ABSTRACT
In WSN, for composite event detection some protocol sends all the sensed data to the sink for further processing which causes huge data exchange, unnecessary network traffic and energy consumption. This reduces the energy level of the battery powered sensor nodes and it is often very difficult to change or recharge batteries for these nodes. In most of event monitoring scenarios, users only want to know the event of interest rather than all the events sensed by the sensors. So the sensor should forward a single decision packet corresponding to event of interest rather than forwarding for all. Thus reduced data volume minimizes overall energy consumption of the nodes in the wireless sensor network. To handle these issues, this composite event detection protocol take decision cooperatively among the sensor nodes and only one decision packet is forwarded to sink with sleep-awake policy. This reduces power consumption of the nodes. Further the decision packet is routed such that the node which has maximum battery power and which is closest to sink will be selected as the next hop. This minimizes the rate of failure of link due to node failure thus improved life time of the network and efficient use of energy is achieved.

General Terms – Algorithm, sleep-awake policy.

Key Words- Wireless sensor network, composite event detection, sleep-awake policy

1. INTRODUCTION
As a special class of wireless sensor networks [1], event-driven wireless sensor networks (EWSNs) are composed of large numbers of sensor nodes that are deployed in the terrain to sense physical phenomena of interests (PoIs) [2]. The main purpose of EWSNs is the accurate notification of the PoI to the policy decision-maker. Thus, EWSNs will have the capability to transmit the sensor data, including critical data (e.g., the location of an event), to one or more centralized sinks who are expected to perform real time processing and to make accurate decision quickly. The drawbacks of most of the existing methods are the real-time requirement is not taken into account, and the amount of the exchange data may be huge as shown in figure 1. For some applications in WSN, decision of the occurrence of any event is depend on occurrence of two or more events. For example to detect the fire one should have the knowledge that high temperature and smoke are detected.

For such composite event detection some protocols send all the sensed data to the sink for further processing which causes huge data exchange, unnecessary network traffic and energy consumption. This reduces the energy level of the battery powered sensor nodes and as stated earlier, it is often very difficult to change or recharge batteries for these nodes.

In most of event monitoring scenarios, users only want to know the event of interest rather than all the events sensed by the sensors. So the sensor should forward a single decision packet corresponding to event of interest rather than forwarding for all. Thus reduced data volume minimizes overall energy consumption of the nodes in the wireless sensor network. If the occurrence of the event is very infrequent in the network we can further extend the policy by keeping the sensor node to sleep mode if sensors are not transmitting data or idle for some predefined time, TA.

This paper proposes an energy efficient composite event detection protocol with sleep-awake policy of sensor nodes.

The rest of the paper is organized as follows. Section II briefly reviews related work on event detection and some routing protocol for event notification in EWSNs. Section III gives the detail of the protocol design including primary event detection and emergency routing and sleep awake policy. Finally, Section IV concludes the paper and gives the future work.

2. RELATED WORKS
For event monitoring in EWSNs, we have to solve two problems, one is the accurate event detection, and the other is the reliable and fast transmission. However, most of the conventional protocols focus on the transmission without...
In this proposed work we are trying to develop a composite event detection scheme with sleep-aware policy in order to reduce the energy consumption of the sensor nodes.

Figure 2. Working of EEDP

In paper [10], the authors have discovered a novel routing protocol for wireless datagram networks that uses the positions of routers and a packet’s destination to make packet forwarding decisions. GPSR makes greedy forwarding decisions using only information about a router’s immediate neighbors in the network topology. The algorithm consists of two methods for forwarding packets: greedy forwarding, which is used wherever possible, and perimeter forwarding, which is used in the regions greedy forwarding cannot be.

In GPSR, packets are marked by their originator with their destinations’ locations. As a result, a forwarding node can make a locally optimal, greedy choice in choosing a packet’s next hop. Specifically, if a node knows its radio neighbors’ positions, the locally optimal choice of next hop is the neighbor geographically closest to the packet’s destination. Forwarding in this way is done, until the destination is reached. Upon receiving a greedy-mode packet for forwarding, a node searches its neighbor table for the neighbor geographically closest to the packet’s destination. If this neighbor is closer to the destination, the node forwards the packet to that neighbor. When no neighbor is closer, the node marks the packet into perimeter mode and then perimeter forwarding comes into picture. GPSR forwards perimeter-mode packets using a simple planar graph traversal.

3. PROTOCOL DESIGN AND PRELIMINARY NOTATIONS

Following are the basic requirements for the implementation of EEDP along with sleep-aware policy:

1. Detection of Primary and Composite Event
2. Forwarding Alarm Packet to Sink
3. Choosing the hop for routing the Alarm Packet.
4. To put nodes in Sleep Mode which are not active.

3.1 Detection of Primary and Composite Event

In the event occurring area, each node broadcasts its primary detection result to make a final decision corporately. For composite event detection protocol uses Primary Detection considering the accurate event detection, leading to a waste of scarce sensor resources. ESRT (Event-Sink Reliable Transport), a transport solution, is developed in [4] to achieve reliable event detection by proper congestion control protocol based on rate adjustment. ESRT adjusts the reporting frequency such that the observed event reliability, which is defined as the number of packets received at the sink, is higher than the desired value while avoiding congestion. However, the self-adjusting method is that all sensor nodes are controlled at once, treating regions of interest in the same way as uninteresting regions explained in[6]. Lige Yu proposed an energy-driven scheme where each sensor node sends out its 1-bit decision if that decision exceeds a predetermined detection accuracy threshold in [5], and sends out all its observations otherwise. This scheme sets a restriction for the maximum number of observations collected by each sensor, which avoids the potential delay at sensors which causes the consequent problem of asynchronism due to large number of observations in the case of sequential detection. Composite event is first proposed in [7] to make accurate event detection. To ensure the quality of surveillance, some applications require that if an event occurs, it needs to be detected by at least k sensors, where k is a user-defined parameter. The basic idea of Composite Event Detection scheme [8] is that, there is a gateway node which is responsible for making a conclusion and reporting it to the user if an event happens. This gateway node is properly selected and every sensor in the network has a chance to serve as a gateway node in order to balance the energy consumptions. During the network operation time, once a sensor detects that the current sensed value is over the threshold of its monitored property, it sends one bit ‘1’ instead of the sensed value to a gateway node. If a gateway node receives a ‘1’, it checks if the compound propositional function which defines an event E derives a TRUE value. If so, it immediately sends a warning to the Base Station. Thus energy is saved by sending only a single packet instead of sending all sensed data.

In the paper [3], EEDP the authors have proposed a protocol which works as follows: In the event occurring area, each event detecting sensor node broadcasts its primary detection result to make a final decision corporately. And then the final decision-made by a sensor node will choose the next hop using the underlying routing protocol to forward a single alarm packet. Thus the traffic volume is reduced as only single decision packet is transmitted. Figure 2 shows the working of EEDP. In paper [9] authors have described T-MAC; a contention based medium access protocol for wireless sensor networks. Authors have discovered that there are some characteristics of WSN, like low message rate, insensitivity to latency that can be exploited to reduce energy consumption by introducing a active sleep duty cycle. Authors have researched a novel way to reduce the amount of energy wastage of node during the idle listening. They have suggested that keep the node in sleeping mode when no activity is detected for certain amount of time. This activation period and sleeping period can be vary dynamically depending upon the traffic in the network. When the nodes go to sleeping mode they maintain the synchronization by sending a SYNC packet, which tells the wake up timing of the node to the neighbor.

When a SYNC packet is received by any node, it schedules itself according to the timing defined in the SYNC. In this way the synchronization is maintained in the T-MAC protocol.
Procedure (PDP). PDP uses the Single Decision Rule (SDR) for single event of interest and Composite Decision Rule (CDR) for final composite event of interest. Algorithm used for detection of primary and composite event is Local Broadcast Algorithm.

3.1.1 Primary Detection procedure (PDP)

PDP uses the Single Decision Rule (SDR) for local event of interest and Composite Decision Rule (CDR) for final composite event of interest as follows:

- Primary Decision Rule

The local atomic decision of a node $i$ is denoted by $\mu_{i,m}$. The independent signal $\mu_{i,m}$ is obtained as

$$x_{m}^{i} = \begin{cases} 
\omega_{m}^{i} & \text{if } H_{0}^{i}, (E \text{ is absent}); \\
ft(r_{i}) + \omega_{m}^{i} & \text{if } H_{1}^{i}, (E \text{ is absent}); 
\end{cases} \quad (1)$$

Where $\omega_{m}^{i} \sim N(0, \sigma_{m}^{2})$ is the noise that follows a normal distribution with mean 0 and standard deviation $\sigma_{m}$; $r_{i}$ is the distance between node $i$ and the phenomenon of interest; and $f$ is a function that monotonically decreases with increasing $r_{i}$. For each sampled signal $x_{m}^{i}$, node $i$ makes a per sample binary decision $\mu_{i,m}^{E} \in \{0, 1\}$ by the Single Decision Rule. In SDR every single ESN decides the presence or absence of event and generates local atomic binary decision.

This atomic binary decision is broadcasted to other ESN. SDR is given as:

$$\mu_{i,m}^{E} = \begin{cases} 
1 & \text{if } x_{m}^{i} \geq \Gamma_{i,m}^{E} \\
0 & \text{otherwise} 
\end{cases} \quad (2)$$

Where

$$\Gamma_{i,m}^{E}$$

is the per-sample threshold of node $i$ for the $m^{th}$ atomic event $E_{m}$. 

- Composite Decision Rule

Further using CDR each ESN aggregates all observations from all other ESN, and generates an alarm packet. The composite decision rule is given by

$$\Delta^{i} = \begin{cases} 
1 & \text{if } \mu_{1}^{i}, \mu_{2}^{i} \AND \mu_{3}^{i} \AND \ldots \AND \mu_{|M|}^{i} = 1 \\
0 & \text{otherwise} 
\end{cases} \quad (3)$$

PDP is implemented in Local Broadcast Algorithm to broadcasting the local event of interest. Local Broadcast Algorithm defines two types of nodes, ESN for sensing the local event and EFN for forwarding the alarm packet.

When a ESN listens the alarm packet from other ESN, it immediately suspends sending local decision. The ESN, who generates the alarm packet, forwards the alarm packet to destination using Local Broadcast Algorithm. When EFN listens the alarm packet, it will send it towards sink and wait for the acknowledgement.

3.1.2 Local Broadcast Algorithm

The primary detection of the event consists of detection of atomic binary event and a combined decision of composite event. This procedure is explained in PDP and the algorithm used for primary detection is Local Broadcast Algorithm as follows:

- When timer $T$ expires, it will keep silent and waits for the acknowledgement.

Where $\Delta^{i}$ is the final decision result of node $i$

- Algorithm:

1: while $t \leq T$ do
2: Step 1: Set a decision timer $T$.
3: Step 2: Node $i$ keeps sending its own primary decision message and overhearing decision messages from neighbors and then node $i$ goes to step 3.
4: Step 3: Using CDR, each node could make the decision $\Delta^{i}$. If $\Delta^{i} = 1$, there must exist certain abnormal event. Node $i$ goes to step 4. Otherwise, node $i$ goes to step 5.
5: Step 4: Node $i$ will generate and forward the alarm packet $\Psi^{i}$, namely alarm to the destination immediately using Fast Broadcast Algorithm.
6: Step 5: Node $i$ will forward $MSG^{\text{local}}_{j}$, to its neighbors and keep in overhearing the $MSG^{\text{local}}_{j}$ from other nodes.
7: Step 6: When node $i$ receives a local primary detection message $MSG^{\text{local}}_{j}$, from node $j$, the emergency information $[\mu_{1}^{i}, \mu_{2}^{i}, \ldots, \mu_{|M|}^{i}]$ of node $i$ will be updated as $\mu_{i,m}^{E} = \mu_{j,m}^{E} \mu_{i,m}^{E}, \forall m \in M$, and then goto step 3.
8: Step 7: When node $i$ receives $\Psi$ from other node, it will suspend to send $MSG^{\text{local}}_{j}$.
9: end while
10: Step 8: When timer $T$ expires, it will keep silent and clear the value of $\Gamma^{i}$. $m \in M$.

- Elaboration of Local Broadcast Algorithm

Firstly, node $i$ sets a decision timer $T$ (Step 1). During this decision time, node $i$ keeps sending its own primary decision message and overhearing decision messages from neighbors (Step 2). Using CDR, each node $i$ could make the decision

| Table 1: Notations And Symbols |
|-------------------------------|----------------|
| $N$                           | number of sensor nodes |
| $N_{i}$                       | the set of neighboring nodes of node $i$ |
| $\Phi$                        | a composite event |
| $M$                           | number of component of sensor nodes |
Δₐ (Step 3). If Δₐ = 1, which means that the final decision determines that event occurs, node i will generate and forward the alarm packet Ψ, namely alarm to the destination immediately (Step 4). Otherwise, it will generate and forward MSGₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐₐ事儿
sleeping (Step 3). The procedure is repeated till the packet is reached to sink.

**3.4 To put nodes in Sleep Mode which are not active.**

The energy consumption of the nodes can be minimized by keeping the nodes to sleep mode when there is no traffic to be received or transferred for some constant predefined time. This activation period and sleeping period can be varying dynamically depending upon the traffic in the network.

**3.4.1 Sleep Awake Procedure**

Sleep Awake Procedure used for finding the nodes which are idle for TA time duration and keep that node to sleep mode. Before going to sleep mode, When the nodes go to sleeping mode they maintain the synchronization by sending a SYNC packet, which tells the wake up timing of the node to the neighbor. When a SYNC packet is received by any node, it schedules itself according to the timing defined in the SYNC.

**3.4.2 Sleep Awake Algorithm**

There are many algorithms are available for sleep-awake policy. Following could be the basic steps to achieve the sleep-awake policy.

This Sleep-awake Algorithm is used for keeping the nodes in sleep mode if it is not active for TA time period. Due to this the idle listening problem of sensor nodes is minimized.

- **o Input parameters:** node, TA, SYNC
  - Where TA is the maximum allowable idle time for a node i.
  - SYNC is the packet containing information about the awake time and the energy level of the node.
- **Output parameters:** SYNC, node
  - Algorithm:
- **Step 1:** When the active period of the node start it tries to receive or send the data.
- **Step 2:** If no data is to be received or transmitted from node i for time TA, node i creates SYNC packet and broadcast it to all its neighbor and go to sleep.
- **Step 3:** At the end of sleep period node sends SYNC to all of its neighbor and go to step 1.

**Elaboration of Sleep-awake Algorithm**

The node can receive or transmit in its active period (Step 1). When node doesn’t have anything send or receive for time TA, it will inform to neighbor about its next active frame and go to sleep (Step 2). At the end of sleep period node sends SYNC to all of its neighbor (Step 3) and go to step 1.

**4. CONCLUSION AND FUTURE WORK**

From above study of EEDP and other routing as well as sleep-awake protocol we feel that EEDP along with sleep awake protocol could prove better approach for improving life of the network. Considering the feasibility of EEDP implementation an addition of sleep-awake scheduling scheme is also feasible.

In future work we will implement the scheme with dynamic topology of sensor nodes.

**REFERENCES**


