An approach to data Aggregation in wireless sensor network using Voronoi fuzzy clustering algorithm

S.Nithyakalyani1* and S.Suresh Kumar2

1Department of Information Technology, K.S.R College of Engineering, Tiruchengode, India
2Vivekanandha College of Technology for Women, Tiruchengode, India

Received 23 July 2012; revised 08 December 2012; accepted 14 March 2013

Data collected through any sensor are needed to be processed for gaining some useful information. Wireless Sensor Network (WSN) is a subclass of sensor node which is evolving as an astonishing technique in wireless communication technology for monitoring large application domains such as weather forecasting, military surveillance, medical diagnosis, fire detection alarming systems, etc. Each sensor will not be able to process itself due to the primary issue of energy (battery power) curb in WSN. Still many investigators desire to find a solution to improve the lifespan of WSN. The best way is to select an optimum head node for data aggregation to reduce the energy of data transmission for the reason that energy required for computing is more than for data transmission. This prototype shifts the attention from the outdated address-centric approaches to data-centric approach. Data centric techniques like data aggregation via energy efficient fuzzy clustering algorithm based on Voronoi diagram is proposed in this paper. The proposed novel algorithm is a combination of Voronoi and modified Fuzzy C-Means clustering algorithm called as Voronoi Fuzzy (VF) algorithm. Cluster head (CH) for VF clustering algorithm is nominated by considering node’s residual energy, distance between CH and its neighbor’s sensor node and Quality of service. Furthermore, data aggregation is employed in each cluster’s CH to reduce the amount of data transmission which effectively extends the network lifetime. Simulation result reveals that this method achieves agreeable performance in extending the network lifetime compared to the existing ones.

Keywords: Clustering algorithm, Fuzzy C-Means, Voronoi diagram, Data Aggregation, Residual energy, QOS.

Introduction

A Wireless Sensor Network (WSN) contains tiny operating system, sensing devices, memory and which normally runs in battery power. WSN is a pool of sensor node that captures the event or data occurred in monitoring environment. To obtain the knowledge or to make some decision about the sensed event, data need to be processed by sensor itself or at other end (base station). WSNs are self-motivated, distributed and adaptive systems that are found in extensive applications in our daily lives, including, medical applications, home appliances, data gathering and monitoring, military, space etc. Various critical problems in WSN are to be addressed by researchers in order to guarantee a sensible degree of cost and quality of the network. Some of the major issues are node clustering, master selection, energy dissipation and field coverage.1-4 The researcher community is aggressively observing and ruling out several optimal solutions to the above issues through distributed data mining techniques via combining clustering, coverage problem and data aggregation (gathering).

Clustering in data mining techniques is a key method that naturally reduces energy costs of WSN’s without compromising the quality of data delivery. The process of splitting the sensor nodes into crowds are called as clustering. This approach allows cluster head to collect data from the distinct sensors and forward it to sink. In clustering, cluster heads (CH) are needed to be elected first then cluster members are selected for each CH, based on distance metric. CH heads aggregate the packets from their cluster members before forwarding them to sink. By rotating the cluster head role uniformly among all nodes, each node tends to expend the same energy over the time. The clustering may allow single hop or multi hop for communication between CH and base station or sink.

Data aggregation is a surplus and different noteworthy function in WSN to conserve the energy. The key idea of this process is to eliminate redundancy in data, minimizing the number of

*Author for correspondence:
E-mail: nithyakalyani.me@gmail.com
transmissions\textsuperscript{6}, integrates all the incoming data in cluster head from diverse sources and compression is also made before enrouting data to the sink. This focuses the data-centric approach. Aggregation algorithms are limited to application requirement that is either in time or energy conservation. In the proposed work the energy consumption is measured at all stages - in the CH election, data aggregation, data routing and maintenance. QoS checks are further considered in data aggregation so that data delivery can meet QoS that are imposed by the specific WSN application. In this proposed algorithm throughput, delay time and delivery ratio are measured as QoS parameters. Voronoi diagram technique is a famous computational geometry structure, often apt for WSN coverage problem and to the find optimal cluster\textsuperscript{7}. It divides the given region into a number of small Voronoi cell in such way that there exists only one sensor in each cell. Hence Voronoi diagram can act as fundamental sampling method for WSN coverage, with sensors acting as sites. In the entire Voronoi polygons, if vertices are covered then the region is said to be completely covered; otherwise coverage hovels exist. Let $S = \{p_1, p_2, p_3, \ldots, p_n\}$ be a set of points or sites in a given area.

Voronoi region $V(p_i)$ is computed as $V(p_i) = \{x: |p_i - x| \leq |p_j - x|, \forall j \neq i\}$. A set of all sites that forms the Voronoi diagram $V(S) = \{V(p_1), V(p_2), V(p_3), \ldots, V(p_n)\}$, is explained in Fig. 1. It splits the given region into N Voronoi cells with exactly one sensor in each cell, such that all the points in a single cell are closer to the sensor in it rather than any other sensor cell in the given region. Two sensors $s_1$ and $s_2$ are considered in the given plane. Voronoi diagram is constructed by taking perpendicular bisector of the line segment joining the two sensors. To build a Voronoi diagram for three sites with a third sensor $x$, first a triangle with vertices ($s_1$, $s_2$, and $x$) at the three sites is considered. Then the perpendicular bisectors of the three sides of a triangle are drawn to meet at a single point. In this paper, the Fuzzy C-Means clustering algorithm is modified by incorporating Voronoi diagram, called as Voronoi Fuzzy (VF) algorithm. It is used to determine optimal cluster head for data aggregation, based on distance, residual energy and QOS. Centroid is used for CH selection. If the centroid value is repeated in single sensor coverage area (i.e. Single VF cell), then that sensor will act as a cluster head for that round. Each CH will then perform data aggregation. The cluster head directs the data every time to the sink node for further advanced processing. Fuzzy set theory is regularly used to handle uncertainty\textsuperscript{5}. Among several fuzzy clustering algorithms, fuzzy c-means (FCM) is the mostly used one. Since it has certain drawbacks, proposed algorithms is modified with incorporating Voronoi diagram to improve the performance.

### The Proposed VF Clustering Algorithm

The aim of the proposed approach is to reduce the consumption of energy in every sensor node and thereby increasing the entire life time of WSN. Sensors are placed to detect the event occurring in the milieu. Sensed data are needed to be transmitted regularly to the base station or sink for acquiring knowledge. However a lot of energy is spent in transmitting captured data chronically. Henceforth to overcome from the above cause, data aggregation and modified Voronoi fuzzy clustering algorithm is proposed in this work. VF clustering algorithm is carried out by sink node. Initially, Voronoi method is applied to partition sensor network into Voronoi cells. In this proposed approach any sensor’s position is projected by $x$ and $y$ co-ordinates and its residual energy using $z$. That is for any sensor node $p_i = (x_i, y_i, z_i)$ and distance function for computing Voronoi diagram is given by

\[
d(p_i, r) = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}
\]

where $r$ is the set of all neighboring nodes. Successively clusters of sensor node is formed using Fuzzy C-Means clustering algorithm, for which the first requirement is to select the CH and then its members. For finding the membership function, a number of Voronoi cells as a CH are selected and calculated. The membership function for every Voronoi cell is with the assumed
cluster head. The Voronoi cell goes to that cluster which has the highest value of membership function. With the help of the membership function, the cluster head among the clusters are calculated. In each CH, data is processed by aggregation or compression function and forwards the gathered data to the sink node.

**Pseudo code**

**Input:** Dataset containing all sensor node \( S = \{ p_1, p_2, p_3, ..., p_n \} \), its position value and residual energy as \( p_i = (x_i, y_i, z_i) \)

1. Construct Voronoi diagram of \( V(S) = \{ V(p_1), V(p_2), V(p_3), ..., V(p_n) \} \) using
   \[ V(p_i) = \{ r : d(p_i, r) \leq d(p_j, r), \forall j \neq i \} \]
2. Arbitrarily select ‘c’ sensor nodes as initial cluster heads \( ch = \{ V(p_1), V(p_2), V(p_3), ..., V(p_c) \} \) from \( V(S) \)
3. Compute membership functions using the equation (1)
   a. Determine distance membership function using the equation (2)
   b. Determine QOS membership function using the equation (3)
   c. Calculate maximum membership function \( \mu_{xy} \) for each \( V(p_i) \) using
      \[ V(p_i) \leftarrow \max(\mu_{xy}) \]
4. Compute and update ‘ch_y’ the fuzzy cluster centers using the equation (4)
5. Go to step 3 and 4 until ch_y is placed in same \( V(p_i) \)
6. Each \( V(p_i) \) send their data \( (V(p_i) \cdot d) \) to corresponding cluster head \( ch_i \)
7. Cluster head \( ch_i \) receives data from its cluster members \( D = (d_1, d_2, ..., d_i) \)
8. Implement data aggregation in each \( ch_i \) Select hope 1. Avg 2. Max 3. Min
   a. If 1 then performs equation (5), else if 2 then performs equation (6), else perform equation (7).
9. Result of step 8 goes to the sink node.
10. Stop

**Calculation of membership function**

Based on the membership functions the nodes are clustered. To find the membership function of each sensor node, certain numbers of sensor nodes are assumed as cluster heads. In order to find the membership function, first ‘c’ Voronoi cells are chosen as a CH that is \( c = 2 \) and any two cells as cluster head \( y_1 = ch_1 \) and \( y_2 = ch_2 \) are assumed.

With the help of assumed cluster head, the membership function of each sensor point is calculated. In this paper, the membership function is obtained based on two functions. The first one is based on the distance and the second one is based on the QOS values. The following equation (1) is used for finding the membership function.

\[
\mu_{xy} = \frac{1}{\alpha + \beta} \left[ \alpha \left( D_{xy} \right) + \beta \left( Q_{xy} \right) \right],
\]

\[ \forall x = 1, 2, .. n \text{ and } \forall y = 1, 2, .. c \]  

... (1)

Where

- \( \mu_{xy} \) → Membership function of \( x^\text{th} \) sensor node with respect to the \( y^\text{th} \) cluster head node
- \( D_{xy} \) → Distance between \( x^\text{th} \) sensor node to the \( y^\text{th} \) cluster head node
- \( Q_{xy} \) → Value of the QOS of \( x^\text{th} \) sensor node with respect to the \( y^\text{th} \) cluster head node
- \( \alpha \) → Weightage of distance (user defined), and
- \( \beta \) → Weightage of QOS (user defined)

**Distance based membership function**

Based on the distance between the cluster head and sensor node, the calculations are made. The Euclidian distance method is employed to find the distance between the two points. This calculation describes how much space is in-between each sensor node and each cluster head node. The sensor node moves to the nearest cluster head node that is with minimum Euclidian distance. The following equation (2) is used to calculate the distance based membership function.

\[
D_{xy} = \left( \sum_{k=1}^{c} \left\| V_i - ch_k \right\| \right)^{-1}
\]

... (2)

Running example of distance based membership function

Table 1, shows the positions of each sensor node. Here the number of nodes is assumed to be 5. In that any one of the nodes is selected as a cluster head and then distance membership function is computed. Since CH is assumed to be 2 in this example, \( C1= \)
(9.5, 1.3), C2= (8.8, 2.5) are selected as cluster heads from table 1. Then the selected values are substituted in the equation (2) to get the distance based membership function and it is shown in Table 2.

**QoS based membership function**

Table 2, contains the values of the QoS of each node with respect to each cluster head. Each of the sensor nodes has different QoS parameter with respect to an application where the sensors are functional. In the proposed work weather forecasting is taken as an application and its QoS functions are considered as throughput, delivery ratio, and delay time. Based on those QoS values the membership function for CH are obtained. Each QoS parameter can be operated as explained below. Each of the nodes has the time slots at the beginning epoch. Once the data is collected, data packets are sent to the cluster head within the next time slot. If the packets are not delivered within the particular time period then it is marked as the delayed packet. The delay time \( V_x^y (z^1) \) is calculated as the ratio of the received time to the sending time of the node; the result of this is subtracted by 1 that is a delay time.

Each of the nodes has a number of packets to send to the cluster head. While sending the data packet to the cluster head some of the packets are not received by the cluster head because of the interruption, crash or noise. The ratio of the number of received data packet in the cluster head to the number of sent data packets from the node is measured as delivery ratio \( V_x^y (z^2) \) and it varies from node to node with respect to the cluster heads.

Each of the nodes generates the data packets and sends it to the cluster head. Flow of the data packet varies from node to node with respect to the cluster head. Number of packets sent to the cluster head in a particular time interval is varied for each node with respect to the cluster head. Here throughput \( V_x^y (z^3) \) of the node is measured by number of received packets to the time period. The membership value based on the QoS values of the node with respect to the cluster heads are determined by equation (3).

\[
Q_{xy} = \frac{\sum_{q=1}^{q} V_{xy}^{y}(z^q)}{\sum_{k=1}^{k} \sum_{q=1}^{q} V_{xy}^{k}(z^q)}
\]

... (3)

**Running example of QoS membership function**

Table 2 consists of QoS values of the running example for the above table 1. Consider the first node with two clusters C1 (9.5, 1.3) and C2 (8.8, 2.5). The QoS membership function for the other nodes with respect to cluster heads C1 and C2 are calculated using equation (3) and its values are shown in column 2&3 of table 2. With the help of \( D_{xy} \) and \( Q_{xy} \) values now calculate the membership function using equation (1). The running example of the membership function is shown below.

**Running example of membership function**

The values of \( D_{xy} \) and \( Q_{xy} \) are taken from the above sections. Then the values are substituted in to the equation (1), to get the membership values for the nodes with the cluster heads C1 and C2. Here, the same weightage 0.5 is assumed for the distance and the QoS membership function. The weightage value may depend upon the user. Usually its range varies between (0-1). If the user wants to group the sensor nodes mainly based on distance then the user gives the value of \( \alpha \) greater than \( \beta \). If the user wants to
group the sensor node based on the QoS then the value of $\beta$ is greater than $\alpha$. The membership function for all nodes with respect to cluster head C1 and C2 is calculated and their values are shown column 4 and column 5 of table 3. Two membership values: $\mu_{x1}$, $\mu_{x2}$, are considered where $x$ is the sensor node, $y1$ and $y2$ are the cluster heads.

If $\mu_{x1} < \mu_{x2}$, then the sensor node $x$ moves towards cluster head y2.

If $\mu_{x1} > \mu_{x2}$, then the sensor node $x$ moves towards cluster head y1.

After finding the membership value for all Voronoi cells, each and every cell is clustered based on the two cluster heads (assumed). The Voronoi cell selects that cluster head which consists of higher value than the other. The fig 2 shows that red cells are assumed to be cluster heads C1, C2 in a wireless sensor network. Membership functions of all the other sensor nodes are gathered by either cluster heads C1 or C2. The equation (4) is used to find the adapted cluster heads with the help of calculated membership function. The m is called as fuzzier used for controlling the fuzziness of the algorithm and the value of m is greater than 1.

**Finding the cluster head**

$$ch_y = \frac{\sum_{x=1}^{n} (\mu_{x})^{m} V_x}{\sum_{x=1}^{n} (\mu_{x})^{m}} \quad \ldots \quad (4)$$

By using equation (4), cluster centroid points (3.15, 5.48) and (8.02, 4.9) are got. Both the points are plotted in Fig 2 and the corresponding Voronoi cell is treated as cluster head. After the selection of the cluster head and its members, members transmit sensed data to their CH. CH now needs to transmits the gathered data to the sink. This procedure leads to transferring sensed data a number of times so energy of the cluster head is drained much more. To solve the above problem, data aggregation or compression is proposed.

**Data aggregation**

In the wireless sensor network, each sensor node transmits its sensed data to the cluster head. The cluster head calculates the average or maximum or minimum of the received sensory data which are sent to the sink node for a fixed interval of time as shown in fig 2. The proposed method, controls the traffic of the wireless sensor network and also saves the energy of the cluster head. This helps to increase the lifespan of the WSN.

Let us consider weather forecasting application, in which sensors are used for monitoring the given environment. Here every sensor transmits the captured sensory data $V(p_i)(d)$ to its corresponding CH. A set of data is received by each cluster head $D = \{d_1, d_2, \ldots, d_i\}$ where $1 \leq i \leq n$, and $n$ is the total number of data received in the cluster head. Each $V(p_i)(d)$ comprises value of temperature, id, date and time $\{id, temp, time, date\}$. The data aggregation technique is computed using equation (5),(6) and (7) respectively.

$$avg = \frac{\sum_{i=1}^{n} d_i}{n} \quad \ldots \quad (5)$$

$$max = \max(D) \quad \ldots \quad (6)$$

$$min = \min(D) \quad \ldots \quad (7)$$

**Simulation Results and Discussion**

**Energy calculation**

The sensors are arbitrarily deployed in the 100 x 100 meters and sink node is located out the sensing area. The QOS based membership function of the cluster head is shown in table 3.

<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>$\mu_{11}$</th>
<th>$\mu_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1</td>
<td>0.55</td>
<td>0.44</td>
<td>0.535</td>
</tr>
<tr>
<td>V2</td>
<td>0.46</td>
<td>0.53</td>
<td>0.49</td>
</tr>
<tr>
<td>V3</td>
<td>0.48</td>
<td>0.51</td>
<td>0.50</td>
</tr>
<tr>
<td>V4</td>
<td>0.51</td>
<td>0.48</td>
<td>0.25</td>
</tr>
<tr>
<td>V5</td>
<td>0.45</td>
<td>0.54</td>
<td>0.72</td>
</tr>
</tbody>
</table>

![Fig. 2—Cluster using VFC in WSN](image)
field. The number of times the nodes need transmit their messages to their CH and cluster heads transmit the messages to the base station, is employed as a measure to compare the network lifetime with different clustering methods. In wireless channel, if collision occurs then the retransmission of message is not possible. The radio model and to finding the amount of energy required for data transmission, energy required to receive k-bits of data and energy required for data aggregation respectively are similar to Heinzelman

**Experimental design**

In the proposed approach, cluster head selection for data aggregation in wireless sensor network has been implemented using MATLAB. The comparative analysis of the cluster head selection approach with VF clustering and K-Means is presented. The performance of the proposed system is analyzed using the evaluation metrics including running time and size of received data in the sink node based on the number of cluster head in the wireless sensor network.

**Evaluation of running time by varying the number of the sensor nodes**

The VF Clustering algorithm is compared with the K-Means algorithm in order to evaluate the performance of the proposed VF algorithm. Fig 3a shows the running time of the VF algorithm and K-Means algorithm. The number of sensor nodes is changed at each execution and the time of both the algorithms are evaluated, the value of the cluster head being constant for both the algorithms to find the execution time. The number of cluster heads is taken as 10 for all execution. Fig 3a shows that, the VF algorithm needs less time to execute when compared with the K-Means algorithm. The reason behind is that if the membership function value of the sensor node is repeated in a single Voronoi cell then the execution of the membership function for that sensor node stops and hence the running time of the VF algorithm gets reduced.

**Evaluation of running time by varying the number of cluster heads**

Here, the running time is evaluated by keeping the number of sensor nodes constant and varying the number of cluster heads each time. Fig 3b shows the running time of VF algorithm and K-Means algorithm. From the graph it is seen that proposed approach via the VF algorithm takes less time to execute compared to the K-Means algorithm. The running time of both the algorithms is directly proportional to the number of cluster heads. The calculation of the membership function for each point with respect to each cluster head takes more time. Hence the running time increases as the number of cluster head increases.

**Number of received data in sink node by varying the number of the sensor nodes**

After the selection of cluster head, each of the sensor nodes sends its sensed data to the cluster head. The cluster head has received many data from the sensor nodes, as per the user instruction. The cluster head operates on the received data and sends that data to the sink node. The sink node receives the data from the cluster heads. Fig 4a shows the number of received data in a sink node for different number of sensor nodes. Based on the position of the cluster head, the received data of the sink node gets varied since the position of the cluster head is different for both VF algorithm and K-Means algorithm. Hence, the number of received data in a sink node as a

![Fig. 3(a) — Running time VS number of nodes](image1)

![Fig. 3(b) — Running time VS Number of clustering head](image2)
The parameter is taken. The graph in fig 4a shows that, the sink node receives more number of data by using VF algorithm than by using the K-Means algorithm.

Number of received data in sink node by varying the number of the cluster heads

The number of received data in a sink node varies depending on the number of cluster heads in a sensor network. Fig 4b shows the number of received node in a sink node based on different number cluster heads for both VF algorithm and K-Means algorithm. The sink node receives more number of data by using VF algorithm than by using the K-Means algorithm.

Conclusions

In this paper, the approach for energy efficient cluster algorithm for data summarization in wireless sensor network has been presented. In order to reduce the energy of the wireless sensor network, we have presented the efficient Voronoi fuzzy clustering algorithm. At first the Voronoi cells are applied for each node in the wireless sensor network based on the energy of the sensor node. Consequently, the cluster head is selected by the fuzzy clustering algorithm. Every node in the wireless sensor network sends its sensed data to the cluster head. The data aggregation and data compression process is done in the cluster head. The cluster head aggregates the received data by taking the minimum, maximum and average values. The aggregated value is sent to the sink node by the cluster head. Finally, the experimental results shows that the efficiency of VF clustering algorithm (60%) is higher when compared with the K means algorithm.

References