A Methodology for Reliability of WSN Based on Software Defined Network in Adaptive Industrial Environment

Ying Duan, Wenfeng Li, *Senior Member, IEEE,* Xiuwen Fu, Yun Luo, and Lin Yang

***Abstract*—As communication technology and smart manufacturing have developed, the industrial internet of things (IIoT) has gained considerable attention from academia and industry. Wireless sensor networks (WSNs) have many advantages with broad applications in many areas including environmental monitoring, which makes it a very important part of IIoT. However, energy depletion and hardware malfunctions can lead to node failures in WSNs. The industrial environment can also impact the wireless channel transmission, leading to network reliability problems, even with tightly coupled control and data planes in traditional networks, which obviously also enhances network management cost and complexity. In this paper, we introduce a new software defined network (SDN), and modify this network to propose a framework called the improved software defined wireless sensor network (improved SD-WSN). This proposed framework can address the following issues. 1) For a large scale heterogeneous network, it solves the problem of network management and smooth merging of a WSN into IIoT. 2) The network coverage problem is solved which improves the network reliability. 3) The framework addresses node failure due to various problems, particularly related to energy consumption. Therefore, it is necessary to improve the reliability of wireless sensor networks, by developing certain schemes to reduce energy consumption and the delay time of network nodes under IIoT conditions. Experiments have shown that the improved approach significantly reduces the energy consumption of nodes and the delay time, thus improving the reliability of WSN.**

***Index Terms*—Industrial internet of things (IIoT), reliability, software defined network (SDN), wireless sensor network (WSN).**

I. INTRODUCTION

Manuscript received May 31, 2016; accepted April 21, 2017. This work was supported by the National Natural Science Foundation of China (61571336), the Science and Technology Project of Henan Province in China (172102210081), and the Independent Innovation Research Foundation of Wuhan University of Technology (2016-JL-036). Recommended by Associate Editor Kamel Barkaoui. *(Corresponding author: Wenfeng Li).*

Citation: Y. Duan, W. F. Li, X. W. Fu, Y. Luo, and L. Yang, "A methodology for reliability of WSN based on software defined network in adaptive industrial environment," *IEEE/CAA J. of Autom. Sinica*, vol. 5, no. 1, pp. 74−82, Jan. 2018.

Y. Duan is with the School of Logistics Engineering, Wuhan University of Technology, Wuhan 430063, China, and also with the Department of Computer Science and Application, Zhengzhou University of Aeronautics, Zhengzhou 450015, China (e-mail: able0607@163.com).

W. F. Li, Y. Luo, and L. Yang are with the School of Logistics Engineering, Wuhan University of Technology, Wuhan 430063, China (e-mail: liwf@whut.edu.cn; yunluo@whut.edu.cn; lyang@whut.edu.cn).

X. W. Fu is with the Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China (e-mail: fuxiuwen1987@163.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2017.7510751

THE industrial internet of things (IIoT) has gained considerable attention recently from both academia and industry. IIoT includes not only wireless sensor networks but also other constituent parts including wireless local area network (WLAN), radio frequency identification (RFID) and mobile agents. Sensor networks are key components of IIoT and the vigorous development of communication technology and intelligent manufacturing have made various services possible in wireless sensor networks (WSNs), significantly improving the reliability and services of IIoT.

A. Traditional WSN

Research and industrial communities have undertaken several projects [1], [2] with the development of WSNs. For example, Distefano [1] used dynamic reliability block diagrams and Petri nets for sleep/wake-up standby policies and interference in order to improve the energy consumption of WSN nodes. The Germany Seeberger Company [2] focused on the traceability of food and quality assurance to monitor the data at each link using WSN, RFID, etc. Sensing devices are organized with a data processing unit and communication abilities, aiming to measure the specific surrounding parameters and join a heterogeneous network. These wireless devices have the capability to monitor a large range of environmental conditions such as sound, light, pressure, temperature, motion etc.

The network coverage problem [3] is a serious problem for sensor networks in traditional IIoT. WSN node transmission will be affected by node channel disturbances and energy depletion. Sensor nodes usually consume a large amount of resources, including processing power and communication bandwidth, so node power consumption is considered to be an important factor that also contributes to the WSN reliability. Moreover, the tightly coupled control and data planes in WSN increase the complexity and cost of network management. A new network framework is expected for the development of WSN.

B. SDN

A new network framework is required in order to develop WSN. Green IoT [4] was proposed a new concept and hierarchical system framework for general IoT development. We choose another new architecture known as a software defined network (SDN) [5] that is characterized as a centrally-controlled network. The relaying and controlling are performed separately, with user-defined virtualization and programming, which can provide flexibility and reliability to the network management. The SDN consists of a controller, a switch, an application layer, and terminal equipment [6]. Based on the ideas of network switches including forwarding and control element separation, integrated control, and programmable function, it is able to separate the controlling function from network switches [7], [8]. The separated functions are arranged using external controllers, which manage the switches and terminal equipment in the network by providing a unified command. The controller acquires the information of the overall network through network communication with the forwarding layer using the connecting switches via a southbound interface (OpenFlow). The forwarding layer is mainly responsible for data processing, forwarding and collection. The forwarding layer performs a corresponding action according to the flow table information, which is sent from the controller. The flow table entry information mainly includes matching an item, an instruction or a count [9]. Consequently, the capability to obtain a view of the whole network can be enhanced. Additionally, the controller realizes its overall control on the whole network by utilizing a northbound interface to connect the application layer, which further enables various applications such as the monitoring function of the controller, topology management, path management, and collaborative management. SDN technology can solve some of the problems in a traditional network. These advances mean that it can be applied in wireless sensor networks to address energy efficiency, optimize network topology and improve network reliability in industrial environments.

C. Contributions of This Paper

An earlier study was undertaken that adopted SDN to support WSN in IIoT and proposed the software defined-wireless sensor network (improved SD-WSN) as a new architecture for WSN within the industry. The contributions of this paper can be summarized as follows.

1) We adopt a framework that has high scalability by separating data plane from control plane. This means that the nodes only need to have a forwarding function.

2) Based on the presented framework, we consider the reliability of the network, propose a virtualization process for the sensor nodes and links, design an improved shortest path algorithm, and solve the network coverage problem. We analyze the energy problem of WSN nodes, reduce the energy consumption of nodes in the data transmission process, and solve the node failure problem using a load balancing method.

3) We perform a simulation that is close to a real environment, and then compare improved SD-WSN with traditional WSN and SD-WSN. Experimental results show that the SDN controller can dynamically control the node address, the energy consumption of each node is significantly smaller than traditional ones, and the reliability of the network is improved.

The rest of this paper is structured as follows. We first discuss related works in Section II, and then we describe an improved SD-WSN framework that is well-suited to IIoT in Section III. We present corresponding schemes and algorithm in Section IV. We conduct extensive experiments and report the results in Section V. Some issues are discussed in Section VI and future work is considered. Finally, the paper is concluded with a brief summary of contributions.

II. Related Work

Currently, most applications and services in WSN, such as network flexibility, network management, energy conservation, network robustness, and packet loss probability, rely on data exchange between the sensor nodes and the switch infrastructure or among sensor nodes.

To improve network flexibility, Luo *et al.* [10] proposed a software-defined WSN architecture to solve the rigidity of policies and the difficulty of management by addressing Sensor OpenFlow. Jacobsson and Orfanidis [11] adopted low-cost off-the-shelf hardware and an individual deployment method to flexibly reconfigure WSN networking and in-network processing functionality. Figueiredo *et al.* [12] used multiple adaptive hybrid approaches, which are considered better than a single algorithm. This work discusses the usage of policies to establish adaptive routing rules for WSN elements to acquire more flexible and accessible development and maintenance tasks of a network.

About the network management, Gante *et al.* [13] presented a smart management in WSN by employing the OpenFlow protocol and modifying the forwarding rule in the controller, which can use the unified standard to communicate with all nodes. A networking solution SDN-WISE was proposed by Galluccio *et al.* [14] to simplify the management of the network. Olivier *et al.* [15] proposed a cluster based architecture with multiple base stations used as hosts for the SDN control functions as well as cluster heads, and adopted a structured and hierarchical management. Modieginyane *et al.* [16] summarized the application challenges faced by WSNs for monitored environments and the opportunities that can be realized in applications of WSNs using SDN.

According to previous reports about energy conservation, Arumugam *et al.* [17] designed an energy-efficient LEACH protocol for gathering data. This protocol offered energy-efficient routing in WSN based on an effective data ensemble and optimal clustering. Wang *et al.* [18] presented a sleep scheduling algorithm (SDN-ECCKN) that completed every computation within the controller, so there was no broadcasting required between any two nodes. Experiments showed that this method reduced the total transmission time over the network lifetime. Jayashree *et al.* [19] proposed a framework for a software-defined wireless sensor network, where the sensor node only performed the forwarding. The controller was implemented as the base station in this framework, thus reducing energy consumption. Ejaz *et al.* [20] designed an energy efficient SD-WSN to minimize the energy consumption of energy transmitters. Xiang *et al.* [21] studied an energy-efficient routing algorithm for SD-WSN by selecting control nodes and assigning different tasks dynamically through control nodes.

For packet loss probability and robustness, Levendovszky *et al.* [22] provided the Hopfield neural network to guarantee uniform packet loss probabilities for the nodes. Sachenko *et al.*

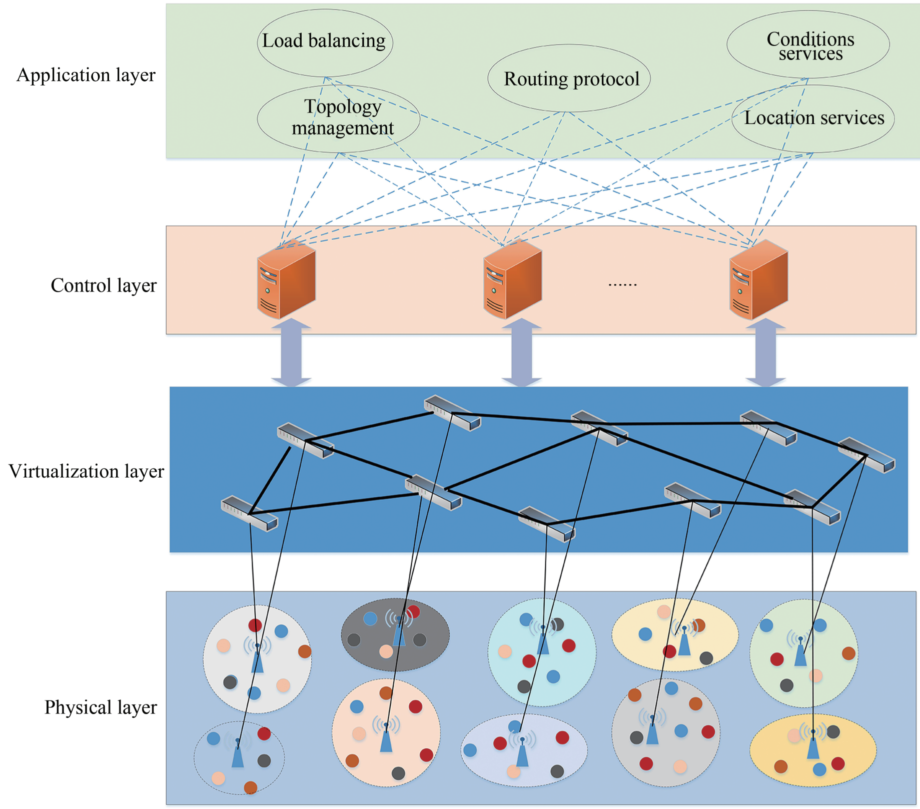


Fig. 1. Four-tier SD-WSN framework.

[23] presented modified correction codes based on a residual number system, high correction characteristics and a simplified coding procedure to show the characteristics of these codes, which can improve the data transmission robustness in WSN. Jin *et al.* [24] dealt with the packet loss by designing a hierarchical data transmission framework of suitable industrial environment.

For a real industrial environment, the commonly-used WSN framework discussed above is not well suited to the environment's unique characteristics, such as a harsh application environment, a strict requirement for data transmission, data diversity and so on. The traditional SD-WSN framework is a universal network framework that is not specific to IIoT. In order to resolve these problems and adapt the framework for an industrial environment, we will present an improved SD-WSN framework in the next section.

III. Network Framework: Software-defined WSN

In this section, we propose an improved software-defined WSN (improved SD-WSN) framework. The basic SDN model and its features are described.

A. Improved Software-defined WSN

This method mainly implements the SDN to achieve network virtualization in a sensor network. Therefore, a one-to-one mapping of key physical and virtual nodes was established using the flowvisor method, combined with optimization of the shortest path algorithm for nodes in the virtual layer. The control layer is separated from the forwarding layer in this project, which combines the above methods and related schemes to solve the problem of sensor network load balancing and node energy consumption, and improves the reliability of the network against node failure. The improved SD-WSN has a four-layer framework. From bottom to top, it has physical, virtual, control and application layers successively, as shown in Fig. 1.

1) Physical Layer: The physical layer includes different sensor nodes, sink nodes, switches, and servers (in contrast with an SDN controller) according to the industrial environment. The sensor nodes are responsible for the collection of messages in the area and transferring this data to the management system. These nodes connect with the SDN controller through the sensor OpenFlow, and receive relevant flow table information from the regional WSN through the SDN controller. The sink nodes that are within reach along the path offer the capability of data transmission to sensor nodes. The switches mainly connect the sink nodes, other equipment and the servers. The servers provide an information service to the sensor nodes, including information such as the condition service, based on requests from the sensor nodes. One node can be connected to multiple servers at the same time.

2) Virtualization Layer: The key nodes of the sensor network in the physical layer map to virtual key nodes, in order to form the virtual node layer (virtual layer) and achieve resource sharing and multi-network cooperation [25]. On the basis of the flowvisor SDN virtualization program, we can unify the scheduling and management of virtual key nodes, optimize the bandwidth of the network and perform CPU utilization and traffic management.

3) Control Layer: In the control layer, the controller connects to every switch and acts as a mediator between the visualization layer and the application layer through the SDN controller. The SDN controller provides routing, security and other services to achieve intelligent management of the network. In this framework, the control layer is responsible for managing the network resources of the virtualization layer, which is the key of node scheduling and routing. By collecting the node information, the network topology can be generated. The OpenFlow protocol can then be used to transmit the routing information for the nodes in the flow table entries. The global network view ensures that the controller can provide a customized network state to applications, which is a function that is not available in the traditional network. In this way, different applications obtain different views, and this benefits both the applications and the network.

4) Application Layer: The strategy of each application is defined in the application layer. The applications provide services for the sensor node in the WSN, such as topology management, location services, condition services, load balancing and the routing protocol. Each application obtains customized network states from the control layer and makes decisions according to its strategy. This strategy instructs how services will be provided to sensor nodes, in order to achieve unified node scheduling for the controller. The strategy of application layer mainly relates to load balancing and node energy consumption in the framework.

B. Pros and Cons of Traditional WSN, SD-WSN and Improved SD-WSN

Table I summarizes the benefits of improved SD-WSN over SD-WSN and traditional WSN technologies. Firstly, in a traditional WSN, the sensor nodes can have many different types of structures including normal nodes, routers and coordinators for node function types. Additionally, the node application types are the same as the function types. In SD-WSN, there is only a small difference between the node structures. In improved SD-WSN, the network structure is smaller than traditional structures, and the sensor nodes are the same as controllers for the virtualization layer.

TABLE I
PROS AND CONS OF CURRENT WSN TECHNOLOGIES AND IMPROVED SD-WSN

Traditional WSN	SD-WSN	Improved SD-WSN
Con: broadcasting messages periodically	Pro: service response	Pro: more flexible
Con: the variety of node structures	Pro: difference of node structures is quite small	Pro: network structure is smaller
Con: data transfer from all directions	Pro: finding the path according to the direction of nodes	Pro: more adaptive for the IIoT
Pro: Do not need a controller	Con: Need a controller	Con: Need a controller

Secondly, in order to send message in a traditional WSN, the messages need to be broadcast periodically to establish a data transfer path [26]. SD-WSN leverages the benefits of OpenFlow and uses switches to forward packets containing flow table information for the controller, so as to install these packets, and this process is a service response.

Finally, in SD-WSN, the controller application can compute the most likely path that a node would reasonably choose to connect nodes and devices, with the support of an up-to-date, global view of the network topology and traffic states. The improved SD-WSN can more effectively solve network coverage problems due to the complex industrial environment.

IV. IMPROVED SD-WSN WORKFLOW

In the previous section the improved SD-WSN components were introduced and in this section we describe how improved SD-WSN is used in a typical IIoT scenario, which is shown in Fig. 2. The overall workflow includes initialization, network virtualization, topology discovery, administration (create and send policy), routing discovery and node information feedback.

A. Initialization

1) The initial SDN controller and sensor node are obtained, and they communicate by OpenFlow.

2) The router nodes of the network are configured, and connected to the controller SDN to report their address, energy, computing power and other related information.

3) The router connects with an adjacent sensor node to obtain information related to energy, load, bandwidth and location. It informs the sensor node about its routing address, generating topological information within the region.

4) The router collects information within the region and transfers this information to the SDN controller.

B. Network Virtualization

1) The overall resource view is obtained according to the initial state of the network. Using the controller SDN to build the SD-WSN virtual layer (SDN flowvisor), the SDN flowvisor manages the physical network through the sensor OpenFlow protocol.

2) As shown in Fig. 1, the router of the physical layer is mapped to the virtual layer, and the router communicates with SDN flowvisor using the sensor OpenFlow protocol. A number of SDN controllers are deployed according to the WSN application service or the type of sensor node (such as temperature, humidity, pressure or special detection sensor nodes). The different WSN subnets are mapped to different SDN controllers and each SDN controller manages a service type of the WSN subnet. For each of the different application services, the SDN flowvisor [26] can reasonably control and allocate transmission traffic and bandwidth etc.

3) The virtual nodes are not physical nodes, but a physical network in a region. The virtual nodes can provide processing capacity of a data packet service for the application layer. Therefore, a direct relationship does not exist between the virtual link and the physical link.

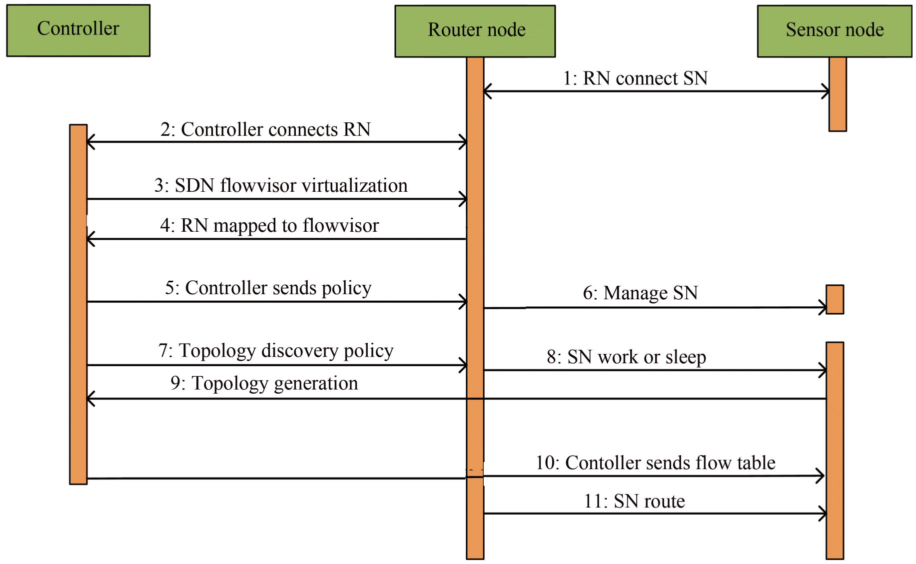


Fig. 2. Sequence diagram of improved SD-WSN within an IIoT scenario.

C. Administration

1) Instructions Issuance and Allocation Strategy: The controller sends a message to install the control command for nodes. This control message contains the configuration commands and parameters that are required for installation of the current physical nodes. For example, the routing service obtains the service path using a discovery algorithm for the service path, and then configures a strategy for each node in the service path.

2) Management of SDN Controller: Depending on the region and the number of routers, the SDN controller can add or reduce the number of clusters to flexibly extend the SDN controller and adopt a one-button deployment of this controller. The control layer uses management of the virtual machine cluster method, according to scale, distribution, and the number of virtual layer nodes of the sensor network to set the number of virtual controllers. The virtual controller sets the thresholds for the CPU and memory according to the regional network conditions. If this threshold is exceeded, the controller will dynamically adjust the controller performance.

3) Node Scheduling Sleep Strategy: Based on the management of SDN controller, we refer to an intelligent nodes work/sleep strategy [27]. Depending on the SDN controllers' information collected from several nodes, energy information of each sensor node in the region is monitored to dynamically adjust the nodes sleep table, and then put the nodes to sleep or work periodically.

4) Centralized Control Rules Predefining Strategy: This strategy is based on a node collection, routing and energy consumption optimization method and it uses a cloud management platform to realize centralized management and schedule different WSN equipment. Both the sensor node (SN) data collection and the active nodes (AN) are used to perform the task.

D. Topology Discovery

The SDN controller is connected to the virtual layer through the sensor OpenFlow protocol to transfer the control information to the virtual layer, in order for the controller to obtain a resource view of the virtual layer network. The SDN controller rapidly acquires sub network resources by SDN flowvisor. Due to frequent changes in the network topology, the ECCKN algorithm [28], [29] has been improved and an algorithm for energy consumption based on K-neighborhood degree (ECKD) has been adopted. The method is based on centralized control by the controller to proactively update and achieve a sleeping state of nodes (as shown in Algorithm 1).

Algorithm 1. Energy consumption based on K-neighborhood degree (ECKD)

1: The SDN controller interacts with the nodes and all node information is collected in the sub-network. The node set is N, the energy of the nodes is E, the degree of the nodes is $K = (k_1 \cdots k_n)$ and each node corresponds to a neighboring node set M.
2: Assumption: if the residual energy of the current node is E_i and the node's neighbor is M_n, then the neighbor of the neighboring node is M_v.
3: $k = \min(K)$.
4: **if** $|M_u| < k$ or $|M_v| < k$ for any $s_v \in N$ **then**
5:     Remain awake;
6:     Return
7: **end if**
8: $E_u = \{s_i | s_i \in M_n, E_i > (E_N/k_i) * k\}$.
9: Go to sleep if both of the following conditions hold. Otherwise remain awake.
10: Any two nodes in E_u are either directly connected themselves or indirectly connected through nodes within s_u's 2-hop neighborhood that have E_u less than E_i.
11: Any node in M_u has at least k neighbors from E_u.
12: Return.

An example is shown in Fig. 3. Firstly, the SDN controller obtains the residual energy of all nodes and their neighbors in the network. According to the requirements of Algorithm 1, the results are: $EA = 5(C)$, $EB = 6(D)$, $EC = 6(D), 5(F), 5(E)$, $ED = 5(C), 5(E), 8(G)$, $EE = 5(C), 6(D), 8(G), 5(F)$, $EF = 5(C), 5(E), 8(G)$, $EG = 6(D), 5(E), 5(F)$. According to the algorithm, when the number of energy sets in the node is less than 2, A, B keep working in the next step. Since $EC, ED, EE, EF, EG > 2$, it is necessary to judge Step 9 and select two nodes to communicate directly or indirectly with five nodes in EE. Every node of set NE has at least two neighboring nodes in the set EE. Otherwise, in EC, ED, EE, EF, EG, only EE satisfies all conditions in Step 9 of the algorithm, so EE goes to sleep and the other nodes remain awake for the next cycle.

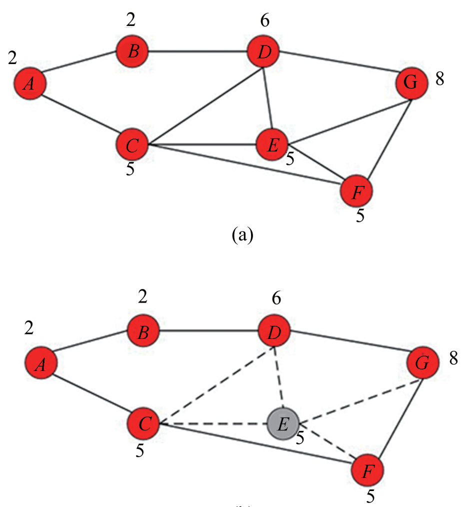


Fig. 3. (a) The original network topology. (b) improved network topology by ECKD.

E. Routing Discovery

We propose a routing algorithm based on depth traversal (RABDT) that considers two factors: energy consumption and node loading. The algorithm can reduce the loading of WSN nodes, prolong the life of nodes, and enhance the reliability of the network (as shown in Algorithm 2).

Algorithm 2. Routing algorithm based on depth traversal (RABDT)

1: In the ECKD algorithm, the sub network topology is obtained based on the energy-saving concept for the nodes, which is expressed by the adjacency matrix $G = (V, E)$.
2: According to the SDN controllers' interaction with the nodes, the controllers collect the CPU loading values in the current working nodes, and then use this to set L to represent the CPU loading values, and set a loading threshold δ.
3: The SDN Controller calculates the route from node s to node d, and traverses neighboring nodes based on the adjacency matrix (V, E).
4: When traversing the neighboring nodes,
5: **if** its IP is equal to IP_d **then**
6: Return
7: **end if**
8: **if** none of the neighboring nodes have an IP that is equal to IP_d **then**
9: The neighboring node L_i is determined
10: **if** $L_i < \delta$ **then**
11: traverse to the neighbor of this node, go to Step 4;
12: **else if** all $L_i > \delta$ **then**
13: Find the loading of the smallest neighboring nodes, go to Step 4
14: **else**
15: Return
16: **end if**
17: **end if**
18: **end if**
19: The SDN controller gets the router from s to d, and then the related flow table is sent to the relevant nodes.

F. Node Information Feedback

Node information upload: The sensor nodes perform the configuration command, complete the service installation, collect the hot spot information in the area, send feedback messages to the SDN controller and observe the node operating conditions. If a key node fails or is asleep, go back to Section I-D.

V. Performance Evaluation and Analysis

A. Experimental Scenario

This experiment is based on SDN and the IEEE 802.15.4 protocol for wireless sensor networks. The network emulator for the framework is Mininet [30] and the controller is Floodlight [31]. Floodlight runs on a server with an AMD Opteron processor 6348 and 16 GB memory. The server is installed with Linux kernel version 2.6.32. Mininet runs on a separate server, and the servers are connected by a 10 Gbps Ethernet network.

The survey region is situated in a pharmaceutical company in a high-tech park and includes an administrative area, a production area and a warehouse and logistics area. At the initialization stage, there are many types of networks found in each region. The two regions of the production workshop and warehouse are chosen. The network deployment is illustrated in Fig. 4. The networks are then added according to the requirements of the extension and the SDN is installed with flow rules according to the improved SD-WSN flow.

B. Theoretical Analysis

1) Reliability Analysis: Software defined wireless sensor network reduces the differences between hardware due to its separation of the forwarding and controlling, and enhances the capacity of network resource scheduling. For instance, the controllers can acquire the whole network nodes information; the topology discovery algorithm can get subnet information more rapidly, which makes the distribution of working nodes more uniform, so it can fully use the overall network nodes resources.

Due to the customization and programming of SDN, the capacity of the network application and the utilization ratio of network resource have been improved, thus improving the

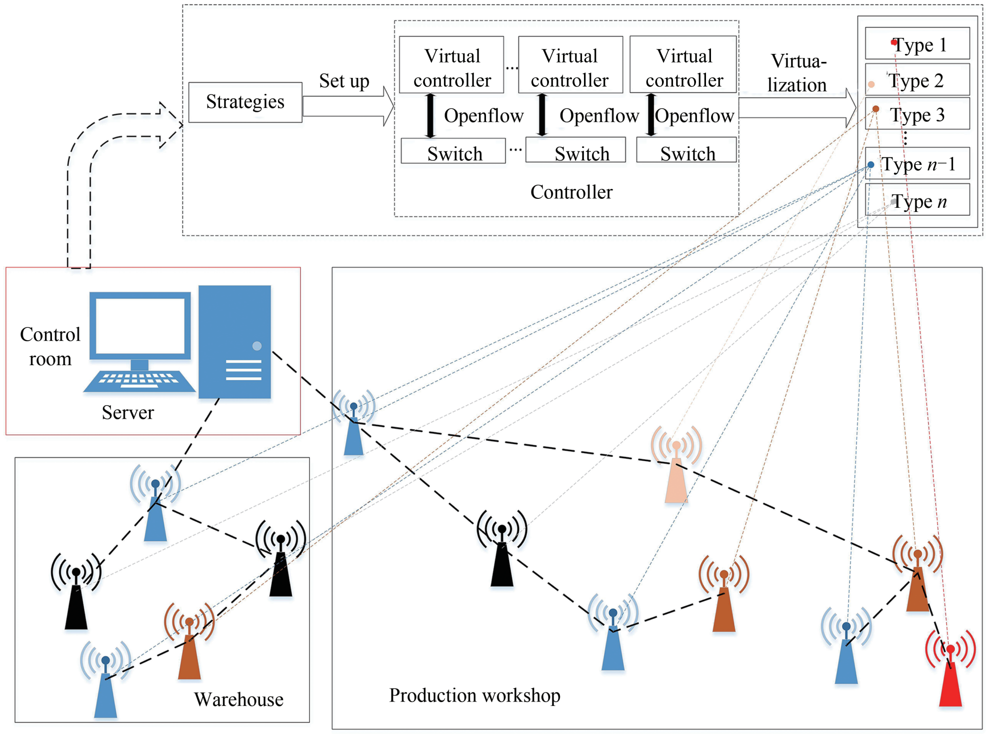


Fig. 4. Heterogeneous networks and multi-regions based on a wireless sensor network.

reliability of network. For instance, the ECKD algorithm effectively reduces the energy consumption and the RABDT algorithm has improved the capacity of data translation. From the following experiment results, it can be seen that the operational time of nodes enhances and the delay of data translation decreases.

The virtualization technology under SDN has reduced the sensitivity of network topology for the network reliability. Sensor network clustering under the virtualization technology has divided sensor network reasonably, and it has reduced the regional network load.

2) Energy Consumption Analysis: Without loss of generality, the following settings are used in the experiment.

a) The energy consumption due to the perceived environment has been ignored, and the number of the network nodes is 120.

b) Sink nodes are not considered, namely all nodes are considered to be of the same size, shape, structure and have the same initial energy.

c) The transmission radius and the receiving radius of each sensor node is the same.

d) The message frequency is once per minute. Therefore, the energy consumption can be calculated by (1) and (2), when transferring 1 bit data between the source node and the destination node [32]:

$$E_{Tx}(l,d) = lE_{\text{elec}} + ld^2\varepsilon_{\text{amp}} \quad (1)$$
$$E_{Rx}(l) = lE_{\text{elec}}. \quad (2)$$

Therefore, we can obtain:

$$E_t = E_{Tx}(l,d) + E_{Rx}(l), \quad t = 0, 1, 2, \ldots \quad (3)$$

$$M_t = \frac{MI - \sum_{t=0}^{n} E_t}{MI} \quad (4)$$

where l indicates the length of the sending or receiving data packets, E_{Tx} indicates the energy required for a node to transmit a 1-bit long packet over a distance d, E_{elec} denotes the electrical energy required for a node to send or receive 1-bit of data, ε_{amp} is the amplification extent of transmission, E_{Rx} denotes the energy consumption of receiving data, t is the elapsed time, E_t denotes the total energy consumption at time t, M is number of nodes, and I and M_t denote the initial energy and the proportion of remaining energy at time t, respectively.

3) Delay Analysis: The delay time includes the processing delay (p), the serialization delay (s) and the transmission delay (d). The calculation formula is proposed as below:

$$D = p_t + s_t + d_t \quad (5)$$

where t denotes the time value. In order to better explain the delay time, the effect of the number of nodes and the time values were analyzed. Therefore, experiments were performed with 60, 120 and 180 nodes for the three networks, respectively, with values of t ranging from 0 s to 60 s.

C. Performance Evaluation

Experiment of energy consumption and delay time has been designed in this section. Table II shows the simulation environment and network parameters.

TABLE II
SET OF NETWORK PARAMETERS

Network parameters	Values
Scale of network nodes (N)	$60, 120, 180$
Network area	$100 \times 100\,\mathrm{m}^2$
Radius of node communication (R)	$20\,\mathrm{m}$
Initial energy of node (I)	$10\,\mathrm{J}$
Electrical energy (E_{elec})	$60 \times 10{-}6\,\mathrm{J/bit}$
Size of packet	$60 \times 10{-}6\,\mathrm{J/bit}$
Loss of power amplifier ($\varepsilon_{\mathrm{amp}}$)	$10 \times 10{-}9\,\mathrm{J/(bit \times m^2)}$

1) Energy Consumption Evaluation: Fig. 5 shows the energy comparison among the improved SD-WSN, the traditional SDN-WSN and the WSN environment. The improved SD-WSN is superior to the other solutions in terms of remaining energy. Two conclusions can be drawn from the results. First, the proposed sleep scheduling mechanism and the path finding method can save a lot of energy. Second, SDN centralized management leads to easier management of the network energy.

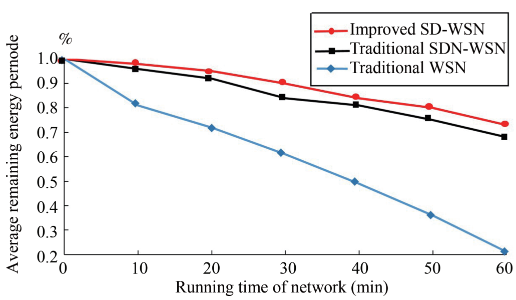
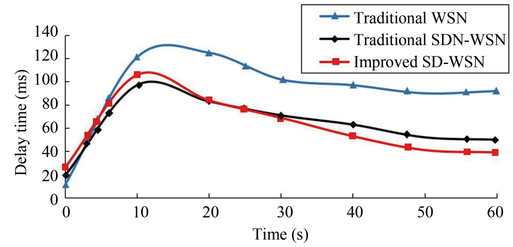
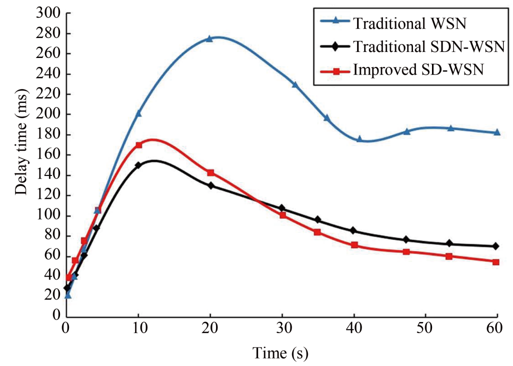


Fig. 5. The average remaining energy in the three environments.

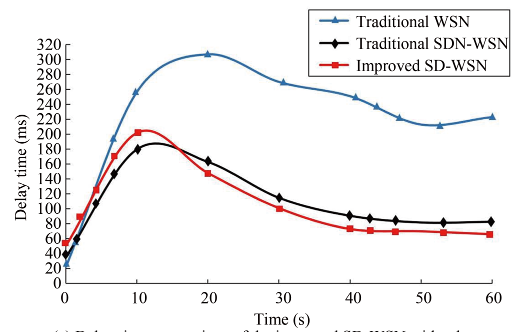
2) Translation Delay Evaluation: Fig. 6 shows the delay time results of the proposed improved SD-WSN and other environments. The reason why the improved SD-WSN results are better than traditional WSN and SDN-WSN is described in Section I. The proposed improved SD-WSN method spends some time for calculating the path from the source node to the destination node by the RABDT algorithm. Once the path is established, the delay time is rapidly reduced, so the total delay time is less than traditional WSN and SDN-WSN.



(a) Delay time comparison of the improved SD-WSN with other methods for a 60 node network



(b) Delay time comparison of the improved SD-WSN with other methods for a 120 node network



(c) Delay time comparison of the improved SD-WSN with other methods for a 180 node network

Fig. 6. (a), (b), (c) respectively represent the delay time comparisons of improved SD-WSN with other methods for 60, 120, 180 nodes of a complete network.

VI. CONCLUSION

This paper has proposed an improved SD-WSN to address the issue of the proprietary and closed nature of WSN design in the industrial environment. This improved SD-WSN has been shown to be efficient due to the characteristics of IIoT. Firstly, this technology can reduce network coupling by separately controlling the functions from the devices, using an external controller to solely issue commands. A workflow has been designed for an IIoT scenario that reduces the energy and loading of some of the nodes in the WSN, and improves the reliability of the network by improving the ECKD algorithm and proposing some suitable schemes. Network personalized application services facilitated by virtualization have achieved resource sharing and flexible extensions of the SDN controller. In the future, we intend to make further development in three aspects: 1) expanding the topology discovery and routing algorithm optimization to the real-time situation in large factory, in order to be more suitable for SD-WSN in IIoT; 2) intelligent scheduling solution for mobile nodes and other nodes of WSN in IIoT [33]; 3) in the case of the same number of sensor nodes and considering the flexibility, researches about the network throughput under the improved SD-WSN framework will be presented in the next paper.

REFERENCES

[1] S. Distefano, "Evaluating reliability of WSN with sleep/wake-up interfering nodes," *Int. J. Syst. Sci.*, vol. 44, no. 10, pp. 1793−1806, Oct. 2013.

[2] Seeberger Company [Online]. Available: http://www.seeberger.de/. Nov. 16, 2017.

[3] W. F. Li and X. W. Fu, "Survey on invulnerability of wireless sensor networks," *Chin. J. Comput.*, vol. 38, no. 3, pp. 625−647, Mar. 2015.

[4] J. Huang, Y. Meng, X. H. Gong, Y. B. Liu, and Q. Duan, "A novel deployment scheme for green internet of things," *IEEE Int. Things J.*, vol. 1, no. 2, pp. 196−205, Apr. 2014.

[5] Q. Y. Zuo, M. Chen, G. S. Zhao, C. Y. Xing, G. M. Zhang, and P. C. Jiang, "Research on OpenFlow-based SDN technologies," *J. Sof.*, vol. 24, no. 5, pp. 1078−1097, May 2013.

[6] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: a comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14−76, Jan. 2015.

[7] Open Networking Foundation, "Software-defined networking: the new norm for networks," *ONF White Paper*, Apr. 2012.

[8] N. McKeown, "Software-defined networking," *INFOCOM Keynote Speech*, vol. 17, no. 2, 2009, pp. 30−32.

[9] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Commun. Mag.*, vol. 51, no. 2, pp. 114−119, Feb. 2013.

[10] T. Luo, H. P. Tan, and T. Q. S. Quek, "Sensor openflow: enabling software-defined wireless sensor networks," *IEEE Commun. Lett.*, vol. 16, no. 11, pp. 1896−1899, Nov. 2012.

[11] M. Jacobsson and C. Orfanidis, "Using software-defined networking principles for wireless sensor networks," in *Proc. the 11th Swedish Nat. Computer Networking Workshop (SNCNW)*, Karlstad, Sweden, 2015.

[12] C. M. S. Figueiredo, A. L. dos Santos, A. A. F. Loureiro, and J. M. Nogueira, "Policy-based adaptive routing in autonomous wsns," in *Proc. the 16th IFIP/IEEE Ambient Networks Int. Conf. Distributed Systems: Operations and Management*, Barcelona, Spain, 2005, pp. 206−219.

[13] A. De Gante, M. Aslan, and A. Matrawy, "Smart wireless sensor network management based on software-defined networking," in *Proc. 2014 the 27th Biennial Symp. Communications*, Kingston, ON, Canada, 2014, pp. 71−75.

[14] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "Reprogramming wireless sensor networks by using SDN-wise: a hands-on demo," in *Proc. 2015 IEEE Conf. Computer Communications Workshops*, Hong Kong, China, 2015, pp. 19−20.

[15] F. Olivier, G. Carlos, and N. Florent, "SDN based architecture for clustered WSN," in *Proc. 2015 the 9th Int. Conf. Innovative Mobile and Internet Services in Ubiquitous Computing*, Blumenau, Brazil, 2015, pp. 342−347.

[16] K. M. Modieginyane, B. B. Letswamotse, R. Malekian, and A. M. Abu-Mahfouz, "Software defined wireless sensor networks application opportunities for efficient network management: a survey," *Comput. Electr. Eng.*, Mar. 2017.

[17] G. S. Arumugam and T. Ponnuchamy, "EE-LEACH: development of energy-efficient leach protocol for data gathering in WSN," *EURASIP J. Wireless Commun. Network.*, vol. 2015, Article ID 76, Dec. 2015.

[18] Y. W. Wang, H. N. Chen, X. L. Wu, and L. Shu, "An energy-efficient SDN based sleep scheduling algorithm for wsns," *J. Netw. Comput. Appl.*, vol. 59, pp. 39−45, Jan. 2016.

[19] P. Jayashree and F. I. Princy, "Leveraging SDN to conserve energy in WSN-AN analysis," in *Proc. 2015 the 3rd Int. Conf. Signal Processing, Communication and Networking (ICSCN), 2015 3rd International Conference on*, Chennai, India, 2015, pp. 1−6

[20] W. Ejaz, M. Naeem, M. Basharat, A. Anpalagan, and S. Kandeepan, "Efficient wireless power transfer in software-defined wireless sensor networks," *IEEE Sens. J.*, vol. 16, no. 20, pp. 7409−7420, Jul. 2016.

[21] W. Xiang, N. Wang, and Y. Zhou, "An energy-efficient routing algorithm for software-defined wireless sensor networks," *IEEE Sens. J.* vol. 16, no. 20, pp. 7393−7400, Oct. 2016.

[22] J. Levendovszky, K. Tornai, G. Treplan, and A. Olah, "Novel load balancing algorithms ensuring uniform packet loss probabilities for WSN," in *Proc. 2011 IEEE the 73rd Vehicular Technology Conference (VTC Spring)*. Yokohama, Japan, 2011, pp. 1−5.

[23] A. Sachenko, Z. B. Hu, and V. Yatskiv, "Increasing the data transmission robustness in WSN using the modified error correction codes on residue number system," *Elektron. Elektrotechn.*, vol. 21, no. 1, pp. 76−81, Sep. 2015.

[24] X. Jin, F. X. Kong, L. H. Kong, H. H, Wang, C. Q. Xia, P. Zeng, and Q. X. Deng, "A hierarchical data transmission framework for industrial wireless sensor and actuator networks," *IEEE Trans. Ind. Inf.*, vol. PP, no. 99, pp. 1, May 2017.

[25] M. M. Islam, M. M. Hassan, G. W. Lee, and E. N. Huh, "A survey on virtualization of wireless sensor networks," *Sensors*, vol. 12, no. 2, pp. 2175−2207, Feb. 2012.

[26] R. Sherwood, G. Gibb, K. K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. Parulkar, "FlowVisor: a network virtualization layer," OpenFlow Switch Consortium, Tech. Rep, OPENFLOW-TR-2009-1, Oct. 2009.

[27] B. J. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wirel. Netw.*, vol. 8, no. 5, pp. 481−494, Sep. 2002.

[28] S. Kumar, T. H. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *Proc. the 10th Ann. Int. Conf. Mobile Computing and Networking*, Philadelphia, PA, USA, 2004, pp. 144−158.

[29] L. Wang, Z. X. Yuan, L. Shu, L. Shi, and Z. Q. Qin, "An energy-efficient CKN algorithm for duty-cycled wireless sensor networks," *Int. J. Dis. Sens. Netw.*, Jan. 2012. doi: 10.1155/2012/106439.

[30] "Floodlight," [Online]. Available: http://www.projectfloodlight.org/floodlight/. Nov. 16, 2017.

[31] "Mininet," [Online]. Available: http://mininet.org/. Nov. 16, 2017.

[32] Z. X. Yuan, L. Wang, L. Shu, T. Hara, and Z. Q. Qin, "A balanced energy consumption sleep scheduling algorithm in wireless sensor networks," in *Proc. 2011 the 7th Int. Wireless Communications and Mobile Computing Conf. (IWCMC)*, Istanbul, Turkey, 2011, pp. 831−835.

[33] X. W. Fu, W. F. Li, G. Fortino, P. Pace, G. Aloi, and W. Russo, "A utility-oriented routing scheme for interest-driven community-based opportunistic networks," *J. Univ. Comput. Sci.*, vol. 20, no. 13, pp. 1829−1854, Jan. 2014.

Ying Duan is a Ph.D. candidate at Wuhan University of Technology under the supervision of Dr. Wenfeng Li. She received the M.S. degree in computer science and technology from Wuhan University of Technology, China. Her research interests include industrial internet of things, including the wireless sensor network, opportunistic networks, software defined network.

Wenfeng Li (M'01−SM'12) is a Full Professor at the School of Logistics Engineering, Wuhan University of Technology. He set up the Institute of Logistics and Robotics, the Center of IoT and Logistics Technologies, and the International Joint Laboratory of Internet of Things. He graduated in mechanical engineering at Zhengzhou Light Industry Institute, and received the master degree from Huazhong University of Science and Technology and the Ph.D. degree from Wuhan University of Technology. He is a Senior Member of both IEEE and China Society of Logistics, TC member of IEEE CSCWD, a committee member of Teaching Guiding Committee for Logistics Education of Ministry of Education of China, a member of Intelligent Manufacture Committee of China. He has contributed to IEEE as a Session Co-Chair of IEEE SMC 2009−2013, a Program Committee Co-Chair of the International Working Group on CSCW in Design (2012) and a Program Committee Co-Chair of the International Conference of ICMA 2010 and ICIA 2010. His research interests include internet of things, swarm robots, logistics and supply chain, and complex systems.

Xiuwen Fu is a Lecturer at the Institute of Logistics Science and Engineering, Shanghai Maritime University. He received the Ph.D. degree at Wuhan University of Technology. His research interests include wireless sensor networks, opportunistic networks, and industrial internet of things.

Yun Luo received the B.S. degree from the Department of Computer Science and Technology, Wuhan University of Technology in Wuhan, China. He is currently working towards the Ph.D. degree in the Department of Logistics Engineering, Wuhan University of Technology. His research interests include sensor networks, multi-mobile robots, and software defined network.

Lin Yang is a Ph.D. candidate at Wuhan University of Technology under the supervision of Dr. Wenfeng Li. He received the B.S. degree in computer science and technology from Navy University of Engineering, China. His research interests include people-centric internet of things, including the mobility model, connectivity and network protocols.