EE550 Design and Analysis of Computer Communication Networks

A Performance Comparison and Analysis of Energy-Efficient Data Routing in Wireless Sensor Networks

Term Project
Due: Dec 06, 2010

=================================
Professor John Silvester
Hsiao-Nung Chang  9790-1443-68
Wei-I Wang       4885-2694-83
Yu-Yu Lin        7996-8793-72
Content

Figure of Content..........................................................................................................................2
Table of Content..........................................................................................................................2
1. Introduction.........................................................................................................................2
2. Energy Analysis of Routing Protocols.................................................................................3
3. Proposed Schemes..................................................................................................................4
   3.1 LEACH (Low-Energy Adaptive Clustering Hierarchy) .................................................4
   3.2 PEGASIS(Power-Efficient Gathering in Sensor Information Systems)...................5
4. Performance comparison and analysis ..................................................................................6
   4.1 Parameter set up.............................................................................................................6
   4.2 Simulation assumption.................................................................................................7
   4.3 Simulation results and discussion ...............................................................................7
      4.3.1 Drawbacks and Advantages ..............................................................................8
5. Improvement for proposed schemes....................................................................................9
   5.1 Improvement for LEACH............................................................................................9
   5.2 Improvement for PEGASIS.........................................................................................10
6. Conclusions.........................................................................................................................11
7. References...........................................................................................................................12

Figure of Content

Figure 1. Simple linear network............................................................................................4
Figure 2. Dynamic clusters: Cluster-head nodes at time t_i and t_i + d...............................5
Figure 3. Chain is constructed using the greedy algorithm..............................................6
Figure 4. Token passing approach in PEGASIS.................................................................6
Figure 5. Node death percentage vs Number of Rounds......................................................8
Figure 6. Number of Rounds vs Coverage ratio.................................................................8
Figure 7. Node death percentage vs Number of Rounds......................................................8
Figure 8. Number of Rounds vs Coverage ratio.................................................................8
Figure 9. Node death percentage vs Number of Rounds......................................................10
Figure 10. Number of Rounds vs Coverage ratio..............................................................10
Figure 11. Binary Chain-based PEGASIS..........................................................................10
Figure 12. Simulation Interface (C++ Builder 6)..................................................................11

Table of Content

Table 1. Cost Comparison. 200 nodes, BS at (50, 200), 5% Cluster heads.......................9
Table 2. Cost Comparison. 200 nodes, BS at (50, 200), 6% Cluster heads.......................11
Table 3. Performance comparison in term of different metric.................................11
1. Introduction

A wireless sensor network (WSN) is a collection of sensor nodes distributed over a particular area to monitor physical or environment conditions. Sensor nodes establish a network that they can exchange data and transmit data back to a base station (BS) [6]. An application for a wireless sensor network can be in both civil and military; for example, battlefield surveillance [6]. Sensor node has a limited battery power and capability. Power consumption has been one of the important factors in mobile wireless network. To make a wireless network operate autonomously, each node in the network must be able to provide adequate communication while consuming as minimum power as possible. There have been many research papers in this field that address such an issue and propose their own algorithm to efficiently minimize the power consumption and effectively maximize the lifespan of the network. In this paper, some classic protocols are compared and discussed. We start from the conventional protocols such as direct transmission protocol that sensor node sends data directly to a distant base station and consumes its energy rapidly. That leads to Minimum-transmission-energy (MTE) routing protocol that reduces distance for transmitting packet to BS by routing a data packet through multiple intermediate nodes [1]. Following by section 2, we briefly introduce two classical energy-efficient algorithms, which is LEACH [1] (low-Energy Adaptive Clustering Hierarchy) and PEGASIS [2] (Power-Efficient Gathering in Sensor Information Systems). LEACH introduces clustering based protocol, where sensor nodes are grouped in several clusters and have randomized rotation of cluster-heads that will transmit a data to BS. PEGASIS is a chain-based protocol built on top of idea from LEACH, which nodes communicate only to its neighbor and takes turn to be leader to send data back to the BS. In section 4, most importantly, the performances among these protocols are compared and analyzed by using C++ to build our simulator to implement these protocols. In section 5, interesting enough, we achieve better performance by improving the original schemes with some ideas.

2. Energy Analysis of Routing Protocols

Two conventional routing protocols in wireless network that we will discuss in this section are direct communication and minimum-transmission-energy routing protocol (MTE). In direct communication, each node connects and transmits data directly to the base station. If base station locates far away from sensor nodes, sensor node will deplete its battery quickly and shortens the system lifetime. In minimum-transmission-energy, node routes data to the base station via its neighbors. Each node will act as a router that route a received packet from one neighbor to another. This technique reduces the transmit amplifier energy because distance between node is shorter than the distance between node and the base station. This concept can be expressed mathematically as following:

\[
E_{Tx}(K, d) = E_{elec} \times k + \varepsilon_{amp} \times k \times d^2
\]

\[
E_{Rx}(K) = E_{Rx-elec}(k) = E_{elec} \times k
\]

Where \(E_{elec}\) is the energy spent in transmitting and receiving data for a sensor; \(\varepsilon_{amp}\) is the energy spent in amplifying. Therefore, the energy is consumed for a sensor to transmit k-bits data over d meters is defined in (1), and that for receiving data is defined in (2). Node A will route a packet to C via B if the following equation holds:

\[
E_{Tx-amp}(k, d=d_{AB}) + E_{Tx-amp}(k, d=d_{BC}) < E_{Tx-amp}(k, d=d_{AC})
\]
However, we need to take into an account that, in minimum-transmission-energy, each message needs to go through multi-hops and the total energy might ends up greater than direct transmission.

Assume we have a linear network as shown in Figure 1. The network consists of $n$ node, each node has $r$ distance. Therefore, the energy used in direct transmission is:

$$E_{\text{direct}} = E_{\text{TX}}(k, d = n*r) = E_{\text{elec}} * k + \varepsilon_{\text{amp}} * k * (nr)^2$$

In MTE routing, we have:

$$E_{\text{MTE}} = n * E_{\text{TX}}(k, d = r) + (n-1)E_{\text{RX}}(k) = n(E_{\text{elec}} * k + \varepsilon_{\text{amp}} * k * (nr)^2)$$

Therefore, direct transmission uses less energy than MTE routing if:

$$E_{\text{direct}} < E_{\text{MTE}} \quad k(E_{\text{elec}} + \varepsilon_{\text{amp}} n^2 r^2) < k((2n - 1)E_{\text{elec}} + \varepsilon_{\text{amp}} n r^2)$$

According to simulation result from [1], it can be concluded that when a distance between nodes is short and the radio electronic is high, data transmission perform better in term of energy efficiency than MTE routing. Therefore, when we consider designing the most energy-efficient protocol, we need to take into account the network topology.

3. Proposed Schemes

That leads us to start thinking about constructing the networks. To further improve energy efficiently, two approaches introduced in the papers are summarized in the following:

3.1 LEACH (Low-Energy Adaptive Clustering Hierarchy)

LEACH [1] is a cluster-based wireless sensor networking protocol. LEACH adapts the clustering concept to distribute the energy among the sensor nodes in the network. LEACH improves the energy-efficiency of wireless sensor networking beyond the normal clustering architecture. As a result, we can extend the life time of our network, and this is the very important issue that is considered in the wireless sensor networking field.

In LEACH protocol, wireless sensor networking nodes divide themselves to be many local clusters. In each local cluster, there is one node that acts as the base station (or we can call it “cluster-head”). Hence, every node in that local cluster will send the data to the cluster-head in
each local cluster. The important technique that makes LEACH be different from the normal cluster architecture (the drain the nodes battery very quickly) is that LEACH uses the randomize technique to select the cluster-head depending on the energy left of the node.

After cluster-head is selected with some probability, the cluster-heads in each local cluster will broadcast their status to the sensor nodes in their local range by using CSMA MAC protocol. Each sensor node will choose a cluster-head that is closest to itself to join that cluster because each sensor node will try to spend the minimum communication energy with its cluster head.

After the clustering phase is set up, each cluster-head will make a schedule for the nodes in its cluster. In paper LEACH, TDMA is used. For more efficiency, each sensor node could turn-off waiting for their allocated transmission.

Cluster-heads will collect the data from the nodes in its cluster, and compresses that data before transmits the data to the base station. By following this protocol, the base station will get the data from all sensor nodes that we are interested, and ready for the end-user to access the data.

Figure 2. Dynamic clusters: Cluster-head nodes at time $t_1$ and $t_1 + d$ [1]

Figure 2 shows that the cluster-heads of each local cluster are not fixed. At time $t_1$, a set $C$ of nodes might be the cluster-heads, and after that, at time $t_1 + d$, a set $C'$ might be the cluster-heads. This is because we want to spread the energy dissipation among all of sensor nodes.

3.2. PEGASIS (Power-Efficient Gathering in Sensor Information Systems)

This is improved version from LEACH. Although LEACH balances the energy cost, by clustering, sensor still needs relative large energy to transmit data to its cluster head. The main idea of PEGASIS is that nodes are formed into a chain where each node receive from and transmit to closest neighbor only. The distance between sender and receiver is reduced as well as decreasing the amount of transmission energy. To construct a chain, PEGASIS [2] uses a greedy algorithm that starts from the farthest node from the base station.
Figure 3. Chain is constructed using the greedy algorithm

In Figure 3, the algorithm starts with node 0 that connects to node 3. Then, node 3 connects to node 1 and node 1 connects to node 2, which is the closest one to the base station. Because nodes already in the chain cannot be revisited, the neighbor distance will increase gradually. When a node dies due to its battery, the chain will be reconstructed by repeating the same procedure and bypass the dead node.

In one round of transmission, a randomized node is appointed to be the leader to transmit data to BS. If the BS locates outside the range of this node, multi-hop transmission will be employed. The leader will be changed randomly in every round, so that overall energy dissipation is balanced out.

For transmitting a packet in each round, a token is used that passing from the one end of the chain to the other end of the chain. Only node that has a token can transmit a data packet to its intermediate node in the chain. When intermediate node receives data from one neighbor along with a token, it fuses the data packet with its own data and transmits a new data packet to the next node in the chain.

\[ \text{C0} \rightarrow \text{C1} \rightarrow \text{C2} \leftarrow \text{C3} \leftarrow \text{C4} \]

\[ \downarrow \]

BS

Figure 4. Token passing approach in PEGASIS

In figure 4, C0 will pass its data and token to C1. C1 fuses a data packet with its own data and pass a new data packet to the leader, C2. C2 does not transmit a data packet to BS yet, but rather it passes a token to C4. When C2 receives a data from C4 and C3, it fuses and transmits the sensed data to BS.

4. Performance comparison and analysis
In this paper, we use C++ to develop the simulation environment for wireless sensor networks. Some assumption and parameters are described as follows.

4.1 Parameters set up
The simulation variables are set up as follows.
- Sensor field: 100m x 100m
- Number of sensor nodes: 200 and 400 nodes uniformly deployed
Initial energy of sensor nodes: 0.25 (J)

The coordinate of base station: (50, 200)

Also, to evaluate energy consumption, the same parameters as in LEACH [1] are used. 
\[ E_{elec}=50\text{nJ/bit}, \ v_{amp}=100\text{pJ/bit/m}^2, \ k=2000 \text{ bits}, \] and every node consumes 5nJ/bit to complete data fusion [5]. (1J = 10^9\text{nJ} = 10^{12}\text{pJ})

In this section, Direct, LEACH[1], and PEGASIS[2] are implemented and compared. Direct represents each node directly transmits its sensed data to base station. Three evaluation metrics, which are widely-used in data gathering for WSNs, are utilized to evaluate the performance. They are defined as follows.

**Round:** a round for data gathering stands for all active sensors successfully transmit its sensed data to base station.

**Coverage ratio:** In some situations, numbers of round is not enough to represent the efficiency of a scheme. Uneven energy consumption, for example, may lead some nodes still having energy to operate, resulting in higher number of rounds, but actually it can not provide user with sufficient and full information about the sensor field. Therefore, in addition to number of rounds, by observing the coverage ratio, how long the complete information can be provided to end users is known.

**Total energy consumption per round:** the sum of every sensor’s energy consumption in one round data gathering. An efficient data gathering scheme should have lowered total energy consumption per round. Here, the equation used for compute this value is

\[
\text{Avg. total energy per round} = \frac{\text{Total energy consumption of a system before the 1st node dies}}{\text{Number of rounds a system has run before the 1st node dies}}
\]

**Energy x Delay metric:** the energy is described above. We refer to the definition of a transmission, which is defined in [3], one transmission for a sensor node takes one unit time, and delay is defined as the unit time that a system need to complete data gathering in one round. For example, in PEGASIS, a WSNs system spends 100 unit time completing one round for WSNs that consists of 100 sensor nodes. More details and explanation will be discussed in following sections.

4.2 Simulation Assumption

In our simulation environment, we assume that all nodes always have data to send and sensor devices are not with mobility, same initial energy, and capable of transmission range adjustment [4]. No multiple access interference problems when sensors broadcast and data can be correctly transmitted and received. Furthermore, for correctness of simulation, initially base station provides address localization for each sensor. We use the same assumption described in [1] and [2], and the number of cluster-heads is 5% of number of sensor nodes.

4.3 Simulation results and discussion
In figure 5 and figure 7, X axis represents node death percentage (percentage of nodes with no power) and Y axis represents number of round. In figure 6 and 8, X axis represents number of rounds; Y axis represents coverage ratio.

### 4.3.1 Advantages and drawbacks

From the simulation results, it makes sense that direct scheme has worst performance because all sensor consume more energy to transmit data directly to base station, resulting in shorter network lifetime whereas LEACH utilizes the advantage of clustering, only a few cluster-heads take the responsibility to send data and every sensor takes turn to be the cluster-head, causing the energy consumption distribute to other sensors so that higher network lifetime can be achieved. However, PEGASIS outperforms LEACH in three ways. First, the distance between neighbors in a chain is much shorter than the distance between a node in a cluster and its head, so each sensor won’t take that much energy. Furthermore, only one node transmits a data packet to BS per transmission round instead of several cluster heads in LEACH. Finally, the amount of data that the leader will receive in PEGASIS is two rather than from all cluster nodes in LEACH. As we can see in Figure 5, it is approximately almost 3 times better than LEACH.

Similarly, their corresponding performance on coverage ratio show reasonable results. If nodes drain battery very quickly, of course, the coverage can not be efficiently provided. However, it is important to mention here that a network with longer lifetime (higher rounds) does not guarantee a better coverage,
and although coverage ratio is related to node death percentage, a good energy-balanced scheme, which well distributes energy consumption among sensors and may lead all sensor die at about the same time, the value it brings is that a good coverage has been provided long enough before they are almost all dead simultaneously. That’s why we include the “coverage ratio” metric in the simulation in addition to number of rounds.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Energy(J)</th>
<th>Delay</th>
<th>Energy*Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1.02134</td>
<td>200</td>
<td>204.268</td>
</tr>
<tr>
<td>LEACH</td>
<td>0.289854</td>
<td>32</td>
<td>9.275328</td>
</tr>
<tr>
<td>PEAGASIS</td>
<td>0.050609</td>
<td>200</td>
<td>10.1218</td>
</tr>
</tbody>
</table>

**Table 1.** Cost Comparison. 200 nodes, BS at (50, 200), 5% Cluster heads

It is the known that energy consumption and delay is a trade-off. Higher energy consumption may bring the decrease of delay. Therefore, as discussed in [3], it is valuable to evaluate the performance from the Energy*Delay cost of view. From table 1, PEGASIS minimize the energy consumption per round by transmitting to closest neighbors within the chain. However, although it achieves less energy consumption per round, it produces longer delay to collect data, especially when it takes the end point node of the chain to be the leader in the current round. Besides, if failure nodes increase, longer distance between nodes needs more energy to communicate with; this will result in the performance drop rapidly. On the contrary, although LEACH has the mechanism that all sensors take turn to be cluster-head, as far as sensor concerns, it still takes a bit more energy to send data to its head when comparing with PEGASIS. But because of the advantage of the clustering architecture, lower delay is achieved. In term of Energy*Delay cost, it outperforms PEGASIS.

5. Improvement for proposed schemes

There is still room to improve for these two schemes.

5.1. Improvement for LEACH

In our simulation, in order to enhance the performance, a possible improvement is also implemented here to compare with the originality. An important disadvantage in LEACH is that if clusters can not be well divided in the clustering phase, each cluster size is huge different, which may result in significant uneven energy consumption among sensors. It could worsen the efficiency of usage of energy. Therefore, in order to well divide all sensors into clusters, we try to divide the area into several grids. Each grid can be treated as a cluster, as shown in Figure 12. Evenly clustering can achieve energy dissipation over all nodes more obvious. The simulation result is shown below in Figure 9 and 10.
Another more complex way may be feasible, we can construct hierarchy structure by which each cluster-head transmit data to its cluster-head in the same level, and ultimately the cluster-head in the highest hierarchy, which is leader, transmit data to BS.

### 5.2. Improvement for PEGASIS

On the other hand, in PEGASIS, Initially, it takes the advantage of sending data to its closet neighbor, which well save the battery for sensors. However, as mentioned in section 4.3, it causes serious delay and if there is one node become inactive, it will have to reconstruct the chain, plus, if failed nodes increase, the distance between nodes becomes larger; the performance will drop very soon.

An improvement idea proposed in [3] is also simulated here, in which after constructing the chain as PEGASIS did, all sensor nodes are paired up, a node send data to its pair node simultaneously, and only the node that receives data can be higher up to next level and send data to its neighbor in the same level, eventually the node in the highest level, which is leader, send data to BS. There will be \( \lceil \log N \rceil + 1 \) levels (N is the # of nodes). Simultaneously transmission for every pair reduces delay significantly. The behavior of Algorithm can be described in Figure 11.

![Binary Chain-based PEGASIS](image)

**Figure 11.** Binary Chain-based PEGASIS

The comparison results are shown in Table2. The more the nodes are, the worse the performance of PEGASIS becomes due to delay issue. In revised version of PEAGSIS, by simultaneous transmission on each pair, it saves relatively large time, although it spends more energy in higher level pairs, the cost is still considered improved from Energy \( \times \) Delay cost of view.
11

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Energy(J)</th>
<th>Delay</th>
<th>Energy × Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEACH</td>
<td>0.138247</td>
<td>31</td>
<td>4.28565</td>
</tr>
<tr>
<td>Revised LEACH</td>
<td>0.109467</td>
<td>31</td>
<td>3.39347</td>
</tr>
<tr>
<td>PEAGASIS</td>
<td>0.050609</td>
<td>200</td>
<td>10.1218</td>
</tr>
<tr>
<td>Chain-based binary PEGASIS</td>
<td>0.061235</td>
<td>9</td>
<td>0.551115</td>
</tr>
</tbody>
</table>

Table 2. Cost Comparison. 200 nodes, BS at (50, 200), 6% Cluster heads

6. Conclusions

The table 3 can briefly state the comparison among these main schemes in WSNs. “X” represents the corresponding protocol outperforms in that metric issue. However, it does not really elaborate what the best scheme is for WSNs. As it can be seen that WSNs have many applications, it is not clear as to what the optimal scheme is for optimization in a sensor network. The protocol is designed depending on the requirement of applications.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Implementation simplicity</th>
<th>Number of Rounds</th>
<th>Coverage</th>
<th>Energy-Balanced</th>
<th>Energy*Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEACH</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Revised LEACH</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEAGASIS</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chain-based binary PEGASIS</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 3 Performance comparison in term of different metric

Figure 12 Simulation Interface (C++ Builder 6)
7. References


Note: Simulation Source Codes
http://www-scf.usc.edu/~hsiaonuc/2010-12-05_NetworkSimulator_Ver1_0_by_hsiaonuc.rar
Please read Readme.txt before testing