A WSN Architecture Based on SDN

Qiaozhi Xu[1] , Jie Zhao[2]

1. College of Computer & Information Engineering, Inner Mongolia Normal University, Hohhot, China

2. College of Math & Information Technology, Jiangsu Second Normal University, Nanjing, China

E-mail: ciecxqz@imnu.edu.cn

Abstract: In this paper, we introduce software-defined networking (SDN) into wireless sensor networks (WSNs) for saving energy and smart management. We propose a generic architecture for the base station in a software-defined wireless sensor network. We also propose a general framework for a software-defined wireless sensor network where the controller is implemented at the base station. Lastly, we compare LEACH protocol and the SDN-WSN framework by discussing and analyzing, the performance and energy consumption of SDN-WSN framework is better than LEACH.

Keywords: Wireless Sensor Network; Software-Defined Networking; WSN; SDN

1. Introduction

With the development of electronics, wireless communications, and sensing technology, wireless sensor network (WSN) has attracted universal attentions of the whole world. WSNs are composed of the sensors with functions of sensing, data processing, and short-distance wireless communication. In military defense, disaster relief, environmental monitoring, biological and commercial applications, and other fields, WSNs have broad application prospects.

WSNs have the traditional characteristics of the wireless network does not have such as the great number of sensor nodes and high density, the sensors' energy, computing power, and storage capacity are limited; network topology changes frequently and has the self-organizing ability, and so on.

Software-Defined Networking (SDN) was developed to facilitate innovation and enable simple programmatic control of the network data-path. The separation of the forwarding hardware from the control logic allows easier deployment of new protocols, applications, network visualization, and management [1]. SDN allows network administrators to manage network services through abstraction of lower level functionality, this is done by decoupling the system that makes decisions about where traffic is sent (control plane) from the underlying systems that forwards traffic to the selected destination (data plane) [2].

Network intelligence is centralized in software-based SDN controllers, which maintain a global view of the network. Leveraging the SDN controller's centralized intelligence, it is possible to alter network behavior and deploy new applications and network services in real-time.

This paper argues for a novel clustered architecture for WSN based on the SDN paradigm. We argue that the global view that the SDN controller possesses will help overcome the inherent weaknesses of WSN.

The rest of this paper is organized as follows. In Section II, we present the benefits of SDN approach in WSNs. In Section III, we present a software-defined WSN framework. In Section IV, we describe the information dissemination in our SDN WSN architecture. In section V, we describe the proposed base station architecture for the controller. In Section VI, we present additional discussions about applying SDN in WSNs. Finally, in Section VII, we present the conclusions and discuss future works.

2. Related Works

An idea for managing WSN with SDN is proposed by Luo [3], with the concept of Software-Defined Wireless Sensor Networks SD-WSN. This study is based on the idea that each sensor node supports OpenFlow, and sensors should be able to recognize the flow table's entries. This architecture proposes a separation between data and control plane. The Sensor OpenFlow (SOF) is a communication protocol between the control plane and data plane. The data plane contains sensor flow packet forwarding, and the control plane is a controller for performing routing and QoS network control.

Copyright © 2016, the Authors. Published by Atlantis Press.
This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

Zeng [4] has studied the Evolution of Software-Defined Sensor Networks, integrating sensors nodes into cloud computing using a SaaS. This model is named Sensing-as-a-service and combines the sensing data with existing cloud services such as mash up services. The controller is a sensor controller with SDN functionalities and it is provided with a local database to store sensed data. In this work the authors assume that every sensor node is able to deliver packets to a sensor control server.

Gante [5] presented base station architecture for WSN based on SDN with a review of benefits of this technology. The authors mentioned the use of an SDN controller as a base station in WSNs, but did not present any detail about communication between sensor nodes. The controller can determine the best routing, forwarding decisions and inserting these decisions into sensor nodes flow tables. In others words, the sensor nodes do not make routing decisions, they only forward and drop packets according to the rules set by the controller (base station).

Contanzo [6] analyze the opportunities and challenges of SDN in IEEE 802.15.4 networks, implementing a scenario called SDWN. The controller is executed into sink nodes, and in order to communicate, each node must learn a path (as convenient as possible) to reach the controller. The controller periodically generates a beacon packet to send it to the nodes. Also, the nodes store the list of nodes from which they received a beacon packet. All discovered neighbors will be linked frequently to the sink node using a packet called a report packet.

Flauzac [7] proposed a clustered WSN Architecture based on SDN. Every node in WSN may be one of the three states: simple node, gateway node and cluster-head node, and the cluster-head is called SDN cluster-head (SDNCH). Each cluster is called a SDN domain. In each SDN domain, the SDNCH is the controller, in charge of managing the operation of the sensor nodes. The collected information about the environment is processed on the domain by nodes and will be routed to the SDNCH, and the gateway is engaged in aggregating and transmitting the data from entire sensor node domains to the other domains.

Many solutions have been proposed by different authors in the SDN and WSN field, but in these solutions, the cluster-head forwards or drops the data from the sensor nodes according to whether mapping the rule of the flow table, this will waste the node's energy. this paper proposed a novel WSN architecture based on SDN, firstly, the controller only generate route table among cluster-heads, when the CH receives data from sensors, it forwards directly based on its route table; secondly, application can custom the scope of data collection for different type sensors.

3. Architecture of SDWSN

In this section, we introduce the concept of SDWSN and explain the different technologies for the realization of SDWSN.

The WSN may contain hundreds or thousands of sensor nodes, and include a base station. The type of sensor nodes and their capabilities varies according to their application, such as temperature sensors, sandstorm sensors, etc. Normally, a large network cannot operate efficiently without some organized structure. For this reason, we propose to cluster the network. There exists one cluster-head (CH) in each cluster, and the other nodes are simple sensor (SN). In each cluster, the CH is in charge of managing the operation of the sensor nodes. With this clustering approach the collected information about the environment on the cluster by nodes will be sent to the CH.

A software-defined sensor network relies on a logically centralized controller. From the network point-of-view, the controller does not necessarily need be a standalone node. We propose that the control-logic be implemented as a part of the base station.

In the software-defined sensor network framework, the sensor nodes do not have to make routing decisions. Instead, they forward packets to the next cluster-head or base station according to the route table generated by the base station. In other words, the routes which are considered the best according to application-specific criteria are calculated by the controller (in the base station). In this framework, the controller makes use of location information gathered by any localization technique when finding the best routes.

In the base station architecture proposed by Gante [5], the controller needs to know the topology of the entire network. SDN has a higher potential to develop forwarding decisions of SN based on the rules set generated by the Controller, permitting a better cooperation among CH and SNs [3, 5, 6, 8]. Moreover, SDN controllers [5], can reduce the energy consumption by different sensor nodes, making the best routing decision for the nodes. With the network management controlled by the SDN, the routing decisions and policies have low convergence time in comparison with routing protocols. To deploy this architecture, CH has to manage the cluster network.

4. Information Dissemination in SDWSN

As shown in Fig.1, the main ideas of this network architecture are that CH node can be controlled by the controller （BS）, the specific forwarding strategy entirely depends on the controller, the whole message link is completely controllable, when CHs or SNs have some abnormal conditions such as energy depletion and fault conditions, the controller node can adjust strategy or change the forwarding path.

The core of this framework is the information dissemination strategy. The whole process can be divided into two stages.

A. Building Network

1) Initially, the controller knows nothing about the network, so the nodes must send their state information including their address, position, residual energy and types to the controller and register themselves to the controller.

2) According to the received information, the controller do three things. Firstly, the controller divides the network into some uniform clusters; secondly, considering the node's position and residual energy synthetically, the controller selects a CH for each cluster; thirdly, the controller generates the route table for all the CHs, telling the CHs the next hop for forwarding data to the BS.

3) The controller sends the cluster results and route tables to the CHs. The cluster results include the CH id and the member-id in the cluster.

4) The CH announces its own identification to the SNs in a cluster, and the SNs should response to the CH so that the CH can know the state in the cluster and report to the controller later.

B. Collecting Data

1) Firstly, the users set specific trigger conditions for different SNs according to the different needs.

2) Secondly, The BS push the user's configuration to all the SNs, then each SN sets itself. Only when the condition is satisfied, the SN start to send data to its CH. The CH collects and fuses the data, and forwards it to the next hop CH appointed by the route table generated by the controller, the CH needn't match the SN's data with flow table, because the data the SN sent is fit for the user's need. Sometimes, the user may use different controller for different application, we embed the FlowVisor [9] into the BS to reach the goal. The information from SNs to CH includes not only the collected data, as well as the residual energy of itself.

After some time interval, the network will change to building phrase again, the difference is that the nodes needn't send state information, because the date collected by SN includes the SN's residual energy, and the BS updates the node state information real-time, selects new CHs and computes new route according to the new network view, then publishes the new results to the new CHs. The new CHs do deployment in the cluster. If the user modifies the trigger conditions, the BS will push the new settings to corresponding nodes.

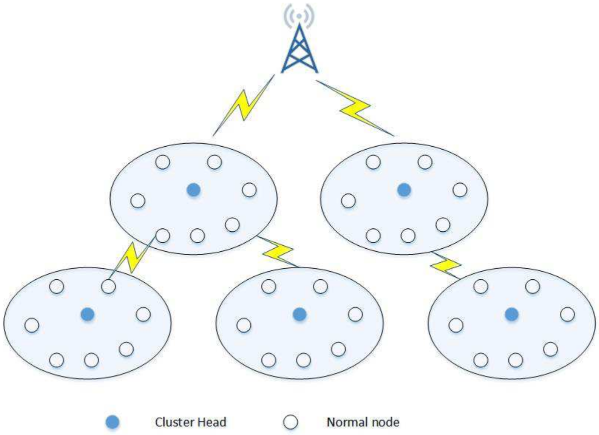

Figure.1 SDWSN Architecture

5. Components of Base Station

We propose a generic base station architecture for wireless sensor networks, in which including the controllers. This architecture simply provides reconfiguration abilities by the SDN paradigm. Simple network reconfiguration improves management features, such as energy saving, topology discover and so on. Fig.2 shows the proposed architecture emphasizing the units involved in the network.

A. Middle-Ware

This module is responsible for parsing and extracting the SN state information from data, updating the database, and forwarding the other data to the App-Manage-Module.

B. Route-Module

This module composites of three functions. CA (Cluster-Algorithm) is responsible for clustering the WSN according to the node state information, then CR (Compute-Route) generates the optimal path for CHs, finally, publish the result to the CHs. There is a timer in the Route-Module, when the time is out, the Route-module will read node's state from database, select new CHs, compute new path and publish new result for the WSN.

C. App-manage-Module

FlowVisor sends the packets to different controller according to the packet types, and different controller is managed by different applications and users. By this way, users can deploy heterogeneous sensors in the WSN and concern to different targets.

D. Node-State-Info

It stores the detail information about the nodes, including node-id, cluster-id, node-type, position, residual energy, etc. When the packets reach the Middle-ware from network, Middle-ware extracts node state information, inserts into or updates the database.

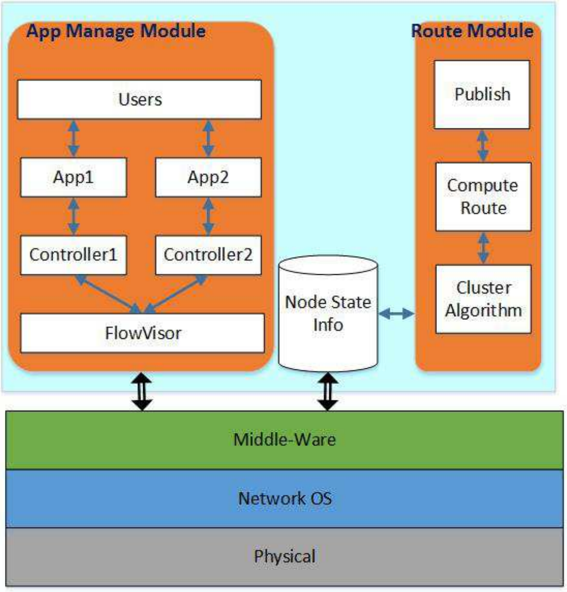

Figure 2. Components in the BS

6. Discussions

Assume for a certain application, a WSN needs to be deployed to cover a certain area where it needs to maintain a continuous monitoring. Using the proposed architecture, system behavior could be changed in an easier way. This could be done by modifying the forwarding rules in the base station (controller) and so, new action policies will be sent to the network devices. Furthermore, scaling the system becomes a simpler task. To extend the network to another area, new devices could be linked to the same WSN (same base station) which should allow for network policies to be sent to the new devices.

IN our SDN-WSN architectures, the sensor nodes don't have to support openflow protocol, the data sensed by the sensors meets the requirements of the users, so the cluster-head need not match the data with the rules, reducing the processing delay and energy consumption of the sensor nodes, in addition, the cluster-heads get the forwarding table through the controller, bringing down the latency and consumed energy of the whole wireless sensor network.

Moreover, introducing the flowvisor component into SDN-WSN architecture, can be convenient for different applications sharing the same physical devices and framework.

Since energy is a highly scarce resource in WSN, we intend to compare the SDN environment with and the traditional distributed clustered WSN network, running leach protocol [10]. In WSN, the node's energy mainly consumed in three aspects: transmitting data, receiving data, fusing data. At the set-up stage of LEACH, each node broadcasts own residual energy and computes to elect the CH, consuming a lot of energy; at the steady-stage, the CH receives data from other nodes, fuses data and running routing algorithm to select a path to forward the data, consuming large energy too, but the path selected by CH maybe not the optimal one. IN SDN-WSN, only the initial stage, the nodes need to report its energy so that the BS can have a global network view, so save a lot of energy. on the other hand, the nodes don't participate in the CH election, the CH don't need to run routing algorithm, these are done by BS, but the routing path is optimal, the whole energy consumption is reduced drastically.

In conclusion, SDN and WSN are both emerging technologies in network and communication fields, and SDN provides many benefits for various application circumstances, but how to integrate it into WSN and accelerate the advance of WSN, still need further discussion and research, we will study the subject more deeply in future.

7. Conclusions

In this paper, we explored the possibility of introducing the concepts of software-defined networks into wireless sensor networks. We argued that the introduction of SDN into WSNs could help in tackling some of the difficult problems in WSNs such as energy saving and network management. We also proposed an architecture for a software-defined wireless sensor networks, where the controller is integrated at the base station for generating the routing table for the cluster-head, controlling the data sensing of the sensor nodes, and providing more feasible and flexible and convenient management function for the users.

In future, we will study the SDN-WSN architecture continually, implement a prototype and build the suitable testing and evaluation methods to verify the performance of SDN-WSN quantitatively.

Acknowledgment

This work was financially supported by the Inner Mongolia Autonomous Region Natural Science Foundation (2012MS0930), and Inner Mongolia Autonomous Region scientific research projects of colleges and universities (NJZY12032).

References

[1] Nunes, Bruno AA, et al. A survey of software-defined networking: Past, present, and future of programmable networks. Communications Surveys & Tutorials, IEEE 16.3 (2014): 1617-1634.

[2] ONF Market Education Committee. Software-defined networking: The new norm for networks. ONF White Paper (2012).

[3] Luo, Tie, Hwee-Pink Tan, and Tony QS Quek. Sensor OpenFlow: Enabling software-defined wireless sensor networks. Communications Letters, IEEE 16.11 (2012): 1896-1899.

[4] Zeng, Deze, et al. Evolution of software-defined sensor networks. Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on. IEEE, 2013: 410-413.

[5] De Gante, Alejandro, Mohamed Aslan, and Ashraf Matrawy. Smart wireless sensor network management based on software-defined networking. Communications (QBSC), 2014 27th Biennial Symposium on. IEEE, 2014: 71-75.

[6] Costanzo, Salvatore, et al. Software defined wireless networks: Unbridling sdns. Software Defined Networking (EWSDN), 2012 European Workshop on. IEEE, 2012: 1-6.

[7] Olivier, Flauzac, Gonzalez Carlos, and Nolot Florent. SDN Based Architecture for Clustered WSN. Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2015 9th International Conference on. IEEE, 2015: 342 – 347.

[8] Yang, Fan, et al. OpenFlow-based load balancing for wireless mesh infrastructure. Consumer Communications and Networking Conference (CCNC), 2014 IEEE 11th. IEEE, 2014: 444-449.

[9] Sherwood, Rob, et al. Flowvisor: A network virtualization layer. OpenFlow Switch Consortium, Tech. Rep (2009): 1-13.