A NOVAL ENERGY EFFICIENT APPROACH FOR COMPOSITE EVENT DETECTION IN WSN

Soniya Avarachan, Prof.Praveen A.N
M Tech Scholar, Assistant Professor
Dept of Information Technology
GEC Idukki, Kerala, India

Abstract : Wireless sensor networks, well known communicating architectures today, are often used to detect the occurrence of some environmental events, such as pollution monitoring, forest fire detection, location and tracking, etc. In order to reduce irrelevant alarms, multiple attributes are used in the event detection process. It is important to reduce the amount of data exchange within a WSN, in order to optimize the use of power and energy resources within nodes. Event-based clustering for composite event detection for wireless sensor networks is proposed in this paper which performs local computation at sensor nodes and local data aggregation at level of each cluster heads in order to reduce the communication overhead and improving the energy efficiency. In real world applications it is important to get the reports as soon as possible. In order to achieve this routing is performed based on the node weights. In this paper our focus is to reduce the reporting delay of nodes without compromising the energy efficiency.

***IndexTerms* – Event detection, Event based clustering, Composite event, node weight, reporting delay.**

I. INTRODUCTION

Wireless Sensor Networks (WSNs) are widely used to detect the occurrence of some environmental events, such as pollution monitoring, forest fires detection, smart homes, and smart cities etc. WSNs play an important role in monitoring and reporting environment information and collecting surrounding context. A WSN event is defined as an observable occurrence of a phenomenon or an object during a period of time in a specific area. The events to be detected by WSNs can be classified into atomic and composite events. Atomic events measure changes of a single attribute in the environment, for example, the temperature, while composite events consist of groups of atomic events. Information from the sensors is aggregated, and events of interest are reported to the sink. Aggregating data closer to the event location saves energy compared to the case when data is aggregated at the sink.

Nodes in WSN are resource constraint in terms of power, bandwidth, memory, and computing capabilities. Since WSNs are sometimes deployed in hostile environments where human access is limited, and recharging or replacing wireless nodes is prohibited. In such situations, mechanisms for event detection and reporting have to minimize power consumption in order to prolong network lifetime. In the case of composite events, WSN consisting of n heterogeneous nodes N_1, N_2, ..., N_n and a sink S are considered. The nodes are densely deployed and they are connected to the sink. All the nodes have the same communication range and the same initial energy. The nodes are heterogeneous, since each node is equipped with one or multiple sensing components. Each sensing component can be used to detect an atomic event for that attribute. There are few reason that nodes have different sets of sensing components. Nodes may be manufactured with different sensing capabilities. Some nodes may have purposely turned off some sensing components due to energy constraints. Some sensing components may fail over time. Some of the sensing components cannot be used due to lack of memory for storing data.

An atomic event is triggered when a single sensing value (or attribute) exceeds a given threshold. An atomic event is denoted by e(t, s,R) where t is the time when the event occurs and it can be a specific time or an interval, s is the location of the event and it can be a point or a region, and R is a logical expression defining the conditions when the event occurs. For example, the atomic event e(t, s, R)= (9/1/2016,(x,y), temperature > 100C) means that the temperature at the location (x,y) on 9/1/2016 was greater than 100C. To detect complex events in certain areas, variations in several attributes have to be detected, not only in one attribute. To detect a composite event, a combination of several sensing values is needed. A composite event is therefore composed of several atomic events.

In order to reduce irrelevant alarms, multiple attributes are used in the event detection process. In WSNs, communication is often by far more expensive and difficult to control than local computation within nodes. Therefore, it becomes critical to reduce the amount of data exchange within a WSN, in order to optimize the use of power and energy resources within nodes. Energy optimization is thus one of the most important aspects of the WSN design. Data gathering is also a challenging problem in WSNs because the information has to be available quickly and effectively without delays and redundancies. The project is focused on minimizing the reporting delay without compromising the energy efficiency.

II. LITERATURE REVIEW

Wendi Rabiner Heinzelman et al. [1] proposes LEACH (Low-Energy Adaptive Clustering Hierarchy). It is a self-organizing, adaptive clustering protocol. The LEACH makes use of rotation of cluster-heads for the even distribution of the energy load among the sensors in the network. LEACH uses localized coordination for scalability and robustness in dynamic environments, and incorporates data fusion into the routing protocol to decrease the information transmitted to the base station. LEACH outperforms classical clustering algorithms with the use of adaptive clusters and rotating cluster-heads. In addition, LEACH is capable of performing local computation in each cluster. This helps in achieving a large reduction in the energy dissipation, as computation is much cheaper. Once the cluster-head has the entire data from the nodes in its cluster, the cluster-head aggregates the data and compressed data is sent to the base station. If the base station is far away from the scenario we are examining, then high energy is required for transmission. It doesn't support event based clustering.

Stephanie Lindsey et al. [2] developed an improved protocol called PEGASIS (Power-Efficient Gathering in Sensor Information Systems), which requires less energy per round compared to LEACH. The main aspect of PEGASIS is the formation of a chain within sensor nodes so that each node communicate with a close neighbour. Gathered data moves from one node to another, and finally a specified node transmits to the BS. Nodes take turns to transmit to the BS so that the average energy spent by each node per round is reduced. With the radio communication energy parameters, a simple chain built with a greedy approach performs quite well. PEGASIS protocol achieves up to 100% improvement with respect to energy cost per round compared to the LEACH protocol. For gathering data in each round, each node transmits to a close neighbour in a given level of the hierarchy. The final node will send the aggregated information to the base station. It also doesn't support event based clustering.

Ameer Ahmed Abbasi et al. [3] introduced a mechanism called Hybrid Energy-Efficient Distributed Clustering (HEED). It is a distributed clustering scheme in which CH nodes are picked from the deployed sensors. HEED considers a combination of energy and cost of communication while performing selection of CHs. Unlike LEACH, it does not select cell-head nodes randomly. Only sensors that have a high residual energy can become cell-head nodes. HEED has three main characteristics. First, the probability that two nodes within each other's transmission range becoming CHs is small. Unlike LEACH, this means that CHs are well distributed in the network. Second, energy consumption is not assumed to be uniform for all the nodes. Third, for a transmission range, CH selection may be adjusted ensuring inter connectivity between cluster heads. In HEED, each node directly communicate with its CH, and finds the CH going through different rotations and transmit with the minimum cost. This type of clustering has large coverage and better energy efficiency. Event based clustering is not supported by this mechanism.

Traian Muntean et al. [4] propose a threshold-based approach for composite event detection in WSNs. The main advantages of this proposed approach is simplicity, energy efficiency, and can be applied more easily. Proposed Cluster-based Energy efficient Composite event detection (CEC) algorithm is a two-level event detection scheme. At the first level, sensor nodes perform local computation to detect atomic events and transmit a report message to the CH, when an atomic event has been observed. Moreover, the second level is carried out at intermediate nodes (i.e., CHs), which perform local data aggregation of received atomic event reports from the sensors in the cluster and take decision for a composite event. The main contributions of proposed scheme are first, all sensor nodes (i.e., CHs and cluster members) send packets in binary format instead of raw data; achieves low overhead on data packet. Second, each cluster member performs local computations on the sensed data, and sends a report message to the cluster head when an atomic event has been observed. Third, CHs perform local aggregation of data for the received data from members and evaluate the composite event. Then, the CHs send particular synthesis of composite event occurrence to the base station, when a composite event has been observed. Finally to reduce false alarm rate to the base station, composite event definition consists of multiple attributes. Major drawback of this scheme is that it consumes high energy if the event spans multiple grids. And if the composite event involves sensing components from nodes located in neighbouring grids then, only the sink can detect the composite event. And also in the paper they divide the area into grids and assume that each grid forms a cluster, with Cluster Heads (CHs) located in the centre of each grid but, it might not be the favoured cluster head.

K.Ali et al. [5] introduced an approach which makes use of Generalized Hebbian Algorithm (GHA) to determine the relative contribution of each attribute in an event without compromising the accuracy of the results. We use General Hebbian Algorithm (GHA) to find out principal components of a multi attribute input data which has a linear complexity as opposed to quadratic complexity with eigen value decomposition (EVD). This allows online computation of contributions of individual attributes for the detected event. Moreover, the hyper-ellipsoidal clustering based event detection algorithm achieves high detection rates (DRs) and very low false positive rates. The drawback of this system is its high memory usage and low network lifetime.

Jing gao et al. [6] investigated the event coverage quality optimization problem along with its complexity analysis. Algorithm OCQ-Naıve is proposed to compute the best deployment scheme which achieves the optimal coverage quality based on the given budget constraint. This is a novel coverage problem which is different from the traditional ones where deployment costs of sensors, total budget and composite events are not considered. Process several composite events at the same time. Use the "small events" concept. The small events are composite events re-written in a disjunctive normal form. In this way it is possible to find sub-events that are similar between nodes. Such sub-events can be stored in only one node and the information can be shared with other nodes. OCQ-Naive algorithm is used to find the best deployment. The min limitation is that it only works with small number of heterogeneous sensors.

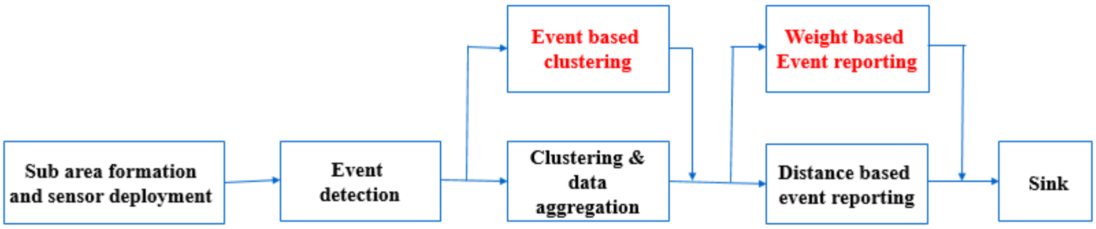
Hejun Wu et al. [7] uses a semantic tree to divide an event in parts called sub-events that describe the spatial and temporal relation between an event and the sub-events. Nodes can decide if they participate or not in a certain task when a dynamic event occurs. The protocol allows nodes to decide which tasks to run according to the sub-event tree, their location, and resources. This mechanism reduces energy consumption, delay, and traffic in the network. DCP uses a sematic-energy-spatial-temporal (SEST) correlation factor to dynamically make decisions about what sub-events to process on each node. The purpose of the SEST factor is to reduce the energy waste of duplicate processing on different sensor nodes and to balance the workload of the sensor nodes. Even though this method has several advantages it has low event reporting rate.

Catalina Aranzazu Suescun et al. [8] proposes an Event-based clustering mechanism. The cluster is formed by the nodes that detect atomic events. The CH collects information from the nodes in the cluster, and is able to detect the composite event. If a composite event is detected, then the CH sends a message to the sink reporting this event through the converge cast tree. Only nodes that detect atomic events and have the residual energy greater than a predefined threshold are candidates to become CH. Nodes that detect atomic events and receive the invitation message from CH, send back an ACK message to confirm that they have joined the cluster. Converge cast tree structure is used for reporting the events to the sink. By using this mechanism total network traffic can be minimized and avoid the congestion. This approach is energy efficient and network life time get improved. Still high reporting delay remains as a limitation to this scheme.

III. System Design

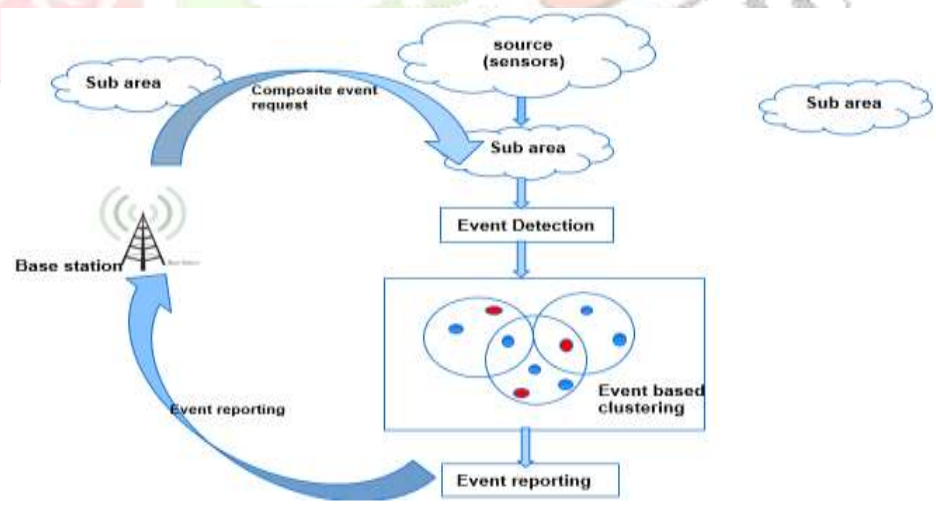
The changes in the values of the attributes measured are known as events. Atomic events occurs due to the changes of a single attribute in the environment. Whereas composite event consists of group of atomic events. In order to perform the event detection sensors are deployed in the required area. The sensor nodes are heterogeneous, since each node is equipped with one or multiple sensing components from the set {s1, s2,… , sn}. Each sensing component can be used to detect an atomic event. The total area is divided into different sub areas prior to the sensor deployment. The events are detected by the sensors based on the composite event request sent by the sink. After the sensing process gets completed the event detected nodes forms clusters in each area. This is known as event based clustering. Then cluster heads are selected for each clusters and the data from the member nodes are aggregated in the cluster heads. Instead of using distance based event reporting the proposed system follows weight based event reporting. In weight based approach the weights are calculated for each cluster heads and arrange them in certain levels. The cluster heads in different levels are communicate with each other and finally the event is reported to the sink.

Fig 3.1 : System Overview



The system is organized in such a way that the uniform distribution of the sensors is maintained throughout. This is achieved by the division of the entire area in to sub areas, each having the same number of sensing nodes. Once this division is done, the events are detected based on the composite event request. Then clustering is performed according to the event detection. After clustering, these events are reported as they are sensed by the sensors to the cluster heads. The cluster heads are selected on the basis of node weights. These cluster heads transmit the reports from one another preserving the minimum energy constraint based on the ECGSR protocol. The reporting continues till the last event is reported to the base station.

Fig 3.2: System Architecture



3.1 Sub Area Formation and Sensor Deployment

The area to be sensed should be divided in to different sub areas before the deployment of sensors. This division is performed based on the node density of sensors according to the respective divided areas. This node density is calculated as,

$$\sigma = N/A$$

Where, N is the total number of nodes and A is the total area to be divided. So, the system contains σ areas, with each of them having "A" number of nodes.

3.2 Sending Composite Event Request

The sink S broadcasts a message CompositeEventRequest E ($(e_1, \delta_1), (e_2, \delta_2), \ldots\ldots(e_k, \delta_k), C_t, C_s, \delta, hops$) in the whole network. The composite event involves atomic events based on the sensing components from the set $\{s_1, s_2, \ldots\ldots, s_n\}$. The field δ_th is the threshold for the confidence of the composite event. More specifically, a composite event should be reported to the sink only if it has confidence greater than or equal to δ_{th}. For example, if not all atomic events are detected by nearby nodes, but if sum of confidence values of the detected atomic events is at least δ_{th}, then the composite event should be reported to the sink. As the CompositeEventRequest message is broadcasted in the network, the nodes passes the message to the nearby nodes. Each node N_i that receives the message from the sink S, increments the hops field, and passes the request. If the same message is received multiple times, then N_i checks the number of hops. If the duplicate message has a smaller number of hops, then N_i accepts it and the number of hops is incremented, and a new message is sent by N_i.

3.3 Event Detection

As each nodes are having different sensing capability due to the heterogeneous property of the system, the request conditions should be checked by each of the nodes. If the node satisfies the parameters sent through composite event request, then the node performs sensing action. This sensing continues till the pre-set time interval terminate. Once it is completed, the detecting value is compared with the threshold value obtained earlier. If the former value is greater, event detection can be confirmed.

3.4 Clustering and Cluster Head Selection

In case of event based clustering, only the nodes which have detected the event participate in cluster formation. It preserves the energy of unused nodes by avoiding the need for using them without the occurrence of an event. The system makes use of a specific node weight assigned to event detector nodes for performing clustering. This node weight (α) is a combination of distance from sink (d), residual energy of the node (q), and the capacity of individual nodes (c). It is calculated as follows:

$$\alpha_i = \frac{q_i c_i}{d(i, BS)}$$

In each area, the node having maximum α will become the cluster head. Such cluster heads request other nodes in the area to join the cluster. In response, these nodes send an acknowledgment for joining the cluster. Likewise, clusters are formed throughout the entire system.

3.5 Event Reporting

Event reporting is aided by the levelling of cluster heads according to their node weights. This is accomplished by dividing the cluster heads in to different ranges with respect to their weights. The cluster heads in the lower range passes the aggregate information to the nearest cluster head in the closest higher level. Similarly this flow goes on till the highest level, from where the event is reported directly to the sink. Such a mechanism reduces the communication overhead between different cluster heads, thereby reducing the reporting delay.

The residual energy of the nodes are recalculated in terms of initial energy (IE), transmission energy (E_{TX}), Receiver energy (E_{RX}), energy used for aggregating the data (E_{Agg}), energy used in the data collection round (E_{DCR}) after reporting the event. It is calculated using the following equation:

$$\text{RE} = \text{IE}-(E_{TX} + E_{RX} + E_{Agg} + E_{DCR})$$

IV. Conclusion

The review done on the Wireless sensor networks have shown that the composite event detection seems to be a prevailing problem which needs to be addressed based on the constraints important to the scenario. This problem can be seen as combination of several problems, based on different metrics, depending on the resource requirements. These include energy, memory, sensing capability which are predominant in determining the efficiency of the network. Within this broad area, the efficient reporting of events is very crucial for the improved performance of the network. The existing algorithms are concentrating more on the network life time, without taking care of event reporting. In the literature, it is seen that algorithms for event reporting are backlogging in terms of the energy requisites. Our work tries to find a feasible solution for this problem by proposing a new algorithm for efficient event reporting, considering the energy requirements as the main scrutinizing factor.

V. Acknowledgment

We thank all our anonymous referees and our friends who have supported in completing this work successfully.

References

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. 2000. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. Proceedings of The 33rd Hawaii International Conference on System Sciences.

[2] S. Lindsey, C. S. Raghavendra, and K. Sivalingam. 2011. Data Gathering in Sensor Networks using the Energy Delay Metric. Proceedings of the 15th International Parallel and Distributed Processing Symposium, pp. 2001 -2008.

[3] Abbasi, Ameer Ahmed, and Mohamed Younis. 2007. A survey on clustering algorithms for wireless sensor networks. Computer communications 30.14:2826-2841.

[4] I. Memon and T. Muntean. 2012. Cluster-based Energy-efficient Composite Event Detection for Wireless Sensor Networks. SENSORCOMM.

[5] K. Ali, T. Anwar, I. H. Naqvy and m. H. Jafry. 2015. Composite event detection and identification for WSNs using General Hebbian Algorithm. IEEE ICC ,Ad-hoc and sensor Networking Symposium.
[6] J. Gao, J. Li, Z. Cai and H. Gao. 2015. Composite event coverage in Wireless Sensor Networks with Heterogeneous Sensors. INFOCOM.
[7] H. Wu, J. Cao and X. Fan. 2016. Dynamic collaborative in-network event detection in wireless sensor networks. Telecommunication Systems, DOI: 10.1007/s11235-015-9981-0.
[8] Catalina Aranzazu Suescun and Mihaela Cardei. 2016. Event-based Clustering for Composite Event Detection in Wireless Sensors Networks. IEEE International Joint Conference on wsn 978-1-5090-5252-3/16.