Numerical Analysis of Multiple and Single Path Schemes for Sensor Networks

Saurabh Mehta*, Sung-Min Oh and Jae-Hyun Kim
School of Electrical and Computer Engineering, Ajou University
Email: {saurabh, smallb01, and jkim}@ajou.ac.kr

Abstract

In this paper we considered a network that consists of massively deployed tiny energy constrained sensors with one sink node to communicate with the outer world. We analyzed multiple and single path schemes for sensor networks and proposed single path with double mode method to solve network lifetime problem. Numerical analysis shows the comparison of the proposed scheme with existing multiple path schemes to increase a network lifetime. The proposed scheme increases a critical node lifetime by 38 % and hence a network lifetime.

1. Introduction

Wireless networks composed of highly integrated sensor nodes hold the promise of sensing that is far superior, in terms of quality, robustness, cost, and autonomous operation, to that offered by using a few, ultra high precision macro-sensors. Such sensor networks are expected to find widespread use in a variety of applications including remote climate monitoring and seismic, acoustic, medical, and intelligence data-gathering.

In a large sensor network all the nodes send their data to sink node for the further processing as shown in figure 1, due to this fact the nodes near to sink node consumed their energy more rapidly compared to other sensor nodes. In [1] authors describe the problem of developing an energy efficient operation of a randomly deployed multi-hop sensor network by extending the lifetime of the critical nodes and as a result the overall network’s operation lifetime, were considered and analyzed but they didn’t propose any solution for the same. In this paper we extended our work further from [1] and proposed scheme to increase the critical node lifetime and hence a network lifetime. In [2], [4], [5], and [6] authors suggested power aware multiple path algorithms to distribute the relay load equally among all the nodes. In a multiple path algorithm a node has to maintain the routing table which requires good memory capacity and periodic update for any new change in topology but this update increases the overhead. Multiple paths also help to distribute loads among the sensor nodes. But for distributing it equally or fairly we need to set some metrics. Generally it may be energy, delay or minimum hop count path and etc. All these metrics calculation requires complex algorithm and high computational resources but beside these disadvantages multiple path structure is very robust and reliable. Especially it is very useful for static applications.

In single path algorithm there is only one path available from source to sink and normally it is a minimum hop path. Due to single path there is always very heavy traffic on the route and also its lifetime is short but we can overcome these disadvantages by implementing the proposed scheme which distributes the load among the nodes. In multiple paths every node is connected to the number of paths and it is not practical to make them work in just one mode. Further we evaluated the efficiency of each scheme from mathematical and numerical analysis.

The contributions of this paper are as follows.
- We presented numerical analysis of multiple and single path.
- We proposed the single path with double mode state scheme.
- We compared the existing power aware multiple path scheme with the proposed scheme.

This paper is organized as follows. In section 2, we introduced our sensor network model. Section 3 described our proposed algorithm. In section 4 we described a node lifetime analysis, followed by numerical results in section 5. Finally conclusions are in section 6.
2. Sensor Networks Model

Here we considered a large network which contains $N$ number of densely deployed sensor nodes with one sink. All nodes have to transfer their data to sink that means a common destination address for all the nodes. All nodes in a sensor networks are static, same in size, battery capacity and etc. Every node has a static ID (Not IP) and does the relying and sensing. We make our further assumption from [1] as follows.

• $E_o = \text{Total energy of a node.}$
• $e_s = \text{The energy needed to sense one bit for } i^{th} \text{ node}$
• $e_{si} = \text{The energy required to transmit a bit.}$ It is given by $e_{si} = e_s + e_f (d/d_0)^n$, where $e_f$ is the energy consumed to transmit a bit to the reference distance $d_0$ and $n$ is the path loss index. $d$ represents actual distance.
• $p_{pi} = \text{The energy consumed per bit for data processing.}$
• $e_{pro, i} = \text{The energy consumed for data processing and aggregation by } i^{th} \text{ node.}$
• $p_{af} = \text{Energy required for data aggregation.}$ It is a function of aggregation ratio $\gamma$. $e_{pro, i} = e_p$ , where $e_{af} (\gamma) = 0$ when $\gamma = 1$.
• $\lambda_s = \text{The number of packets generated per unit time by } s_i.$
• $\lambda_r = \text{The number of packets relayed per unit time by } s_i.$
• $L = \text{Length of a data packet.}$

For simplicity we considered $\gamma = 1$ and $d = d_0$. Still we can simplify above terms by assuming that $e_p = e_s = e / 2$ and $e_{af} = e$. [1]. From all above assumption we can rewrite (1) in the following way:

\[
P_i = [2e + e, f] \lambda_s L + [2.5e + e, f] \lambda_r L \]

(2)

And let $E(t_i)$ be $i^{th}$ node lifetime that we have

\[
E(t_i) = E_o / P_i .
\]

(3)

From (2) we can observe that power consumed by a node is divided into two terms. First term is used only for sensing and transmitting its own data and second term used for the relaying purpose. Figure 2 (a) shows the above condition. From (2) we can conclude that 65% of its energy gets used only for the relaying data that is main cause of energy consumption for a node [3].

We can make second term equal to zero only if a node doesn’t need to relay any external data packet. This can be possible only in one case when a sink node is in range of all sensors as shown in figure 2 (b). This means we need to create an infrastructure or backbone in a sensor networks which will carry the node’s relay load but still this will cause a problem of an early disconnection in the networks. Here we proposed our scheme which helps nodes to create an energy efficient routing backbone based on energy consumption metric.

3. Proposed Scheme

The basic idea of the proposed scheme can be understand from the following steps.

1. All nodes are working under two mode, relay and sense and keep changing their modes as per the scheme’s set condition. Figure 3 gives clear idea about these two modes and their transition.
2. At first, sink node will broadcast an advertisement for hop count information and after some delay all nodes will know its hop position.
3. After the hop information, sink node choose random nodes which are one hop apart from each other for carrying relay loads.

For simplicity we considered $\gamma = 1$ and $d = d_0$. Still we can simplify above terms by assuming that $e_p = e_s = e / 2$ and $e_{af} = e$. [1]. From all above assumption we can rewrite (1) in the following way:

\[
P_i = [2e + e, f] \lambda_s L + [2.5e + e, f] \lambda_r L \]

(2)

And let $E(t_i)$ be $i^{th}$ node lifetime that we have

\[
E(t_i) = E_o / P_i.
\]

(3)

From (2) we can observe that power consumed by a node is divided into two terms. First term is used only for sensing and transmitting its own data and second term used for the relaying purpose. Figure 2 (a) shows the above condition. From (2) we can conclude that 65% of its energy gets used only for the relaying data that is main cause of energy consumption for a node [3].

We can make second term equal to zero only if a node doesn’t need to relay any external data packet. This can be possible only in one case when a sink node is in range of all sensors as shown in figure 2 (b). This means we need to create an infrastructure or backbone in a sensor networks which will carry the node’s relay load but still this will cause a problem of an early disconnection in the networks. Here we proposed our scheme which helps nodes to create an energy efficient routing backbone based on energy consumption metric.

3. Proposed Scheme

The basic idea of the proposed scheme can be understand from the following steps.

1. All nodes are working under two mode, relay and sense and keep changing their modes as per the scheme’s set condition. Figure 3 gives clear idea about these two modes and their transition.
2. At first, sink node will broadcast an advertisement for hop count information and after some delay all nodes will know its hop position.
3. After the hop information, sink node choose random nodes which are one hop apart from each other for carrying relay loads.

For simplicity we considered $\gamma = 1$ and $d = d_0$. Still we can simplify above terms by assuming that $e_p = e_s = e / 2$ and $e_{af} = e$. [1]. From all above assumption we can rewrite (1) in the following way:

\[
P_i = [2e + e, f] \lambda_s L + [2.5e + e, f] \lambda_r L \]

(2)

And let $E(t_i)$ be $i^{th}$ node lifetime that we have

\[
E(t_i) = E_o / P_i.
\]

(3)

From (2) we can observe that power consumed by a node is divided into two terms. First term is used only for sensing and transmitting its own data and second term used for the relaying purpose. Figure 2 (a) shows the above condition. From (2) we can conclude that 65% of its energy gets used only for the relaying data that is main cause of energy consumption for a node [3].

We can make second term equal to zero only if a node doesn’t need to relay any external data packet. This can be possible only in one case when a sink node is in range of all sensors as shown in figure 2 (b). This means we need to create an infrastructure or backbone in a sensor networks which will carry the node’s relay load but still this will cause a problem of an early disconnection in the networks. Here we proposed our scheme which helps nodes to create an energy efficient routing backbone based on energy consumption metric.

3. Proposed Scheme

The basic idea of the proposed scheme can be understand from the following steps.

1. All nodes are working under two mode, relay and sense and keep changing their modes as per the scheme’s set condition. Figure 3 gives clear idea about these two modes and their transition.
2. At first, sink node will broadcast an advertisement for hop count information and after some delay all nodes will know its hop position.
3. After the hop information, sink node choose random nodes which are one hop apart from each other for carrying relay loads.
4. Let us denote all randomly chosen nodes as Cluster Node (CN). All CN(s) advertise about their status and collects information about their members.

5. All CN(s) will establish connection with just one node which is only one hop apart and has lower hop count than CN and it will be also consider as CN.

6. All CN will operate in relay mode and the rest of the nodes are in sense mode. CN will operate in relay mode until the remaining energy reach to its set energy threshold. Then it will broadcast an advertisement for a mode change. This advertisement contains its ID, upper and lower hop CN ID and its energy level.

7. CN will wait for some random delay. During this delay some response will come from the near by node and CN will change its mode from relay to sense. Furthermore it will choose new CN for the relay. Due to densely deployment of sensors, backbone connectivity with upstream and downstream neighbors will maintain.

8. Now old CN will work in sense mode. If any advertisement comes for relay and if that advertisement satisfied all decision condition then it will again enter into the relay mode otherwise remain in a same mode.

**Proposed Scheme Pseudo Code**

```
Begin
  If (mode==sense) sense_mode ();
  else relay_mode ();
End

sense_mode ();
Begin
  If (senser_pkt_int==arrived) ++sense;
  --energy; create_data_tx ();
  Begin
    If (energy>adv_energy) change_mode ();
    else cont_sense_mode ();
  End
End

relay_mode ();
Begin
  If (relay_pkt_int==arrived) ++relay;
  --energy; next_hop_tx ();
  Begin
    If (energy<limit) energy_adv_tx ();
    else cont_relay_mode ();
  End
End
```

Where,
1. `create_data_tx ()` = Procedure for creating data frame and transmitting to near by CN(s).
2. `change_mode ()` = Procedure for changing the operation mode.
3. `cont_sensing_mode ()` = Procedure for continuing the operation in sensing mode.
4. `next_hop_tx ()` = Procedure for transmitting packet to next hop.
5. `energy_adv_tx ()` = Procedure for transmitting an advertisement for changing the mode and change the mode on receiving acknowledgement.

4. A Node Lifetime Analysis

As we proposed in the scheme all nodes will work in two modes and their energy consumption will change according to their operating mode. Energy consumed in relay mode is given by

\[ P_r = (2.5 \varepsilon + \varepsilon_J) \lambda_n L. \]  \tag{4} 

Energy consumed in sense mode is given by

\[ P_s = (2 \varepsilon + \varepsilon_J) \lambda_n L. \]  \tag{5} 

but in our proposed scheme sensor node also need to transmit and receive some overhead signals. We added \( 0.5 \varepsilon \) in the above term. So modified equation is as follows.

\[ P_i = (2.5 \varepsilon + \varepsilon_J) \lambda_n L. \]  \tag{6} 

In multiple paths, a node’s energy consumption is given by

\[ P_i = [(\lambda_n + \lambda_m) (2.5 \varepsilon + \varepsilon_J)] L. \]  \tag{7} 

In the proposed algorithm nodes change its state according to energy threshold set that is decided by a number of neighbor, so to find lifetime of a node we need to find average power consumption at node during all modes and it is given by

\[
P_i = \frac{\sum_{i=1}^{n} \left( \sum_{j=i}^{n} \lambda_{ij} (2.5 \varepsilon + \varepsilon_J) L \right) + \sum_{j=1}^{m} \left( (2.5 \varepsilon + \varepsilon_J) \lambda_n L \right)}{(m + n)},
\]  \tag{8} 

where \( n \) and \( m \) is a number of time node enter in relay and sense mode, while \( H_r \) and \( H_i \) represents total number of hop count and individual node respectively.

To know networks lifetime we need to calculate average lifetime of critical node. Here critical node means a node which connects sink with other sensor nodes. To calculate average lifetime of a node we need to calculate maximum traffic rate arriving at critical node in multiple path as well as single path case.

**A. Multiple Path Analysis**

Maximum traffic that can arrive at critical node is given by

\[
Relay \text{ packets } + \text{ Own generated packets } = \left[ \sum_{i} \lambda_n f(s) + \sum_{i} \lambda_n + \lambda_m \right]. \tag{9}
\]
Where \( n_p \) is number of path connected to critical node and \( n_{cp} \) means number of critical paths connected to critical node. If we consider that multiple path algorithm is energy aware, \( n_p \) is depends on a function of energy \( (f(e)) \) and its value varies from 1 to 0. Here critical path means, a path don’t have any other routing path except one connected to critical node. Now from (1), (7) and (9), power consumed at multiple paths node is given by

\[
P_t = \left[ \sum_{i}^{n_p} \lambda_i f(e) + \sum_{i}^{n_{cp}} \lambda_{cp} \right] \times [2.5\varepsilon + e_{tr}] L.
\]  

(10)

From (3) and (10) node lifetime is given by

\[
E(t) = \frac{E_{s}}{\left( \sum_{i}^{n_p} \lambda_i f(e) + \sum_{i}^{n_{cp}} \lambda_{cp} \right) (2.5\varepsilon + e_{tr}) L}.
\]  

(11)

B. Single Path Analysis

Maximum traffic that can arrive at node when it is in relay mode is given by

\[
\lambda_i = \sum_{j \neq i}^{n} \lambda_{sch}. \quad (12)
\]

Where \( \lambda_{sch} \) is the number of packets relay by a CN. Maximum traffic at node when it is in sense mode is given by \( \lambda_s \) and average \( P_f \) is given from (8). From (3) and (8) node lifetime is given by

\[
E(t) = \frac{E_{s}(m+n)}{\left( \sum_{i}^{n_p} \lambda_i (2.5\varepsilon + e_{tr}) L \right) + \sum_{j}^{n_c} \left( (2.5\varepsilon + e_{tr}) \lambda_{cp} L \right)}.
\]  

(13)

To understand our analysis considered the example shown in figure 4. As shown in figure 4 nodes A is surrounded by four nodes B, C, D and E respectively and located at first hop so all nodes are considered as critical nodes. Node A’s lifetime represents a network lifetime. First, node A selected as CN node and work in relay mode. Rest of the nodes will work in sense mode. Now our proposed scheme will control the topology.

5. Numerical Results

In this section, we show numerical results based on the node lifetime analysis introduced in the previous section. All nodes have 6 joule battery capacity which can support 9000 packets of 32 bytes long for receiving and transmitting. In sensor networks relay rate is always higher than the packet generating rate. From [1], [3], and [6], we assumed some parameter’s values and summarized them in TABLE I.

<table>
<thead>
<tr>
<th>Parameters Assumed value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E_s ) 6 j</td>
</tr>
<tr>
<td>( \varepsilon ) 50 nj</td>
</tr>
<tr>
<td>( \lambda_s ) 5 pkt/hr</td>
</tr>
<tr>
<td>( L ) 32 byte</td>
</tr>
<tr>
<td>( e_{tr} ) 2.5 ( \mu ) j</td>
</tr>
</tbody>
</table>

We divided the total energy into the number of threshold level, as per the proposed scheme node will change its state on every threshold level. As we can see it from figure 5, the number of packet arrival rate will also change accordingly. When a node is in a relay mode it has to process on a larger number of packets than in sense mode. This is the key factor of the proposed scheme. Because of two operating modes average arrival rate of packets is low compared to energy aware multiple path algorithm.

Figure 6, 7 and 8 show some important numerical results which are based on a node lifetime analysis and assumptions. Figure 6 shows the critical node A’s lifetime. From figure 6 we can observe that the proposed algorithm increases the critical node lifetime and hence a networks lifetime. From figure 6 we can observe that the arrival rate of packets in multiple paths scheme is depends on a function of energy. As the energy level decreases the number of packet process by a node decreases but the average arrival rate of packet is higher than a single path algorithm. This is the main difference between the two schemes. From figure 7 we can observe the lifetime of node B, C, D and E. They change their mode according to proposed scheme. If we take cumulative effect of all nodes (A, B, C, D and E) then it will increase networks life time by 4 times but here we
Figure 5. Average number of packets process by a node consider only one critical node lifetime as networks lifetime. Figure 8 shows the energy consumed by one packet to reach destination from source.

The proposed scheme always creates a minimum hop path from source to destination. But in multiple path algorithm it generally varies from a minimum hop path to maximum as multiple path scheme is depends on a function of energy. For real time application the number of hop required to transmit the data from source to destination is very important because it involves the delay factor in transmission.

6. Conclusions

In this paper we proposed the scheme to increase a network lifetime and we compared it with multiple energy aware path scheme. The proposed scheme increases a network lifetime by fairly distributing the relay load among the nodes with the help of two different operating modes. So our approach is suitable for a densely deployed sensor networks. Numerical result shows the good improvement in a critical node lifetime. The proposed scheme increase the lifetime by around 38% which looks quite promising and the proposed scheme generate a minimum hop path which is very important result for the real time data applications.

References