WSN life span improvement utilize vivacity and compactness establish vertex clustering

Uzma Khan[1], Dr. Shafeeq Ahmad[2] and Masood Ahmad[3]

Computer Science & Engineering, Azad Institute of Engineering and Technology, Lucknow, India[1, 2, 3]

Abstract— Wireless Sensor Networks (WSNs) are different networks be composed of sensor confluence in huge numbers and remarkable dispensation. WSNs have different sensing potentiality and collaborate to proper usual job. Clustering is one of the faultiest skills used to settle the difficulty of vitality use in WSNs. Grating based bunch has demonstrate its organization specially for high active networks. The girds technique used in this experimentation is execute on heavy network and bits the network district into lots of or elements grid oubliette with various densities (High, Low, and empty). Then grids are consolidating to form clusters as common and precede clusters. Cluster head is select for each cluster based on large vitality. This new submits technique is experimented and investigated using MATLAB. The results show that this suggested technique toil well at 150 node WSN and grid size between 5-10 units where the network life time is 633 seconds around.

Keywords— Wireless Sensor Networks (WSN), Cluster, grid, MATLAB

I. INTRODUCTION

A wireless sensor network (WSN) is a group of spatially scattered hundreds or thousands sensor nodes that has the ability of sensing, communicating and computing. It embedded in physical spaces, continuously gather a big amount of data from the environment. Thereafter WSN is beneficial technology in many domains such as tracking, monitoring, scientific investigations and more.

Sensor networks contain the following components:

- **Data collecting:** that has the ability of sensing and acquisition during transducers.
- **Data transport:** during the wireless/adhoc channels.
- **Processing:** that has the ability of analyzing date.

We briefly report on the work performed on the presented problem by other researchers. A number of topology management algorithms and schemes have been proposed to increase WSN lifetime. This section discusses briefly the main results of the most relevant work related to lifetime extension and power management for WSN.

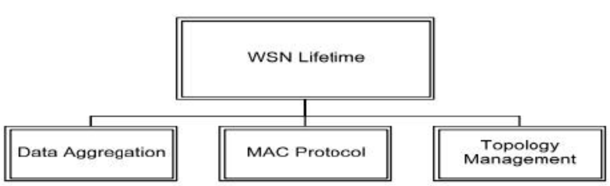

Figure 1: Three main categories of solutions extending WSN lifetime.

The solutions extending WSN lifetime can be divided into three main classes (cf. Figure1): (i) Data Aggregation solutions, (ii) MAC Protocol solutions, and (iii) Scheduling of sensor nodes solutions.

II. METHODOLOGY

Deployment of Wireless Sensor Nodes and Object Coverage:

There are two widely used types of deployment techniques for WSNs. First, predesigned deployment (including as a special case grid placement) is usually performed by human operators and results in a well-planned layout. Second, random deployment is made in the ad hoc fashion, which includes throwing out sensor nodes from a moving vehicle or an airplane. In this research we consider the second case, that is, the random deployment scheme. Recall that we assume static sensor nodes.

Density Grid-Based Clustering Methodology:

This research focuses on density-based and grid-based clustering methods. In grid-based clustering, the data space is partitioned into a finite number of equal space cells called grids, and then topological neighbor search is conducted on these grids to group the points of the closer grids. Grid-based clustering has fast processing time when it is compared with other types of clustering algorithms because all clustering operations are performed on the grid cells instead of the data objects. Many challenges are faced in this type of clustering which are :

- Determining the best size of grids: If the grid size is too large, more than one cluster could be formed inside one grid cell. In the other hand, if the grid size is small, a single cluster could be spread through more than one grid cell.
- The locality problem: When the data space has clusters with variable densities and arbitrary shapes, the global density threshold cannot result in clusters with less density.
- Selecting the merging condition that produces efficient clusters. Density-based clustering is another great method of clustering.

The idea of this type of clustering is about classifying the data set based on the dense regions, for example, if two points are very close to dense region, then these points considered to be elements of the cluster of the dense region. Density-based method has the ability to differentiate between the arbitrary shape clusters and to detect noise by considering dense areas of objects as clusters and low density area as noise.20 Density grid-based algorithms work on three stages. In the first stage, partitioning the data space into equally space partitions called grids is performed. Then, each data point in data set is mapped into a grid. Finally, one of the density-based algorithms is applied into grids to form the clusters

Density Grid-Based Clustering Algorithm:

The proposed algorithm by this research combines the density-based and grid-based methods together to develop hybrid powerful clustering algorithm. It can discover arbitrary shape clusters, detect the outliers and noise, also it has a fast processing time, in which it only depends on the number of cells instead of the number of data objects. The algorithm goes through four stages as shown in Fig. 2. Create grids and map points, Classify grids, Create Clusters and Selecting CHs (pulling). The following subsections detailed the algorithm stages:

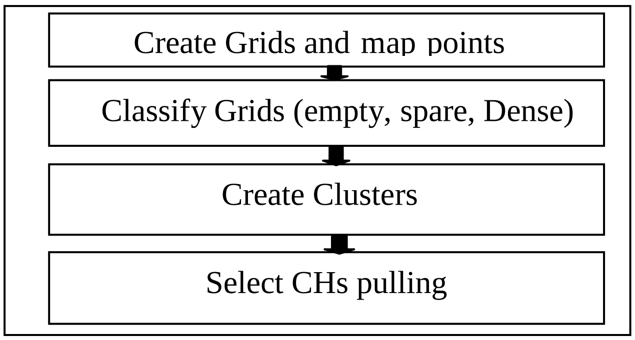

Figure 2: Four Stages of Density Grid-based algorithm

C.1 Create Grids and Map Points

In this step the data space is partitioned into equally space partitions called grids. Then, each node is mapped into a grid by using algorithm1. A grid is considered as a cells matrix where every cell contains a number of data points (nodes).

C.2 Classifying Grids

After the network space is divided, the number of data points in every non empty grid is calculated in order to determine the grid type (low or high dense grid). The standard deviation of the total number of nodes in the network is calculated according to equation (1). If the number of data points in a grid is more than or equal to the double of the standard deviation, the grid is considered as high dense grid. Thus the dense grid threshold is not need to set manually. (Algorithm2)

$$\sigma = \sqrt{\frac{1}{N}\sum_{i}^{l}}\text{(i=1) ^N(xi-x)^-2} \qquad \text{equation (1)}$$

Where xi is a single node in the network is the mean of the network nodes, and N is the total number of nodes in the space.

Algorithm 1 Create grid

1. Input: static data
2. Output: Grid matrix with data points
3. Gridx=ceil (Node x dimension /width)
4. Gridy=ceil (Node y dimension /length)
5. Grid Counter (grid x dimension, grid y dimension) = Grid Counter (grid x Dimension, grid y dimension) + 1

Algorithm 2 classify grids

1. Input: Grid Matrix
2. Output: Grid type Matrix
3. Standard Deviation = Standard Deviation Function (Grid matrix);
4. Threshold= Standard Deviation *2;
5. for every grid in the network field
6. If (there is no node in the Grid)
7. Grid type is 'empty';
8. Else if (the number of nodes in the grid is less than threshold)
9. Grid type is 'Low'
10. Else if (the number of nodes in the grid is more than or equal than threshold)
11. Grid type is 'High';
12 end if
13. End for

Create Clusters:

Clustering aims to minimize the overhead generated by the topology control and saves the node energy. In this proposed algorithm the nodes are clustered around the minimum high dense grid until certain number of nodes is reached as cluster threshold (algorithm 3). This is done according to the following steps as shown in :

Step 1: Searching for the minimum high dense grid. Grids at the borders are excluded. (Algorithm 4)

Step 2: Examining the eight neighbors of the minimum high grid. If the neighbor grid is high or low dense, does not clustered before and the cluster threshold is not reached, then this grid will be included. (Algorithm 5)

Step 3: Repeat step 2 satisfying the following conditions: o If the neighbor is high dense and any of its adjacent neighbors are also high dense, then both are included in the cluster.

- If the neighbor is high dense and any of its adjacent neighbors are low dense, then both will be included in the cluster.
- If the neighbor is low dense and its neighbor is high dense, then both are included in the cluster.
- If the neighbor is low dense and its neighbor is also low dense, then both are included and this forms the border of the cluster.

Step 4: Repeat step 2 and step 3 for every next minimum high dense grid if it is not included in any cluster before.

Algorithm 3 Create clusters

Input: Grids in the field Output: Clusters

Input: Grids in the field

Output: Clusters

1. %determines first high
2. for every grid not in the border of the network field
3. If (the grid type is dense)
4. Number of High grid incremented;
5. Minimum = hold the x and y dimensions for the Minimum high dense grid;
6. End if
7. End for
8. Minimum high dense grid x and y dimensions = Find Min High (Cluster Matrix, width, length);
9. %Cluster around the minimum high

10. for (every High dense grids in center sub matrix)
11. If (Neighbour not in cluster && Neighbour Type is "High" && Cluster value less than or equal Cluster threshold)
12. Add Neighbour to cluster;
13. Go to the next Neighbour(Grid);
14. else if (Neighbour not in cluster && Neighbour Type is "Low "
&& Cluster value less than or equal Cluster threshold)
15. Add Neighbor to cluster;
16. Go to the next Neighbor (Grid);
17. End if
18. End for

Cluster Head Selection Second point

In this algorithm, after clusters are formed, the sensors would communicate only with cluster heads, subsequently cluster heads will communicate with the processing centre (base Station) This would save node energy as well as network energy. In order to select a cluster heads (algorithm 6), the nodes in the network are divided into two types, normal nodes and advanced nodes with α times more energy than normal node. Advanced nodes have more opportunity to become a CH. Initially, the nearest advanced node from the base station is chosen as a cluster head Then CH node in each cluster is elected according to the remaining energy.

Algorithm 4 finds Min High function

Input: Cluster Matrix, width, high
Output: minimum high dense grid x and y dimensions
1. for (every grid in center sub matrix)
2. If (Grid type is 'High' && it is not in a cluster
3. && grid nodes< minimum)
4. Minimum = The Grid x and y dimension;
5. End if
6. End for

Algorithm 5 add Neighbor of the Neighbor function
Input: Cluster Matrix
Output: Add Grid to Cluster
1. % adds Neighbour of the Neighbour function
2. Go to the next Neighbour (Grid) {
3. While end not reached
If (Neighbour not in cluster &&Neighbour Type is "High" &&next Neighbour Type is "High" && Cluster value less than or equal Cluster threshold)
5. Add Neighbour to cluster;
6. Go to the next Neighbour(Grid);
7. Else if(Neighbour not in cluster && Neighbour Type is "High " && next
Neighbour Type is "low" &&Cluster value less than or equal Cluster
Threshold)
8. Add Neighbour to cluster;
9. Go to the next Neighbour(Grid);
10. else if (Neighbour not in cluster && Neighbour Type is "Low "
&&next Neighbour Type is "High" && Cluster value less than or
Equal Cluster threshold)

11. Add Neighbour r to cluster
12. Go to the next Neighbour(Grid);
13. Else if (Neighbour not in cluster && Neighbour Type is "Low" &
&next Neighbour Type is "low" &&Cluster value less than or
Equal Cluster threshold)
19. Add Neighbour to cluster;
20. Break;
21. End if
22. End while
23.} end of go to the next Neighbour function

Algorithm 6 select CH
Input: Cluster Matrix, NumOfNodeInCluster
Output: Cluster head of cluster
1. for every cluster in Cluster Matrix
2. for every nod in a cluster
3. If (NodeEnergy >= m*numOfNodeInCluster+1)
4. Node type is "Normal"
5. Else
6. Node type is "Advance"
7. End if
8. End for
9. End for
10. for every cluster in Cluster Matrix
11. for each advance node
12. Select the nearest Advance node to base station to be "a cluster head"
13. End for
14. End for
15. // after that select the height energy advance node to be
16. //"a cluster head"

III. RESULTS

Experimental Results
The algorithm run for different grid sizes to show the effect of grid size on the computational time, number of formed clusters, and network life time Table 1 shows the computation time in seconds for different grid size range from 1 to 15 units. For this range, number of formed clusters range from 90 to 6 clusters. **Fig. 4** represents grid different grid size with the number of formed clusters. Network lifetime is an important consideration of WSN. **Fig.** 5 represents the network lifetime versus grid size. From the figure, network lifetime range between 230 to 250 seconds between grid sizes 10 to 15. Although networks with 1 grid size have most better performance (life time) than other grid sizes that is 900 seconds, this experiment is excluded because each cluster has average of 1 to 2 nodes maximum which violates the logic of cluster forming strategy. In addition, its computation time to form clusters is very high as shown in Table

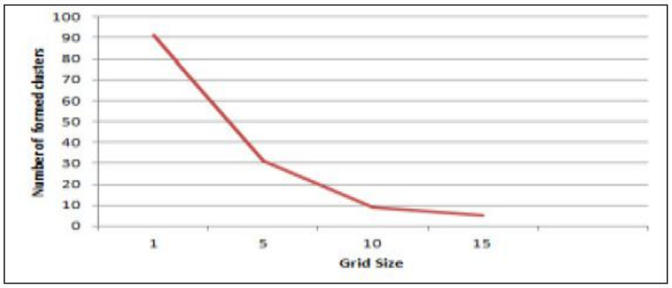

Figure 4: Number of Formed Clusters for different Grid Sizes.

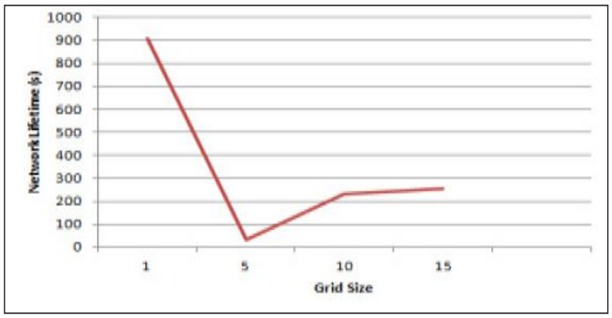

Figure 5: Network Lifetime for different Grid Sizes.

Table 1: Computation Time for Different Grid Sizes.

Grid Size	Computation Time/s
1	12.098
5	0.6488
10	0.55667
15	0.2863

The effect of increasing the number of nodes in WSN field is an important result for our experiments; we tried to test the clustering method on higher number of nodes range between 100 nodes to 300 nodes. We choose the grid
Size of 10*10 dimensions because it shows the most suitable results in terms of network life time and computation time The results are shown in **Fig. 6** and **Fig. 7**. The computation time of the experiments are shown in Table 2. The number of formed clusters ranges from 8 to 25 clusters while the network lifetime ranges between 100 to 630 seconds approximately. This clustering method achieves the best performance when the number of nodes is 150 nodes in an area of 100*100 because the network life time has the highest value as 633.0864 seconds.

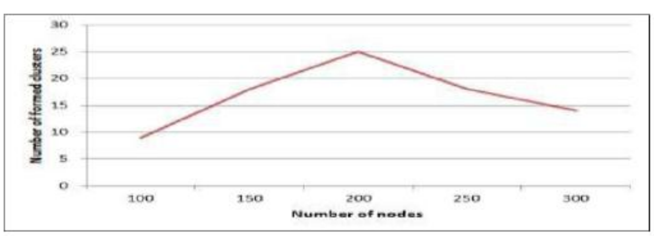

Figure 6: Number of Formed Clusters for Different Number of Nodes.

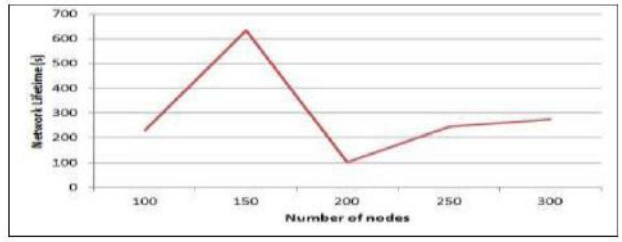

Figure 7: Network Lifetime for Different Number of Nodes.

Table 2: Computation Time for Different Number of Nodes.

Number of Nodes	Computation Time/s
100	0.5568
150	1.2212
200	0.0373
250	0.4061
300	0.4068

IV. CONCLUSION

Conclusions and Future Work

In this paper, we proposed a new clustering method for wireless sensor networks WSNs based on of density grid based clustering. In this method, the network area divided into grids which are classified as high dense, low dense and empty grids according to the number of nodes in the cluster. These grids combined to form clusters where empty girds are excluded, two adjacent high dense grids are joined in the cluster, two adjacent high dense grid and low dense grid are also joined in the cluster, and two adjacent low dense grids will become as outlier of a cluster. To determine an appropriate cluster head, cluster nodes are distributed as normal nodes and advance nodes where the cluster head initially is chosen from the advance nodes with minimum distance to base station. Then the cluster head will be elected based on highest energy. This new suggested method is implemented and tested using MATLAB. Experiments showed that to determine the appropriate grid size then the number of nodes for each cluster, we conducted the experiments for approximately 10 times where average results are considered. The most appropriate grid size which reveals best results in terms of network life time is between 5 to 10 units with 150 nodes in the WSN area.

REFERENCES

[1] Heinzelman, Wendi Rabiner, Anantha Chandrakasan, and Hari Balakrishnan. "Energy-efficient communication protocol for wireless microsensor networks."System Sciences, 2000.Proceedings of the 33rd Annual Hawaii International Conference on IEEE, 2000.

[2] Lindsey, Stephanie, and Cauligi S. Raghavendra. "PEGASIS: Power efficient gathering in sensor information systems." Aerospace conference proceedings, 2002.IEEE.Vol. 3. IEEE, 2002.

[3] Younis, Ossama, and Sonia Fahmy. "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks." Mobile Computing, IEEE Transactions on 3.4 (2004): 366-379.

[4] Tie Luo, Hwee-Pink Tan, and Tony Q. S. Quek. (2012) Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks Dynamic. IEEE COMMUNICATIONS LETTERS, VOL. 16, NO. 11, NOVEMBER 2012.

[5] Q. Ai-hua, "Research of Clustering Algorithms and its Improvements in Wireless Sensor Network " in Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on, 2012, pp. 1835-1838

[6] Li, Chengfa, et al. “An energy-efficient unequal clustering mechanism for wireless sensor networks.” Mobile Adhoc and Sensor Systems Conference, 2005.IEEE International Conference on.IEEE, 2005.

[7] Jingxian Wu, Member, IEEE, and Ning Sun, Student Member, IEEE. (2012). Optimum Sensor Density in Distortion-Tolerant Wireless Sensor Networks. IEEE Transactions On Wirless Communications , Vol.11, No.6, June 2012.

[8] G. Sikander, M. H. Zafar, A. Raza, M. I. Babar, S. A. Mahmud, and G. M. Khan, "A survey of cluster-based routing schemes for wireless sensor networks", in Smart Computing Review , vol. 3, no. 4, 2013.

[9] H. M. Abdulsalam and L. K. Kamel, "W-LEACH: Weighted Low Energy Adaptive Clustering Hierarchy aggregation algorithm for
data streams in wireless sensor networks," in Data Mining Workshops (ICDMW), 2010 IEEE International Conference on, 2010, pp.

[10] N. Javaid, M. Waseem, Z. A. Khan, U. Qasim, K. Latif and A. Javaid, “ACH: Away Cluster Heads Scheme for Energy Efficient Clustering Protocols in WSNs”, 2nd IEEE Saudi International Electronics, Communications and Photonics Conference

[11] Loscri, V., G. Morabito, and S. Marano. “A two-levels hierarchy for low-energy adaptive clustering hierarchy (TL-LEACH).” IEEE Vehicular Technology Conference.Vol. 62.No. 3.IEEE; 1999, 2005. (SIECPC 13), 2013, Riyadh, Saudi Arabia

[12] Liangrui Tang and Sheng Liu,“Improvement on LEACH Routing Algorithm for Wireless Sensor Networks” in Internet Computing & Information Services (ICICIS), 2011 International Conference.

[13] Ding,Holliday and Celik “Distributed energy-efficient hierarchical clustering for wireless sensor networks” in DCOSS'05 Proceedings of the First IEEE International conference on Distributed Computing in Sensor Systems

[14] S. Jerusha, K. Kulothungan, and A. Kannan, "Location aware cluster based routing in wireless sensor networks," International Journal of Computer & Communication Technology ISSN, 2012.

[15] P. A. Forero, A. Cano, and G. B. Giannakis, "Distributed clustering using wireless sensor networks," Selected Topics in Signal Processing, IEEE Journal of, vol. 5, pp. 707-724, 2011

[16] M. Bala Krishna and M. N. Doja , "self -Organized Energy Conscious Clustering" Advanced Communication Technology (ICACT), 2012

[17] S. V. J. E, Ganeshkumar.P, V. Suganthi.G, and K. D. Sultan.M, "A Survey on Algorithms for Cluster Head Selection in WSN," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) vol. 2, pp. 1909-1913, May 2013.