An Improved Soft K-means Clustering Protocol for Balancing Energy in Wireless Sensor Networks

NAME

***Abstract*— Energy load balance is the essential issue in designing the wireless sensor networks (WSNs). Clustering technique is utilized as an energy-efficient routing to balance the network energy and prolong the lifetime, which is influenced by the cluster head selection and clustering method. In this paper, we propose an approach based on the idea clustering by fast search and ind of density peaks (CFSFDP) and kernel density estimation (KDE) to optimize the selection of the initial cluster heads of clustering algorithm. And then, we use reassigning member nodes and multi-heads scheme to balance the energy consumption of all nodes in the whole network. Simulation results demonstrate that the proposed approach can postpone the first node death by almost 3 times and the half of nodes death by 2 times compared to LEACH. Also, smoother energy distribution curve of all nodes in different round and smaller energy variance are obtained by our proposed algorithm.**

***Index Terms*— WSNs, KDE, soft K-means**

I. Introduction

Energy limitation is a key challenge in WSNs, since nodes' batteries cannot be rechargeable or nodes are placed in areas where it hard to reach to replace batteries. So, energy consumption is an important design issue for WSNs [1]. Clustering in energy limited WSNs has been widely pursued in order to solve energy issue of sensor networks. The clustering-based algorithm groups sensor nodes into distinct clusters with a head in a cluster and each sensor node belongs to only one cluster. All member nodes sense environment information and send it to the cluster heads (CHs), the CHs collect and process the data and send it to the base station (BS) via single-hop or multi-hop [2]. Each node consumes a certain amount of energy when it collects, processes, and sends data. A node is defined to be dead when it is out ot energy [3]. Hence, it is crucial to find efficient clustering algorithms to balance the energy consumption among sensor nodes in WSNs.

Machine Learning-based clustering algorithms are considered to be the most effective clustering methods in WSNs. Reference [4] implements both centralized and distributed K-means clustering algorithm in WSNs. EEK-means [5] creates symmetric clusters and reduces the average intra-cluster communication distance by the K-means classification method in order to save energy of nodes and improve the network lifetime. K-LEACH protocol [6] prolongs the lifetime of sensor networks by balancing the nodes' energy consumption, which uses K-medoids for clustered WSNs. Reference [7] uses K-means and Gauss algorithms to achieve the optimization of energy consumption of a network and the extension of its duration of life.

In this paper, an improved soft K-means (ISK-means) clustering protocol is proposed. Specifically, the main contributions of this paper are as follows.

1) Compared with some existing cluster-based protocols that select CH randomly, we choose the initial centroids of soft K-means clustering algorithm [8] by using the idea of density from clustering by fast search and find of density peaks (CFSFDP), which is implemented by kernel density estimation (KDE).

2) According to the characteristics of soft K-means, reallocating member nodes that locate in the boundary of two or more clusters is employed to balance the number of nodes in different clusters.

3) Since the clustering process needs to be repeated continually, it can increase the communication cost during the period of clustering. Multi-heads method is used to balance traffic load of CHs of different clusters and reduce the frequency of clustering.

The rest of this paper is organized as follows: Section II reviews the related works. The background for our research are described in Section III. Section IV proposes the implementation of the proposed ISK-means algorithm. In section V, we compare the performance of the proposed ISK-means with other routing algorithms. Finally, the paper is concluded in Section VI.

II. RELATED WORKS

Different clustering techniques have been proposed in designing WSNs to obtain energy efficiency and maximize network lifetime. Low energy adaptive clustering hierarchy (LEACH) is one of the first energy efficient routing protocols, proposed by Heinzelman et al. [9]. In this protocol, it selects CHs using a threshold among nodes by rotation and other nodes choose the nearest CH to form cluster, which can spread energy dissipation to all nodes in the network. However, it may result in a nonuniform distribution of CH thus causing high energy consumption. Low energy adaptive clustering hierarchy centralized (LEACH-C) [10] is the modified version of LEACH. In LEACH-C, each node sends its current location and residual energy to BS. BS determines the number of CH and arranges network into various clusters, which can ensure that load is eventually distributed in different clusters. However, this centralized approach can increase communication

overhead during the period of selection of CH. Energy-aware clustering algorithm [11] considers two factors: the energy factor for cluster head selection and distance factor for non-cluster heads to select its cluster head, which achieves a good performance in terms of lifetime by balancing the energy load among all the sensor nodes in the network.

K-means clustering algorithm is used in WSNs, which tries minimize the sum of Euclidean distances between the head and member nodes according to a specified number of clusters [4]. Due to selection of CHs randomly, it can result in sub-optimal clusters and uneven distribution of load. Reference [12] proposes a modified K-means algorithm that considers two factors: distance between CHs and its member nodes and remaining energy of nodes to reduce energy consumption and extend the lifespan compared to LEACH. BPK-means [13] balances the clusters to improve intra-cluster communication consumption, which can achieve better load-balance. In [14], the authors propose a hybrid clustered routing algorithm based on K-means clustering algorithm and LEACH protocol, which outperforms LEACH in terms of energy consumption. However, it can increase energy consumption during the phase of CH election process. EECPK-means [15] balances the load of CHs in WSNs by producing balanced clusters where midpoint method is used to improve the initial centroids selection of K-means algorithm.

III. PRELIMINARIES

A. Soft K-means

K-means is the simplest clustering algorithm in unsupervised learning, which partitions the data set in to k clusters using some distance measurement methods, like Euclidean distance. It is a hard clustering method, that is to say the membership degree of one node has only two values 0 and 1 for a specific class. However, in some cases, there are a few data points for which it is not quite so obvious to which cluster they belong. Soft K-means clustering decides to which degree each data point belongs to , the assignments to clusters will be probabilistic. Generally speaking, soft K-means clustering can be seen as the problem of finding k cluster centroids with the aim of minimizing the error function. Given a set of data points $X = \{x_1, x_2, ..., x_n\}$, the error function is

$$E(\mu_1, \mu_2, ..., \mu_k) = \sum_{i=1}^{k} E(\mu_i) = \sum_{i=1}^{k}\sum_{j=1}^{n} z_{ij}||x_j - \mu_i||^2 \quad (1)$$

where μ_i is the centroid of each cluster, z_{ij} is the indicator variable.
For the traditional K-means clustering

$$z_{ij} = \begin{cases} 1, \text{if } x_j \in C_i \\ 0, \text{otherwise} \end{cases} \quad (2)$$

where C_i represents cluster i, this equation means whether x_i belongs to cluster C_i. However, z_{ij} is not a integer value for soft K-means clustering

$$z_{ij} = \frac{e^{-\beta||x_j - \mu_i||^2}}{\sum_{l=1}^{k} e^{-\beta||x_j - \mu_l||^2}} \quad (3)$$

where β is the stiffness parameter, its impact for clustering result will be discussed in simulation section. From equation (3), we can get $z_{ij} \in [0, 1]$ and $\sum_i z_{ij} = 1$.
How to update the centroid μ_i of each cluster until convergence? For a specific class C_i

$$E(\mu_i) = \sum_{j=1}^{n} z_{ij}||x_j - \mu_i||^2 \quad (4)$$

The target is to minimize the error function, so the problem can be turned into the following optimization problem

$$e = \text{argmin} \sum_{j=1}^{n} z_{ij}||x_j - \mu_i||^2 \quad (5)$$

which is a convex function and has a unique minimum value. The equation (4) can be written as

$$E(\mu_i) = \sum_{j=1}^{n} z_{ij}(x_j - \mu_i) \top (x_j - \mu_i) \quad (6)$$

$$= \sum_{j=1}^{n} z_{ij}(x_j \top x_j - 2x_j \top \mu_i + \mu_i \top \mu_i) \quad (7)$$

deriving $E(\mu_i)$ with respect to μ_i and denoting to zero, we can have

$$\mu_i = \frac{\sum_{j=1}^{n} z_{ij} x_j}{\sum_{j=1}^{n} z_{ij}} \quad (8)$$

Each cluster updates the centroid according to equation (3) and (8) until the probabilities of the clusters in which the data points are located remain unchanged or the maximum number of iterations are reached.

B. Kernel Density Estimation

For an observation value x of random variable X, the probability that it falls into the interval $[a, b]$ can be computed by

$$P = \int_a^b \hat{p}(x)\, dx \quad (9)$$

where $\hat{p}$ is the probability density function. When $|b - a| \ll \varepsilon$, $\varepsilon \to 0$, equation (9) becomes to

$$P = \hat{p}(x)(b - a) \quad (10)$$

Hence

$$\hat{p} = \frac{P}{b - a} \quad (11)$$

If there are k observation value of n falling into the interval $[a, b]$, the probability will be

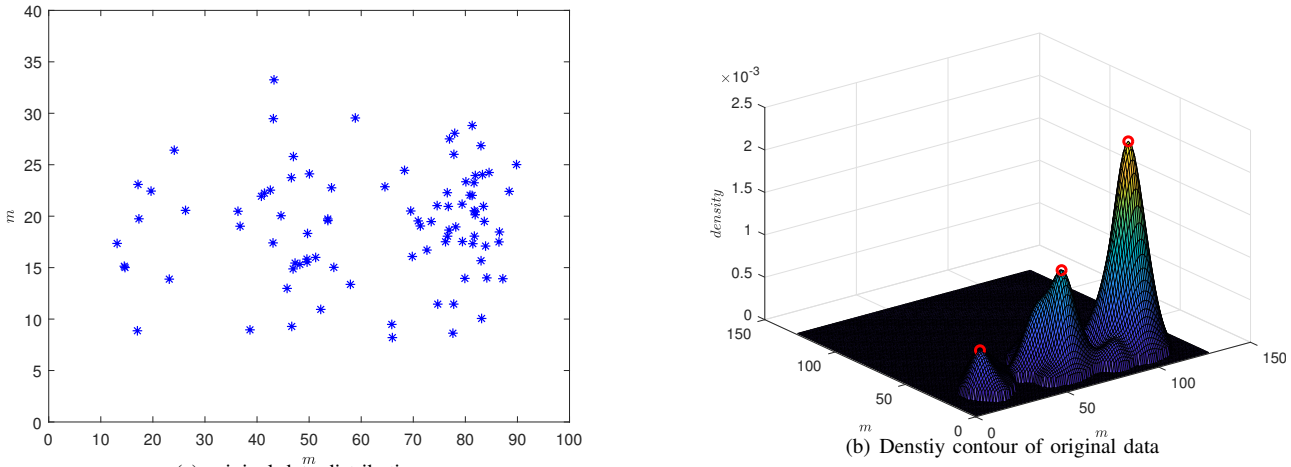
$$P = \frac{k}{n} \quad (12)$$

the probability density function

$$\hat{p} = \frac{k}{n(b - a)} \quad (13)$$

We define the kernel function

$$K(u) = \begin{cases} 1, |u| \le \frac{1}{2} \\ 0, \text{otherwise} \end{cases} \quad (14)$$



(a) original data distribution (b) Denstiy contour of original data

Fig. 1. A example of kernel density estimation

Assuming the center of interval $[a, b]$ is x and $h = b - a$, any sample x_i falling into the interval $[a, b]$ needs to meet the following requirement.

$$|x - x_i| \leq \frac{b-a}{2} \tag{15}$$

$$\frac{|x - x_i|}{h} \leq \frac{1}{2} \tag{16}$$

From equation (14) and (16), we can obtain

$$K\left(\frac{x - x_i}{h}\right) = \begin{cases} 1, |\frac{x-x_i}{h}| \leq \frac{1}{2} \\ 0, \text{otherwise} \end{cases} \tag{17}$$

So, the value k can be expressed by

$$k = \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{18}$$

and then,

$$\hat{p}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{19}$$

which is the kernel density function.

[17]Kernel density is one of the non-parametric estimation methods used to estimate the unknown density function. Each point is covered with a smooth surface. The surface value at the location of the point is the highest and decreases with the increase of distance until the value is zero when the distance equals the search radius, which looks like the density map of Gauss distribution function. Then, the values of kernel density estimation of all points are superimposed and a set of points is transformed into a surface showing continuous density changes. Fig.1 is an example of kernel density estimation for a non-uniform distribution network. The discrete point set is transformed into a smooth density map, shown in Fig.1(b), which shows its spatial distribution. The higher the density value, the higher the aggregation degree of the point is.

IV. Proposed Algorithm

The proposed protocol is divided into many rounds, like LEACH. Each round contains setup phase and steady phase. In this research, we only focus on the setup phase and improve some existing methods. During the setup phase, the sink node gathers the location and residual energy of each node in the whole wireless network. Also, it generates k initial centers by CFSFDP and KDE as the input of ISK-means clustering algorithm, which can avoid the local optimum by traditional K-means clustering method. And then, classification algorithm is implemented. Each node calculates the distance between itself and center points and chooses to join the nearest cluster with a certain probability. At the same time, if the nodes on the edge of two clusters have similar distance to these two cluster centers, they will preferentially join the cluster with smaller density. After clustering, the final CHs are selected according to the energy. The detailed discussion is given below.

A. Energy Model

Because the main energy consumption of the protocol is only for receiving and sending data, the first order radio model as the energy model is in line with the needs. The consumption energy in the transmitter nodes and in the receiver node can be calculated as follows:

$$E_T = \begin{cases} lE_{elec} + l\varepsilon_{fs}d^2 & d \leq d_0 \\ lE_{elec} + l\varepsilon_{mp}d^4 & d > d_0 \end{cases} \tag{20}$$

$$E_R = lE_{elec} \tag{21}$$

where E_{elec} is the dissipated energy per bit in both transmitter nodes and receiver nodes, d is the transmission distance and d_0 is defined as the distance threshold, $d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}$, l is the packet size. The free space ε_{fs} and multipath fading channel ε_{mp} represent the energy consumption parameters of amplifier, which one can be used depends on the distance between the transmitter and the receiver.

If the transmission distance d_{toBS} between the cluster head and the sink node is larger than d_0 and the distance between member node and the cluster head d_{toCH} is less then d_0, the energy consumption of cluster head can be calculated in this round by

$$E_{CH} = klc\left(E_T + E_{DA} + \varepsilon_{mp}d_{toBS}^4\right) + klE_R \tag{22}$$

where E_{DA} represents the dissipated energy of data aggregation and c is the data aggregation ratio. The first part of equation (22) is the energy consumption of the cluster head for sending aggregated data to the sink node and the second part is the energy consumption of receiving and aggregating data of k member nodes. The energy consumption of a non-cluster node sending data to the cluster head by

$$E_{nonCH} = lE_T + l\varepsilon_{fs}d_{toCH}^2 \tag{23}$$

Hence, the residual energy of node i in each round r can be computed by

$$E_i(r) = \begin{cases} E_i(r-1) - [klc\left(E_T + E_{DA} + \varepsilon_{mp}d_{toBS}^4\right) + klE_R] & i \in CH \\ E_i(r-1) - lE_T + l\varepsilon_{fs}d_{toCH}^2 & i \notin CH \end{cases} \tag{24}$$

where $E_i(r-1)$ is the residual energy for node i in the $r-1$ round and CH is the cluster heads.

B. Selection of Initial Cluster Heads

[18] discovered that clusters can be recognized regardless of their shape and the dimensionality of the space in which they are embedded. The basic idea is that cluster centers are surrounded by neighbors with lower local density and they are at a relatively large distance from any points with a higher local density. Hence, the larger the number of nodes in its neighborhood is, the higher the value of local density will be. The cluster heads are selected by the maximum distance σ and relatively high local density p.

In this paper, we use the similar idea of density maximum to select the initial cluster heads. However, we only consider the density parameter p, which can be obtained by KDE. The nodes with highest local density will be chosen as the cluster heads, because the area with the larger number of nodes can easily form the maximum local density in non-uniform distribution wireless networks.

$$\mathrm{N}_{p_{max}} = \max\left(\mathrm{N}_{\hat{p}_i}\right), i \in \text{local region nodes} \tag{26}$$

According to above equation, we can obtain a maximum density node set $\{\mathrm{N}_{p_{max1}}, \ldots, \mathrm{N}_{p_{maxK}}\}$. After the number of maximum density regions, K, is determined, each normal node joins the nearest cluster head with higher density to form initial clusters by CFSFDP algorithm. Then, the nodes with the largest energy in each initial cluster are selected as the input of improved soft K-means clustering algorithm. There are several benefits through the above steps to ensure the initial cluster heads: (1) the number of clusters k is determined by maximum density principle. (2) the distances between the greatest energy nodes of each cluster are relatively large. The detailed description is shown in algorithm 1.

Algorithm 1: Selection of initial cluster heads

Input: the set of n data items X
Output: initial cluser heads set

```
for i = 1, 2,...,n do
    calculate p_i by (19)
end for
obtain density set P = {p_1, ..., p_n}
get local maximum density node set {N_{p_max1}, ..., N_{p_maxK}} by statistical tool as the initial cluster heads
each member node joins its cluster head N_{p_maxK} via CFSFDP algorithm to form K initial clusters {C_1, ..., C_K}
for j = 1 : K do
    calculate the highest energy nodes of C_j
end for
return the set of the highest energy nodes of each cluster μ = {μ_1, ..., μ_K}
```

C. Cluster Formation

Machine learning has been widely used in wireless sensor networks for forming cluster, such as distributed K-means clustering algorithm [19], improved K-means cluster-based routing [20] and LEACH-CKM [21]. The above algorithms are based on the distances between normal nodes and cluster head nodes to form clusters, which can easily lead to a large gap in the number of different clusters in non-uniform distribution networks. Furthermore, the energy consumption of cluster head nodes is not balanced. Hence, compared with these K-means clustering algorithms, our proposed algorithm uses soft K-means clustering algorithm. Each node is assigned a probability of belonging to cluster head rather than completely being a member of just one cluster. Therefore, the nodes close to the boundary of clusters may have the similar probabilities belonging to different clusters. And, we also take one more step after the classification convergence, assigning nodes at the edge of different clusters to join the approximate cluster for balancing the number of normal nodes in clusters. We discuss this problem in two scenarios.

1) Scenario 1: The node N is at the edge of two clusters, shown in Fig. 2. The distance from node N to the center of cluster B is a little bigger than that from node N to the center of cluster A, which means it has a higher probability to join cluster A. However, it is a different story if we consider the densities of different clusters. Cluster A has 5 member nodes and cluster B has 10 member nodes. Assuming that all normal nodes send messages to its CH in each round, CH of cluster B will deal with more information from its member nodes. In order to balance the energy consumption of CHs, it is better for node N to join cluster A. Here, we give a simple definition of re-assigning nodes. When the difference of probability of a node belonging to two clusters is less than 25%, the node will join the cluster with low density.

2) Scenario 2: The node N is at the boundary of three or more clusters, shown in Fig.3. After clustering, each node will have a probability set representing the possibilities of belonging to different clusters, $\boldsymbol{z}_i = \{z_{i1}, z_{i2}, \ldots z_{iK}\}$. The cluster of the node belonging to with low probability will not be considered. So, we only choose the first two maximum probabilities and calculate their difference, D_i. If $D_i < 10\%$, the node will join the cluster with smaller density between two maximum probability clusters. As shown in Fig. 3, node N

has similar probability belonging to cluster A, B and C, which finally joins cluster C because cluster C has lower density than cluster A and B.

Algorithm 2: Cluster formation

Input: Initial cluster heads $\boldsymbol{\mu} = \{\mu_1, \ldots, \mu_K\}$ from **algorithm** 1, Data items $\mathbf{X} = \{x_1, x_2, \ldots, x_n\}$, Iterations r_{max}

Output: K clusters

for r = 1 : r_{max} **do**
  for i = 1 : n **do**
    $z_{ik} = \frac{e^{-\beta||x_i-\mu_k||^2}}{\sum_K e^{-\beta||x_i-\mu_k||^2}}$
  end for
  $\boldsymbol{z} = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1K} \\ \vdots & \vdots & \vdots & \vdots \\ z_{i1} & \cdots & \cdots & z_{iK} \end{bmatrix}$
  for k = 1 : K **do**
    $\mu_k = \frac{\sum_{i=1}^{n} z_{ki} x_i}{\sum_{i=1}^{n} z_{ki}}$
  end for
end for
K clusters $\boldsymbol{C} = \{C_1, \ldots, C_K\}$
for i = 1 : n **do**
  check z_i and reassign the node i by scenario 1 and 2
end for
return K clusters $\boldsymbol{C}_{new} = \{C_{new1}, \ldots, C_{newK}\}$

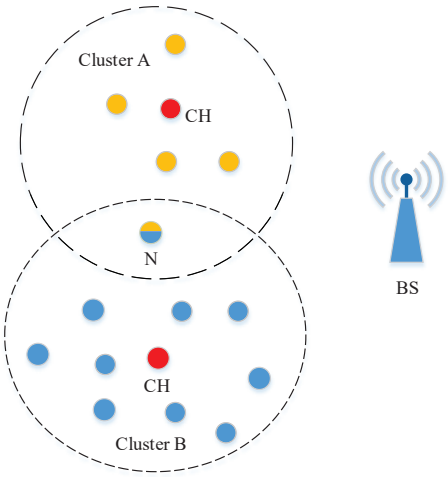


Fig. 2. Node at the boundary of two clusters

D. Selection of Final Cluster Heads

In non-uniform distributed wireless sensor networks, the cluster size will be different. If only one CH is selected for each cluster, CH will consume too much energy to deal with the information from its member nodes in the large cluster, which will cause its death too early. Hence, our proposed algorithm designs the scheme of multi-cluster heads. The number of cluster head nodes is not fixed in each cluster, which is determined by the number of the nodes in the cluster. The larger of the number of nodes in the cluster is, the more the

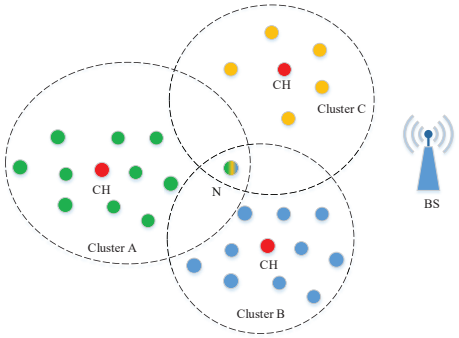


Fig. 3. Node at the boundary of three clusters

number of CH will be. In a cluster, the nodes which have more remaining energy and close to the center are preferentially selected as CHs. The total remaining energy of cluster k can be computed by:

$$E(k)_{total} = \sum_{i=1}^{S_k} E_i(r) \tag{27}$$

where S_k is the size of cluster k, $E_i(r)$ is the residual energy of node i in current round, which can be obtained by equation (24).

The average energy of cluster k is:

$$E_{ave} = \frac{E(k)_{total}}{S_k} \tag{28}$$

This scheme can balance the energy consumption of cluster heads of different cluster in non-uniform distributed wireless sensor networks, shown in Fig.4.

Algorithm 3: Selection of final cluster heads

Input: $\boldsymbol{C}_{new} = \{\mathrm{C}_{new1}, \ldots, \mathrm{C}_{newK}\}$

Output: K cluster heads

for k = 1:K **do**
  Calculating the size of cluster C_{newk}, S_k
  Calculating average energy of cluster E_{ave}
  $Num = \frac{S_k}{constant}$, the number of CHs of cluster
  for i = 1 : S_k **do**
    Iterating x_i from near the center of cluster
    if $E_i > E_{ave}$ and $Num > 0$ **then**
      $Num = Num - 1$
      $\mathrm{CH}_k(Num) = x_i$
    end if
  end for
end for
return $\boldsymbol{CH} = \{\mathrm{CH}_1, \ldots, \mathrm{CH}_K\}$, K cluster heads

E. Switching Cluster Head

In the first round, the first node in CH_k is selected as the current CH for cluster k. After CHs for the current round are selected, the sink node notifies all nodes to join the cluster to which they belong. CHs broadcast TDMA schedules to their member nodes for transmitting data in different time slots to

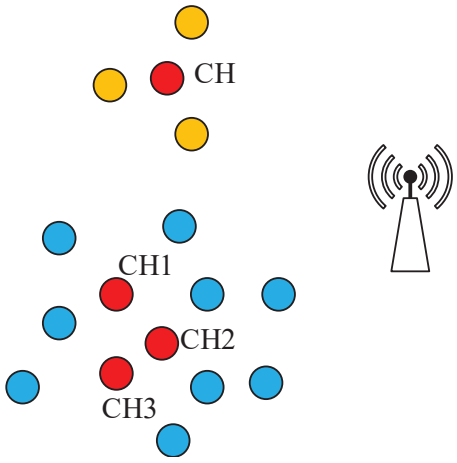


Fig. 4. Multi-heads scheme

avoid data collision. Then, the network is in a steady phase and begins to exchange data between normal nodes and their CHs. For balancing the energy consumption of CH, when the energy of current CH of any cluster is below the threshold value, the next candidate CH in that cluster is enabled. Until all CH in the cluster are executed, the algorithm starts re-clustering.

Algorithm 4: Selection of final cluster heads

Input: K cluster heads $\boldsymbol{CH} = \{\mathrm{CH}_1, \ldots, \mathrm{CH}_K\}$
Output: Next cluster head

```
Current round
for k = 1 : K do
    ρ = (the residual energy of CH_k last round) / (the residual energy of CH_k current round)
    if ρ < Threshold then
        if CH_k has next then
            switching next CH
        else
            reclustering
        end if
    end if
end for
```

V. Experiment Results and Analysis

A. Simulation Settings

The simulation is executed in MATLAB R2017a. Sensor nodes are deployed in an area of $100 \times 100m^2$, the sink node is located at $(50m, 150m)$. The main parameters are shown in Table I.

TABLE I
Simulation Parameters

Parameter	Value
Area	$100m \times 100m$
Sink node	(50,150)
Initial energy	0.2J,1J
Packet length	4000bits
Control length	100bits
E_T	50nJ/bit
E_R	50nJ/bit
ε_{fs}	10pJ/bit
ε_{mp}	0.0013pJ/bit
E_{DA}	5nJ/bit
d_0	88m
Number of sensor nodes	26,100

B. Reassigning Nodes of Improved Soft K-means Analysis

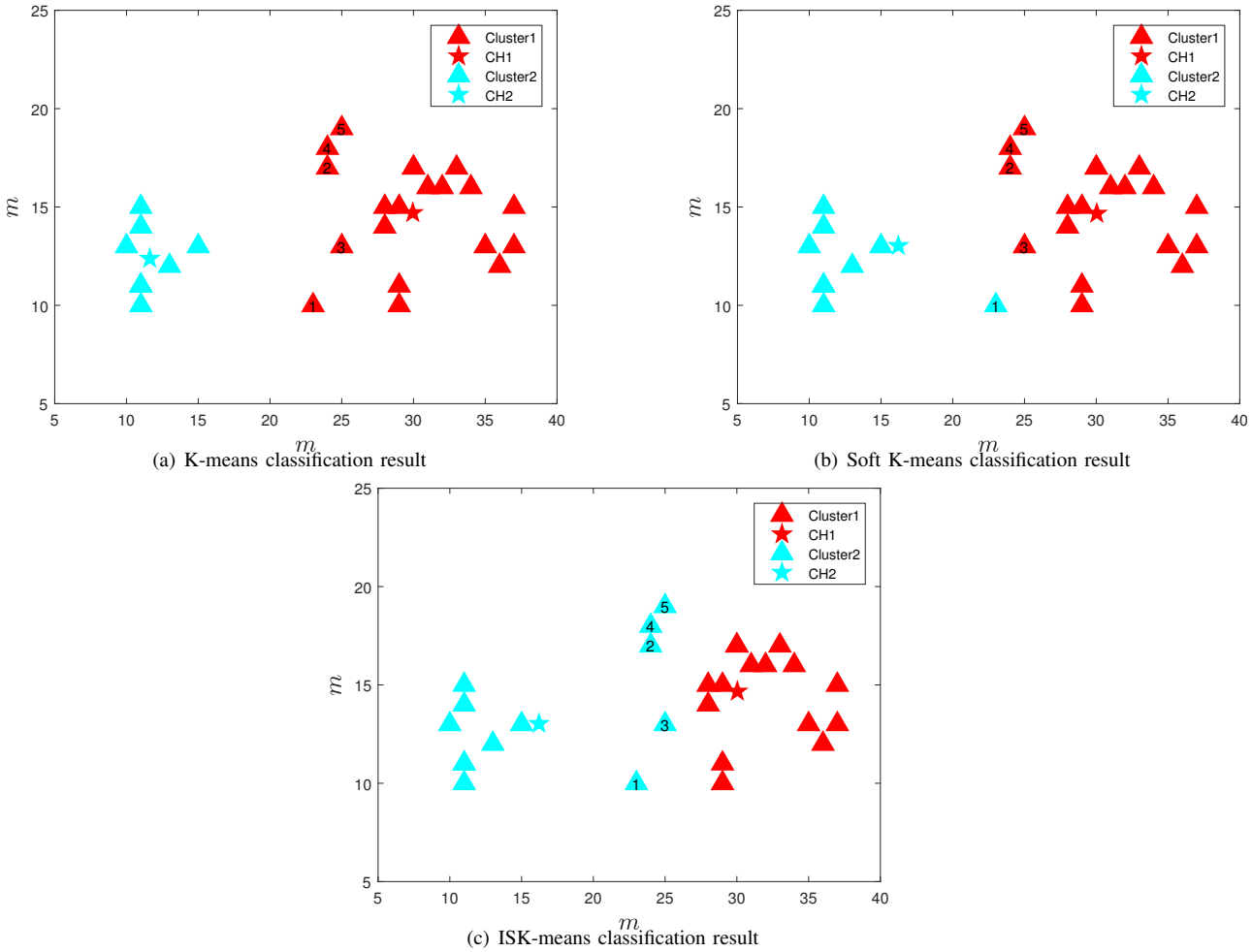
In this section, we will discuss the impact of reassigning nodes scheme of improved soft K-means for balancing energy consumption of CH. A non-uniform distributed network with 26 nodes is generated. Firstly, we use K-means classification method to classify these nodes and get two clusters, shown in Fig.5(a). It is found that cluster 1 contains 19 nodes, which is much higher than cluster 2. As a result, CH of the heavily loaded cluster, which contains 26 nodes will be exhausted much earlier than another cluster. Fig.5(b) shows the similar classification result by soft K-means method when β is 0.3, which can not balance the number of different clusters very well. However, we find that the nodes on the edge of two clusters have similar probabilities belonging to these two clusters, like node 1, node 2, node 3, node 4 and node 5. At the same time, when the value of β changes, the probabilities also change. Table II lists the probabilities of the above nodes belonging to different clusters and their change for different β. In section 1.2, we know β represents stiffness parameter, which is the tightness of node belong to a cluster. As can be seen from Table II, when $\beta = 1$, all five nodes belong to the cluster with a higher probability than the case, $\beta = 0.3$. Hence, the larger the value of β, the closer the nodes is associated with a cluster. We set $\beta = 0.3$ in the simulation for using the ISK-means algorithm better. According to the principles described in scenario 1, node 2, node 3, node 4 and node 5 are reassigned to cluster 2 from cluster 1, which balances the number of these two clusters. The residual energy of CH, computed by equation (24), in each round could be used to check the result and advantage of this scheme. Fig.6(a) and Fig.6(b) show that both K-means and soft K-means result in unbalanced energy consumption of CH, however, the ISK-means method achieves an equilibrium of energy consumption in both CH.

TABLE II
Probabilities Comparison

Probability		Node 1	Node 2	Node 3	Node 4	Node 5
$\beta = 0.3$	Cluster 1	0.4852	0.5537	0.5684	0.6120	0.6120
	Cluster 2	0.5148	0.4463	0.4316	0.3880	0.3880
$\beta = 1$	Cluster 1	0.0438	0.9787	0.9860	0.9992	0.9992
	Cluster 2	0.9562	0.0213	0.014	0.0008	0.0008

C. Network Lifetime

To test the performance of ISK-means, we compare it with LEACH [**?**] and K-means [**?**]. Fig.7 shows the first node death(FND), half of nodes death(HND) and the last node death(LND) for three methods when the number of nodes is 100 and the area is $100m \times 100m$. If the protocol can balance energy very well, the first node death will be very late. In this paper, we assume that the death of 95% nodes means all nodes

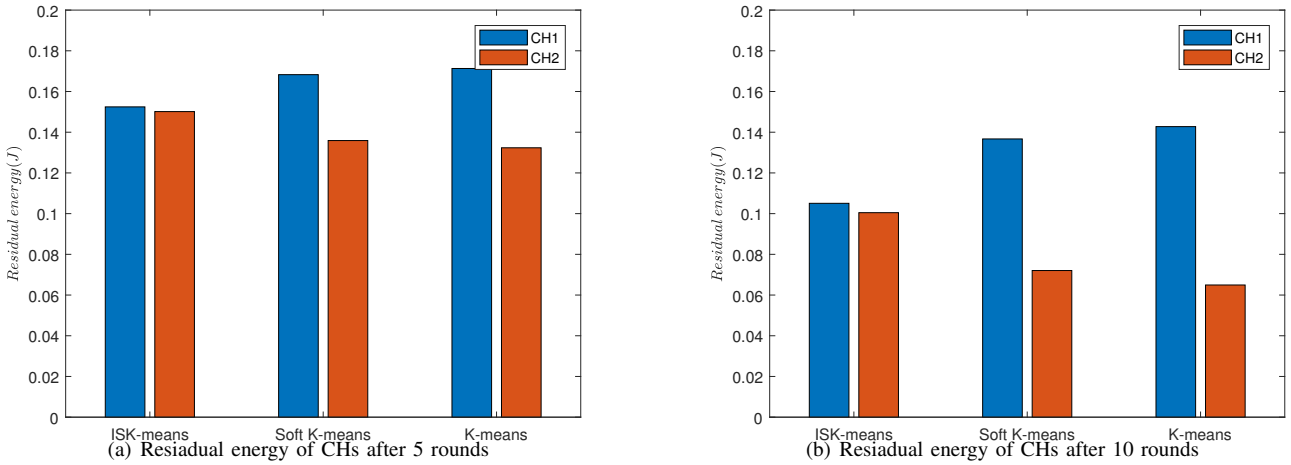


(a) K-means classification result

(b) Soft K-means classification result

(c) ISK-means classification result

Fig. 5. Comparison of different classification methods



(a) Resiadual energy of CHs after 5 rounds

(b) Resiadual energy of CHs after 10 rounds

Fig. 6. Comparison of residual energy of CHs

dead. In view of Fig.7 and Fig.8, the sensor nodes lifetime of our proposed algorithm is better than that of LEACH and K-means. For K-means, the round of FND is 191, which is much earlier than 962 in LEACH and 2663 in ISK-means. However, the LND in K-means happens later compared with LEACH. It is obvious that the energy consumption of K-means is unbalanced, this is because K-means algorithm selects the initial cluster heads randomly, which is sensitive to noise and abnormal data, especially in non-uniform distributed wireless networks. The proposed ISK-means algorithms takes energy into account. The result shows it can effectively postpone the FND, HND and LND. The FND of ISK-means is 2663, which

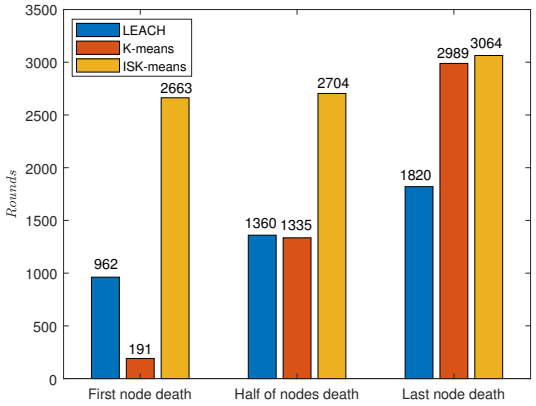


Fig. 7. Comparison of network lifetime

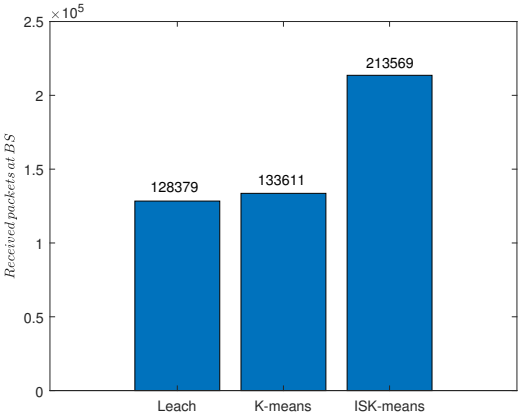


Fig. 9. Received packets at BS

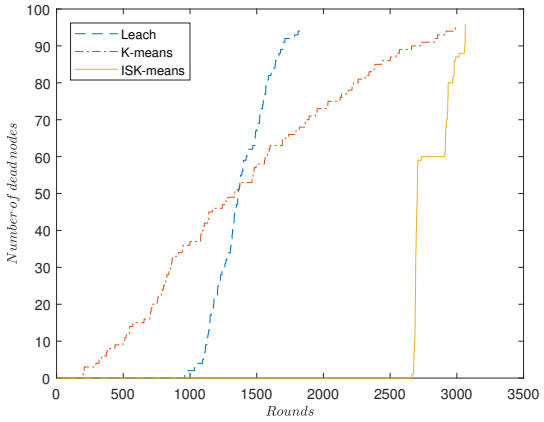


Fig. 8. Comparison of node death curve

is about 3 times that 962 of Leach and 14 times of K-means. The HND is also around 2 times of both LEACH and K-means, which means ISK-means can balance the energy consumption of the network effectively. Obviously, ISK-means can keep most of the nodes alive in the network, so that the existence of the death of first nodes happens later.

D. Received Packets at BS

Fig.9 shows the received packets at BS for among three algorithms. Because of using multi-heads scheme in ISK-means, the number of re-clustering decreases, which reduces the communication cost of clustering between CHs and their member nodes. As a result, data transmission frequency increases and more packets are successfully transmitted to the BS compared to that in LEACH and K-means.

E. Energy Variance

Fig.10 shows the comparison of residual energy of all 100 nodes in the network for among three algorithms after different rounds. It is found that the energy distribution curve of all nodes by using ISK-means protocol is smoother than that of LEACH and K-means, and the curve of K-means is the worst. This result demonstrates that ISK-means is good at balancing energy consumption of all nodes in the whole network. For purpose of estimating the performance of proposed algorithm, we denote a new parameter: energy variance(EV), can be expressed by

$$\sigma^2 = \frac{\sum_{i=1}^{n} \left(E\left(x_i\right) - \overline{E}\right)^2}{n} \tag{29}$$

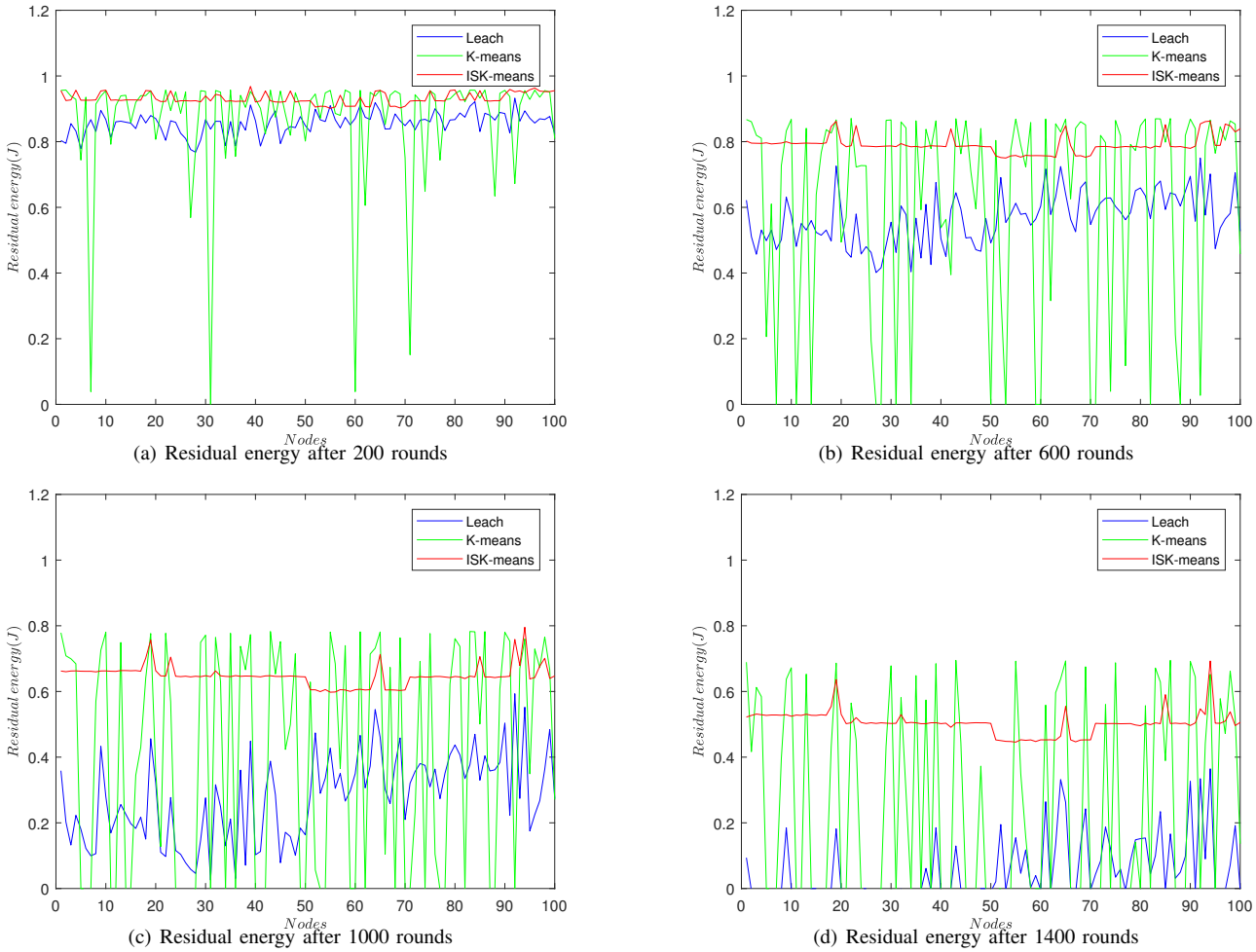
where $E\left(x_i\right)$ is the energy of node i in current round and $\overline{E}$ is the average energy of all nodes. In table III, it clearly reveals that ISK-means obtains smaller variances than LEACH and K-means in different rounds, which demonstrates that ISK-means can keep the energy distribution of 100 nodes in the network to be the most uniform.

TABLE III
COMPARISON OF ENERGY VARIANCE IN DIFFERENT ROUNDS

	Variance						
	200 r	400 r	600 r	800 r	1000 r	1200 r	1400 r
LEACH	0.0011	0.003	0.0062	0.0108	0.017	0.017	0.0088
K-means	0.0345	0.0752	0.1043	0.1194	0.1158	0.1028	0.0907
ISK-means	0.00028	0.0005	0.0007	0.00083	0.001	0.0011	0.0013

VI. CONCLUSIONS

In this paper, we propose an energy balanced ISK-means algorithm protocol based on soft K-means for non-uniform distributed wireless sensor networks. Firstly, it optimizes the selection of initial cluster heads of soft K-means clustering method by CFSFDP and KDE algorithms and a better cluster formation is obtained. Secondly, in order to balance the size of different clusters in non-uniform networks, we use the soft classification characteristics of soft K-means to reassign some nodes locating at the boundary of different clusters to smaller size cluster. Furthermore, multi-heads scheme is used in the selection of final cluster heads, which can effectively balance the traffic load of cluster heads, reduce the number of re-clustering and save communication cost. Experiments have demonstrated that ISK-means algorithm can balance energy very well for all nodes in the network during the period of network survival and the amount of data transmitted to the BS is increased remarkably.



(a) Residual energy after 200 rounds

(b) Residual energy after 600 rounds

(c) Residual energy after 1000 rounds

(d) Residual energy after 1400 rounds

Fig. 10. Comparison of residual energy distribution

REFERENCES

[1] J. Yan, M. Zhou, and Z. Ding, "Recent advances in energy-efficient routing protocols for wireless sensor networks: A review," *IEEE Access*, vol. 4, pp. 5673–5686, 2016.

[2] S. K. Singh, P. Kumar, and J. P. Singh, "A survey on successors of LEACH protocol," *IEEE Access*, vol. 5, pp. 4298–4328, 2017.

[3] S. Zafar, A. Bashir, and S. A. Chaudhry,"Mobility-aware hierarchical clustering in mobile wireless sensor networks," *IEEE Access*, vol. 7, pp. 20394–20403, 2019.

[4] P. Sasikumar and S. Khara, "K-means clustering in wireless sensor networks," in *Proc. IEEE CICN*, 2012, pp.140–144.

[5] I. Quchan, "Towards energy efficient K-means based clustering scheme for wireless sensor networks," *International Journal of Grid and Distributed Computing*, vol. 9, no. 7, pp. 265–276, 2016.

[6] A. S. D. Sasikala and N. Sangameswaran, "Improving the energy efficiency of LEACH protocol using VCH in wireless sensor network," *International Journal of Engineering Development and Research*, vol. 3, no. 2, pp. 918–924, 2015.

[7] E. Rabiaa, B. Noura, and C. Adnene, "Improvements in LEACH based on K-means and Gauss algorithms," *Procedia Computer Science*, vol. 73, pp. 460–467, 2015.

[8] C. Bauckhage, "Lecture notes on data science: Soft k-means clustering," 2015.

[9] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan , "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. HICSS*, USA, Jan. 2000.

[10] W. R. Heinzelman *et al.*, "New clustering scheme for wireless sensor networks," in *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002.

[11] M. Mohamed-Lamine, "An application-specific protocol architecture for wireless microsensor networks," in *Proc. IEEE WoSSPA*, May 2013, pp. 487–491.

[12] S. Randhawa and S. Jain, "Performance analysis of leach with machine learning algorithms in wireless sensor networks," *International Journal of Computer Applications*, vol. 147, no. 2, pp. 7–12, 2016.

[13] L. Tan, Y. Gong, and G. Chen, "A balanced parallel clustering protocol for wireless sensor networks using k-means techniques," in *Proc. IEEE SENSORCOMM*, Aug. 2008, pp. 300–305.

[14] A. Mahboub, M. Arioua et al., "Energy-efficient hybrid k-means algorithm for clustered wireless sensor networks," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 4, p. 2054–2060, Aug. 2017.

[15] A. Ray and D. De, "Energy efficient clustering protocol based on k-means (eecpk-means)-midpoint algorithm for enhanced network lifetime in wireless sensor network," *IET Wireless Sensor Systems*, vol. 6, no. 6, pp. 181–191, 2016.

[16] W. H¨ardle, "Applied Nonparametric Regression, ser. Econometric Society Monographs," Cambridge University Press, 1990.

[17] T. Buch-larsen, J. P. Nielsen, M. Guill´en, and C. Bolanc´e, "Kernel density estimation for heavytailed distributions using the champernowne transformation,"*Statistics*, vol. 39, no. 6, pp. 503–516, 2005.

[18] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks,"*Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[19] J. Qin, W. Fu, H. Gao, and W. X. Zheng, "Distributed k -means algorithm and fuzzy c -means algorithm for sensor networks based on multiagent consensus theory," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 772–783, Mar. 2017.

[20] M. Lehsaini and M. B. Benmahdi, "An improved k-means cluster-based routing scheme for wireless sensor networks," in *Proc. IEEE ISPS*, Apr. 2018, pp. 1–6.

[21] D. Mechta, S. Harous, I. Alem, and D. Khebbab, "Leach-ckm: Low energy adaptive clustering hierarchy protocol with k-means and mte," in *Proc. IIT*, Nov. 2014, pp. 99–103.