# A Delay and Load-balancing based Hierarchical Route Planning Method for Transmission Line IoT Sensing and Monitoring applications


Cheng Ding
*State Key Laboratory of Networking and Switching Technology*
*Beijing University of Posts and Telecommunications, Beijing, 100086, China*
Beijing, China
470123956@qq.com

Siya Xu
*State Key Laboratory of Networking and Switching Technology*
*Beijing University of Posts and Telecommunications, Beijing, 100086, China*
Beijing, China
xusiyaxsy@hotmail.com

Xingyu Chen
*State Key Laboratory of Networking and Switching Technology*
*Beijing University of Posts and Telecommunications, Beijing, 100086, China*
Beijing, China
chenxy@bupt.edu.cn

Guiping Zhou
*State Grid Liaoning Electric Power Co., Ltd. Shenyang,110006 China*
Liaoning, China
18900911559@163.com

Peihao Zheng
*Beijing Guodiantong Network Technical Co. LTd., Beijing,100070 China*
Beijing, China
zhengpeihao@sgitg.sgcc.com.cn

Yueyue Li
*Shenyang Electric Power Supply Company, State Grid Liaoning Electric Power Company*
Liaoning, China
914974069@qq.com



***Abstract*—The Internet of Things (IoT) have been widely applied into human social activities for a long time. Wireless sensor networks (WSN) are an important part of IoT. WSN consist of a large number of inexpensive micro-sensor nodes deployed in the monitoring area. These nodes form a multi-hop self-organizing network system by wireless communication. To better utilize the limited energy resource, we design a delay and load-balancing based hierarchical route planning method for transmission line IoT sensing and monitoring applications (DLHRP) in WSN. The Algorithm has two sub-mechanisms which are the node clustering strategy and the data forwarding method. Firstly, we propose a node clustering strategy based on energy balancing and energy consumption optimization. It preferentially selects nodes with high residual energy and high traffic load as cluster-head nodes, and then distributes the cluster-head nodes evenly. Next, we present a data forwarding method based on load-balancing to differentiate services and select the appropriate routes for different services according to service priority and delay requirements. The simulation results show that the proposed route planning method can lower the total energy consumption of data transmission and thus extend the life cycle of WSN under the premise of ensuring the delay requirements.**



***Keywords—IoT sensing and monitoring, wireless sensor networks, transmission lines, load-balancing, service priority***


## I. INTRODUCTION

IoT technology is widely applied into the power industry, such as online transmission line monitoring, intelligent substations, and intelligent computer rooms. IoT sensor devices will meet requirements of applications such as smart home, body/health monitoring, environmental monitoring, etc. [1-3]. In the power grid, transmission lines are the power link with the largest assets, the widest distribution, the complicated operating environment of the equipment and the large influence of external forces. Due to equipment safety problems of transmission lines, such as excessive operating temperature of the wires, sag changes, wind partial discharge, breeze vibration, tower tilt, etc., it is urgent to install effective sensing equipment on transmission lines. The fault online monitoring system transmits various equipment parameters to the data processing center through IoT gateways, and then uses advanced data analysis technology to process various types of device data, and finally determines whether the transmission line is faulty. It has become a basis of decisions for grid safety [4-5]. Although people try to use cloud platform to enhance IoT devices, there are still a difficult problem about how to reduce network congestion and network delay [6]. Wireless sensor network (WSN) is flexible, autonomous and energy efficient. It can be a prominent choice for gathering IoT information [7]. Therefore, optimizing the resource allocation scheduling mechanism in WSN of transmission lines is very important. Fig. 1 shows the structure of sensor networks for transmission line IoT sensing and monitoring.

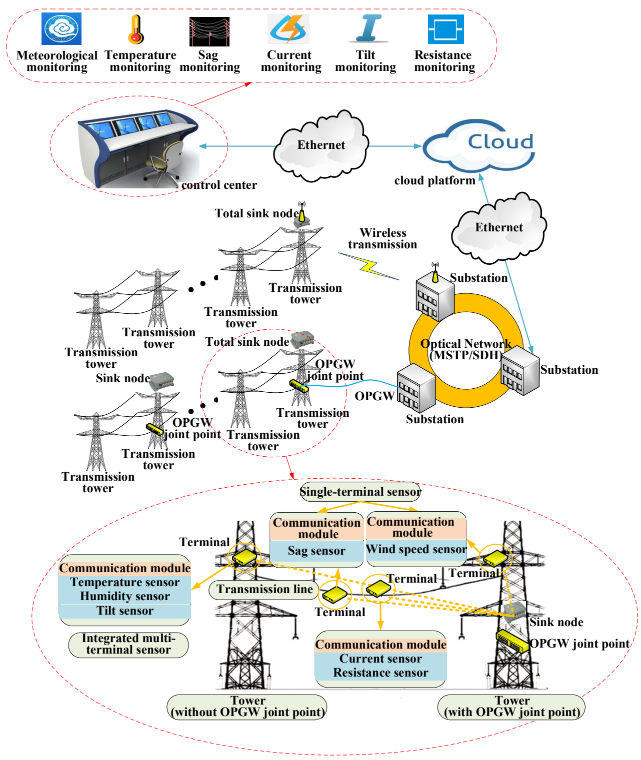


Fig. 1. IoT sensing and monitoring sensor network of transmission lines

In Fig. 1, the sensor networks for transmission line IoT sensing and monitoring are mainly composed of sensors, transmission towers, substations, Ethernet, cloud platforms and control centers. The sensors are mainly arranged on transmission lines and transmission towers for collecting information such as temperature, current, etc., and use ZigBee technology to transmit collected information to the nearest sink node by multi-hop transmission. Sensor terminals in IoT are mainly divided into two categories: single-terminal sensors and integrated multi-terminal sensors. The single-terminal sensors are used to collect a kind of service information, and the integrated multi-terminal sensors integrate multiple sensing devices on one sensor terminal in order to collect various monitoring information. In fiber-optic cable transmission scheme, the tower is mainly divided into two types: the tower with OPGW joint point and the tower without OPGW joint point. The sink nodes are generally arranged on the tower with OPGW joint point, and each sink node transmits collected information to the total sink node through optical fiber. Then, the total sink node sends collected information to substation through OPGW cable. In the wireless transmission scheme, the sink nodes are generally located on the transmission tower adjacent to substation. Each sensor node transmits collected information to the sink node by wireless multi-hop transmission. Then, the sink node uses a large-capacity wireless point-to-point long-distance communication device to transmit convergence information to substation by a wireless private network or a public network. The substations share sensing information through the MSTP/SDH optical network, and each substation uploads monitoring information to the cloud platform through Ethernet. The cloud platform aggregates and processes sensing information and sends it to the control center through Ethernet. Finally, the control center displays information about transmission lines.

This paper focuses on the routing planning algorithm at the phase where sensors on transmission lines use wireless communication to send information to the sink node. In transmission lines scene, positions of sensors are fixed, the nodes are distributed in a long chain and the types of monitoring services are diverse. In this paper, we will propose a delay and load-balancing based hierarchical route planning method (DLHRP). It optimizes two aspects of routing planning in WSN. In terms of clustering, we introduce a node clustering strategy based on energy balancing and energy consumption optimization which optimizes the selection and distribution of cluster-head nodes. In terms of route planning, we present a data forwarding method based on load-balancing which optimizes the energy consumption of data transmission and balances the communication load between nodes. Our main contributions are as follows:

- A clustering mechanism is proposed to plan the entire sensor network topology. This mechanism regards the residual energy and the communication load of nodes in the selection of cluster-head nodes. The node with higher residual energy and higher communication load is more likely to be the cluster-head node. In the distribution of cluster-head nodes, the mechanism sets the optimal distance range of adjacent cluster-head nodes according to the entire network size and node distribution. Based on this, the cluster-head nodes are more evenly distributed by adding and deleting cluster-head nodes.
- A data forwarding mechanism is proposed to plan the data transmission of the entire network. The cluster-head nodes send data according to the mechanism. Firstly, the cluster-head nodes sort the different data packets by priority according to the delay requirement and then forward different data packets in the manner consistent with the entire network load balancing. Finally, the life cycle of the entire sensor network is extended.

The rest of this paper is organized as follows. The second section introduces the related work on clustering algorithms and route planning algorithms. The third section introduces the system model of DLHRP. The fourth section introduces the algorithm design of DLHRP. The fifth section introduces the simulation and evaluation of DLHRP. The sixth section summarizes the paper.

## II. RELATED WORK

The past important work in hierarchical route planning algorithms can be divided into two parts: clustering and route planning improvement.

### *A. Clustering*

In hierarchical routing algorithms, some nodes will be selected as cluster-head nodes according to certain rules. These cluster-head nodes will collect and forward data packets of intra-cluster nodes and adjacent cluster-head nodes. Such algorithms have attracted lots of attention for the advantages of high adaptability, energy efficiency, robustness, and scalability. The Low Energy Adaptive Clustering Hierarchy protocol (LEACH) randomly selects cluster-head nodes, therefore the quality of cluster-head nodes cannot be guaranteed [8]. In order to improve the quality of cluster generation, Heinzelman et al. proposed a centralized adaptive topology control algorithm (LEACH-C) [9]. Firstly, it arranges the base station to collect position and residual energy information of all nodes, and then selects the nodes whose residual energy is not lower than the average value as the candidate nodes, and finally chooses the cluster-head nodes among the candidate nodes by using the simulated annealing algorithm. Chiwewe et al. proposed a distributed energy balance based topology control algorithm [10]. It enables each node to make local decisions about its transmit power, resulting in a network topology that maintains global connectivity. The algorithm can effectively reduce the energy consumption of information interaction and data transmission. Parul et al. proposed an algorithm based on K-medoids clustering (K-LEACH) [11]. It selects the node located in the K-medoids clustering center as the cluster-head node. Beiranvand et al. proposed a new cluster-head node selection mechanism (I-LEACH) [12]. The protocol improves the cluster-head node selection formula of the LEACH protocol. The selection of cluster-head node takes the distance, the number of surrounding nodes and the residual energy into account. Based on the distance between the cluster-head node and the sink node, Chien et al. proposed a new clustering algorithm [13]. In this algorithm, the size of the cluster is affected by the distance from the sink node. After the cluster is generated, the optimal cluster-head node is selected according to the energy of nodes and the number of adjacent nodes in the cluster. The algorithm equalizes the energy consumption of the data transmitted by the cluster-head nodes and extends the life cycle of the entire network.

## B. Route planning

Currently, most inter-cluster routing planning algorithms are not mature. So we have researched the traditional routing planning algorithm.

For the energy-aware routing problem in WSN, Giang et al. proposed a new WSN route planning algorithm to minimize node energy consumption and make rational use of the available resources of the node. The algorithm uses the properties of the convex function to simulate the nonlinear programming problem, and uses the sub-gradient algorithm to solve the problem, and finally achieves the purpose of extending the network life cycle [14]. Minhas et al. proposed a fuzzy algorithm to solve the maximum life cycle problem in wireless sensor networks. The algorithm uses fuzzy functions. This algorithm is mainly divided into two parts, namely fuzzy maximum life cycle algorithm and fuzzy multi-objective algorithm. The former is used to maximize the WSN life cycle, while the latter is used to optimize the life cycle and energy consumption [15]. In response to the integration of heterogeneous nodes and device in the IoT framework, Oteafy et al. proposed a Pruning Adaptive Internet of Things Routing (PAIR) protocol, which selectively established communication routes between IoT nodes to reduce communication overhead [16]. In response to the unacceptable QoS of delay sensitive information in the field of smart devices network, Sharma et al. proposed a scheduling scheme based on dynamic Markov chain. Traffic in the IoT is divided into priority and non-priority to ensure QoS for delay-sensitive traffic [17].

Most hierarchical routing algorithms focus on clustering, and cluster-head nodes generally send aggregated data directly to the sink node. This causes high energy consumption of cluster-head nodes. In addition, the intelligent algorithm-based clustering method can make the cluster-head nodes evenly distributed, but these algorithms converge too slowly. In other clustering methods, the choice of cluster-head nodes is usually less than ideal. To address the problems above, we propose a delay and load-balancing based hierarchical route planning method. The method makes the entire network evenly clustered in the clustering phase. Then, it balances the entire network load as much as possible while satisfying the service delay requirements in the route planning phase. In the end, the goal of extending the network life cycle is achieved.

# III. System Model

## A. Network model

### 1) Network architecture

The topology diagram of transmission lines is shown in Fig. 2. Sensors transmit data packets to the sink node by ZigBee technology. After receiving the data packets of different services, the cluster-head node sends packets through different paths under the requirements of services delay. After receiving the data packets, the sink node can transmit data to the substation via private wired networks (such as OPGW), private wireless networks (such as LoRa), or public wireless networks (such as NB-IoT).

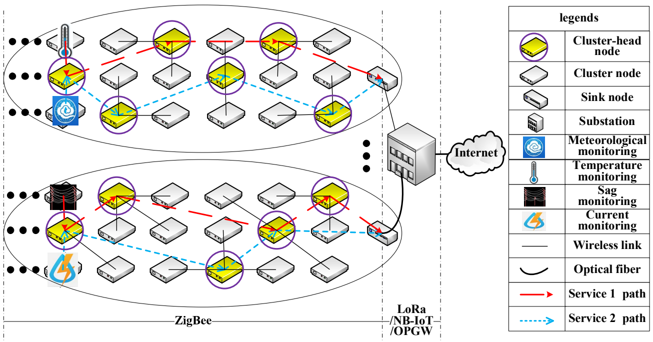


Fig. 2. WSN topological structure diagram of DLHRP algorithm in long-chain transmission lines

The sensor topology of transmission lines is link structure. Link structure is a special structure with low complexity. It is known that each transmission line has three phases, and sensors are deployed on those three-phase lines. We deploy a hierarchical network model as shown in Fig. 2 and make the following assumptions: 1. All sensors are stationary. Although the monitoring services are different, data packets capacity and transmission frequency are basically the same. When the emergency is detected, the related data packet will be sent immediately, and be assigned a higher priority. 2. All sensors can adjust their transmission power adaptively according to the communication distance. 3. The link is symmetrical. The radio signal has the same energy attenuation in all directions. 4. All sensors can be selected as cluster-head nodes.

### 2) Energy model

When $t$-bit data is transmitted between two nodes with distance $d$, the energy consumption function of the sender is [18]:

$$E_{consum}(t,d) = t \cdot (E_{elec} + \varepsilon \cdot d^{index}) \tag{1}$$

$$index = \begin{cases} 2 & d < d_0 \\ 4 & d \geq d_0 \end{cases} ; \quad \varepsilon = \begin{cases} \varepsilon_{fs} & d < d_0 \\ \varepsilon_{mp} & d \geq d_0 \end{cases}$$

Where $E_{elec}$ represents the energy loss of the transmitting circuit, $d$ is the Euclidean distance between two nodes, $d_0$ is the distance threshold, $\varepsilon_{fs}$ and $\varepsilon_{mp}$ are the energy required for power amplification in two models, respectively.

The energy consumed of receiving $t$-bit data is [19]:

$$E_r(t) = t \cdot E_{elec} \tag{2}$$

## B. System model

### 1) The optimal number of cluster-head nodes

Assume that all nodes have the same initial energy and data transmission capability. We deployed $N$ sensors evenly on the three-phase line. The area size is set to be $L \cdot W$ . The sink node is arranged on one side of the transmission line. In each clustering cycle, each cluster node sends $t$-bit data to the related cluster-head node. According to the energy consumption function in (1), the total energy consumption of the entire network in each clustering cycle can be expressed as [20]:

$$E_{round} = t(2NE_{elec} + M\varepsilon_{mp} d^4_{ave-sink} + N\varepsilon_{fs} d^2_{ave-ch}) \tag{3}$$

Where $d_{ave\text{-}sink}$ represents the average distance from cluster-head nodes to the sink node, $d_{ave\text{-}ch}$ represents the average distance from cluster nodes to the related cluster-head node, and $M$ is the number of cluster-head nodes.

Because all nodes are evenly distributed, (4) is given:

$$L \cdot W = 2\pi d_{ave-ch}^2 \cdot M \quad (4)$$

$$d_{ave-ch} = \frac{\sqrt{L \cdot W}}{\sqrt{2\pi M}} \quad (5)$$

Taking (5) into (3) and deriving $E_{round}$ with $M$ as an independent variable and making the derivative of $E_{round}$ equal to 0, we obtain the optimal number of cluster-head nodes by:

$$M = \sqrt{\frac{N}{2\pi}} \cdot \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \cdot \frac{\sqrt{L \cdot W}}{d_{ave\text{-}sink}^2} \quad (6)$$

*2) The optimal rotation period of becoming a cluster-head node for each node*

In hierarchical routing algorithms, the nodes selected as cluster-head consume more energy than cluster nodes due to forwarding data packets of other nodes. Therefore, the rotation of cluster-head nodes is crucial to balance nodes energy consumption in the entire network. We assume $T_{ro}$ is the rotation period of each cluster-head node. Which means, after a certain node is selected as a cluster-head node with probability $p$ at the beginning, it must pass at least $T_{ro}$ clustering cycles before it can be re-selected as a cluster-head node again with probability $p$, so $T_{ro}=1/p$.

We assume that the probability of the $i$-th node becoming a cluster-head node in the $r$-th clustering cycle is:

$$p_i(r) = p_{ave}\left[1 - \frac{\overline{E}(r) - E_i(r)}{\overline{E}(r)}\right] = p_{ave}\frac{E_i(r)}{\overline{E}(r)} \quad (7)$$

Where $p_{ave}$ is the probability for each node being a cluster-head node at the ideal state, $E_i(r)$ is the residual energy of the $r$-th clustering cycle of the $i$-th node, and $\overline{E}(r)$ is the average residual energy of all nodes in the $r$-th clustering cycle. $\overline{E}(r)$ is as:

$$\overline{E}(r) = \frac{1}{N}E_{total}(1 - \frac{r}{R}) \quad (8)$$

Where $R$ is the life cycle of the entire network under ideal conditions, which can be expressed by:

$$R = \frac{E_{total}}{E_{round}} \quad (9)$$

In the $r$-th round, the rotation period of the $i$-th node can be expressed by:

$$T_{ro-i}(r) = \frac{1}{p_i(r)} = \frac{\overline{E}(r)}{p_{ave}E_i(r)} = \frac{E_{total} - rE_{round}}{Np_{ave}E_i(r)} \quad (10)$$

Where $E_{round}$ is given by (3) and $E_{total}$ is the initial total energy of the entire sensor network. $E_{total}$ can be expressed as:

$$E_{total} = \sum_{n=1}^{N} E_{init} \quad (11)$$

$E_{init}$ is the initial residual energy of each node. And all nodes have the same value of $E_{init}$.

*3) Selecting cluster-head nodes*

The process of selecting cluster-head nodes consists of two phases. These two phases perform pre-cluster-head node selection and cluster-head node adjustment respectively. The pre-cluster-head nodes are selected in the first phase, and then, the distribution of cluster-head nodes is optimized in the second phase.

The pre-cluster-head node is selected as follows: a node generates a random number between 0 and 1. If the random number is less than the threshold $T(n)$, the node is selected as the pre-cluster-head node in the current clustering cycle. $T(n)$ is:

$$T(n) = \frac{p_{ave}}{1 - p_{ave}(r \bmod \frac{1}{p_{ave}})} \cdot \frac{E_{remain}(n)}{\overline{E}_{remain}} \cdot \frac{E_{r-total}(n,t)}{\overline{E}_{r-total}} \qquad n \in G \quad (12)$$

Where $G$ is the nodes set non-cluster-head nodes in the optimal rotation period. If node n is not in set $G$, then $T(n)$=0. $r$ is the current clustering cycle.

The average residual energy of a node is:

$$\overline{E}_{remain} = \frac{1}{N}\sum_{n=1}^{N} E_{remain}(n) \quad (13)$$

We assume that a node has $k$ adjacent nodes within a distance of $d_{max}$, and each adjacent node sends a $t$-bit packet, so the energy consumption of receiving corresponding data packet is $E_r(t) = t \cdot E_{elec}$ . For the central node $n$, the total energy consumption of receiving adjacent data packets is:

$$E_{r-total}(n,t) = \sum_{i=1}^{K} E_r(t) \quad (14)$$

The average energy consumption of each node receiving its adjacent node information is：

$$\overline{E}_{r-total} = \frac{1}{N}\sum_{n=1}^{N} E_{r-total}(n,t) \quad (15)$$

We assume that the communication distance from a cluster node to a cluster-head node satisfies:

$$\pi d_{ch}^2 \cdot M \geq L \cdot M \quad (16)$$

So, we have:

$$d_{ch} \geq \sqrt{\frac{L \cdot W}{\pi \cdot M}} \quad (17)$$

Considering the characteristics of the scene, we introduce factors $L/Lm$ and $W/Wm$ to satisfy:

$$d_{max} = \sqrt{\frac{L \cdot W}{\pi \cdot M} \cdot \frac{L}{Lm} \cdot \frac{W}{Wm}} \quad (18)$$

Where $L$ is the length of the scene, $Lm$ is the lateral spacing of sensors placement, $W$ is the width of the scene, $Wm$ is the longitudinal spacing of sensors placement, and $M$ is the number of cluster-head nodes.

Assume that $d_{max}$ is the optimal communication distance from a cluster node to a cluster-head node. $D_{max}$ is the communication range of sending data packets for a cluster-head node. The relationship between $D_{max}$ and $d_{max}$ is:

$$d_{max} = \frac{1}{2} D_{max} \tag{19}$$

In the second phase, each pre-cluster-head node broadcasts information selected a cluster-head node. If the distance between two pre-cluster-head nodes is less than $D_{min}$, the node with smaller weight will be demoted to a cluster node. $D_{min}$ is:

$$D_{min} = \frac{1}{2} d_{max} = \frac{1}{4} D_{max} \tag{20}$$

The weight $w(n)$ of node $n$ is:

$$\omega(n) = \alpha \ln \frac{E_{remain}(n)}{\overline{E}_{remain}} + \beta \ln \frac{E_{r-total}(n)}{\overline{E}_{r-total}} + \chi \ln \frac{d_{ave\text{-}sink}}{d_{sink}(n)} \tag{21}$$

Where $\alpha$, $\beta$ and $\chi$ are the coefficients of the three factors, respectively.

If there is no cluster-head node within the $d_{max}$ range, a node will be updated to an unclassified node. After that, new cluster-head nodes are re-selected among these unclassified nodes and the re-selected probability is:

$$p'(n) = \frac{M'}{N'} \cdot \frac{E_{remain}(n)}{\overline{E}_{remain}'} \cdot \frac{E_{r-total}(n,t)'}{\overline{E}_{r-total}'} \tag{22}$$

Where $M'$ is the difference between the desired number of cluster-head nodes $M$ and the number of actual cluster-head nodes. $N'$ is the number of unclassified nodes. Other symbols are the corresponding symbols of unclassified nodes.

#### *4) Delay constraint of data transmission*

We assume that there are multiple services in the entire network. The $k$-th service is labeled as $S_k$. Its delay requirement is $D_k$. We assume that the processing delay is constant for every cluster-head node to process a kind of service data packet, and the queuing delay is also constant. Compared to queuing delay and processing delay, the delay of data transmission in the channel can be ignored. So, we set $D_{ave}=D_{queue}+D_{process}+D_{transmission}$ to a fixed value in this article. For a packet, $D_{queue}$, $D_{process}$ and $D_{transmission}$ represent its queuing delay, processing delay, and transmission delay, respectively. Therefore, we convert the delay constraint into the hop constraint:

$$Hop_k = \frac{D_k}{D_{ave}} \tag{23}$$

## IV. ALGORITHM DESIGN

This section describes DLHRP algorithm. This algorithm includes two mechanisms: a uniform clustering mechanism based on energy balance and optimal energy consumption and a data forwarding mechanism on measuring service priorities.

### *A. A uniform clustering mechanism based on energy balance and optimal energy consumption*

We propose a uniform clustering mechanism based on energy balance and optimal energy consumption. First, according to the attenuation power formula, $d_{max}$, $D_{max}$ and $D_{min}$ are determined. Second, the network is clustered according to (12) and pre-cluster-head nodes are selected. Third, if the distance between pre-cluster-head nodes is less than $D_{min}$, the node with smaller weight will demote to be a cluster node. Fourth, if the actual number of cluster-head nodes is less than the expected number, all ordinary cluster nodes will connect to their nearest cluster-head node within the communication range and the remaining nodes are updated to unclassified nodes. Otherwise, if there are no unclassified nodes or the number of cluster-head nodes is greater than or equal to the expected value, all cluster nodes connect to their nearest cluster-head node and the clustering phase ends. Fifth, unclassified nodes are re-selected cluster-head nodes according to (22) and the algorithm return to the fourth step. The detailed process is shown in Algorithm 1.

**Algorithm 1** A uniform clustering mechanism based on energy balance and optimal energy consumption

**Input:** Minimum communication distance between cluster-head nodes $D_{min}$, optimal communication distance from cluster node to cluster-head node $d_{max}$, maximum communication distance between cluster-head nodes $D_{max}$;
**Output:** Number of cluster-head nodes $M'$, cluster-head node set $C=(E_{remain}, P)$, cluster node set connecting the cluster-head node $G^H=(E_{remain}, P)$;
**Initialization:** Number of nodes $N$, node set $G^C=(E_{remain}, P)$, optimal number of cluster-head nodes $M$, total energy consumption of receiving adjacent node data $E_{r-total}$, the distance from node n to the sink node $d_{sink}(n)$, cluster-head node set $C=\varnothing$, number of cluster-head nodes $M'=0$;

**for** each node $n$ **do**
  generate a random number $p\in(0\sim1)$, and calculate the threshold $T(n)$ by (12);
  **if** $p<T(n)$ **then**
    $M'=M'+1$;
    $n$ join $C$;
    calculate the weight $\omega(n)$ by (21);
  **end if**
**end for**
**for** each cluster-head node $n$ **do**
  **for** each cluster-head node $m$ **do**
    **if** $m$ != $n$ and the distance between $m$ and $n$ is less than $D_{min}$ **then**
      $M'=M'-1$;
      **if** $\omega(n)>\omega(m)$ **then**
        remove $m$ from $C$;
      **else**
        remove $n$ from $C$;
      **end if**
    **end if**
  **end for**
**end for**
**if** $M'<M$ **then**
  **for** each node $n$ **do**
    **for** each cluster-head node $c$ **do**
      **if** the distance between $n$ and $c$ is less than or equal to $d_{max}$ and $c$ is the nearest cluster-head node **then**
        $n$ join $G^H(c)$;
      **end if**
    **end for**
  **end for**
  **for** each node without cluster-head node **do**

```
30:         reselect the cluster-head node by calculating the
            probability p'(n) by (22);
31:         if n join C then
32:             M'=M'+1 ;
33:         end if
34:     end for
35:     go to line 21;
36: else
37:     each node connects its nearest cluster-head node;
38: end if
```

## B. A service priority based data forwarding mechanism

We propose a service priority based data forwarding mechanism. In each round of data transmission, the cluster-head nodes receive data packets from the nodes in their cluster and the adjacent cluster-head nodes. The received packets are sorted according to the service priority. Each packet carries a timestamp and hop limit *hop*. Each time a data packet is forwarded, the hop limit *hop*-1. In the data forwarding process, the cluster-head nodes send data packets with *hop*=1 to the sink node. After that, the cluster-head nodes regard these packets based on the service priority. For a cluster-head node, if there is no path that satisfies the packet delay requirements, the next hop cluster-head node of the shortest path is selected to forward the packet and then the selected cluster-head node is marked. Otherwise, selected cluster-head node tends to satisfy the following conditions in order: 1. The path satisfies the delay requirement. 2. The cluster-head node is not marked. 3. The path is the shortest in unmarked nodes. The detailed process is shown in Algorithm 2.

**Algorithm** 2 A service priority based data forwarding mechanism

**Input:** Packet hop limit *hop*, cluster-head node set *C*, unlabeled cluster-head node set $C^*$, data set regarded *B*, data set to be forwarded $B^*$;
**Output:** Packet hop limit *hop-1*, $C^*=C$;
**Initialization:** $C^*=C$, $B^*=\varnothing$;

```
1:  for arrived data set B in each timeslot do
2:      sort B in descending order by its service priority;
3:      regard the first packet in set B;
4:      if hop=1 then
5:          the packet is sent to the nearest sink node;
6:          update set B;
7:          go to line 26;
8:      else
9:          calculate the shortest path from the cluster-head
            node to the destination node in C* by applying
            the Dijkstra algorithm;
10:         if the path meets hop limit then
11:             send data to the next hop cluster-head node in
                the path;
12:             mark the next hop cluster-head node;
13:             update C*, B;
14:             go to line 26;
15:         else if C*=C then
16:             go to line 11;
17:         else
18:             the packet belongs to set B*, update B;
19:             if B = ∅ then
20:                 B = B*;
21:                 B* = ∅;
22:                 C*=C;
23:             end if
24:             go to line 2;
25:         end if
26:     end if
27:     if B = ∅ and B* = ∅ then
28:         return packet hop limit hop-1, C*=C;
29:     else
30:         go to line 19;
31:     end if
32: end for
```

# V. Simulation and analysis

## A. Simulation settings

In order to prove the effectiveness of the proposed algorithm, we carry out simulation experiments in MATLAB. The parameter settings for the simulation experiment are shown in TABLE I:

TABLE I. SIMULATION PARAMETER SETTINGS

| parameter | value | parameter | value |
|---|---|---|---|
| Number of node location $N$ | 90 | Energy consumption of data transmission $E_0$ (J) | 2×10^(-8) |
| Estimated number of nodes $M$ | 90*0.9 | Energy consumption of sending $E_{consum}$ (J/bit×m) | 50 $E_0$ |
| Probability of selecting cluster-head node $p_{ave}$ | 0.1 | Energy consumption of receiving $E_r$ (J/bit) | 50 $E_0$ |
| Distribution area $A$ (m$^2$) | 2×3000 | Energy consumption of convergence (J/bit) | 5 $E_0$ |
| Cluster node communication radius $d_{max}$ (m) | 300 | Control information size (bit) | 32 |
| Cluster-head node communication radius $D_{max}$ (m) | 600 | Data information size (bit) | 4000 |
| Initial energy of a node $E_{init}$(J) | 1500 | Sink node location | (3000,1) |
| Higher priority service hop limit (hop) | 10 | Lower priority service hop limit (hop) | 15 |
| The lateral spacing of sensors placement *Lm* (m) | 100 | The longitudinal spacing of sensors placement *Wm* (m) | 1 |

## B. Analysis of simulation results

DLHRP algorithm simulation is mainly divided into two phases, which are the clustering phase and the routing phase. In the clustering phase, DLHRP algorithm is compared with I-LEACH algorithm. I-LEACH algorithm improves the cluster-head nodes selection mechanism of LEACH algorithm. The algorithm regards the number of adjacent nodes, residual energy and distance factors in the selection process of cluster-head nodes [12]. In the routing phase, DLHRP algorithm is compared with ant colony algorithm and CRLA algorithm. Ant colony algorithm is an intelligent algorithm which converges the planned path to the optimal path by leaving the pheromone. CRLA algorithm is an improved LEACH-C algorithm. CRLA combines single-hop routing and multi-hop routing in inter-cluster routing process. Whenever the distance of a data packet is smaller than the set threshold, the data packet will be directly sent to the sink node [21]. In order to ensure the reliability of the results, the simulation results are the mean of multiple experiments.

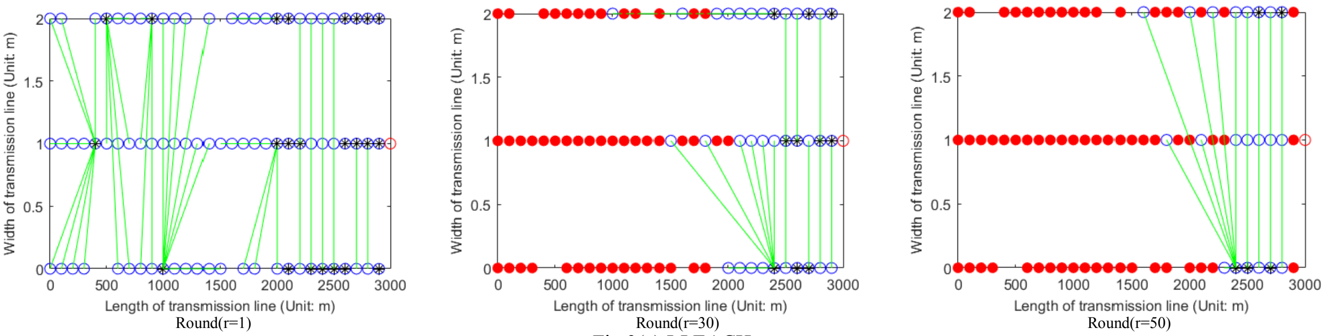


Fig 3(a) I-LEACH

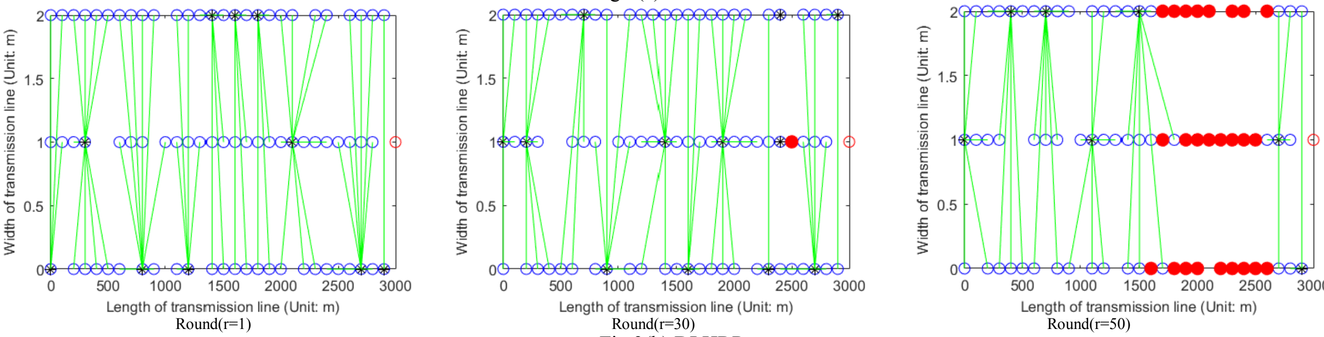


Fig 3(b) DLHRP

Fig. 3. The comparison of the death nodes state of two algorithms

*1) A uniform clustering mechanism based on energy balance*

In the paper, we call the nodes with 0 residual energy death nodes [12]. Fig. 3 shows the comparison of the death nodes of two algorithms on the transmission line. The nodes are represented by "o". The sink node is located at the far right end of the transmission line, and its coordinates are (3000, 1). The cluster nodes are arranged on the transmission line. The cluster-head nodes are represented by "*". The line segments indicate the connecting relationship between cluster nodes and cluster-head nodes. The solid nodes represent death nodes. In Fig.3, the node survival state of DLHRP algorithm is better than I-LEACH algorithm. Cluster-head nodes directly send data packets to the sink node and cluster-head nodes are unevenly distributed in I-LEACH algorithm, so the cluster-head nodes which are far away from the sink node die quickly and the energy consumption of the entire network is very large. However, cluster-head nodes are evenly distributed and cluster-head nodes send data packets to the sink node through multi-hop transmission in DLHRP algorithm, so the energy consumption of the entire network is relatively small.

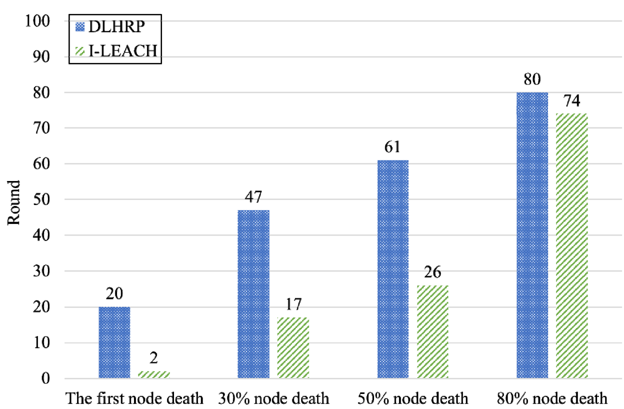


Fig. 4. The comparison of the nodes death degree

In order to compare the performance of those two algorithms more clearly, we use a column chart to represent the node death degree. In Fig. 4, the nodes death time of DLHRP algorithm is later than I-LEACH algorithm. Compared with I-LEACH algorithm, DLHRP algorithm effectively extends the network life time, improves network connectivity, and makes the network load more balances.

*2) A data forwarding mechanism based on service priority*

Based on DLHRP clustering algorithm, we simulate CRLA algorithm, ant colony algorithm and service differentiation algorithm respectively. Fig. 5 shows the characteristics of path planning using different algorithms and TABLE II records these paths. It can be seen that after a certain number of iterations, the planned path converges to a shorter path using ant colony algorithm. CRLA algorithm combines single-hop routing and multi-hop routing in inter-cluster routing process. Whenever the distance of a data packet is smaller than the set threshold, the data packet is directly sent to the sink node. For the algorithm based on service differentiation, the transmission paths of high priority services are shorter than those of low priority. The low priority services data packets tend to choose the paths that save energy and balance the load of the entire network.

TABLE II. THE PATH PLANNING RESULT INFORMATION

| Path planning algorithm | Path | Hop |
|---|---|---|
| Ant Colony | 2-35-66-70-17-49-25-sink | 7 |
| CRLA | 2-34-62-40-41-44-17-75-sink | 8 |
| Service Differentiation (Higher service priority) | 2-34-11-49-22-56-sink | 6 |
| Service Differentiation (Lower service priority) | 2-34-65-71-73-49-20-56-85-sink | 9 |

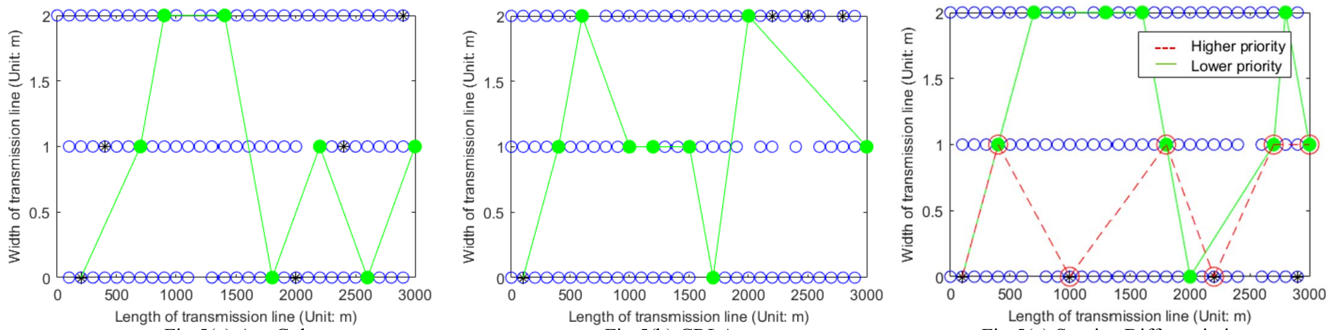

Fig 5(a) Ant Colony　Fig 5(b) CRLA　Fig 5(c) Service Differentiation

Fig. 5. Comparing Service Differentation with CRLA and Ant Colony

The node death degree is compared in Fig. 6.

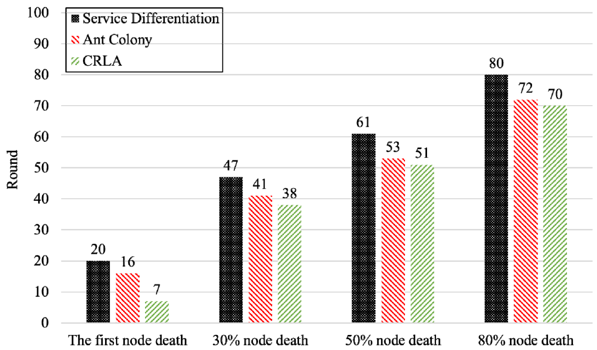

Fig. 6. The comparison of the node death degree

In Fig. 6, the node death time of the service differentiation algorithm is later than the other two routing algorithms. The service differentiation algorithm sends different packets to different cluster-head nodes, which effectively balances the energy consumption of each node. Therefore, nodes died more later.

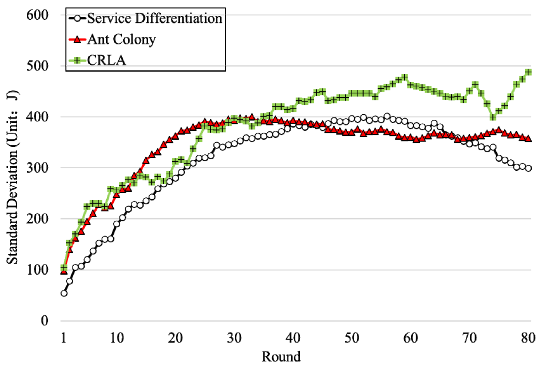

Fig. 7. The standard deviation of the nodes remaining energy

In order to see the balance degree of energy consumption in the entire network more clearly, Fig. 7 records the standard deviation of the nodes remaining energy in the entire network.

The remaining energy standard deviation of the service differentiation algorithm is the smallest from Fig. 7. The node energy consumption of CRLA algorithm is dependent on the selection of the transmission threshold. The nodes that send packets directly to the sink node have high energy consumption, so these nodes died quickly. In the other two algorithms, the service differentiation algorithm performs multi-path forwarding, therefore, nodes energy consumption is more balanced, so that the number of surviving of the nodes in each clustering cycle is optimized and eventually the entire network life cycle is extended.

## VI. CONCLUSION

In order to extend the life cycle of WSN, we propose a delay and load-balancing based hierarchical routing method for transmission line IoT sensing and monitoring applications (DLHRP). The algorithm improves two aspects of route planning in WSN. In the clustering phase, we propose a uniform clustering strategy based on energy balance and optimal energy consumption. It regards the residual energy and the communication load to select cluster-head nodes. And then, the distribution of cluster-head nodes is improved according to the distance factor. In the path planning phase, we propose a data forwarding method based on service priority. It allows a cluster-head node can send different data packets to different adjacent cluster-head nodes. The simulation results show that the proposed algorithm can balance the communication load of each cluster-head node, reduce the total energy consumption of the data transmission process, and ultimately extend the life cycle of the network.

In fact, the algorithm proposed in this paper is suitable for solving routing problems in multi-service scenarios in which the transmission services have the characteristics of small data amount and different data delay requirements. In the future work, we will further reduce the complexity of our algorithm according to the chained transmission line topology. We will also study a link backup mechanism to improve the reliability of the proposed algorithm.

## ACKNOWLEDGMENT

This work is supported by National Key R&D Program of China (2016YFB0901200).

## REFERENCES

[1] Trihinas, Demetris, G. Pallis, and M. D. Dikaiakos. "ADMin: Adaptive monitoring dissemination for the Internet of Things." INFOCOM 2017 - IEEE Conference on Computer Communications, 2017, pp. 1-9.

[2] Gyrard A, Datta S K, Bonnet C, K Boudaoud. "Standardizing generic cross-domain applications in Internet of Things." IEEE GLOBECOM Workshops, 2015, pp. 589-594.

[3] Mayzaud A, Sehgal A, Badonnel R, I Chrisment, J Schönwälder. "Using the RPL protocol for supporting passive monitoring in the Internet of Things." IEEE Network Operations and Management Symposium, 2016, pp. 366-374.

[4] Kong, Peng Yong, C. W. Liu, and J. A. Jiang. "Cost-Efficient Placement of Communication Connections for Transmission Line Monitoring." IEEE Transactions on Industrial Electronics 64.5(2017), pp. 4058-4067.

[5] Du L, Wang C, Li X, Yang L, Mi Y, et al. "A Novel Power Supply of Online Monitoring Systems for Power Transmission Lines." IEEE Transactions on Industrial Electronics 57.8(2010), pp. 2889-2895.

[6] Trihinas, Demetris, G. Pallis, and M. Dikaiakos. "Monitoring Elastically Adaptive Multi-Cloud Services." IEEE Transactions on Cloud Computing (2015), pp. 1-1.

[7] Zhu Q, Wang R, Chen Q, Liu Y, Qin W. "IOT Gateway: BridgingWireless Sensor Networks into Internet of Things." Ieee/ifip, IEEE International Conference on Embedded and Ubiquitous Computing, 2011, pp. 347-352.

[8] Heinzelman, W., A. Chandrakasan, and H. Balakrishnan. "Energy-Efficient Protocol for Wireless Microsensor Networks." Hawaii International Conference on System Sciences 2000, pp. 3005-3014.

[9] Heinzelman, W. B., A. P. Chandrakasan, and H. Balakrishnan. "An application-specific protocol architecture for wireless microsensor networks." IEEE Trans Wireless Commun 1.4(2002), pp. 660-670.

[10] Chiwewe, Tapiwa M., and G. P. Hancke. "A Distributed Topology Control Technique for Low Interference and Energy Efficiency in Wireless Sensor Networks." IEEE Transactions on Industrial Informatics 8.1(2012), pp. 11-19.

[11] Parul Bakaraniya#1, Sheetal Mehta. "K-LEACH: An improved LEACH Protocol for, Lifetime Improvement in WSN." International Journal of Engineering Trends & Technology 4.5(2013).

[12] Beiranvand, Zahra, A. Patooghy, and M. Fazeli. "I-LEACH: An efficient routing algorithm to improve performance & to reduce energy consumption in Wireless Sensor Networks." IEEE Information and Knowledge Technology, 2013, pp. 13-18.

[13] Chien, Chih Hsien, and M. S. Wang. "An Improving of Load Balancing in Clustering Algorithm for Wireless Sensor Network Based on Distance." IEEE International Conference on Mechanical Computer Society, 2017, pp. 165-168.

[14] Giang D V, Taleb T, Hashimoto K, Kato N, Nemoto Y. "A Fair and Lifetime-Maximum Routing Algorithm for Wireless Sensor Networks." (2007), pp. 581-585.

[15] Minhas, Mahmood R., S. Gopalakrishnan, and V. C. M. Leung. "Fuzzy Algorithms for Maximum Lifetime Routing in Wireless Sensor Networks." IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference, 2008, pp. 1-6.

[16] Oteafy, Sharief M. A., F. M. Al-Turjman, and H. S. Hassanein. "Pruned Adaptive Routing in the heterogeneous Internet of Things." IEEE Global Communications Conference, 2013, pp. 214-219.

[17] Sharma, R., and N. Kumar. "QoS-Alert Markov Chain Based Scheduling Scheme in Internet of Things." IEEE GLOBECOM Workshops, 2015, pp. 1-6.

[18] Sarma, Hiren Kumar Deva, P. Rai, and B. Deka. "Energy efficient communication protocol for wireless sensor networks with mobile node." IEEE Recent Advances and Innovations in Engineering, 2014, pp. 1-6.

[19] Paul, A K, and T. Sato. "Effective Data Gathering and Energy Efficient Communication Protocol in Wireless Sensor Network." Bibliogr (2010), pp. 1-5.

[20] Li, Qing, Q. Zhu, and M. Wang. "Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks." Computer Communications 29.12(2006), pp. 2230-2237.

[21] Yao, Fu. "Cluster Routing based on Low Energy Adaptive Clustering Hierarchy Centralized Algorithm in Wireless Senor Network." Information Security & Technology (2014).