Clustering of Sensor Nodes using Spatio-Temporal Correlation

Surbhi Madan[*]

CSED, NIT Hamirpur, Hamirpur, India

E-mail: er.surbhimadan.2013@gmail.com

Abstract: To maximize network lifetime and to get the accurate information in wireless sensor networks, the nodes are grouped together in such a way that it support better data aggregation, high scalability and reliability. These disjoint and non-overlapping subsets are called clusters. In this paper, we propose a correlated spatio-temporal clustering algorithm based on magnitude of sensor nodes to form the clusters. Moreover, network energy consumption issue is also considered in cluster head selection process by considering minimum distance node as new cluster head. The experimental results show that the proposed method efficiently performs clustering and prolong network lifetime with good accuracy and also perform data aggregation with accurate results.

Keywords: Wireless sensor network; Sensor nodes algorithm; Cluster head; Spatio-temporal correlation

I. INTRODUCTION

The interconnection of spatially distributed sensor nodes via wireless links forms Wireless Sensor Network (WSN). Each and every sensor node contains one or more sensing unit, communication component, processing unit and also a source of power such as battery [1-7]. After sensing the data, data is collected, processed and send to the designated sink. Due to affordable sensors and storage, spatio-temporal data are becoming ubiquitous randomly. The goal of cluster detection is to answer two essential questions: is anything interesting going on, and if so, where? Clustering is a type of unsupervised data mining technique. It is very successful especially for static data. But a little work needed in case of spatio-temporal fields because it supplies continuous spatio-temporal data and dynamic clusters are to be form. The cluster detection problem presents both statistical and computational challenges. The statistical challenge is to accurately detect relevant clusters, while keeping false positives to a minimum. The computational challenge is to detect these clusters very rapidly even for massive real-world datasets. To deal with these challenges, we have developed new algorithmic techniques, for rapid and efficient detection of dynamic clusters. Most importantly, this framework is sufficiently general to be usable for a wide variety of applications and sufficiently flexible to be easily adapted to new application domains. In this paper, we propose a novel spatio-temporal clustering paradigm to identify clusters in a continuous spatio-temporal field where clusters are dynamic and may change their size, shape, location, and statistical properties from one time-step to the next. Spatial correlation usually exists among the readings of close sensor nodes meaning that the measures from a sensor node may be predicted from that of its nearby sensor nodes with high confidence. Therefore, given a certain requirement on spatial accuracy, only part of the sensor nodes should be required to work for sampling and data transmission, in order to save energy.
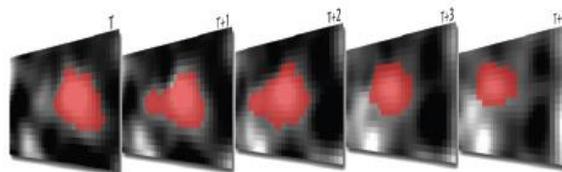
Figure 1: An example of spatio-temporal clusters.

In Figure 1, at each time step (ti) we must extract the cluster from the continuous field (red pixels) and track its evolution over time [8].

Data streams can define as:
Data Streams= Continuous + Temporally ordered + Fast Changing + Massive + Potentially Infinite

Unlike spatial correlation, temporal correlation exists in the time series from a single sensor node, meaning that the future readings of a sensor node can be predicted based on the previous readings from the same node.

II. BACKGROUND AND RELATED WORK

2.1 Problem Formulation
The goal of this work is to form the clusters of data streaming sensor nodes after exploiting spatio-temporal correlation. There are some examples of continuous spatio-temporal fields like, earth sciences or remotely sensed data where each location has unique spatial coordinates and time vector series characterized them by representing the evolution of the feature vector over time. Thus, given a spatio-temporal field, our task is to form the clusters in space and associate the cluster across time [9].

2.2 Existing Clustering Approaches
Clustering is done to relate similar nodes and saves necessary energy wasted in direct data transmission to the base station. Many clustering algorithms have also been proposed in the past for various contexts but according to survey conduct these algorithms work for static data and are heuristic in nature and their aim is to generate the minimum number of clusters. To our knowledge only generation of minimum number of clusters does not ensure minimum energy consumed. In k-mean clustering algorithm [10], author proposed two types of clustering as: Centralized k-mean clustering and Distributed k-means clustering. These algorithms are based on Euclidian distances and energies for choosing cluster head. This information is obtained by every node by exchanging messages among themselves. The Weighted Clustering Algorithm (WCA) elects a node as a cluster head based on the number of neighbours, transmission power, and battery-life and mobility rate of the node [5]. In Geodic Sensor Clustering Protocol, a new distributed clustering is introduced. The proposed protocol is based on novel localized metric for measuring the value of a node [11]. In this clustering is performed after forming the graph structure by given number of nodes. The node that is surrounded by maximum number of nodes becomes cluster head. The calculation of this matric is linear in the number of nodes. In Distributed WSN Data Stream Mining based on Fuzzy Clustering [12], author proposed a new clustering approach of given data streams. This approach was based on SUBFCM algorithm that is, Subtractive Fuzzy C-Means Algorithm. In this, numbers of clusters are not pre-known. In this algorithm a node having highest potential will be selected as cluster head. In this CH does not send raw data to base station instead transmit the result after processing of the raw data. In Clustering Dynamic Spatio-Temporal Patterns in the Presence of Noise and Missing Data [8], author proposed an algorithm in which first core points are detected. Core points are that points who always present or exist in a cluster. On the basis of those core-points, other points or nodes are covered or linked to form the clusters. Core points are always chosen to be cluster head. This algorithm effectively works on spatio-temporal correlation of data streams.

2.3 Challenges
To work with data streams is not an easy task because it is very different from static data. Despite of technical limitation clustering data streams has many challenges to face. The data are routinely missing. As there is no systematic methodology for streamed pre-processing. So clustering data streams should work with this limitation. There are not standard methods available for ensuring the privacy with incomplete information as data arrives, while taking into account the evolving nature of the data. Apart from this data stream clustering need one pass algorithm to form the clusters. Spatio-temporal correlation also faces to types of problem: Heterogeneity in space and time [13].

III. SPATIO-TEMPORAL CORRELATION BASED CLUSTERING PARADIGM

To work with above challenges, we propose an approach of spatio-temporal correlation based clustering. In this, values of nearby sensor nodes are taken into consideration over time. We propose an approach in which all above challenges are to be considered and we try to represent a general paradigm to provide spatio-temporal correlation. Our paradigm consists of two main steps: Identification of head sensor node a head sensor node is a node that will rely in the cluster surely that means it has sure membership in the clusters) and adding another node to form the clusters by defining threshold for a cluster [14].
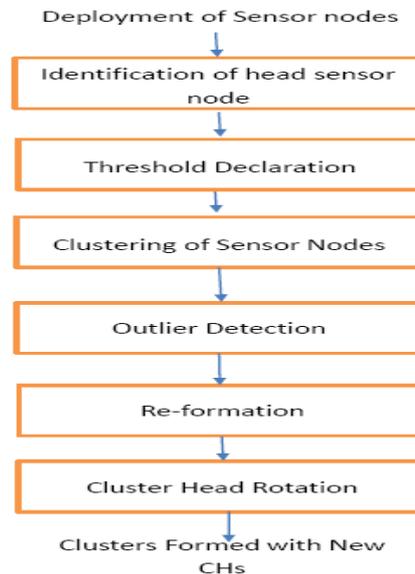
Figure 2: Our proposed steps for spatio-temporal clustering of data streams.

In our approach, we also work on re-formation of clusters means to re-form the clusters. After forming initial clusters we detect and remove the outliers from the clusters and re-cluster them into new clusters so that we can get the accurate aggregated results. Our approach includes steps shown in Figure 2. Head sensor nodes are chosen or identified randomly and these nodes will be the base or head points to form the clusters. On the basis of these points thresholds are declared and our approach will cover all those sensor nodes that are part of those clusters. And by using our proposed algorithm for clustering, these clusters are formed. After forming the clusters by given deployed sensor nodes our approach works on identification of outliers that means to detect that nodes that are not giving the best results after staying in that cluster and will give accurate result if it is part of another cluster. So after applying outlier detection algorithm we will be able to see those sensor nodes that are not part of those clusters that were first formed. By those sensor nodes our approach will be able to re-cluster the sensor nodes.

IV. PROPOSED METHOD

In the proposed approach, the advantage of the fact that in many domains, although the clusters may move, there are "head points" that never change cluster memberships for a given time window, is taken. This is an important observation to work with noisy and missing data (as a challenge in data streams). Although there are many ways to implement this paradigm, this section presents one of the methods to implement this scenario so that our work can contribute a small part to the community. Our method also uses data mining techniques for collection of spatio-temporal data.

4.1 Clustering Objective
The first step is to define the objective of our clustering methodology. We are focusing on "Temperature" property of the earth. Our objective is to form the clusters on the basis of temperature property and then calculation of aggregated value to see which part of the land is suitable for lives or agriculture. The goal is to extract the clusters and their dynamics over one year period of time.

4.2 Data Collection
To collect the data, our approach uses data mining technique called "Event Detection". Every sensor nodes are having data frames for different time-slots. So the data is in the form of time-series data. In this step, on every time-slot if event is happening that means sensor node is providing value after sensing, value is recorded into data-frames. In this way all sensor nodes are having their individual data frames for different time slots. This process is repeated over a specified time period and then these data-frames are sending to the sink node also called base station. After collecting

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Vol. 5, Issue 11, November 2017

these data frames, sink node create a database in which these values are stored and also the position of each sensor nodes (each sensor nodes knows their position and also position of their neighbours that are maximum one hop away). After collecting all these data its sink node's responsibility to create a summarized database in which each and every sensor node is assigned their average value and its position value.

4.3 Clustering

After specifying clustering objective and collection of data, the third step of this approach is to form the clusters of sensor nodes that rarely change the cluster membership for a given time of period. The main motivation to form the clusters is that nodes are grouped together not only based on their spatial properties but also their long-term temporal similarity. Our approach is also using basis of DBSCAN algorithm to form the clusters. DB-SCAN Ester et al. is a density based clustering algorithm. It forms the cluster of nodes that are closely packed together in the feature space. The algorithm identifies "head nodes" that have at least m nodes in their cluster and other nodes are maximum of n-hop away from head nodes. In our approach we consider value of m to be 3 and value of n to be 1. That means after identifying head nodes, the nodes that are maximum of 1 hop away from the head node and cluster have at least 3 nodes , are ready to be a cluster. In this step, we are proposing three basic algorithms to form the perfect clusters as: Clustering algorithm, Outlier detection algorithm and Re-clustering algorithm. In our clustering approach, there are some assumptions made. The assumptions and used terms are given as:

4.4 Assumptions
- The nodes that is close to each other in terms of their position values, having similar range of magnitude.
- Sink Node knows the position values of each sensor node.
- All sensor nodes are having the knowledge of their neighbor node.

Used terms:
X_i = Magnitude of Sensor node (S_i)
T = Threshold value
C = Clusters formed
Min_node = Minimum number of nodes in a cluster
N = Total number of sensor nodes
C = Total number of clusters formed (initially, C = 0)
Z = Clusters
D_i = Distance from Cluster head
Min_T_c = Minimum value of threshold
Max_T_c = Maximum value of threshold
T_D = Distance threshold

Input: Randomly deployed sensor nodes with its magnitude and position value.
Output: Clusters formed by these values

```
Algorithm:
Step 1: for i=1 to N
Step 2: if ((Status of Si = Assigned) or (Value of sensor node (Xi) = 0) then go to step-1
Step 3: else
 {
 C = C + 1;
 Define a threshold as: (Xi-2 ≥ Tc ≥ Xi+2)
 Add Si to cluster Zc
 Set, Status Si = Assigned
 Si = Core-Node
 }
Step 4: for j = i+1 to N
 {
```

Step 5: if (Status Sj = Assigned) then Go to step 4
Step 6: else
{
if ((Xj lies within the range of Tc) and (node is maximum of 1-hop away ($|Di - Dj| \leq TD$)))
{
Add Sj to Zc
Set, Status Sj = Assigned
if ((Sj == Min_Tc) or (Sj == Max_Tc)) then Set, Sj = Boundary Node
}
else
Goto step-4
} / end of j-loop
Step 7: C = C+1
} / end of i-loop
Step 8: if (number of nodes (M) in a cluster Zi < Min_node) then ignore the cluster Zi
Step 9: Mark all the hedd nodes as Cluster Head and Return

In above algorithm, outcome is clusters formed by deployed sensor nodes. In our next algorithm, we will detect the outliers from above formed clusters.

4.5 Outlier Detection Algorithm
The steps of the algorithm and used symbols are given below:
Used symbols:
CH = Cluster head
SN = Sensor node
D_x = Distance of CH from sink node
D_y = Distance of first node from CH
R_i = Data transfer rate of CH
R_j = Data transfer rate of SN

Data Transfer Rate = No of packets sent / Time taken to send the data packets
C = Total number of clusters
Y = List of outliers

Input: Clusters formed from SNs
Output: List of outliers Y

Algorithm:
Step 1: for i=1 to C
{
Step 2: Calculate Dissimilarity Threshold as:
$DST_i = \sqrt{(Dx - Dy)^2 + (Rx - Ry)^2}$
Step 3: Calculate Dissimilarity Measure between Nodes as: for each pair of nodes (a,b)
{
$DS = \sqrt{(Da - Db)^2 + (Ra - Rb)^2}$
Step 4: if (DS < DST_i) then nodes lie in that cluster
Step 5: else,
Set node S_b as outlier and add this node to list Y
}
}

4.6 Re-Formation

After detecting outliers, our approach will focus to choose the right clusters for these outliers and in that way next algorithm is for Re-clustering. Terms that are used in our algorithm is given below:

Used terms:
X = number of nodes in list Y
J = number of thresholds
Input: List of outlier nodes
Output: New Clusters after Re-formation

Algorithm:
Step 1: for i = 1 to X
 {
Step 2: if (($S_i \in (T_c \vee T_{c-1} \vee T_{c-2} \ldots\ldots\ldots T_1)$)) and (node is maximum of 1-hop away from CH)) then Add node i to cluster Z_c
Step 3: else
 Goto step-1
Step 4: Return

4.7 Cluster Head Rotation Algorithm

After formation of CHs (Cluster Heads), our approach will focus on to rotate the old CHs so that in case of failure of old CH the processing of complete paradigm will not stop. Steps of the algorithm and used terms are given as:

C = Number of Clusters
M = Number of nodes in a Cluster
D_j = Distance of jth node from CH
X_{CH} = x-co-ordinate of CH
Y_{CH} = y-co-ordinate of CH
X_j = x-co-ordinate of jth node
Y_j = y-co-ordinate of jth node
R = Communication Range of CHs
New_CH = New cluster head
Input: Clusters with old Cluster Heads (CHs)
Output: Clusters with new CHs

Algorithm:
Step 1: for i = 1 to C
 {
Step 2: for j = 1 to M
 {
Step 3: Calculate distance as:
 $D_j = \sqrt{(Xch - Xj)^2 + (Ych - Yj)^2}$
Step 4: for every calculated distance D_j, do
 {
 if ($D_j < R$)
 {
 Set min = D_j
 Head_node = S_j
 }
 } /end of j-loop
Step 5: Set New_CH = Head_node
 } /end of i-loop
Step 6: Return

International Journal of Innovative Research in Computer and Communication Engineering

V. REFERENCES

1. SV Manisekaran, R Venkatesan, et al. Mobile Adaptive Distributed Clustering Algorithm for Wireless Sensor Networks. International Journal of Computer Applications 2011; 20: 12-19.
2. CR Lin, M Gerla, Adaptive Clustering for Mobile Wireless Networks. IEEE Journal on Selected Areas in Communication 1997; 15: 1265-1275.
3. Halkidi Maria, K Iordanis, Online Clustering of Distributed Streaming Data using Belief Propagation Techniques. IEEE International Conference on Mobile Data Management 2011.
4. J Hossein, An Introduction to Various Basic Concepts of Clustering Techniques on Wireless Sensor Networks. International Journal of Mobile Network Communication and Telematics 2013; 3: 1-17.
5. C Mainak, K Das Sajal, et al. WCA: A Weighted Clustering Algorithm for Mobile AD hoc Networks. Journal of Cluster Computing 2002; 5: 193-204.
6. K Vinay, J sanjay, et al. Energy Efficient Clustering Algorithms in Wireless Sensor Networks: A Survey. International Journal of Computer Science Issues 2011; 8: 259-268.
7. W Yi, Network Big Data: A Literature Survey on Stream Data Mining. Journal of Software 2014; 9: 2427-2434.
8. CC Xi, FH James, et al. Clustering Dynamic Spatio-Temporal Patterns in the Presence of Noise and Missing Data. Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI) 2015; 2575-2581.
9. A. Boukerche and S. Samarah, "A Novel Algorithm for Mining Association Rules in Wireless Ad Hoc Sensor Networks," IEEE Transactions on Parellel and Distributed Sytems, vol. 19, no. 7, 2008.
10. P Sasikumar and K Sibaram, K-Means Clustering in Wireless Sensor Networks. IEEE International Conference on Computational Intelligence and Communication Networks, 2012.
11. KR Suneetha, R Krishnanmoorth, Data Preprocessing and Easy Access Retrieval of Data through Data Ware House. Proceeding of the Worlds Congress and Computer Science 2009; 1: 1-6.
12. C Tharini, P V Ranjan, An Energy Efficient Spatial Correlation Based Data Gathering Algorithm For Wireless Sensor Networks. International Journal of Distributed and Parallel Systems 2011; 2: 16-24.
13. Z Yujie, Clustering Methods in Data Mining with its Applications in High Education. International Conference on Education Technology and Computer. 2012; 43: 1-7.
14. A Mahmood, S Ke, et al. Data Mining Techniques for Wireless Sensor Networks: A Survey. International Journal of Distributed Sensor Networks 2013.