



# Analysis of Maximum Depth of Wireless Sensor Network Based on RPL and IEEE 802.15.4

Yun-Shuai Yu<sup>1</sup>, Cheng-Che Huang<sup>2</sup>, and Chih-Heng Ke<sup>2</sup>(✉)

<sup>1</sup> Department of Electronic Engineering,  
National Chin-Yi University of Technology, No. 57, Sec. 2, Zhongshan Road,  
Taiping District, Taichung 411, Taiwan (R.O.C.)  
yys@ncut.edu.tw

<sup>2</sup> Department of Computer Science and Information Engineering,  
National Quemoy University, No. 1, University Road, Jinning Township,  
Kinmen 892, Taiwan (R.O.C.)  
neol\_d2022@outlook.com, smallko@gmail.com

**Abstract.** The nodes in wireless sensor networks (WSN) are typically resource constrained so that they can maintain only a few routes. More-capable nodes can insert extra routing information, i.e. source routing header (SRH), into packets to instruct the resource constrained ones to route the packet. A WSN with deeper depth requires longer SRH, thus leaving less space of a packet for the user data. We analyze the relationship between the length of the user data and the maximum depth of a WSN based on RPL and IEEE 802.15.4. The results can guide the application designers and the network administrators in selecting a suitable length of user data to guarantee that the data can be routed to each sensor nodes. Simulation results prove the correctness of our analysis.

**Keywords:** Wireless sensor network · RPL · IEEE 802.15.4 · Depth  
Source routing header

## 1 Introduction

In the last decade, wireless sensor networks (WSN) [1–4], have gained a tremendous attention due to the fourth industrial revolution or Industry 4.0. In a WSN, most nodes are constrained in resources such as processing capacity, energy capacity, and memory. Due to the resource constraints on the nodes, IETF proposes IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [5] to address the routing problem. Within an RPL routing domain, a sensed environmental datum will be routed upward from a node to the root of the WSN. Typically, the root node relays the sensed data to a process automation controller or a computer via a wired link. Messages, such as queries or configurations, may be sent from the process automation controller or the computer to the sensor nodes. Those messages, at first, are relayed to the root node and then routed downward to the destination nodes. In an RPL domain, each sensor node records at least one parent node which is one of the immediate successors of the node on a path towards the root. Each node forwards the data packets to its parent node, thus achieving the

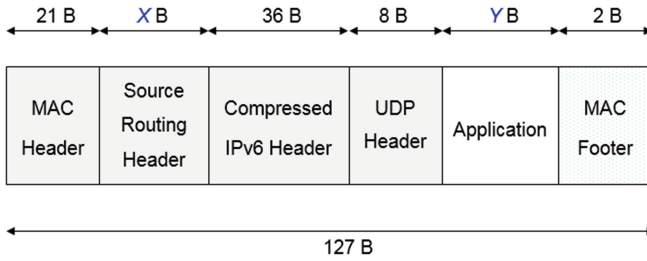
upward routing. In contrast to the upward routing, less-capable nodes are limited to maintain routes to other deeper nodes. So the process automation controller, the computer, or the root has to insert a source routing header (SRH) into the packets to instruct the resource constrained ones to route the packets. Therefore, downward routing consumes more space of a packet so that a user message may be too large to fit the packet.

A straightforward workaround is to be aware of the actual available space for user message and then the user can regulate the message to conform to the space limitation. In another way, the network administrator has to reconfigure the network topology to ensure that the depth of the WSN is not too deep since there should be a longer SRH for deeper nodes. Usually, it is expensive to design a flexible application which can accommodate the user messages to the topology of a WSN. Thus, network depth adjustment is a more economic solution. So the network administrator should be able to predict the maximum depth of the WSN when the maximum length of the user messages can be known in advance. One challenge is that WSNs based on RPL usually adopts IEEE 802.15.4 [6] as its physical layer and data link layer. IEEE 802.15.4 only supports frames of up to 127 bytes. It means that an adaptation layer is required to be above the IEEE 802.15.4 to compress/decompress the network layer protocol data units. Apparently, the performance of the adaptation layer affects the available space for user messages, which further affects the maximum depth of the WSN. To this end, this paper discusses how to determine the maximum depth of a WSN in given network topologies. For a better understanding of the analysis of the maximum network depth, OpenWSN [1] is used as an example WSN throughout this paper to explain how the above-mentioned factors consume the space. OpenWSN currently adopts 6LowPAN [7] as the adaptation layer and uses 6LoRH [8] to generate SRH. We adopt the simulator of OpenWSN to conduct experiments for the validation of our analysis.

The remainder of this paper is organized as follows. Section 2 describes the analysis of the maximum depth of a practical WSN. Section 3 describes the experimental methodology and results. Finally, Sect. 4 concludes this study and indicates the intended direction of future research.

## 2 Analysis of Maximum Depth of WSN

When a source host outside a WSN sends a user message to a sensor node inside the WSN, downward routing is performed to forward the message from the root of the WSN to the sensor node. In OpenWSN, the structure of a data frame for the above-mentioned user message is shown in Fig. 1. According to all the related standards and specifications, only the MAC footer field has a fixed length of 2 bytes. The lengths of all the remaining five fields in Fig. 1 should be variable. Thus, it is difficult to derive the maximum length of the SRH, which further determines the maximum depth of the WSN. However, the lengths of the UDP header and the compressed IPv6 header are currently fixed due to the simplified implementation of OpenWSN. When routing downwards, the UDP header is not compressed and the IPv6 header is compressed from 40 bytes to 36 bytes. In addition, the length of the MAC header is also fixed to 21 Bytes due to the simple network configuration. Since all nodes join the same PAN, i.e. personal area network, the addressing mode of the MAC header remains identical.



**Fig. 1.** The structure of a data frame using downward routing. Its source is a host outside a WSN and its destination is a node inside the WSN.

Now, only the SRH field and the application protocol data unit (APDU) have variable lengths. Since the maximum length of the APDU can be determined by the manual of the WSN application, the maximum length of SRH can be calculated by the following equation where  $X$  is the length of the SRH and  $Y$  is the length of the APDU.

$$X = 127 - 21 - 36 - 8 - Y - 2 = 60 - Y. \tag{1}$$

After determining the maximum length of the SRH, the maximum depth of the WSN can be derived based on the collected network topology and the SRH structure as shown in Fig. 2. The first byte of the SRH is a 6LoWPAN dispatch indicating the following values have to be parsed according to 6LoRH. One or several *Type-Length-Value* (TLV) field(s) will follow the dispatch. The Length field consists of a *Critical Format* field and a *Type Specification Extension* (TSE) field. In OpenWSN, TSE field is used as a Size, which will be explained later. The *Type* field can have five different values, which are 0, 1, 2, 3, and 4. The length of the *Value for the type* field is determined by the TSE value and the Type value.

Name	6LoWPAN Paging Dispatch	Critical Format	Type Specific Extension	Type	Value for the type	...
Length	1 Byte	3 bits	5 bits	1 Byte		...
Value	$F1H$	$100_2$		0, 1, 2, 3, 4		...

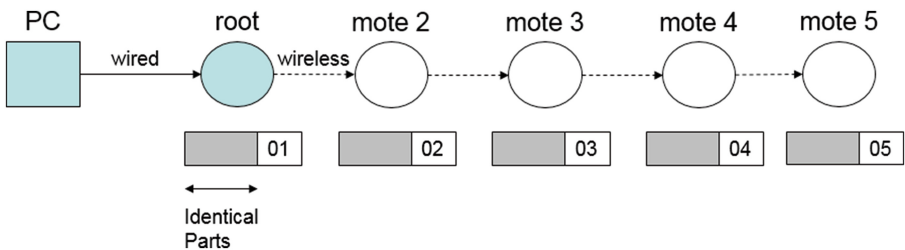
Type-Length-Value (TLV) field
 
 Other TLV fields

**Fig. 2.** The structure of the source routing header specified by 6LoRH.

The value of the *Type* field indicates the similarity of the IPv6 addresses of two consecutive nodes, i.e. two nodes of one hop, on the path towards the destination node.

If the most significant 15 bytes of their IPv6 addresses are the same, the value of the *Type* field is 0. If the length of the identical parts is 14, 12, or 8 bytes, the value will be 1, 2, or 3 respectively. For the remaining cases, the value will be 4.

The *TSE* field encodes the number of hops with the same type minus 1. Figure 3 shows an example network to explain how to determine the value of the *TSE* field. In the example, the PC sends a message to mote 5. Since the compressed IPv6 header can teach the mote 4 to forward the packet to the mote 5, the SRH should contain only the routing rules for three hops: (1) one from the PC to the mote 2; (2) one from the mote 2 to the mote 3; and (3) one from the mote 3 to the mote 4. Note that the wired link connects the data link layers of the PC and the root. Hence, the first hop should be from the PC to the mote 2. Usually, the IPv6 address of the PC differs a lot from the nodes of the WSN. So, the type of the first hop is usually 3 or 4. Since only the least significant one byte of the nodes of the second hop and the third one is different, the type of the other two hops is 0. Therefore, the *TSE* fields of the first TLV field should be 0, which is 1 minus 1. The *TSE* fields of the second TLV field should be 1, which is 2 minus 1.

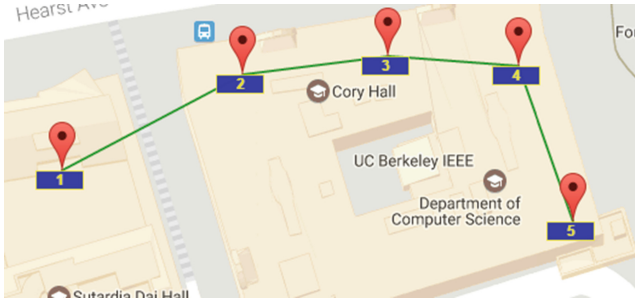


**Fig. 3.** Example network for describing the value of the *TSE* field. The mote 5 is the destination node while the PC is the source node. The most significant 15 bytes of the IPv6 addresses of all the wireless nodes are identical. They only differ in the least significant one byte.

The *Value for the type* field records the compressed IPv6 addresses of the destination nodes of the hops of the same type in the same *TLV* field. For example, assume that the IPv6 address of the PC is `bbbb::1` and the one of the mote 2 is `bbbb::1415:92cc:0:2`, the *Value for the type* field of the first *TLV* field is `1415:92cc:0:2`. For the second *TLV* field, the *Value for the type* field contains one byte of value `0x03` followed by one byte of value `0x04`.

### 3 Experiments

We adopt the simulator, OpenSim, provided by OpenWSN to validate our analysis. The network topology and the IPv6 address configuration is shown in Fig. 3. Besides, the PDR (Packet Delivery Ratio) of each wireless link is set as 1. That is, all frames can be successfully transmitted except collision occurs. Figure 4 shows the network topology displayed in the user interface of the OpenWSN simulator.



**Fig. 4.** Topology displayed in the user interface of the OpenWSN simulator. There are five wireless nodes. Besides, the PC is not shown in the user interface.

After all the nodes join the WSN, we send some messages from the PC to the mote 5. According to our analysis, the length of the SRH will be 15. Thus, the maximum available space for user messages will be 45 bytes. Figure 5 shows the simulation results. If the length of the user message is 45 bytes, which is shown in the red circle on the right of Fig. 5, the mote 5 can receive the message, as shown in the red circle in the left of Fig. 5. If the length of the user message is 46 bytes, which is shown in the yellow square on the right of Fig. 5, some critical errors happened, as shown in the left of Fig. 5. Thus, the simulation results validate our analysis. In addition, assume that the depth of the root is 1 and the maximum length of the user message is 40, the maximum depth of the WSN, in this example, will be 10.

Mote 5 joins WSN.

<pre> received RPL DAO from 14-15-92-cc-00-00-00-05 - parents:   . 14-15-92-cc-00-00-00-04 - children: UTyphoon received a message. 22:33:23 INFO 5 [UTYPHOON] unknown message type 0 [CRITICAL] radio_loadPacket() failed setting list item [CRITICAL] radio_getTimerValue() returned NULL [CRITICAL] radiotimer_schedule() returned NULL Exception in thread Timeline: Traceback (most recent call last):   File "C:\Python27\lib\threading.py", line 810, in __bootstrap     self.run()                 </pre>	<pre> D:\OpenWSN\openwsn-sw\python udp_ipu6_client.py UDP target IP: bbbb::1415:92cc:0:5 UDP target port: 15001 Message: 01234567890123456789012345678901234567890123456789012345 Message length: 45 D:\OpenWSN\openwsn-sw\python udp_ipu6_client.py UDP target IP: bbbb::1415:92cc:0:5 UDP target port: 15001 Message: 01234567890123456789012345678901234567890123456789012345 Message length: 46                 </pre>
---	--

**Fig. 5.** Simulation results.

## 4 Conclusions

In this paper, we analyze the relationship between the maximum depth of a wireless sensor network and the maximum length of user messages. The knowledge is helpful for network administrators to configure their network topology. The simulation results prove the correctness of our analysis.

**Acknowledgements.** The authors would like to thank the Ministry of Science and Technology, Taiwan, R.O.C., for supporting this research under grant MOST 105-2221-E-167-035.

## References

1. Watteyne, T., et al.: OpenWSN: a standards-based low-power wireless development environment. *Trans. Emerg. Tel. Tech.* **23**, 480–493 (2012)
2. Song, J., Han, S., Mok, A.K., Chen, D., Lucas, M., Nixon, M.: WirelessHART: applying wireless technology in real-time industrial process control. In: 2008 IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 377–386 (2008)
3. Yu, Y.S.: A framework supporting centralized routing in multi-hop TSCH networks. *IJ3C* **4**, 27–34 (2015)
4. Contiki: The Open Source OS for the Internet of Things. <http://www.contiki-os.org/>
5. Winter, T., et al.: RPL: IPv6 routing protocol for low-power and lossy networks. RFC 6550 (2012)
6. IEEE Std 802.15.4TM-2006: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPANs) (2006)
7. Hui, J.W., Thubert, P.: Compression format for IPv6 datagrams over IEEE 802.15.4-based networks. RFC 6282 (2011)
8. Thubert, P., Bormann, C., Toutain, L., Cragie, R.: 6LoWPAN routing header. draft-ietf-roll-routing-dispatch-05 (2016)