A Clustering Protocol Using Multiple Chain Strategy in WSNs

Zheng Gengsheng, Hu Zhengbing
School of Computer Science & Engineering, Wuhan Institute of Technology, China
Department of Information Technology, Huazhong Normal University, China
Email: zhenggengsheng@sina.com

Abstract—In WSNs, energy efficiency and low latency are considered as two key issues in designing routing protocol. This paper introduces two schemes to combine clustering strategy with chain routing algorithm in order to satisfy both energy and delay constraints in WSNs. Furthermore by one scheme, we propose a two layer hierarchical routing protocol called Chain Routing Based on Coordinates-oriented Clustering Strategy (CRBCC), which gives a good compromise between energy consumption and delay. First, CRBCC makes balanced clustering according to y coordinates where each cluster has approximately equal number of nodes. Second, CRBCC makes chain routing by simulated annealing algorithm (SA) inside the cluster and elects chain leader in the order of x coordinates. Third, CRBCC makes chain routing again by SA method among chain leaders. Simulation results show that CRBCC performs better than PEGASIS in terms of energy efficiency and network delay.

Index Terms—WSNs; PEGASIS; CRBCC; simulated annealing algorithm

I. INTRODUCTION

The requirement for designing routing protocol in WSNs is different from the other network. The main constraint of WSNs is their very limited battery energy, which has great influence on the lifetime and the quality of the network ([1], [2], [3], [4], [5], [6]). On the other hand, transmission time is also another important factor to a routing protocol in the real-time environment. For that reason, the protocols running on sensor networks must consume the energy efficiently while transmit information in a low delay. There are many routing protocols designed for wireless sensor networks to satisfy energy efficiency or low latency requirement ([7], [8], [9], [10], [11], [12]).

Many algorithms have been proposed addressing the application and architecture requirements. Most of the protocols can be classified as either flat or hierarchical based. In hierarchical routing protocol, the network is divided into clusters with one cluster-head acting as sender to base station (BS). Data are gathered and fused at each cluster-head before transmission to the BS. Among the hierarchical protocols, Low-Energy Adaptive Clustering Hierarchy (LEACH) ([13], [14]) and Power Efficient Gathering in Sensor Information Systems (PEGASIS) [15] are very representative.

The rest of the paper is organized as follows: First we give an overview of the related works in section 2. Then the network and communication models of our proposal are discussed in section 3. Two scheme of clustering protocol using chain routing algorithm are introduced in section 4. A detail description of our approach CRBCC is presented in section 5. Simulation results make a comparative analysis between PEGASIS and CRBCC in section 6. Finally, section 7 presents a short conclusion with future work.

II. RELATED WORKS

Low-Energy Adaptive Clustering Hierarchy (LEACH) ([13], [14]) is a clustering protocol. LEACH reduces the number of nodes communicating directly with BS. The protocol achieves this by forming different clusters in a self-organizing manner, where each cluster-head collects the data from all nodes in its cluster, fuses and sends the information to BS. This decision is made by the node n choosing a random number between 0 and 1. If the number is less than a threshold T(n), the node n becomes a cluster-head for the current round.

\[
T(n) = \begin{cases} 
\frac{P}{1 - P \cdot r \mod(1/P)} , & \exists \in G \\
0, & \text{otherwise}
\end{cases} \tag{1}
\]

where P is the desired percentage of cluster-heads, r is the current round, and G is the set of nodes that have not been cluster-heads in the last 1/P rounds. Using this threshold, each node will be a cluster-head at some point within 1/P rounds. LEACH can distribute the energy among the nodes in the network and enhance system lifetime.

PEGASIS [15] is a chain-based protocol, which makes further improvement on LEACH. PEGASIS makes a chain routing through Greedy algorithm, which is also called the nearest neighbor algorithm. In the chain, each node receives from and transmits to the closest neighbor. The elected chain-leader is responsible for transmitting the final information to the base station.

Greedy chain algorithm begins at a farthest node from BS, which is the only node in the chain at first. Each terminal node of the chain finds a closest node from the remaining nodes set which are not in the chain. Then the closest node will join the chain and be the new terminal node of the chain. The process repeats till all nodes join.
the chain. The Greedy chain algorithm in PEGASIS is as follows. Here, \( N \) is the nodes set, which includes all nodes at first. END is the terminal node of the chain. Variable \( chain \) is the required chain to be constructed. The function \( \text{FindCloseNode}(N, \text{END}) \) means to find the closest node to \( \text{END} \) node in the remaining nodes which are still not in the chain. The function \( \text{Append}(\text{chain}, \text{END}) \) means to add the terminal node \( \text{END} \) into the chain \( \text{chain} \) tail.

![PEGASIS chain](image)

**Figure 1.** PEGASIS chain

<table>
<thead>
<tr>
<th>Procedure ConstructGreedyChain(N,END)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Begin</td>
</tr>
<tr>
<td>2. ( N={\text{all nodes}}; )</td>
</tr>
<tr>
<td>3. END = farthest node from BS;</td>
</tr>
<tr>
<td>4. chain = {END};</td>
</tr>
<tr>
<td>5. ( N=N-{\text{END}}; )</td>
</tr>
<tr>
<td>6. if(( N! =\text{NULL} ))</td>
</tr>
<tr>
<td>7. {</td>
</tr>
<tr>
<td>8. END=FindCloseNode(N,END);</td>
</tr>
<tr>
<td>9. Append(chain,END);</td>
</tr>
<tr>
<td>10. goto 5.</td>
</tr>
<tr>
<td>11. }</td>
</tr>
<tr>
<td>12. END</td>
</tr>
</tbody>
</table>

**Figure 2.** Greedy chain algorithm.

The main advantages of PEGASIS are: (1) The transmission distances between nodes are minimized. (2) The number of sensor nodes that must send packets to the base station is minimized. The main drawbacks of PEGASIS are: (1) It has excessive delay introduced by the single chain. (2) Greedy algorithm using in PEGASIS is a local search, which cannot provide an approximate global optimal route.

In this work we propose a coordinates-based clustering routing protocol, which satisfies the requirements stated above. By using multiple short chains, we can alleviate the excessive delay caused by PEGASIS while achieving more energy gain by a more global optimal routing.

### III. THE NETWORK AND RADIO MODELS

#### A. Network Model

We consider a 100-node network with randomly distributed nodes in a (100 x 100) meter area. The BS is located at \((x=50, y=300)\). The length of each signal is 2000 bits and the energy required for data aggregation is 5nJ/bit/signal. Moreover, data processing time per node is taken as 5-10 milliseconds. The radio speed is considered as 2 Mbps.

In the paper, we assume the following:

1. Each sensor node has power control and the ability to transmit data to any other sensor node or directly to the BS.
2. Our model sensor network contains homogeneous and energy constrained sensor nodes with initial uniform energy.
3. Every node has location information.
4. There is no mobility.

#### B. Radio Model

In our work, we used the first order radio model. In the radio model, the energy dissipation of the radio is to run the transmitter or receiver circuitry. Depending on the distance between transmitter and receiver, the free space propagation channel model is taken into account to measure the energy loss due to wireless transmission. Therefore, the energies expended to transmit a \( k \)-bit packet to a distance \( d \) and to receive that packet with this radio model are as follows.

For transmitter,

\[
E_{tx}(k,d) = E_{tx-elec}(k) + E_{tx-amp}(k,d) \tag{2}
\]

\[
E_{tx}(k,d) = E_{elec} \cdot k + E_{amp} \cdot k \cdot d^2 \tag{3}
\]

For receiver,

\[
E_{rx}(k) = E_{rx-elec}(k) \tag{4}
\]

\[
E_{rx}(k) = E_{elec} \cdot k \tag{5}
\]

Here, \( E_{elec} \) are energies required in running transmitter and receiver electronics. \( E_{amp} \) is the energy consumed in transmitter amplifier for transmission.

We make the assumption that the radio channel is symmetric such that the energy required to transmit a message from node A to node B is the same as the energy required to transmit a message from node B to node A for a given SNR. For our environment, we also assume that all sensors are sensing the environment at a fixed rate and thus always have data to send to the end-user.

### IV. CLUSTERING SCHEME BASED ON CHAIN ROUTING

We propose two schemes to combine clustering strategy with chain routing algorithm in order to satisfy both energy and delay constraints in WSNs. Clustering strategy is useful for low latency while chain routing algorithm is beneficial to energy efficiency. There are two schemes for clustering protocol based on chain routing.

#### A. Scheme 1

Clustering strategy + Direct transmission inside cluster + Chain routing among cluster-heads

First, all nodes make clustering. Then chain routing is made among cluster-heads from each cluster. When data transmission begins, each node in cluster sends data to its cluster-head directly. Then each cluster-head fuses and transmits its data along the chain. At last, one chain-leader sends data to BS directly.
B. Scheme 2

Clustering strategy + Chain routing inside cluster + Chain routing among chain-leaders

First, all nodes make clustering. Then chain routing is made among all nodes in each cluster and among all chain-leaders respectively. When data transmission begins, each node in cluster sends data along the lowest chain. Then each chain-leader from each cluster fuses and transmits its data along the top chain. At last, one selected top-leader in top chain sends data to BS directly.

By the contrast between two schemes, the second scheme performs better using Energy*Delay metrics. This is mainly for chain routing among all nodes in cluster in the second scheme is more energy efficient than direct transmission to cluster-head in cluster in the first scheme.

In the following sections, a protocol called CRBCC based on the second scheme is proposed in detail. Simulations show it can give a good compromise between energy consumption and delay.

V. PROTOCOL OF CRBCC

This paper proposes a two layer hierarchical routing protocol called CRBCC, which ensures the minimum energy consumption and low latency. Network routing begins with the formation of clusters. Several clusters are formed with the approximately equal size according to y coordinates. All nodes within a cluster form a chain among them using a simulated annealing algorithm. Then a chain leader of each cluster is elected in x coordinates order. After that a higher-level chain is formed including all leader nodes from every cluster using SA too. Among this leader-to-leader route only one randomly chosen top leader sends data to BS. The operation of CRBCC can be divided by three phases: route formation phase, data transmission phase and route maintenance phase. In the following subsections we discuss them in details.

A. TSP Problem in CRBCC

One of the TSP problems in WSNs is the chain routing where we want to find an approximate optimal route to minimize total energy consumption in gathering data. The traveling salesman problem (TSP) is as follows: Given a finite number of “cities,” along with the cost of travel between each pair of cities, find the cheapest way to visiting all the cities and returning to the starting point. There is several well known classical methods for TSP including Simulated Annealing (SA) algorithm ([16], [17]). Simulated Annealing is based on heuristics from annealing process. During the initial search phase (when temperature is high) local search is carried out and routes that minimize the total distance are always accepted. To escape the problem of getting stuck in local minima occasionally routes with distance more than the current route is also accepted but with a probability similar to the probability in the dynamics of the annealing process. As the temperature decreases, this probability of accepting a bad solution is decreased and in the final stages it becomes similar to gradient-based search. The SA algorithm uses Metropolis rule to get the probability of acceptance $P_a$ as follows

$$P_a = \begin{cases} 1 & \text{if } f(x_{\text{new}}) < f(x_{\text{old}}) \\ \exp((f(x_{\text{old}}) - f(x_{\text{new}})) / T) & \text{else} \end{cases} \text{if } f(x_{\text{new}}) \geq f(x_{\text{old}})$$

(6)

where $f(x)$ is object function which represents the total distance along the route $x$. But in CRBCC, object function $f(x)$ means the total energy consumption along the route $x$. In WSNs, we want to find an approximate lowest energy consumption route $x$ to minimize $f(x)$ through SA algorithm. So, we can set

$$f(x) = \sum_{d_{ij}} (E_{\text{elec}}^i + E_{\text{amp}}^i k d_{ij}^2)$$

(7)

where $d_{ij}$ belongs the asked route $x$ and represents the distance between node $i$ and $j$. That is,

$$f(x) = \sum_{d_{ij}} (2 E_{\text{elec}}^i + E_{\text{amp}}^i k d_{ij}^2)$$

(8)

Since $E_{\text{elec}}^i$, $k$ and $E_{\text{amp}}^i$ are all constants, $f(x)$ can be as

$$f(x) = \sum_{d_{ij}} d_{ij}^2$$

(9)

The cooling schedule is given by

$$T(n + 1) = \alpha * T(n)$$

(10)

where $\alpha$ is constant with $\alpha < 1$ and $n$ is the times of cooling temperature.

Procedure

ConstructSAChain(current_route, threshold)
1. current_cost = energy_cost(current_route);
2. while( iterations < threshold1 )
3. new_route = SwapNode(current_route);
4. new_cost = energy_cost(new_route);
5. diff = abs(new_cost - current_cost);
6. if(new_cost < current_cost)
7. current_route = new_route;
8. current_cost = new_cost;
9. if(temperature_iterations >= threshold2)
10. cooling(temperature);
11. temperature_iterations = 0;
12. end
13. temperature_iterations ++;
14. iterations ++;
15. else
16. if(rand(1) < \exp(-diff/(temperature)))
17. current_route = new_route;
18. current_cost = new_cost;
19. temperature_iterations ++;
20. iterations ++;
21. end
22. end
23. end

Figure 3. SA Chain algorithm.
The SA chain algorithm is as figure 3. The function \( \text{cost} = \text{energy\_cost}(\text{route}) \) evaluates the energy consumption \( \text{cost} \) along the route \( \text{route} \). The function \( \text{new\_route} = \text{SwapNode}(\text{old\_route}) \) makes a new route \( \text{new\_route} \) based on the old route \( \text{old\_route} \). The function \( \text{cooling}(T) \) cools the temperature \( T \) down.

**B. Route Computation**

The main activities in this phase are cluster setup, chain routing in cluster, chain leader election, leader-to-leader chain formation, top chain-leader election and schedule creation.

The CRBCC routing algorithm is as follows.

```
Procedure Routing(void)
1. Begin
2. if (in routing formation phase)
3. {
4. make balanced clustering by Y coordinates;
5. SA chain routing in cluster;
6. elect chain leader in X coordinates order;
7. SA chain routing among chain leaders;
8. elect top leader;
9. }
10. if (in data transmission phase)
11. {
12. communication in cluster;
13. communication among chain leaders;
14. }
15. if (node is failed)
16. make route maintenance;
17. End
```

Figure 4. CRBCC routing algorithm.

First, CRBCC makes balanced clustering according to \( y \) coordinates where each cluster has approximately equal number of nodes. Normally in many literature \( p=5\% \) nodes acting as cluster-heads is considered as an optimal result. Thus, in our experiment, 100 nodes of WSNs are divided into 5 clusters with 20 nodes in each. Second, CRBCC makes chain routing with SA algorithm inside cluster. After that, a chain leader of each cluster is elected in \( x \) coordinates order which can ensure the short distance between leader-leader. Till now, there are 5 chains with 5 chain leaders. Last, CRBCC makes chain routing again with SA method among chain leaders. Then a top chain leader is randomly chosen to forward the data to BS. BS without energy limitation can make route computation. Then the whole routing table is broadcasted to all nodes in WSNs. Each node keeps the routing table information related to it.

**C. Data Gathering**

The data gathering phase is divided into two parts, which are communication phase in the cluster and communication phase among clusters.

At the end of this phase, BS should broadcast the end of communication in current round and initiate the next round communication.

1) **Communication in cluster**

During this data communication phase each normal node is responsible to collect and fuse data from neighbor nodes in the chain of the cluster. At the end of this phase, one chain leader in each cluster gets the information of all nodes in that chain. The schedule can use Token or TDMA scheme.

2) **Communication among clusters**

In this collection phase, each chain leader node fuses and transmits its data along the top leader-leader chain. The schedule assignment can also use Token or TDMA method. Last, one randomly chosen top leader fuses the information of all nodes in WSNs and transmits to BS directly.

To reduce the delay, multiple chains make parallel communication using different CDMA codes. When a node decides to become a chain leader, it chooses randomly from a list of spreading codes. It informs all the nodes in the cluster to transmit using this spreading code. The chain leader then filters all received energy using the given spreading code. Thus neighboring clusters’ radio signals will be filtered out and not corrupt the transmission of nodes in the cluster.

**D. Route Maintenance**

Every a periodic time, neighbor nodes in chain will contact with each other using short detection messages. If a node does not reply, the neighbor node considers it to be dead. The neighbor node will skip the dead node on the routing table and send the information to the next sensor node.

When 10% of the sensor nodes are found to be dead in the WSN, the base station will move into the routing computation phase again.

© 2010 ACADEMY PUBLISHER
TABLE I. NORMAL ROUTING INFORMATION TABLE

<table>
<thead>
<tr>
<th>Node</th>
<th>Routing Information Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>HABCDF</td>
</tr>
<tr>
<td>A</td>
<td>HABCDF</td>
</tr>
<tr>
<td>B</td>
<td>HABCDF</td>
</tr>
<tr>
<td>C</td>
<td>HABCDF</td>
</tr>
<tr>
<td>D</td>
<td>HABCDF</td>
</tr>
<tr>
<td>E</td>
<td>HABCDF</td>
</tr>
<tr>
<td>F</td>
<td>HABCDF</td>
</tr>
</tbody>
</table>

![Figure 7. Recover chain routing.](image1)

TABLE II. RECOVER ROUTING INFORMATION TABLE

<table>
<thead>
<tr>
<th>Node</th>
<th>Routing Information Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>HABDEF</td>
</tr>
<tr>
<td>A</td>
<td>HABDEF</td>
</tr>
<tr>
<td>B</td>
<td>HABDEF</td>
</tr>
<tr>
<td>C</td>
<td>Dead</td>
</tr>
<tr>
<td>D</td>
<td>HABDEF</td>
</tr>
<tr>
<td>E</td>
<td>HABDEF</td>
</tr>
<tr>
<td>F</td>
<td>HABDEF</td>
</tr>
</tbody>
</table>

![Figure 8. Placement of WSNs.](image2)

VI. SIMULATION RESULTS

We make a comparative analysis of route structure, energy consumption and delay between PEGASIS and CRBCC. The parameters of experiment is the same in two protocol tests. With energy radio model, we set $k=2000$bit, $E_{elec}=50$J/bit and $E_{amp}=100$pJ/bit/m$^2$. In WSNs, 100 nodes place randomly in a 100m*100m area with BS (50,300). We assume $v=2$Mbps and $p=5\%$.

We run the simulations for at least 100 times. At last, we make a conclusion that CRBCC shows a better performance in both energy efficiency and low latency metric than PEGASIS.

A. Route Structure Comparison

We can see that routing map by PEGASIS shows many crosses, and many of these crosses are long-range. But there is no similar case in routing map by CRBCC. That means in CRBCC, the transmission distances between nodes are greatly minimized.

![Figure 9. Routing map by PEGASIS.](image3)

![Figure 10. Chain of 1st cluster in CRBCC.](image4)
Figure 11. Chain of 2nd cluster in CRBCC.

Figure 12. Chain of 3rd cluster in CRBCC.

Figure 13. Chain of 4th cluster in CRBCC.

Figure 14. Chain of 5th cluster in CRBCC.

Figure 15. Top leader-leader chain in CRBCC.

Figure 16. Routing map by CRBCC.
B. Energy Consumption Comparison

We can see that CRBCC spends less energy than PEGASIS, which can prolong the lifetime of WSNs. It is noted that one energy unit in y-axis is $\varepsilon_{\text{amp}} \times k = 2 \times 10^{-7}$ J. This improvement is mainly achieved by SA chain algorithm in CRBCC instead of greedy chain algorithm in PEGASIS.

![Figure 17. Energy consumption comparison.](image)

C. Delay Comparison

We can see that CRBCC reduces the data delivery time greatly than PEGASIS. This advantage is achieved by coordinates-based clustering method in CRBCC.

Assume 1 unit time is needed for one node to transmit k bit message with v bps to the neighbor node. On a 2Mbps link, a 2000 bit message can be transmitted in 1ms. Therefore, each unit of delay will correspond to about 1ms time for the case of a single channel and non-CDMA sensor nodes.

$$delay(\text{unit}) = \frac{k}{v} \quad (11)$$

The delay value will depend on the effective data rate for the CDMA sensor nodes. For example, as to the DS-CDMA system, suppose $m=8$ chips sequence per bit, then 1 unit time is needed for one node to transmit k bit message with v bps as

$$delay(\text{unit}) = \frac{mk}{v} \quad (12)$$

In CRBCC, there only need 19 unit time to collect data in the lowest chain for balanced clustering. After that nodes need 4 unit time to collect information in the top chain. At last, the top leader sends data to BS using 1 unit time. So, in total 24 unit time may require for transmission in CRBCC.

But in PEGASIS, there is only one single chain. There need 99 unit time to collect data in the chain and 1 unit time for chain leader to BS. Therefore, in total 100 unit time of delay may occur. Moreover, latency advantage in CRBCC is more obvious in large-scale WSNs.

![Figure 18. Delay comparison.](image)

TABLE III. DELAY COMPARISON (ONE ROUND)

<table>
<thead>
<tr>
<th>Protocol</th>
<th>PEGASIS</th>
<th>CRBCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay(unit)</td>
<td>100</td>
<td>24</td>
</tr>
</tbody>
</table>

VII. CONCLUSION

This paper proposes a hierarchical routing approach for WSNs in an energy and time constraint scenario. The creativity of the work is its clustering strategy and its chain routing algorithm in the whole routing protocol. Simulation results show that our protocol offers better performance than PEGASIS in terms of energy consumption and network delay. Later, we will make use of very fast simulated annealing algorithm (VFSA) and more fine clustering method in the routing protocol of WSNs.

In the future, we will study the scheme about clustering protocol based on tree routing algorithm. Tree routing algorithm has better performance than chain routing algorithm in both energy efficiency and latency metrics. Four tree routing schemes can be:

(1) Scheme 1: Tree routing among all nodes in WSNs.
(2) Scheme 2: Clustering strategy + Direct transmission inside cluster + Tree routing among cluster-heads.
(3) Scheme 3: Clustering strategy + Chain routing inside cluster + Tree routing among chain-leaders.
(4) Scheme 4: Clustering strategy + Tree routing inside cluster + Tree routing among cluster-heads.

ACKNOWLEDGMENT

The project is supported by the Research Foundation of Education Bureau of Hubei Province, China (No. B20081506) and Youths Science Foundation of Wuhan Institute of Technology, China (No. Q200808).

REFERENCES

Gengsheng Zheng received the B.S. degree in communication engineering from Xi’an Technology University, China, in 1994 and the M.S. and Ph.D. degrees in computer science from Wuhan University, China, in 1999 and 2006, respectively. Since January 2007, he has been an Associate Professor at the School of Computer Science and Engineering, Wuhan Institute of Technology, China. His current research interests lie in the area of network and embedded system.