International Journal of Hybrid Information Technology
Vol.8, No.9 (2015), pp.289-302
http://dx.doi.org/10.14257/ijhit.2015.8.9.27

A Distributed Clustering Algorithm with Optimal Search Model for Wireless Sensor Networks

Tao Lü[1,*], ZhongBin CAI[1], YuYu ZHU[2]

[1] *School of National Defence Science and Technology, Southwest University of Science and Technology, Mianyang, Sichuan 621010, China*
[2] *School of Information Engineering, Southwest University of Science and Technology, Mianyang, Sichuan 621010, China*
[*]*Email:lvtao@swust.edu.cn*

Abstract

In view of the operating mode of wireless sensor networks, the node initial probability and channel listening, a distributed clustering algorithm DCOS based on the optimal search model is proposed, along with the corresponding mathematical optimization model. The goal of model optimization is to find a feasible allocation scheme for search resource so that the probability of successful communication reaches its maximum under the constrained condition for search resource. Clustering algorithm is designed to balance the network energy consumption, thereby maximizing network lifetime. In the clustering algorithm, the cluster head is generated in the comprehensive consideration of the minimum number of nodes in the overlay network, and the node residual energy and the number of neighboring nodes, during the operation of the network, setting the maximum wait time for cluster heads may avoid the their energy consumption. Analysis and simulation results show that, compared to several important clustering algorithms, the performance of DCOS is proven to be superior in balancing node energy consumption and prolonging the network lifetime.

Keywords: *wireless sensor networks, operational mode, distributed clustering algorithm, optimal search model*

1. Introduction

In wireless sensor networks (WSN), the nodes may have very limited energy. Once the battery is run out, either to charge the nodes or replace the battery is generally unrealistic. Therefore, how to improve energy efficiency, balance node energy consumption and prolong the network lifetime, avoid network splitting and other issues have drawn great attention in the research work of WSN. As the work goes further, the focus has come to that how to conduct a reasonable topology control to optimize the topology and prolong the lifetime of WSN. Typically, the research of topology control covers two aspects: First, transmission power control, Second, distributed clustering mechanisms. Transmission power control is achieved by adjusting the transmit power of the node, where the first thing is to ensure the network connectivity, followed by reducing communication interference between nodes to improve the communication efficiency of the entire network. Distributed clustering mechanism requires that some nodes are selected according to certain rules to play the role of the cluster head node. The selected cluster heads may form the backbone of network to forward the data, while the other nodes can be dormant in the absence of the task.

The core of clustering algorithm is how to select the cluster head set. Optimal choice of cluster head is virtually to solve the minimum dominating set problem, while it is NP-hard problem, so most of the clustering algorithms are solved by seeking the heuristic or

ISSN: 1738-9968 IJHIT

approximated paths. LEACH [1] is the best known of the WSN clustering algorithms, and its good performance in that: nodes are completely dispersed and formed a cluster to process the data. Nonetheless, LEACH cluster head may be randomly generated, and not likely to form good clusters, for which energy may not be effectively used [2]. HEED [3] takes into account the residual energy of node in the selection of cluster heads. Subject to a master-slave relationship, a number of constraints are introduced to act on the process of the cluster head selection, which can produce a more uniform distribution of clusters head and a more reasonable network topology. However, after running for some time in the network, the energy consumption might become inconsistent. Comparing to the high energy nodes, HEED may enable the lower energy nodes to have a greater probability of being a cluster head, resulting in a shortened life expectancy of network. On the other hand, HEED in the cluster head election fails to take into account the distribution of nodes in the cluster, where the cluster nodes may come with an uneven distribution, causing the problem of imbalanced energy consumption, the same as LEACH. EECF [4], in the cluster head node selection, is able to take into account a combination of both the energy factor and the number of neighboring nodes, which are rated and sorted through three kinds of message exchange mechanisms. In this way, local optimal nodes are selected as the cluster head to form a better topology. DEECIC [5] cleverly have cluster members that are divided into the 1-hop and 2-hop categories, where member nodes are maintained by 1-hop to obtain an improved the network coverage quality compared with EECF.

The theory of Optimal Search involves the study of how to use a "best" way to find a pre-identified object (usually called "search target"). In short, it is to find the best allocation method for a search resource (e.g. time or energy) to either maximize the likelihood of successfully detecting the target, or to minimize the expected value for detecting the target (i.e., the resources consumed). The optimal search theory has developed rapidly in recent years, with a large number of accumulated results, and being widely used in all trades and professions. For example, geological exploration and prospecting, tracking and navigation, disease prevention and control, market research, criminalistics, economic management, and network scheduling optimization, information retrieval, mobile computing and many other fields [6-7].

This study presents a clustering algorithm based on the optimal search model for a wireless sensor network, or the Distributed Clustering with Optimal Search (DCOS). DCOS works with the custom node in five working conditions. Sleeping: the sensor module turned off, the communication module slept, the energy consumption minimized, Sensing: sensor module opened, communication module slept, where node only perceives events, Listening: sensor module closed, communication module in idle state, Receiving: sensor module turned off, the communication module to receive data, Sending: sensor module turned off, the communication module to send data. Nodes are ready to switch between different states for the completion of specific tasks.

DCOS runs in Rounds similar to LEACH for the same duration. Each round is divided into several frames, with the maximum duration of each frame determined by setting the threshold. In the initial moments of each round all nodes are in a dormant state, then to some probability, node switches to a different state. The node that first switches from the hibernation state to the sensing state is sensor node, i.e. non-cluster head node, which is responsible for sensing the environment and sending data. The node that first switches from the sleep state to the reception state is the cluster head, responsible for receiving and forwarding data. During each round of operation of the network, the cluster head may need to be adjusted due to the distribution or energy or other factors, so the roles of sensor node and cluster head may be swapped. At the end of each round all nodes come back to sleep, if all nodes run out of energy, indicating the running end of the network.

In comparison of the classic clustering algorithm, the biggest difference of DCOS is no stage established between the sensor nodes and the cluster head. Therefore, DCOS needs to address the following issues:

(1)At a time in the scene a certain number of cluster head nodes are needed for the backbone composition, ready for receiving and transmitting sensor data. This ensures that the cluster head provides a communication area that covers the whole network and communicates with each other,

(2)Ensuring that data sent by the sensor node can be received by cluster head, and the data can be transmitted in the backbone network,

(3)Enabling the cluster heads to be more evenly distributed.

2. Materials and methods

2.1. Running mode of node

To reduce the energy consumption of the working nodes, the nodes should be in sleep mode for most of the time. Only when there is a sensing event, should a node have data communications. As the cluster head should, after the incident has occurred, assist the sensing node in forwarding the data. Nodes, through transitions to the work state, enable the communication module coming to sleep as much as possible in the absence of tasks, thereby reducing the energy consumption of nodes. The conversion of the working state of nodes is both essential and effective to the sensor network. The mode of operation can be presented through different states of nodes. As the sensing node and the cluster head, because of different responsibilities at different times, may work in different modes, as shown in Figure 1.

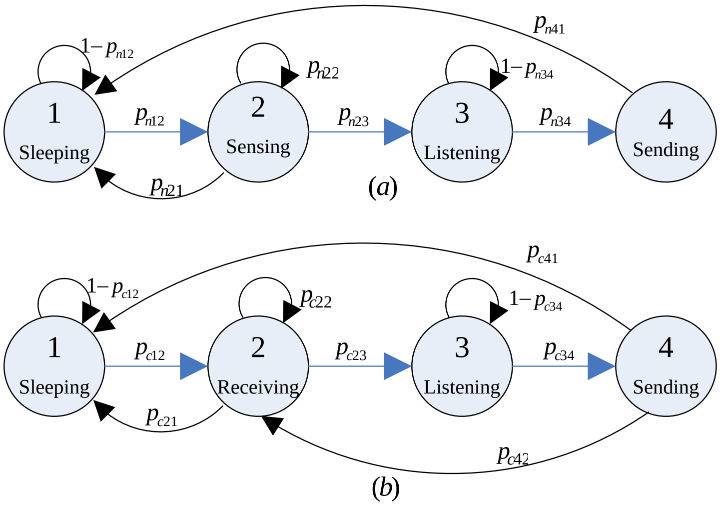


Figure 1. Operation mode of sensor nodes and cluster head (a) Operation Mode 1 (b) Operation Mode 2

Sensor nodes run in the following way: sensor node runs the Mode 1 at the first frame in each round, to the probability of p_{n12} , and then converts from hibernation state to sensing state, after sensor data are obtained, the node is switched to the listening state, where the node listens on channel. If the channel is busy, then listening continues, if the channel is idle, then the node transmits the data and comes back to sleep. Sensor node ends the operation at this frame. Sensor node begins to run the second frame by way of Mode 2, mainly to collect information on adjacent nodes. Such information is the basis to select the operating mode for the third frame.

Cluster head runs in the following way: cluster head start to run the Mode 2 in the first frame of each round, and the node, to the probability of p_{c12} switches from hibernation into the reception state. At this time, head node as the cluster is responsible for receiving sensor data and the information on adjacent nodes, cluster head in the receiving state, after staying for some time, may switch to the listening state to the probability of p_{c23} , listening state runs the same way as sensor node, where the cluster head is in the idle channel to send packets, after that, the cluster head returns to the receiving state, indicating that this frame operation is finished, if the cluster head does not comply with the agreed terms, then in the next frame begins the running in the Mode 1.

2.2 Initial probability of node

Assume that node's communication radius r_c is fixed, and $r_c < \frac{1}{2} d_0$ (d_0 is threshold for a free-space model or a multi-path fading model). The distance d_{toNext} between two adjacent nodes can satisfy the relationship $r_c < d_{toNext} < d_0$. This constraint allows the distribution of nodes in the network to achieve a better state. As the node has a fixed communication range and satisfies the constraint above, in the determined scene X, covering the entire area may require the number of the desired nodes that is directly associated with the area and the communication radius r_c of the node. The minimum required number n of nodes in the scene X is [8]:

$$\frac{X}{n \pi r_c^2} = \frac{2\pi}{\sqrt{27}} \tag{1}$$

Considering the residual energy and the number of neighbor nodes as an important parameter for the selection of cluster head, the probability for each node can be defined it as the cluster head is

$$p_{c12} = \frac{\sqrt{27}\, X}{2 r_c^2 \pi^2 N} \cdot \left(\alpha \frac{s(i)}{N-1} + (1-\alpha)\frac{E_p}{E_{init}}\right) \tag{2}$$

N is the total number of nodes in the scene X , $s(i)$ is the number of neighbor nodes of node i , α is the weighting value to balance the energy and the number of nodes, E_p is the residual energy of the node, E_{init} is the initial energy of the node.

2.3 Channel listening

Multiple nodes may communicate with each other being covered. In the use of the same channel to send data, competition will cause the node to enter the sending state at different timing states. When a node is ready to send data, it first listens on the channel, if the channel is busy, it insists listening to the next slot, if the channel is idle, then it transmits data to the probability q , and with probability $1-q$, the listening is postponed to the next slot, if the next slot channel remains idle, then it is still sending data to the probability q , with the listening being postponed to the next slot to the probability $1-q$, This is sustained until the data is sent. In order to ensure a low latency, q ranges from $[0.3, 0.5]$ [9]. Considering the worst case, where there is a certain time when $k = p_{n12} \cdot N$ sensor nodes may be distributed within a single cluster, and simultaneously in the listening state, transmitting data is desired. The node state is defined as (k, m) , where $0 \le m \le k$. Then $P(k, m)$ defined by

$$P(k,m) = \binom{k}{m} q^m (1-q)^{k-m}, \quad \sum_{m=0}^{k} P(k,m) = 1 \tag{3}$$

$(k, 0)$ indicates that no node sends data, network being in an idle state, $(k, 1)$ represents a particular node successfully transmitted the message, when $m > 1$, it means that plurality of nodes are contenting a channel, where a conflict has occurred to the networks. One of the k nodes has the probability of successful transmission as

$$P_s(k) = P(k,1) = kq(1-q)^{k-1} \tag{4}$$

The network with k nodes has the probability of being in an idle state as

$$P_i(k) = P(k,0) = (1-q)^k \tag{5}$$

A contention period may appear to the probability of $1 - P_s - P_i$. therefore k nodes may have m contention periods to the probability of

$$P_m(k) = P_s(k)(1 - P_s(k) - P_i(k))^m \tag{6}$$

Thus in the state (k,m) the expected value of the contention period is

$$E[(k,m)] = \sum_{m=1}^{k} mP_m(k) = \frac{P(k,m)}{P_s(k)} \tag{7}$$

If the time to transmit a packet data comprises the channel contention time, transmission time and the end-to-end delay, assuming that each contention period is 2τ, the end-to-end delay is τ, the k nodes may send a packet of data in an average time:

$$T_{send}(k) = 2\tau \frac{P(k,m)}{P_s(k)} + \frac{l}{s} + \tau \tag{8}$$

Where l is the packet length, s for the channel rate. $P_s(k)$ is the probability of node being switched from listening to sending, T_{send} is the average time for the node to be switched from the beginning of listening to the end of sending, with the upper bound to be determined by the number of nodes expected to send data.

2.4 Optimal Search Model

Regarding the previous question (2), namely, the way to ensure the inter-node communication, an optimal search model may be established to solve it. at some point of the operation of the network , nodes that are expected to be the sending and receiving states may appear in pairs, with minimal interference between other nodes. If the data sent out happens to be received by the node in the receiving state, and also relying on this principle, the data is transmitted in the network. Then the node in the receiving state (cluster head) is assumed to be the searcher, and the node in the sending state (sensor nodes) to be the target. The problem with the node in successful communication comes to the searcher, at a minimum price (wait time for reception), successfully finds out to the target. According to the theory of optimal search, the problem (2) may be attributed to an optimal search for the discrete search space and discrete resources.

J is defined as the set of all the communication nodes covered by a certain searcher in the scene, then J is a positive integer subset. The probability of target J to be in the searcher is denoted by $p(j),\ j = 1,2 \ldots J$. Under normal circumstances,

$$\sum_{j \in J} p(j) = 1 \tag{9}$$

If the position of the target is not fully distributed, that is, the target might not be within communication range covered by the searcher , then

$$\sum_{j \in J} p(j) < 1 \tag{10}$$

Assume that the target is located in the searcher j, then the probe function $b(j,r)$ indicates the probability of successful communication within the waiting time not exceeding r times between the target j and the searcher, i.e. the searcher, with the wait time of r times , may have the probability of transmitting data to the target j. Assuming each waiting time is fixed, comprising the data transmission time and the end-to-end time delay, that is

$$t^* = \frac{l}{s} + \tau + \beta\tau \qquad \beta = 1,2,... \tag{11}$$

β is the adjustment factor of t^*, determined by the lower bound of listening time. If the communication is always successful, that means searchers can always receive data from the target, then there is $b(j,r) \to 1, r \to \infty$. Nonetheless, the interference may be actually present with the blind area from the network coverage, usually $b(j,r) \to \alpha < 1$. According to the nature of the probe function, the longer the searcher in the receiving state, the greater the probability of receiving packets. This principle enables the probe function $b(j,r)$ to be a regular one, that is, $b(j,r)$ may have continuous derivative to each j, and the derivative function $b'(j,r)$ is a monotonically decreasing function satisfying $b'(j,r) > 0$, and. Assume

$$b(j,r) = 1 - e^{-r}, (j = 1,2,... J) \tag{12}$$

The detection function is derived from the theory of optimal search using the Koopman random search equation[11]. Let $c(j,r)$ represent the consideration for the searcher to wait for r times. The following talks about the case of the total number of wait times with a fixed upper bound. Let R be a permitted total wait times, and $\xi(r) = (\xi_1, \xi_2 ... \xi_r)$ means the program for r wait times. For each $\xi(r)$, it is associated with a resource allocation function $f_{\xi(r)} : (1,2,... J) \to (1,2,... r)$, representing the wait times of the searcher in the implementation of the search program $\xi(r)$, that is

$$f_{\xi(r)}(j) = \sum_{i=1}^{r} I_{\xi_i = j}, \; j = 1,2,... J \tag{13}$$

Here $I_{\xi_i = j}$ is instruction function for the set $\{\xi_i = j\}$. If $\xi_i = j$ is established, then $I_{\xi_i = j} = 1$. Otherwise, $I_{\xi_i = j} = 0$. In addition, the total waiting times is limited less than or equal to R, which means

$$f_{\xi(J)}(j) = \sum_{i=1}^{J} I_{\xi_i = j} \le R \tag{14}$$

The total consideration for the search:

$$C[f] = \sum_{j \in J} c(j, f_{\xi(r)}(j)) = \sum_{j \in J} f_{\xi(r)}(j) = r \sum_{i=1}^{J} (i+1) \cdot T_{send}(i) \cdot E_{idle} \tag{15}$$

E_{idle} is the node energy consumption when idle listening. According to the above model, the communication issue is to find a workable plan for the distribution of resources and the related search programs, so that the probability of successful communication is able to achieve maximum subject to the resource constraints $C[f] \le R$. Using the Lagrange multiplier method to calculate the optimal search strategy f^*, the Lagrangian function can be defined $l(j,\lambda,r) = p(j)b(j,r) - \lambda r$, where $1 \le j \le J, \lambda > 0, r \ge 1$. By seeking the extreme Lagrangian function, the allowable optimal allocation of resources on the parameter r is given by

$$\frac{\partial l}{\partial r} = p(j)e^{-r} - \lambda = 0 \tag{16}$$

Thus,

$$r = \ln\frac{p(j)}{\lambda} = r_j \tag{17}$$

The constraints lead to

$$\sum_{j=1}^{J} \ln\frac{p(j)}{\lambda} \le R \tag{18}$$

The inequality is taken with the equal sign at the upper bound of resource constraints. Optimal search program has a detection probability $P[f^*]$ is given by:

$$P[f^*] = \sum_{j=1}^{J} p(j)(1-e^{-r_j}) = \sum_{j=1}^{J} p(j)(1-e^{-\ln\frac{p(j)}{\lambda}}) \tag{19}$$

Easy to get the upper limit of resource allocation r_j^*. From (18):

$$\lambda \ge [p(1)\ldots p(j)]^{1/J} e^{-R/J} \tag{20}$$

Then

$$r_j^* = \ln\frac{p(j)}{\lambda} \le \ln\frac{p(j)}{[\prod_{j=1}^{j} p(j)]^{1/J}} + \frac{R}{J} \tag{21}$$

Where, $\frac{R}{J}$ is the average waiting time assigned to each target, whereas the waiting time is given r_j^* as the upper bound. This conclusion can be understood as follows: If the data can always be successfully received, i.e. assuming that all sensor nodes have $p(j)=1$, and then there must be $b(j,1)=1$ and $r_j^*=1$. In normal circumstances $p(j)$ is determined by sensor nodes and probability of listening, so the cluster head needs to wait for a number of times.

2.5. Description of Distributed Clustering Algorithm

For sensor nodes, each transmitted data should include sensing data, the cluster head set S_{CH}, the node identifier ID and cluster head flag CH. For cluster head nodes, each transmitted data, in addition to sensor data and its corresponding ID, CH, should also include the adjacent node set S_{nbr} that is to replace cluster head set S_{CH}.

1) At the first frame of each round, nodes within a scene, based on a random probability, are switched from sleeping state to working state, nodes that have come into receiving state is set to the cluster head node, with the CH flag set as TRUE, nodes that have come into the sensing node status is set as sensor nodes, with the CH flag set as FALSE.

2) Sensor nodes in each frame completes a data transmission for once, and then is switched to the dormant state, waiting for the next frame to start, sensor node runs second frame by way of mode 2, and entering the receiving state.

3) Cluster head is responsible for data reception and forwarding within each frame of time. In the receiving state, the time is determined by $t_i^*(i \le \delta)$, where δ is the node threshold number, which enables the uniform number of clusters being distributed. Receiving process consists of the following situations:

- If the cluster head t_1^* and t_2^* receive no data, then the cluster head flag is set to FALSE, and the packet to EMPTY, then switched to the listening state.
- If the cluster head t_1^* confirmed the receipt of data, and continues to await t_2^*, if the data is received, it will continue to await t_3^* until t_δ^*, then switched to the listening state.
- If the cluster head received data at t_{i-2}^*, no data is received at t_i^*, then switched to the listening state.
- If the cluster head node received data twice in a frame, then switched to listening state.

A cluster head, after the end of sending action, should first determine whether the frame has ended, if the time has come to t_δ^*, as long as the CH flag is not FALSE, it is switched to receiving state after the end of the transmission, if the time arrives only at $t_i^*(i < \delta)$, then $(\delta - i)t^*$ is a required dormant state before entering the receiving state.

For processing the received data, the following situations may be involved:

- Data received from the sensor node, with the ID to be written into s_{nbr} .
- Cluster head u received data from the cluster head v, with the ID to be written into s_{nbr} , if $s_{u-nbr} \subseteq s_{v-nbr}$, then no data is forwarded for such cluster head. After this transmission is completed, again CH is set to FALSE, and switched to sleep mode, otherwise, forwarding the data not duplicate for cluster head v, that is, $s_{u-nbr} - (s_{u-nbr} \cap s_{v-nbr})$.
- An empty packet received, and left unattended, with the ID written into s_{nbr} .

4) In the second frame sensor node will enter the receiving state. The data received may be classified for discussion:

- Sensing node x received data from the sensor node y , with the ID written into s_{nbr} , if it has the same cluster head, then the packet is discarded and set to EMPTY, otherwise, transmitted.
- Sensing node x received date from the cluster head v, with the ID being written into s_{CH} , if $s_{x-nbr} \subseteq s_{v-nbr}$ or $u_{ID} \in s_{v-nbr}$, then not to forward the data for the cluster head. Meanwhile, after this transmission is completed, the CH is set to FALSE, and switched to sleep mode. Always, when starting from the next frame, node is running by way of mode 1 until the end of the round, if $s_{x-nbr} \not\subset s_{v-nbr}$, then data not duplicate is forwarded for the head v, with CH being set to FALSE, and the next frame run by way of mode 2, if $u_{ID} \notin s_{v-nbr}$, then CH is set to TRUE, with the next frame run by way of mode 2,
- An empty packet of an ID received, and CH is set to TRUE, with the ID written into s_{nbr} .

Through the description of the algorithm, it can be seen that the problem (3) contains two meanings: one is that cluster head has the position within the scene being evenly distributed, two, the number of sensor nodes is uniform within each cluster. Since the node position within the scene is unknown, so to get an evenly distributed cluster head position can only be judged through mutual degree of coverage. In this algorithm, the node is compared with the s_{nbr} for the adjacent cluster head to determine whether it is possible to become a cluster head, the second point may be adjusted by the time elapsed for cluster head to be in the receiving state, similar to the mechanism of TDMA with a time slice to be used for re-planning. DCOS is guaranteed by setting a node threshold.

2.6. Energy Estimates

Assuming the sending side works with the energy that includes signal processing and power amplification, whereas the receiving end has the energy to be reserved for signal processing only. When there is a data transmission occurring to two nodes in a distance of d between them, the sending end and receiving end may have the energy consumption, respectively:

$$E_{tx}(l,d) = E_{elec}(l) + E_{tx-amp}(l,d) = \begin{cases} lE_{elec} + l\varepsilon_{fs} d^2, d < d_0 \\ lE_{elec} + l\varepsilon_{mp} d^4, d \geq d_0 \end{cases} \tag{22}$$

$$E_{rx}(l) = lE_{elec} \tag{23}$$

Where E_{elec} is the energy needed for signal processing, depending on the digital coding, modulation, filtering, spread signal and other factors. Amplifier power ε_{fs} and ε_{mp} are determined by the distance from the transmitter and the bit error rate of the receiver. The values of parameters in the energy model are listed in Table 1.

Table 1. List of the Simulation Parameters in the Energy Model

Parameter	Value
Packet length l	500Byte
Signal processing power E_{elec}	50nJ/bit
Threshold for the energy model d_0	40m
Amplifier power ε_{fs}	10pJ/bit/m^2
Amplifier powerε_{mp}	0.0013pJ/bit/m^4

Since the distribution of nodes and communication probability of success are unknown, estimation can only be done on the average energy consumption of nodes, the simulation parameters of energy estimates are listed in Table 2.

Table 2. List of the Simulation Parameters of Energy Estimates

Parameter	Value
Scene X	[100,100]
Node energy E_{init}	2J
Total number of nodes N	100/200
Communication radius r_c	20m
Threshold for the number of cluster members δ	15
Idle power E_{idle}	13.5mW
Sleep power E_{sleep}	15uW
Sensing time t_{sensor}	100ms
Sensing Power E_{sensor}	1mW
Duration of each round S	20sec

For the first frame the sensor node may have an estimated energy value:

$$E_{sr1} = NP_{n12}(t_{sensor} \cdot E_{sensor} + E[(NP_{n12}, \frac{P_{n12}}{P_{c12}})] \cdot E_{idle} + E_{tx}(l, r_c) + (\delta - 1)2t^* \cdot E_{sleep}) \tag{24}$$

For the second frame, energy estimate of sensor nodes:

$$E_{sr2} = NP_{n12} \cdot \frac{P_{n12}}{P_{c12}} \cdot E_{rx}(l) + E[(NP_{c12}, m_2)] \cdot E_{idle} + m_2 E_{tx}(l, r_c) + (NP_{n12} - m_2)(\delta - 1)2t^* \cdot E_{sleep}$$

(25)

For remaining frames:

$$E_{sri} = (NP_{n12} - m_i) \cdot E_{sr1} + m_i \cdot E_{sr2}, \quad i = 3, 4 \ldots. \tag{26}$$

m_i is the number of sensor nodes in the No. i frame that are required to forward data or to become a cluster head

Energy estimate of cluster head for the first frame

$$E_{cr1} = NP_{c12}(\frac{P_{n12}}{P_{c12}} \cdot P[f^*] \cdot E_{rx}(l) + E[(NP_{c12}, 1)] \cdot E_{idle} + E_{tx}(l, r_c) + (\delta - \frac{P_{n12}}{P_{c12}})2t^* \cdot E_{sleep}) \tag{27}$$

Estimated energy of cluster head for the remaining frames

$$E_{cri} = (NP_{c12} + m_i - n_i) \cdot E_{cr1}, \quad i = 2, 3, \ldots \tag{28}$$

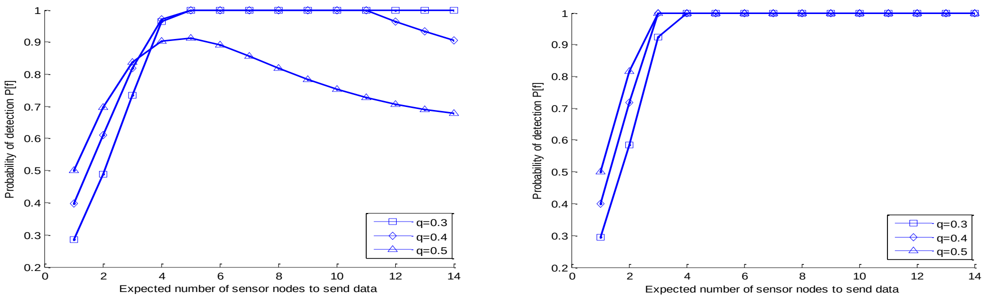
n_i is the number of head nodes for the No. i frame that cannot serve as a cluster head.

Therefore, a single round of the network energy is estimated to be:

$$E_{round} = (\frac{S}{\delta \cdot 2t^*} - 1) \cdot E_{cri} + E_{cr1} + (\frac{S}{\delta \cdot 2t^*} - 2) \cdot E_{sri} + E_{sr1} + E_{sr2} \tag{29}$$

3. Results

Figure 2 shows two cases as $r_j^* = 1$ and $r_j^* = 2$, respectively, under which the sensing node at different listening probability presents the statistics on detection probability. The analysis says, at $r_j^* = 1$, $q = 0.3$, when the expected nodes in the cluster exceed 4 ($J > 4$), then there must be $P[f^*] = 1$, At $r_j^* = 2$ $r_j^* = 2$, $q = 0.3$, when the expected nodes in the cluster exceed 3, then $P[f^*] = 1$, At $q = 0.4$ or $q = 0.5$, when the node number is more than 2, then $P[f^*] = 1$. From the statistical data of cost shown in Figure 3, when $J < 5$, the cost of probe basically tend to be consistent. Therefore, let $r_j^* = 2$, $q = 0.5$ in subsequent experiments.



(a) **(b)**

Figure 2. Statistics on Detection Probability of Nodes in Different Listening Probabilities

(a) $r_j^* = 1$ **(b)** $r_j^* = 2$

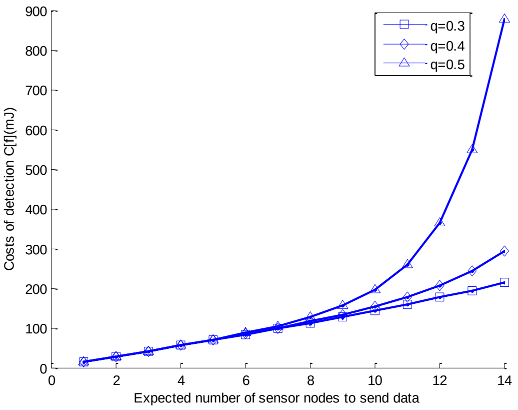


Figure 3. Costs on Statistic Detection Probability of Nodes in Different Listening Probabilities ($r_j^* = 2$)

When the scene X and communication radius rc are determined, Equation (3) can give $n = 7$, i.e. the scene accommodates the least 7 cluster heads, at the initial time the energy of nodes is the same, the adjacent node information is not obtained, so setting $\alpha = 1$. According to the EECF that made a comparative analysis on EESH, from the beginning of the second frame, $\alpha = 0.1$ is the optimal setting[10]. By Equation (4) when $N = 100$ is obtained, $p_{c12} = 0.07$, Based on the statistical results in Figure 3, $p_{n12} = 3p_{c12}$ is set, considering the worst case, if the sensing nodes are in the same cluster, then the node that sends a packet may take an average time to the upper bound of $T_{send}(p_{n12} * N) \approx 500\ ms$, and to the lower bound of $T_{send}(p_{n12} / p_{c12}) = 104\ ms$. Therefore $\beta = 3$, $t^* = 104\ ms$, to ensure that when the distribution of the sensor nodes is poor, the cluster head will not wait indefinitely in the consumption of energy, when distribution is better, it ensures the cluster head has a maximum probability of successfully receiving data.

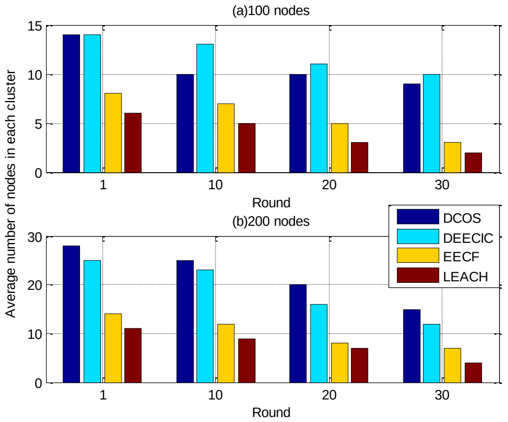
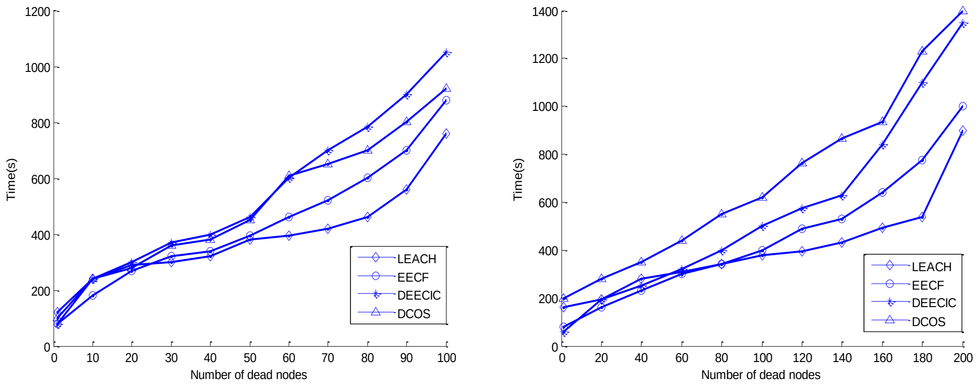


Figure 4. Statistics on the Average Nodes per Cluster in Different Scenes

From the analysis of the statistical results in Figure 4, when the network scene has only 100 nodes, DEECIC gets the maximum average number of nodes per cluster, superior to the other algorithms. This is because DEECIC always chooses the smallest cluster head sets to cover the entire network, where the redundant cluster heads are reduced, and the larger ones are easier to improve the fusion efficiency of data. Therefore compared with DEEIC, DCOS in this scene exhibits no advantage, yet better than the other two algorithms. When the network has 200 nodes in the scene, DCOS begins to present a better performance than that of DEECIC. The reason is here, according to the optimal search model, the higher node density ensures that DCOS has a successful communication probability, with the cluster head being easier to stabilize. As a result, DCOS could be more effective on the scene with densely deployed nodes.



(a) **(b)**

Figure 5. Statistics on the Time of Node Death in different scenes (a) 100 nodes (b) 200 nodes

Figure 5 shows the statistics on the time of nodes death in the different scenes. When there are only 100 nodes, DCOS outperforms EECF and LEACH, When the number of death is less than 60 nodes, DCOS and DEECIC almost share the same time of node death. Nevertheless, as the node is declining in number, the DCOS performance may face a faster degradation, which is associated with the features of the algorithm itself, when the scene was increased to 200 nodes, DCOS was highlighted in its stability, with the node's time of death being significantly better than that of DEECIC, As nodes increased, DCOS presents a growing higher communication probability, so the cluster head and sensor nodes work more stable, without frequently switching roles.

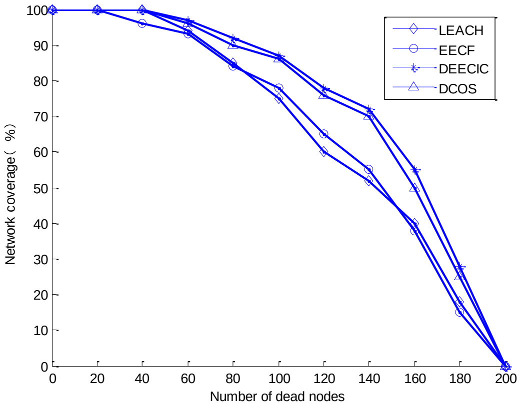


Figure 6. Statistics of 200 Nodes on Network Coverage

Figure 6 shows the statistical results that DCOS network has a coverage degree that is basically consistent with that of DEECIC, and better than LEACH and EECF. DCOS in the algorithm used a method that is different from DEECIC to control the network coverage. DEECIC uses the 1-hop node in the cluster to control the 2-hop node, not only increasing the degree of network coverage, but also reducing the redundant nodes. For DCOS, a minimum number of nodes were introduced to cover the network. Meanwhile, adjacent node set in the running process may be included, upon which the running way of nodes is determined. The experiment proved that this method is feasible and effective.

4. Discussion

Search theory is one of the oldest areas of operations research [11].The Initial developments were made by Bernard Koopman and his colleagues in the Anti-Submarine Warfare Operations Research Group of the US Navy during World War II to provide efficient methods of detecting submarines.

In this study, a distributed clustering algorithm DCOS is presented based on the optimal search model for wireless sensor networks. Experiments and data show that the algorithm is more suitable for densely deployed scenarios. Although the algorithm excels the other three algorithms in some performance indicators, there are still some shortcomings and areas for improvement: In the optimal search model where detecting function as a regular convention may also be a non-regular one in the practical application. This requires to continue to improve the mathematical model. The density function $p(j)$ in the model is achieved by way of probability model in the node listening state. In the actual operation of the network, a node can also be trained to obtain more accurate distribution function, which then leads to further optimization of algorithm. That is, in the search program, the distribution of functions on different cluster heads may be based for preparing different search plans.

Acknowledgments

This research was supported by the Doctor Foundation of Southwest University of Science and Technology (Grant No. 14zx7104).

References

[1] Heinzelman,W.R., A.Chandrakasan and H.Balakrishnan. Energy efficient communication protocol for wireless microsensor networks. Proceedings of the Hawaii International Conference on System Sciences. **(2000)**Jan 4-7, Wailea, USA.

[2] Lindsey, S., C. Raghavendra and K.M. Sivalingam. Data gathering algorithms in sensor networks using energy metrics. IEEE Transactions on Parallel and Distributed Systems, 13, 9 **(2002)**.

[3] Younis, O. and S. Fahmy. HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks.. IEEE Trans. Mobile Comput, 3 **(2004)**.

[4] Chamam, A. and S. Pierre. A distributed energy-efficient cluster formation protocol for wireless sensor networks. Proceedings of the Consumer Communications and Networking Conference, **(2009)**January 10-13, Las Vegas, USA.

[5] Liu, Z., Q. Zheng, L. Xue and X. Guan. A distributed energy-efficient clustering algorithm with improved coverage in wireless sensor networks. Future Gener. Comput. Syst, 28**(2012)**.

[6] Zhu, Q. and J. Oommen. On the optimal search problem: The case when the target distribution is unknown. Proceedings of the 17th International Conference of the Chilean Computer Science Society. **(1997)**Nov 10-15, Los Alamitos, USA.

[7] Zhu, Q. and B. Peng. Optimal search strategy for web-based 3D model retrieval. Proceedings of the International Conference on Advanced Web and Network Technologies, and Applications, **(2006)**Jan 16-18,Harbin China.

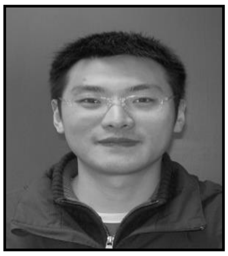
[8] Slijepcevic, S. and M. Potkonjak. Power efficient organization of wireless sensor networks. Proceedings of IEEE International Conference on Communications. **(2001)**Jun 11-14, Helsinki,Finland.

[9] Bouabdallah, N., M.E. Rivero-Angeles and B. Sericola. Continuous Monitoring Using Event-Driven Reporting for Cluster-Based Wireless Sensor Networks. IEEE Transactions on Vehicular Technology, 58,7 **(2009)**.

[10]Zhou, W., H.M. Chen and X.F. Zhang. An Energy Efficient Strong Head Clustering Algorithm for Wireless Sensor Networks. International Conference on Wireless Communications, Networking and Mobile Computing. **(2007)**Sept. 21-25,New York, USA.

[11]Stone, L.D. Search Model. Theory of Optimal Search. Kettelle, J.D. (Ed.).Institute for Operations Research and the Management Sciences ,USA **(1975)**, pp:1-34.

Authors



Tao Lü received the PhD degree in computer science from the University of Electronic Science and Technology of China in 2013. He is a lecturer of Southwest University of Science and Technology. His research interests include wireless ad hoc and sensor networks.

ZhongBin CAI, senior engineer in Southwest University of Science and Technology. His research interests include wireless communication theory and technology.

YuYu ZHU, lecturer of Southwest University of Science and Technology.His research interests include intelligent instrument and wireless communication.