BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **23**, No 3
Sofia • 2023 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2023-0031

A Novel Hypergraph Clustered Gray Relational Analysis HGPSO Algorithm for Data Aggregation in WSN

Shailendra Pushkin, Ranvijay

Motilal Nehru National Institute of Technology, Allahabad Prayagraj Computer Science Department, India
E-mails: s.pushkinyadav@gmail.com ranvijay@mnnit.ac.in

Abstract**:** *Wireless Sensor Networks* (*WSN*) *aggregate data from multiple sensors and transfer it to a central node. Sensor nodes should use as little energy as possible to aggregate data. This work has focused on optimal clustering and cluster head node selection to save energy. HyperGraphs* (*HGC*) *and cluster head selection based on distance and energy consumption are unique approaches to spectral clustering. GRA computes a relational matrix to select the cluster head. The network's Moving Agent* (*MA*) *may use Hypergraphed Particle Swarm Optimization* (*HGPSO*) *to collect data from cluster heads. Compared to the clustering algorithm without agent movement, the HGC-GRA-HGPSO approach has increased residual energy by 5.59% and packets by 2.44%. It also has improved residual energy by 2.45% compared to Grey Wolf Optimizer-based Clustering* (*GWO-C*)*.*

Keywords**:** *Hypergraph, Spectral clustering, Relational analysis, Energy consumption, Data aggregation, Traveling agent.*

1. Introduction

WSNs are widely used in remote and power outage areas and in military surveillance, battlefields, and disaster-prone areas. Due to the limitations in energy resources of the sensory nodes constituting WSN, energy-efficient data gathering, and transportation have become a challenging issue. Data aggregation is used to minimize the transmission count of the data packets being broadcast to the base station to meet the energy and distance constraints of the nodes' data transmission [1]. In the context of data aggregation in WSNs, a pivotal component is the WSN head agent as seen in Fig. 1, which plays a crucial role in aggregating data and transmitting it to the appropriate base stations located at a reasonable distance. However, when the base station is situated several hops away, the transmission latency and packet loss tend to increase [2]. This underscores the need for effective optimization of the movement of the data-gathering agent, particularly when it needs to traverse multiple hops to reach the base station.

The main data aggregation mechanisms are Cluster-based data aggregation and Tree-based data aggregation [1]. In cluster-based routing protocols, the sensory nodes are grouped into small clusters, each cluster has a head node called cluster head which acts as a local station for data collection for its affiliated nodes and eventually sends the collective data to the base station [3]. In tree-based data aggregation, the base station acts as the root node with leaves assumed as the head node where data is transferred from the leaves node to the base, and aggregation is performed through the head node [4]. However, tree-based aggregation suffers from uneven distribution of nodes in the hierarchy, depleting the energy of such nodes and causing network failure [4]. The clustering-based method suffers from poor energy balancing and high latency [1]. Over the last decade, a slew of improvements have been proposed to address the inadequacies of clustering-based approaches. Data aggregation has been the subject of a plethora of studies.

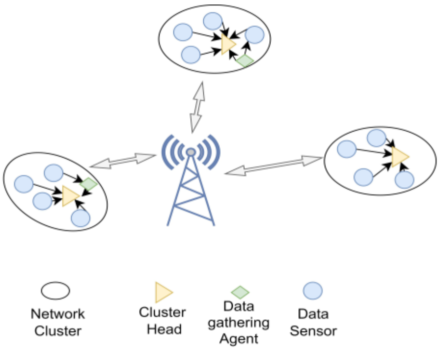


Fig. 1. Architecture of WSN

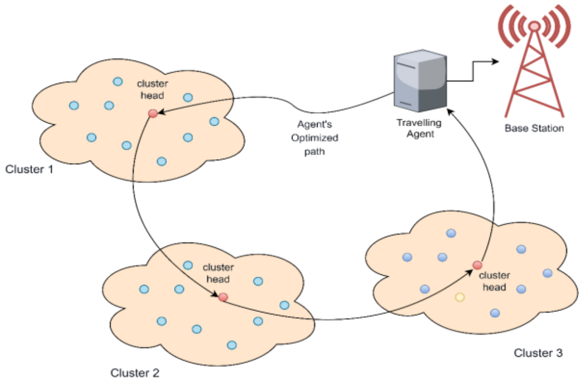


Fig. 2. WSN Data aggregation model with moving agent in the field

Among these advancements, the concept of a traveling data-gathering agent has gained attention [5]. The data-gathering agent refers to a mobile entity within the WSN that is responsible for collecting data from the parent nodes identified through clustering. This agent exhibits unique capabilities and characteristics, distinguishing it from other components in the network. Its mobility allows it to traverse the network efficiently, gathering data from various parent nodes and contributing to effective data aggregation [6]. The power of the data-gathering agent lies in its ability to minimize transmission latency, reduce energy consumption, and improve overall network performance, making it a promising solution for data collection and aggregation in WSNs. On the basis of these findings, Fig. 2 illustrates the WSN aggregation model, which includes a clustering of nodes for the selection of a parent node and a moving agent for collecting data from the parent nodes.

The remainder of the paper is organized as a scrutinized review of the pertinent contemporary literature in Section 2 followed by Section 3, which delineates the intricacies of the proposed solution. It is juxtaposed by the ensuing Section 4 demonstrating the performance analytics of the proposed solution and finally, the inference of the entire work proposed is summarized in Section 5.

2. Related work

Karunanithy and Velusamy [3] have proposed a Traveling Salesman Approach (TSP) to reduce the time taken by the UAV drone to collect the data from the cluster heads. Agrawal et al. [7] is proposed an efficient technique for the election of cluster heads in WSNs to increase the network lifespan by employing a Grey Wolf Optimizer (GWO). Mirjalili [8] propose Moth-Flame Optimization (MFO) algorithm for optimization in a very effective mechanism for traveling in a straight line for long distances, which can be extended to finding the optimal path to collect data from cluster heads. Hadia et al. [9] have proposed a modified Particle Optimization Algorithm (PSO) to find the shortest tour of minimum length on a fully connected graph, which can be extended to solve hop sequence optimization problems of the data-collecting agent. Yan et al. [19] have used autocorrelation output errors to calculate the distance between nodes and determined cluster heads for PSO. Kiran, Smys and Bindhu [20] use the modified PSO to get the optimal path from the base station to the cluster heads. Kotary, Nanda and Gupta [21] propose a reference-based leader selection and distance and energy transmission using Whale optimization. Verma et al. [22] have devised a clustering method called GABAT that combines a generic algorithm and BAT algorithm to reduce energy depletion and increase transmission security. Suleiman and Hamdan [23] have used multiple LEACH-based clustering algorithms [24], focusing mainly on the re-election of cluster heads using the proposed Adapt-P method [25]. To improve the battery life in WSN caused due to forming communication holes [26], GMM is considered for clustering [27] with a combination of social spider optimization models.

A brief analysis of attributes considered in the previous works is tabulated in Table 1.

These optimization procedures aim to reduce the latency and throughput by defining an itinerary that reduces the distance traveled to collect the data from the network where the cluster heads act as a proxy. The related works address the solutions to minimize energy consumption and maximize packet transmission by cluster head selection but do not highlight the residual energy of the cluster head whose depletion may cause no packet transfer from that cluster to the base station decreasing the throughput of the network significantly also, the moving agent's energy is restricted which decreases the latency for long-distance transmission.

Table1. Attributes used for efficient data aggregation (√ – discussed)

References	Review of availability of Attributes for efficient data aggregation					
	Energy consumption	Alive nodes	Thro-ughput	Packet delivery	Late-ncy	Moving agent
Devi, Ravi and Priya [1]	√		√	√	√	
Karunanithy and Velusamy [3]	√		√	√		√
Selvin and Kumar [4]	√		√		√	√
Agrawal et al. [7]	√	√	√			√
Arora, Sharma and Sachdeva [10]	√			√	√	√
Dwivedi and Sharma [11]	√	√		√		
Chauhan and Soni [12]	√			√		√
Yan et al. [19]	√	√				√
Kiran, Smys and Bindhu [20]	√					√
Kotary, Nanda and Gupta [21]	√				√	√
Verma et al. [22]	√		√	√	√	
Suleiman and Hamdan [23]	√				√	√

The proposed work takes the following factors into consideration:

1. The clustering of the sensory nodes should not be random but should depend on performance attributes such as transmission distance and energy consumption rate. The parent node from each cluster must have a considerable amount of energy left to avoid transmission failure.

2. To minimize the latency at the base station, the data collector for parent sensory nodes must have sufficient energy to operate at any distance constraints and minimize the distance traversed to aggregate the data.

3. The optimization algorithm must be versatile to any number of clusters and faster convergence to maximize the throughput.

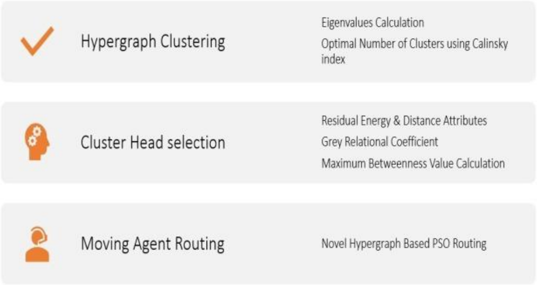
In the next section of this article, the proposed solution is discussed in detail. The simulation results are discussed in Section 3 followed by the conclusion of the work.

3. Proposed solution

The proposed work of the data aggregation model is three-fold: an optimal number of sensor nodes' groups, cluster head selection+, and optimal path scheduling of the moving agent to collect the data. As discussed in the previous section, the optimal number of clusters reduces the latency and the network lives longer than randomly selected groups. The decision depends upon the density and distributed geographical area of the nodes. For this purpose, the nodes are presented as a hypergraph and the eigenvalue for each node with respect to others in the transmission range is fed into the K-means algorithm to calculate the Calinski-Harabasz index. This work has used the Calinski-Harabasz index to select the optimal number of clusters. The maximum number of clusters in a network can be 10% of the total number of nodes. So, this serves as the maximum number of possible clusters The Calinski-Harabasz index is calculated for each cluster made by the K-means Clustering Algorithm, and the maximum index valued clustered is considered as the optimal number of clusters.

For clustering, a Grey Relational Analysis Algorithm [13] is suggested. The energy consumption in each node in the transmission of 1200 packets to every node and the distance with corresponding nodes are considered as the attributes for relational analysis. The node with the maximum grey relational value is considered the cluster head. The cluster head aggregates the data from fellow cluster members and shares this data with a Moving Agent (MA) in the field. The travel route is to be optimized for the MA. This work has introduced a novel particle swarm optimization that harnesses the power of hypergraph and is termed HyperGraph Particle Swarm Optimization (HGPSO).

The proposed three-fold methodology is presented in Fig. 3.



✓ Hypergraph Clustering

Eigenvalues Calculation

Optimal Number of Clusters using Calinsky index

Cluster Head selection

Residual Energy & Distance Attributes

Grey Relational Coefficient

Maximum Betweenness Value Calculation

Moving Agent Routing

Novel Hypergraph Based PSO Routing

Fig. 3. Proposed methodology Steps for WSN data aggregation

3.1. Optimal cluster numbers & Network hypergraph-based clustering

The WSN network is presented as a network graph where each node $V_1, V_2, \dots, V_k$ makes a disjoint set and is connected with the nodes in the transmission range. The distance between two nodes d_{ij} is the weight of the connecting edge between two nodes i and j. Based on the distances between the nodes, a similarity distance adjacency matrix is generated. The adjacency matrix highlights the relation between the nodes and the edges formed. As discussed in the previous section, the problem is the partitioning of the weighted hypergraph V into k disjoint sets, $V_1, \dots, V_k$, such that the total weight of edges within each cluster is high (denser connectivity among the sensor nodes), and the partitions are balanced. The number of nodes connected to a node is defined as that degree of any node (deg (v), $v \in V$), defining the total weight of edges v that is incident, i.e.,

(1) $$(v) = \sum_{e \in E : v \in e} w_e.$$

Here, the weight w_e associated with an edge $e \in E$. This represents the weight or similarity distance between the nodes connected by the edge.

Next, is the volume defined as

(2) $$\mathrm{CM}(V_1) = \sum_{v \epsilon V_1} \deg \deg (v),$$

which is the number of nodes incident on the node V_1 such that $V_1 \subseteq V$. The association between the edges contained within V_1 is defined as

(3) $$(V_1) = \sum \; w_e.$$

The normalized associativity of these individual partitions is given as

(4) $$\mathrm{Par}\ (V_1, \dots, V_k) = \sum_{i=1}^{k} \frac{\mathrm{assoc}(V_i)}{\mathrm{CM}(V_i)}.$$

The adjacency matrix defined here is of the tensor (order z):

$$A_{i_1, i_{2,\ldots,}i_z} = \begin{cases} w\{i_1, i_{2,\ldots,} i_z\} \text{ if } i_1, i_{2,\ldots,} i_z \text{ are distinct,} \\ 0 \qquad\qquad \text{otherwise.} \end{cases} \tag{5}$$

The normalized associativity can be now rewritten using the adjacency matrix as in Equation 1 and normalized associativity then becomes as

$$\mathrm{Par}_{i\in\{1,\ldots,k\}} = \frac{1}{z!}\mathrm{Trace}\Big(A \times_1 Y^{(1)^{\mathrm{T}}} \times_2 Y^{(2)^{\mathrm{T}}} \times_3 Y^{(3)^{\mathrm{T}}} \ldots \times_z Y^{(z)^{\mathrm{T}}}\Big). \tag{6}$$

Here $\times_l$ is the model-l product and $Y^{(1)^{\mathrm{T}}}, Y^{(2)^{\mathrm{T}}}, Y^{(3)^{\mathrm{T}}}, \ldots, Y^{(z)^{\mathrm{T}}} \in R^{k\times z}$ which represents the number of CMs connected to each node v_i for each vertex. $Y^{i\in\{1,2,\ldots,z\}}$ as shown in the next equation,

$$Y^i = \frac{1}{\sum_k \mathrm{CM}(V_i)}. \tag{7}$$

The designed adjacency matrix is considered for spectral clustering, and this hypergraph is now transverse to get the diagonal (degree matrix) Dig that is the sum of only runs over all the vehicle nodes that are one-hop adjacent to the node v_i,

$$\mathrm{Dig}_{ii} = \textstyle\sum_{j=1}^{N} A_{ij}. \tag{8}$$

Then comes Laplacian graph computation; this study utilizes the Laplacian matrix as in the next equation:

$$L = \mathrm{Dig}^{-1/2} A \mathrm{Dig}^{-1/2}. \tag{9}$$

The top k eigenvector ($X = \mathrm{eig}(L)$) is taken for the K-means clustering that provides the k-partitions of the sensor nodes' hypergraph structure. These partitions resemble the cluster formation in the network. These partitions are further punned to get the optimal set of clusters. Algorithm 1 lists the spectral clustering for the Hypergraphed sensor nodes.

Algorithm. 1. TTM Clustering in Hypergraph Theory

```
Input: Location of each node: $N_{\mathrm{Loc}}$, Node's Transmission range Tr
Step 1. Calculate the distance of each node to another and store it in a variable $\mathrm{dist}_{i\times j}$
Step 2. if $\mathrm{dist}_{i\times j} < \mathrm{Tr}$
    a. $A_{ij} = 1$
Step 3. else
    b. $A_{ij} = 0$
Step 4. end if
Step 5. if $D \in$ ⍰$^{m\times m}$is the diagonal matrix such that $D_{ii} = \sum_{j=1}^{n} A_{ij}$,
Step 6. Calculate the Laplacian matrix ($L$) using Equation (9)
Step 7. Calculate the $k$ eigenvectors of $L$, represented as $X = R^{n\times k}$
Step 8. Normalize the $X$
Step 9. Cluster the normalized $X$ using K-means
```

These eigenvalues $X = R^{n\times k}$ are used to find out the optimal number of clusters in the area. The possible Clusters Selection are selected using Calinski-Harabasz Index as it generally shows better cluster validation results when K-means is used as the clustering algorithm [14], which is defined as the sum of inter-cluster dispersion and the sum of intra-cluster dispersion of all clusters. The Inner-cluster dispersion or Between Group Sum of Squares (BGSS) is calculated as shown in the equation

$$\text{BGGS} = \sum_{k=1}^{K} n_k \times \left|\left|C_k - C\right|\right|^2, \tag{10}$$

where:

n_k is the number of observations in cluster k;

C_k is the centroid of cluster k;

C is the centroid of cluster k;

K is the number of clusters.

Also, the intra-cluster dispersion or Within Group Sum of Squares (WGSS) is calculated as shown in the equation [14],

$$\text{WGSS}_k = \sum_{i=1}^{n_k} \left|\left|X_{ik} - C_k\right|\right|^2, \tag{11}$$

where:

n_k is the number of observations in cluster k;

X_{ik} is the i-th observation of cluster k;

C_k is the centroid of cluster k.

And the sum of all individuals within groups sum of squares:

$$\text{WGSS} = \sum_{k=1}^{K} \text{WGSS}_k. \tag{12}$$

Finally, the CH index is calculated as

$$\text{CH} = \frac{\text{BGSS}}{\text{WGSS}} \times \frac{N-K}{K-1}, \tag{13}$$

where:

N is the total number of observations;

K is the total number of clusters.

Which validates and selects k clusters using the K-means clustering algorithm.

3.2. Cluster head selection

Once the clusters are decided, nodes are marked in their respective clusters. A node is selected as the Cluster Head (CH) in each cluster to aggregate the data. This CH shares this aggregated data with a moving agent in the network. This requires high energy residual with the node to hold the data for longer. The node also consumes energy in data transmission and reception to fellow cluster mates. So, a CH must be selected to be nearest to cluster nodes and have maximum residual energy. This paper proposes a novel metric for selecting the CH that meets the maximum energy and minimum distance requirements based on Grey Relational Analysis (GRA). The fundamental advantage of GRA is that it can handle data that is imprecise, ambiguous, or vague. This is due to the fact that GRA constructs the GRG using a mathematical technique. GRA is thus more robust than other systems that rely on heuristics or subjective judgments. GRA also has the advantage of being able to work with both quantitative and qualitative data [13]. GRA is thus more adaptable than other approaches that demand data in a certain format.

When deployed in a static WSN network, active sensors help save on energy costs. Since full network activity results in a significant increase in power consumption, we only activate a subset of the nodes in this case.

The assumption of the active nodes is explained as

$$\text{Active}N \;=\; \text{Total}N - \text{Sleep}N, \tag{14}$$

where Active$N$ is Active nodes, Total$N$ is the total number of nodes deployed that is $100{:}\,50{:}\,500$ (from Table 2), and Sleep$N$ are the nodes in sleep state.

The total energy consumption is calculated as

$$\text{Total}E \;=\; \text{Edt}x \;+ \text{Edt}x_{bs}, \tag{15}$$

where $\text{Edt}x = (\text{Total}N - \text{Sleep}N - 1) \times \text{Ndata} \times \text{Eec} + \text{Ndata} \times \text{dst}^2 \times \text{Eair}$ is Consumption of energy for data transmitted and

$$\text{Edt}x_{bs} = \text{aggData} \times \text{Eec} + \text{aggData} \times \; \text{dst}_{\text{CH}}^2 \times \text{Eair}$$

is the consumption of energy of the sending data from the CH and Base station.

Here, Ndata is the bytes of data to be transmitted, aggData is Aggregated data, dst_{CH} is the distance from the Base station to CH, Eec is the Consumed energy at each node on the circuit, Eair is the radio frequency amplification for the loss.

In grey relational analysis, there are three steps for decision-making:

1. Find the grey relational grade,
2. Find the grey relational coefficient,
3. Use the grey relational coefficient to make a decision.

These steps are mathematically formulated as below.

Step 1. The eigenvalues with k attributes $X_i \epsilon \, \{x_{i1}, x_{i2}, \dots, x_{ik}\})$ from Algorithm 2 are used to generate a comparable matrix for relational matrix generation as in the equation

$$Y_{ik} = \frac{(x_{ik})}{(x_{ik})} = 1. \tag{16}$$

Equation (16) creates a normalized matrix of eigenvalues to avoid biasing caused due to larger sample values in any attribute.

Step 2. The grey relational coefficient is the closeness value between Y_{ik} and Y_{0k}. A higher coefficient value indicates the closer are two samples. It can be calculated as

$$\gamma(Y_{0k}, Y_{ik}) = \frac{\Delta_{\min} + \zeta\Delta_{\max}}{\Delta_{ik} + \zeta\Delta_{\max}}, \tag{17}$$

where γ is the grey relational coefficient between Y_{0k} and Y_{ik}. Here $\Delta_{ij} = |Y_{0k} - Y_{ik}|$ and

$$\Delta_{\min} = \Delta_{ij}, \tag{18}$$

$$\Delta_{\max} = \Delta_{ij}, \tag{19}$$

where ζ is the distinguishing coefficient and randomly lies between 0 and 1. It regulates the expansion and compression of the relational coefficient. Using Equation (17) coefficient, grey relational reward is calculated, which is used to select the higher relational samples. However, in our case, we use the output from Equation (15) to generate a graph, and the betweenness degree $g(v)$ is calculated. A higher $g(v)$ sensor node among the cluster nodes is considered as the cluster head. The process is depicted in Algorithm 2.

Algorithm. 2. Cluster Head Selection Using Grey Relational Analysis

```
Input: Eigenvalue matrix $X$
Step 1. Normalize the matrix $X$
Step 2. for $i = 1{:}n$
    a. for $j = 1{:}k$
    Calculate the grey relational coefficient $\gamma_{ij}$ using Equation (17)
  b. end for
Step 3. end for
Step 4. generate a graph object from $\gamma_{ij}$
Step 5. calculate the betweenness $g(v)$ for each element
Step 6. $\mathrm{CH}_{\mathrm{ind}x} = g(v)$.
```

3.3. Moving agent routing

If the cluster head is far distant from the sink, multi-hop data transfer from the cluster head to the sink is an option. As a result, a Moving Agent (MA) is dispatched to the area in order to collect data from each cluster head to avoid the high energy consumption of CH. This section aims to find the best route for the MA to travel the shortest distance while covering every CH node. The problem formulation of moving agents to travel least is presented in Section 2. This nonlinear problem is solved with a novel HyperGraphed Particle Swarm Optimization (HGPSO) method. The HGPSO is developed on the ground of PSO. However, it improves the PSO’s premature convergence drawback by facilitating the convergence towards a good solution and diversified space. The root concept of introducing the hypergraph in the PSO is to benefit each particle with the central particle's cognitive experience. The proposed HGPSO has two main advantages:

1. It can solve the problem of premature convergence in traditional PSOs more efficiently,
2. It improves the diversity and global search performance of traditional PSOs, thus reducing the number of iterations required to find the optimally low value of the cost function.

In recent years hypergraphs have been widely used in some fields of computer science such as image segmentation [16], data mining [17], and social network analysis [18]. A hypergraph is a generalization of an ordinary graph model that can be applied to problems with more than two variables or objects. The advantage of hypergraphs is that they have higher connectivity than traditional graphs, thus making it easier to obtain connections between different nodes. The detailed discussion of HGPSO starts from the brief of PSO ahead in the section.

Particle Swarm Optimization (PSO) was proposed by J. Kennedy in 1995 and is one of the most popular evolutionary algorithms. PSO simulates social behavior to solve complex optimization problems, so it has many advantages such as simple structure, easy implementation, and low computational complexity [1]. Therefore, PSO has been widely used in many fields including mechanical engineering design, fuzzy control systems, and image processing applications. However, traditional PSOs have two main disadvantages: 1) they are easy to fall into local optima; 2) they require a large number of iterations before converging to global optima. These defects lead to poor performance for some complex optimization problems with high dimensions or large-scale applications. In order to find better solutions, many scholars have proposed some improved PSOs by introducing new ideas and methods [7]. We propose a hypergraph-based particle swarm optimization where the position update in the exploration process of a vanilla PSO is modified to incorporate a new direction vector to get better results. The velocities of PSO for a sequence optimization problem are defined as a series of swap operations based on a probabilistic update rule for the current position update.

Each particle has two attributes: location and velocity, which are represented as vectors x_i and v_i, respectively. The objective function is applied to each particle's position vector (location) in order to obtain its fitness value $f(x_i)$. The global best position $\hat{g}$ and personal best position $\hat{p}_i$ of each particle can be obtained by comparing the fitness values of particles with their own previous locations or other particles' locations. The velocity and position of each particle will be updated according to the following formula:

(20) $$v_{i,t+1} = \omega \times v_i + c_1 r_1 \times (\hat{p}_{i,t} - x_{i,t}) + c_2 r_2 (\hat{g}_{i,t} - x_{i,t}),$$

(21) $$x_{i,t+1} = x_{i,t} + v_{i,t+1}.$$

Here: ω, c_1, and c_2 are constant weighting factors; the term $\hat{p}_{i,t}$is the personal best location at time t; the term $\hat{g}_{i,t}$ represents the global best position of all particles, which can be obtained by comparing the fitness values of particles with each other; r_1 and r_2 are two independent random variables in the range [0, 1].

Equation (20) is the exploration step of the PSO and is modified for the HyperGraph PSO (HGPSO). A new fourth parameter is added to the equation as in the next equation:

(22) $$v_{i,t+1} = \omega \times v_i + c_1 r_1 \times (\hat{p}_{i,t} - x_{i,t}) + c_2 r_2 (\hat{g}_{i,t} - x_{i,t}) + c_3 r_3 (\hat{h}_{i,t} - x_{i,t}).$$

Here $\hat{h}_{i,t}$ is the best position in the hypergraph generated by the current fitness values of the particles. The update process of HGPSO is shown in Fig. 4.

The concept of hypergraph generation is similar to that discussed in Section 3.1. An adjacency matrix of $n \times n$ is used to get the weights of each particle's connection to another. The particle's fitness in any iteration is a vector quantity and it has to be converted into an adjacency matrix using the nearest neighbor calculation scheme.

(23) $$A_{n \times n} = \text{Adjacency}(f(x_1) \dots f(x_n)),$$

where A is the adjacency matrix of the vector of costs of the particles in the swarm then a hypergraph k is calculated for A to get an eigenvector using Equation (6). The centroid of k is calculated using k-means clustering, where the number of clusters is one and the index of the particle with minimum distance is calculated as

(24) $$h_{i,t} = \text{arg arg } \|\text{kmeans}(k, 1)\|_2.$$

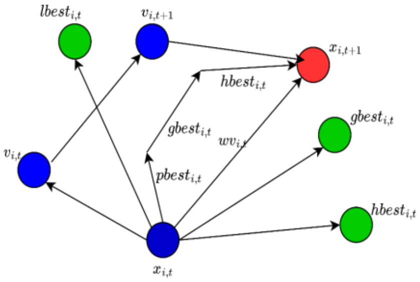


Fig. 4. Particle's positions update in Hypergraphed Particle Swarm optimization

The pseudo code for HGPSO is in Algorithm 3.

```
Algorithm. 3. HGPSO for Sequence Optimization
Input: Epoch size, Swarm size $n$ , $\omega$ , $r_1$ , $r_2$, $r_3$
Initialize: initial position of the swarm
Step 1. Calculate $p$ , $g$ , ⍰ , $f(x_i)$ for the initial positions
Step 2. do
    a. for each sequence x in the swarm do
    b. Update the velocity using $v_{i,\,t} = V_{i,\,t+1}(x,\ p,\ g,$ ⍰, $\omega,\ r_1,\ r_2,\ r_3)$
    c. Calculate new position $\underline{x}_{i,\,t}$ to update particle's position
    d. if ( $f(\underline{x})$ Is better than $f(x)$ ) then
        i. $f(x) = f(\underline{x})$;
        ii. $x = \underline{x}$
    e. end if
    f. if ( $f(\underline{x})$ is better than $f(g)$)
        i. $f(g) = f(\underline{x})$
    g. end for
    h. Calculate $\underline{⍰}$ for current epoch and particle positions
    i. if ( $\underline{⍰}$ Is better than ⍰ )
        i. ⍰ = $\underline{⍰}$
    j. end_if
Step 3. while (number of epochs are not satisfied)
```

4. Results and discussion

The proposed data aggregation scheme is evaluated with randomly placed nodes in the geographic area. The network parameters for data transmission are tabulated in Table 3. MATLAB has been used as the coding platform to implement the hypergraph for clustering, grey relational analysis for cluster head selection, and HGPSO for routing. All experiments have been performed on an Intel Core i7-2600 CPU with 3.40 GHz and 4 GB RAM running Windows 7, 64-bit operating system.

To give a full picture, the list of the various communication technologies utilized by WSNs, along with their typical transmission distances given in Table 2.

Table 2. WSN Network parameters

Communication technology	Transmission range (approximate)
Bluetooth	Up to 100 meters
Zigbee	Up to 100 meters
Wi-Fi	Up to 100 meters
LoRa	Several kilometers
Cellular (GSM, 4G, 5G)	Varies (extensive coverage)

It is necessary to note that the transmission range of the chosen communication technology should match the needs and specifications of the WSN deployment. A communication range of 20 meters is assumed in the scenario presented. To accommodate varying communication distances, the transmission range can be adjusted and different communication technologies can be selected based on the requirements of the application.

Table 3. WSN Network parameters

Number of nodes	100:50:500
Transmission range	20 m
Packet size	2000 bytes
Number of packets transmitted	64
Initial energy of nodes	1 J
Simulation time	1000 s

The proposed algorithm is evaluated on the evaluation parameters like residual energy, alive nodes and number of packets delivered. The HGPSO clustered 100 nodes with corresponding selected CH by GRA is presented in Fig. 4. The maximum number of possible clusters is considered as per the convention of 10% of the nodes' density. A sink node is placed at coordinates of $x = 50, y = 50$. The sink node and cluster heads are marked as "×" in Fig. 5.

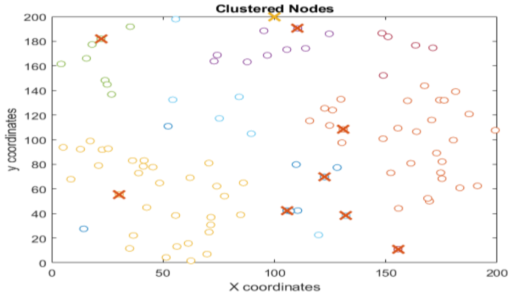


Fig. 5. Clustered WSN and selected CH with the proposed Hypergraph clustering and GRA

The evaluating parameters are calculated with the moving agent in the field at its optimal route and without any MA. A state-of-the-art comparison is also presented with other clustering algorithms such as FA-LEACH [12], GWO-Clustering [7], ACO [10], and SMA-LEACH [15] with MA in the field.

The Firefly algorithm is used for cluster head selection in a network divided into rectangular regions [12]. The cluster head selection in [7] is made using a grey wolf optimizer. The cluster members are bound with the selected cluster heads based on the minimum distance from a CH and the maximum away from other CHs. The work in [10] uses the LEACH for energy-efficient clustering and ant colony optimization is used to select the time-efficient route for 3 mobile sinks. The work in [15] is also presented on a similar line of action as in [10]; however, the sink routing is selected by the Slime mould algorithm. The proposed hypergraph-based clustering, followed by grey relational analysis for CH selection and HGPSO for MA route selection, is compared with these algorithms.

Fig. 6 compares the energy consumption of these state-of-the-art algorithms with the proposed stack of algorithms. The energy residual in transmitting the 2000 packets is simulated for 1000 seconds and calculated for proposed clustering and cluster head selection without MA in the field and with MA in the field. Other comparative algorithms are also evaluated with MA in the field. This helps to evaluate the algorithms on a common benchmark. The residual energy without MA is lesser than the energy with MA in the field. This is because MA facilitates the single-hope communication for CH to transfer the data, which could be multi-hop for many of the cluster heads. The energy consumption with the proposed clustering and MA routing has transferred the 2000 packets with 5.59% higher residual energy than the proposed clustering only. This improvement is calculated for the residual energy at the end of simulation time. HGC-GRA-HGPSO is viable to perform highest for energy residual in the same simulation environment as for other state-of-the-art schemes. It has achieved an improvement of 2.45% from the GWO-C [7] due to hypergraph spectral clustering with grey relational CH selection. The SMA-RE from [15] has performed the least and all energy is depleted in 1000 s simulation. Therefore, no alive node is in the network at the end of the simulation, as depicted in Fig. 7.

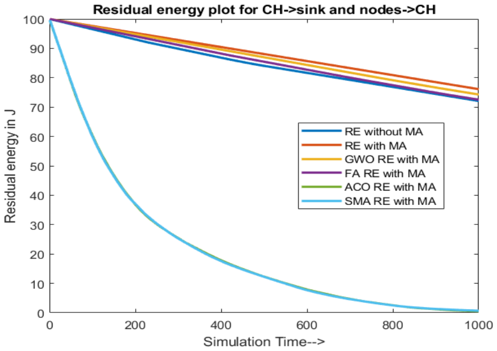


Fig. 6. Residual energy comparative plot of HGC-GRA-HGPSO with state-of-the-art algorithms for 1000 sec simulation for 2000 packets transmission

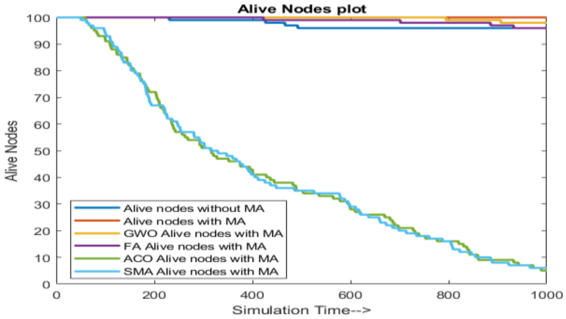


Fig. 7. Alive nodes comparative plot of HGC-GRA-HGPSO with state-of-the-art algorithms for 1000 s simulation for 2000 packets transmission

The packet delivery is calculated and the comparative plot is plotted in Fig. 8. The proposed HGC-GRA-HGPSO has all 100 nodes as the alive nodes after the 1000 s simulation. However, the number of alive nodes calculated by the algorithms in [15] and [10] is less than the proposed HGC-GRA. If the HGC-GRA-HGPSO is compared to the second highest GWO-C [7], an improvement of 2.04% is achieved. The proposed scheme with MA has 4.16% higher alive nodes than without MA. This validates the purpose of introducing spectral clustering with GRA for data aggregation.

It is noticed that the work presented in [15] has the least residual energy and no alive node, although the packet delivery is highest in it, as in Fig. 8. The 25.3% more packets are delivered by the work in [15]. Since all nodes are dying in the process, so this cannot be a reliable solution. The proposed scheme has achieved 2.44% improvement than without MA and 0.30% higher packet delivery than the work in [7].

Along with this network parameters evaluation, the routing path efficiency is also evaluated. The total distance covered by the MA should be minimized by the novel HGPSO. Fig. 9 shows the comparison of the total distance traveled by the MA. Fig. 9 indicates the total distance reduction comparison in each iteration. The HGPSO converges after 67 iterations at a lower value than PSO. An improvement of 3.67% in total distance traveled is achieved by the HGPSO than PSO. This concludes that the suggested routing algorithm collects the data from aggregating cluster heads with the least energy consumption.

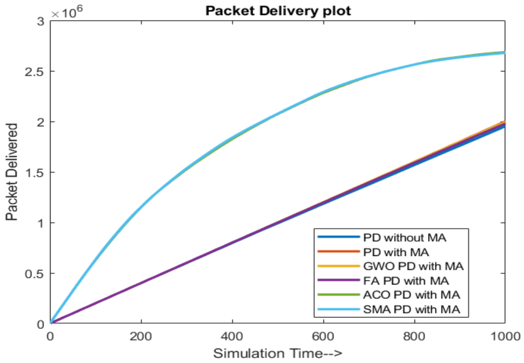


Fig. 8. Packet delivery comparative plot of HGC-GRA-HGPSO with state-of-the-art algorithms for 1000 s simulation for 2000 packets transmission

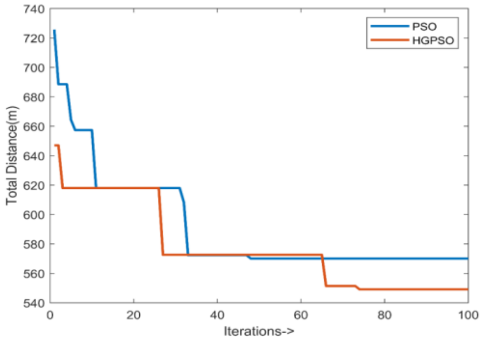


Fig. 9. Routing distance comparison by the proposed HGPSO with the state-of-the-art schemes

5. Conclusion

This study provides a novel Wireless Sensor Network (WSNs) data aggregation model. Hypergraph-based clustering, grey relational analysis, and hypergraph particle swarm optimization are used. The results show that the model achieves energy savings and reliability in Wireless Sensor Network data aggregation. Cluster analysis uses the Calinski-Harabasz index to determine the optimal number of clusters to reduce latency and increase network lifetime. Residual energy and cluster node proximity determine cluster head selection. GRA is used for this selection. A Mobile Agent (MA) improves data collection optimization, while the HyperGraph Particle Swarm Optimization (HGPSO) technique addresses premature convergence in the PSO algorithm. Compared to the clustering technique that does not include agent mobility around the field, the HGC-GRA-HGPSO approach increased residual energy by 5.59% and packets by 2.44%. The HGC-GRA-HGPSO algorithm improves residual energy by 2.45% compared to the leading Grey Wolf Optimizer-based Clustering (GWO-C) solution. This study will evaluate and optimize the model in large-scale Wireless Sensor Network (WSN) implementations. It also combines energy harvesting for sustainability and data security and privacy for confidentiality. The model's robustness and efficiency can be improved by testing its usefulness in other network topologies, implementing it in real-world circumstances, and developing adaptive network dynamics methods. Integrating the model with IoT, cloud computing, machine learning, and energy-efficient communication protocols can improve its capabilities. Real-time data aggregation methods will also improve the model's applicability to time-sensitive scenarios, promoting the development of data aggregation strategies for Wireless Sensor Networks (WSNs) and improving their dependability and energy efficiency across varied settings and applications.

References

1. Devi, V. S., T. Ravi, S. Baghavathi Priya. Cluster Based Data Aggregation Scheme for Latency and Packet Loss Reduction in WSN. – Computer Communications, Vol. **149**, 2020, pp. 36-43.
2. Kamble, S., T. Dhope. Reliable Routing Data Aggregation Using Efficient Clustering in WSN. – In: Proc. of International Conference on Advanced Communication Control and Computing Technologies (ICACCCT'16), IEEE, 2016, pp. 246-250.
3. Karunanithy, K., B. Velusamy. Energy Efficient Cluster and Travelling Salesman Problem Based Data Collection Using WSNs for Intelligent Water Irrigation and Fertigation. – Measurement, Vol. **161**, 2020, 107835.
4. Selvin, S. V., S. M. Kumar. Tree Based Energy Efficient and High Accuracy Data Aggregation for Wireless Sensor Networks. – Procedia Engineering, Vol. **38**, 2012, pp. 3833-3839.
5. Munusamy, N., S. Vijayan, M. Ezhilarasi. Role of Clustering, Routing Protocols, MAC Protocols and Load Balancing in Wireless Sensor Networks: An Energy-Efficiency Perspective. – Cybernetics and Information Technologies, Vol. **21**, 2021, No 2, pp. 136-165.
6. Raju, M., K. P. Lochanambal. An Insight on Clustering Protocols in Wireless Sensor Networks. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 2, pp. 66-85.
7. Agrawal, D., M. H. W. Qureshi, P. Pincha, P. Srivastava, S. Agarwal, V. Tiwari, S. Pandey. GWO-C: Grey Wolf Optimizer-Based Clustering Scheme for WSNs. – International Journal of Communication Systems, Vol. **33**, 2020, No 8, e4344.
8. Mirjalili, S. Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm. – Knowledge-Based Systems, Vol. **89**, 2015, pp. 228-249.
9. Hadia, S. K., A. H. Joshi, C. K. Patel, Y. P. Kosta. Solving City Routing Issue with Particle Swarm Optimization. – International Journal of Computer Applications, Vol. **47**, 2012, No 15.
10. Arora, V. K., V. Sharma, M. Sachdeva. ACO Optimized Self-Organized Tree-Based Energy Balance Algorithm for Wireless Sensor Network. – Journal of Ambient Intelligence and Humanized Computing, Vol. **10**, 2019, No 12, pp. 4963-4975.

11. Dwivedi, A. K., A. K. Sharma. EE-LEACH: Energy Enhancement in LEACH Using Fuzzy Logic for Homogeneous WSN. – Wireless Personal Communications, Vol. **120**, 2021, No 4, pp. 3035-3055.

12. Chauhan, V., S. Soni. Mobile Sink-Based Energy Efficient Cluster Head Selection Strategy for Wireless Sensor Networks. – Journal of Ambient Intelligence and Humanized Computing, Vol. **11**, 2020, No 11, pp. 4453-4466.

13. Kuo, Y., T. Yang, G.-W. Huang. The Use of Grey Relational Analysis in Solving Multiple Attribute Decision-Making Problems. – Computers & Industrial Engineering, Vol. **55**, 2008, No 1, pp. 80-93.

14. Arbelaitz, O., I. Gurrutxaga, J. Muguerza, J. M. Pérez, I. Perona. An Extensive Comparative Study of Cluster Validity Indices. – Pattern Recognition, Vol. **46**, 2013, No 1, pp. 243-256.

15. Li, S., H. Chen, M. Wang, A. A. Heidari, S. Mirjalili. Slime Mould Algorithm: A New Method for Stochastic Optimization. – Future Generation Computer Systems, Vol. **111**, 2020, pp. 300-323.

16. Ding, L., A. Yilmaz. Interactive Image Segmentation Using Probabilistic Hypergraphs. – Pattern Recognition, Vol. **43**, 2010, No 5, pp. 1863-1873.

17. Jin, M., H. Wang, Q. Zhang. Association Rules Redundancy Processing Algorithm Based on Hypergraph in Data Mining. – Cluster Computing, Vol. **22**, 2019, No 4, pp. 8089-8098.

18. Arya, D., M. Worring. Exploiting Relational Information in Social Networks Using Geometric Deep Learning on Hypergraphs. – In: Proc. of 2018 ACM International Conference on Multimedia Retrieval, 2018, pp. 117-125.

19. Yan, Z., P. Goswami, A. Mukherjee, L. Yang, S. Routray, G. Palai. Low-Energy PSO-Based Node Positioning in Optical Wireless Sensor Networks. – Optik, Vol. **181**, 2019, pp. 378-382.

20. Kiran, W. S., S. Smys, V. Bindhu. Clustering of WSN Based on PSO with Fault Tolerance and Efficient Multidirectional Routing. – Wireless Personal Communications, Vol. **121**, 2021, No 1, pp. 31-47.

21. Kotary, D. K., S. J. Nanda, R. Gupta. A Many-Objective Whale Optimization Algorithm to Perform Robust Distributed Clustering in Wireless Sensor Network. – Applied Soft Computing, Vol. **110**, 2021, 107650.

22. Verma, S., S. Zeadally, S. Kaur, A. K. Sharma. Intelligent and Secure Clustering in Wireless Sensor Network (WSN)-Based Intelligent Transportation Systems. – IEEE Transactions on Intelligent Transportation Systems, 2021.

23. Suleiman, H., M. Hamdan. Adaptive Probabilistic Model for Energy-Efficient Distance-Based Clustering in WSNs (Adapt-P): A LEACH-Based Analytical Study. – arXiv preprint arXiv:2110.13300 (2021).

24. Vijayan, S., M. Nagarajan. Deterministic Centroid Localization for Improving Energy Efficiency in Wireless Sensor Networks. – Cybernetics and Information Technologies, Vol. **22**, 2022, No 1, pp. 24-39.

25. Rathna, R. Simple Clustering for Wireless Sensor Networks. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 1, pp. 57-72.

26. Gupta, A., S. Kumar, M. Pattanaik. Coverage Hole Detection Using Social Spider Optimized Gaussian Mixture Model. – Journal of King Saud University-Computer and Information Sciences, Vol. **34**, 2022, No 10, pp. 9814-9821.

27. Ananda Kumar, S., P. Ilango, D. G. Harsh. A Modified LEACH Protocol for Increasing Lifetime of the Wireless Sensor Network. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 3, pp. 154-164.

Received: 23.12.2022; Second Version: 05.07.2023; Third Verion: 05.08.2023;
Accepted: 22.08.2023