PBQM: A PRIORITY-BASED QUEUE MANAGEMENT METHOD FOR WIRELESS SENSOR NETWORKS

XIANG-LAN YIN, HUA CHEN and YANG SHEN
Institute of Command Automation
PLA University of Science & Technology
Najing 210007, China

Received 8 August 2006
Accepted 23 May 2007

Event detection is important in many sensor network applications. Traffic in these applications exhibits a many-to-one pattern, in which multiple source nodes send sensing data to a single sink node. Since bandwidth and storage capacity is highly constrained in sensor networks, packet loss is inevitable when a great deal of traffic rushes to sink. Based on the observation that the packets are not equal in the serving of event detection, we propose to employ priority-based queue management method (PBQM) to improve packet delivery ratio of more important packets. We put forward three priority mapping schemes and evaluate the performance of PBQM method by simulation with Ns-2.

Keywords: Priority queue management; wireless sensor networks.

1. Introduction

Wireless sensor networks (WSN) is a new sensing and signal processing technology based on the collaborative effort of large number of self-organizing, low-cost nodes. These nodes are distributed randomly and equipped with kinds of sensors, low-power processor and short-distance radio. Compared with traditional information gathering and processing systems, WSN has several advantages, such as inexpensive devices, close to the sensing object, cooperate management, therefore, it has many potential applications, including target tracking in battlefields [He et al., 2004], emergency response [Welsh et al., 2003], habitats monitoring [Mainwaring et al., 2002], fire detection in forests.

In this paper, we focus on the event-source wireless sensor network applications, in which multiple source nodes send sensing data to a single sink node. Since bandwidth and storage capacity is highly constrained in sensor networks, packet loss is inevitable when a great deal of traffic rushes to sink [Gupta et al., 2000]. This situation will get deteriorated in large-scale and dense sensor networks.

Some researchers want to solve this problem through decreasing the communication traffic. Because there will be some redundancy between data originated from nearby source nodes in a dense network, in-network filtering and processing techniques can eliminate the redundancy and decrease the traffic [Bhaskar et al., 2005]. However, the depth of the aggregation
is bounded by entropy [Intanagonwiwat et al., 2001], furthermore, when the network fails to transmit the aggregated data to sink in time, the performance of the system will be worsen. The traditional congestion-control algorithm will improve the performance of the network through inspecting congestion and controlling transmission rate, which are trying to provide equal bandwidth for each node. However, these methods will enlarge the delay, also lead to the queue overflow when the source nodes sample the signals with high frequency.

In this paper, we propose a priority-based queue management (PBQM) method to solve this problem. The observation behind this idea is that different packets may be of different importance in the serving of event detection. For example, since the precision of sensors decreases when the distance grows, the data collected by the nodes closer to the event are usually more precise than those collected by nodes farther away, and therefore more critical to the application. When the network fails to transmit all packets to sink, it should try its best to transmit the packets which are more important. We utilize priority to describe the importance of the packets. The packets with more importance will be granted higher priority. We propose three algorithms to grant priority to the packets and the packets will be ordered in the queue according with their priority, i.e. packets with higher priority will be serviced before those with lower priority. The effect of PBQM method is evaluated with Ns-2 [NS Manual] and the simulation results indicate it will increase the PDR of more important packets. Furthermore, our method can be integrated with other mechanisms, such as data aggregation and congestion control, to further improve the performance of the network.

2. Related Work

In this section we introduce some aggregation methods and congestion-control algorithms briefly.

In [Petrovic et al. 2003], the network was divided into several cells and the data in same cell would be aggregated and then be transmitted to sink through border nodes. It was a loss-free aggregation, just reduced the overhead of several packet headers.

Escan [Zhao et al., 2005] utilized the correlation of node’s geographical location to implement in-network aggregation. Escan thought it was wasteful to send all the data back to the sink because the data collected by nearby sources were similar or even the same. It was a lossy aggregation method, which discarded some unimportant information. The lossy aggregation can greatly reduce the traffic than loss-free aggregation, but the depth of it is bounded by entropy [Intanagonwiwat et al., 2001]. It is to say the packets have to remain some certain information after they are aggregated. The basic principle of packet aggregation is to only eliminate redundant information.

Although these methods will reduce number of transmissions and save energy, it will introduce great delay. Because the data from nearer source nodes have to be kept for a long time at an intermediate node in order to be aggregated with data coming from source nodes that are farther away. In the worst case, the latency because of aggregation will be proportional to the number of hops between the sink and the farthest source node. Therefore, it is improper to adopt aggregation when the application requires real-time service.

Ee et al. [2004] discussed how to distribute same available bandwidth to each node when the network adopted many-to-one routing, prevented the sensors farther away from the base station experiencing higher packet dropped ratio than the sensor nodes closer to the base station. The congestion control method was simple, which required each node measure the average packet rate \( r \) of itself and adjust \( r \) according the rate \( r_{data, parent} \) assigned by its parent, it would also allot its children the data packet generation rate \( r/n \) (\( n \) is the number of the children). This method ensures the fair packet delivery ratio of each source node. Our intention is in opposite to it, that we think different node should be assign different bandwidth because their packets have different importance.
3. PBQM Method

We make the following assumptions about our network model:

1. The distance between node $i$ and the event is $d_i$, and we call it $NE_{distance}$.
2. The destructive radius of the event is $r$, the nodes in this area $(0 \leq d_i < r)$ would not work any more after the event happens, such as, they are damaged because of the blast.
3. Each node has the same sensing radius $R$, that is, when $d_i$ meets the condition that $r < d_i < R$, node $i$ is a source node.

Our network is designed to detect some rare, random, and ephemeral events. The application requires the nodes to sample signals in high frequency. Whenever an event occurs, there would be a great volume of traffic directed to the sink node, resulting serious congestion and packet loss. Traditional FIFO queue management method treats packets as equal, pays no attention to the different importance among packets. The important packets may be dropped because of the above reason while the unimportant packets may be received successful. In this paper, we try to solve this problem by utilizing a priority-based queue management (PBQM) method. We grant different priority to packets based on their importance and schedule the packets with the highest priority first, trying to improve the PDR of more important packets when the available bandwidth is seriously constrained.

3.1. Priority mapping

In order to apply the new queue management method to our network, a mechanism is needed to judge the importance of the packets and grant appropriate priority to them. The basic principle of it is the sensors closer to the event are more likely to take precise samples, and packets originated from these nodes are more critical to the application, therefore, the former will be granted higher priority than the latter. Three detailed algorithms are provided in the following text.

3.1.1. $NE_{distance}$ Based

It is easy to grant priority to the packets when each node can obtains $NE_{distance}$ in advance. For convenience, we divide the priority into 5 levels. Priority 1 is the highest and Priority 5 is the lowest. Accordingly, we divide the distance $(R - r)$ into 5 section, as shown in Fig. 1, with respective range as $[r, r + d], [r + d, r + 2d], [r + 2d, r + 3d], [r + 4d, R]$ $(d = (R - r)/5)$. The packets produced by the nodes within the same range are granted the same priority, which is in inverse proportion to $NE_{distance}$.

Although this algorithm is simple, the pre-condition of it is hard to be met. Only those applications where the event will happen is predefined, such as appointed area detection, can adopt it.

3.1.2. Signal strength based

In many applications, such as target tracking in battlefields, fire detection in forests, where the event will happen is uncertain, the above priority-mapping algorithm is inapplicable to these applications. We have to find a new method to judge the importance of the packets.

Based on the usual observation, we get a conclusion that the strength of some signals, for example, the radio signal, the audio signal, attenuates along with the transmission distance, and signal strength detected by closer source node is stronger than which detected by further
away nodes. Therefore, we utilize the data of the packets to compute \( NE_{\text{distance}} \), that great value corresponds with short \( NE_{\text{distance}} \) and high priority. The transmitter nodes can judge the priority of the packets based on the value. But this is not enough. Suppose the source nodes divide the total sample time into several small time intervals, and sample the signals only once in a time interval. Node \( A \) samples the signals in time interval \( n_1 \) and gets a value \( S_A \), node \( B \) gets a value \( S_B \) in time interval \( n_2 \). Only when \( n_1 \) and \( n_2 \) are equal, it is useful to judge the priority of the packets by comparing \( S_A \) and \( S_B \). Therefore, the node has to judge the priority of the samples based on the time interval and value. The samples with higher value than others sampled in the same time interval will be granted high priority.

This signal-strength based method requires the attenuation of the signal strength is obvious when the transmission distance grows and only related with the distance. But the signals often spread very quickly, only the distance between nodes exceeds a certain threshold can detect the difference between the signal strength. Furthermore, the WSN are often located outdoors, where environment is complex, the attenuation of signal strength will be greatly influenced by the environment. For example, in Fig. 1, the \( NE_{\text{distance}} \) of nodes \( A \) and \( B \) is equal and the packets originated from node \( A \) and node \( B \) in same time interval should have nearly equal value and be granted same priority. However, there is a big barrier between the event and node \( A \). The signal strength detected by node \( A \) will be much less than that of node \( B \), so the value of node \( A \) is smaller than that of node \( B \). The intermediate node \( C \) judges the priority of the packets by the value, so it may conclude the priority of the packets originated from node \( A \) is lower than that of node \( B \) and the packets of node \( A \) should be processed after those of node \( B \) have been managed.

3.1.3. Timestamp based

In order to eliminate the influence of environment, we design a new algorithm which based on the timestamp, because the source nodes closer to the event will detect the event earlier than the further away nodes. The intermediate nodes judge the priority of the packets from their timestamp, the packets sampled earlier will be granted higher priority. Assume in time interval \( n \), nodes \( A \) and \( B \) sample the signals at different time \( t_1 \) and \( t_2 \), when \( t_1 \) is less than \( t_2 \), the packets of node \( A \) will have higher priority than that of node \( B \).

The signals often spread very quickly, there will be little difference between the timestamp of nodes \( A \) and \( B \) when nodes sample signals with low frequency, therefore, this method only suits the application when nodes sample signals with high frequency.

When the conditions of the above algorithms are unable to be met, we need to find other methods to map priority to packets. After the priorities of the packets are granted, the management of the packets in the intermediate nodes is similar to the method provided in this paper.

3.2. Priority-based queue management

Many applications adopt FIFO and drop-on-overflow queue management. FIFO means the packet to arrive at buffer first will also be served first. Drop-on-overflow means the new incoming packet will be dropped when the queue is full. The priority-based queue management method is different from them. When the packet will be serviced is decided by its priority. The packets with highest priority will be ordered at the queue head and be serviced first. The lowest priority packet will be dropped when the queue is full. This method would not increase the efficient throughput of the network, but it will improve the PDR of the more important packets.

4. Evaluation

4.1. Parameters of the simulation

We use Ns-2 to evaluate the effect of PQMPS. The queue management and packet scheduling were modified in the experiments, from
Table 1. Parameters for simulation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td>20 kbps</td>
</tr>
<tr>
<td>Dimensions of Space</td>
<td>400 m * 300 m</td>
</tr>
<tr>
<td>Number of Nodes</td>
<td>567</td>
</tr>
<tr>
<td>Distribution of Nodes</td>
<td>grid</td>
</tr>
<tr>
<td>Radio Range</td>
<td>40 m</td>
</tr>
<tr>
<td>Routing Protocol</td>
<td>AODV</td>
</tr>
<tr>
<td>Length of the Queue</td>
<td>50 packets</td>
</tr>
<tr>
<td>Application Data Payload Size</td>
<td>36 bytes/packet</td>
</tr>
<tr>
<td>Source Data Pattern</td>
<td>5 packets/second</td>
</tr>
<tr>
<td>r</td>
<td>25 m</td>
</tr>
<tr>
<td>R</td>
<td>50 m</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>600 seconds</td>
</tr>
</tbody>
</table>

FIFO to “packets with highest priority serviced first”.

The parameters in paper [Yin et al., 2006] is not in accordance with the wireless sensor nodes widely used today, such as Mica2. Therefore, according the hardware limitation of Mica2 [Mica2 manual] and the characteristics of TinyOS operating system [Hill et al., 2000], in the simulation experiment, we adopt the parameters in Table 1.

To make the results more representational, we studied 20 different nodes distribution scenes. Figure 2 is one of them. Three typical locations of each scene were chosen to be studied. In the experiment, we use the first method to grant priority to the packets.

The parameters in paper [Yin et al., 2006] is not in accordance with the wireless sensor nodes widely used today, such as Mica2. Therefore, according the hardware limitation of Mica2 [Mica2 manual] and the characteristics of TinyOS operating system [Hill et al., 2000], in the simulation experiment, we adopt the parameters in Table 1.

To make the results more representational, we studied 20 different nodes distribution scenes. Figure 2 is one of them. Three typical locations of each scene were chosen to be studied. In the experiment, we use the first method to grant priority to the packets.

4.2. Analyze the simulation results

We studied the PDR of different nodes when priority-based packet scheduling method had been applied to the network. Figures 3 to 5 are the simulation results. For comparison, we have also included the result when FIFO method had been applied to the network.
The $y$-axis shows the packet delivery ratio, and $x$-axis shows the priorities of the source nodes, which is equivalent to the priority of the packets originated from the node. “1” denotes the highest priority, “5” denotes the lowest priority.

(i) The event happened at the left down corner of the network.

It was farthest away the sink. Firstly, we study the simulation results when the system adopted FIFO scheduling. The average PDR of the network was low, only 60%. From Fig. 3, we got two conclusions. The first one was the PDR of the node did not have obvious relation with its priority. For example, the PDR of some nodes with Priority 1 is below 20%; however, the PDR of some nodes exceeds 80%. The second one was the PDR of the nodes in same priority area had great difference. For example, in Priority 1, the lowest PDR was only 16%, the highest was 100%. This great difference has close relation with the location of the nodes. As in the Fig. 1, Nodes A and B had same priority. Node B was closer to the sink than Node A, also the routing to sink did not need to go around the damaged nodes, so the PDR of Node B will be greater than Node A.

Now we begin to study the simulation results when the PQBM method had been applied to the network. The average PDR of the network was still 60%, but the PDR of nodes changed greatly. For example, the average PDR of Priorities 1 improved to 88%, the PDR of Priority 2 improved from 70% to 80%, the PDR of other Priority nodes had some decline, especially the Priority 3. It is easy to get the conclusion the PDR of the nodes has close relation with their priorities. The nodes with high priority had great PDR. Besides this, the difference between the PDR of each nodes with same priority was also reduced.

(ii) The event took place at the middle of the network.

The average PDR of the network had a certain increase from 60% (in Fig. 3) to 65%. After the PBQM method had been applied, the PDR of Priorities 1 and 2 separately increased 20% and 8%. Same as the Fig. 3, the PDR of the node is in proportion to its priority, when the priority grows, the PDR is also increased.

(iii) Where the event happens was close to the sink.

In this condition, the average PDR of the network is 66%. Similar to Figs. 3 and 4, when the PBQM method has been applied, the PDR of the node is in proportion to its priority.

From the above analysis, we would get such a conclusion, when the priority-based queue management method had been applied, the PDR of the source nodes have close relation with their priorities. Compared with the result that FIFO is applied, the PDR of the nodes with high priority will be increased, otherwise, the PDR will be decreased. When the priority of the nodes grows, the changes become distinct. They are also related with the distance between the source nodes and sink, which will be more and more obvious when the distance increases. It is consistent with the observation in paper [Bhaskar et al., 2005].

Although the simulation parameters are different from the paper [Yin et al., 2006], such as the network topology, packet length, the results are similar, which indicate our protocol are independent with those factors.

There are many reasons why the packets are lost. Paper [Cheng et al., 2004] classified them into four types, include signal attenuates,
interferences between nodes, self-interference, queue overflow. Our methods only assure the packets with high priority not be dropped because of queue overflow, but they may be dropped because of other reasons, so in Figs. 3–5, some packets with high priority were still dropped.

5. Conclusion

In this paper, we propose a priority-based queue management method — PBMQ, which will improve the PDR of the more important packets when the efficient throughput is fixed. The most important presupposition of PBMQ method is to grant appropriate priority to the packets. The packets are granted higher priority when the packets are more important. When the packets will be serviced in the queue is decided by their priority. The packets with high priority will have preferable rights. Most of the lost packets will be the packets with low priorities.

References


The ns Manual (formerly ns Notes and Documentation) http://www.isi.edu/nsnam/ns/ns-documentation.html.


Biography

Xiang-Lan Yin received the BS degrees from Institute of Communications Engineering in 2002. She is now working towards her PhD degree in the Institute of Command Automation. Her research interests are in the areas of wireless sensor networks and network security.

Hua Chen received the BS degree from the Institute of Communications Engineering in 2001. He is now working towards his PhD degree in the Institute of Command Automation. His research interests are in the areas of wireless sensor networks.

Yang Shen received the BS and MS degrees from the Institute of Communications Engineering in 1999 and 2001. He is now working towards his PhD degree in the Institute of Command Automation. His research interests are in the areas of wireless sensor networks.