Technical Disclosure Commons

Defensive Publications Series

October 16, 2018

IMPROVING QUALITY OF SERVICE PERFORMANCE IN LOW POWER AND LOSSY NETWORKS

Lele Zhang

Yajun Xia

Yinfang Wang

Chuanwei Li

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation

Zhang, Lele; Xia, Yajun; Wang, Yinfang; and Li, Chuanwei, "IMPROVING QUALITY OF SERVICE PERFORMANCE IN LOW POWER AND LOSSY NETWORKS", Technical Disclosure Commons, (October 16, 2018)
https://www.tdcommons.org/dpubs_series/1598

This work is licensed under a Creative Commons Attribution 4.0 License.

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

IMPROVING QUALITY OF SERVICE PERFORMANCE IN LOW POWER AND LOSSY NETWORKS

AUTHORS:
Lele Zhang
Yajun Xia
Yinfang Wang
Chuanwei Li

ABSTRACT

Techniques are described herein to improve Quality of Service (QoS) performance of Low power and Lossy Networks (LLN). This guarantees a special high speed link for high priority data packets for LLN without reducing the performance of normal priority throughput.

DETAILED DESCRIPTION

In smart grid applications, a Personal Area Network (PAN) usually consists of a border router and thousands of nodes. Each node may have more than hundreds of other nodes within its neighborhood. However, they build a mesh network by using Routing Protocol for Low power and Lossy Networks (RPL) to generate a tree-like topology, as illustrated in Figure 1 below.

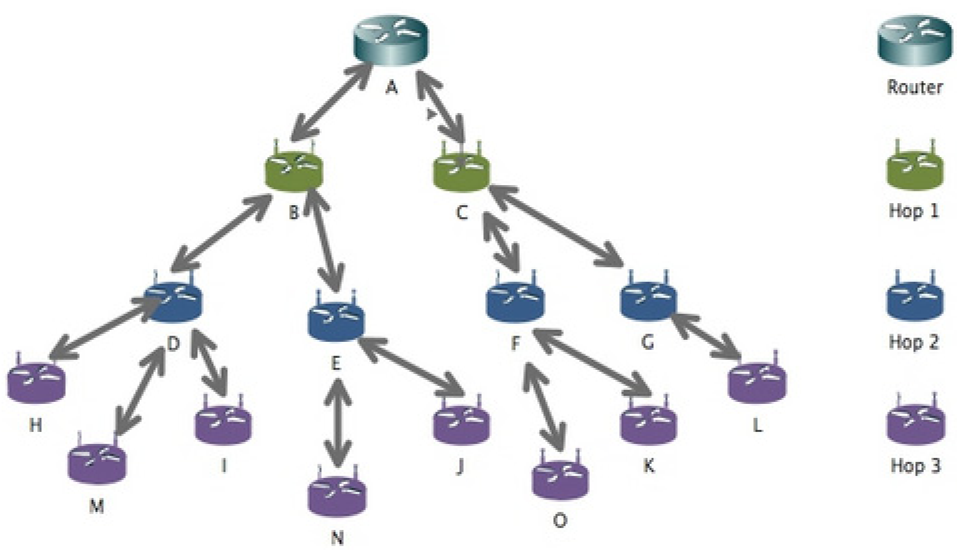

Figure 1

There are two typical priorities of data in a mesh network: high priority and normal priority. High priority data include network control messages, network status report messages, and alert data. Normal priority data include daily report data and metering record data.

The data of each priority is pushed in a different queue for transmission. As illustrated in Figures 2 and 3, nodes always send high priority packets to their parent nodes before normal priority packets.

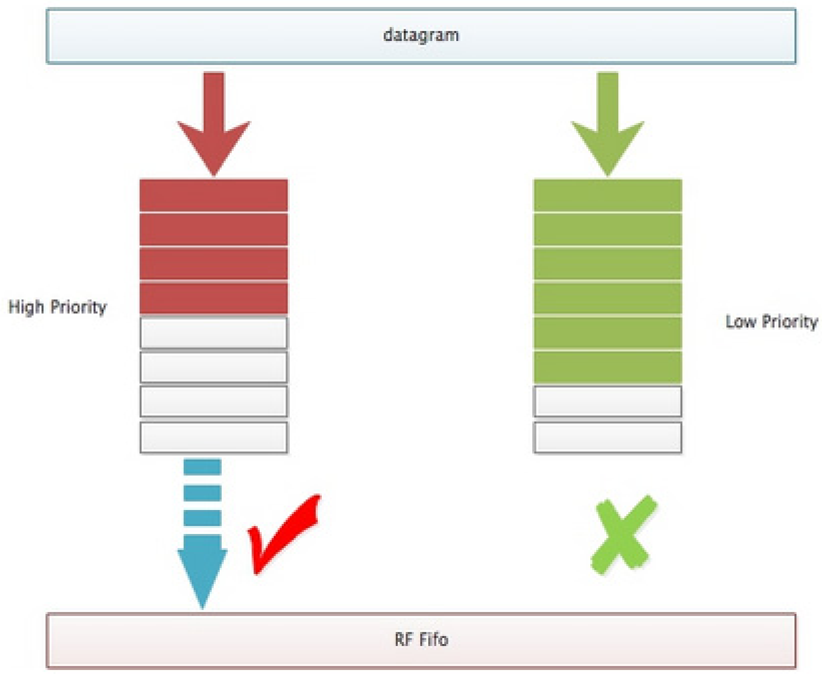

Figure 2

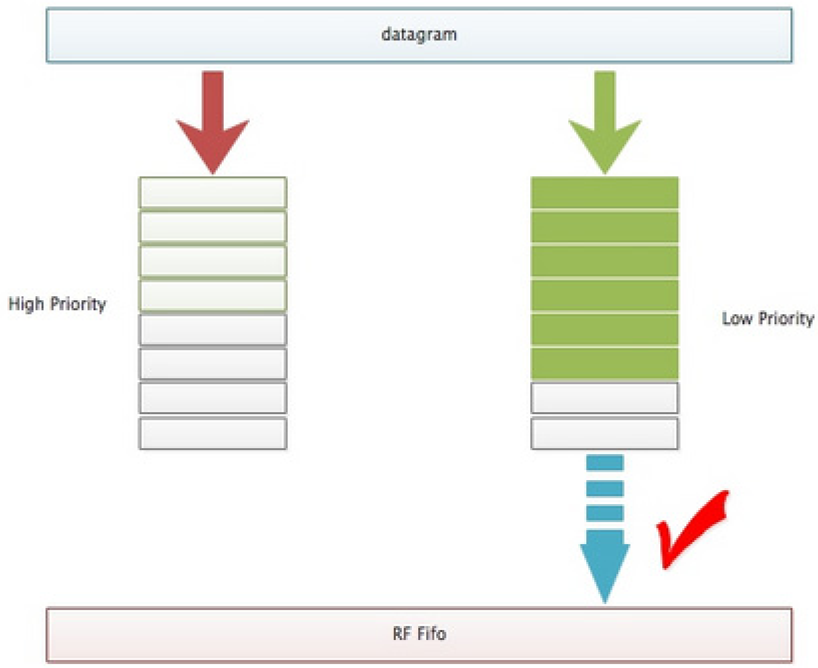

Figure 3

Thus, a normal priority queue cannot empty into a Radio Frequency (RF) First In, First Out (FIFO) before the high priority queue empties. This strategy is useful for an independent node, but is not efficient among many neighbor nodes that all want to send packets simultaneously.

Despite the strategy to use different queues for different priority data packets, some problems still exist. For example, sibling nodes still compete for bandwidth resources with low priority data packets when a node is sending high priority data. Therefore, potential collision risks reduce the Quality of Service (QoS) on this link. Moreover, one characteristic of radio is that it uses half-duplex to send and receive data. Thus, if a parent node relays both priority types of data, it spends an inordinate amount of time receiving and relaying low priority data, which results in fewer time slots for high priority data. This is not the proper manner in which to deal with high priority data.

As illustrated in Figure 4 below, Nodes D, E, and F are all sub-nodes of Node A. Node F is going to send high priority data to Node A, but its sibling Nodes D and E are not aware of that. Node F has to compete with Nodes D and E for channel access right from Node A, even though its data priority is higher than its neighbors. Node F may try to choose another link for transmitting its high priority data (e.g., through Node C to Node B) to finally arrive at the router.

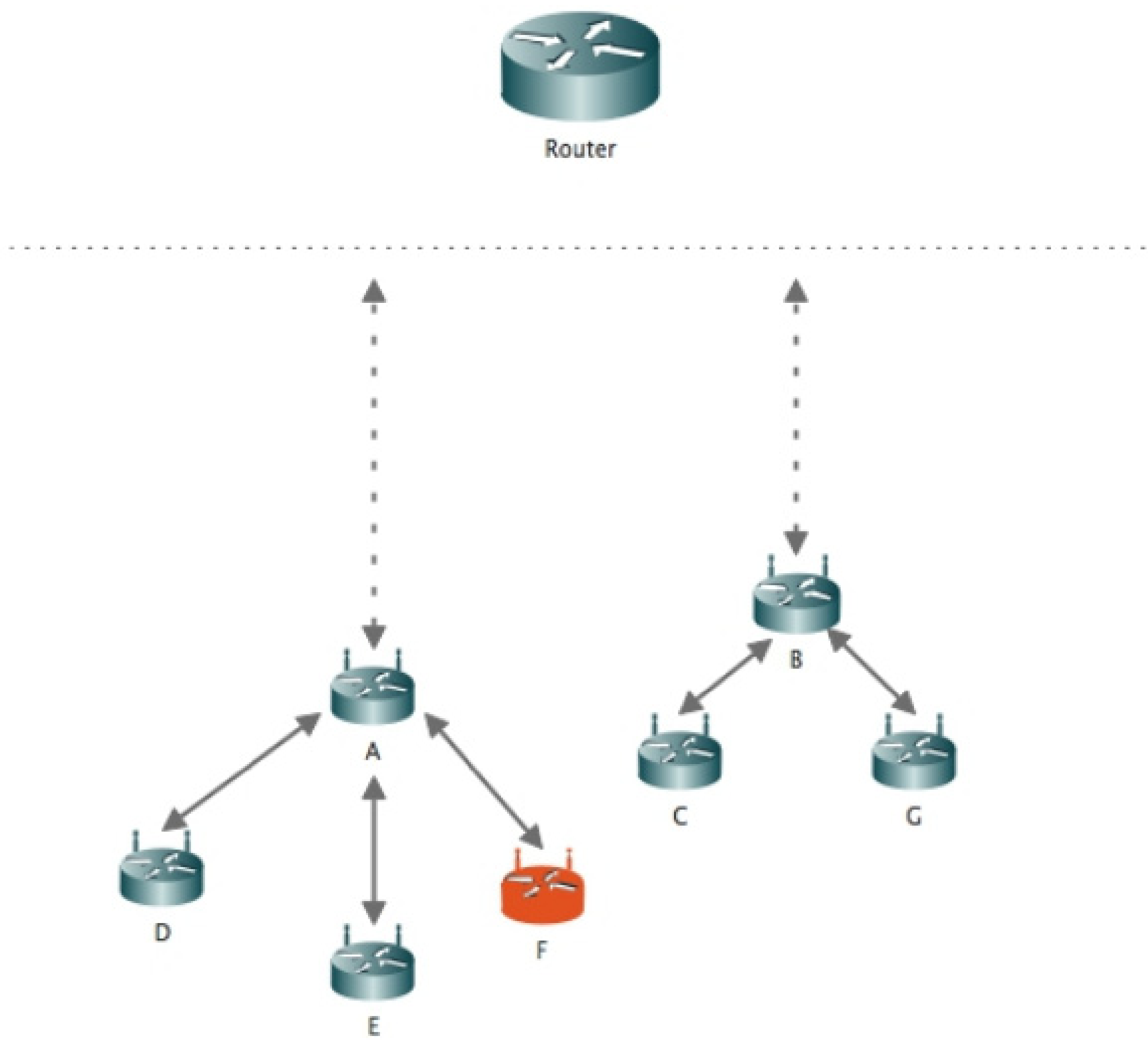

Figure 4

As described herein, multi-routes may be used to solve this problem. Two different routes may be generated for each node according to data priority. This may be analogous to a traffic system, where lower priority data uses a “highway” and higher priority data uses an “expressway”. Furthermore, the nodes may be divided into two groups based on a ratio n depending on the particular application scenario. This may help prepare for forming the multi-routes. Every node needs to form two routes for different priority data, so it is important to choose the right amount for either type of data. If too many nodes are used for high priority data but have very little throughput, this will waste valuable bandwidth resource, and vice versa. That is why a fixed ratio may be less advantageous.

In a first example step, nodes may be divided into two groups with a predefined ratio n (e.g., if n is equal to four, the number of High Priority Path Unit (HPPU) will be 25 percent of the total nodes.).

Every node may generate a pseudo-random number by using a pseudo-algorithm, (e.g., hash algorithm with a Media Access Control (MAC) address as the seed). Using this pseudo-number modulus n, two groups may be generated.

$$\text{NodeIndex} = \text{hash}(\text{MAC address})\text{mod}(n)$$

If the value of NodeIndex is equal to zero, this node will be set for HPPU. Otherwise, it may be a Normal Priority Path Unit (NPPU) node. If n is equal to four, 25% of nodes are HPPUs and the remaining 75% are NPPUs, as illustrated in Figure 5 below. The brown is the border router, the green nodes are NPPUs, and the other red units are HPPUs.

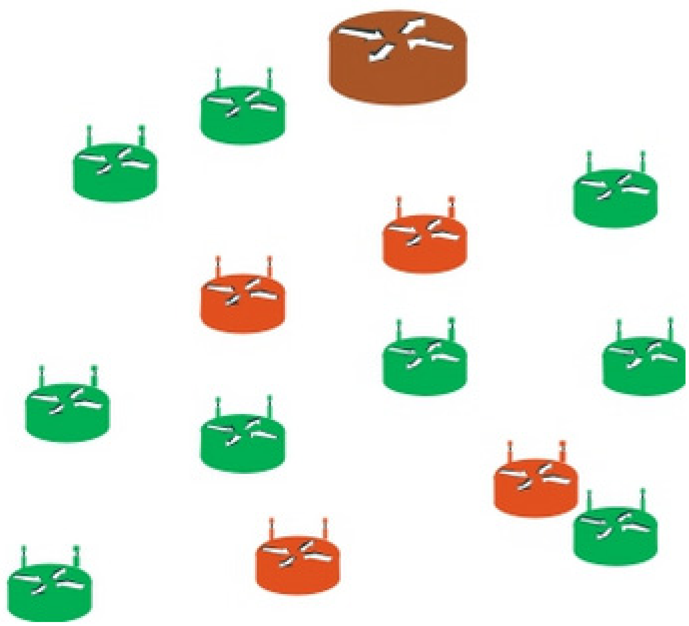

Figure 5

The more high priority throughput that exists, the more HPPUs are needed, which depends on application scenario requirements.

In a second example step, as illustrated in Figure 6 below, each node should have two routes from itself to the border router. One is the normal link route which only contains green nodes and the other is the high priority link route which only contains red nodes.

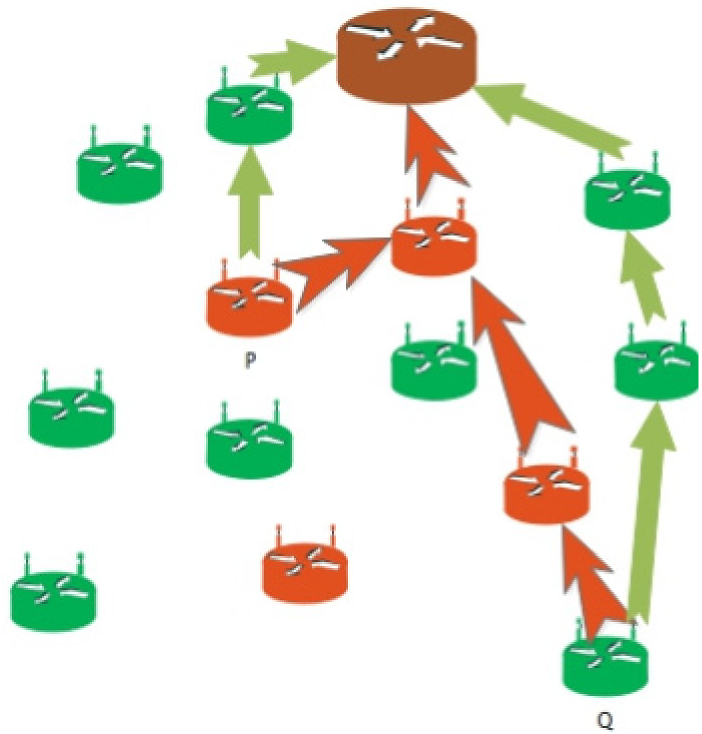

Figure 6

In one example, Node Q is a NPPU and has two routes to the border router. The green path is used to transmit normal packets and the red path is used for high priority packets. Similarly, Node P, which is a HPPU, also has two routes to the border router.

Either the NPPU or HPPU could produce two priority packets. Thus, two parents are needed.

At a third example step, due to the randomness of deployment in practice, there may exist some other extreme cases aside from what is already described above.

In a first example case, a node (either HPPU or NPPU) is surrounded by all NPPUs, so it cannot find a HPPU to set up a high priority link. As illustrated in Figure 7 below, Node O cannot find a HPPU neighbor as its high link parent.

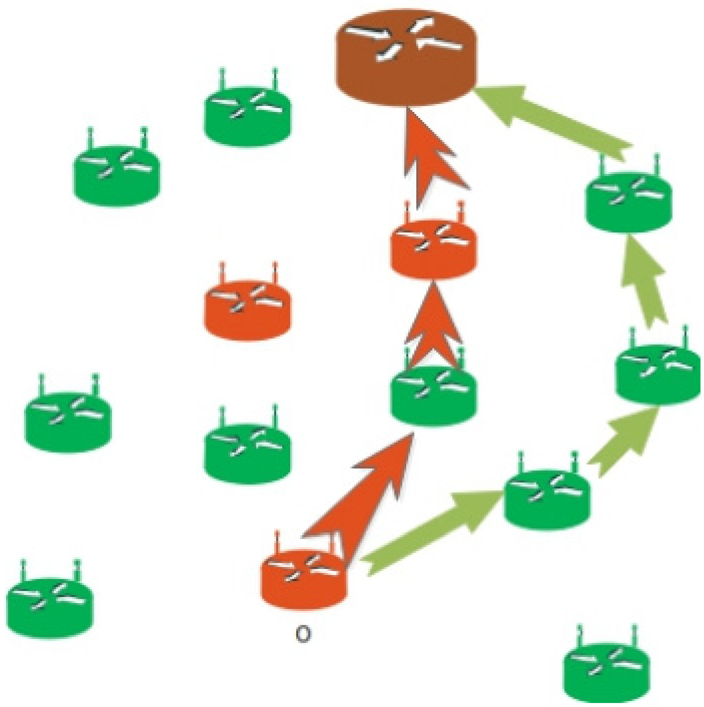

Figure 7

As shown in Figure 8 below, it could pick up a NPPU as its high link parent and send a request to request that this node serves as a HPPU.

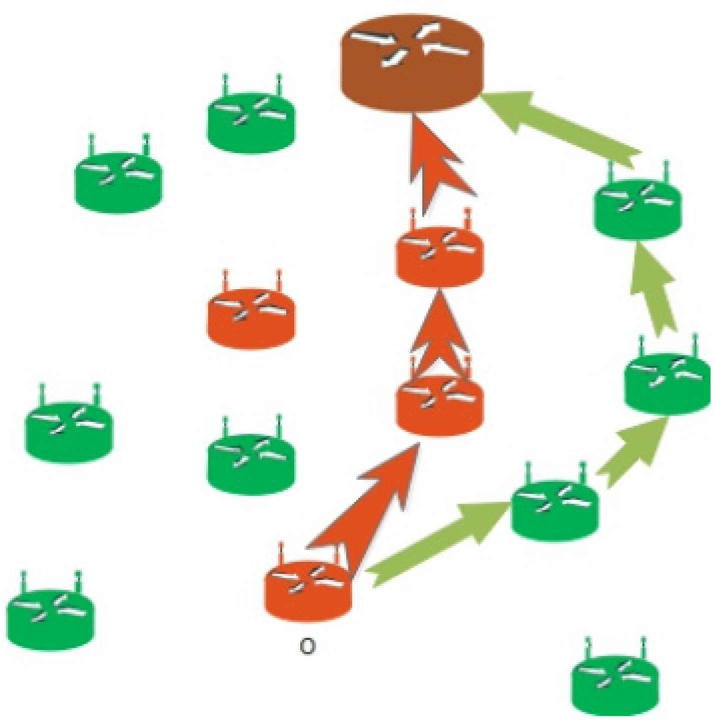

Figure 8

In a second example case a node (either HPPU or NPPU) is surrounded by all HPPUs. This is similar to the first example case, and as such the solution is also similar. Here, the node that has this trouble may send a request for a HPPU neighbor to serve as a NPPU as its normal link parent node.

In a third example case, a node has only one parent and cannot find any more candidate parent nodes. In this situation, its parent must relay any priority data.

At a fourth example step, once the node has built two links to the router, the router may learn the route information for each node. Thus, the router may send packets downstream along the corresponding routes according to the priority.

In summary, techniques are described herein to improve QoS performance of LLN. This guarantees a special high speed link for high priority data packets for LLN without reducing the performance of normal priority throughput.