In wireless sensor networks (WSNs), an ideal transport layer needs to support reliable message delivery and provide congestion control in an efficient manner in order to extend the lifetime of a WSN. The main use of transport protocol in WSNs is to overcome the congestion and the reliability with energy efficiency. In this chapter a reliable and energy efficient transport protocol (REETP) is presented, which mainly focuses on the reliability and energy efficiency. REETP consist of an Efficient Node Selection Algorithm to determine a set of efficient nodes called E-Nodes, which form a near optimal coverage set with largest area and highest residual energy level. The key idea of REETP is to transfer encoded packets using LT codes from the source to the sink block by block and each block is forwarded to an E-node. After receiving encoded packets, the E-node tries to reconstruct the original data packets and it encodes the original data packets again and relays them to the next E-node until it reaches the sink. By simulation results, it is shown that the REETP has more packet delivery ratio with reduced packet loss and energy consumption.

Rest of the chapter is organized as follows. Section 5.1 presents introduction. The assumption and terminology for the developed protocol are given in Section 5.2. Section 5.3 deals with REETP. Simulation results and discussion of REETP is explored in section 5.4. Finally work of this chapter is summarized in Section 5.5.

5.1. Introduction
The purpose of the transport protocols is to provide reliable data transfer services between end users, thus relieving the upper layers’ responsibility for providing reliable and cost-effective data transfer. The transport layer, which is the second highest WSNs protocol stack, responds to the services requested from the application layer and issues service requests to the network layer. The transport layer provides dependable data transfers between hosts. It is usually responsible for end-to-end error recovery and flow control and for ensuring complete data transfer.
The transport protocols in WSN should support

- Reliable message delivery
- Congestion control
- Energy efficiency

The following are the suggestions for the requirement of transport layer protocol in WSNs.

- Loss detection and recovery can be handled below the transport layer and mitigated using data aggregation.
- Congestion is not an issue because sensor nodes spend most of the time sleeping resulting in sparse traffic in the network. Generally the deployment of sensor nodes produces congestion in WSNs in the contradiction to the above arguments against the need for a transport layer protocol. In the absence of congestion control, data from sensor nodes to sink may suffer from channel contention, which in turn decreases the ability of the sensor nodes to deliver data to the sink. Since the layers under the transport layer do not provide guaranteed end-to-end reliability, it is inadequate to depend upon the loss detection and reliability techniques, in the situation where data’s are delivered reliably in WSNs [78].

Like other networks, WSNs should have a transport layer in order to posses reliable message delivery and congestion control. An ideal transport layer needs to support reliable message delivery and provide congestion control in an efficient manner in order to extend the lifetime of WSNs. [78]

The following are the some of the transport protocols developed in the wireless sensor networks:

1. TCP/IP – Transmission Control Protocol
2. PCCP - Priority-based Congestion Control Protocol
3. STCP - Sensor Transmission Control Protocol [80]
4. MQTT - Message Queuing Telemetry Transport [81]
5. PORT - Price-Oriented Reliable Transport Protocol [82]
6. PSFQ - Pump Slowly, Fetch Quickly [83]
7. RMST - Reliable Multi-Segment Transport [84]
8. ESRT - Event to Sink Reliable Transport [85]

Except STCP, the above-mentioned protocols consider either congestion control or reliability guarantees. Some protocols use end-to-end and others hop-by-hop controls.
and also some guarantees event reliability and others provide packet reliability. The following are the two fundamental demerits of the existing protocols for WSNs

- Since sensor nodes in WSNs can be installed with different kinds of sensors and used in different geographical locations, it may have different priorities.
- The existing transport protocols for WSNs assume that single path routing is used in the network layer without considering the multipath routing.

Summary of requirements of a transport layer protocol for sensor networks is as follows

- **Generic:** The transport layer protocol should be independent of the application, Network and MAC layer protocols to be applicable for several deployment scenarios.
- **Heterogeneous data flow support:** Continuous and event-driven flows should be supported in the same network.
- **Controlled variable reliability:** Some applications require complete reliability while others might tolerate the loss of a few packets. The transport layer protocol should leverage this fact and conserve energy at the nodes.
- **Congestion detection and avoidance:** The congestion detection and avoidance mechanism helps in reducing packet retransmissions, thereby conserving energy.
- **Base station controlled network:** Since sensor nodes are energy constrained and limited in computational capabilities, majority of the functionalities and computation intensive tasks should be performed by the base station.
- **Scalability:** Sensor networks may comprise of large number of nodes, hence the protocol should be scalable.
- **Future enhancements and optimizations:** The protocol should be adaptable for future optimizations to improve network performance and support new applications.

The main use of transport protocol in the wireless sensor networks is to overcome the congestion and the reliability with energy efficiency. To overcome the above issues the following protocol is developed.

### 5.2. Assumption and Terminologies for REETP

The key idea of the Reliable and Energy Efficient Transport Protocol (REETP) is based on the following assumption and terminology.
5.2.1 Efficient Node Selection Algorithm

In the efficient node selection algorithm, a set of efficient nodes called E-Nodes is determined, which form a near optimal coverage with set with largest area and highest residual energy level.

Also it is assumed that sensors are able to monitor their residual energy because many electronic devices are equipped with energy monitoring functions. The energy level (EL) of sensors $s_i$ at the beginning of update interval (UI), denoted by $EL$ is calculated as:

$$EL = \frac{RE(UI)}{IE}$$  \hspace{1cm} (5.1)

Where $IE$ is the initial energy corresponding to fully charged battery and $RE$ (UI) is the residual energy of sensors $s_i$ at the beginning of the update interval.

In each iteration E-Node Algorithm selects one node from the unselected sensors, which covers the largest area with highest residual energy level. For this purpose, a weight value is defined to represent the weight of a sensing region of a sensor based on its residual energy. For a given region, the weight value based on the residual energy level of a sensor is

$$W(R_i) = EL \times A(R_i)$$  \hspace{1cm} (5.2)

Where $EL$ is the energy level given in (5.1) and $A(R_i)$ is the area of sensing region $R_i$. Then, the gain of selecting each sensor using the weight value is calculated. To do this, first find the size of the area that can be covered by sensor $s_i$ and has not been covered yet. Consider the sensor $s_i$ with sensing region $R_i$. Let $R_{CS}$ be the area that sensors of $C$ covered so far, i.e., $R_{CS} = \bigcup_{s_j \in C} R_j$. Beneficial area of $s_i$ is defined to be the region inside the sensing field which has not been covered, i.e., $RB = (R_i \cap A) / R_{CS}$.

Hence, gain function for sensor $s_i$ is the total weight of its beneficial area, which is given as:

$$G(S_i) = W(RB), \quad s_j \in C$$  \hspace{1cm} (5.3)

Where $G(S_i)$ is the gain function and $RB$ is the beneficial area.

E-Node Algorithm is to find a near-optimal coverage set $C$. Then each member of the set $C$ is known as an E-node.

87
E-Node Algorithm

Let \( C = \emptyset \)
Let \( R_{CS} \) be the total sensing region of \( C \)
Let \( S - C = \{s_1, s_2, \ldots, s_n\} \)
\( G_{max} = 0 \)
For each \( s_i \in S - C \)
Calculate the energy gain of \( s_i \)
\[ G(s_i) = \sum_{a_j \in (R \cap A) \cap RCS} W_i(a_j) \]
If \( (EL_B \geq G_{max}) \)
\[ G_{max} = G \]
\[ temp: = s_i \]
End if
End for
\[ C = C \cup temp \]
If \( A \subseteq R_{CS} \) then
Return \( C \)
Else
Repeat from 3
End if

5.2.2. Forward Error Correction using LT Codes

LT codes are rate less because the number of encoding symbols, which are generated from the data, is unlimited. The required encoding symbols can be generated immediately. From any set of the generated encoding symbols, the decoder can recover an exact copy of the data. Thus the required encoding symbols can be generated without depending on the loss model on the erasure channel. In order to recover the data, the generated symbols are sent over the erasure channel until the adequate number has been arrived at the decoder. The LT codes are near optimal with respect to any erasure channel because the decoder can recover the data from the near optimal number of possible encoding symbols. Moreover, as a function of the data length, the encoding and decoding times are very efficient. When compared with the previous erasure codes, LT codes provide various advantages for different types of data delivery applications. Using LT codes, the minimal number of encoding symbols can be generated and send the packets to the receivers. The minimal number of encoded symbols is required to recover the original data from each receiver [86].
Robust distributed storage, delivery of streaming content, delivery of content to mobile clients in wireless networks, peer-to-peer applications and delivery of content
along multiple paths in order to ensure resiliency to network disruptions, are some of the other applications of the LT codes [86].

5.2.3. LT Process
The preferable length $L$ of the encoding symbols can be selected. Due to the overheads with the accounting operations, the overall encoding and decoding is more efficient for larger values of $L$ and this value does not have any influence on the history. Sometimes the length $L$ is selected to be closer to the length of the packet payload in case of transport applications [86].

5.2.4. Encoding: The data of length $N$ is partitioned into $K=N/L$ input symbols such that each transport symbol is of length of $L$. Each encoding symbols are connected with a key. In order to produce the degree and set of neighbors of the encoding symbol, both the encoder and decoder applies the same function to the key. In order to generate an encoding symbol, the encoding symbol may choose each key randomly and this key is passed to the decoder along with the encoding symbols. Alternatively, a deterministic process may produce each key, e.g., each key may be larger than the previous key. The encoder and decoder have the access to the same set of random bits. In order to produce the degree and the neighbors of the encoding symbol, each key is used as the seed to a pseudo-random generator, which uses these random bits.

5.2.5. Decoding: For a given group of encoding symbols and some illustrations of their associated degrees and sets of neighbors, the decoder recovers the input symbols repeatedly using the following rule as long as it applies. Since the neighbor is a copy of the encoding symbol, it can be recovered immediately if there is at least one encoding symbol, which has exactly one neighbor. The value of the recovered input symbol is XORed into any remaining encoding symbols that also have that input symbol as a neighbor. The recovered input symbol is removed as a neighbor from each of these encoding symbols and one to reflect this removal decreases the degree of each such encoding symbol.

5.3. Reliable and Energy Efficient Transport Protocol (REETP)
The idea of REETP is to transfer encoded packets using LT codes, block by block. In order to reconstruct the original data packets, the receiver has to receive sufficient
encoded packets. The REETP has to guarantee that the receiver can receive enough encoded packets in such a limited time interval. By setting the block size $n$ (i.e., the number of original data packets in each block) appropriately, REETP can control the transmission time and allow the receiver to be able to receive enough packets in order to reconstruct original block even in node motion.

In REETP, a data source first groups data packets into blocks of size $n$. Then the source encodes these blocks of packets, and sends the encoded blocks into the network. The data packets are forwarded from the source to the sink block by block, and each block is forwarded to an E-node. In each E-node relay, the sender first estimates the number of packets needed to send for the E-node to reconstruct the original packets. This number is called as “MaxPacket”. Within the MaxPacket, the sender pushes the encoded packets to the network fast. When the packet is reached, the sender slows down packet transmission, waiting for a positive feedback from the E-node. After receiving encoded packets, the receiver tries to reconstruct the original data packets. If the reconstruction is successful, it sends back a positive feedback. Upon the reception of a feedback, the sender stops sending packets, while the E-node encodes the original data packets again and relays them to the next E-node until the sink is reached.

The operations performed on the sender and the receiver (E-node) is described in the following. The overall working of the REETP is shown in Figure 5.1.
5.3.1. REETP Sender
1) Estimates the MaxPacket.
2) Encodes a block using LT codes.
3) Pumps encoded packets fast in a random order within the MaxPacket.
4) Sends encoded packets slowly outside the MaxPacket until receiving a positive feedback from the E-node.

5.3.2. REETP Receiver
1) Keeps receiving packets until it can reconstruct the original data packets, and sends a positive feedback to the Sender.
2) Encodes the reconstructed packets again and relay them to the next E-node.
The above description shows that REETP reduces the burden of sender and receiver by requiring only one feedback per block. The sender has no additional responsibilities except encoding and injecting packets, and the receiver only needs to send one feedback after reconstructing the original packets.

5.4. Simulation Results and Discussion
NS2 simulator is used to simulate the proposed protocol. In the simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (DCF) of IEEE 802.11 as the MAC layer protocol is used.
In the simulation, 100 sensor nodes are deployed in a 1000 m x 1000 m region for 50 seconds simulation time. All nodes have the same transmission range of 250 meters. The simulated traffic is Constant Bit Rate (CBR). The simulation scenarios are summarized in the following table.

<table>
<thead>
<tr>
<th>Number of Nodes</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area Size</td>
<td>1000 X 1000</td>
</tr>
<tr>
<td>Mac</td>
<td>802.11</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>50 sec</td>
</tr>
<tr>
<td>Traffic Source</td>
<td>CBR</td>
</tr>
<tr>
<td>Packet Size</td>
<td>512</td>
</tr>
<tr>
<td>Transmit Power</td>
<td>0.360 w</td>
</tr>
<tr>
<td>Receiving Power</td>
<td>0.395 w</td>
</tr>
<tr>
<td>Idle Power</td>
<td>0.335 w</td>
</tr>
<tr>
<td>Initial Energy</td>
<td>3.1 J</td>
</tr>
<tr>
<td>Number of sources</td>
<td>2,4,6,8</td>
</tr>
<tr>
<td>Transmission Rate</td>
<td>250,500,750 and 1000 kb.</td>
</tr>
</tbody>
</table>

| Table 5.1. Simulation Parameters |
5.4.1. Performance Metrics
Performance of the proposed REETP protocol is compared with A MAC-aware Energy Efficient Reliable Transport Protocol (MAEERTP) [43] for WSNs. Mainly the performance according to the following metrics is evaluated:

- **Average Energy Consumption**: The average energy consumed by the nodes in receiving and sending the packets is measured.
- **Packet Delivery Ratio**: It is the ratio of the fraction of packets received successfully and the total number of packets sent.
- **Average Packet Loss**: It is average number of packets lost at each receiver and the sink.

5.4.2. Simulation Results
The REETP protocol’s performance is measured based on the two conditions namely by varying number of sources and varying the transmission rate.

**Varying Number of Sources**
In the first experiment, in order to study the impact of increased the number of sources, vary the number of sources as 2, 4, 6 and 8 and measure the performance of the protocols.

![Figure 5.2. Numbers of Sources Vs. Packets Lost](image)
Figure 5.2 shows the packet lost obtained with REETP protocol compared with MAEERTP protocol. It shows that the packet lost is significantly less than the MAEERTP, as sources increases.

![Figure 5.2](image)

**Figure 5.2.** Packet Lost Obtained with REETP Protocol Compared with MAEERTP Protocol

Figure 5.3 shows that the packet delivery Ratio (PDR) for REETP increases, when compared to MAEERTP protocol.

![Figure 5.3](image)

**Figure 5.3.** Number of Sources Vs. Delivery Ratio

Figure 5.4 shows that the packet delivery Ratio (PDR) for REETP increases, when compared to MAEERTP protocol.

![Figure 5.4](image)

**Figure 5.4.** Number of Sources Vs. Energy

Figure 5.3 shows that the packet delivery Ratio (PDR) for REETP increases, when compared to MAEERTP protocol.
Figure 5.4 shows that the average energy consumed by the nodes in receiving and sending the data. Since REETP make use of energy efficient scheduling, the values are considerably less in REETP when compared with MAEERTP protocol.

**Varying the Transmission Rate**

In the second experiment, in order to study the performance of increased traffic sending rate, vary the transmission rate as 100, 200, 300, 400 and 500Kb to measure the performance of the protocols.

![Energy vs Rate Graph](image.png)

**Figure 5.5. Rate Vs. Energy**

Figure 5.5 shows that the average energy consumed by the nodes in receiving and sending the data. Since REETP make use of energy efficient scheduling, the values are considerably less in REETP when compared with MAEERTP protocol.
Figure 5.6. Rate Vs. Packet Lost

Figure 5.7. Rate Vs. Delivery ratio

Figure 5.6 shows the packet lost obtained with proposed REETP protocol compared with MAEERTP protocol. It shows that the packet lost is significantly less than the MAEERTP, as rate increases.

From Figure 5.7, it is seen that the packet delivery Ratio (PDR) for REETP increases, when compared to MAEERTP protocol.
5.5. Summary
In this chapter we have presented an energy efficient and reliable protocol, i.e., reliable and energy efficient transport protocol (REETP). The REETP consist of an Efficient Node Selection Algorithm to determine a set of efficient nodes called E-Nodes, which form a near optimal coverage set with largest area and highest residual energy level. The objective of REETP is to transfer encoded packets using LT erasure codes from the source to the sink block by block and each block is forwarded to an E-node. The sender first estimates MaxPacket, which is the number of packets needed to send for the E-node to reconstruct the original packets. When the packet reached at its destination, the sender slows down pack transmission, waiting for a positive feedback from the E-node. After receiving encoded packets, the receiver tries to reconstruct the original data packets. If the reconstruction is successful, it sends back a positive feedback. Upon the reception of a feedback, the sender stops sending packets, while the E-node encodes the original data packets again and relays them to the next E-node. In next chapter we present route alteration based congestion avoidance methodologies for WSNs.