A Data-aggregation Scheme for WSN based on Optimal Weight Allocation

Pinghui Zou and Yun Liu*
School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China
*Corresponding author, Email: 08111002@bjtu.edu.cn, liuyun@bjtu.edu.cn

Abstract—Since the measuring accuracy and environment of each sensor are different, there must exist difference in the correctness of measurement. If the testing data is not processed and utilized with distinction, it will cause imprecision in the testing results and lead to errors of the system. It is necessary to selectively distinguish the importance among the sensors, contraposing the situation of each sensor in the testing system and the accuracy of tests. So the related concepts of data aggregation technology in wireless sensor networks and the aggregation algorithm performance evaluation criteria are introduced. The core problem in WSN, aggregation operation for sensing data, is studied deeply. The problems in node data group when the distributed clustering technology is implemented to WSN are also analyzed. Then a distributed K-mean clustering algorithm based on WSN is proposed. On the basis of this improved algorithm, we realize a network data aggregation processing mechanism based on adaptive weighted allocation of WSN. DKC algorithm is mainly used to process the testing data of bottom nodes. When reducing the data redundancy it can provide more accurate field testing information and system status information. It can make rapid packet for the network nodes. The packed data will be used to provide correct judgement, according to the size of its corresponding weight, to acquire more reasonable results. The experiments have demonstrated that our method can greatly decrease the data redundancy of WSN and save large amount of storage resources. The network bandwidth consumption is also reduced. So this scheme has high efficiency and good scalability.

Index Terms—Data Aggregation; Self-Adaptive Weighted; K-Means Clustering; DKC; WSN

I. INTRODUCTION

Wireless Sensor Networks (WSN) is made up of lots of small-scaled sensor equipments and it is a self-organized network [1] which is deployed to a designated area, to supervise and perceive the task through limited wireless communication mode.

As the ability of node communication is limited, WSN can only work in a relatively smaller scope. To increase the accuracy and authenticity of target monitoring, the sensory area of these nodes must be mutually overlapped. Data aggregation technology [2] uses this feature to transmit collected data of each node to other nodes or base station, which will reduce the count of data and delete redundancy. At present, data aggregation algorithms of wireless sensor network are mainly based on traditional Client/Server (C/S) model [3, 4], but this model has many problems:

(1) Network delay and energy consumption is large. Each sensor node can simultaneously transmit data to processing node and processing node can only receive data in a given order. When sensor node number is increasing and the total quantity of sensor data is enlarged, network delay and energy consumption are also increasing.

(2) The scalability is inefficient. When WSN supplements new sensor nodes, the network structure often needs to adjust to keep load balance.

(3) The energy consumption of nodes in unbalanced. Since nodes processing need connection of each sensor node to deal with its data, the sensor nodes will consume more energy. That needs to present super powerful energy nodes to be taken as processing data or adopt an algorithm to process the nodes in rotation, which brings additional network overhead.

Towards above problems, reference [5] provides Local Closest First (LCF) heuristic Mobile Agent (MA) data aggregation routing algorithm. When selecting the distance, the most approximate node of current sensor node is taken as the next node. It performs data aggregation on each accessing sensor node. However, it is only suitable for small-scaled sensor network in simple environment. When the network scale is constantly expanding and the distribution of sensor node is more complicated, heuristic algorithm can obtain the local optimum MA routing intelligently. Reference [6] uses Genetic Algorithm (GA) [7] to solve the local optimum MA routing of the minimized network energy. However, its aggregation energy is the function of time, instead of data size. Therefore, we cannot explain that data transmission quantity is reduced because of data aggregation, which also reduces transmission energy to approach the purpose of saving energy. Reference [8] proposes a WSN data aggregation algorithm named AGA based on adaptive genetic algorithm. The algorithm summarizes MA routing as an optimized problem. By means of grid method, AGA is applied to solve the optimal routing node sequence for mobile agents [9, 10]. Currently, wireless sensor network data aggregation technology based on Client/Server (C/S) model and MA is gradually developing in universities and research institutions. However, their basis for research is mainly established on current foreign algorithms. As one of key
technologies on wireless sensor network, data aggregation still has many problems for us to study. Contrapositing the defects in existing technology, we propose a fast packet method for the sensing data in WSN based on K-mean clustering algorithm. By analysis on current distributed clustering problems and research on clustering processing in WSN, a Distributed K-mean Clustering (DKC) method for WSN is proposed. On the basis of DKC, an in-network data aggregation algorithm based on the optimal weight distribution is also put forward. DKC algorithm can realize fast and rational packet for nodes transmission in WSN. However, the data aggregation method based on adaptive weight allocation can make effective and correct judgment with the weighted value of sensor data. The experiments results have shown that our scheme is simple and practical. It greatly reduces the data redundancy of the whole network, which also has higher efficiency and better scalability.

II. DISTRIBUTED K-MEAN CLUSTERING IN WSN

A. Process of K-mean Clustering Method

When a dataset contains \( n \) data objects and the number of clusters is \( k \), the object of clustering algorithm is finding an algorithm for division to divide the data object into \( k \) \( (k \leq n) \) partitions [11]. Each division represents a cluster. Generally, a guideline will be adopted, such as distance. The best known and most commonly used methods are K-means or K-center and their variants [12-16]. The K-means clustering algorithm based on centroid take \( k \) as parameter to divide \( n \) objects into \( k \) clusters. It makes relatively higher similarity inside the cluster, while the similarity among the cluster is lower. The calculation of similarity is based on the mean of object in one cluster. K-means clustering method is an unsupervised learning algorithm. First it randomly selects \( k \) objects and each object represents a mean value or centroid of a cluster initially. For the other objects, they will be assigned to the nearest cluster according to the distance with the center of cluster. Then the mean value of each cluster is recalculated. This process is repeated continuously until the criterion function is converged. Commonly the formula \( E = \sum_{i=1}^{k} \sum_{m \in C} |p - m|^2 \) is used for judgement. \( E \) is the sum of squared errors of all the objects.; \( p \) is time point which denotes the given data object; \( m \) is the mean value of cluster \( C \). This criterion tries to make the generated cluster compact and independent. The whole procedures can be summarized as the follows:

1. Select \( k \) objects by computer randomly or artificially as the initial clustering center: \( Z_i(1), Z_2(1), \ldots, Z_k(1) \).
2. Compute the squares of distance for any two points;
3. According to the result of above procedure find new cluster center \( Z_i(j+1), Z_2(j+1), \ldots, Z_k(j+1) \).

B. Process of Distributed Data

(a) Distributed clustering problem

We assume there are \( p \) nodes \( N_1, N_2, \ldots, N_p \) in a WSN system. \( X = X^{(1)} \wedge X^{(2)} \wedge \ldots \wedge X^{(p)} \) is the whole data set. \( X^{(i)} \) is the subset of \( X \) \((i = 1,2,\ldots, p)\) and it denotes the subset of data in node \( N_i \). Set \( X^{(i)} = \{x_{i1}, x_{i2}, \ldots, x_{in}\} \) as the set of \( n \) data nodes which are perceived by the sensor in node \( i \). The target is dividing each dataset \( X^{(i)} \) into \( K \) cluster with k-means clustering algorithm and keeping it consistent with the global clustering of set \( X \). That means that if \( X^{(i)} \) is the subset of \( X \) \((i = 1,2,\ldots, p, j = 1,2,\ldots, K)\), then \( X^{(i)} \) contains the part of data nodes belonging to cluster \( j \) in \( X \), after k-means clustering algorithm is implemented. Therefore we have \( X_i = X_i^{(1)} \cup X_i^{(2)} \cup \ldots \cup X_i^{(p)} \).

(b) Treatment in WSN

The nodes in WSN are connected with Ad Hoc and each node can only communicate with neighbour nodes in its range [17]. In addition, sending data will consume more energy than its own consumption. So the data transmission should be avoided as few as possible. We need to make full use of the computation resource of the node itself, cooperating each other to complete the aggregation of the whole network. We assume there are \( m \) nodes in WSN which are used to inspect the data and each node creates data flow \( S_j \). There is a node sin \( k \) which is responsible for each node in the network to send and receive messages. It has stronger processing ability, which can communicate with outside users by wired or wireless means.

The process of algorithm is described in detail: First node sin \( k \) randomly extract \( k \) records from the data in WSN as initial centroid of the data cluster to be divided. Tree communication structure is adopted to issue the information of \( k \) centroids to each node. Each node will divide local data into \( k \) clusters according to these centroids and send them to the parent nodes. When the parent nodes receive the information of all the sub-nodes, they will combine the local information with the cluster information and continue to transmit it to upper node, until node sin \( k \) received all the information of sub-nodes. Then sin \( k \) combines the information of \( k \) clusters and recalculate the mean value of each cluster. If it is not convergent then new \( k \) centroids will be calculated and be issued to the network for iteration. If it is convergent then the centroids of global \( k \) clusters will be created.

In the latter simulation of our system, all the messages are spread by broadcasting, which can be received by the nodes belonging to the effective transmission radius of sending nodes. The system designs a simple MAC protocol which satisfies random delay, carrier sense and
timing delay. The length of packet is not beyond the maximum data length that can be transmitted in one time. During the process of sensing data group a relative simply message format is used, without adopting the general packet heads. If the system need to be expanded in future, then more complicated network transmission control strategy will be used. That needs to add corresponding packet heads to the protocol. When the node data is grouped the exchanged message are mainly consist of two parts: message type and data. The message processing layer explains the message segment according to the different types of message data. The length of message segment is different due to the different length of message types. The network constructing request uses initial message InitNetwork. During clustering sink sends message Clustering and each node sends using message of k clusters LocalCluster. When the clustering results are stable sink will send message EndMining to terminate the process. The following program provides the data structure of main message:

```
typedef struct Clustering {
  int k; // number of clustering to be divided
  int centroid[k,p]; // k centroids
};
typedef struct Cluster {
  int clustered; //id of node sending messages
  int num; // number of data nodes in cluster
  int sum[p]; //cumulative sum of each node in the cluster
  int sumd[p]; // squares of distance of each node to the centroid
};
typedef struct LocalCluster {
  int nodeid; // id of node sending messages
  int k; // number of clusters
  int sumd[k][p]; //k c
};
```

C. Distributed K-means Clustering Algorithm for WSN

We provide the distributed K-means clustering algorithm for WSN as the follows. We adopt the Euclidean distance as dissimilarity function. Assume \( i = (x_1, x_2, ..., x_p) \) and \( j = (x_1, x_2, ..., x_p) \) is two objects in dataset and each object has \( p \) attributes. Then the distance between them is

\[
d(i, j) = \sqrt{|x_1 - x_j|^2 + |x_2 - x_j|^2 + ... + |x_p - x_j|^2}.
\]

Input: Initial centroids of \( k \) cluster and the number of clustering

Output: Centroid of \( k \) cluster

Methods:

Step 1: Node \( \text{sink} \) randomly select \( k \) as the initial centroid of \( k \) cluster to be divided, which are sent to each node. The initial sum of square error is set as a bigger number;

Step 2: Each node calculates the distance to \( k \) centroids of every data node in the local dataset, which are divided into \( k \) clusters;

Step 3: Beginning from the leaf node, each node in the same layer will send the information of local cluster to its parent node, including the number of data nodes, cumulative sum of each data node and the squares of distance from node to centroid;

Step 4: When the information of all sub-nodes is received, it will be combined with the information of \( k \) clusters by each parent node, to create new information. After the nodes at the same layer are processed, that will be sent to upper layer;

Step 5: When the information of all the sub-nodes is received by \( \text{sink} \), the information of \( k \) clusters are combined to compute the sum of squared error. If the result is bigger than old value, then it will be replaced by old value. The mean value of object in each cluster will be recalculated to create new \( k \) centroids, which are sent to each node.

Step 6: Repeat step 2-5, if the sum of squared error is smaller than old value, then the clustering results are stable. \( \text{sink} \) will send EndMining message and compute the mean value, so as to create the final centroid of \( k \) clusters and output.

III. DATA AGGREGATION ALGORITHM FOR WSN BASED ON OPTIMAL WEIGHT ALLOCATION

A. Principle Idea

After the process of DKC, the nodes in WSN will acquire relative correct packet groups. However, there are always noises in the data measured by sensors. So the estimated value acquired by testing data has estimation error. Because the estimation error is a random quantity, Mean Squared Error is usually taken as the evaluation indicators to evaluate a estimated algorithm. For the testing data with multiple nodes, since the measuring accuracy and environment of each sensor are not the same, there must exist difference in the correctness of measurement. If multiple sensors are equally treated and the testing data are processed without discrimination, it will bring big errors to the testing results and system operating results. So the importance of the sensors should be differentiated, which is the basis of optimal weight allocation algorithm for data aggregation.

To balance different accuracy of each data, we introduce the feature number weight \( W \) which signs the measuring accuracy to represent the relative importance of each measuring data, in ranging accuracy measurement. The weight should be proportional with the size of data error. According to accuracy the testing data are multiplied by the weight and averaged to raise the accuracy of testing results. In our study, the self-adaptive weighted data-aggregation method is mainly used to process the testing data of bottom nodes. When decreasing the data redundancy it can provide more accurate information for test and system. As is shown in figure 1, the sensors of each node correspond to different weight value. Under the optimal condition that the total MSE is the minimum, according to the measuring value of each sensor the optimal weight value can be found in a self-adaptive way. The aggregated value of \( X \) can get the optimum [18].
The corresponding minimum MSE is:
\[ \sigma_{min}^2 = 1/k \sum_{p=1}^{n} \sigma_p^2 \]  

(6)

Above estimations are based on the measuring value at some hour of each sensor. When \( X \) is a constant, we can provide estimation according to the average value of historical data. If
\[ \hat{X}_p(k) = \frac{1}{k} \sum_{i=1}^{k} X_p(i), \quad p = 1, 2, ..., n \]  

(7)

Then the estimation value is
\[ \hat{X} = \sum_{p=1}^{n} W_p X_p(k) \]  

(8)

The total MSN is
\[ \sigma^2 = E[(X - \hat{X})^2] \]
\[ = E[\sum_{p=1}^{n} W_p^2 (X - X_p(k))^2] + 2 \sum_{p=1}^{n} \sum_{q=1, q \neq p}^{n} W_p W_q (X - X_p(k))(X - X_q(k)) \]  

(9)

Similarly, \( X_1(k), ..., X_n(k) \) must also be the unbiased estimation value of \( X \), so
\[ \sigma^2 = E[\sum_{p=1}^{n} W_p^2 (X - X_p(k))^2] \]
\[ = \frac{1}{k} \sum_{p=1}^{n} W_p^2 \sigma_p^2 \]  

(10)

Obviously, when \( \sigma^2 \) is obtained the minimum value its corresponding optimal weighted factor \( W_p \) still satisfies formula 5, the minimum MSE is
\[ \hat{\sigma}_{min}^2 = 1/k \sum_{p=1}^{n} \frac{1}{\sigma_p^2} = \sigma_{min}^2 / k \]  

(11)

From formula 11 we can see, because \( k > 1 \), \( \hat{\sigma}_{min}^2 \) must be smaller than \( \sigma_{min}^2 \) and the value of \( \sigma_{min}^2 \) will decrease with the increasing of \( k \).

B. Computation of Variance \( \sigma_p^2 \)

As is known from above analysis, the optimal factor \( W_p \) is determined by the variance \( \sigma_p^2 \) of each sensor, which is unknown generally. We can obtain it according to provided measuring value of each sensor and corresponding algorithms.

We assume any two different sensor \( p \) and \( q \), whose measuring value is \( X_p \) and \( X_q \). Their corresponding observing error is \( V_p \) and \( V_q \).
\[ X_p = X + V_p, \quad X_q = X + V_q \]  

(12)
In formula 12, \( V_p \) and \( V_q \) represent zero mean stationary noise, so the variance of sensor \( p \) is

\[
\sigma_p^2 = \mathbb{E}[V_p^2]
\]  
(13)

Since \( V_p \) and \( V_q \) are not correlated each other and their mean is 0. So the cross-correlation coefficient \( R_{pq} \) of \( X_p \) and \( X_q \) satisfy the follows.

\[
R_{pq} = \mathbb{E}(X_p X_q) = \mathbb{E}(X^2)
\]  
(14)

The autocorrelation coefficient \( R_{pp} \) satisfies

\[
R_{pp} = \mathbb{E}(X_p X_p) = \mathbb{E}(X^2) + \mathbb{E}[V^2]
\]  
(15)

Formula 15 pluses 14 we get

\[
\sigma_p^2 = \mathbb{E}[V_p^2] = R_{pp} - R_{pq}
\]  
(16)

The value of \( R_{pp} \) and \( R_{pq} \) can be acquired by its estimation of time domain. Set the number of sensor measuring data is \( k \). The estimation of \( R_{pp} \) and \( R_{pq} \) is \( R_{pp}(k) \) and \( R_{pq}(k) \) respectively. Then

\[
R_{pp} = \frac{1}{k}\sum_{i=1}^{k} X_p(i)
\]

\[
= \frac{1}{k}\sum_{i=1}^{k} X_p(i)X_p(i) + X_p(k)X_p(k)
\]  
(17)

\[
= \frac{k-1}{k} R_{pp}(k-1) + \frac{1}{k} X_p(k)X_p(k)
\]

Similarly,

\[
R_{pq} = \frac{k-1}{k} R_{pq}(k-1) + \frac{1}{k} X_p(k)X_p(k)
\]  
(18)

If sensor \( q \) ( \( q \neq p, q=1,2,\ldots,n \) ) can provide correlated calculation with \( p \), we can get the value of \( R_{pq}(k) \). Then the mean value \( \bar{R}_{pq}(k) \) of \( R_{pq}(k) \) can be taken as the estimation of \( R_{pq} \).

\[
R_{pq} = \bar{R}_{pq}(k) = \frac{1}{n-q+1}\sum_{i=q}^{n} R_{pq}(k)
\]  
(19)

Thus, we obtain the estimation value of \( R_{pp} \) and \( R_{pq} \) in the time domain, based on the measuring value of each sensor. Accordingly the variance \( \sigma_p^2 \) of each sensor can be estimated.

C. Application Procedures

In actual application, the computation procedures for adaptive weighted data aggregation are:

1. Compute and \( R_{pq}(k) \) at sampling hour \( k \) with formula 17 and 18

2. Compute \( \bar{R}_{pq}(k) \) at hour \( k \) with formula 19

3. Compute \( \sigma_p^2 \) at hour \( k \) with formula 16;

4. Compute the mean value \( \bar{X}_p(k) \) of sensor \( p \) measured at hour \( k \) with formula 7;

5. Acquire the optimal weighted factor \( W_p \) according to formula 5;

6. Acquire the aggregation estimation value \( \hat{X} \) with formula 8.

In the experiments of adaptive weighted data aggregation, we adopt 10 groups uncorrelated zero mean white noise data to simulate the observing error of each sensor. The variance of white noise is 0.05, 0.07, 0.1, 0.2, 0.3, 0.25, 0.1, 0.1, 0.2, 0.3 and true value \( X = 0.1 \). If \( X \) is added to the white noise, the measuring data of 10 groups of sensors can be simulated. We perform adaptive weighted data aggregation to these measuring data. Each sensor only performs one measurement to the environment in this system. Then we get the results as table 1.

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>( x_5 )</th>
<th>( \sigma_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.05</td>
</tr>
<tr>
<td>( \sigma_i )</td>
<td>( \sigma_i^2 )</td>
<td>( \sigma_i^3 )</td>
<td>( \sigma_i^4 )</td>
<td>( \sigma_i^5 )</td>
<td>( \sigma_i^6 )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( \sigma_i^2 )</th>
<th>( \sigma_i^3 )</th>
<th>( \sigma_i^4 )</th>
<th>( \sigma_i^5 )</th>
<th>( \sigma_i^6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
<td>0.3</td>
</tr>
</tbody>
</table>

From formula 5 we know the optimal weight factor of the first measuring data is

\[
W_i = 1/(\sigma_i^2) \sum_{i=1}^{10} 1/\sigma_i^2
\]

\[
= 1/(0.05 \times 84.952)
\]  
(20)

\[
= 0.235
\]

Similarly we can obtain the weighted factors of other measuring data. The value of each weighted factor in table 2 satisfies formula 2. So we can conclude the weighted factor will be smaller for the data which has bigger error. While the weighted factor will be larger for the measuring data which has smaller error and it has more important status in the testing results.

IV. SIMULATION AND IMPLEMENTATION

A. Performance Analysis based on K-mean Clustering Algorithm

The first experiment is realized on PC configured with VC++6.0 and the OS is Windows XP professional. Other setting is 1G memory, 500G hard disk space and CPU frequency is 2.0 GHz. The multithreading work mode is adopted to simulate the environment of WSN. During the process, DKC method is compared with the centralized method [19] which will compute the \( \sin k \) nodes returned.
by all data. The following figures describe the changing of error rate and time with the increase of the number of network nodes.

![Figure 2. Error rate of DKC.](image)

![Figure 3. Execution time of DKC and centralized method.](image)

Due to the feature of WSN, DKC algorithm may generate data loss in the process of iteration. Therefore, the accuracy of global clustering results may be decreased. To measure the accuracy of the improved method, when the initial conditions of algorithms are equivalent, the same k centroids are given to two algorithms. Then after the algorithms are terminated each particular point will lie in the same cluster of clustering results. We assume there are 100 data points of each node in WSN, which has 5 attributes each, $k = 4$. Figure 3 reflects the changing relation of the percentage that is occupied by error data points in the whole dataset, with the increase of nodes in network. It can be seen that the computation time of our scheme is obviously less than that of centralized method. Moreover, the data transmitted in centralized method is too large to be adaptable for WSN.

**B. Effect on Algorithms Caused by Aggregation Cost**

The aggregation cost will cause some effect on the algorithms, as is referred in the influence by data correlation. In this experiment we set the communication radius of node as 30m, the range for data correlation is 50m. The range of aggregation cost is 10 nJ / bit to 200 nJ /bit. The total energy costs of algorithm are shown as figure 4(a). With the increase of aggregation cost, the energy consumption of MST [20] and centralized method increase rapidly. While DKC method has less energy consumption when the aggregation cost is less than 80 nJ /bit. When the aggregation cost is beyond 80 nJ /bit, the total energy consumption of DKC is obviously optimal.

![Figure 4. The effect on algorithms caused by aggregation cost.](image)
C. Case for Implementation

We take the application of data aggregation in ambient temperature detection as an example, to explain the working process of self-adaptive weighted algorithm. Eight WSN nodes equipped with temperature sensors are sporadically deployed in a room. Its detecting data is \(35.56, 35.72, 35.82, 35.64, 35.77, 35.69, 35.93, 35.65\) (\(^\circ\)C). The variance of each sensor (\(\sigma^2_i, i=1,2,\ldots,8\)) is 0.44, 0.47, 0.52, 0.53, 0.49, 0.68, 0.81, 0.55. As is known from the above, the adaptive weight factors (\(W_p, p=1,2,\ldots,8\)) is 0.184, 0.161, 0.131, 0.126, 0.148, 0.077, 0.056, 0.117. So the aggregation value calculated by formula 8 is:

\[
X = \sum_{p=1}^{8} W_p X_p(k)
\]

\[
= \sum_{p=1}^{8} W_p X_p(k)
\]

\[
= 35.704
\]

Then the ambient temperature measured is 35.704\(^\circ\)C. According to the variance of each sensor the weighted factors can be acquired. The sensor owning higher measuring accuracy has bigger weighted factor. Furthermore, with the increase of times for detection, the weighted factor can be calculated based on the measuring data in each measurement. The importance in data detecting will be reflected by weighted factor. So it can fully take into account the advantage of sensors and the factor of environment, which reduces the influence of larger data deviation and improve the accuracy of measuring system.

V. CONCLUSION

Wireless sensor network has a lot of advantages like lower cost in wiring reduction, high monitoring accuracy, excellent fault tolerance, remote monitoring, easy diagnosis and maintenance etc. It has broad application prospect on network manufacturing and intelligence manufacturing. Its basic task is to correctly acquire valuable information in physical world. WSN depends on time and position information of nodes to realize the control between sensor nodes. We can also get high velocity of sensor data and low delay exchange to guarantee the correctness and real-time of the whole detection in system. However, WSN has some limitations in calculation, storage and network resource. So it is very significant to study the systematic structure of WSN, clock synchronization of nodes, node location and data aggregation.

This paper has discussed deeply the fundamental principles of node localization algorithm in WSN. It also proposes the node data packet method based on K-mean clustering algorithm. In the same packet, the adaptive weighted data aggregation is used to process the network node data. DKC algorithm can realize fast and rational packet of node sensor data in WSN. After nodes sensor data are packet, simple and effective adaptive weighted data aggregation methods can be applied to complete in-network data aggregation, which has been tested in a prototype system. The experiment results show that this method is simple and feasible. It largely reduces the data redundancy in the whole network and saves amounts of storage resource and the network bandwidth.

There are still a lot of aspects to improve our research and the next job mainly includes:

- The environmental factor of WSN is a big problem to the effectiveness of algorithm. It is necessary to analyze and study real-time and reliability of the algorithm.
- Since this paper do not explain many relevant routing methods in WSN, it is necessary to be studied in the future job.
- As the energy effectiveness, calculation resource, storage resource of WSN nodes have large limitations, it needs related research in detail to obtain better performance in energy saving. Related specifications and performance standards also need further research.

ACKNOWLEDGEMENTS

The study was supported by the Natural Science Foundation of China (61071076).

REFERENCES


