ISBN 978-93-84422-80-6
17th IIE International Conference on Computer, Electrical, Electronics and Communication Engineering (CEECE-2017)
Pattaya (Thailand) Dec. 28-29, 2017

An Efficient Routing Protocol with Multihop in Ad-hoc Sensor Network in SDN Environment

Jungha Jin[1], Minyoung Shin[2] and Keecheon Kim[1]

[1]Dept. of Computer Science & Engineering, Konkuk University

[2]Dept. of IT Convergence Information Security, Konkuk University

***Abstract*:** *In the autonomous control network environment, there are various ways to manage the nodes located under the controller management. We have proposed a node management method for managing nodes of ad-hoc networks that operate in a wireless network environment, by proceeding with research previously. In ad-hoc networks, communication between users is provided only through direct connections between nodes. That means are Ad-hoc network technology supports mobility directly through routing algorithms. Therefore, in an ad hoc network environment, when a node connected to the node loses its loss due to node movement, the routing protocol transmits the traffic to be transmitted to the other node. In the SDN environment, the controller monitors, detects, and manages the traffic generated by the lower nodes. In this paper, we propose a hop count management method that supports multipath algorithm for node management in SDN controller.*

Keywords: *SDN, Autonomous, Multipath, Path-through Hop-Count*

1. Introduction

As sensor technologies have advanced, with recent wireless technologies, autonomous and flexible wireless sensor networks can now be used to send or receive data between moving nodes where by each node recognizes and measures its own sensor environment and processes data without infrastructure system. In particular, ad-hoc sensor network technology enables direct communication between devices so that they can send or receive signals directly between two close users without passing through APs or base station communication networks, thereby effectively providing mobility between nodes compared to existing centralized network technologies. The connections between nodes in the SDN environment show similar behaviour to those of the ad-hoc network environment. However, networks between nodes are developed based on existing fixed APs or base station communication networks. Therefore, there is still a long way to go until an ad hoc network is implemented for direct communication between the devices of the sensor network operating in the SDN environment. Furthermore, ad-hoc networks need rapid response to failures between nodes because it consists of real-time communication. But existing routing protocols only support rapid recovery of lost nodes, so ad-hoc networks have a problem of significantly low efficiency owing to the lack of support while data is transferred [1][2][3]. To address this, many similar techniques have been explored to manage multiple connection paths to manage inter-node connectivity in an SDN environment.

In this paper, we propose an efficient and practical connection management scheme by adding hop count in the existing multipath maintenance multi - path management scheme [4]. The proposed scheme can effectively overcome the restriction of data transmission due to the occurrence of missing nodes in the ad hoc sensor network of the SDN environment, and it can provide the best connection continuity with minimum resource consumption. In addition, it provides more efficient management by recording the counter hops describing the hops to be passed in the routing table.

https://doi.org/10.17758/ERPUB.E1217120

This paper is organized as follows. In Chapter 2, we describe the routing protocols used in ad hoc networks in the SDN environment. In Chapter 3, we propose a multi-path-hop count routing algorithm proposed in this paper. In Chapter 4, we prove that the proposed method generates more efficient routing path than existing single-path routing protocol or existing multi-path routing protocol. Finally, the conclusion is presented in Chapter 5.

2. Related Work

Chapter 2 describes the SDN and explains the routing protocol method used to facilitate communication between nodes in an ad-hoc sensor network in the SDN environment

2.1. SDN (Software Defined Networks)

SDN is a networking technology that can flexibly handle network routing and control and complex operation management with software programming. The SDN separates the data plane and the control plane of the network so that the network operator can control the communication functions in the data plane in various ways through the programming of the control palne with various situation in mind. It is a network control and management technology that enables flexible control and effective network management based on the separation of the network control function from the switch and centralization by securing the visibility of network resources. SDN is based on the two basic principles [5].

SDN should be capable of Software Defined Forwarding, which means that the data forwarding function that is processed by hardware-based switches and routers must be controlled through open interfaces and software. And in SDN, the development of advanced network management tools should be made possible through abstraction. These abstraction tools include the ability to monitor events across the network and control events or network elements such as topology changes and new flow inputs. The conceptual structure of SDN is represented by three hierarchial structures which are an infrastructure layer, a control plane, and an application layer.

The control plane is a network controller that controls network operations using a global view of the overall network state and the application layer is located in an application operating in the upper layer. Southbound interfaces and Northbound interfaces exist for seamless interoperation between layers.

Figure 1 shows the hierarchical architecture of SDN at the level based on this.

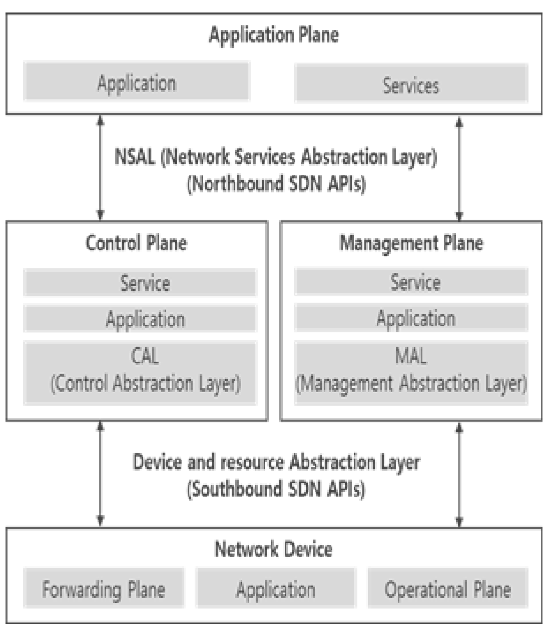

Fig 1 : SDN hierarchical architecture

SDN is proposed as a solution to the flexibility and scalability of the existing network due to the increase of traffic and data moving through the network through OpenFlow [6] which was developed as part of future Internet infrastructure technology research. Since then, the standardization of the underlying technologies that make up the SDN has been done, and the conceptualization has developed and technically embodied. The SDN consists of three layers: the network device layer where the data plane is located, the control layer where the control plane and the management plane are located, and the application plane where the third-party application is located.

The network management function is concentrated in the SDN controller, so that it manages the network in a comprehensive manner and network design and operation can be configured in a simple form of network. Finally, the network manager does not need to manually manage the configuration information distributed in a plurality of devices by simply configuring the abstracted network as a program, and is easily controllable.

2.2. Routing Protocols for Ad-Hoc Sensor Networks

Routing protocols used in ad-hoc sensor networks can be classified into proactive and reactive techniques depending on routing path setting timing or how they operate. First, with proactive techniques, all nodes hold the routing table and exchange routing information between adjacent nodes, thereby setting the routing paths for all nodes in the ad-hoc network in advance.

DSDV (Destination-Sequenced Distance-Vector Routing) and OLSR (Optimized Link State Routing Protocol) techniques are proactive. In reactive techniques, when data to be transferred is generated in a sending node, path-setting packets are exchanged between nodes to set up the routing paths.

AODV (Ad-hoc On-demand Distance Vector) and DSR (Dynamic Source Routing) techniques are reactive [7]. Since all communication nodes manage the routing table and set up the routing paths for other nodes based on routing information with adjacent nodes in the proactive technique, it is called a table-driven method. On the other hand, the reactive technique is called an on-demand method because when data to be transferred are generated, a sending node starts a setting procedure for transfer paths.

Table 1 summarizes table-driven and on-demand methods.

TABLE I : Ad-hoc routing protocol

Classification	table-driven METHOD	on-demand METHOD
Type	DSDV, OLSR	AODV, DSR
Characteristic	The routing path between all nodes is set in advance	When the transfer date is generated, the routing path is then set
Advantage	Transmission delay minimization	Reduce administrative overhead messages
Disadvantage	Increase in management overhead message	Transmission delay occurs

2.3. Reliable Hybrid Multi-Path Routing

To construct a reliable ad-hoc sensor network, the RHMR (reliable hybrid multi-path routing) [8] method is used. This method is operated as the routing protocol sets the primary path back-up path. When failures occur in the primary path, connections between nodes are recovered in real time through back-up paths to replace the failed primary paths and new back-up paths are then setup. To achieve this operation, a table management mode, such as PRDM (periodic route discovery message), or an on-demand method, such as ORRM (on-demand route recovery message), is used. In order to update routing information between nodes periodically, routing information is updated dynamically with regards to link states or lost nodes owing to node movements via PRDM. If paths cannot be connected because of lost nodes over the primary path, the affected primary paths are immediately replaced with back-up paths to recover the connection in real time and a new back-up path is set up through ORPM. Using the above method, multi-paths are updated to improve reliability in the ad-hoc network through rapid path recovery.

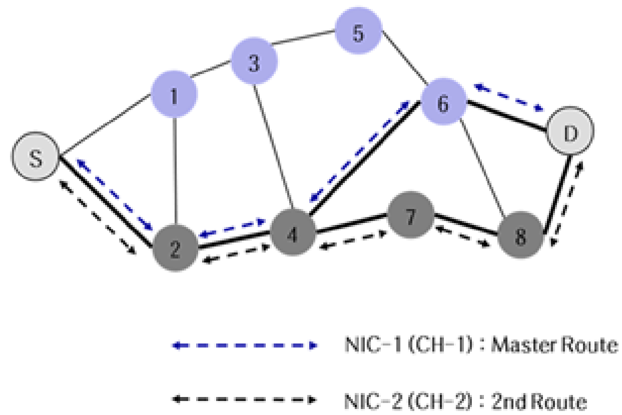

Fig 2: Reliable hybrid multi-path routing (RMHR) behaviour

2.4. Multipath Routing Algorithm Performed in pre-Research

The basic structure of the Multipath routing protocol is as follows:

Message type	Time to live	Sequence number	Source	Destination	Previous hop

Fig. 3: Routing protocol

- Message type: there are three message types: route requests (RREQs), route replies (RREPs), and route errors (RERRs).
- Time to live: a source node sets a certain lifetime value for a routing packet and whenever a packet passes through a hop, this value is decremented by one. If this value becomes 0, the packet is destroyed. By calculating this value conversely, a hop-count is calculated.
- Sequence number: in RREQ, a random number is used and if this value is duplicated, it is distinguished by using a source or destination value. In RREP, a number returned from RREQ is used without change. In RERR, node number is used where the routing table has a problem.
- Source: in RREQ, information from the node that sent the packet first is used and in RREP, information received from RREQ is used without change. In RERR, information from the node that detected the problem is used.
- Destination: in RREQ, information from the destination node is used and in RREP, information received from RREQ is used without change. In RERR, information from the node that had the problem is used.
- Previous hop: each node inserts its own information here when sending a routing message.

3. Proposed Method

In the routing method proposed in this paper, each node over the SDN environment ad-hoc sensor network maintains two different next hops for a single destination. The reason for maintaining two different hops is because if only one hop is maintained then the multipath method cannot be used and data transfer using another path cannot be made immediately if a node is lost. Moreover, if more than three hops are maintained for a multipath configuration, it can increase resource consumption exponentially, such as increasing the number of messages transferred and memory usage. Then, the number of hops passes through each hop, and the number of hops is increased.

The reason for increasing the count hop is to optimize the route of the routing table based on the accumulated information as it goes through the hops. In order to maintain multipath in an SDN network environment, it is possible to allocate optimal paths using increasing hops via paths, more efficient management than conventional methods, and improve availability.

3.1. Routing Protocol

The structure of the routing protocol proposed in this paper is as follows:

Fig 4 : PTHc Routing Protocol

- The Message Type, Time to Live, Sequence Number, Source Address, Destination Address, Previous Hop are the same as those described in the conventional multipath routing algorithm described above.
- Path Through Hop Count: The value that the message increments via hop. Use this value to assign an optimal path.

3.2. Multipath Routing Algorithm Operation

The routing table stores flags, master next hop, second next hop, destination, sequence, and Path Through Hop count(PTHc). Figure 5 below shows the routing table configuration at the source node. Flag is set to 0 if the destination is a non-adjacent node. Sequence is stored to determine the operation for the routing packet of the duplicated sequence. PTHc shows how many hops have passed from the departure place.

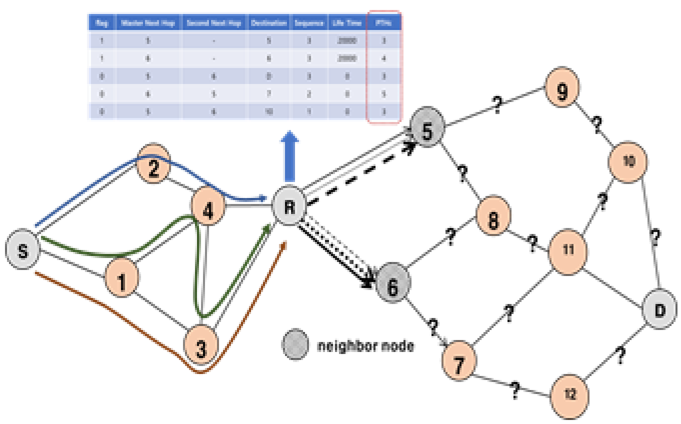

Fig 5 : Multipath Routing Table

In the single path, as the number of hops increases, the flooding for path re-searching increases, resulting in a higher cost for recovery than the cost for path generation. Therefore, comparing with the multipath algorithm with multipath, it can be seen that the cost of re-searching the path becomes higher. As the number of disconnected links increases, the cost of path creation / maintenance will increase.

3.3. Add on Path Through Hop Count(PTHc)

In a routing algorithm that uses a multipath, resources are consumed to maintain the multipath path, and it may be confusing to decide which path to select. To overcome this problem, we propose more efficient path management by adding Path Through Hop count(PTHc) to existing multipath routing algorithm. If the node is routed through a node, it is determined which path the node moves through by adding a hop count, and the network is managed more smoothly by discarding an inefficient path. The benefit of this operation is that the node can more easily connect to the optimal path. The add on PTHc will be described in detail in the following section.

4. Path Through hop Count Routing Algorithm

The addition of the route hops is simple. The number of hops that will increase each time through a node works in the similar way as the Ant Colony Optimization(ACO) algorithm [9][10] which is one of bio-inspired algorithms. However, in contrast to ACO algorithm, it is possible to exclude routes with a large number of hops, enabling efficient multipath management. In other words, the fact that the hop count is increased means that the efficiency of the routing is degraded, and it is possible to provide an optimized route based on this.

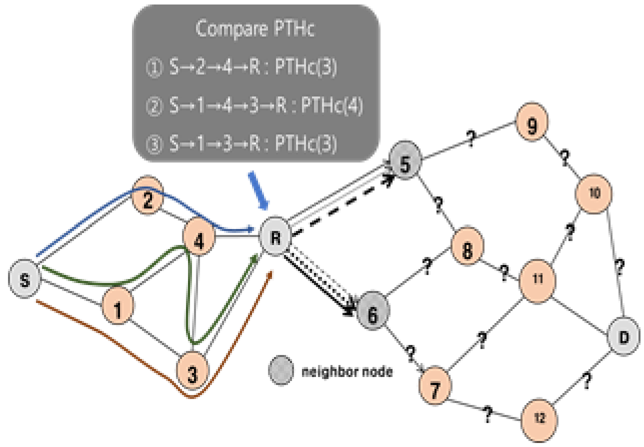

Fig 6 : compare Path Through Hop count

4.1. Remove Routing Path

As mentioned earlier, the proposed algorithm shows the opposite result to the ACO algorithm. The reason for this operation is to quickly catch an overload on a specific node, and to discard the path where congestion occurs, thereby performing autonomous management. However, in order to perform such management, a part of the existing routing table structure needs to be modified, and it has a structure as shown in the following figure.

flag	Master Next Hop	Second Next Hop	Destination	Sequence	Life Time	PTHc
1	5	-	5	3	20000	3
1	6	-	6	3	20000	4
0	5	6	D	3	0	3
0	6	5	7	2	0	5
0	5	6	10	1	0	3

Remove large PTHc in the routing table

Fig 7 : Path Through hop count routing table

In this figure, a column called PTHc is added, which means hop count. Based on the numbers entered here, the node can choose where to route the next, so it can keep the connected network efficient and secure. Therefore, through comparing the existing path registered in the routing table and the PTHc received in the packet to be newly transferred, it operates to maintain the optimum path that can be efficiently transmitted.

4.2. Provide Reliable Path Information

It can be used by transforming the algorithm that was proposed previously with a funny idea. Users are concerned about how to check reliable nodes that are operating stably with many nodes connected. From the viewpoint of the node, it is interpretable that many nodes can trust meaning that PTHc is large and that various paths can be used. If large PTHc comes from a lot of places, it may mean that it is easy to connect. Of course, in the opposite case, it may take a line to load, but the large PTHc may be the more reliable source route.

The hint obtained here can be interpreted as a reliable signal that the hop count is high in a situation where the resources of the node are sufficient. So, if we construct a reliability chain based on this information and connect the nodes, we think it will be a safe network environment implementation. This part is considered to be one of applications of biomimetic algorithm.

5. Conclusion

In this paper, we examined the effective network configuration of nodes operating in SDN environment. We propose a path management scheme based on multipath routing based on the existing research. The operation of this algorithm autonomously manages to maintain an efficient network by discarding paths that path through. The advantage of this algorithm is that it provides the advantage of autonomous network management in the communication through synchronization between nodes. If it is solved in the opposite way, reliable nodes can be selected, and reliable chaining can be constructed based on this. The multipath algorithm enables immediate data transmission to other hops that can be connected, ensuring the continuity of the connection, thus providing faster connectivity than the conventional method and showing excellent performance. When the proposed method is applied to the sensor network and the multi-hop is maintained through synchronization between the nodes, the function of solving the problem of data transmission due to the frequent movement and loss of nodes is solved. However, it may be difficult to maintain the route through multiple routes in areas where multiple routes are crowded. Therefore, we propose a multi-hop hop count scheme that can manage the network autonomously.

6. Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (2014-0-00547, Development of Core Technology for Autonomous Network Control and Management) and Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (No. NRF-2017M3C4A7083678).

7. References

[1] Stephen Mueller, Rose P.Tsang, Dipak Ghosal, “Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges”, Performance Tools and Applications to Networked Systems, 2004, vol. 2965, pp. 209-234.

[2] A. Gecyasar, T. Braun, “Ad-hoc multipath routing protocol”, Bachelor, Institute of Applied Informatics and Mathematics, 2006.

[3] M. Radi, B. Dezfouli, K.A. Bakar, M. Lee, “Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges”, Sensors, Jan 2012, vol. 12, no. 1, pp. 650-685B. Smith, “An approach to graphs of linear forms (Unpublished work style),” unpublished.

[4] Woobin Kim, Jungha Jin, Keecheon Kim, "A Routing Protocol Method that sets up Multi-hops in the Ad-hoc network", ACM IMCOM(ICUIMC)’16, January 4–6, 2016

[5] ONF White Paper, “Software-Defined Networking: The New Norm for Networks,” Apr. 2012.

[6] Nick McKeown et al, “OpenFlow: enabling innovation in campus networks,” ACM SIGCOMM Computer Communication Review, April 2008, vol. 3, pp. 69-74.

[7] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Boloni, D. Turgut, “Routing protocols in ad hoc networks:A survey,” Computer Networks, 2011, vol. 55, pp. 3032-3080.

[8] Sung-Won Lee, Ji Yong Choi, Keun Woo Lim, Young-Bae Ko, and Byeong-Hee Roh, “A Reliable and Hybrid Multi-path Routing Protocol for Multi-Interface Tactical Ad Hoc Networks,” MILITARY COMMUNICATIONS CONFERENCE, 2010-MILCOM 2010, pp. 2237-2242.

[9] Dorigo, M., Birattari, M., Stutzle, T., “Ant colony optimization”, IEEE Computational Intelligence Magazine, vol. 1, no. 4, pp. 28-39, Nov. 2006.

[10] M. Dorigo, C. Blum, “Ant colony optimization theory:A survey”, Theoretical Computer Science, vol. 344, pp.243-278, Nov. 2005.