Multicast Scaling in Heterogeneous Wireless Sensor Networks for Security and Time Efficiency

Ramdas Vankdothu (vramdas786sap@gmail.com)
Osmania University University College of Engineering

Hameed Mohd Abdul
Osmania University University College of Engineering

Fatima Husnah
Osmania University

Subbarao Akkala
Maulana Azad National Institute of Technology

Research Article

Keywords: Trust validation, Delay measures, Congestion level measure, Clustering, Routing, and Optimization.

Posted Date: December 29th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-835963/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
Multicast Scaling in Heterogeneous Wireless Sensor Networks for Security and Time Efficiency

A. Ramdas Vankdothu, B. Mohd Abdul Hameed, C. Husnah Fatima, D. Akkala Subbarao

Abstract— Heterogeneous wireless sensor networks (HWSNs) satisfy researchers' requirements for developing real-world solutions that handle unattended challenges. However, the primary constraint of researchers is the privacy of the sensor nodes. It safeguards the sensor nodes and extensions in the HWSNs. Therefore, it is necessary to develop secure operational systems. Multicast scaling with security and time efficiency is described in heterogeneous wireless sensor networks to maximize network performance while also successfully protecting network privacy. This study evaluates the initial security and time efficiency measures, such as execution time, transmission delay, processing delay, congestion level, and trust measure. Subsequently, the optimal location of the heterogeneous nodes is determined using sigmoid-based fuzzy c-means clustering. Finally, successful cluster routing was achieved via support-value-based particle swarm optimization. The experimental results indicate that the proposed strategy surpasses existing strategies in terms of network delivery ratio, end-to-end delay, throughput, packet delivery, and node remaining energy level.

Keywords—Trust validation, Delay measures, Congestion level measure, Clustering, Routing, and Optimization.

1. INTRODUCTION

The advancements in wireless communication technology and low-power digital circuits in recent years, Wireless sensor networks (WSNs) have found widespread use in a range of industries, including military surveillance, target tracking humanitarian assistance, smart home, urban management [1, 2]. A wireless sensor network (WSN) comprises a base station (BS) and an undetermined number of sensor nodes. The sensor node receives the majority of its power from the battery. Battery capacity is limited, and charging and replacing the battery becomes more challenging when sensor nodes are placed in a complex or harsh environment [3, 4]. As a result, the most critical difficulty in developing WSN routing protocols is energy efficiency [5–7]. Heterogeneous wireless sensor networks with varying initial energies directly and practically impact network energy consumption and lifespan [8, 9]. Sensors with different sensing, power, processing, and communication capabilities comprise heterogeneous WSNs. By leveraging the small-world characteristics of sensor networks, deploying heterogeneous nodes in densely populated areas that can communicate directly with sink nodes, and forming super link heterogeneous sensor networks [10, 11] can achieve a shorter average path length while maintaining a higher clustering coefficient, allowing them to adjust their communication distance by adjusting the clustering coefficient.

Clustering is the most energy-efficient and longest-lived technique for building a routing protocol for heterogeneous WSNs [14, 15]. Clustering is a technique for dividing sensor nodes into numerous locations, referred to as clusters. A cluster head (CH) leads to each cluster, with the remaining sensor nodes labeled as cluster members. Cluster members convey their perceived data to the cluster's nearest CH, which collects and transmits it to the BS. According to their different distances, the CH carries data packets to the BS in a single-hop or multi-hop method [16].

Numerous clustering-based routing protocols have been proposed to extend the life of networks, including hybrid energy-efficient distributed clustering (HEED), threshold sensitive energy-efficient sensor network protocol (TEEN), low-energy adaptive clustering hierarchy (LEACH), and LEACH-centralized (LEACH-C) [17]. (BLAC). However, the majority of methods for WSN clustering routing use homogeneous networks [18]. In practice, when node resources and topology change, heterogeneous wireless sensor networks (HWSNs) are gaining traction. Energy heterogeneity is a critical issue in HWSNs. Stable election protocols (SEPs), modified election protocols (M-SEPs), and protracted election protocols (P-SEP) have been created [19], as has an improved version of the energy-aware distributed unequal clustering protocol (improved-EADUC) [20]. DEEC has been proposed as a routing technique for HWSNs with heterogeneous node energy [21]. However, a significant challenge for the HWSN routing protocol [21] is maximizing the heterogeneity of node energy to extend the network's life and performance.

A. Ramdas Vankdothu is pursuing his doctoral degree in Computer Science & Engineering at Osmania University Hyderabad, India (e-mail: vramdas786sap@gmail.com).
B. Mohd Abdul Hameed, Assistant Professor in Department of Computer Science & Engineering University College of Engineering (A). Osmania University Hyderabad, India.
C. Husnah Fatima, Assistant Professor, Department of Electronics & Communication Engineering, Nawab Shah Alam Khan College of Engineering and Technology, Osmania University Hyderabad, Telangana, India.
D. Akkala Subbarao, Assistant Professor in the Department of Electronics and Communication Engineering in Maulana Azad National Institute of Technology, Bhopal, India.
Corresponding mail id: vramdas786sap@gmail.com
The following section outlines the essential contributions of this study.

- Security and time efficiency measures, such as execution time, transmission delay, processing delay, congestion level, and trust measure are examined for heterogeneous WSNs.
- The optimal location of the heterogeneous nodes is calculated using the fuzzy c-means clustering technique based on the sigmoid.
- The ideal location of heterogeneous nodes is determined using the sigmoid-based fuzzy c-means clustering algorithm.
- Effective cluster routing is accomplished using support-value-based particle swarm optimization.

The structure of the manuscript is as follows: Section 2 reviews the literature on the proposed technique. Section 3 summarizes the planned strategy, Section 4 reports the findings of the examination, and Section 5 concludes the paper.

2. RELATED WORK

Xiu-wu et al. [22] introduced a clustering routing method for heterogeneous wireless sensor networks (CHRA) with the goal of balancing energy consumption and extending the lifetime of the network. By identifying the appropriate number of heterogeneous nodes for each round, the proposed routing algorithm selects the cluster heads. There are two types of common nodes in CHRA. The first group includes nodes that transmit data to a sink via a heterogeneous node, whereas the second category includes all the others. Additionally, LEACH-C grouped common nodes in each round. According to the numerical results and performance evaluations, the proposed routing algorithm can significantly increase the network's longevity and stability.

SonamMaurya et al. [23] introduced the delay-aware energy-efficient reliable routing (DA-EERR) as a game-changing data transfer technique for various sensor systems. The proposed approach imposed a search space constraint on the transmission of time-sensitive data to ensure timely transmission. Additionally, an algorithm was developed to calculate the ideal energy-delay routing between the source and sink within the supplied search area, allowing for rapid data transfer via an energy-efficient hop. By employing data aggregation and providing suitable load balancing across the network, the proposed DA-EERR technique increases the proportion of adequately received data packets at the sink in large dense networks.

Sahoo et al. [24] proposed combining a particle swarm optimization (PSO) algorithm with an energy-efficient clustering and sink mobility strategy (PSO-ECSM) to address both cluster head selection and sink mobility issues.

The PSO performance ECSMs were determined after extensive computer simulations. Five variables were considered when choosing a CH: residual energy, distance, the degree of a node, average energy, and rate of energy consumption (ECR). The ideal values for these variables were determined using the PSO-ECSM algorithm. Additionally, PSO-ECSM integrates sink mobility to address the issue of data traffic transmission across numerous hops in a multi-hop network.

Sandeep Verma et al. [25] devised a genetic algorithm-based optimized clustering (GAOC) technique for CH selection optimization that incorporated residual energy, distance to the sink, and node density into the fitness function. Additionally, a GAOC with multiple data sinks (MS-GAOC) was developed to alleviate the hot-spot issue and shorten the distance between the nodes and the sink. MS-GAOC empirical research was conducted using protocols that addressed a variety of data sources to ensure a balanced comparison analysis. According to simulation testing, the GAOC and MS-GAOC protocols surpass existing protocols on a variety of performance parameters, including the stability period, network lifetime, number of dead nodes in each round, throughput, and residual energy of the network.

Muhammad Aslam et al. [26] asserted the following: two approaches to energy-efficient path planning are presented for three layers of heterogeneous WSNs: centralized energy efficient clustering with two-hop heterogeneity awareness (ACEEC) and advanced heterogeneity-aware path planning (AHAP). Clustering for centralized energy efficiency (ACEEC) (THCEEC). These systems are based on two concepts: centralized energy-efficient clustering, which is used to regulate network deployment variations, and wireless sensor networks, which make use of transmission ranges (CEEC). The BS determines the cluster heads that are appropriate depending on the initial energy, regional flag, residual energy, and distance to the BS (CHs).

3. PROPOSED METHODOLOGY

This work proposed adequate security and time-efficiency multicast scaling in heterogeneous wireless sensor networks. To achieve this, various security and time measures were evaluated. Subsequently, sigmoid-based fuzzy c-means clustering was used to optimize heterogeneous node deployment in WSNs. Finally, the support-value-based particle swarm optimization algorithm was used to provide effective multicast routing. The proposed methodology is depicted in Figure 1 as a block diagram.
1.1 Security and Time Efficiency Measures

Efficiency and security in time. The multicast scaling approach is offered for naturally transforming the required assets to the application in demand in heterogeneous wireless sensor networks. Multicast scaling is performed using execution time, transmission delay, processing delay, and trust validation, as well as the organization of expected resources via security and time efficiency processing.

1.1.1 Execution Time Calculation.

The execution time is the difference between the time required to complete and submit a specified number of tasks (1).

$$E_t = \sum_{i=1}^{n} \left( \frac{T_{c,i} - T_{s,i}}{I} \right)$$  \hspace{1cm} (1)

where $E_t$ denotes the execution time, $T_{c,i}$ denotes the task completion time nodes, $T_{s,i}$ denotes the task submission time of nodes, and denotes the number of nodes processed in multicasting.

1.1.2 Congestion Level Factor.

The multiple nodes connecting the source to the sink are generated first, and then the cross-layer data are consumed as a state frame. This frame is sent upstream to update and communicate information about node congestion. The level of congestion at node is denoted by

$$C_i = \frac{T_r}{S_r}$$  \hspace{1cm} (2)

where $T_r$ is the input traffic rate, and $S_r$ is the service rate. A node's input traffic rate is defined as the number of packets flowing into the physical layer of the protocol stack in a unit of time. In addition, the service rate refers to the number of packets that travel downward to the channel over a specified time period.

1.1.3 Processing Delay.

The processing time required to complete a multicast operation depends on the volume of data traffic to be processed and the resources allocated to it. We expect that the processing delay $d_{k,l}$ of each multicast task $r_k f_l$ is proportional to the amount of traffic that it needs to process as
where $\alpha_i$ is a given corresponding element of the task $f_i$ and $b_k$ is the size of the data traffic of task $r_k$. The aggregate processing delay is caused by the traffic handling by network functions in $SC_k$ of $r_k$ accordingly condition (4),

$$d_k^p = \sum_{f_i \in SC_k} d_{k,f_i}^p$$  \hspace{1cm} (4)

1.1.4 Transmission Delay.

Let $P_k$ be the set of routing paths from source $s_k$ to destination in $D_k$, with every path $P_m \in P_k$ denoting a routing path from $s_k$ to destination $t_m \in D_k$. The transmission delay of each $r_k$ is the maximum end-to-end delay incurred in the paths $P_k$. Represent by $d_k'$ the transmission delay of the task $r_k$, which can be characterized as:

$$d_k' = \arg \max_{P_m \in P_k} \sum_{e \in P_m} d_e \cdot b_k$$  \hspace{1cm} (5)

The delay experienced by multicast tasks $r_k$ along these lines is

$$d_k = d_k^p + d_k'$$  \hspace{1cm} (6)

This should be no more prominent than its predetermined delay requirement as,

$$d_k \leq D_k$$  \hspace{1cm} (7)

1.1.5 Trust Calculation.

The level of consistency in the data is referred to as trust. Similarly, trust can be defined as a node's confidence in completing assigned tasks over time. This condition is used to determine the hub trust (8).

$$T_n = \frac{D_n^{\star \tau}}{D_n}$$  \hspace{1cm} (8)

where $T_n$ is the trust of node and $\tau$ is the discrete trust level, which is defined by condition (9).

$$D_n = \Pi(1 - \alpha_i)$$  \hspace{1cm} (9)

where $\tau$ is the trust of every task of node and $\alpha_i$ is determined by condition (10).

$$\alpha_i = \frac{1}{1 + t}$$  \hspace{1cm} (10)

Here, $t$ is the task-processing time for a specific node. Subsequently, the local gradient is estimated depending on the execution time, processing delay, trust, and transmission delay validations.

$$L_G = E_j \cdot d_k^p \cdot d_k' \cdot T_m \cdot C_j$$  \hspace{1cm} (11)

The local gradient is determined utilizing condition (11), and the nodes are deployed depending on the local gradient validation.

1.2 Optimal Deployment of Heterogeneous Nodes

Because the optimal location of heterogeneous energy nodes is equal to the sum of all common and sink nodes, the optimal location of heterogeneous energy nodes is equal to the sum of all common and sink nodes.

1.2.1 Cluster Head Selection.

Each node's cluster head selection factor determines the next-hop node, which is a function that involves three factors: the survivability factor $P_i$ of the path to the destination through that next-hop, the SINR value $\tilde{\theta}(e_i)$ of the link $e$ between the current node and the next-hop node, and the congestion level $C_j$ at the next hop. That is,

$$C_f = (\alpha \cdot L_G) + (\beta \cdot L_e) + (\gamma \cdot L_c)$$  \hspace{1cm} (12)

Here, $C_f$ denotes the cluster head selection factor, and $\alpha$, $\beta$, and $\gamma$ values are utilized for setting various weights on the local gradient $L_G$. Their values are defined by the requirement to force the strength of these three components throughout the
cluster head selection procedure. Each of the three weighting coefficients is treated similarly in our scenario $\alpha = \beta = \gamma = \frac{1}{3}$ to demonstrate the equivalent impact of all the components. The values were standardized with the end goal:

$$\alpha + \beta + \gamma = 1$$  \hspace{1cm} (13)

The cluster head selection factor specified in condition (13) is used as an input to the fuzzy c-means clustering algorithm based on sigmoids. This is accomplished effectively by including three aspects of the clustering of sensor node networks: latency, execution time, and trust.

1.2.2 SIGMOID Based Fuzzy C-Means Clustering Algorithm.

The node with the lowest delay value, execution time, and trust local gradient is transformed into a cluster head among the nodes of the system. The sensor nodes are clustered using the sigmoid-based fuzzy c-means (SFCM) clustering algorithm. Here, the support kernel values are confined using the deliberate cluster head selection factor in clustering. This algorithm starts with many initial cluster centers. The SFCM algorithm dispenses the information data of each class using fuzzy membership.

$$\tilde{J}_{\sigma n} = \sum_{l=1}^{L} \sum_{m=1}^{M} (v_l) \sqrt{\frac{C_{\beta} - q_m}{\sigma}}$$ \hspace{1cm} (14)

Condition (14) $C_{\beta}$ signifies that the cluster head value $q_m$ signifies the $m^{th}$ cluster center and signifies $n$ the constant esteem. The cluster head selection factor demonstrates the cluster head choosing factor in the cluster, referenced in condition (12). The membership function describes the probability that a pixel has a place with a particular cluster. The membership functions and cluster centers are updated by conditions (15) and (16).

$$v_{lm} = \frac{1}{\sum_{l=1}^{L} \sqrt{C_{\beta} - q_m}} \left( \frac{C_{\beta} - q_m}{\sigma_l} \right)^{n-1}$$ \hspace{1cm} (15)

The clusters centroid is processed by utilizing the condition (16),

$$z_{\beta} = \sum_{l=1}^{L} \sum_{m=1}^{M} v_{lm}^{-n}. C_{\beta}$$ \hspace{1cm} (16)

Calculate again until the coefficient change between the two cycles is close to $\psi$ the given limit.

$$\max_{lm} \left| \tilde{J}_{lm}^{(k)} - \tilde{J}_{lm}^{(k+1)} \right| < \psi$$ \hspace{1cm} (17)

In condition (8), $\psi$ is in the range of 0 to 1. The steps are repeated until effective clustering is achieved. The SFCM clustering is denoted by Algorithm 1.

**Algorithm 1: SFCM clustering Algorithm**

Begin
For $j = 1$ to $N$ do
Node $j$ is given the coefficient $v_l$ for being a member of the cluster $i$
End for
Repeat
For do
Compute the centroid of each cluster using condition (16)
End for
Repeat
Until the stopping condition reached
End

Once the clustering of nodes is completed, the nodes in the clusters are assumed to forward the packets.
1.3 Cluster Routing SPSO for Heterogeneous Sensor Networks

This section presents a mechanism for routing diverse energy networks that address the disadvantages of static networks. The algorithm for optimizing particle swarms with the help of support values is as follows: First, the network's heterogeneous nodes are counted and deployed optimally using the algorithm outlined in Section 3, the network's common nodes are then clustered using a sigmoid-based fuzzy c-means clustering algorithm, and cluster heads are selected. After clustering, member nodes submit data to the cluster head node, which processes both its own and received data and routes it to the sink node or associated heterogeneous node via the cluster head layer node's multiple hops.

1.3.1 Support value-based Particle swarm optimization (SPSO).

Particle swarm optimization (PSO) is a probabilistic, population-based pursuit strategy inspired by the behavior of bird flocks. The PSO algorithm maintains a particle swarm, each representing a potential solution. Particles exhibit a fundamental behavior: they mimic the accomplishments of neighboring particles and their victories. The particle's location is affected by the best particle in a neighborhood \( p'_{\text{best},j} \), just as the best arrangement is found by the particle \( g'_{\text{best}} \). The particle position \( y_j \) is balanced by utilizing the following condition:

\[
y_j(k' + 1) = y_j(k') + v_j(k' + 1)
\]

where the velocity component \( v_j \) represents the step size. The velocity is updated using condition (19).

\[
v_j(k' + 1) = w'v_j(k') + c_1r_1(p'_{\text{best},j} - x_j(k')) + c_2r_2(g'_{\text{best}} - x_j(k'))
\]

where \( w' \) is the inertia weight, \( c_1 \) and \( c_2 \) are the acceleration coefficients \( r_1, r_2 \in [0,1] \), \( p'_{\text{best},j} \) is the individual best position of a particle \( j \), and \( g'_{\text{best}} \) is the best position of the particles.

At that point, the location of each particle is transferred into the solution space and its fitness value is determined using the support value-based fitness function. In the meantime, updates \( p'_{\text{best},j} \) and \( g'_{\text{best}} \) positions are required if required. The support value is estimated by utilizing condition (20).

\[
\bar{S}_v = \frac{y_1 + y_2 + \ldots + y_n}{n}
\]

Here, \( \bar{S}_v \) denotes the support value and \( y_1, y_2, \ldots, y_n \) denotes the input population. This procedure is repeated until a requirement is satisfied, typically a sufficient fitness level or a maximum number of iterations. Algorithm 2 contains the pseudo-code for the support-value-based adaptive PSO algorithm.

---

**Algorithm 2: Support value based adaptive PSO Algorithm**

**Step 1:** Initialization
Set the generation number \( k' = 0 \)
Initialize a population of \( NP \) individuals
Initialize velocities \( v_j \) of the particles

**Step 2:**
While the stopping criterion is not satisfied
**Do**
**For** \( j = 1 \) to \( NP \)
**Step 3:** Calculate \( p'_{\text{best},j} \) and \( g'_{\text{best}} \) evaluate the fitness of the particles using (20).
**Step 4:** Update position and velocity
Calculate the positions and velocities of particles utilizing (18) and (19)
**End for**
**Step 5:** Increase the generation count
\( k' = k' + 1 \)
**End while**
This section describes a technique for safe clustered routing based on the SPSO algorithm. Assume that the total number of heterogeneous nodes in the sensor network equals, and that the total number of cluster heads of common nodes equals (20).

**Step 1:** The algorithm for the optimal placement of heterogeneous nodes presented in this section optimizes the placement of heterogeneous nodes.

**Step 2:** The SPSO method is used to choose cluster heads from common nodes.

**Step 3:** The member node communicates the gathered data to the cluster head during a specific time window, where it is processed.

**Step 4:** When the distance between the cluster head and the heterogeneous node is less than one hop, the data is transmitted directly to the heterogeneous node; when the distance is larger than one hop, the cluster head is routed to the next heterogeneous node via multiple hops of cluster head layer nodes.

**Step 5:** When the cluster head node's energy is depleted, the SPSO algorithm repeats the cluster head selection and clustering. The transmission mechanisms of the cluster nodes and CHs remain unchanged.

### 4. RESULTS AND DISCUSSION

The proposed multicast scaling technique was implemented in MATLAB's heterogeneous wireless sensor network working stage. To evaluate the performance of the proposed method, we compared the network throughput, packet delivery ratio, end-to-end delay, and residual energy level of nodes to those obtained using the uniform benchmark, sparse cluster regime, and cluster dense regime approaches.

1.4 **Packet Delivery Ratio**

The packet delivery ratio (PDR) is defined as the percentage of data packets that are correctly transported from the source to the destination, which can be expressed mathematically as

\[
\%PDR = \frac{n(P_T)}{n(P_T)} \times 100
\]

(21)

where \(n(P_T)\) is the number of packets transmitted, and \(n(P_R)\) is the number of packets received. The comparison graph for the packet delivery ratio is shown in figure 2.

![Comparison analysis of proposed packet delivery ratio](image)

**Figure 2:** Comparison analysis of proposed packet delivery ratio

Figure 2 present the outcome of packets arrival ratio for various techniques. The most secured proposed multicast routing sustains a constant high PDR for every case of the values of pause time.

1.5 **Throughput**

The network's throughput demonstrates its efficiency in transporting a significant amount of data from a source to a destination. The following equation can be used to numerically express throughput:

\[
Throughput = \frac{n(P_T)}{n(P_T) + n(P_{\text{lost}})}
\]

(22)

where \(n(P_T)\) is the total number of packets sent, \(n(P_{\text{lost}})\) and is the total number of packets sent. Figure 3 illustrates the throughput comparison graph.
The throughput value increased, while connectivity performed better. It is noted that the performance of the existing methods decreases significantly when the proposed technique improves.

### 1.6 End To End Delay

Throughput is defined theoretically as the total of the various delays contained in each data packet received by the destination node and the time required for sensor nodes to supply a data packet.

\[
E2E \text{Delay} = \frac{n(P_R T \_R - T_T)}{n(P_R)} \quad (23)
\]

where \( n(P_R) \) is the number of packets received, \( T_R \) is the period of the data packet obtained from the destination node, and \( T_T \) is the period of a data packet transmitted by the source node. A comparison graph for an end-to-end delay is shown in Figure 4.

As illustrated in Figure 4, the proposed technique results in a shorter end-to-end delay than the existing techniques. One of its characteristics is that it produces a small amount of data, resulting in a small amount of latency.

### 1.7 Remaining Energy

The remainder of the energy is used to extend the lifetime of the node. The system is more dependable if each node consumes a small amount of energy for data transport, and the remaining energy is distributed fairly evenly across the network. The following formula was used to determine the remaining energy:

\[
E_\text{r} = E_a - E_\text{r} \quad (24)
\]
where $E_R$ is the remaining energy, $E_{in}$ is the node’s initial energy, and $E_i^t$ is the amount of energy consumed by a node $i$ in time $t$. The residual energy in the proposed technique is shown in Figure 5.

The comparison graph in Figure 5 demonstrates that our proposed technique left more energy in the system than the previous techniques.

5. CONCLUSION

This paper presented security and time-efficiency multicast scaling in heterogeneous wireless sensor networks. To accomplish this, various security and time-efficiency measures were extracted. Additionally, we performed sigmoid-based fuzzy $c$-means clustering to optimize the deployment of heterogeneous WSNs. Finally, enhanced multicast routing is accomplished using support-value-based particle swarm optimization. The performance of the proposed scheduling approach was tested using a uniform benchmark, sparse cluster regime, and dense cluster regime methodologies. It outperforms the network in terms of throughput, end-to-end latency, packet delivery ratio, and residual energy in the nodes.

DECLARATIONS

This work was supported by the Department of Computer Science and Engineering, University College of Engineering(A), Osmania University, Hyderabad-500007, India.

REFERENCES


