Culhane et al. [73] address this issue by suggesting appropriate aggregating purposes and data edifices. Multi-path methods are advantageous for systems that experience regular packet losses caused by agility or conduit defects. The additional overhead in the form of duplicates compensates for these losses and enhances the network's resiliency. In the event of a connection letdown, the data may still be successfully sent to the sink by using an alternative route.

Hybrid Data Aggregation Methods

To fully use the benefits of both multiple path and tree-based systems, it is feasible to develop hybrid techniques that dynamically adjust their data aggregation edifice to achieve optimum presentation [74]. The following procedure is described.

Deltas and Tributaries [75] - The Deltas and Tributaries procedure aims to address the limitations of both tree and multiple-path established architectures by integrating the most advantageous characteristics of both approaches. The outcome is a hybrid procedure in which information accretion edifices may operate concurrently in various parts of the system. Under conditions of minimal package loss, data aggregation tree is the optimal edifice due to its ability to implement efficient sleeping modes (as discussed in earlier sections) and its effectiveness in representing and compressing data. Alternatively, when experiencing high rates of data loss or when transferring aggregated partial results from several sensor interpretations, using a multiple path strategy may be the most suitable choice owing to its enhanced resilience.

Therefore, nodes are classified into two distinct groups: nodes that use a tree-based method for packet forwarding (referred to as T nodes) and nodes that utilize a multiple path system (known as M nodes). This indicates that the system is structured into sections that apply one of two strategies. The primary challenge is in establishing connections across areas that use distinct data gathering frameworks. To do this, it is necessary to adhere to the following guidelines:

- *Edge Accuracy:* An edge that starts from a M node can never be connected to a T node. It signifies that the outcome of combining data in an area with several paths may only be received by a certain node called the M node (refer to **Fig. 7**).
- *Path Correctness:* A subgraph having M nodes, out of which one is the sink node, is generated. This subgraph is provided by trees that are constructed from T nodes (refer to **Fig. 7**).

VI. CONCLUSION

In-network aggregation (INA) is a basic design paradigm that helps to optimize data transfer and resource usage in multihop networks, particularly WSNs. The results described in this work have provided a comprehensive view of the complexities of INA and the effects of the data depiction, aggregation functions, and routing protocols on the network's performance and efficiency. Routing protocols are vital INA in the determination of data packets' flow in the system through complex routing algorithms that take into consideration the content of the data packages. This assist in assembling data and also assist in the conservation of energy. Other sophisticated approaches that are category based and tree based and clustering based are other forms of scalable approaches to combine nodes and data. An analysis of their features shows the strengths and weaknesses of each type. Also, the aggregation functions are very beneficial in the removal of data at the middle nodes and also in the size of the data packets. This can be done through the use of various methods for aggregation such as the lossy and lossless types. Some of these methods like the accuracy of the data that has to be transmitted together with the time taken in transmitting the data. There are several methods of data encoding, which may allow to manage data flood and to store and transmit data effectively in the conditions of limited networks. Distributed source coding strategies provide options for representing data in a concise form as well as the ability to expand it. We have also attempted, by examining the interrelatedness of these components, to identify potential enhancement in the utilization of resources and in the consolidation of the network. This lays the groundwork to construct better and more robust topologies that are essential for solving new problems and applying in-network aggregation in various situations.

CRediT Author Statement

The authors confirm contribution to the paper as follows:

Conceptualization: Fabio Caccioli Capra and Vincenzo Anselmi; **Methodology:** Fabio Caccioli Capra and Vincenzo Anselmi; **Data Curation:** Vincenzo Anselmi; **Writing-Original Draft Preparation:** Vincenzo Anselmi; **Visualization:** Vincenzo Anselmi; **Investigation:** Fabio Caccioli Capra and Vincenzo Anselmi; **Supervision:** Vincenzo Anselmi; **Validation:** Fabio Caccioli Capra and Vincenzo Anselmi; **Writing-Reviewing and Editing:** Fabio Caccioli Capra and Vincenzo Anselmi; All authors reviewed the results and approved the final version of the manuscript.

Data Availability

No data was used to support this study.

Conflicts of Interests

The authors declare no conflict of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests.

References

[1]. Z. Yang and Y. Liu, "Understanding Node Localizability of Wireless Ad Hoc and Sensor Networks," IEEE Transactions on Mobile Computing, vol. 11, no. 8, pp. 1249–1260, Aug. 2012, doi: 10.1109/tmc.2011.122.

[2]. A. Sebaa and A. Tari, "Query optimization in cloud environments: challenges, taxonomy, and techniques," The Journal of Supercomputing, vol. 75, no. 8, pp. 5420–5450, Mar. 2019, doi: 10.1007/s11227-019-02806-9.

[3]. I. Solis and K. Obraczka, "In-network aggregation trade-offs for data collection in wireless sensor networks," International Journal of Sensor Networks, vol. 1, no. 3/4, p. 200, 2006, doi: 10.1504/ijsnet.2006.012035.

[4]. A. Deligiannakis, Y. Kotidis, and N. Roussopoulos, "Hierarchical In-Network Data Aggregation with Quality Guarantees," Advances in Database Technology - EDBT 2004, pp. 658–675, 2004, doi: 10.1007/978-3-540-24741-8_38.

[5]. I. Solis and K. Obraczka, "The impact of timing in data aggregation for sensor networks," 2004 IEEE International Conference on Communications (IEEE Cat. No.04CH37577), pp. 3640-3645 Vol.6, 2004, doi: 10.1109/icc.2004.1313222.

[6]. Y. Yang, P. Grover, and S. Kar, "Rate Distortion for Lossy In-Network Linear Function Computation and Consensus: Distortion Accumulation and Sequential Reverse Water-Filling," IEEE Transactions on Information Theory, vol. 63, no. 8, pp. 5179–5206, Aug. 2017, doi: 10.1109/tit.2017.2710059.

[7]. Y. Yang, M. I. Fonoage, and M. Cardei, "Improving network lifetime with mobile wireless sensor networks," Computer Communications, vol. 33, no. 4, pp. 409–419, Mar. 2010, doi: 10.1016/j.comcom.2009.11.010.

[8]. N. Shabbir and S. R. Hassan, "Routing Protocols for Wireless Sensor Networks (WSNs)," Wireless Sensor Networks - Insights and Innovations, Oct. 2017, doi: 10.5772/intechopen.70208.

[9]. R. E. Mohamed, A. I. Saleh, M. Abdelrazzak, and A. S. Samra, “Survey on Wireless Sensor Network Applications and Energy Efficient Routing Protocols,” Wireless Personal Communications, vol. 101, no. 2, pp. 1019–1055, May 2018, doi: 10.1007/s11277-018-5747-9.

[10]. S. Peter, K. Piotrowski, and P. Langendorfer, “In-network-aggregation as case study for a support tool reducing the complexity of designing secure wireless sensor networks,” 2008 33rd IEEE Conference on Local Computer Networks (LCN), pp. 778–785, Oct. 2008, doi: 10.1109/lcn.2008.4664280.

[11]. D. Westhoff and O. Ugus, “Malleability resilient (premium) Concealed Data Aggregation,” 2013 IEEE 14th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM), pp. 1–6, Jun. 2013, doi: 10.1109/wowmom.2013.6583470.

[12]. S. Peter, D. Westhoff, and C. Castelluccia, “A Survey on the Encryption of Convergecast Traffic with In-Network Processing,” IEEE Transactions on Dependable and Secure Computing, vol. 7, no. 1, pp. 20–34, Jan. 2010, doi: 10.1109/tdsc.2008.23.

[13]. J. Jose, J. Jose, and Muhammed Ilyas H, “Symmetric concealed data aggregation techniques in wireless sensor networks using Privacy Homomorphism: A review,” 2016 International Conference on Information Science (ICIS), pp. 275–280, Aug. 2016, doi: 10.1109/infosci.2016.7845340.

[14]. O. R. Merad Boudia, S. M. Senouci, and M. Feham, “Secure and efficient verification for data aggregation in wireless sensor networks,” International Journal of Network Management, vol. 28, no. 1, Sep. 2017, doi: 10.1002/nem.2000.

[15]. Y. Wang, J. Min, and Y. Chen, “Impact of the hybrid gain ensemble data assimilation on meso-scale numerical weather prediction over east China,” Atmospheric Research, vol. 206, pp. 30–45, Jul. 2018, doi: 10.1016/j.atmosres.2018.02.014.

[16]. E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, “In-network aggregation techniques for wireless sensor networks: a survey,” IEEE Wireless Communications, vol. 14, no. 2, pp. 70–87, Apr. 2007, doi: 10.1109/mwc.2007.358967.

[17]. A. Raja Basha, “Energy efficient aggregation technique-based realisable secure aware routing protocol for wireless sensor network,” IET Wireless Sensor Systems, vol. 10, no. 4, pp. 166–174, Aug. 2020, doi: 10.1049/iet-wss.2019.0178.

[18]. N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, “Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey,” IEEE Communications Surveys & Tutorials, vol. 15, no. 2, pp. 551–591, 2013, doi: 10.1109/surv.2012.062612.00084.

[19]. A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, “Routing protocols in ad hoc networks: A survey,” Computer Networks, vol. 55, no. 13, pp. 3032–3080, Sep. 2011, doi: 10.1016/j.comnet.2011.05.010.

[20]. G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, and R. Jafari, “Enabling Effective Programming and Flexible Management of Efficient Body Sensor Network Applications,” IEEE Transactions on Human-Machine Systems, vol. 43, no. 1, pp. 115–133, Jan. 2013, doi: 10.1109/tsmcc.2012.2215852.

[21]. M. R. Jakobsen and K. Hornbaek, “Interactive Visualizations on Large and Small Displays: The Interrelation of Display Size, Information Space, and Scale,” IEEE Transactions on Visualization and Computer Graphics, vol. 19, no. 12, pp. 2336–2345, Dec. 2013, doi: 10.1109/tvcg.2013.170.

[22]. G. Anastasi, M. Conti, and M. Di Francesco, “Extending the Lifetime of Wireless Sensor Networks Through Adaptive Sleep,” IEEE Transactions on Industrial Informatics, vol. 5, no. 3, pp. 351–365, Aug. 2009, doi: 10.1109/tii.2009.2025863.

[23]. H. Tang, G. Liu, X. Chen, and N. Xiong, “A Survey on Steiner Tree Construction and Global Routing for VLSI Design,” IEEE Access, vol. 8, pp. 68593–68622, 2020, doi: 10.1109/access.2020.2986138.

[24]. R. A. Guerin and A. Orda, “QoS routing in networks with inaccurate information: theory and algorithms,” IEEE/ACM Transactions on Networking, vol. 7, no. 3, pp. 350–364, Jun. 1999, doi: 10.1109/90.779203.

[25]. D. G. Leibovici, “Defining Spatial Entropy from Multivariate Distributions of Co-occurrences,” Spatial Information Theory, pp. 392–404, 2009, doi: 10.1007/978-3-642-03832-7_24.

[26]. X. J. Luo and L. O. Oyedele, “Forecasting building energy consumption: Adaptive long-short term memory neural networks driven by genetic algorithm,” Advanced Engineering Informatics, vol. 50, p. 101357, Oct. 2021, doi: 10.1016/j.aei.2021.101357.

[27]. K. Lakshmanna, N. Subramani, Y. Alotaibi, S. Alghamdi, O. I. Khalafand, and A. K. Nanda, “Improved Metaheuristic-Driven Energy-Aware Cluster-Based Routing Scheme for IoT-Assisted Wireless Sensor Networks,” Sustainability, vol. 14, no. 13, p. 7712, Jun. 2022, doi: 10.3390/su14137712.

[28]. D. Bilò, L. Gualà, and G. Proietti, “A Faster Computation of All the Best Swap Edges of a Shortest Paths Tree,” Algorithmica, vol. 73, no. 3, pp. 547–570, Jul. 2014, doi: 10.1007/s00453-014-9912-6.

[29]. D. R. Godschalk, A. Rose, E. Mittler, K. Porter, and C. T. West, “Estimating the value of foresight: aggregate analysis of natural hazard mitigation benefits and costs,” Journal of Environmental Planning and Management, vol. 52, no. 6, pp. 739–756, Aug. 2009, doi: 10.1080/09640560903083715.

[30]. E. L. Boone and B. P. Bullock, “Spatial correlation matrix selection using Bayesian model averaging to characterize inter-tree competition in loblolly pine trees,” Journal of Applied Statistics, vol. 35, no. 9, pp. 967–977, Aug. 2008, doi: 10.1080/02664760802185845.

[31]. R.-J. Liu and J. Brookfield, “Stars, Rings and Tiers: Organisational Networks and Their Dynamics in Taiwan’s Machine Tool Industry,” Long Range Planning, vol. 33, no. 3, pp. 322–348, Jun. 2000, doi: 10.1016/s0024-6301(00)00039-x.

[32]. B. Zarebavani, K. Cheshmi, B. Liu, M. M. Strout, and M. M. Dehnavi, “HDagg: Hybrid Aggregation of Loop-carried Dependence Iterations in Sparse Matrix Computations,” 2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS), pp. 1217–1227, May 2022, doi: 10.1109/ipdps53621.2022.00121.

[33]. P. Narvaez, Kai-Yeung Siu, and Hong-Yi Tzeng, “New dynamic algorithms for shortest path tree computation,” IEEE/ACM Transactions on Networking, vol. 8, no. 6, pp. 734–746, 2000, doi: 10.1109/90.893870.

[34]. V. K. Quy, V. H. Nam, D. M. Linh, and L. A. Ngoc, “Routing Algorithms for MANET-IoT Networks: A Comprehensive Survey,” Wireless Personal Communications, vol. 125, no. 4, pp. 3501–3525, May 2022, doi: 10.1007/s11277-022-09722-x.

[35]. M.-C. Lee and J.-P. Sheu, “An efficient routing algorithm based on segment routing in software-defined networking,” Computer Networks, vol. 103, pp. 44–55, Jul. 2016, doi: 10.1016/j.comnet.2016.03.017.

[36]. S. Zou, I. Nikolaidis, and J. J. Harms, “Efficient Data Collection Trees in Sensor Networks with Redundancy Removal,” Ad-Hoc, Mobile, and Wireless Networks, pp. 252–265, 2004, doi: 10.1007/978-3-540-28634-9_20.

[37]. R. Lachowski, M. Pellenz, M. Penna, E. Jamhour, and R. Souza, “An Efficient Distributed Algorithm for Constructing Spanning Trees in Wireless Sensor Networks,” Sensors, vol. 15, no. 1, pp. 1518–1536, Jan. 2015, doi: 10.3390/s150101518.

[38]. J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim, “Energy Efficient Routing Algorithm with Mobile Sink Support for Wireless Sensor Networks,” Sensors, vol. 19, no. 7, p. 1494, Mar. 2019, doi: 10.3390/s19071494.

[39]. S. Azar, A. Avokh, J. Abouei, and K. N. Plataniotis, “Energy- and Delay-Efficient Algorithm for Large-Scale Data Collection in Mobile-Sink WSNs,” IEEE Sensors Journal, vol. 22, no. 7, pp. 7324–7339, Apr. 2022, doi: 10.1109/jsen.2022.3152180.

[40]. P. Sreedevi and S. Venkateswarlu, “An Efficient Intra-Cluster Data Aggregation and finding the Best Sink location in WSN using EEC-MA-PSOGA approach,” International Journal of Communication Systems, vol. 35, no. 8, Feb. 2022, doi: 10.1002/dac.5110.

[41]. H. Al-Mahdi, M. Elshrkawey, S. Saad, and S. Abdelaziz, “An Intelligent Energy-Efficient Data Routing Scheme for Wireless Sensor Networks Utilizing Mobile Sink,” Wireless Communications and Mobile Computing, vol. 2024, pp. 1–20, Mar. 2024, doi: 10.1155/2024/7384537.

[42]. Y. Xue, Y. Cui, and K. Nahrstedt, “Maximizing Lifetime for Data Aggregation in Wireless Sensor Networks,” Mobile Networks and Applications, vol. 10, no. 6, pp. 853–864, Oct. 2005, doi: 10.1007/s11036-005-4443-7.

[43]. K. Akkaya, M. Demirbas, and R. S. Aygun, “The impact of data aggregation on the performance of wireless sensor networks,” Wireless Communications and Mobile Computing, vol. 8, no. 2, pp. 171–193, Sep. 2006, doi: 10.1002/wcm.454.

[44]. M. Kumar, P. Mukherjee, K. Verma, S. Verma, and D. B. Rawat, "Improved Deep Convolutional Neural Network Based Malicious Node Detection and Energy-Efficient Data Transmission in Wireless Sensor Networks," IEEE Transactions on Network Science and Engineering, vol. 9, no. 5, pp. 3272–3281, Sep. 2022, doi: 10.1109/tnse.2021.3098011.

[45]. H. Artail, A. Ajami, T. Saouma, and M. Charaf, "A faulty node detection scheme for wireless sensor networks that use data aggregation for transport," Wireless Communications and Mobile Computing, vol. 16, no. 14, pp. 1956–1971, Jan. 2016, doi: 10.1002/wcm.2661.

[46]. W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks," IEEE/ACM Transactions on Networking, vol. 12, no. 3, pp. 493–506, Jun. 2004, doi: 10.1109/tnet.2004.828953.

[47]. P. V. Vinh and H. Oh, "An Optimized Approach for Time-Constrained and Reliable Bursty Data Acquisition in WMSNs," International Journal of Distributed Sensor Networks, vol. 11, no. 6, p. 826937, Jun. 2015, doi: 10.1155/2015/826937.

[48]. A. Cimitile and G. Visaggio, "Software salvaging and the call dominance tree," Journal of Systems and Software, vol. 28, no. 2, pp. 117–127, Feb. 1995, doi: 10.1016/0164-1212(94)00049-s.

[49]. S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TAG," ACM SIGOPS Operating Systems Review, vol. 36, no. SI, pp. 131–146, Dec. 2002, doi: 10.1145/844128.844142.

[50]. A. Brayner, A. Lopes, D. Meira, R. Vasconcelos, and R. Menezes, "An adaptive in-network aggregation operator for query processing in wireless sensor networks," Journal of Systems and Software, vol. 81, no. 3, pp. 328–342, Mar. 2008, doi: 10.1016/j.jss.2007.06.021.

[51]. C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," IEEE/ACM Transactions on Networking, vol. 11, no. 1, pp. 2–16, Feb. 2003, doi: 10.1109/tnet.2002.808417.

[52]. E. Shakshuki, H. Malik, and M. K. Denko, "Software agent-based directed diffusion in wireless sensor network," Telecommunication Systems, vol. 38, no. 3–4, pp. 161–174, Apr. 2008, doi: 10.1007/s11235-008-9102-4.

[53]. S. Peng and C. P. Low, "Energy neutral directed diffusion for energy harvesting wireless sensor networks," Computer Communications, vol. 63, pp. 40–52, Jun. 2015, doi: 10.1016/j.comcom.2015.02.017.

[54]. M. Chen, T. Kwon, Y. Yuan, Y. Choi, and V. C. M. Leung, "Mobile Agent-Based Directed Diffusion in Wireless Sensor Networks," EURASIP Journal on Advances in Signal Processing, vol. 2007, no. 1, Oct. 2006, doi: 10.1155/2007/36871.

[55]. Xie Meihua and Wang Zhengming, "Fingerprint Enhancement Based on Edge-Directed Diffusion," Third International Conference on Image and Graphics (ICIG'04), pp. 274–277, doi: 10.1109/icig.2004.68.

[56]. A. M. Khedr, A. Aziz, and W. Osamy, "Successors of PEGASIS protocol: A comprehensive survey," Computer Science Review, vol. 39, p. 100368, Feb. 2021, doi: 10.1016/j.cosrev.2021.100368.

[57]. V. Katiyar, N. Chand, Gopal Chand Gautam, and A. Kumar, "Improvement in LEACH protocol for large-scale wireless sensor networks," 2011 International Conference on Emerging Trends in Electrical and Computer Technology, pp. 1070–1075, Mar. 2011, doi: 10.1109/icetect.2011.5760277.

[58]. A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015, doi: 10.1109/comst.2015.2444095.

[59]. H. Jeong, M.-G. Kim, and H.-S. Park, "A Data Profile-Based Contention-Free MAC Protocol for the Patient Monitoring Systems on WBANs," Global Trends in Computing and Communication Systems, pp. 192–201, 2012, doi: 10.1007/978-3-642-29219-4_23.

[60]. S. J. Begum and T. Purusothaman, "Hierarchical Tree Structure Based Clustering Schemes for Secure Group Communication," Mobile Networks and Applications, vol. 21, no. 3, pp. 550–560, Sep. 2015, doi: 10.1007/s11036-015-0649-5.

[61]. H. M. N. D. Bandara, A. P. Jayasumana, and T. H. Illangasekare, "A Top-Down Clustering and Cluster-Tree-Based Routing Scheme for Wireless Sensor Networks," International Journal of Distributed Sensor Networks, vol. 7, no. 1, p. 940751, Jan. 2011, doi: 10.1155/2011/940751.

[62]. R. Velmani and B. Kaarthick, "An Efficient Cluster-Tree Based Data Collection Scheme for Large Mobile Wireless Sensor Networks," IEEE Sensors Journal, vol. 15, no. 4, pp. 2377–2390, Apr. 2015, doi: 10.1109/jsen.2014.2377200.

[63]. G. Chalhoub and M. Misson, "Cluster-tree based energy efficient protocol for wireless sensor networks," 2010 International Conference on Networking, Sensing and Control (ICNSC), pp. 664–669, Apr. 2010, doi: 10.1109/icnsc.2010.5461577.

[64]. M. J. Handy, M. Haase, and D. Timmermann, "Low energy adaptive clustering hierarchy with deterministic cluster-head selection," 4th International Workshop on Mobile and Wireless Communications Network, doi: 10.1109/mwcn.2002.1045790.

[65]. D. Chu, A. Deshpande, J. M. Hellerstein, and Wei Hong, "Approximate Data Collection in Sensor Networks using Probabilistic Models," 22nd International Conference on Data Engineering (ICDE'06), pp. 48–48, 2006, doi: 10.1109/icde.2006.21.

[66]. M. Bagaa, M. Younis, A. Ouadjaout, and N. Badache, "Efficient multi-path data aggregation scheduling in wireless sensor networks," 2013 IEEE International Conference on Communications (ICC), pp. 1560–1564, Jun. 2013, doi: 10.1109/icc.2013.6654736.

[67]. S. S. Sruthi and G. Geethakumari, "An Efficient Secure Data Aggregation Technique for Internet of Things Network: An Integrated Approach Using DB-MAC and Multi-path Topology," 2016 IEEE 6th International Conference on Advanced Computing (IACC), pp. 599–603, Feb. 2016, doi: 10.1109/iacc.2016.116.

[68]. D. Messina, M. Ortolani, and G. L. Re, "A Network Protocol to Enhance Robustness in Tree-Based WSNs Using Data Aggregation," 2007 IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems, pp. 1–4, Oct. 2007, doi: 10.1109/mobhoc.2007.4428635.

[69]. A. Berfield, P. K. Chrysanthis, and D. Mosse, "LSynD: Localized Synopsis Diffusion," IEEE, May 2007, doi: 10.1109/isorc.2007.41.

[70]. A. P. Santhanam, B. C. Neyveli, and M. Chatterjee, "Traffic diffusion analysis for adaptive multi-path routing algorithm in sensor networks," GLOBECOM '05. IEEE Global Telecommunications Conference, 2005., pp. 5 pp. – 3101, 2005, doi: 10.1109/glocom.2005.1578327.

[71]. V. Kulathumani, M. Nakagawa, and A. Arora, "EZ-AG: structure-free data aggregation in MANETs using push-assisted self-repelling random walks," Journal of Internet Services and Applications, vol. 9, no. 1, Mar. 2018, doi: 10.1186/s13174-018-0077-4.

[72]. M. Shi, X. Jiang, Y. Hu, L. Ling, and X. Wang, "An improved meta-heuristic algorithm for developing high-quality ReaxFF force fields of Fe/Ni transition metals and alloys," Computational Materials Science, vol. 221, p. 112083, Mar. 2023, doi: 10.1016/j.commatsci.2023.112083.

[73]. W. Culhane, K. Kogan, C. Jayalath, and P. Eugster, "Optimal communication structures for big data aggregation," 2015 IEEE Conference on Computer Communications (INFOCOM), pp. 1643–1651, Apr. 2015, doi: 10.1109/infocom.2015.7218544.

[74]. M. Naghibi and H. Barati, "SHSDA: secure hybrid structure data aggregation method in wireless sensor networks," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 12, pp. 10769–10788, Jan. 2021, doi: 10.1007/s12652-020-02751-z.

[75]. M. R. Baerwald and B. May, "Characterization of microsatellite loci for five members of the minnow family Cyprinidae found in the Sacramento–San Joaquin Delta and its tributaries," Molecular Ecology Notes, vol. 4, no. 3, pp. 385–390, Jun. 2004, doi: 10.1111/j.1471-8286.2004.00661.x.

ISSN: 3080-7484