Received June 22, 2017, accepted July 11, 2017, date of publication July 14, 2017, date of current version August 8, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2727139

A Game-Theoretic and Energy-Efficient Algorithm in an Improved Software-Defined Wireless Sensor Network

LI PEIZHE, WU MUQING, LIAO WENXING, AND ZHAO MIN

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Li Peizhe (lipeizhe@bupt.edu.cn)

This work was supported by the Beijing Laboratory of Advanced Information Networks.

ABSTRACT Because wireless sensor networks (WSNs) are becoming increasingly integrated into daily life, solving the energy efficiency problem of such networks is an urgent problem. Many energy-efficient algorithms have been proposed to reduce energy consumption in traditional WSNs. The emergence of software-defined networks (SDNs) enables the transformation of WSNs. Some SDN-based WSNs architectures have been proposed and energy-efficient algorithms in SDN-based WSNs architectures have been studied. In this paper, we integrate an SDN into WSNs and an improved software-defined WSNs (SD-WSNs) architecture is presented. Based on the improved SD-WSNs architecture, we propose an energy-efficient algorithm. This energy-efficient algorithm is designed to match the SD-WSNs architecture, and is based on the residual energy and the transmission power, and the game theory is introduced to extend the network lifetime. Based on the SD-WSNs architecture and the energy-efficient algorithm, we provide a detailed introduction to the operating mechanism of the algorithm in the SD-WSNs. The simulation results show that our proposed algorithm performs better in terms of balancing energy consumption and extending the network lifetime compared with the typical energy-efficient algorithms in traditional WSNs.

INDEX TERMS SD-WSNs, OPGEA, WSNs, game theory, energy-efficient algorithm.

I. INTRODUCTION

Traditional wireless sensor networks (WSNs) operate as distributed networks. Because of their flexible deployment, convenient access and low cost, WSNs are widely used in many fields, and many sensor nodes are equipped with various types of sensors to provide services, including temperature monitoring, humidity monitoring and even audio and video monitoring, among others. However, in most WSNs, sensors and communication module are typically powered by batteries, and replacing the depleted batteries is cost prohibitive or even impossible. The energy constraints have caused formidable challenges for the network lifetime of the entire WSN system, which substantially limits the applications of WSNs. Therefore, designing an energy-efficient algorithm to control the transmission energy balance of all the nodes is of particular importance.

A. TRADITIONAL ENERGY-EFFICIENT ALGORITHMS

Many energy-efficient algorithms to improve the energy efficiency of WSNs have recently been presented. The authors in [1]–[3] provided surveys of energy-efficient algorithms applied in WSNs, and they provided surveys of the energy-efficient routing protocols for WSNs. The authors in [2] divided the energy-efficient algorithms into five categories: radio optimization, data reduction, sleep-wakeup schemes, charging solutions and energy-efficient routing. Each category covers a variety of newly proposed algorithms, which have greatly promoted the research on energy efficiency in WSNs.

B. THE INTRODUCTION OF GAME THEORY AND ITS INSUFFICIENCY

Among all the latest energy-efficient algorithm studies, most researchers focus on the routing and data transmission schemes. As the foundation of WSNs, energy-efficient routing has become a hot topic in energy-saving areas, and in recent years, game theory has emerged as a key tool for designing novel energy-efficient routing algorithms. Game theory provides a powerful tool for describing the phenomenon of competition and cooperation between intelligent, rational decision makers [4]. In the latest studies, many researchers have introduced a game-theoretic approach to energy conservation [5]–[11]. In [5], Feng *et al.* proposed an efficient and rapid convergence coalition formation algorithm to obtain the stable coalition partition in the game, a reliable coalition formation routing (RCFR) protocol was designed to enhance the packet delivery ratio, and the algorithm decreased the routing establishing time and balanced energy consumption. Shamshirband *et al.* in [6] introduced a game theoretic method, namely cooperative Game-based Fuzzy Q-learning (G-FQL), which implements cooperative defense counter-attack scenarios for the sink node and the base station to operate as rational decision-maker players through a game theory strategy. In [7], Farzaneh and Yaghmaee designed an evolutionary game theoretical resource control protocol (EGRC) for WSNs, which developed a non-cooperative game containing a large number of sensors as players for alleviating and controlling congestion in a WSN by utilizing the available resources and controlling the radio transmission power. In [8], Sun *et al.* proposed an energy-balanced unequal clustering routing protocol based on game theory named GBUC, attempting to achieve energy balance and improve network performance. A topology control and routing algorithm based on non-cooperative game theory in UWSNs was presented by Misra *et al.* in [9], which can optimize the network structure and prolong the network life cycle. In [10], Jamin *et al.* used game theory to propose an efficient routing scheme in a WSN that considers the signal-to-interference-plus-noise ratio (SINR) to determine the most optimal paths, and the WSNs in this paper were modeled as a dynamic, non-cooperative and incomplete information game with the static sensor nodes as the players in the network. In [11], Sathian *et al.* proposed a cluster head cooperative trustworthy energy-efficient MIMO (CH-C-TEEM) routing algorithm based on game theory, and the game theory in this algorithm is used to select healthier cluster heads that have sufficient residual energy and a high trust level and to select the cooperative nodes for MIMO communication.

However, all the above game-theoretic energy-efficient algorithms are based on distributed schemes, and the disadvantages are as follows. 1) The edge nodes have a great probability of failure in one routing discovery procedure, which makes the energy-efficient schemes invalid; 2) In distributed routing algorithms, more routing overhead is necessary to guarantee the success of finding a suitable path to the sink node, which can result in considerable energy waste; 3) In a distributed architecture, all the nodes need to have calculation ability, and the redundant energy consumption is high; Finally, 4) there is no flexibility to adjust the deployment in a game-theoretic algorithm without high energy loss in each node. To further reduce energy consumption and make game theory more suitable for WSNs, one feasible approach is to adopt centralized control algorithms. In fact, in most WSNs, the sensor nodes are stationary, and the connection relations between sensor nodes rarely change. Frequent routing discovery process in distributed routing algorithms becomes less necessary. Moreover, the sink node in WSNs is generally better equipped than the sensor nodes, the computing power and energy reserves of the sink node can be much stronger than other nodes. The above characteristics make it possible to apply a centralized control routing scheme, which is the main idea of integrate an SDN into WSNs. SDN decouples the control plane from the data plane, thus moving the control logic from the node to a central controller. A WSN is a great platform for low-rate wireless personal area networks with little resources and short communication ranges [12]. As the scale of WSNs expand, several challenges arise, such as network management and heterogeneous-node networks. The SDN approach to WSNs seeks to alleviate most of the challenges and ultimately foster efficiency and sustainability in WSNs.

C. SDN EMBEDDED AS A SOLUTION

Some SDN-based WSNs architectures have been proposed [13]–[16]. There are two major architectures: two-layer architecture [13]–[15] and three-layer architecture [16]. The feasibility and the advantages and disadvantages of SD-WSNs have been analyzed in detail. In [17], O'Shea *et al.* provided a survey of related works considering both SDN and centralized non-SDN approaches to network management and control, examined the challenges and opportunities for SD-WSNs, and provided an architectural proposal for an SD-WSNs. In [18], Jayashree and Princy developed a platform in which the data plane and the control plane are separated. By adding the SDN into WSNs, the sensor nodes perform only forwarding and do not make any routing decisions, thereby reducing energy usage. The authors in [12] presents a comprehensive review of the SD-WSNs literature. Challenges such as energy, communication, routing, security and configuration are summarized. Many SDN-based architectures [19]–[21] can be referenced. In [19] and [20], Li and Chen provided a survey of the software-defined network function virtualization (NFV) architecture, which benefits a wide range of applications (e.g., service chaining) and is becoming the dominant form of network function virtualization. The logic of packet forwarding is determined by the SDN controller and is implemented in the forwarding devices through forwarding tables. Efficient protocols can be utilized as standardized interfaces in communicating between the centralized controller and distributed forwarding devices. In [21], the state of the art on the application of SDN and NFV to internet of things (IoT) was investigated, some general SDN-NFV-enabled IoT architectures are reviewed, and the feasibility of applying SDN to WSNs is described. These literatures can be referred to in the realization of SD-WSNs.

In WSNs, many SDN-based energy-efficient algorithms have been proposed [22]–[24]. In [22], Zeng *et al.* focused on energy minimization in multi-task SD-WSNS. These authors investigated three issues, namely, sensor activation, task mapping and sensing scheduling by the centralized control algorithms. The three issues are jointly considered and formulated as a mixed-integer with quadratic constraints programming (MIQP) problem, which is then reformulated into a mixed-integer linear programming (MILP) formulation with low computation complexity via linearization. And an efficient online algorithm using local optimization is developed to deal with dynamic events. The proposed online algorithm approaches the globally optimized network energy efficiency with much lower rescheduling time and control overhead. Sleep scheduling [23] and load balance [24] algorithms have also been proposed to achieve energy efficiency in SD-WSNs. The authors in [23] proposed a SDN-based sleep scheduling algorithm (SDN-ECCKN) to manage the energy of the network, in this algorithm, every computation is completed in the controller rather than the sensor nodes and there is no broadcasting between each two nodes. In [24], the authors propose a multi-controller load balancing approach called HybridFlow in SD-WSNs, which adopts the method of distribution and centralization and designs a double threshold approach to evenly allocate the load. We found that the energy-efficient algorithms are easier to implement in SD-WSNs than in traditional WSNs. However, the aforementioned SDN-based energy-efficient algorithms are based on the ideal SD-WSNs and ignore the process of integrating SDN and WSNs. A new SD-WSNs architecture and a customized game-theoretic energy-efficient algorithm urgently need to be designed.

D. COGNITIVE RADIO FOR SECURE CHANNEL

There are several surveys in literature discussing cognitive radio approaches for WSNs. A survey on multichannel assignment protocols in WSNs is presented in [25], which is further extended in [26]. Authors in [27] present a review and categorization of existing Intrusion Detection and Prevention Systems (IDPS) schemes in terms of traditional artificial computational intelligence with a multi-agent support. Akyildiz in [28] overviewed the recent proposals for spectrum sharing and routing in wireless networks, spectrum sensing and spectrum sharing cooperate with each other to enhance spectrum efficiency.

The establishment of a secure channel in the SD-WSNs is a key issue, which can be propagated by the cognitive radio. An intelligent and distributed channel selection strategy for efficient data dissemination in multi-hop cognitive radio network has been proposed in [29], which can be used to choose the optimal channel as a secure channel. The channel bonding algorithm in [30] can utilizes the white spaces hence, furthermore, the bonding channel can minimizing the re-transmissions, which can meet the performance requirements of secure channel. The authors in [31] presented a novel energy-efficient block-based sharing scheme, which intensive security services and achieved energy-efficient, this idea can be used as a reference in a secure channel.

In this paper, we introduce a new SD-WSNs architecture as a novel and feasible solution to achieve the integration of SDN and WSNs. Based on this novel SD-WSNs architecture, a game-theoretic energy-efficient algorithm is designed, and the distributed and centralized control features of the novel SD-WSNs architecture are applied flexibly, which can improve the energy efficiency and increase the network lifetime.

The main contributions of this paper are as follows:

- A comprehensive analysis of the feasibility of the integration between SDN and WSNs and the defects in recent SD-WSNs are summarized;
- A novel SD-WSNs architecture is presented, which is capable of achieving the integration of SDN and WSNs;
- A game-theoretic and energy-efficient algorithm is realized, and the detailed steps for applying this innovative algorithm to SD-WSNs are described.

The remainder of this paper is organized as follows. In section II, the feasibility of applying SDN in WSNs is analyzed, and the novel SD-WSNs architecture is introduced. In section III, the design of our game-theoretic and energy-efficient algorithm is presented. Simulation scenarios and the results are analyzed in section IV. Finally, we conclude this paper in section V.

II. DESIGN OF THE INNOVATIVE SD-WSNs ARCHITECTURE

In this paper, the involved WSNs are considered to be specific-application-oriented communication networks. In the scenarios where the novel SD-WSNs are adopted, the sensor nodes should be stationary or have low mobility after being deployed. SDN was first designed to improve the management of wired networks, such as data centers [32]. The network is divided into a control plane and a data plane, and the programmable network switches in the data plane are simplified as forwarding devices. The controller in the control plane runs centralized control software to manage the entire networks [33], [34]. By simplifying the function of network switches, the network is easier to manage, and the efficiency of network switches is improved. The core idea of an SDN is feasible for applying in WSNs because of the following:

1. The sink node and sensor nodes in WSNs can correspond to the controller and switches in an SDN. The network structure can be divided into two layers, as shown in Figure 1. In an SDN, the controller runs the centralized control software to manage the entire networks, whereas the switches simply forward packets according to their flow tables. In WSNs, the sink node is responsible for collecting sensing data from all the sensor nodes, whereas the sensor nodes simply collect the environment data and send them to the sink node, additional, the sensor nodes in WSNs have to bear the work of routing discovery. The structural similarity makes it possible to integrate WSNs and SDN.
2. The controller in an SDN has a very strong computing capability, and the functions of the network switches are greatly simplified. This feature is similar in WSNs because the sink node is always better equipped with adequate energy, whereas all the sensor nodes have limited energy and calculation resources. Moreover, the

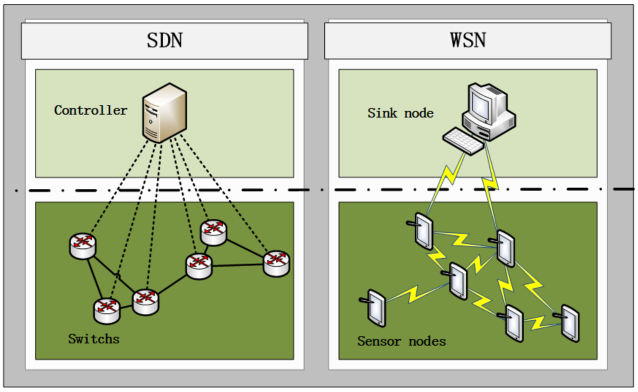


FIGURE 1. Network structures of SDN and WSN.

assumption that the sink node in WSNs has a strong computing capability and sufficient energy makes centralized control possible.

3. In an SDN, the unification of the network equipment can easily be achieved, which makes the network management convenient, and all the switches can be managed by a controller. In WSNs, the sink node typically has to manage hundreds of sensor nodes; however, different types of sensor nodes should have the ability to forward messages, which means that the centralized control mechanism in an SDN can provide the same convenience in the entire network control for WSNs if the sink node can function as a controller.

The SDN approach to WSNs is envisaged to potentially solve most of the inherent WSN challenges [13], [35], [36]. In recent years, many innovative SD-WSN architectures have been proposed. Huang *et al.* [37] propose a cognitive SD-WSN framework to improve energy efficiency and adaptability of WSNs for environmental monitoring. Olivier *et al.* [15] propose a cluster-based SD-WSN architecture, the cluster head referred herein as SDN cluster head (SDNCH), controls and coordinate all sensor nodes in its domain. Zeng *et al.* [38] propose an architecture model that combines SD-WSN and cloud computing. Authors in [35] and [39] indicated that the SDN approach can simplifies the management in WSNs considerably with its simplicity and ability to evolve. Costanzo *et al.* [39] state that SDN in WSN should support common energy conscious measures as currently being explored in traditional WSNs. The SDN model also enables flexible configuration moving away from the cumbersome and error prone manual process currently in place [40]. Mobility and localization are critical for better routing in wireless sensor networks, and SDN can simplifies this by managing the mobility from the central controller i.e. routing decisions and policies are managed at the controller [35]. However, the particularity of WSNs brings some hidden problems which are ignored in the current designed SD-WSNs. Problem 1: control problem Sensor nodes are randomly distributed in the monitoring area. At the initial phase of the network, the sink node or the controller has no idea where the sensor nodes are. The sensor nodes also don't know their neighbor relations as well as the path to the sink node or the controller. Under these circumstances, the control messages cannot easily reach the sensor nodes because of the unable establishment of TCP/IP connectivity. This problem is addressed in [13] but no appropriate solutions are provided. Problem 2: secure channel problem Most proposed SD-WSNs architectures ignore the establishment of secure channel [13], [36], [15]. The secure channel can be established by traditional routing algorithms [35]. However, the process of finding the secure channel is not inevitably successful. The secure channel consists of multiple wireless links which are not stable and has to be hosted in band [13]. Control messages have to share the same multi-hop path as well as the packet loss rate with the sensor data packets. As a result, the secure channel may not be secure enough. Problem 3: topology problem In WSNs, the network topologies change dynamically. The change of topology means the redefine of flow rules and the rebuilt of secure channel which bring a lot of additional traffic overhead. Authors in [35] apply the traditional routing protocols to deal with the node mobility. However, this approach is against the principle of centralized control. The additional routing overhead and the probability of routing failure make it inefficient or unacceptable in WSNs. In fact, whether it is feasible to apply centralized control mechanism in WSNs with too-frequently-change topologies is still not clear. Problem 4: flow problem WSNs are designed for specific applications and the types of services are limited and known. Ignoring the particularity of WSNs and directly applying the flow mechanism in OpenFlow for wired networks in WSNs [13], [36], [15] is not reasonable. There is no need for the sensor nodes to request the flow table frequently and in consideration of the limited resources of sensor nodes, the flow table should be as simple as possible. These problems urgently need to be solved in the new designed SD-WSNs. It is not necessary for the SDN mechanism in wired networks to deal with these problems however. Therefore, it is not suitable to directly apply wired SDN mechanism in WSNs. In full consideration of the particularity of WSNs, some appropriate modifications in current specification of wired SDN are indispensable.

The SD-WSNs should be able to inherit the advantages of the SDN. Moreover, taking the particularity of WSNs into account, our designed WSN-fit-in SD-WSNs architecture can be described as shown in Figure 2. The infrastructure of the SD-WSNs consists of a controller and many sensor nodes. The sink nodes in WSNs are replaced by a controller, which has a stronger calculation ability and sufficient energy, making it possible to achieve network topology analysis, flow table generation, network management and adjustment in a very short time interval. The sensor nodes in the data plane simply act as switches, and the processing capacity of the sensor nodes can be very low, which can substantially reduce their energy consumption. The data interaction in the SD-WSNs can be intuitively described as shown in Figure 2. The secure channel should have the lowest latency and highest reliability, making it suitable to transmit control messages

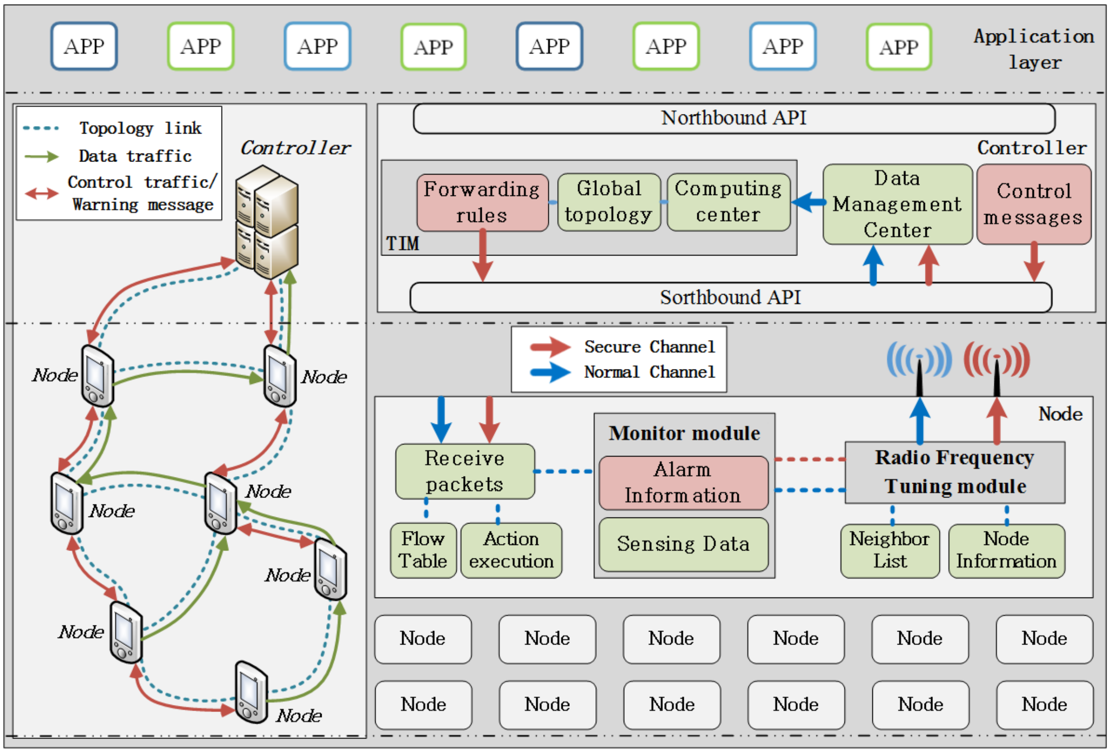


FIGURE 2. **Architecture of the innovative SD-WSNs.**

and forwarding rules from the controller to sensor nodes, the warning information and topology change messages from the sensor nodes to the controller should also occupy the secure channel. The secure channel is determined in the topology information module (TIM). The common channel is determined in the TIM according to the predefined policies, too, which is used to transmit massive sensing data and neighbor list (NL) from sensor nodes to the controller. However, to construct this SD-WSNs architecture, some basic **principles** need to be obeyed:

1) *Prior to the start of the SDN mode, a topology discovery is necessary. This process can help us obtain the position information and the maximum NL of all the nodes.*
2) *The designed SD-WSNs architecture fits in with the low dynamic WSNs because the topology of the entire network should not change frequently.*
3) *The secure channel should be used only for control messages, flow table and warning information, and the links in the security channel should be occupied by the common channel as little as possible.*

The sensor nodes do not run the distributed routing algorithms; they simply receive the forwarding rules and fill the flow table, which has greatly reduced their calculation pressure. The controller generates the secure channel using the shortest path method, and it generates forwarding rules (including the normal channel) using the new algorithm as described in the next section, whose foundation is the topology information and the minimum transmission power between neighbor nodes. After the forwarding rules are generated, they should be delivered to all the sensor nodes through the secure channel.

It worth noticing that there are two major differences between traditional SDN and our designed SD-WSNs. 1) In the control plane, the controller has a flow table which is used to process the data from the sensor nodes. The forwarding rules of the flow table are modified by the controller itself according to the predefined policies. 2) In the data plane, the sensor nodes are equipped with sensors to collect sensor data and create flows itself. Actually, by separating the data processing unit and the data collecting units from the controller and the sensor nodes respectively, our designed architecture is the same as the traditional SDN architecture. To further understand the principles of our designed SD-WSNs architecture, we propose the protocol structures of both the controller and the sensor nodes. These two new structures are shown in figure 3. The sensor nodes don't run the distributed routing algorithms. They just receive the forwarding rules and fill the flow table which has greatly reduced their calculation pressure. The controller creases forwarding rules according to the topology information and delivers them to the sensor nodes.

The SD-WSNs can be considered as a network graph $G(t) = (\mathcal{V}, \mathcal{L}(t))$, where $\mathcal{V}$ is the set of sensor nodes and $\mathcal{L}(t)$ is the set of connections between neighbor nodes. The sensor nodes have low mobility and are equipped with limited power sources. It is assumed that the sensor nodes

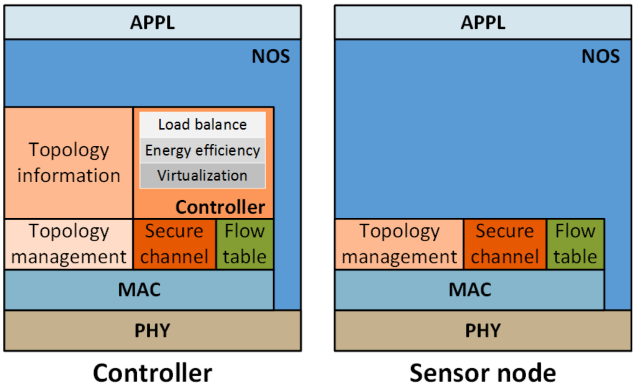


FIGURE 3. Protocol structures of sensor nodes and controller.

will not stop working until the energy is depleted, and the initial energy of all nodes is the same and is set as E_{ini}. The maximum transmission power of node i can be set as $P_i^{\max}$, and the transmission power of node i at time t can be flexibly adjusted; we have $P_i(t) \in \left(0, P_i^{\max}\right]$. The operation of the innovative SD-WSNs architecture is divided into two phases: initialization phase and maintenance phase.

A. INITIALIZATION PHASE

As mentioned in the first principle, prior to the start of the SDN mode, the controller has to learn information such as the node distribution and the initial energy of all the nodes in the network. In this phase, the controller and the sensor nodes will start a topology discovery procedure, which is against but necessary for the paradigm of SD-WSNs. The initialization processes are different in the controller and the sensor nodes.

In the sensor nodes, the main purpose of topology discovery is to find the neighbors and fill the neighbor table. Sensor node i periodically broadcasts HELLO_RQ (hello request) packets with power $P_i^{\max}$ and waits for the HELLO_RP (hello response) packets from its neighbors, which are feedback at $P_j^{\max}$ (if the neighbor is node j). The sensor nodes fill their neighbor tables with the information obtained from the HELLO_RP packets. Once the sensor nodes have received the TOPOLOGY_RQ (topology request) packets from the controller, the sensor nodes will packet the neighbor information (neighbor node address, neighbor node distance, and received power of the corresponding HELLO_RP packet) together with the node information (residual energy, $P_i^{\max}$ and so on) into TOPOLOGY_RP (topology response) packets. The TOPOLOGY_RP packets will be sent to the controller along the channel from which the TOPOLOGY_RQ packet comes from, and the TOPOLOGY_RQ will be broadcast to their neighbors with power $P_i^{\max}$; then, the node state can be set as topology maintenance.

In the controller, the main purpose of topology discovery is to obtain the network interconnection map (shown in an adjacency matrix) and other information (link quality, residual energy and the minimum transmission power between neighbor nodes, among others); based on such information, the controller can flexibly generate forwarding rules. A preset time after the start of the network (to ensure that the sensor nodes have found their neighbors), the controller broadcasts TOPOLOGY_RQ packets and waits for a period of time to receive the TOPOLOGY_RP packets. During the topology discovery period, the controller sends the information in the TOPOLOGY_RP packet to the TIM as long as it receives a TOPOLOGY_RP from a sensor node. After a fixed amount of time, the forwarding rules will be generated based on the selected algorithm in the TIM. Then, the initialization phase of the controller ends.

B. MAINTENANCE PHASE

In the maintenance phase, the main task is to adjust the forwarding rules according to changes in the network. The processes in the controller and the sensor nodes are also different.

In the sensor nodes, the main purpose of topology maintenance is to detect changes in neighbor relations. After sending the TOPOLOGY_RP to the controller, the sensor nodes start the topology maintenance procedure. The controller will periodically broadcast HELLO_RQ packets to collect the neighbor information, and the cycle length is adjustable. Prior to the next broadcast of HELLO_RQ, the sensor nodes compare the neighbor information collected during the last two HELLO_RQ periods, and the residual energy will be detected. If any differences are detected or the residual energy reaches the threshold, the sensor nodes packet the change information into TOPOLOGY_REPAIR packets and send them to the controller.

In the controller, the topology maintenance procedure is relatively simple. After the topology discovery, the controller continues waiting until it has received a TOPOLOGY_REPAIR packet or a new TOPOLOGY_RP packet. Then, the controller modifies the forwarding rules in the TIM according to the changed topology.

The operating mechanism of the SD-WSNs can be visualized in the process pseudo code, which is described in detail in the next section, to integrate our new algorithm.

III. DESIGN OF ORDINAL POTENTIAL GAME-BASED ENERGY-EFFICIENT ALGORITHM

Based on the novel SD-WSNs architecture, we can design a proprietary energy-efficient topology control algorithm to improve the energy efficiency of the SD-WSNs and extend the lifetime of the sensor node network. The innovative algorithm used the concept of ordinal potential game (OPG) in game theory, and we named it OPG-based energy-efficient algorithm (OPGEA).

OPGEA runs in the TIM, which is the most critical module in the controller. As the process of the controller shows, in the initialization phase, the network topology, secure channel, forwarding rules and the transmit power of each node will be decided in the TIM. And in the maintenance phase, whenever a TOPOLOGY_RP packet or a TOPOLOGY_REPAIR packet is received, the new topology information and node energy information will be transferred to the TIM. Subsequently, the network topology, secure channel, forwarding rules and the transmit power of each node will be adjusted immediately in the TIM based on the information carried in these packets. The following describes how OPGEA assimilates into the SD-WSNs in detail.

A. PRELIMINARY KNOWLEDGE

The SD-WSNs can be modeled as an undirected graph. Through game theory analysis, we can formally describe the topology control algorithm as a non-cooperative game $\Gamma = \langle \mathcal{V}, \mathcal{S}, \{u_i\} \rangle$, where $\mathcal{V} = \{1, 2, \cdots, n\}$ is the set of sensor nodes that correspond to the players in the game (we do not take the controller into account because the energy of the controller is assumed to be infinite, and it can always transmit messages at maximum power), $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_n$ is the Cartesian product of the strategy sets $\mathcal{S}_i$ for any sensor node $i \in \mathcal{V}$, and u_i is the utility function that the *ith* player desires to maximize. For each sensor node in the SD-WSNs, the utility function is dependent not only on the strategy $s_i \in \mathcal{S}_i$ that it has selected but also on the decisions made by other sensor nodes, represented by s_{-i}. The strategy s_i of sensor node i is said to be the best response to the fixed s_{-i} if it satisfies the following inequality:

$$u_i (s_i, s_{-i}) \geq u_i \left(s'{}_i, s_{-i}\right), \quad \forall s' \in S_i. \tag{1}$$

A desired stable solution in non-cooperative game theory is Nash equilibrium (NE), in which no player may improve its utility function by unilaterally deviating from it. The ultimate goal of our OPGEA is to determine the strategy tuple $\mathbf{p}^* = \left(p_1^*, p_2^*, \cdots, p_n^*\right)$, which is the set of power values of all the sensor nodes in the SD-WSNs, and $\mathbf{p}^*$ is an NE to operate the SD-WSNs in the optimum power combination to minimize energy consumption and ensure the premise of network connectivity.

Definition 1: A strategy tuple $\mathbf{s}^* = \left(s_1^*, s_2^*, \cdots, s_n^*\right)$ is an NE if s_i^* is the best response to s_{-i}^* for every player i. Formally, the strategy tuple $\mathbf{s}^*$ is an NE if

$$u_i \left(\mathbf{s}^*\right) \geq u_i \left(s_i, s_{-i}^*\right) \quad \textit{for } \forall i \in V \textit{ and } \forall s_i \in S_i. \tag{2}$$

B. SOME ASSUMPTIONS AND CONCEPTS

In our OPGEA, there are some assumptions and concepts that need to be established.

1) THE MINIMUM THRESHOLD FOR TRANSMISSION POWER

For any node i to communicate with its neighbor node j, its transmit power p_i should ensure that the received signal strength at node j exceeds the threshold. The threshold is the minimum received power to ensure that the received signal can be detected and correctly decoded, which is one of the node parameters. This condition can be formulated as

$$p_i \cdot G_{ij} \geq p^{th}. \tag{3}$$

where G_{ij} is the propagation factor that depends on the propagation channel model, and we assume that G_{ij} is a symmetric function, i.e., $G_{ij} = G_{ji}$. In the free space propagation mode, as an example, the propagation factor $G_{ij} = C \cdot d_{ij}^{-\alpha}$, where C is a constant, d_{ij} is the distance between sensor nodes i and j, and α is the path loss factor, which is typically in the range $2 \leq \alpha \leq 6$ [41].

2) THE CALCULATE OF THE TRANSMISSION POWER

Because the SD-WSNS is described as an undirected graph, all links in the network $\mathcal{G}(t)$ are bidirectional, given that the vast majority of channel access and routing protocols use only bidirectional links for their operations. Mathematically, a bidirectional link $l_{ij} \in \mathcal{L}(t)$ between sensor nodes i and j exists if and only if $\min\left\{p_i, p_j\right\} \geq p^{th}/G_{ij}$. We define $\omega(i, j) \triangleq p^{th}/G_{ij}$ as the minimum transmit power that supports a connection from i to j. Clearly, from our assumption, we have $\omega(i, j) = \omega(j, i)$. Here, the minimum transmit power $\omega(i, j)$ can be determined by measuring the received power of request (and/or reply) packets. In the free space propagation model, the relation between the power used to transmit packets P^t and the power received P^r can be characterized as $P^r = P^t \cdot G \cdot \lambda^2/(4\pi d)^2 \cdot L$, where G is the propagation factor between the transmitter and the receiver and L is the system loss. The formula can be reduced to $P^r = P^t \cdot \tilde{G}$, where $\tilde{G}$ is a function of G, λ, d, L, and α and is time invariant if all the above parameters are time invariant [42]. In the initialization phase, all the sensor nodes broadcast HELLO_RQ packets with maximum power and reply with HELLO_RP packets with maximum power. When node i receives the HELLO_RP from node j, the received power P_{ij}^r is recorded in its NL_i, and after the NL_i is sent to the controller, the calculation can be performed in the TIM as $\tilde{G}_{ij} = P_{ij}^r/P_i^{\max}$, leading to the following:

$$\omega(i, j) = p^{th}/\tilde{G}_{ij}. \tag{4}$$

3) THE MAXIMUM POWER

We assume that the maximum power values are the same for all the sensor nodes to simplify the calculation, which are set as $P^{\max}$. The network induced by all sensors transmitting with $P^{\max}$ can be defined as $\mathcal{G}_{\max} = (\mathcal{V}, \mathcal{L}_{\max})$, where $\mathcal{L}_{\max} = \left\{l_{ij} | \mathbf{p} = (P^{\max}, \cdots, P^{\max})\right\}$, and $\mathcal{G}_{\max}$ is the maximum neighbor graph of the SD-WSNs system.

4) THE GENERATION OF DYNAMIC TOPOLOGY

The transmit power of node i at time t, $p_i(t) \in \mathcal{S}_i$, is regarded as the strategy of node i at time t. The strategy set $\mathcal{S}$ can be obtained from the Cartesian product of all $\mathcal{S}_i$ $(1 \leq i \leq n)$, where $\mathcal{S}_i = [0, P^{\max}]$ is the set of power levels that can be selected by node i. Each power profile $\mathbf{p}(t) = (p_1(t), p_2(t), \cdots, p_n(t))$ induces a new topology graph $\mathcal{G}_{\mathbf{p}}(t) = \left(\mathcal{V}, \mathcal{L}_{\mathbf{p}}(t)\right)$, and $\mathcal{L}_{\mathbf{p}}(t) \subset \mathcal{L}_{\max}$ [43].

5) THE BENEFIT AND THE RESISTANCE

Every node in the SD-WSNs can benefit from connecting to other sensors in network $\mathcal{G}$, which we call **benefit** (Ben_i); simultaneously, each node has selfishness because the delivery of packets will lead to energy consumption, which we name **resistance** (Res_i). Intuitively, an excessive transmit power $p_i(t)$ will lead to faster energy consumption, which will increase the resistance, and a low residual energy $E_i(t)$ incurs a greater resistance for node i to become a relay node. The resistance is closely related to the utility function of this game method, and its concrete calculation is explained in the specific process follow-up. The benefit received by sensor node i from a connected network should cover its Res_i, and we assume that node i can receive a fixed benefit if the network is connected with a fixed path and zero benefit if the network loses connectivity. Therefore, we define the benefit of the network by $\text{Ben}\left(\mathcal{G}_{\mathbf{p}(t)}\right) = h_\Lambda(\mathbf{p}(t))$, where $h_\Lambda(\mathbf{p}(t))$ is the indicator function of strategy profile set $\Lambda = \left\{\mathbf{p}(t) : a\left(\mathcal{G}_{p(t)}\right) > \xi\right\}$ and ξ is a parameter that indicates the connectivity redundancy of network $\mathcal{G}_{\mathbf{p}(t)}$ [44].

C. THE PROCESS OF OPGEA

The implementation of our OPGEA can be mapped to the two phases of an SD-WSNs.

1) INITIALIZATION PHASE

In the initialization phase of an SD-WSNs, the main work of each node is to collect the information from its neighbor nodes and generate its NL. This process can be accomplished via the following steps with OPGEA embedded:

Step 1: When the system starts, each sensor node broadcasts a HELLO_RQ packet at $P^{\max}$ and attempts to obtain the message from all possible neighbor nodes and store them in $NL^{\max}$ (i.e., the maximum NL). After any neighbor node replies with a HELLO_RP packet, the received power of this packet will be recorded in $NL^{\max}$, which can be denoted as p^r_{ij} (the received power of HELLO_RP from the neighbor node j of node i).

Step 2: When a sensor node receives a TOPOLOGY_RQ packet from the controller, it will packet the $NL^{\max}$ into a TOPOLOGY_RP packet together with the residual energy $E_i(t)$. The TOPOLOGY_RP packet will be sent back to the controller along the same route that TOPOLOGY_RQ comes from. In the controller, further processing of the $NL^{\max}$ will be performed in the TIM, and after this process is complete, we can obtain $\mathcal{G}_{\max}$ and $\omega(i,j)$ for any adjacent nodes (as shown in Equation (4)).

Step 3: In the TIM, based on $\mathcal{G}_{\max}$ and $\omega(i,j)$, we can use the shortest path method to obtain the secure channel from each node to the controller, and the minimum power required to transmit a packet to its next hop in the secure channel can be obtained from the previous step, denoted as $\mathbf{p}^s = \left\{p^s_1, p^s_2, \cdots, p^s_n\right\}$.

Step 4: In the TIM, the Res_i is related to several factors:

- The power required to transmit a packet, i.e., $p_i(t)$: as $p_i(t)$ increases, Res_i will increase. According to the node's selfishness, a larger $p_i(t)$ will cause greater loss to the node i, therefore, node i will resist the increase in $p_i(t)$.
- The residual energy of the sensor node, i.e., $E_i(t)$: as $E_i(t)$ decreases, Res_i will increase. As the residual energy decreases, node i will protect its energy, thereby increasing its resistance.
- The importance of sensor nodes: we divided all the sensor nodes into two classes, namely, key nodes and common nodes. When a node satisfies any of the following conditions, it can be considered a key node: i) if the node is an articulation point in the network topology, and ii) if the number of neighbors of the node is in the top 10% of all nodes. If node i is classified as a key node, then Res_i should be stronger than the common nodes to ensure that the energy of key nodes will not be depleted prematurely.

Combined with the above factors, refer to the design of [41], we can design the resistance function as follows:

$$\text{Res}_i = \frac{1}{M_i} \cdot \int_{E_{ini}-E_i(t)}^{E_{ini}-E_i(t)+p_i(t)\cdot T} (\varphi k_i) \cdot f_i(x)\, dx, \tag{5}$$

where T denotes the unit transmission time satisfying $p_i(t)\, T \leq E_i(t)$, M_i is sufficiently large such that the value of Res_i belongs in $[0,\ 1]$, and k_i is defined as

$$k_i = \begin{cases} E_{ini}/E_i(t) & \textit{node i is a key node} \\ 1 & \textit{else}, \end{cases} \tag{6}$$

φ is a constant related to the impact of the node being a key node or not, and the increasing function $f_i(x)$ is defined as a pricing function indicating the price when x units of energy has been used.

Step 5: Because $Ben_i = 0$ when the network loses connectivity, the utility function can ensure the network connectivity as a top priority, and it captures the fact that sensor nodes can regulate their powers to change the routing path. Ben_i is also associated with $p_i(t)$. As $p_i(t)$ decreases, the benefit Ben_i is non-decreasing, i.e., Ben_i is a non-decreasing function of $p_i(t)$. Moreover, the number of hops to the controller, denoted as $\mathcal{H}_i$, should also affect Ben_i. As the transmit power of node i decreases, $\mathcal{H}_i$ will be non-decreasing, and the energy consumption of the entire network may also increase (the energy consumption of node i will decreases, however, more nodes will consume energy in this transmission process), i.e., $\mathcal{H}_i$ is also related to $p_i(t)$. We can design the benefit function as follows:

$$Ben_i = \begin{cases} \text{network is not connected}: \\ 0 \\ \text{network is connected}: \\ \beta_i \cdot \int_0^{p_i(t)\cdot T} h(\mathcal{H}_i) \cdot \eta(p_i(t)) \cdot g_i(x)\, dx, \end{cases} \tag{7}$$

where T denotes the unit transmission time satisfying $p_i(t)\, T \leq E_i(t)$, β_i is used to ensure that $Ben_i > 1$ as Ben_i should cover Res_i, $\mathcal{H}_i$ is non-decreasing as $p_i(t)$ increases. And $h(\mathcal{H}_i)$ is a decreasing function of $\mathcal{H}_i$, as the income will

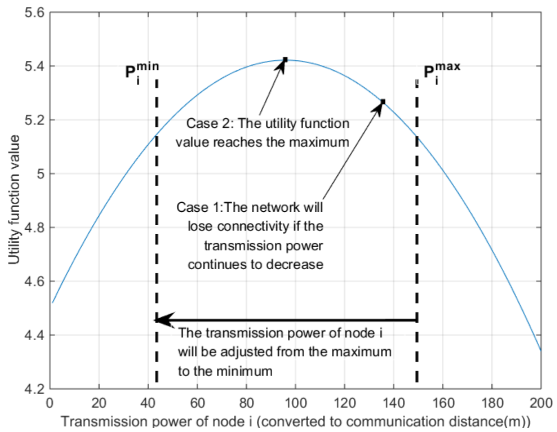


FIGURE 4. The sketch of utility function (Part I).

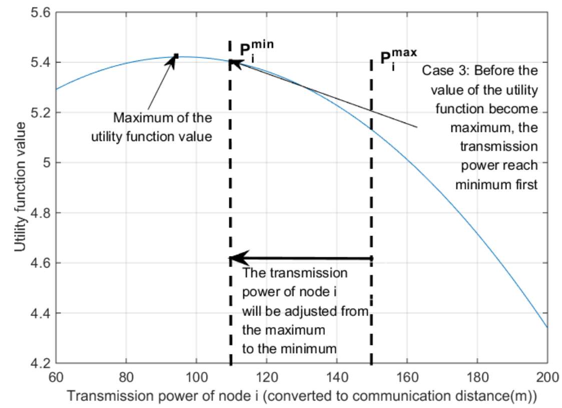


FIGURE 5. The sketch of utility function (Part II).

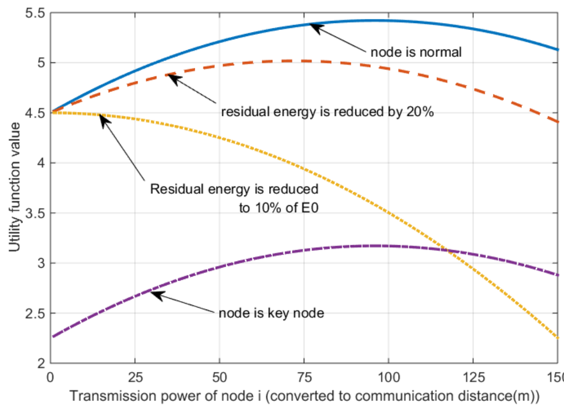


FIGURE 6. The changing trend of utility function.

be scattered as the number of hops to the controller increases, and $h(\mathcal{H}_i)$ should also be affected by the network topology. $\eta(p_i(t))$ is a decreasing function of $p_i(t)$, and the increasing function $g_i(x)$ is defined as an income function indicating the income when x units of energy has been used.

$p_i(t)$ has an impact on both Ben_i and Res_i. Combining the benefit with the resistance, we can obtain the utility of sensor node i as follows:

$$u_i(\mathbf{p}(t)) = \text{Ben}_i(\mathcal{H}_i, p_i(t)) - \text{Res}_i(p_i(t), E_i(t), k_i)$$
$$= \begin{cases} \text{network is not connected}: \\ -\text{Res}_i(p_i(t), E_i(t), k_i) \\ \text{network is connected}: \\ \beta_i \cdot \int_0^{p_i(t)\cdot T} h(\mathcal{H}_i) \cdot \eta(p_i(t)) \cdot g_i(x)\,dx \\ -\frac{1}{M_i} \cdot \int_{E_{ini}-E_i(t)}^{E_{ini}-E_i(t)+p_i(t)\cdot T} (\varphi k_i) \cdot f_i(x)\,dx. \end{cases} \quad (8)$$

Step 6: The procedure of the power adaptation (to obtain the optimal utility function) in an iteration can be expressed as follows:

a: RESTRICTIONS

S1: the network connectivity is guaranteed;

S2: $P_i^{\min} \le p_i(t) \le P_i^{\max}$, the initial value of $P_i^{\max}$ is $P^{\max}$;

S3: $p_i(t)$ should be adjusted from $P_i^{\max}$ to $P_i^{\min}$, and the magnitude of this decrease, i.e., δ_i, should be sufficiently small to ensure that at most one link is dropped (actually, if a large step size δ_i is chosen, then the topology derived by the game-based algorithm may deviate from the energy efficiency [25]);

S4:

$$p_i^* = \underset{p_i \in p_i^* \cup (p_i^* - \delta_i)}{\arg\max} \; u_i(p_i, p_{-i}). \quad (9)$$

Through the iterative iteration of the above optimization problem, we can find the p_i^* to reach the NE of the utility function (8) of node i. Sketches showing the curve trend and the iteration process are presented in Figure 4 and Figure 5. We fit the utility function curve with a quadratic function curve because the trend of the two curves is approximately the same; thus, the quadratic function is more intuitive. In each round of the adjustment process, the transmission power of node i will be adjusted from $P_i^{\max}$ to $P_i^{\min}$. We attempt to find the maximum utility function value, and the iteration process stops in the following three cases:

Case1: Once the network loses connectivity, this process will be terminated, and the $p_i(t)$ will be set to the transmission power value in the last round.

Case2: When the utility function value reaches its maximum, the process will be terminated, and the $p_i(t)$ will be set to the corresponding transmit power value.

Case3: If $p_i(t)$ reaches its minimum first, then the process will be terminated, and the $p_i(t)$ will be set to $P_i^{\min}$.

Figure 6 presents the effect of parameter variations on the utility function. As shown in this figure, the key nodes have lower utility values; thus, the probability that the key node is selected as the relay node in the same situation is less than the normal node. Moreover, as the residual energy decreases, the utility value also decreases, and its peak is shifted to the left and the selected $p_i(t)$ will be lower than the previous power value. When the residual energy of the node reaches its threshold, the peak of the utility function curve will be shifted to the left in a large step, and the $p_i(t)$ is most likely to be selected as $P_i^{\min}$, which will greatly improve the lifetime of endangered nodes.

Step 7: As defined by the utility function, a game is played by all sensors selecting their individual powers. The game with the utility function of each sensor node by Function (8) is an OPG; then, the existence of NEs is guaranteed (For the proof of the process, reference the Appendix).

Definition 2: A strategic game $\Gamma = \langle \mathcal{V}, \mathcal{S}, \{u_i\} \rangle$ is an OPG if a function $V : S \rightarrow \mathbb{R}$ exists such that $\forall i \in \mathcal{V}, \forall p_{-i} \in \mathcal{S}_{-i}$, and for all $p_i, q_i \in \mathcal{S}_i$, we have

$$V(p_i, p_{-i}) - V(q_i, p_{-i}) > 0$$
$$\Leftrightarrow u_i(p_i, p_{-i}) - u_i(q_i, p_{-i}) > 0. \quad (10)$$

where V is called the ordinal potential function (OPF) of Γ.

Based on *Definition 2*, we can have the OPF of the entire network as

$$V(\mathbf{p}(t)) = h_\Lambda(\mathbf{p}(t)) - \frac{1}{M}\sum_{i=1}^{n}\int_{E_{ini}-E_i(t)}^{E_{ini}-E_i(t)+p_i(t)T}(\varphi k_i) f_i(x)\,dx. \quad (11)$$

With all the sensor nodes attempting to maximize their utility function, the OPF (Function (11)) of the entire network can have the optimal power configuration for all the sensor nodes, i.e., $\mathbf{p}^* = \{p_1^*, p_2^*, \cdots, p_n^*\}$; it is the NE of the OPG, and it is Pareto optimal.

Step 8: Through the determination of the transmission power, we can obtain the network connection relationship, and the path from each node to the controller can easily be determined. The initial forwarding rules, the optimal power configuration, the secure channel and the $\mathbf{p}^s$ will be packed into F_RULES (forwarding rules) packets and transmitted to each node through the secure channel with transmission power p_i^s.

Subsequently, the initialization phase ends, and the SD-WSNs enters the topology maintenance phase.

2) MAINTENANCE PHASE

The maintenance phase consists of two parts: normal operation and endangered node operation.

a: NORMAL OPERATION

In the maintenance phase of an SD-WSNs, all the sensor nodes simply perform the work of receiving and forwarding. The transmission power of each node can be found in the $\mathbf{p}^*$, which has been sent to each node in the initialization phase. Nodes do not need to perform any calculations, they only have to detect the residual energy E_i after each time that a packet is forwarded.

b: ENDANGERED NODE OPERATION

1) We set a threshold for the residual energy, which is 10% of the E_{ini}. When the residual energy of node i reaches the threshold, node i will be considered an endangered node, warning information will be packed into the TOPOLOGY_REPAIR packet and sent to the controller through the secure channel. In the TIM, we set the endangered node as invisible, and the node endangered will only transport packets in the following three cases:
   - When node i is a node on the secure channel and a control message or an emergency message needs to be transmitted;
   - The node itself has information that needs to be transmitted to the controller;
   - One of the neighbor nodes of node i has only one path to connect to the controller even to transmit at $P^{\max}$, and this neighbor node will become an isolated node if node i becomes invisible.
2) When the controller receives a TOPOLOGY_REPAIR packet, the process will be performed in the TIM. The new routing will be calculated, node i will be set as invisible as previously mentioned, and it will become the first node to adjust its transmit power, because the "first-mover advantage", the transmit power of node i may result in a greater utility value and can obtain a relatively lower transmit power than before (as the Fig. 5 shows). Before the route is re-planned, the maximum power of the endangered node i should be set to p_i^* because the power of node i should not be greater than the previous value to ensure that the power loss rate of node i does not increase.
3) By adjusting the order in which the nodes maximize the utility function, the controller can re-plan the route, repeat Steps 4 ∼ 7 of the initialization phase in TIM, and re-obtain the updated $\mathbf{p}^* = \{p_1^*, p_2^*, \cdots, p_n^*\}$.
4) Repeats Step 8 of the initialization phase, packs the updated forwarding rules and the optimal power configuration into a MODIFIED_RULES (modified forwarding rules) packet, and sends the packet to each node through the secure channel with transmission power $\mathbf{p}^s$.

Subsequently, the maintenance phase continues.

D. THE PSEUDO CODE OF OPGEA OPERATES IN SD-WSNs

The behavior of nodes and the controller in the two phases of OPGEA in the SD-WSNs architecture can be described by the pseudo code in Listing 1 (nodes) and Listing 2 (controller).

IV. SIMULATION RESULTS AND ANALYSIS

In this section, we first construct a simulation platform for our proposed SD-WSNs in OPNET to prove the feasibility of SDN over WSNs. Processes such as topology discovery, topology maintenance and forwarding rule generation are simulated in the simulation platform. Then, we add OPGEA into the simulation platform to evaluate the performance of our proposed algorithm. Our proposed algorithm is compared with the shortest-path algorithm [45] and some traditional energy-efficient routing algorithms, such as LEACH-C [46] and E-TORA [47].

```
Node internal operation:
 Get the necessary information.
 Start a timer $t$ to periodically broadcast HELLO_RQ.
 Initialize the node state as Initialization Phase.
 Initialize maximum transmission power of node $i$ as $P_i^{\max}=P^{\max}$.
while 1 do
  Wait until an interrupt occur
  if Timer $t$ runs out then
    Broadcast a HELLO_RQ packet with $P_i^{\max}$;
    Reset the timer $t$ (to prevent network topology changes);
  else if A packet is received then
    if The node state is Initialization Phase then
      switch Type of the packet
        case HELLO_RQ packet:
          Respond a HELLO_RP packet with $P^{\max}$;
          break;
        case HELLO_RP packet:
          Update the $NL$, deposit the neighbor node information
          and the received power of this HELLO_RP packet;
          break;
        case TOPOLOGY_RQ packet:
          Packet the $NL$ into TOPOLOGY_RP;
          Send TOPOLOGY_RP to controller along the same route that
          the TOPOLOGY_RQ comes from;
          Rebroadcast the TOPOLOGY_RQ to its neighbors;
          Set node state as Maintenance Phase;
          break;
        case F_RULES packet:
          Store the forwarding rules;
          Transmission power of node $i$ in normal channel: $p_i(t) = p_i^*$;
          Transmission power of node $i$ in secure channel: $p_i(t) = p_i^s$;
          break;
      end switch
    else if The node state is Maintenance Phase then
      switch Type of the packet
        case Data packet:
          if Node $i$ is the destination node then
            Receive this data packet;
          else
            Forward the data packet to next hop node with $P_i^*$ in
            normal channel;
          end if
          break;
        case HELLO_RQ packet:
          Respond a HELLO_RP packet with $P^{\max}$;
          break;
        case HELLO_RP packet:
          if $NL$ has changed then
            Packet the updated $NL$ into TOPOLOGY_REPAIR;
            Send TOPOLOGY_REPAIR to next hop node with $P_i^s$ in
            secure channel;
          else
            break;
          end if
          break;
        case TOPOLOGY_REPAIR packet:
          Send TOPOLOGY_REPAIR to next hop node to the controller
         with $P_i^s$ in secure channel;
          break;
        case F_RULES packet:
          Store the forwarding rules;
          Transmission power of node $i$ in normal channel: $p_i(t) = p_i^*$;
          Transmission power of node $i$ in secure channel: $p_i(t) = p_i^s$;
          break;
        case MODIFIED_RULES packet:
          Store the forwarding rules;
          Adjust the transmission power in normal channel and secure
          channel;
          break;
      end switch
      After each forwarding packet, read the residual energy $E_i$;
      if $E_i < 10\% \cdot E_{ini}$ then
        Bale the warning information into the TOPOLOGY_REPAIR
         packet;
        Forward the TOPOLOGY_REPAIR packet to next hop with $P_i^s$
         in secure channel;
      else
        The node remains in its original state;
      end if
    end if
  end if
end while
```

LISTING 1. Process of nodes in OPGEA.

```
Controller internal operation:
 Get the necessary information.
 Start a timer $t1$ to broadcast TOPOLOGY_RQ.
 Set the controller state as Initialization Phase.
 The transmission power of the controller is fixed to $P^{\max}$.
while 1 do
  Wait until an interrupt occur;
  if Timer $t1$ runs out then
    Broadcast a TOPOLOGY_RQ packet with $P^{\max}$;
    Set timer $t2$ to collect the neighbor information;
  else if Timer $t2$ runs out then
    Establish secure channel based on the shortest path method,
    determine the transmission power $\mathbf{p}^s = \{p_1^s, p_2^s, \cdots, p_n^s\}$ of the
    secure channel;
    According to the information already obtained, generate the
    forwarding rules in TIM based on OPGEA, specific implementation
    can be find in step 4~7, the F_RULES packet will be delivered to
    all the nodes in secure channel with $\mathbf{p}^s = \{p_1^s, p_2^s, \cdots, p_n^s\}$;
    Set controller state as Maintenance Phase;
  else if A packet is received then
    if Controller state is Initialization Phase then
      switch Type of the packet
        case HELLO_RQ packet:
          Respond a HELLO_RP packet with $P^{\max}$;
          break;
        case TOPOLOGY_RP packet:
          Update the topology information;
          break;
        case TOPOLOGY_REPAIR packet:
          Update the topology information;
          break;
      end switch
    else if Controller state is Maintenance Phase then
      switch Type of the packet
        case HELLO_RQ packet:
          Respond a HELLO_RP packet with $P^{\max}$;
          break;
        case TOPOLOGY_RP packet:
          Update the topology information;
          if The network structure changes then
            Adjust the forwarding rules based on the OPGEA, the
            updated forwarding rules will be packed into
            MODIFIED_RULES packet, the packet will be sent to each
            node via secure channel, the transmission power is
            $\mathbf{p}^s = \{p_1^s, p_2^s, \cdots, p_n^s\}$;
          else
            break;
          end if
          break;
        case TOPOLOGY_REPAIR packet:
          Update the topology information;
          Adjust the forwarding rules based on the OPGEA, revise
          the order of node updates, the routing of endangered node
          $i$ will be planned first, and the maximum power of the
          node $i$ will be changed to its previous transmission power;
          if The forwarding rules changes then
            Delivery the new forwarding rules (MODIFIED_RULES
            packet) to all the nodes via secure channel with power
            $\mathbf{p}^s = \{p_1^s, p_2^s, \cdots, p_n^s\}$;
          else
            break;
          end if
          break;
      end switch
    end if
  end if
end while
```

LISTING 2. Process of controller in OPGEA.

A. SIMULATION PARAMETER SETTINGS

The multihop IEEE 802.15.4 OPNET simulation model [48] developed with OPNET Modeler is used as a basis for implementing and testing the proposed OPGEA. We adopt the event radius (ER) model [49] to simulate the impulsive traffic triggered by temporally and spatially correlated monitoring events in a disk area. Following the ER model, the monitoring area of SD-WSN is divided into an event gathering region, a data relaying region, and a decision making region. The first two regions belong to data plane, and the third one belongs to control plane. N sensor nodes are randomly deployed in a squared monitoring area with side length L, and the Controller is placed at the center of the monitoring

TABLE 1. Simulation parameters.

Monitoring networks deployment	
Application	ER model
L	$800m$
N	150
$P^{\max}$(Convert to radius)	$150m$
Deployment type	Random
Antenna	Omni
Network architecture	Homogeneous
Poisson arrival rate	Adjustable
Path loss exponent	2.4
Controller coordinates	(L/2,L/2)
Energy model	First-order
Communication parameters	
MAC	802.11 edcf
MAC frame	272 bits
PHY layer rate	46∼512 kbit/s
PHY frame	128 bits
ε_{fs}	$10pJ/bit/m^2$
ε_{mp}	$0.0013pJ/bit/m^4$
p	$0.01packet/s$
k	$5,000bits$
d_0	$87m$
ε_0	$10J$
E_{elec}	$50nJ/bit$

area $(L/2, L/2)$. The arrival of events follows a Poisson distribution in the time domain. Note that all the experiment results include the energy consumptions of both data gathering and control traffic. In the energy consumption calculation, we adopt the energy consumption model in [50]. It is assumed that the radio expends E_{elec} to run the transmitter and receiver circuitry. The radio expends ε_{fs} to run the transmit amplifier in the free space model, while it expends ε_{mp} in the multipath model. d^2 and d^4 energy loss due to the channel fading are assumed. Let d_0 denote the distance threshold to decide which radio model is used. The energy consumption of transmitting a $k-bit$ packet between sensor nodes with distance d is given by the following:

$$E_{tr}(k, d) = \begin{cases} k \times E_{elec} + k \times \varepsilon_{fs} \times d^2 & d < d_0 \\ k \times E_{elec} + k \times \varepsilon_{mp} \times d^4 & d \geq d_0. \end{cases} \quad (12)$$

The energy consumption of the receiver is given by the following:

$$E_{rx}(k) = k \times E_{elec}. \quad (13)$$

The parameter values used for the experiment setup are given in Table 1.

B. ANALYSIS OF SIMULATION RESULTS

1) ENERGY BALANCE

Figure 7 presents snapshots of the energy distributions in different algorithms at the same simulation time. As shown in (d), the sensor nodes that have similar distances to the controller have almost the same residual energy in our OPGEA, i.e. the color that indicates the residual energy is evenly distributed, and the band is centered on the Controller. This phenomenon shows that these nodes have similar distances to the controller have the same rate of energy consumption and can deplete their energy at almost the same time. As shown in (c), the shortest-path algorithm is designed to quickly deliver messages without take the energy loss into account, the energy consumption of the nodes becomes very uneven, therefore, some key nodes that forward too many messages will deplete their energy within a short time period. And in (a) and (b), we can see that the energy consumption in the E-TORA and the LEACH-C are both better than that in the shortest-path algorithm, as the energy-efficient has been taken into account. However, the maximum difference in residual energy in these two energy-efficient algorithm is greater than that in OPGEA, which side reflects that the energy consumption in OPGEA is more average, and the death nodes will appear earlier in the E-TORA and the LEACH-C, i.e. the OPGEA has longer network lifetime. The energy distribution in OPGEA is flatter than that in other algorithms because we take the residual energy into consideration. With the reduction of residual energy, the resistance will become stronger, and the load will be spread to its neighbor nodes. As the energy consumption becomes more uniform, the network lifetime can be greatly improved, which is intuitively displayed in the next section. However, the energy consumption of the entire network in OPGEA may be greater than that in other algorithms, which is because the link adjustment in OPGEA when an endangered node appears may lead to an increase in the number of hops to the controller, which can be see in Figure 8.

2) ENERGY CONSUMPTION

The energy consumption of the entire network and the energy consumption per packet (in a single node) are shown in Figure 8 and Figure 9, respectively. We assume that when all sensor nodes are dead nodes or isolated nodes, the network is considered a dead network, whose energy consumption remains the same. As shown in Figure 8, before the death of the network, the energy consumption of the shortest-path algorithm is the lowest at the same time. In energy-efficiency algorithms, to balance the energy consumption of sensor nodes at key positions, the flow may travel a longer path to the controller compared to the shortest-path algorithm, therefore, the energy consumption of the entire network will become more, and the network will die earlier (the time point that the curve becomes horizontal). As the energy consumption becomes more uniform, the OPGEA can consume more energy (the corresponding abscissa that the curve becomes horizontal), the energy of the entire network can be more fully applied, i.e. there will be less residual energy in the dead network. Figure 9 shows that in the OPGEA, the average energy consumption per packet in each node is the lowest, because the nodes in OPGEA have selfishness, always try

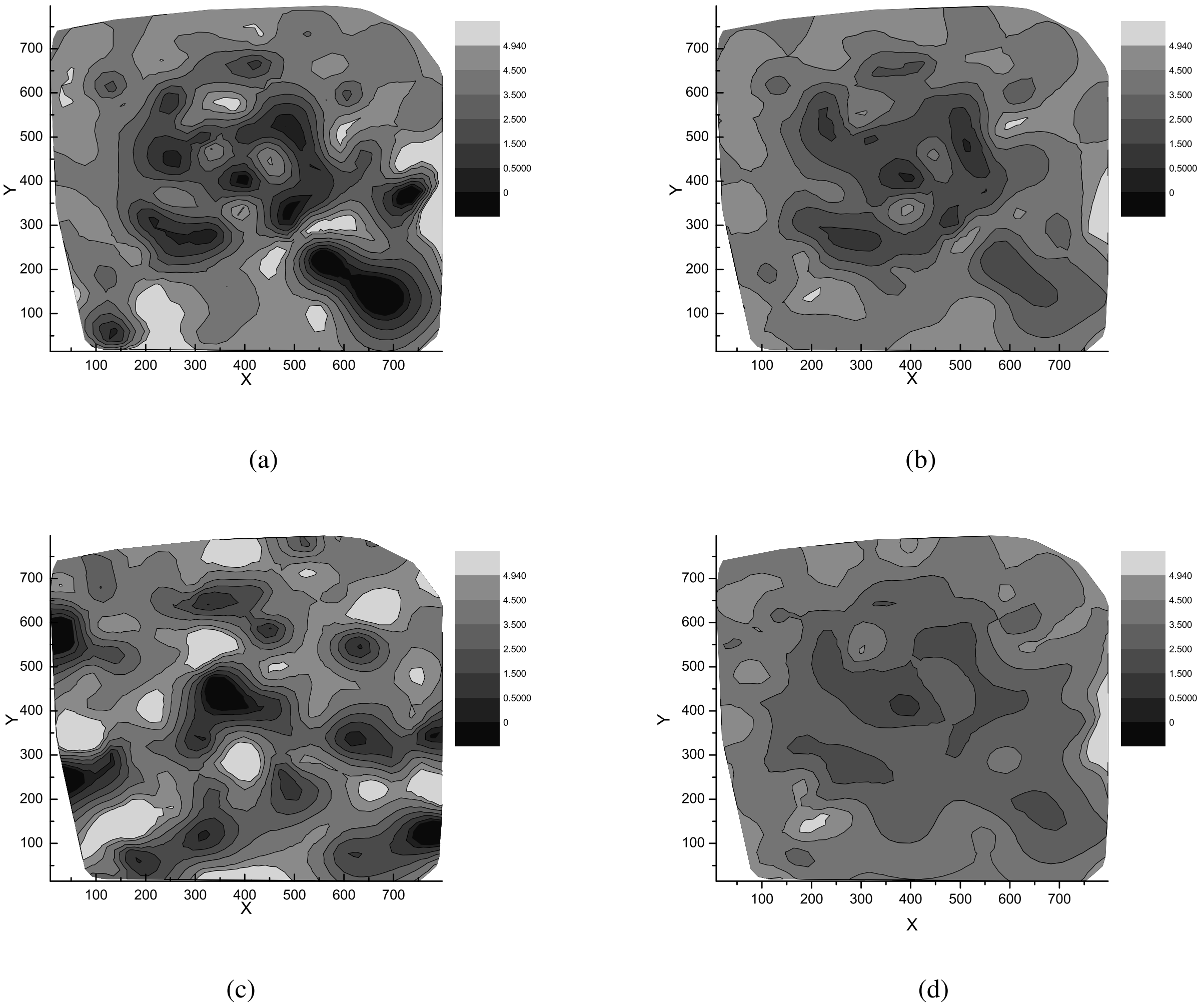


FIGURE 7. The snapshots of the energy distributions in different algorithms. (a) E-TORA. (b) LEACH-C. (c) Shortest path. (d) OPGEA.

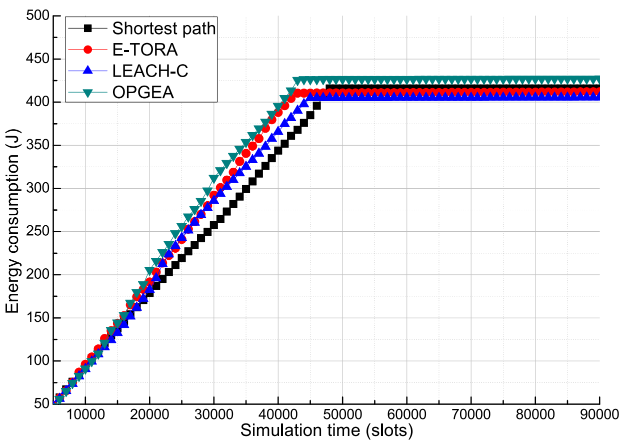


FIGURE 8. The energy consumption curve.

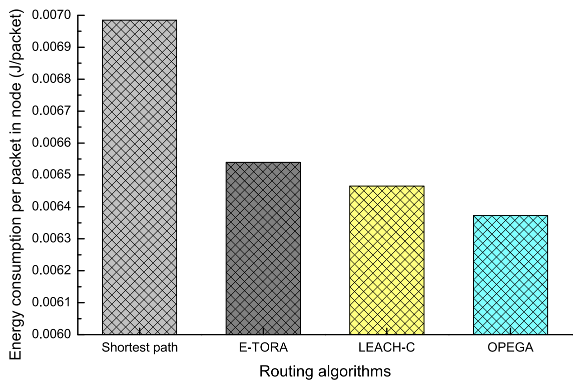


FIGURE 9. The energy consumption per packet.

to minimize transmission energy while meeting the needs of the network. However, the total energy consumption of the network in OPGEA increased by the number of hops, in the process of minimizing the energy consumption of the node, the number of hops to the controller is unavoidably increased (see Step 5 for details), which will lead to more energy consumption of the entire network. And in the shortest-path algorithm, the small number of hops can compensate for the high energy consumption of a packet in a single node.

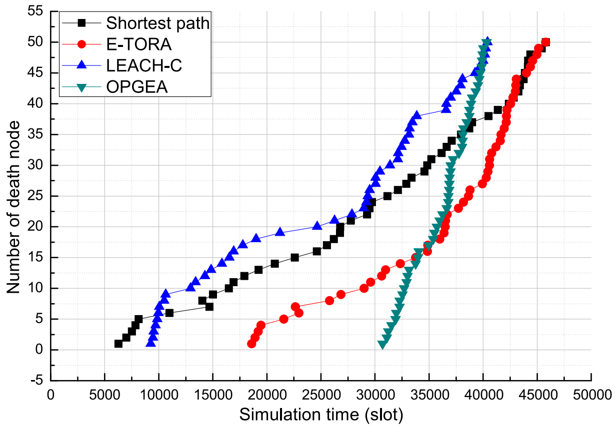


FIGURE 10. **Number of dead sensor nodes.**

3) NETWORK LIFETIME

Figure 10 shows the number of dead nodes of different algorithms versus the simulation time. The number of dead nodes, which directly reflects the network lifetime, increases continuously with the passage of time. In this paper, we define the network lifetime as the time from the beginning to the time when the first dead node appears. As shown in Figure 10, we can easily observe that OPGEA has a longer network lifetime than the other algorithms, which is represented by the time corresponding to the starting point of each curve, and the curve of OPGEA starts later than those of the other algorithms. The reason can be found in the Figure 7, the energy consumption in the OPGEA is evenly distributed, there will be no node to prematurely deplete its energy. The E-TORA and the LEACH-C can have a longer network lifetime than the shortest-path algorithm, as the energy efficiency is higher. The sensor nodes around the controller with the same distance deplete their energy at almost the same time, the slope of the curve reflects this property because the curve of OPGEA is steeper, i.e. more nodes will become dead nodes in a shorter period of time. There are more nodes that are blacked out in a short period of time, which means that the energy consumption of the node is very close. This property can be observed in Figure 7 (the residual energy is evenly distributed).

V. CONCLUSION

In this paper, we conceived an innovative SD-WSNs architecture that fully considers the particularity of traditional WSNs and the superiority of SDN. The SD-WSNs architecture is primarily applicable to the case of massive low-mobility nodes, and the advantage of centralized control in SD-WSNs will greatly reduce the computational requirements of sensor nodes, thereby reducing the energy consumption of all the sensor nodes. Based on the SD-WSNs architecture, a game-theoretic and energy-efficient algorithm is presented, i.e., OPGEA. OPGEA is designed based on the concept of OPG in game theory, which takes the residual energy of sensor nodes into consideration and introduces node selfishness. In this algorithm, the energy consumption of the nodes can be balanced, and the lifetime of the network can be extended. The simulation results show that our algorithm performs better in balancing energy consumption, prolonging network lifetime and increasing energy efficiency than other existing algorithms. Further studies will focus on the design of the SD-WSNs hardware platform, and the OPGEA can be applied to this platform to implement low-power WSNs.

APPENDIX

Theorem 1: The game $\Gamma = \langle \mathcal{V}, \mathcal{S}, \{u_i\} \rangle$ is an OPG. The OPF is given by

$$V(\mathbf{p}(t)) = h_\Lambda(\mathbf{p}(t)) - \frac{1}{M}\sum_{i=1}^{n}\int_{E_{ini}-E_i(t)}^{E_{ini}-E_i(t)+p_i(t)T} (\varphi k_i) f_i(x)dx, \tag{14}$$

where $M = \max\{M_1, M_2, \cdots, M_n\}$.

Proof: This proof is processed according to the definition of OPG. We drop the time parameter t for brevity. For each sensor node $i \in \mathcal{V}$, we have $p_{-i} \in \mathcal{S}_{-i}$ and $p_i, q_i \in \mathcal{S}_i$, it is sufficient to prove that the difference in utility for node i unilaterally changing its strategy from p_i to q_i and the difference in values of the global potential function have the same sign. Without loss of generality, let $p_i > q_i$. From the property of algebraic connectivity, we immediately know $a\left(G_{(p_i,p_{-i})}\right) \geq a\left(G_{(q_i,p_{-i})}\right)$. Meanwhile, we have $\text{Res}\,(p_i, E_i, k_i) > \text{Res}\,(q_i, E_i, k_i)$ owing to our conceived Resistance function. Firstly, the difference in utility of node i is

$$\begin{aligned}\Delta u_i &= u\,(p_i, p_{-i}) - u\,(q_i, p_{-i}) \\ &= h_\Lambda\,(p_i, p_{-i}) - h_\Lambda\,(q_i, p_{-i}) \\ &\quad + \text{Res}_i\,(q_i, E_i, k_i) - \text{Res}_i\,(p_i, E_i, k_i)\,.\end{aligned} \tag{15}$$

Similarly,

$$\begin{aligned}\Delta V &= V\,(p_i, p_{-i}) - V\,(q_i, p_{-i}) \\ &= h_\Lambda\,(p_i, p_{-i}) - h_\Lambda\,(q_i, p_{-i}) \\ &\quad + \frac{1}{M}\left(M_i\text{Res}_i\,(q_i, E_i, k_i) + \sum_{j\neq i} M_j\text{Res}_j\left(p_j, E_j, k_j\right)\right) \\ &\quad - \frac{1}{M}\left(M_i\text{Res}_i\,(p_i, E_i, k_i) + \sum_{j\neq i} M_j\text{Res}_j\left(p_j, E_j, k_j\right)\right).\end{aligned} \tag{16}$$

It obvious that the difference $h_\Lambda\,(p_i, p_{-i}) - h_\Lambda\,(q_i, p_{-i})$ is equal to 1 if $(p_i, p_{-i}) \in \Lambda$, $(q_i, p_{-i}) \notin \Lambda$, and 0, otherwise. By the fact $M \geq M_i$ and $1 \geq \text{Res}_i^{\max}$, we have

$$\begin{aligned}&\text{sgn}\,(\Delta u_i) \\ &= \text{sgn}\,(\Delta V) \begin{cases} < 0 & (p_i, p_{-i}) \in \Lambda, \quad (q_i, p_{-i}) \in \Lambda \\ < 0 & (p_i, p_{-i}) \notin \Lambda, \quad (q_i, p_{-i}) \notin \Lambda \\ \geq 0 & (p_i, p_{-i}) \in \Lambda, \quad (q_i, p_{-i}) \notin \Lambda. \end{cases}\end{aligned} \tag{17}$$

Therefore, the game $\Gamma = \langle \mathcal{V}, \mathcal{S}, \{u_i\} \rangle$ is an OPG and V is the OPF.

Theorem 2: The NE $\mathbf{p}$ is Pareto optimal, if the network $\mathcal{G}_{\mathbf{p}}$ is connected.

Proof: Let $\mathbf{p}$ be an NE point and $\mathcal{G}_{\mathbf{p}}$ be connected. We assume that $\mathbf{p}$ is not a Pareto optimal, i.e., there exists another power profile $\mathbf{q} = (q_1, \cdots, q_n)$ such that $u_i(\mathbf{q}) \geq u_i(\mathbf{p})$ for $\forall i \in \mathcal{V}$ and for some $k \in \mathcal{V}$, $u_k(\mathbf{q}) > u_k(\mathbf{p})$. That is, $\text{Res}(p_i, E_i, k_i) \geq \text{Res}(q_i, E_i, k_i)$ for $\forall i \in \mathcal{V}$ and for some $k \in \mathcal{V}$, $q_k < p_k$. The reduction from p_k to q_k for any node k leads to a disconnected network topology. Otherwise, the profile $\mathbf{p}$ is not an NE. Which is a contradiction, so we can have the conclusion.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interest regarding the publication of this article.

REFERENCES

[1] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 551–591, 2nd Quart., 2013.

[2] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Comput. Netw.*, vol. 67, pp. 104–122, Jul. 2014.

[3] M. Eslaminejad and S. A. Razak, "Fundamental lifetime mechanisms in routing protocols for wireless sensor networks: A survey and open issues," *Sensors*, vol. 12, no. 10, pp. 13508–13544, Jan. 2012.

[4] H.-Y. Shi, W.-L. Wang, N.-M. Kwok, and S.-Y. Chen, "Game-theory for wireless sensor networks: A survey," *Sensors*, vol. 12, no. 7, pp. 9055–9097, Jul. 2012.

[5] R. Feng, T. Li, Y. Wu, and N. Yu, "Reliable routing in wireless sensor networks based on coalitional game theory," *IET Commun.*, vol. 10, no. 9, pp. 1027–1034, Oct. 2016.

[6] S. Shamshirband, "Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks," *Eng. Appl. Artif. Intell.*, vol. 32, no. 2, pp. 228–241, 2014.

[7] N. Farzaneh and M. H. Yaghmaee, "An adaptive competitive resource control protocol for alleviating congestion in wireless sensor networks: An evolutionary game theory approach," *Wireless Pers. Commun.*, vol. 82, no. 1, pp. 123–142, 2015.

[8] Q. Sun, Y. Qiang, and W. Song, "Energy-balanced unequal clustering routing protocol based on game theory for wireless sensor networks," *J. Comput. Appl.*, vol. 34, no. 11, pp. 3164–3169, 2014.

[9] S. Misra, T. Ojha, and A. Mondal, "Game-theoretic topology control for opportunistic localizationin sparse underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 5, pp. 990–1003, May 2015.

[10] B. S. Jamin, M. Chatterjee, and T. Samanta, "A game theoretic routing framework based on energy-delay conservation in WSNs," in *Proc. Intell. Sensors, Sensor Netw. Inf. Process. (ISSNIP)*, Apr. 2015, pp. 1–5.

[11] D. Sathian, R. Baskaran, and P. Dhavachelvan, "Lifetime enhancement by Cluster Head Cooperative Trustworthy Energy Efficient MIMO routing algorithm based on game theory for WSN," in *Proc. 3rd Int. Conf. Comput. Commun. Network. Technol. (ICCCNT)*, Jul. 2012, pp. 1–5.

[12] H. I. Kobo, A. M. Abu-Mahfouz, and G. P. Hancke, "A survey on software-defined wireless sensor networks: Challenges and design requirements," *IEEE Access*, vol. 5, pp. 1872–1899, 2017.

[13] T. Luo, H.-P. Tan, and T. Q. S. Quek, "Sensor OpenFlow: Enabling software-defined wireless sensor networks," *Commun. Lett.*, vol. 16, no. 11, pp. 1896–1899, Nov. 2012.

[14] A. De Gante, M. Aslan, and A. Matrawy, "Smart wireless sensor network management based on software-defined networking," in *Proc. 27th Biennial Symp. Commun. (QBSC)*, Jun. 2014, pp. 71–75.

[15] F. Olivier, G. Carlos, and N. Florent, "SDN based architecture for clustered WSN," in *Proc. 9th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput. (IMIS)*, Jul. 2015, pp. 342–347.

[16] Z.-J. Han, and W. Ren, "A novel wireless sensor networks structure based on the SDN," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 3, 2014, doi: 10.1155/2014/874047.

[17] D. O'Shea, V. Cionca, and D. Pesch, "The presidium of wireless sensor networks—A software defined wireless sensor network architecture," in *Proc. Int. Conf. Mobile Netw. Manage.*, 2016, pp. 281–292.

[18] P. Jayashree and F. I. Princy, "Leveraging SDN to conserve energy in WSN-an analysis," in *Proc. 3rd Int. Conf. Signal Process., Commun. Netw. (ICSCN)*, 2015, pp. 1–6.

[19] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2015.

[20] A. Aissioui, A. Ksentini, A. M. Gueroui, and T. Taleb, "Toward elastic distributed SDN/NFV controller for 5G mobile cloud management systems," *IEEE Access*, vol. 3, pp. 2055–2064, 2015.

[21] N. Bizanis and F. A. Kuipers, "SDN and virtualization solutions for the Internet of Things: A survey," *IEEE Access*, vol. 4, pp. 5591–5606, 2016.

[22] D. Zeng, P. Li, S. Guo, T. Miyazaki, J. Hu, and Y. Xiang, "Energy minimization in multi-task software-defined sensor networks," *IEEE Trans. Comput.*, vol. 64, no. 11, pp. 3128–3139, Nov. 2015.

[23] Y. Wang, H. Chen, X. 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.

[24] H. P. Yao, "A multicontroller load balancing approach in software-defined wireless networks," *Int. J. Distrib. Sensor Netw.*, vol. 15, p. 10, Jan. 2015.

[25] R. Soua and P. Minet, "A survey on multichannel assignment protocols in Wireless Sensor Networks," in *Proc. Wireless Days IEEE*, Oct. 2011, pp. 1–3.

[26] R. Soua and P. Minet, "Multichannel assignment protocols in wireless sensor networks: A comprehensive survey," *Pervasive Mobile Comput.*, vol. 16, pp. 2–21, Jan. 2015.

[27] S. Shamshirband, "Survey An appraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique," *Eng. Appl. Artif. Intell.*, vol. 26, no. 9, pp. 2105–2127, 2013.

[28] F. I. Akyildiz, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Comput. Netw.*, vol. 50, no. 13, pp. 2127–2159, 2006.

[29] M. H. Rehmani, "SURF: A distributed channel selection strategy for data dissemination in multi-hop cognitive radio networks," *Comput. Commun.*, vol. 36, nos. 10–11, pp. 1172–1185, 2013.

[30] S. H. R. Bukhari, S. Siraj, and M. H. Rehmani, "PRACB: A novel channel bonding algorithm for cognitive radio sensor networks," *IEEE Access*, vol. 4, pp. 6950–6963, 2016.

[31] A. N. Khan, "BSS: Block-based sharing scheme for secure data storage services in mobile cloud environment," *J. Supercomput.*, vol. 70, no. 2, pp. 946–976, 2014.

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

[33] S. Sezer *et al.*, "Are we ready for SDN? Implementation challenges for software-defined networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 36–43, Jul. 2013.

[34] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[35] A. De Gante, M. Aslan, and A. Matrawy, "Smart wireless sensor network management based on software-dened networking," in *Proc. 27th Biennial Symp. Commun.*, Jun. 2014, pp. 71–75.

[36] Z. Han and W. Ren, "A novel wireless sensor networks structure based on the SDN," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 3, pp. 1–8, 2014.

[37] R. Huang, X. Chu, J. Zhang, and Y. H. Hu, "Energy-efficient monitoring in software defined wireless sensor networks using reinforcement learning: A prototype," in *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 10, pp. 1–13, 2015.

[38] D. Zeng, T. Miyazaki, S. Guo, T. Tsukahara, J. Kitamichi, and T. Hayashi, "Evolution of software-defined sensor networks," in *Proc. IEEE 9th Int. Conf. Mobile Ad-Hoc Sensor Netw.*, Dec. 2013, pp. 410–413.

[39] S. Costanzo, L. Galluccio, G. Morabito, and S. Palazzo, "Software defined wireless networks: Unbridling SDNs," in *Proc. Eur. Workshop Softw. Defined Netw. (EWSDN)*, Oct. 2012, pp. 1–6.

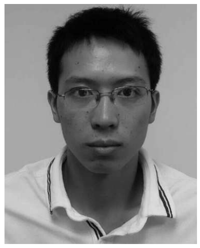
[40] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for WIreless SEnsor networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr./May 2015, pp. 513–521.

[41] M. Xu, Q. Yang, and K. S. Kwak, "Distributed topology control with lifetime extension based on non-cooperative game for wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 9, pp. 3332–3342, Sep. 2016.

[42] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 1195–1206, May 2005.
[43] Z. Han, "Game theory in wireless and communication networks," in *Game Theory in Wireless and Communication Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[44] R. S. Komali, A. B. MacKenzie, and R. P. Gilles, "Effect of selfish node behavior on efficient topology design," *IEEE Trans. Mobile Comput.*, vol. 7, no. 9, pp. 1057–1070, Sep. 2008.
[45] D. Fan and P. Shi, "Improvement of Dijkstra's algorithm and its application in route planning," in *Proc. 7th Int. Conf. FSKD*, vol. 4. Aug. 2010, pp. 1901–1904.
[46] M. Omari and S. Laroui, "Simulation, comparison and analysis of wireless sensor networks protocols: LEACH, LEACH-C, LEACH-1R, and HEED," in *Proc. 4th Int. Conf. Electr. Eng. (ICEE)*, 2015, pp. 1–5.
[47] F. Yu, Y. Li, F. Fang, and Q. Chen, "A new TORA-based energy aware routing protocol in mobile Ad Hoc networks," in *Proc. 3rd IEEE/IFIP Int. Conf. Central Asia Internet (ICI)*, Sep. 2007, pp. 1–4.
[48] B. C. Villaverde, R. De Paz Alberola, S. Rea, and D. Pesch, "Experimental evaluation of beacon scheduling mechanisms for multihop IEEE 802.15.4 wireless sensor networks," in *Proc. 4th Int. Conf. Sensor Technol. Appl. (SENSORCOMM)*, 2010, pp. 226–231.
[49] E. F. Nakamura and E. L. Souza, "Towards a flexible eventdetection model for wireless sensor networks," in *Proc. 15th IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2010, pp. 459–462.
[50] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

LI PEIZHE was born in 1990. He received the bachelor's degree in communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2013, where he is currently pursuing the Ph.D. degree. His research interests focus on smart grid wireless communication network and mobile ad hoc network.

WU MUQING was born in 1963. He received the Ph.D. degree. He is currently a Professor with the Beijing University of Posts and Telecommunications and a Senior Member of the China Institute of Communications. His current research interests focus on mobile ad hoc network, UWB, high-speed network traffic control and performance analysis, and GPS locating and services.



LIAO WENXING was born in 1990. He received the bachelor's degree in communications engineering from Jilin University, Jilin, China, in 2013. He is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications. His research interests focus on mobile ad hoc network and embedded development.

ZHAO MIN received the Ph.D. degree in information and telecommunication systems from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2014. She is currently a Lecturer with the Laboratory of Network System Architecture and Convergence, BUPT. Her research interests are in the areas of wireless communication systems.